3-2007

Approximate Analysis of an Unreliable M/M/2 Retrial Queue

Brian P. Crawford

APPROXIMATE ANALYSIS OF AN UNRELIABLE M/M/2 RETRIAL QUEUE

THESIS

Brian P. Crawford, 1 Lt, USAF

AFIT/GOR/ENS/07-05

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the United States Air Force, Department of Defense or the United States Government.

AFIT/GOR/ENS/07-05

APPROXIMATE ANALYSIS OF AN UNRELIABLE M/M/2 RETRIAL QUEUE

THESIS

Presented to the Faculty

Department of Operational Sciences

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

in Partial Fulfillment of the Requirements for the

Degree of Master of Science in Operations Research

Brian P. Crawford, B.A.

1 Lt, USAF

March 2007

AFIT/GOR/ENS/07-05

APPROXIMATE ANALYSIS OF AN UNRELIABLE M/M/2 RETRIAL QUEUE

Brian P. Crawford, B.A.

1 Lt, USAF

Approved:

Dr. Jeffrey P. Kharoufeh Thesis Advisor	Date
Dr. Sharif H. Melouk Committee Member	Date

AFIT/GOR/ENS/07-05

Abstract

This thesis considers the performance evaluation of an M/M/2 retrial queue for which both servers are subject to active and idle breakdowns. Customers may abandon service requests if they are blocked from service upon arrival, or if their service is interrupted by a server failure. Customers choosing to remain in the system enter a retrial orbit for a random amount of time before attempting to re-access an available server. We assume that each server has its own dedicated repair person, and repairs begin immediately following a failure. Interfailure times, repair times and times between retrials are exponentially distributed, and all processes are assumed to be mutually independent. Modeling the number of customers in the orbit and status of the servers as a continuous-time Markov chain, we employ a phase-merging algorithm to approximately analyze the limiting behavior. Subsequently, we derive approximate expressions for several congestion and delay measures. Using a benchmark simulation model, we assess the accuracy of the approximations and show that, when the algorithm assumptions are met, the approximation procedure yields favorable results. However, as the rate of abandonment for blocked arrivals decreases, the performance declines while the results are insensitive to the rate of abandonment of customers preempted by a server failure.

Acknowledgements

In completing the research contained herein, I am indebted to several individuals who assisted me along the way. First, I would like to thank Dr. Jeff Kharoufeh, my advisor, for his patience and willingness to guide me throughout the research process. He has been a continuous source of motivation and drive. I am grateful for his strict attention to detail and have the utmost respect for his mathematical ingenuity. I would also like to thank my reader, Dr. Sharif Melouk, for providing comments on the final document and for his invaluable assistance with regards to simulation in Arena. Additionally, I am grateful to my fellow students Ryan Harrell, Jonathan Hudson and Benjamin Kallemyn for their unselfish assistance despite being busy with their own research. Last of all, but definitely not least, this work could not have been accomplished had it not been for the support of my beautiful wife and our three children. They have been a source of inspiration and joy during an otherwise demanding and challenging period of my life. I will be eternally grateful for their many sacrifices.

Brian P. Crawford

Table of Contents

List of Figures

List of Tables

APPROXIMATE ANALYSIS OF AN UNRELIABLE
M/M/2 RETRIAL QUEUE

1. Introduction

1.1 Background

In recent years, military planners have sought to gain a war-fighting advantage by quickly gathering information about enemy whereabouts and their objectives. The ability of military forces to obtain critical information on enemy objectives before the enemy can do the same is termed *information superiority*, the achievement of which may result in a swift victory with minimal loss of life. To facilitate the sharing of information between key players, elaborate communication networks are required that employ several types of transmission mediums to include computer networks, audio and video transmitters (on land, sea or in the air), and satellites to name a few. Such networks must also have the ability to accommodate multiple data types including standard text, audio and video. Effective information sharing through these types of network configurations is critical for implementing the concept of network-centric warfare (NCW).

Miller [32] defines NCW as the "conduct of military operations through the utilization of networked information systems, which supply the war-fighter with the right information at the right time...." The ability to gather the correct information and share it in a timely manner is the objective of NCW. The proper implementation of NCW leads directly to the attainment of information superiority, which in turn provides the war-fighter and commander with shared situational awareness aiding in successful mission completion.

Understanding the movement of information through information processing networks is helpful to the process of modeling the flow of data from sender to receiver. Multiple transmission mediums are necessary to accurately and quickly process the flow of information as it relates to NCW. Each of the transmission mediums may be susceptible to disruptions due to traffic congestion, the effects of weather, damage due to enemy fire or mechanical failure. Consider, for example, that the transmitted information takes the form of time-sensitive data packets that arrive randomly to various transmission mediums. If a medium is busy transmitting other data packets, or experiences a disruption, then the arriving data packet is delayed. Furthermore, a medium that fails during transmission also results in delayed data packets (e.g. packet collision on a shared medium). Depending on the importance and time-sensitivity of the information, it can be retransmitted later or possibly dropped altogether. For example, it is possible that the location of a terrorist group is known for the next few minutes. If an attack message is delayed, the location of the group may change by the time retransmission occurs. In this case, the message could be rendered useless. Naturally, retransmission can only occur after the disrupted medium is again operational and available.

Many real-world, stochastic service systems, including the aforementioned information sharing network, may be modelled as unreliable retrial queueing systems with multiple servers. Some of these include cellular telephone networks, computer networks, and customer contact centers (e.g., customer call centers, email centers, etc.). These centers employ multiple operators to fulfill customer service requests with a quality of service guarantee. However, if service is not initiated in a timely manner, customers may choose to abandon their requests. Furthermore, the service may be interrupted by random events such as network congestion, misdirected (or dropped) calls, mechanical failures or other unforeseen circumstances. These can all lead to customer dissatisfaction which may result in customer abandonment following a disruption. Because service organizations are most interested in providing a

high level of customer satisfaction, it is imperative that such systems be analyzed in order to evaluate their performance and to provide a means by which they may be optimally designed, staffed, and operated.

The retrial queueing literature contains a significant amount of work devoted to the performance analysis of unreliable single-server retrial systems. However, to the best of our knowledge, the multi-server case has remained relatively unexplored. The primary objective of this research is to formally analyze an unreliable, two-server retrial queue. As will be shown in Chapter 3, an exact analysis of such a system is complex and possibly intractable. For this reason we shall focus our attention on an approximate analysis.

1.2 Problem Definition and Methodology

Consider an unreliable M/M/2 retrial queueing system. Arriving customers who find both servers busy or failed are given the choice to abandon their service request or enter a retrial orbit. We assume that a server can breakdown when active or idle. Should a failure occur while a customer is in service, the customer is given the option to depart the system or proceed to the retrial orbit. We also assume that preempted customers, once able to regain access to a server, repeat their service requests.

Multi-server retrial queueing systems, in general, are difficult to analyze from a mathematical standpoint. Exact results for the steady-state probabilities of reliable systems are given only for the single and two-server cases. In the unreliable model, there are no exact solutions when the number of servers exceeds one. Therefore, we seek to approximate the steady-state joint distribution of the number of customers in orbit and the status of the two servers for the case of Markovian arrival and service times. We also provide approximate expressions for several queueing performance measures. Our approach to deriving the approximate steady-state probabilities em-

ploys a phase-merging algorithm outlined by Korolyuk and Korolyuk [23]. The algorithm is useful for the analysis of general, two-dimensional continuous-time Markov chains.

Due to the scarcity of analytical results for unreliable, multiple-server retrial queues, the queueing research community can benefit from the results of this thesis. These systems are difficult to analyze using standard methods familiar to queueing researchers. In lieu of exact results, approximation procedures are often employed to study the steady-state behavior of such systems, and this is the approach we employ here. It is our hope that the model and approximation algorithm will stimulate future work in this branch of queueing theory.

The results of this thesis may also benefit the military analysis community in the area of NCW. Nearly every military organization uses computer networks to share information. For example, email has overwhelmingly become the default method of communication. Live streaming audio and video applications are used extensively in military operations to include simple meetings, conferences, and most critically, war-fighting. As in the private sector, the military also maintains and operates customer contact centers that provide an essential link to military personnel worldwide. Unreliable multi-server retrial queues may potentially be used to model all of these systems and lend much needed insight to their optimal design and operation.

1.3 Thesis Outline

The next chapter introduces a substantial portion of the retrial queueing literature, covering both reliable and unreliable systems. A section is also devoted to those who first considered standard queueing models with servers subject to breakdowns. In chapter 3 we provide the formal model description and state the assumptions that are needed to implement the approximation procedure. The algorithm is then

formally reviewed and illustrated with an example. Applying it to our model, we derive approximations for the steady-state probabilities and several standard queueing performance measures. In chapter 4, we assess the quality of the approximations by comparing results with those obtained using a discrete-event simulation model. In chapter 5, we summarize the main contributions of the thesis and provide some concluding remarks regarding the effectiveness of the approximation. Finally, some ideas for future research are suggested that might further advance the field of unreliable retrial queueing systems.

2. Relevant Literature

This research analyzes a multi-server retrial queue with servers that breakdown. Considerable work has been done in the analysis of single-server retrial queues as compared to multi-server models. With regard to failures, the open literature contains a substantial amount of work that deals with normal queueing systems with single or multiple servers that are subject to breakdowns. Comparatively, results for retrial queueing systems with server breakdowns are not as abundant. The case of multiple server retrial queues with breakdowns is even more sparse. In this chapter we first review the literature pertaining to single and multi-server retrial queues with no breakdowns. Subsequently, standard queues with server breakdowns are explored. This is followed by a review of results for retrial queues with breakdowns for both the single-server and multiple-server cases.

2.1 Retrial Queues

Retrial queueing systems differ from conventional queueing systems in that customers arriving to a server station and finding all servers unavailable enter a retrial orbit (or source of repeated calls) instead of a normal queue. They remain there for a random amount of time (usually exponentially distributed), and then check to see if a server is available. If a server is available, they enter service immediately; otherwise they return to the orbit and wait again. In the meantime, a new or primary call can arrive to the system and obtain service if a server is free. Unlike a normal queue, the retrial orbit generally has no queueing discipline, and thus a customer that exits the orbit can be viewed as the winner of a competing event. In some cases, retrial queues are assumed to have a normal queue in addition to the retrial orbit. For example, an M/M/1/k retrial queueing system has one server and a waiting room of size $k-1$. Most retrial queues in the literature, however, assume no additional

waiting space for customers. In the literature, retrial queues are also referred to as queues with *repeated attempts*, *repeated calls* or queues with *returning customers*.

Retrial queueing models have been used in analyzing and designing many types of systems. A few of these include telephone-switching systems (e.g. customer contact centers), telecommunication networks such as cellular phone networks and computer networks and systems. Historically, an interest in retrial queues emerged in the study of telephone traffic theory and several papers are devoted to this theory beginning in the late 1940s to include Clos [11], Wilkinson [44] and Cohen [12]. These early researchers focused on the distribution of the number of busy trunks (lines of a telephone system) and customer behavior when an "all-trunks-busy" signal was obtained. It was found that many customers who get a busy signal persist and retry their call until it can be completed. Thus an interest in the distribution of times between retrials was generated. In [12], Cohen proposed a main problem of telephone traffic theory and what was needed to solve it. He discovered the following items were essential: the number of callers who subscribe to the phone service and their arrival time distribution, the distribution of the duration of calls, the behavior of callers who find the system busy; and the manner in which calls are handled by the telephone lines. The basic problem is to determine the distribution of the number of busy trunks, the probability that all trunks are busy and the number of lost calls. Defining a bivariate stochastic process consisting of the number of busy trunks and the number in orbit, Cohen derived steady-state probabilities via a stochastic birth-and-death process.

A decade later Keilson, Cozzolino and Young [21] examined both the M/G/1 retrial queue and the M/M/2 system. For the M/G/1, customers finding the server busy enter an orbit and spend an exponential amount of time there and try the server again independently of all others. Because the service times are assumed to be generally distributed, the authors use the method of supplemental variables to transform the stochastic process to a Markovian one. Cox [14] first developed

this methodology which is widely used when general service time distributions are assumed. The authors make use of generating functions to calculate the mean queue length, the mean waiting time of a customer, and the number of calls per customer. The M/M/2 case is solved by means of writing flow balance equations and solving them using a normalization equation.

In 1987, Hanschke [20] solved the same flow balance equations resulting from a M/M/2 retrial queue using hypergeometric differential equations. He then calculated the probability of blocking along with the mean length of the orbit. An example of a multi-server retrial queue studied most recently was by Abramov in 2006 [1] in which customers arrive according to a general renewal process with m servers whose service time is exponentially distributed. The time between retrials is also exponentially distributed. Using a martingale approach, the author establishes stability conditions and studies the behavior of the limiting distribution of the queue length as the retrial rate approaches infinity.

Another noteworthy contribution is that of Kulkarni [24] who considered an M/G/1 queueing system with retrials and two types of customers arriving according to a Poisson process with distinct rates. Kulkarni proved that the mean arrival rate times the average number of unsuccessful retrials is equal to the mean service completion rate times the average number of unsuccessful retrial attempts during one service period. He then used this result to compute the expected number of retrial customers of each type, the expected number of retrials conducted by each type, and the expected number of customers in the system of each type.

Since the late 1980s the most important results can be found in Yang and Templeton [47], Falin [16], Kulkarni and Liang [26]. In 1997 Falin and Templeton [17] contributed an excellent text providing substantial analysis on many various retrial queues. Their analysis includes a lengthy section devoted to multi-server models. They give some results for the M/M/c model, but as of the writing of this thesis, no closed form solution exists for the steady-state probabilities for $c > 2$. An

extremely useful bibliography was contributed by Artalejo [9] who provided a list of 163 references on retrial queues.

2.2 Queueing Systems with Breakdowns

The first work done in the area of queueing systems with breakdowns was by White and Christie [43] in 1958. They examined a multi-class M/G/1 queue in which customers arriving to the system who have a higher priority than any other customer in the system immediately receive service, thus preempting the customer currently in service. They showed that a breakdown can be equivalent to these types of customer arrivals with preemptive priority. Their model assumed preempted customers rejoin the queue at the head of the line.

Thiruvengadam [40] considered an M/G/1 system with breakdowns arriving according to a Poisson process and generally distributed repair times. Three models were examined in that paper. The first assumed that a queue of breakdowns can exist. That is, one or more breakdowns can occur even when the server is under repair. Service resumes after all the breakdowns are repaired. The second model assumed that a queue of breakdowns is not permissible and that the server is subject only to active or idle breakdowns. The third model assumed that idle breakdowns cannot occur. For each model, the expected number of breakdowns and the expected number of customers in the system are derived. In models two and three the author used Laplace transforms to derive generating functions for the steady-state probabilities.

Avi-Itzhak and Naor [10] extended the work of White and Christie [43] by investigating five models (labeled A-E) of an M/G/1 system with server breakdowns. Model A considered active and idle breakdowns while Model B was concerned with active breakdowns only. Model C assumed that a failed server begins the repair process only when customers are present in the system. Model D is unique in that a breakdown can be initiated by a customer who requests the server be repaired

so as to improve service. Model E simply assumed that only idle breakdowns be considered. Using conditional arguments, the authors calculated the expected queue length and other operating characteristics.

Avi-Itzhak and Mitrany [33] extended the model studied by White and Christie [43], Thiruvengadam [40] and Avi-Itzhak [10] to include multiple, independent servers. This is one of the first works to consider such a system in which the servers are subject to breakdowns. The authors studied a M/M/N queueing system with customers preempted by a server breakdown returning to the head of the queue. Using generating functions the expected number of customers in the system was derived for the cases $N = 1$ and $N = 2$. For $N \geq 2$, numerical methods were discussed.

In 1979, Neuts and Lucantoni [34] revisited the M/M/N queue and considered the addition of c ($c < N$) repair crews where one repair crew is assigned to fix a single server breakdown. They noted that the number of failed servers may exceed the number of repair crews resulting in the formation of an additional queue. The authors focused on an algorithmic approach using matrix-analytic techniques to approximate the steady-state probabilities and stationary waiting time distributions. Additionally, they investigated the effect of reducing the number of repair crews and the effect of reducing the arrival rates during a server failure.

Sztrik and Gal [38] studied a single server system with breakdowns in which entities are viewed as jobs created by terminals that arrive according to a Poisson process at a CPU. The terminals are subject to failures just as is the CPU; however the rate at which jobs arrive to the CPU is still Poisson. All service, repair and times to failure are assumed to be exponentially distributed and breakdowns are serviced by r repair crews, thus creating a second queue, that of failed terminals. The authors defined a trivariate stochastic process as follows: $X(t) = 1$ if system is operational at time t, 0 otherwise, $Y(t)$ is the number of jobs at the CPU at time t and $Z(t)$ is the number of failed terminals at time t. They then proceeded to recursively solve the steady-state equations and calculate the mean number of jobs at the CPU, mean

number of operational terminals, average number of busy repair persons as well as server utilizations of the CPU, terminals and repair persons.

Queues with server breakdowns have been studied extensively in the past decade. What follows are a few papers worth notable mention. In 1997, Wei, et al. [31] considered a M/G/1 queue with server breakdowns and vacations. Besides assuming the service time to be generally distributed, the authors provided a reliability analysis of the system. Using the supplementary variable technique they derived transient solutions for standard queueing and reliability measures. In the same year, Tang [39] also considered a M/G/1 queue in which the server was subject to both active and idle breakdowns. The inter-failure times for active breakdowns followed a Poisson process while the inter-failure times for idle breakdowns followed a generic renewal process. Repairs occur immediately and are generally distributed. Preempted customers hold the server during the repair and resume service once the repair is complete. Using transform methods the author derived several queueing measures as well as some main reliability indices.

In 2002, Gray, et al. [19] studied two models that both employed backup servers. For each model the authors considered two cases, the first case allowing for only active breakdowns, and the second allowing for both active and idle breakdowns. The first model assumed two ranked servers, a primary server and a backup server. The second model assumed an infinite amount of identical, unranked servers. All service times are assumed to be exponential. The inter-arrival times are also exponentially distributed, however, if all available servers are failed the arrival rate changes. For Model I, the servers may have different rates and in Model II the authors assumed homogeneous servers. Using matrix-geometric techniques the authors derived the distribution of the queue length and stability condition for each model.

In 2003, Yuan and Li [46] considered a GI/PH/1 queue with server breakdowns. For their model they assumed customers who were interrupted by a failure remain at the server and resume service immediately following the repair. Just as the service

time, the repair time also followed a phase type distribution. Using matrix-geometric methods, the authors derived the condition for system stability and analyzed the transient and steady-state behavior.

2.3 Single-Server Retrial Queues with Server Breakdowns

Retrial queues in which the server is susceptible to failures presents an additional way for an arriving customer to enter the retrial orbit. If a customer finds the server either busy or failed it must enter the orbit and attempt to access the server later. A customer is admitted to the server only when the server is found idle and not failed. Customers whose service is interrupted by a failure may have the option of remaining at the server until the repair is complete, leaving the system entirely, or returning to the orbit to repeat or resume service.

These types of systems were first studied independently by Aissani [2] in 1988 and by Kulkarni and Choi [25] in 1989. Aissani [2] considered an M/G/1/1 queueing system with repeated orders and an unreliable server while Kulkarni and Choi considered two different M/G/1 models. In the first model, a customer whose service is interrupted by a server failure either joins the retrial orbit with probability c or leaves the system with probability $1 - c$. The second model allows the customer to remain at the service station while the server is being repaired and service is restarted once the repair is complete. The latter model can be solved using the results of the former. In the first model, the authors assumed that a server, at any time, can be in one of the following three states: idle-up (0), busy (1), or down (2). An idle-up server fails at an exponential rate and stays down for a random amount of time, D_i. A busy server fails at an exponential rate and its random down time is denoted by, D_b. A customer who cannot obtain service enters the orbit and retries after an exponential amount of time. The limiting behavior of the stochastic process, $\{(Q(t), X(t)), t \geq 0\}$ where $Q(t)$ is the random number of customers in the orbit and $X(t)$ is the state of the server, is studied as $t \to \infty$.

Aissani and Artalejo [5] extended the results of Kulkarni and Choi [25] in 1998 by focusing on the reliability of an M/G/1 system when the server is subject to breakdowns. The pair considers the model in which customers that arrive and find the server busy are sent to the orbit while customers that access the server and are then interrupted by a breakdown either join the retrial orbit with probability c or leave the system with probability $1 - c$. Customers who join the orbit retry after an exponential amount of time and those who were interrupted retain no memory of being served. It was assumed that the time between both active and idle breakdowns is exponentially distributed with distinct rates. Repair times, however, were assumed to be generally distributed. They then define a variable, F, which is the random amount of time from the epoch at which a customer begins service to the epoch at which the server is able to begin a new service time (note that this could apply to the same customer). This period of time is referred to as the *fundamental server period*. The duration of F is determined by the competition between service time and failure time. Another concept that the authors introduce is an auxiliary queueing system where a customer interrupted by a failure can hold the server and resume service after the repair is complete. The option exists, however, for the customer to leave the server station and enter the orbit. By investigating the embedded Markov chain at idle-up epochs, the authors provided a stability condition and then proceeded to analyze the system with generating functions and a recursive scheme to compute the limiting probabilities.

Aissani [3] continued his work on the M/G/1 retrial queue this time making more general assumptions. Arrivals to the system are according to a batched Poisson process with all members of the batch moving to the retrial orbit if the server is busy or failed. If the server is idle then one unit of the batch is admitted to the service area where it is processed according to a general distribution and the remaining join the orbit. Additionally, times between retrials are generally distributed and the inter-failures times of the server are dependent upon the state of the server. Repair times

are also generally distributed with the time to repair depending on the status of the server at the time of failure (idle or busy). Aissani used the method of supplemental variables to transform the jump process (or the random number of customers in the system at time t) to a piecewise Markov process and proved stationarity of the process. The author then proceeded to derive the steady-state distributions of both the number of customers in orbit and the number of customers in the system. Aissani [4] again visits the M/G/1 system this time assuming that customer retrials, times to failure and repair times are distributed exponentially. The other difference is that he considered a warm back-up server in case the primary server fails with the assumption that the repair of the primary server takes place during the busy time of the substitute. This assumption leads to a system that never fails. Using the same techniques in [3] he derived similar performance characteristics.

Many different variations of the unreliable M/G/1 retrial queue exist in the literature and Yang and Li authored two works [48] and [29] that further investigated the system. In [48] customers arriving to the system who find the server idle are admitted to service and "turn on" the server which can operate normally with certain probability or fail, thus forcing the customer to join the retrial orbit. This type of failure is referred to as a starting failure in the literature. Assuming retrial times are exponential and repair times are generally distributed, the authors presented a necessary and sufficient condition for system stability and derived (making use of probability generating functions) the server utilization, average number of customers in the system and the steady-state probability that the server is down. The second paper [29] assumes a finite number of sources that can be active or inactive. Active sources generate customers according to a Poisson process. The source subsequently becomes inactive and is activated again after the customer completes service. Servers are in one of three states: idle, busy, or on vacation. Customers finding the server busy or on vacation leave the system and retry later. When the server is idle, it serves new or returning customers with probability α_k or takes a vacation with probability

$1 - \alpha_k$, where k is the number of customers in the orbit. If the server takes a vacation, the arriving customers proceed to the orbit. Retrial times are exponentially distributed and service and vacation times are generally distributed. The authors then examined the system in its steady-state using the method of supplemental variables and generating functions and derived server utilization, mean number of customers in the system and the mean time each customer spends in the system. Artalejo [8] also studied M/G/1 retrial queues with server vacations and Kumar, et al. [27] in 2002 considered the M/G/1 with Bernoulli feedback[1] and starting failures. Interestingly, the authors assumed an orbit with an FCFS discipline with waiting time generally distributed.

Other M/G/1 retrial queues were studied by Wang, et al. [41] in 2001 where the authors considered a customer who waits at the server during repair. They defined this period of time as *generalized service time* which may or may not include repair time. Besides calculating the traditional steady-state characteristics they also provide a detailed reliability analysis of the system. Four years later in [42] Wang performed the same analysis for an M/G/1 retrial queue under the assumptions that the retrial orbit has an FCFS discipline and that an idle server searches for customers in the orbit. The search time is generally distributed and if a primary call arrives to the system, the search is interrupted and the primary caller begins service. Djellab in 2002 [15] studied a model similar to that of Kulkarni and Choi [25], but assumed general distributions for times to failure and repair times.

In 2003, Wu, et al. [45] were the first to consider two retrial orbits in their M/G/1 system. The first orbit (I) is in the traditional sense with an FCFS discipline. The second orbit (II) is reserved specifically for customers preempted by a server failure. Repair times and retrials from orbit (I) are generally distributed while retrials from orbit (II) are distributed exponentially. The authors also assumed that

[1] A system with feedback allows customers that have been served an opportunity to return to the system if not satisfied.

customers retain accrued service time throughout the model. Balking is also considered, that is, customers have the option of leaving the system when assigned to orbit (I) or orbit (II). The customer may also choose to leave (non-persistent) while attempting retrials in orbit (I) while customers in orbit (II) remain persistent since they have already completed some amount of service. The authors also assumed only active breakdowns can occur. Additionally, a server that fails is repaired immediately and must complete service for the preempted customer before any new customers are allowed service. The time between repair completion and preempted customer service resume is known as *reserved time*.

In 2006 Li, et al. [28] extended the work of [41] by examining a system in which customers arrive according to a batched Markovian arrival process (BMAP) with m phases. The authors considered a single server whose service times are generally distributed with exponential times to failure and generally distributed repair times. A customer whose service is interrupted by a failure remains at the server until the repair is complete. Thus the idea of "generalized service time" was employed throughout the work. Using the method of supplementary variables and matrix-analytic techniques, the authors derived the standard queueing and reliability indices. They also developed two algorithms, the first to compute the stationary probability vector of a M/G/1 continuous-time level-dependent Markov chain, and the second to calculate the mean of the first passage time with regard to this M/G/1.

Not all customers arrive to a queueing system according to a Markov process. In 2003 Yuan and Li [49] investigated the effect that generally distributed interarrival times and non-exponential service times have on the availability of the server. In their study of a GI/PH/1 system with server breakdowns, the authors assumed that inter-failure times were exponential and repair times follow a phase-type (PH) distribution. Just as Wang, et al. did in [41], customers preempted by a server failure wait at the station until the server is repaired, and then resume service once the repair is complete. The authors used matrix analysis theory to derive certain

performance characteristics to include steady-state probabilities that the server is busy, being repaired or idle. They also developed formulas for the availability of the system.

Recently, some authors have investigated M/M/1 retrial systems with breakdowns. In 2005, Almasi, Roszik and Sztrik [6] considered a finite source system with server failures and repairs. Servers can fail either in a busy or idle state and do so with different probabilities. Times to failure are generally distributed with repair times exponentially distributed. Customers preempted by a failure can choose to remain at the server and resume service once the repair is complete, or join the retrial orbit. Retrial entities retry at an exponential rate. In deriving the usual stationary measures Almasi, et al. [6] used a tool called MOSEL (Modeling, Specification and Evaluation Language). Later that year, Li and Zhao [30] assumed a M/M/1 system with two queues both for waiting or preempted customers. All times in their model are distributed exponentially and only active breakdowns are considered. Customers preempted by a server failure join a normal queue at the head position and arriving customers who find the server busy or failed join the normal queue with probability p or the retrial orbit with probability $1-p$. The retrial orbit assumes an FCFS discipline. Retrial customers unable to access the server can join the orbit again with probability q or leave the system (impatient) with probability $1-q$. The authors model the process as a (quasi birth-and-death process) QBD, and used a matrix-analytic approach to prove that the system decays geometrically.

Sherman and Kharoufeh [37] considered an unreliable M/M/1 retrial queue with an infinite waiting room and retrial orbit. In their model, customers preempted by server failures join the orbit while the normal queue is reserved only for new arrivals. All times between events are assumed to be exponentially distributed and active and idle breakdowns can occur. The authors give the steady-state joint distribution of the orbit length and queue length for each state of the server (idle, busy or failed) and derived generating functions for orbit size, queue size and total system

size. Lastly, they proved stochastic decomposability of the orbit and system size and provided standard queueing performance measures.

2.4 Multiple-Server Retrial Queues with Breakdowns

Consider a retrial queue with c servers. Customers arriving to a system and finding the total number of busy servers and failed servers equal to c must enter the orbit and attempt to obtain service later. If the total is less than c, then the customer enters service and is processed according to the service rate. These types of systems are not found in abundance in the open literature. In 1994 Artalejo [7] was the first to consider such a model. The author examined a M/M/c/k retrial system in which the server is subject only to active breakdowns. Preempted customers proceed to the orbit with probability H_o or depart the system with probability $1 - H_o$. The author then defined a persistence function that assigns to retrial customers a probability of staying in the system based on the number of retrials they have performed. Sufficient conditions for the ergodicity of the system are proved and the rest of the paper is devoted to analysis of the M/M/1 and M/G/1 systems. For the M/M/1, the author introduced two new measures, the orbit idle and orbit busy periods, derived their distributions, and examined asymptotic behavior. In the M/G/1 system he employed a recursive scheme to calculate steady-state probabilities for the number of customers in orbit, number of customers in service and the number of operational servers.

In 2004 Roszik and Sztrik [35] extended their work in [6] by investigating a finite source retrial queue with multiple unreliable servers that have distinct (heterogeneous), exponentially distributed service times. Additionally, the servers have distinct times to failure which are exponentially distributed and distinct exponential repair times. The authors assumed both active and idle breakdowns with preempted customers becoming a source of repeated calls to the system. With the assistance of the software tool MOSEL, a stochastic Petri Net package was used to calculate the probability that at least one server is idle, the mean orbit length, utilization of the

kth server, mean number of busy servers, mean customer wait time and mean time in system, to name a few. Using tools in MOSEL, they illustrated graphically the effect unreliable servers have on mean time in system.

In 2005 Gharbi and Ioualalen [18] studied a finite-source retrial system with multiple servers subject to breakdowns and repairs. In addition to assuming active and idle breakdowns, the authors include a dependent breakdown scenario in which the probability of failure depends on the server state. Customers preempted by a failure return to the orbit with a memory of their elapsed service time. The authors used a generalized stochastic petri nets (GSPN) model to derive several performance and reliability indices, some of which are the mean length of the orbit, mean number of customers in the system, the mean number of failed and operational servers, mean rate of service and repair and the failure frequency of busy and idle servers. Lastly, a sensitivity analysis of the mean time in system is conducted when rates of failure, repair and retrial as well as the number of servers vary.

These three works are the only ones found in the open literature addressing multiple-server retrial queues with server breakdowns. To our knowledge, outside of Artalejo's ergodicity proof [7], no other analytical methods are available. An analytical solution to the steady-state probabilities of unreliable multi-server retrial models are extremely difficult to derive. As mentioned previously, no results exist for models with more than two servers in the reliable case. Unreliable models contribute even more to the analytical complexity mainly due to preempted customers joining the orbit. As such, it is not surprising to see that the two sources, [35] and [18] resort to computer-aided solution methods.

It is evident that the literature is lacking with respect to modeling multiple-server retrial queues with server breakdowns. With the exception of using petri nets, no other approximation methods have been employed in the study of such models. Therefore, new methods for analyzing these types of systems are needed. In an effort to further understand the complexity of these systems, this thesis attempts to

contribute to the current state-of-the-art by proposing another method for approximating the steady-state probabilities of unreliable retrial queues in the two-server case. This approximation will be completely analytical, and may lend insight into the analysis of unreliable multi-server retrial queues with an arbitrary number of servers.

3. Model Description

In this chapter we describe the M/M/2 retrial queue in which both servers are subject to breakdowns and repairs. Arriving customers that are unable to access a server due to congestion or failure can choose to enter a retrial orbit for an exponentially distributed amount of time and persistently attempt to gain access to a server, or abandon their request and depart the system. Once a customer is admitted to a service station, they remain there for a random duration until service is complete and then depart the system. However, if the server fails during service, i.e., an active breakdown, the customer may choose to abandon the system or proceed directly to the retrial orbit while the server begins repair immediately. Many models in the literature explore cases in which the preempted customer has a choice between joining the orbit or abandoning the system, or remaining at the server until the repair is complete. The server can also fail while it is idle i.e., an idle breakdown. This thesis analyzes a two-server system in which both servers are subject to both active and idle breakdowns.

3.1 Model Description

The model is an unreliable M/M/2 retrial queueing system in which customers arrive according to a Poisson process with rate λ ($\lambda > 0$). If at least one of the servers is idle and not failed, then an arriving customer occupies a server immediately. However, if an arriving customer finds no available servers (due to congestion or failure), the customer enters the orbit with probability q_a or abandons the system with probability $1 - q_a$, $0 \leq q_a \leq 1$. Recall that there is no additional waiting space in a standard retrial queue. Customers who enter the orbit wait for an exponentially distributed time with rate θ ($\theta > 0$) before attempting to access a server again. The service times are assumed to be exponential with mean $1/\mu$. Failures for both servers occur independently via a Poisson process with rate ξ ($\xi > 0$) and the repair times

for each server are exponentially distributed with rate α ($\alpha > 0$). It is assumed that each server has a dedicated repair person. Furthermore, interarrival times, service times, retrial times, interfailure times and repair times are mutually independent. This model accounts for both active and idle breakdowns. For active breakdowns, the customer that is preempted by a server failure enters the retrial orbit with probability q_f or abandons the service request with probability $1 - q_f$. Customers are lost if they decide not to join the orbit. Figure 3.1 provides a pictorial representation of the system.

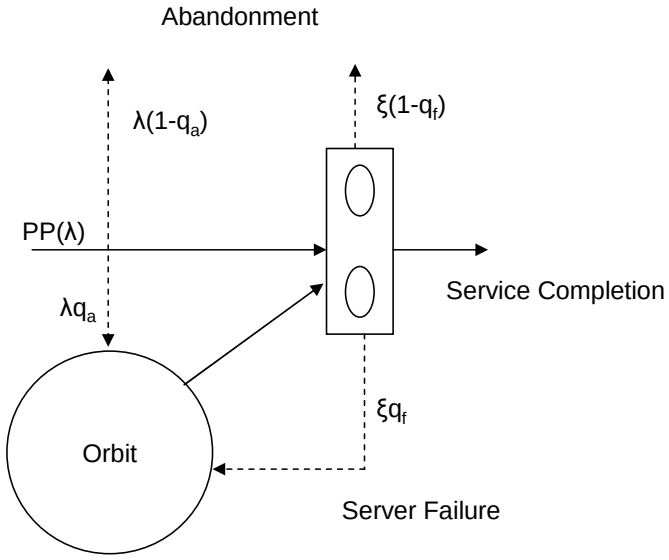

Figure 3.1 Retrial queueing system with two unreliable servers.

The state of the system can be described by a trivariate stochastic process in continuous time, $\{(R(t), B(t), F(t)) : t \geq 0\}$, where $R(t)$ is the number of customers in the orbit at time t, $B(t)$ is the number of busy servers at time t and $F(t)$ is the number of failed servers at time t. Since all the random times are exponentially distributed, the stochastic process is a continuous-time Markov chain (CTMC) on the state space $S = \{(i, j, k) : \ i \geq 0, \ j + k \leq 2, \ j, k \in \{0, 1, 2\}\}$. We assume that as $t \to \infty$ the steady-state distribution of $\{(R(t), B(t), F(t)) : t \geq 0\}$ exists. Figure 3.2 depicts the transition diagram for the CTMC. The levels directly correspond to

the size of the orbit. The ordered pairs represent the number of busy servers and number of failed servers, respectively.

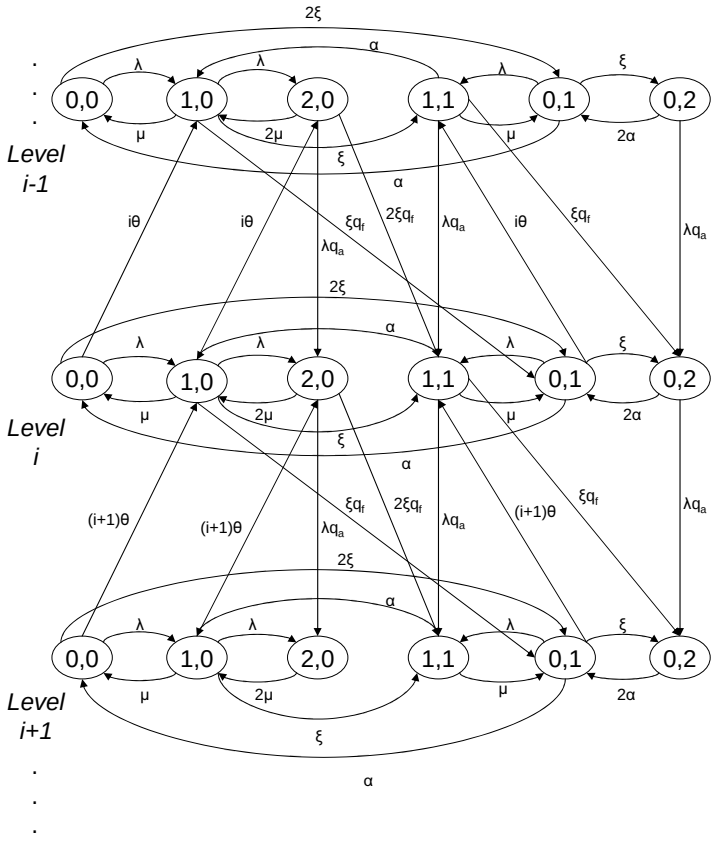

Figure 3.2 Transition rate diagram for retrial queue with two unreliable servers.

Define $p(i, j, k)$ as the limiting probability that the system is in the state (i, j, k) where $(i, j, k) \in S$. Defined mathematically,

$$p(i, j, k) = \lim_{t \to \infty} P(R(t) = i, B(t) = j, F(t) = k).$$

Note that a set of only six ordered pairs of (j, k) are needed to completely characterize the status of the servers at any time. This set is,

$$E = \{(0, 0), (0, 1), (0, 2), (1, 0), (1, 1), (2, 0)\}.$$

Analyzing the flow in and out of each node of Figure 3.2, the following balance equations, boundary condition and normalization equation are obtained. For $i = 0$,

$$(\lambda + 2\xi)p(0,0,0) = \mu p(0,1,0) + \alpha p(0,0,1),$$

and for $i \geq 1$,

$$\begin{aligned}
(\lambda + i\theta + 2\xi)p(i,0,0) &= \mu p(i,1,0) + \alpha p(i,0,1) \\
(\lambda + i\theta + \xi + \xi q_f + \mu)p(i,1,0) &= \lambda p(i,0,0) + \alpha p(i,1,1) + 2\mu p(i,2,0) \\
&\quad + (i+1)\theta p(i+1,0,0) \\
(\lambda q_a + 2\mu + 2\xi q_f)p(i,2,0) &= \lambda q_a p(i-1,2,0) + \lambda p(i,1,0) + (i+1)\theta p(i+1,1,0) \\
(\lambda q_a + \mu + \xi q_f + \alpha)p(i,1,1) &= \lambda q_a p(i-1,1,1) + \lambda p(i,0,1) + (i+1)\theta p(i+1,0,1) \\
&\quad + \xi p(i,1,0) + 2\xi q_f p(i-1,2,0) \\
(\lambda + i\theta + \xi + \alpha)p(i,0,1) &= \mu p(i,1,1) + \xi q_f p(i-1,1,0) + 2\xi p(i,0,0) + 2\alpha p(i,0,2) \\
(\lambda q_a + 2\alpha)p(i,0,2) &= \lambda q_a p(i-1,0,2) + \xi p(i,0,1) + \xi q_f p(i-1,1,1)
\end{aligned}$$

$$\sum_{i=0}^{\infty} [p(i,0,0) + p(i,1,0) + p(i,2,0) + p(i,1,1) + p(i,0,1) + p(i,0,2)] = 1.$$

Due to transitions that correspond to successful retrial attempts, deriving the steady-state probabilities in most retrial queueing systems is challenging. In this model, the difficulty is compounded by server failures that also result in transitions between levels. Therefore, solving this system in a recursive fashion is non-trivial. Another way of computing the steady-state probabilities is by the method of generating functions. Define

$$\phi_{j,k}(z) = \sum_{i=0}^{\infty} p(i,j,k)z^i, \quad (j,k) \in E \tag{3.1}$$

as the probability generating function (p.g.f.) of $p(i,j,k)$ with respect to the orbit size. Applying this function to the balance equations in the usual manner and summing over all values of i we obtain the following system of differential equations and normalization equation in the transform variable z:

$$\begin{aligned}
(\lambda+2\xi)\phi_{0,0}(z)+\theta z\phi'_{0,0}(z) &= \mu\phi_{1,0}(z)+\alpha\phi_{0,1}(z)\\
(\lambda+\xi+\xi q_f+\mu)\phi_{1,0}(z)+\theta z\phi'_{1,0}(z) &= \lambda\phi_{0,0}(z)+\alpha\phi_{1,1}(z)+2\mu\phi_{2,0}(z)+\theta\phi'_{0,0}(z)\\
(\lambda q_a+2\mu+2\xi q_f)\phi_{2,0}(z) &= \lambda q_a z\phi_{2,0}(z)+\lambda\phi_{1,0}(z)+\theta\phi'_{1,0}(z)\\
(\lambda q_a+\mu+\xi q_f+\alpha)\phi_{1,1}(z) &= \lambda q_a z\phi_{1,1}(z)+\lambda\phi_{0,1}+\theta\phi'_{0,1}(z)+\xi\phi_{1,0}(z)\\
&\quad +2\xi q_f z\phi_{2,0}(z)\\
(\lambda+\xi+\alpha)\phi_{0,1}(z)+\theta z\phi'_{0,1}(z) &= \mu\phi_{1,1}(z)+\xi q_f z\phi_{1,0}(z)+2\xi\phi_{0,0}(z)+2\alpha\phi_{0,2}(z)\\
(\lambda q_a+2\alpha)\phi_{0,2}(z) &= \lambda z\phi_{0,2}(z)+\xi\phi_{0,1}(z)+\xi q_f z\phi_{1,1}(z)\\
\sum_{(j,k)\in E}\phi_{j,k}(1) &= 1.
\end{aligned}$$

The solution of this system of equations requires the simultaneous solution of three differential equations, one for each of $\phi'_{0,0}(z)$, $\phi'_{1,0}(z)$ and $\phi'_{0,1}(z)$, and back substituting to solve for the remaining three. However, this is not easily accomplished, and thus, due to the complexity of solving for the state probabilities recursively, or by the method of generating functions, we instead resort to an approximate analysis of the system. Due to the structure of the transition diagram (Figure 3.2), a phase-merging algorithm developed by Korolyuk and Korolyuk [23] and Courtois [13] will be employed and is summarized in the following sections.

3.2 The Phase-Merging Algorithm

Beginning with a CTMC on a state space that completely describes a retrial queueing system, a two-dimensional transition rate diagram is constructed as in Figure 3.2. The objective of the phase-merging algorithm is to approximate the

steady-state probability distribution of $\{(R(t), B(t), F(t)) : t \geq 0\}$ by approximating the conditional probability distribution of the status of the servers, given the level of the orbit, and by approximating the marginal probability distribution of the number of customers in orbit. The algorithm proceeds by partitioning the state space into disjoint and mutually exhaustive sets that correspond to levels of the orbit. Each level is analyzed as a CTMC from which the approximate conditional probabilities are obtained. Each level itself is subsequently considered as a state of an aggregated CTMC where the transition rates between levels correspond to customers entering or leaving the orbit. Analyzing this system of "macrostates" yields the approximate marginal probability distribution of the number of customers in the orbit. The product of the conditional and marginal probabilities is, therefore, the approximate joint probability distribution of the level of the orbit and status of the servers. Using this joint distribution, we then approximate standard queueing performance measures.

To begin, we reduce the dimensionality of the state space by defining $X(t)$ as the status of the servers at time t (outlined in Table 3.1), such that $X(t) \in \{1, 2, 3, 4, 5, 6\}$. In order to accurately approximate the joint probability distribution

Table 3.1 Substitution for server status.

State (j, k)	Index (l)
(0,0)	1
(1,0)	2
(2,0)	3
(1,1)	4
(0,1)	5
(0,2)	6

of the number of customers in the orbit and status of the servers, it is necessary that the rates of flow within levels of the orbit are significantly greater than those rates flowing between levels. Referring to Figure 3.2, we need

$$\lambda \gg \theta, \xi \quad \mu \gg \theta, \xi \quad \alpha \gg \theta, \xi.$$

The algorithm, which was developed in [23] and [13] proceeds in the following manner. First, partition the state space, S, into disjoint sets that are conditional upon i such that,

$$S = \bigcup_{i=0}^{\infty} S_i,\ S_i \cap S_j = \emptyset, \quad i \neq j$$

where $S_i \equiv \{(i,l) : l = 1, 2, \ldots, 6\},\ i \geq 0$. This step results in an infinite number of classes (or levels) which can be analyzed individually.

Next we obtain the steady-state distribution of each class or level, S_i, by determining the infinitesimal generator matrix, $\boldsymbol{Q}_i$ defined by

$$q_{l,m} = \begin{cases} q_{(i,l),(i,m)} & l \neq m \\ -\sum_{l \neq m} q_{(i,l),(i,m)} & l = m \\ 0 & \text{otherwise} \end{cases} . \tag{3.2}$$

Denote by $p_{l|i}$ the steady-state conditional probability that the status of the servers is state l, given there are i customers in orbit, $i \geq 0$, $l = 1, 2, \ldots, 6$. Letting $\boldsymbol{p}_i = [p_{l|i}]$, we solve the system of equations $\boldsymbol{p}_i \boldsymbol{Q}_i = \boldsymbol{0}$ and $\boldsymbol{p}_i \boldsymbol{e} = 1$ (where $\boldsymbol{e}$ is a column vector of ones) to obtain the approximate conditional probability distribution.

Following this, we merge, or aggregate, all states within the class S_i, into one state corresponding to the level of orbit, i. These "macrostates" form the overall state space of the merged model which are defined as $\hat{S} \equiv \{i : i \geq 0\}$. The infinitesimal generator, $\boldsymbol{Q}_M$, of the merged model is

$$q_{i,j} \equiv \sum_{(i,l) \in S_i} p_{l|i} \left(\sum_{(j,m) \in S_j} q_{(i,l),(j,m)} \right) .$$

Denote π_i as the marginal probability that there are i customers in the orbit. Letting the infinite-dimensional vector $\boldsymbol{\pi} \equiv [\pi_0, \pi_1, \pi_2, \pi_3, \ldots]$, we solve the system of equations $\boldsymbol{\pi} \boldsymbol{Q}_M = \boldsymbol{0}$ and $\boldsymbol{\pi} \boldsymbol{e} = 1$ to obtain the approximate steady-state marginal

probabilities. Finally, the steady-state distribution of $\{(R(t), X(t)) : t \geq 0\}$ may be approximated by

$$p(i,l) \approx \hat{p}(i,l) = p_{l|i} \times \pi_i, \quad i \geq 0,\ l = 1, 2, \ldots, 6. \tag{3.3}$$

Making use of these joint probabilities, we can obtain approximations for the performance measures for the unreliable two-server retrial queue.

To illustrate the algorithm, let us first consider a reliable M/M/2 retrial queue whose customers arrive according to a Poisson process with rate λ. Each customer brings an exponential service requirement with mean time $1/\mu$. Customers who find both servers busy enter the orbit with probability c or leave the system with probability $1 - c$. The time between retrials is exponentially distributed with mean $1/\theta$. All times are assumed to be mutually independent. Define the continuous-time stochastic process as $\{(R(t), B(t)) : t \geq 0\}$ where $R(t)$ is the number of customers in the orbit at time t and $B(t)$ is the number of busy servers at time t. The process is a CTMC on the state space, $S = \{(i,j) :\ i \geq 0,\ j = 0, 1, 2\}$. For the purpose of illustrating the phase-merging algorithm, we will assume the system is stable and denote $p(i,j) = \lim_{t\to\infty} P(R(t) = i, B(t) = j)$ as the limiting probability that the system is in state (i,j), $i \geq 0$, $j = 0, 1, 2$. Figure 3.3 depicts the two-dimensional transition rate diagram.

To make use of the algorithm we assume that each of λ and μ are significantly greater than θ and proceed as follows: First, partition the state space into individual levels where the index of each level corresponds to the number of customers in the orbit. Denote this as class S_i for level $i, i \geq 0$. Note that each class has an identical structure and, therefore, the generator matrices, $\boldsymbol{Q}_i$ are identical for all $i \geq 0$. This fact will be extremely useful for analyzing the case of unreliable servers.

Next we compute the steady-state conditional distribution of the status of the servers given there are i customers in orbit. Denote these probabilities by $p_{j|i}$,

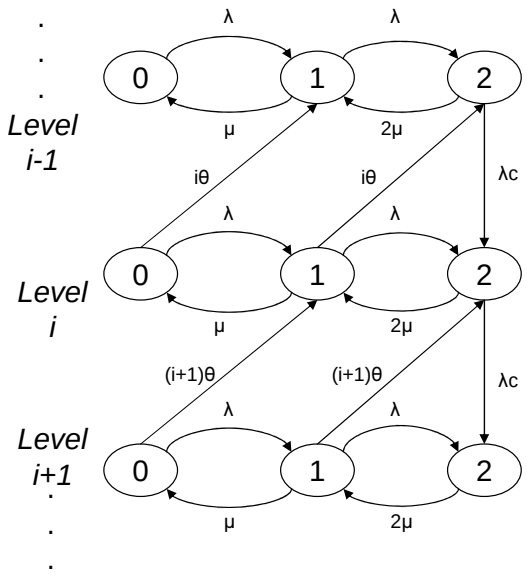

Figure 3.3 Transition rate diagram for a reliable M/M/2 retrial queue.

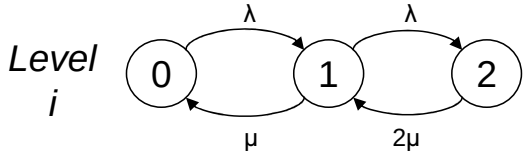

Figure 3.4 Transition rate diagram for level i: reliable M/M/2 retrial queue.

$j = 0, 1, 2$. Standard nodal analysis works best in this example, and it is easy to obtain the following conditional probabilities for all $i \geq 0$:

$$p_{0|i} = \frac{\mu^2}{\mu^2 + \lambda\mu + \lambda^2}, \tag{3.4}$$

$$p_{1|i} = \frac{\lambda\mu}{\mu^2 + \lambda\mu + \lambda^2}, \tag{3.5}$$

$$p_{2|i} = \frac{\lambda^2}{\mu^2 + \lambda\mu + \lambda^2}. \tag{3.6}$$

The next step is to aggregate the states of each class to form a series of merged states, i where $i \geq 0$ and investigate the transitions between them. The elements of

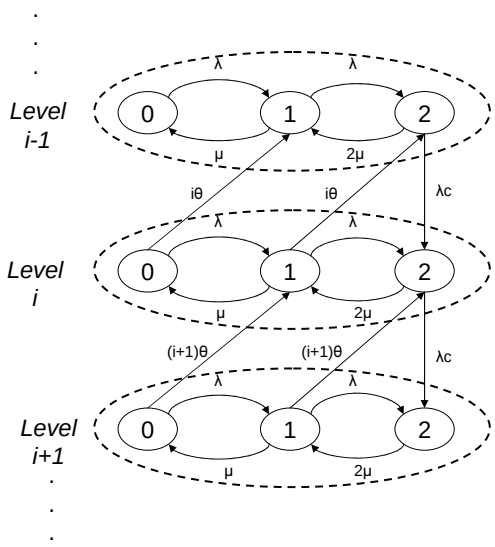

Figure 3.5 The merged model for the reliable M/M/2 retrial queue.

the infinitesimal generator matrix for the merged states are

$$q_{i,j} = \begin{cases} \lambda c p_{2|i} & i \geq 0,\ j = i+1 \\ i\theta(p_{0|i} + p_{1|i}) & i \geq 1,\ j = i-1 \\ -[\lambda c p_{2|i} + i\theta(p_{0|i} + p_{1|i})] & i = j \\ 0 & \text{otherwise} \end{cases}.$$

Using the substitutions, $\hat{\lambda} = \lambda c p_{2|i}$ and $\hat{\theta} = \theta(p_{0|i} + p_{1|i})$, we see that the analysis of this system is analogous to the M/M/∞ queueing system. Thus, defining the steady-state marginal probability vector as $\boldsymbol{\pi} = [\pi_0, \pi_1, \pi_2, \ldots]$ we have,

$$\pi_i = \frac{1}{i!}\left(\frac{\hat{\lambda}}{\hat{\theta}}\right)^i e^{-\hat{\lambda}/\hat{\theta}}, \quad i \geq 0. \tag{3.7}$$

Finally, the approximate steady-state distribution of $\{(R(t), B(t)) : t \geq 0\}$ is given by

$$\begin{aligned} p(i,j) \approx \hat{p}(i,j) &= p_{j|i} \times \pi_i \\ &= \frac{p_{j|i}}{i!} \left(\frac{\hat{\lambda}}{\hat{\theta}} \right)^i e^{-\hat{\lambda}/\hat{\theta}}, \quad i \geq 0,\ j = 0, 1, 2. \end{aligned} \tag{3.8}$$

The advantages of using the phase-merging algorithm are two-fold. First, good approximations for the steady-state probabilities can be computed quickly as compared to simulating the system. Second, for many multi-server retrial queueing systems, obtaining exact solutions can be extremely difficult, if not impossible. A disadvantage of the algorithm is that it depends on the assumption that transition intensities *within* each level are significantly greater than those *between* levels. Thus, the algorithm is most effective when this requirement is satisfied.

3.3 Approximation Using the Phase-Merging Algorithm

We now apply the phase-merging algorithm described in [23] and [13] to the unreliable M/M/2 retrial queue. Recall that the interarrival times, service and repair times, time between failures and time between retrials are all exponentially distributed with the parameters defined previously. Since the number of customers in the orbit can theoretically reach infinity, the state space of the system can be partitioned into a countable number of classes. As noted previously, the state space S is partitioned as the countable union

$$S = \bigcup_{i=0}^{\infty} S_i,\ S_i \cap S_j = \emptyset \quad i \neq j,$$

where $S_i = \{(i, l) : l = 1, 2, \ldots, 6\}, i \geq 0$. Just as in the reliable M/M/2 retrial queue, each class is identical in structure so that only one class needs to be analyzed.

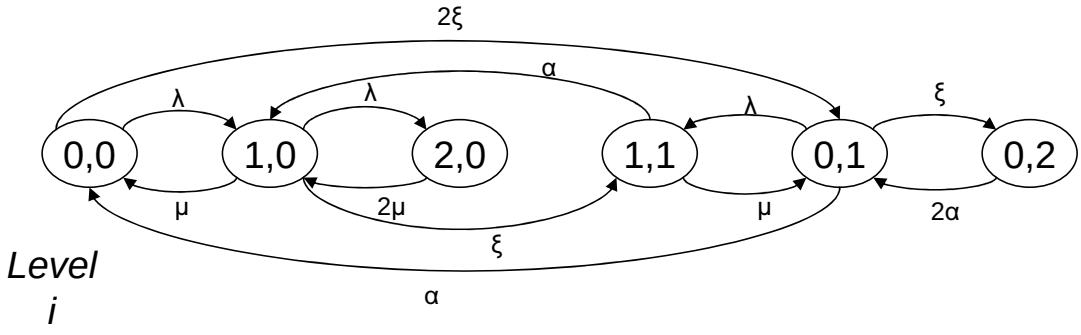

Figure 3.6 The class S_i for the unreliable M/M/2 retrial queue.

To determine the steady-state probabilities for the class S_i, define the stochastic process $\{(B(t), F(t)) : t \geq 0\}$ where $B(t)$ represents the number of busy servers and $F(t)$ represents the number of failed servers at time t. Clearly, the process is a CTMC on the state space E defined previously. Using the notation defined in Table 3.1 we denote $p_{l|i}$ as the limiting conditional probability of the servers being in state l given that there are i customers in orbit,

$$p_{l|i} = \lim_{t\to\infty} P(X(t) = l|R(t) = i),\ l = 1, 2, \ldots, 6.$$

For each $i \geq 0$, the transition rates for this process are described in the following generator matrix, $\boldsymbol{Q}_i$.

$$\boldsymbol{Q}_i = \begin{bmatrix} -(\lambda+2\xi) & \lambda & 0 & 0 & 2\xi & 0 \\ \mu & -(\lambda+\xi+\mu) & \lambda & \xi & 0 & 0 \\ 0 & 2\mu & -2\mu & 0 & 0 & 0 \\ 0 & \alpha & 0 & -(\alpha+\mu) & \mu & 0 \\ \alpha & 0 & 0 & \lambda & -(\lambda+\xi+\alpha) & \xi \\ 0 & 0 & 0 & 0 & 2\alpha & -2\alpha \end{bmatrix}.$$

Let $\boldsymbol{p}_i$ be the steady-state conditional probability vector where $\boldsymbol{p}_i = [p_{l|i}]$, $l = 1, 2, ...6$. Solving the equations $\boldsymbol{p}_i\boldsymbol{Q}_i = \boldsymbol{0}$ and $\boldsymbol{p}_i\boldsymbol{e} = 1$ yields the following system,

$$\begin{aligned}
(\lambda + 2\xi)p_{1|i} &= \mu p_{2|i} + \alpha p_{5|i} && (3.9)\\
(\lambda + \xi + \mu)p_{2|i} &= \lambda p_{1|i} + 2\mu p_{3|i} + \alpha p_{4|i} && (3.10)\\
2\mu p_{3|i} &= \lambda p_{2|i} && (3.11)\\
(\alpha + \mu)p_{4|i} &= \xi p_{2|i} + \lambda p_{5|i} && (3.12)\\
(\lambda + \xi + \alpha)p_{5|i} &= 2\xi p_{1|i} + \mu p_{4|i} + 2\alpha p_{6|i} && (3.13)\\
2\alpha p_{6|i} &= \xi p_{5|i} && (3.14)\\
\sum_{l=1}^{6} p_{l|i} &= 1. && (3.15)
\end{aligned}$$

Replacing Equation (3.13) with the normalization equation (3.15), the solution to the conditional probabilities are obtained for all $i \geq 0$

$$\begin{aligned}
p_{1|i} &= D^{-1}(2(\alpha + \lambda + \xi + \mu)\alpha^2\mu^2)\\
p_{2|i} &= D^{-1}(2(\alpha + \mu + \lambda + 2\xi)\alpha^2\mu\lambda)\\
p_{3|i} &= D^{-1}((\alpha + \mu + \lambda + 2\xi)\alpha^2\lambda^2)\\
p_{4|i} &= D^{-1}(2(\alpha + \lambda + 2\mu + 2\xi)\alpha\mu\xi\lambda)\\
p_{5|i} &= D^{-1}(2\alpha\xi\mu^2(\lambda + 2\alpha + 2\mu + 2\xi))\\
p_{6|i} &= D^{-1}(\mu^2\xi^2(\lambda + 2\alpha + 2\mu + 2\xi))
\end{aligned}$$

where the constant D is given by,

$$\begin{aligned}
D = {} & \mu^2\xi^2\lambda + 6\mu^2\xi^2\alpha + 2\mu^3\xi^2 + 2\mu^2\xi^3 + 6\alpha^2\mu\xi\lambda + 2\mu\alpha^3\lambda + 4\lambda\alpha^2\mu^2\\
& + \lambda^2\alpha^3 + 3\lambda^2\alpha^2\mu + \lambda^3\alpha^2 + 2\xi\lambda^2\alpha^2 + 4\mu^3\xi\alpha + 6\alpha^2\mu^2\xi + 2\mu^2\alpha^3\\
& + 2\mu^3\alpha^2 + 2\mu\xi\lambda^2\alpha + 6\mu^2\xi\lambda\alpha + 4\alpha\mu\xi^2\lambda.
\end{aligned}$$

Aggregating the states of each class S_i yields a system of macro-states which we denote as i, $i \geq 0$. The rates of transition between the "macrostates" are expressed

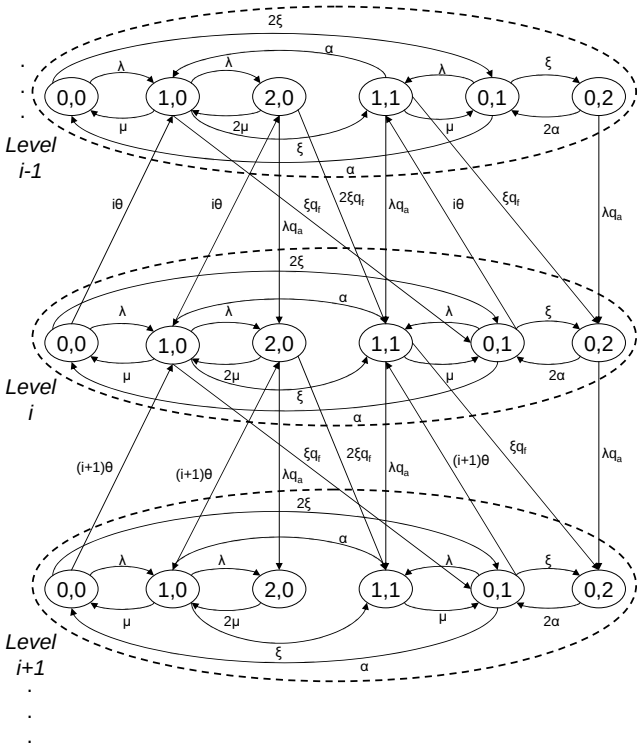

Figure 3.7 Transition rate diagram for the merged model.

in the infinitesimal generator matrix with elements,

$$q_{i,j} = \begin{cases} \xi q_f p_{2|i} + (\lambda q_a + 2\xi q_f) p_{3|i} + (\lambda q_a + \xi q_f) p_{4|i} + \lambda q_a p_{6|i} & i \geq 0,\ j = i+1 \\ i\theta(p_{1|i} + p_{2|i} + p_{5|i}) & i \geq 1,\ j = i-1 \\ -[\xi q_f p_{2|i} + (\lambda q_a + 2\xi q_f) p_{3|i} + (\lambda q_a + \xi q_f) p_{4|i} + \lambda q_a p_{6|i} & \\ +i\theta(p_{1|i} + p_{2|i} + p_{5|i})] & i = j \\ 0 & \text{otherwise} \end{cases} .$$

To simplify the analysis of the merged states we use the following substitutions for $i \geq 0$,

$$\hat{\lambda} = \xi q_f p_{2|i} + (\lambda q_a + 2\xi q_f) p_{3|i} + (\lambda q_a + \xi q_f) p_{4|i} + \lambda q_a p_{6|i} \tag{3.16}$$

$$\hat{\theta} = \theta(p_{1|i} + p_{2|i} + p_{5|i}). \tag{3.17}$$

Making the substitutions, the elements of the generator matrix are,

$$q_{i,j} = \begin{cases} \hat{\lambda}, & i \geq 0,\ j = i+1 \\ i\hat{\theta}, & i \geq 1,\ j = i-1 \\ -(\hat{\lambda} + i\hat{\theta}), & i = j \\ 0, & \text{otherwise} \end{cases}.$$

This new model is a state dependent birth-and-death process, the analysis of which is analogous to the $M/M/\infty$ queueing system. Using the method of arc cuts, we recursively solve for the steady-state probability vector, $\boldsymbol{\pi} = [\pi_0, \pi_1, \pi_2, \pi_3, \ldots]$.

$$\begin{aligned} \hat{\lambda}\pi_0 &= \hat{\theta}\pi_1 \Rightarrow \pi_1 = \frac{\hat{\lambda}}{\hat{\theta}}\pi_0 \\ \hat{\lambda}\pi_1 &= 2\hat{\theta}\pi_2 \Rightarrow \pi_2 = \frac{1}{2}\left(\frac{\hat{\lambda}}{\hat{\theta}}\right)^2 \pi_0 \\ \hat{\lambda}\pi_2 &= 3\hat{\theta}\pi_3 \Rightarrow \pi_3 = \frac{1}{6}\left(\frac{\hat{\lambda}}{\hat{\theta}}\right)^3 \pi_0 \\ \hat{\lambda}\pi_3 &= 4\hat{\theta}\pi_4 \Rightarrow \pi_4 = \frac{1}{24}\left(\frac{\hat{\lambda}}{\hat{\theta}}\right)^4 \pi_0 \end{aligned}$$

Continuing inductively, it can easily be shown that,

$$\pi_i = \frac{1}{i!}\left(\frac{\hat{\lambda}}{\hat{\theta}}\right)^i \pi_0,\ i \geq 0. \tag{3.18}$$

Using the normalization equation, $\sum_{j=0}^{\infty} \pi_j = 1$, the solution for π_0 is obtained by

$$\begin{aligned}
\pi_0 + \frac{\hat{\lambda}}{\hat{\theta}}\pi_0 + \frac{1}{2}\left(\frac{\hat{\lambda}}{\hat{\theta}}\right)^2 \pi_0 + \frac{1}{6}\left(\frac{\hat{\lambda}}{\hat{\theta}}\right)^3 \pi_0 + \ldots &= 1 \\
\pi_0 \left(1 + \frac{\hat{\lambda}}{\hat{\theta}} + \frac{1}{2}\left(\frac{\hat{\lambda}}{\hat{\theta}}\right)^2 + \frac{1}{6}\left(\frac{\hat{\lambda}}{\hat{\theta}}\right)^3 + \ldots\right) &= 1 \\
\pi_0 \left(\sum_{j=0}^{\infty} \frac{1}{j!}\left(\frac{\hat{\lambda}}{\hat{\theta}}\right)^j\right) &= 1. \qquad (3.19)
\end{aligned}$$

The infinite series of (3.19) is the Maclaurin power series expansion for $e^{\hat{\lambda}/\hat{\theta}}$. Thus, we see that

$$\pi_0 = e^{-\hat{\lambda}/\hat{\theta}}.$$

Substituting π_0 into Equation (3.18) we have the following expression,

$$\pi_i = \frac{1}{i!}\left(\frac{\hat{\lambda}}{\hat{\theta}}\right)^i e^{-\hat{\lambda}/\hat{\theta}}, \quad i \geq 0, \qquad (3.20)$$

which is the probability mass function for a Poisson distributed random variable with rate parameter $\hat{\lambda}/\hat{\theta}$. Finally, we approximate the steady-state distribution of $\{(R(t), X(t)) : t \geq 0\}$ by

$$\begin{aligned}
p(i,l) \approx \hat{p}(i,l) &= p_{l|i} \times \pi_i \\
&= \frac{p_{l|i}}{i!}\left(\frac{\hat{\lambda}}{\hat{\theta}}\right)^i e^{-\hat{\lambda}/\hat{\theta}},\ i \geq 0,\ l = 1, 2, \ldots, 6. \qquad (3.21)
\end{aligned}$$

3.4 Approximate Queueing Performance Measures

In this section we provide approximations for the limiting mean orbit length, mean number of customers in service, the mean number of customers in the system, the mean sojourn time and the mean time spent in orbit.

3.4.1 Mean Orbit Length

In the aggregated model, each level corresponds to the number of customers in orbit. It was shown that the steady-state distribution is Poisson with parameter $\hat{\lambda}/\hat{\theta}$. Therefore, the long-run mean orbit length is approximately the expected value of this Poisson random variable. Denoting R as the steady-state number of customers in orbit, the mean orbit size is approximated by

$$\begin{aligned} E[R] &\approx \frac{\hat{\lambda}}{\hat{\theta}} \\ &= \frac{\xi q_f p_{2|i} + (\lambda q_a + 2\xi q_f) p_{3|i} + (\lambda q_a + \xi q_f) p_{4|i} + \lambda q_a p_{6|i}}{\theta(p_{1|i} + p_{2|i} + p_{5|i})}. \end{aligned} \tag{3.22}$$

3.4.2 Mean Number of Customers in Service

The approximate expression for the expected number of customers at the servers can be computed using the approximate steady-state joint probabilities derived in the last step of the algorithm. Let N_s be defined as the random number of customers at the servers.

$$\begin{aligned} E[N_s] &= \sum_{i=0}^{\infty} [p(i,1,0) + p(i,1,1) + 2p(i,2,0)] \\ &\approx \sum_{i=0}^{\infty} [\hat{p}(i,2) + \hat{p}(i,4) + 2\hat{p}(i,3)]. \end{aligned} \tag{3.23}$$

3.4.3 Steady-State System Size and Sojourn Time

To calculate L, the steady-state number of customers in the system, we simply sum the expressions for $E[R]$ and $E[N_s]$. The steady-state mean sojourn time, W, follows directly from Little's law.

$$L \approx E[R] + E[N_s] \tag{3.24}$$

$$W \approx \frac{L}{\lambda}, \tag{3.25}$$

where $E[R]$ is obtained by Equation (3.22) and $E[N_s]$ is obtained by Equation (3.23).

3.4.4 Total Expected Time in Orbit

Due to server failures and blocking when making a retrial attempt, customers may enter the orbit more than once. Therefore, the expected time a customer spends in orbit is $1/\theta$ times the expected number of retrial attempts before gaining access to the server. Define Y as the random number of retrials a customer performs until it gains access to a server. Then Y is a geometric random variable with parameter p_u, the steady-state probability that at least one server is available. The approximation for p_u is given by

$$\begin{aligned} p_u &= \sum_{i=0}^{\infty}[p(i,0,0)+p(i,1,0)+p(i,0,1)] \\ &\approx \sum_{i=0}^{\infty}[\hat{p}(i,1)+\hat{p}(i,2)+\hat{p}(i,5)]. \end{aligned} \tag{3.26}$$

The expected number of retrials performed, $E[Y]$, is therefore, $1/p_u$ and letting W_r be the random time spent in orbit once they are there we have,

$$E[W_r] \approx (\theta p_u)^{-1}. \tag{3.27}$$

In this chapter we have formally defined the mathematical model and, employing the phase-merging algorithm, have derived approximate expressions for the steady-state joint probability distribution of the number of customers in orbit and status of the servers. Using these probabilities we approximated several performance characteristics of the unreliable M/M/2 retrial queue. In the next chapter, we will assess the quality of our approximations by comparing the results with those obtained by a discrete-event simulation model.

4. Numerical Experiments

In this chapter we assess the quality of the phase-merging approximation presented in Chapter 3 for the unreliable M/M/2 retrial queue. Using a benchmark discrete-event simulation model, we will compare results for congestion and delay measures. We modified a validated unreliable M/M/1 retrial queue simulation model created by Sherman [36] to include an additional unreliable server. We then execute the new model without failures and compare results to an exact analysis of the *reliable* M/M/2 retrial queue (see Falin and Templeton [17]). Subsequently, we turn our attention to the case of two unreliable servers with two dedicated repair persons. To begin, we review the exact analysis of the reliable M/M/2 retrial queue.

4.1 Review of the Reliable M/M/2 Retrial Queue

Falin and Templeton [17] provide a detailed analysis of the standard M/M/2 retrial queue wherein customers arrive to the system according to a Poisson process with rate λ ($\lambda > 0$). Without loss of generality, the authors assume the service rate, μ, is equal to unity. Customers who perform retrials do so according to an exponential distribution with mean $1/\theta$. They define the stochastic process $\{(R(t), B(t)) : t \geq 0\}$, where $R(t)$ is the number of customers in the orbit at time t and $B(t)$ is the number of busy servers at time t. The process is a CTMC on the state space, $S = \{(i, j) : i \geq 0, j = 0, 1, 2\}$. The stability condition for this system is $\lambda < 2$ (assuming $\mu = 1$). The authors recursively derived the steady-state joint distribution of orbit size and the number of busy servers in terms of hypergeometric functions. The performance measures of interest are the steady-state mean number of customers in the orbit, denoted by $E[R]$, and the probability of blocking which

we denote here by p_B. These quantities are given by

$$E[R] = \frac{1+\theta}{\theta} \cdot \frac{\lambda^3 + (\lambda^2 - 2\lambda + 2)g}{(2-\lambda)(2+\lambda+g)}, \tag{4.1}$$

$$p_B = \frac{\lambda^2 + (\lambda - 1)g}{2+\lambda+g}, \tag{4.2}$$

where

$$g = \frac{\lambda^3}{2+3\lambda+2\theta} \cdot \frac{F(a+1, b+1, c+1; \frac{\lambda}{2})}{F(a, b, c; \frac{\lambda}{2})}.$$

The function F is the hypergeometric function defined by,

$$F(a, b, c; x) \equiv \sum_{i=0}^{\infty} \frac{x^i}{i!} \prod_{k=0}^{i-1} \frac{(a+k)(b+k)}{c+k},$$

where

$$\begin{aligned} a &= \frac{2\lambda + 1 + \sqrt{4\lambda+1}}{2\theta}, \\ b &= \frac{2\lambda + 1 - \sqrt{4\lambda+1}}{2\theta}, \\ c &= \frac{2+3\lambda+2\theta}{2\theta}. \end{aligned}$$

4.2 Validation of Arena® Simulation

Sherman [36] provided an exact analysis for an unreliable M/M/1 retrial queue. Using the exact results, the author validated a discrete-event simulation model in the Arena® environment. We extend his validated simulation model by including an additional unreliable server. The simulation model for the unreliable M/M/2 retrial queue was created using the professional version of Arena® and executed on an IBM® Thinkpad with a 1.86 GHz Intel® Centrino processor and 0.99 GB of RAM.

To further ensure the accuracy of our simulation model, we compared the exact queueing measures of the reliable M/M/2 retrial queue to the output of simulations run with the failure parameter $\xi = 0$. Choosing $\mu = 1$ and $\theta = 0.5$, we varied λ so as

to compute the mean orbit length and probability of blocking under different traffic intensities. The exact solutions for mean orbit length, $E[R]$, and the probability of blocking, p_B, were computed using two functions coded in MATLAB® and can be found in the appendix. The main program, *MeanQueueLength*, computes both p_B and $E[R]$, given values for the parameters λ and θ. Recall that μ is assumed to be 1. Using these parameters, the function computes the values a, b and c which are passed into another function, *Hypergeometric*. The purpose of this MATLAB® function is to compute the hypergeometric functions, F, that are needed to obtain g, which in turn is used in the main function, *MeanQueueLength*, to compute $E[R]$ and p_B.

To conduct the simulation experiments, we first determined an appropriate run length for each replication. By investigating the transient period for a few test cases, we determined that a warm-up period of 400,000 hours was needed to reduce the bias for the point estimates of our two measures, $E[R]$ and p_B. Each replication ran for 1,000,000 hours, including the 400,000 hour initialization period. To determine the number of replications, n, for each experiment we used the following formula:

$$n \geq \left(\frac{z_{\alpha/2} S_0}{\epsilon} \right)^2 \tag{4.3}$$

We desired a half-width of $\epsilon = 0.01$ in estimating mean orbit length, $E[R]$ and a half-width of $\epsilon = 0.001$ in estimating the probability of blocking, p_B both with 95% confidence. We ran the experiments for 30 replications to obtain the sample standard deviation, S_0 and using $\alpha = 0.05$, we determined that 10 additional replications were needed to estimate within the specified values of ϵ. The following table displays our results for the experiment with 40 replications, each lasting 1,000,000 hours including a 400,000 hour warm-up. We also provide a 95% confidence interval for each performance measure, as well as the absolute difference between the midpoint of the interval and the exact result from Falin and Templeton [17].

Table 4.1 Simulated versus exact results for the reliable M/M/2 retrial queue.

λ		Lower CI Limit	Midpoint	Upper CI Limit	Exact	Abs. Diff.
0.50	$E[R]$	0.13335	0.13374	0.13413	0.13366	0.00008
	p_B	0.09144	0.09160	0.09175	0.09159	0.00001
0.75	$E[R]$	0.45986	0.46113	0.46240	0.46042	0.00071
	p_B	0.18512	0.18536	0.18560	0.18533	0.00003
1.00	$E[R]$	1.18465	1.18739	1.19012	1.18639	0.00099
	p_B	0.30211	0.30237	0.30264	0.30227	0.00010
1.60	$E[R]$	9.13450	9.16290	9.19131	9.16640	0.00350
	p_B	0.66828	0.66886	0.66944	0.66891	0.00005
1.70	$E[R]$	14.00109	14.05065	14.10021	14.04110	0.00955
	p_B	0.74258	0.74313	0.74368	0.74295	0.00018

The relatively small absolute difference values lead us to conclude that the simulation model excluding failures provides valid results for the reliable M/M/2 queue. We subsequently incorporate failures into the extension of the simulation model by Sherman [36].

4.3 Approximated Versus Simulated Performance Measures

In this section, we use the phase-merging algorithm to approximate values for the mean orbit length, $E[R]$, mean sojourn time, $E[W]$, expected number of customers at the servers, $E[N_s]$ and mean time spent in orbit, $E[W_r]$ for the unreliable M/M/2 retrial queue. These approximations are then compared with the results of an Arena® simulation model. The approximations are computed using a MATLAB® function, *ClassProbs*. Given values for the parameters $\lambda, \mu, \xi, \alpha, \theta, q_a$ and q_f, the function first calculates the conditional steady-state probabilities given each level of the orbit (which are equivalent for all levels) and stores them in a vector. Next, the values $\hat{\lambda}$ and $\hat{\theta}$ are computed by Equations (3.16) and (3.17). Us-

ing the fact that the marginal distribution of the number of customers in orbit is approximately Poisson with rate $\hat{\lambda}/\hat{\theta}$, we approximate the steady-state joint distribution of the number of customers in orbit and status of the servers by multiplying the conditional and marginal probabilities. The function then uses the steady-state distribution to approximate the various queueing performance measures.

To begin, we chose the following values for the parameters so as to meet the requirements of the phase-merging algorithm: $\mu = 6$, $\xi = 0.01$, $\alpha = 5$, $\theta = 0.1$ and $q_f = 0.5$. Recall, that for the algorithm to produce effective results we require that the flows within levels of the orbit be significantly greater than those between levels. For each value of λ ($\lambda = 2, 4$ and 6) selected, we varied q_a from 0 to 1 in increments of 0.1. For consistency in experimentation, 40 replications were executed using a run length of 1,000,000 hours including a 400,000 hour warm-up. The following tables and figures provide comparisons between the approximations and the simulated performance measures. For the simulated means, we provide 95% confidence intervals and include an absolute difference between the midpoint of the interval and the approximation.

We are also interested in the sensitivity of the approximation procedure to perturbations of q_f. Tables 4.8 and 4.9 and Figures 4.13 through 4.16 provide the results for $\lambda = 2$, $\mu = 6, \xi = 0.01, \alpha = 5, \theta = 0.1$ and $q_a = 0.5$. The simulation was again replicated 40 times, each one for 1,000,000 hours including a 400,000 hour warm-up.

Table 4.2 Numerical results for unreliable M/M/2 retrial queue with $\lambda = 2$.

q_a		Lower CI Limit	Midpoint	Upper CI Limit	Approximation	Abs. Diff.
0.0	$E[R]$	0.01645	0.01668	0.01691	0.01667	0.00002
	$E[W]$	0.16797	0.16809	0.16822	0.16824	0.00014
0.1	$E[R]$	0.10254	0.10254	0.10254	0.10128	0.00126
	$E[W]$	0.21165	0.21191	0.21217	0.21054	0.00137
0.2	$E[R]$	0.18955	0.19029	0.19103	0.18590	0.00439
	$E[W]$	0.25581	0.25618	0.25655	0.25285	0.00333
0.3	$E[R]$	0.27884	0.27986	0.28087	0.27051	0.00935
	$E[W]$	0.30113	0.30162	0.30211	0.29516	0.00646
0.4	$E[R]$	0.36891	0.36985	0.37078	0.35512	0.01472
	$E[W]$	0.34681	0.34726	0.34771	0.33746	0.00980
0.5	$E[R]$	0.46042	0.46157	0.46272	0.43974	0.02183
	$E[W]$	0.39328	0.39384	0.39440	0.37977	0.01407
0.6	$E[R]$	0.55345	0.55490	0.55635	0.52435	0.03054
	$E[W]$	0.44046	0.44118	0.44191	0.42208	0.01911
0.7	$E[R]$	0.64746	0.64921	0.65096	0.60897	0.04024
	$E[W]$	0.48819	0.48906	0.48994	0.46439	0.02468
0.8	$E[R]$	0.74497	0.74660	0.74824	0.69358	0.05302
	$E[W]$	0.53762	0.53843	0.53924	0.50669	0.03174
0.9	$E[R]$	0.84182	0.84344	0.84507	0.77820	0.06525
	$E[W]$	0.58674	0.58756	0.58837	0.54900	0.03856
1.0	$E[R]$	0.94083	0.94272	0.94460	0.86281	0.07991
	$E[W]$	0.63686	0.63776	0.63867	0.59131	0.04646

Table 4.3 Numerical results for unreliable M/M/2 retrial queue with $\lambda = 2$.

q_a		Lower CI Limit	Midpoint	Upper CI Limit	Approximation	Abs. Diff.
0.0	$E[N_s]$	0.32333	0.32345	0.32358	0.31980	0.00365
	$E[W_r]$	10.35055	10.44296	10.53536	10.42307	0.01989
0.1	$E[N_s]$	0.32461	0.32473	0.32485	0.31980	0.00492
	$E[W_r]$	10.48955	10.53118	10.57280	10.42307	0.10810
0.2	$E[N_s]$	0.32592	0.32604	0.32617	0.31980	0.00624
	$E[W_r]$	10.50424	10.53548	10.56671	10.42307	0.11240
0.3	$E[N_s]$	0.32718	0.32731	0.32744	0.31980	0.00751
	$E[W_r]$	10.51904	10.54525	10.57146	10.42307	0.12218
0.4	$E[N_s]$	0.32853	0.32866	0.32879	0.31980	0.00886
	$E[W_r]$	10.52712	10.54940	10.57168	10.42307	0.12633
0.5	$E[N_s]$	0.32984	0.32996	0.33008	0.31980	0.01016
	$E[W_r]$	10.52572	10.54373	10.56173	10.42307	0.12065
0.6	$E[N_s]$	0.33122	0.33134	0.33146	0.31980	0.01154
	$E[W_r]$	10.53213	10.55058	10.56902	10.42307	0.12750
0.7	$E[N_s]$	0.33258	0.33271	0.33283	0.31980	0.01290
	$E[W_r]$	10.53818	10.55705	10.57592	10.42307	0.13398
0.8	$E[N_s]$	0.33400	0.33412	0.33425	0.31980	0.01432
	$E[W_r]$	10.53549	10.55473	10.57396	10.42307	0.13165
0.9	$E[N_s]$	0.33545	0.33557	0.33569	0.31980	0.01577
	$E[W_r]$	10.53799	10.55515	10.57231	10.42307	0.13208
1.0	$E[N_s]$	0.33697	0.33712	0.33726	0.31980	0.01731
	$E[W_r]$	10.54580	10.56103	10.57625	10.42307	0.13795

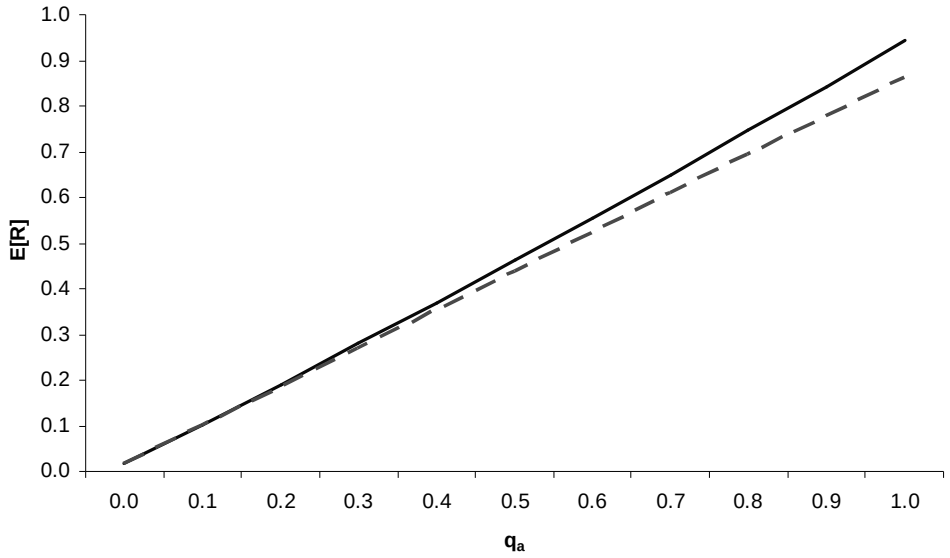

Figure 4.1 Mean orbit length for $\lambda = 2$: approximated (- - -), simulated (—).

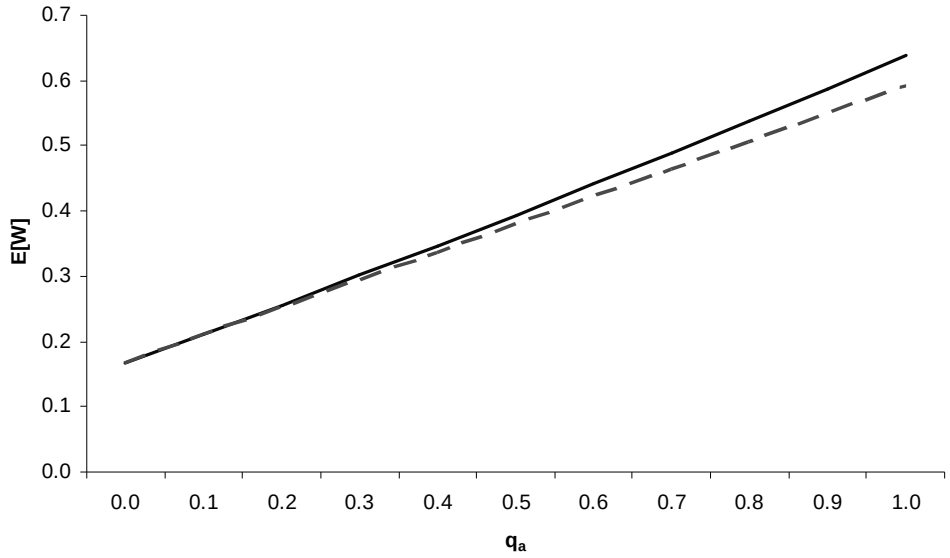

Figure 4.2 Mean sojourn time for $\lambda = 2$: approximated (- - -), simulated (—).

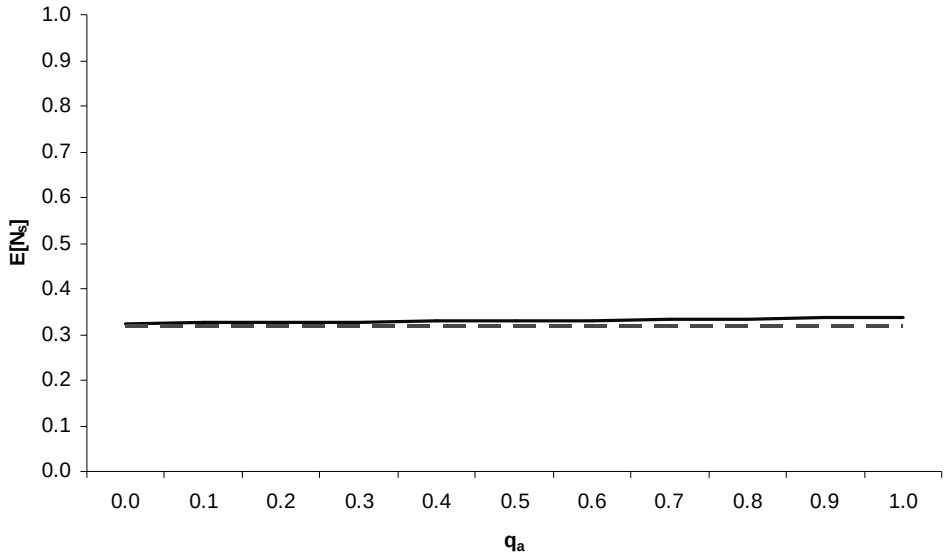

Figure 4.3 Mean number of customers at the servers for $\lambda = 2$: approximated (- - -), simulated (—).

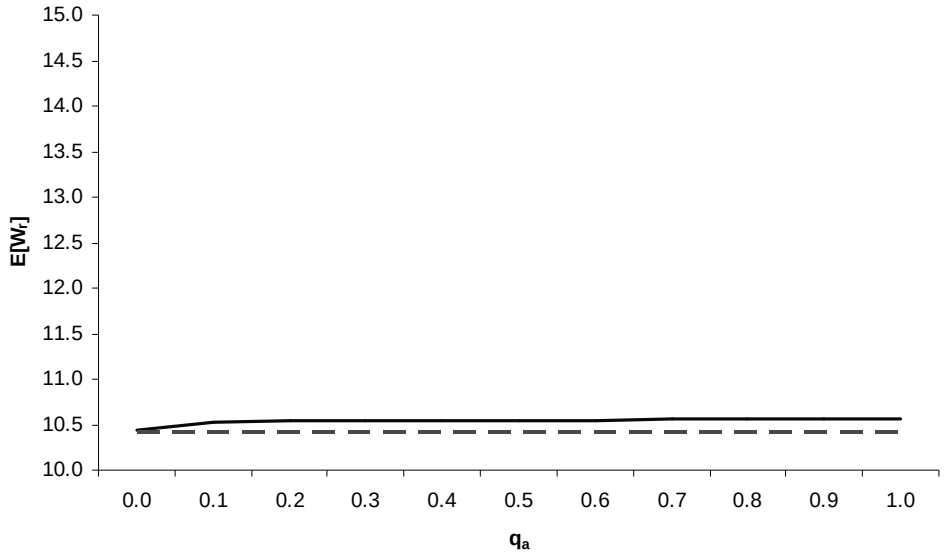

Figure 4.4 Mean time in orbit for $\lambda = 2$: approximated (- - -), simulated (—).

Table 4.4 Numerical results for unreliable M/M/2 retrial queue with $\lambda = 4$.

q_a		Lower CI Limit	Midpoint	Upper CI Limit	Approximation	Abs. Diff.
0.0	$E[R]$	0.03289	0.03329	0.03369	0.03333	0.00004
	$E[W]$	0.15498	0.15508	0.15519	0.15523	0.00015
0.1	$E[R]$	0.58545	0.58726	0.58908	0.57025	0.01701
	$E[W]$	0.29476	0.29522	0.29567	0.28951	0.00570
0.2	$E[R]$	1.16627	1.16850	1.17074	1.10717	0.06133
	$E[W]$	0.44167	0.44223	0.44279	0.42374	0.01849
0.3	$E[R]$	1.77391	1.77629	1.77867	1.64409	0.13220
	$E[W]$	0.59535	0.59593	0.59652	0.55796	0.03797
0.4	$E[R]$	2.41168	2.41476	2.41783	2.18101	0.23375
	$E[W]$	0.75667	0.75739	0.75810	0.69219	0.06519
0.5	$E[R]$	3.08009	3.08419	3.08829	2.71792	0.36627
	$E[W]$	0.92568	0.92663	0.92759	0.82642	0.10021
0.6	$E[R]$	3.78274	3.78787	3.79300	3.25484	0.53303
	$E[W]$	1.10329	1.10453	1.10576	0.96065	0.14387
0.7	$E[R]$	4.52689	4.53178	4.53666	3.79176	0.74001
	$E[W]$	1.29137	1.29258	1.29379	1.09488	0.19770
0.8	$E[R]$	5.31320	5.32039	5.32757	4.32868	0.99171
	$E[W]$	1.49016	1.49190	1.49364	1.22911	0.26279
0.9	$E[R]$	6.14850	6.15499	6.16147	4.86560	1.28939
	$E[W]$	1.70133	1.70282	1.70431	1.36334	0.33948
1.0	$E[R]$	7.03039	7.03764	7.04489	5.40252	1.63512
	$E[W]$	1.92421	1.92596	1.92770	1.49757	0.42839

Table 4.5 Numerical results for unreliable M/M/2 retrial queue with $\lambda = 4$.

q_a		Lower CI Limit	Midpoint	Upper CI Limit	Approximation	Abs. Diff.
0.0	$E[N_s]$	0.59087	0.59104	0.59121	0.58777	0.00327
	$E[W_r]$	11.29144	11.37490	11.45836	11.34230	0.03260
0.1	$E[N_s]$	0.59732	0.59748	0.59763	0.58777	0.00971
	$E[W_r]$	11.45817	11.47780	11.49743	11.34230	0.13550
0.2	$E[N_s]$	0.60412	0.60428	0.60443	0.58777	0.01651
	$E[W_r]$	11.48961	11.50620	11.52279	11.34230	0.16390
0.3	$E[N_s]$	0.61116	0.61132	0.61149	0.58777	0.02355
	$E[W_r]$	11.53750	11.54770	11.55790	11.34230	0.20540
0.4	$E[N_s]$	0.61844	0.61863	0.61881	0.58777	0.03085
	$E[W_r]$	11.58087	11.58980	11.59873	11.34230	0.24750
0.5	$E[N_s]$	0.62602	0.62622	0.62642	0.58777	0.03845
	$E[W_r]$	11.61671	11.62590	11.63509	11.34230	0.28360
0.6	$E[N_s]$	0.63400	0.63421	0.63442	0.58777	0.04644
	$E[W_r]$	11.65429	11.66393	11.67356	11.34230	0.32163
0.7	$E[N_s]$	0.64232	0.64252	0.64272	0.58777	0.05475
	$E[W_r]$	11.70235	11.70923	11.71610	11.34230	0.36693
0.8	$E[N_s]$	0.65105	0.65124	0.65142	0.58777	0.06347
	$E[W_r]$	11.74962	11.75730	11.76498	11.34230	0.41500
0.9	$E[N_s]$	0.66015	0.66036	0.66057	0.58777	0.07259
	$E[W_r]$	11.80197	11.80893	11.81588	11.34230	0.46663
1.0	$E[N_s]$	0.66975	0.66994	0.67014	0.58777	0.08217
	$E[W_r]$	11.85769	11.86385	11.87001	11.34230	0.52155

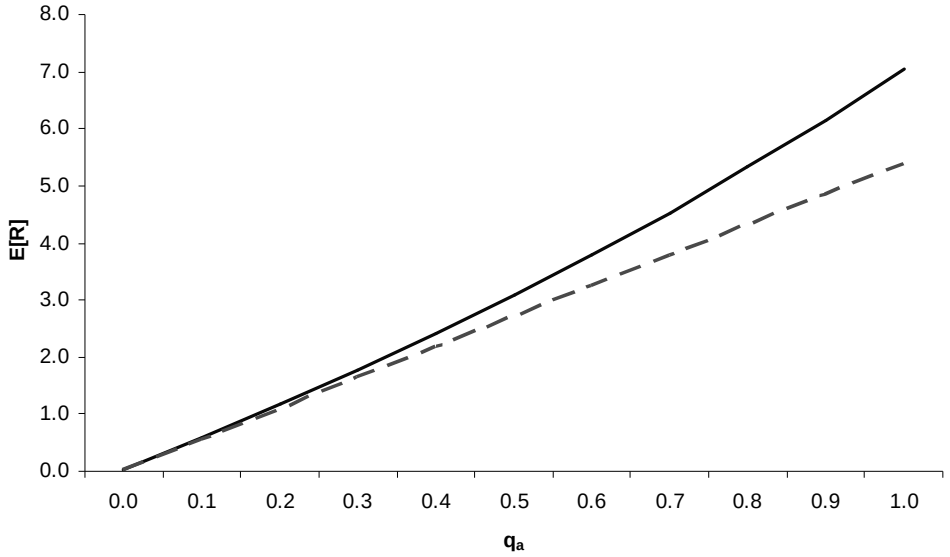

Figure 4.5 Mean orbit length for $\lambda = 4$: approximated (- - -), simulated (—).

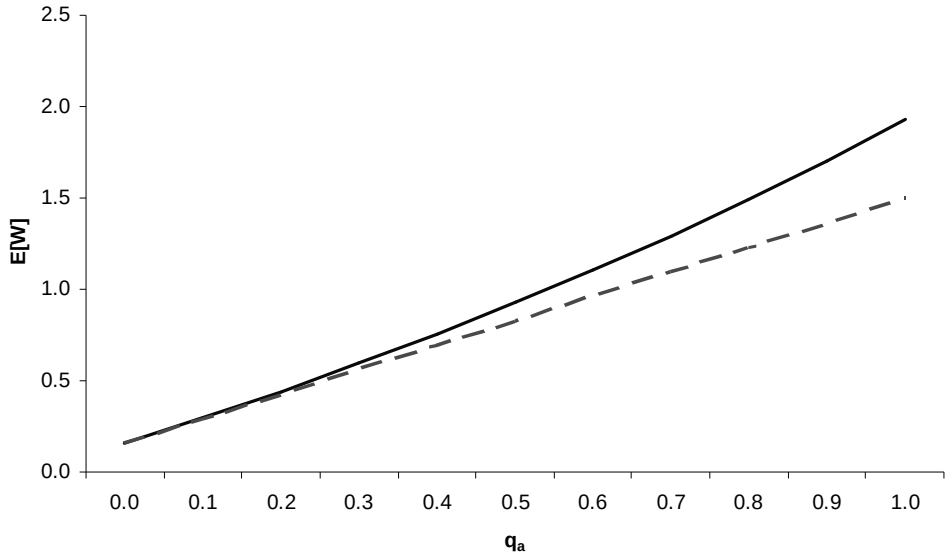

Figure 4.6 Mean sojourn time for $\lambda = 4$: approximated (- - -), simulated (—).

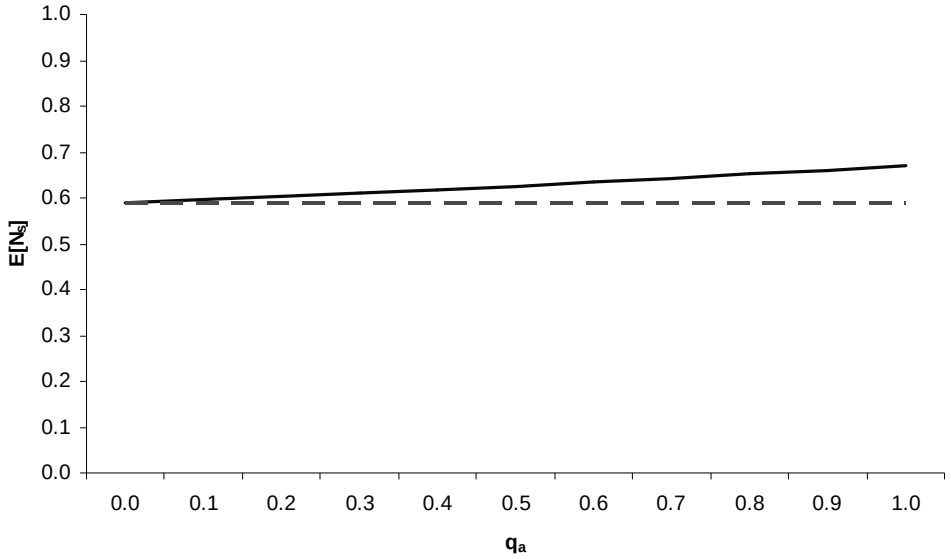

Figure 4.7 Mean number of customers at the servers for $\lambda = 4$: approximated (- - -), simulated (—).

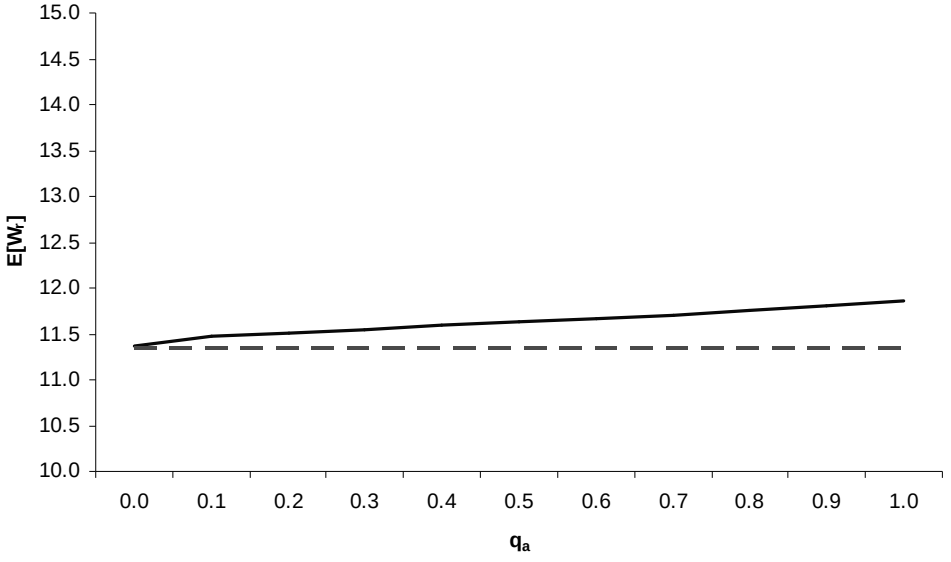

Figure 4.8 Mean time in orbit for $\lambda = 4$: approximated (- - -), simulated (—).

Table 4.6 Numerical results for unreliable M/M/2 retrial queue with $\lambda = 6$.

q_a		Lower CI Limit	Midpoint	Upper CI Limit	Approximation	Abs. Diff.
0.0	$E[R]$	0.05004	0.05054	0.05104	0.05000	0.00054
	$E[W]$	0.14139	0.14148	0.14157	0.14156	0.00008
0.1	$E[R]$	1.62920	1.63122	1.63325	1.55609	0.07513
	$E[W]$	0.40701	0.40733	0.40765	0.39257	0.01475
0.2	$E[R]$	3.34038	3.34448	3.34858	3.06218	0.28230
	$E[W]$	0.69473	0.69540	0.69606	0.64359	0.05181
0.3	$E[R]$	5.20687	5.21205	5.21722	4.56826	0.64378
	$E[W]$	1.00853	1.00937	1.01020	0.89460	0.11476
0.4	$E[R]$	7.24197	7.24951	7.25705	6.07435	1.17516
	$E[W]$	1.35062	1.35182	1.35303	1.14562	0.20621
0.5	$E[R]$	9.48227	9.49357	9.50486	7.58044	1.91312
	$E[W]$	1.72713	1.72896	1.73078	1.39663	0.33232
0.6	$E[R]$	11.96474	11.97630	11.98786	9.08653	2.88977
	$E[W]$	2.14406	2.14590	2.14773	1.64765	0.49825
0.7	$E[R]$	14.73599	14.74998	14.76396	10.59262	4.15736
	$E[W]$	2.60978	2.61191	2.61403	1.89866	0.71325
0.8	$E[R]$	17.83898	17.85718	17.87537	12.09870	5.75847
	$E[W]$	3.13117	3.13394	3.13671	2.14968	0.98426
0.9	$E[R]$	21.37754	21.39643	21.41531	13.60479	7.79163
	$E[W]$	3.72507	3.72786	3.73064	2.40069	1.32716
1.0	$E[R]$	25.44877	25.47183	25.49488	15.11088	10.36094
	$E[W]$	4.40802	4.41145	4.41489	2.65171	1.75975

Table 4.7 Numerical results for unreliable M/M/2 retrial queue with $\lambda = 6$.

q_a		Lower CI Limit	Midpoint	Upper CI Limit	Approximation	Abs. Diff.
0.0	$E[N_s]$	0.80210	0.80228	0.80246	0.79935	0.00293
	$E[W_r]$	12.48687	12.58335	12.67983	12.51015	0.07320
0.1	$E[N_s]$	0.81665	0.81681	0.81697	0.79935	0.01746
	$E[W_r]$	12.71181	12.72520	12.73859	12.51015	0.21505
0.2	$E[N_s]$	0.83192	0.83209	0.83226	0.79935	0.03274
	$E[W_r]$	12.82172	12.83100	12.84028	12.51015	0.32085
0.3	$E[N_s]$	0.84819	0.84836	0.84853	0.79935	0.04901
	$E[W_r]$	12.95080	12.95820	12.96560	12.51015	0.44805
0.4	$E[N_s]$	0.86560	0.86579	0.86597	0.79935	0.06644
	$E[W_r]$	13.08647	13.09380	13.10113	12.51015	0.58365
0.5	$E[N_s]$	0.88416	0.88436	0.88457	0.79935	0.08501
	$E[W_r]$	13.23764	13.24510	13.25256	12.51015	0.73495
0.6	$E[N_s]$	0.90426	0.90447	0.90468	0.79935	0.10512
	$E[W_r]$	13.40492	13.41205	13.41918	12.51015	0.90190
0.7	$E[N_s]$	0.92577	0.92599	0.92621	0.79935	0.12664
	$E[W_r]$	13.59890	13.60425	13.60960	12.51015	1.09410
0.8	$E[N_s]$	0.94910	0.94935	0.94959	0.79935	0.14999
	$E[W_r]$	13.81498	13.82035	13.82572	12.51015	1.31020
0.9	$E[N_s]$	0.97473	0.97497	0.97522	0.79935	0.17562
	$E[W_r]$	14.06239	14.06728	14.07216	12.51015	1.55713
1.0	$E[N_s]$	1.00278	1.00303	1.00329	0.79935	0.20368
	$E[W_r]$	14.35586	14.36155	14.36724	12.51015	1.85140

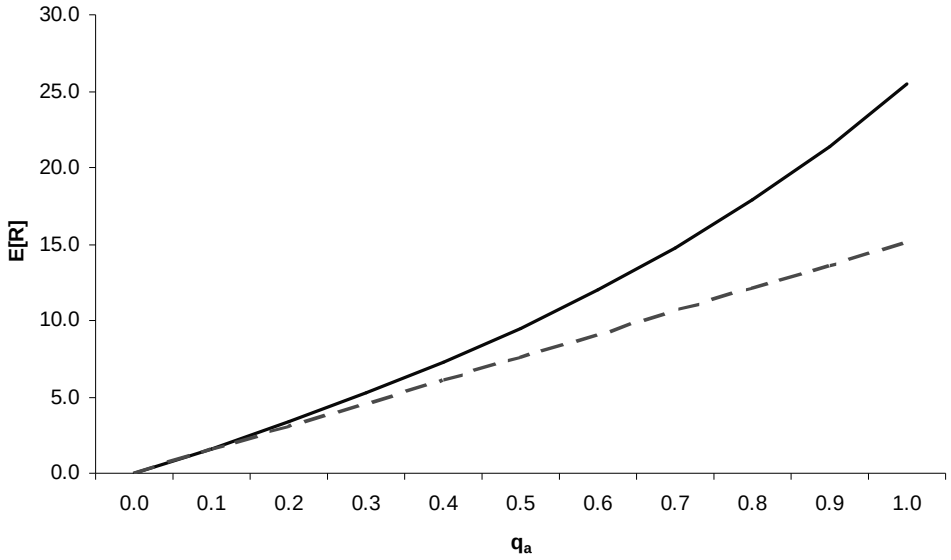

Figure 4.9 Mean orbit length for $\lambda = 6$: approximated (- - -), simulated (—).

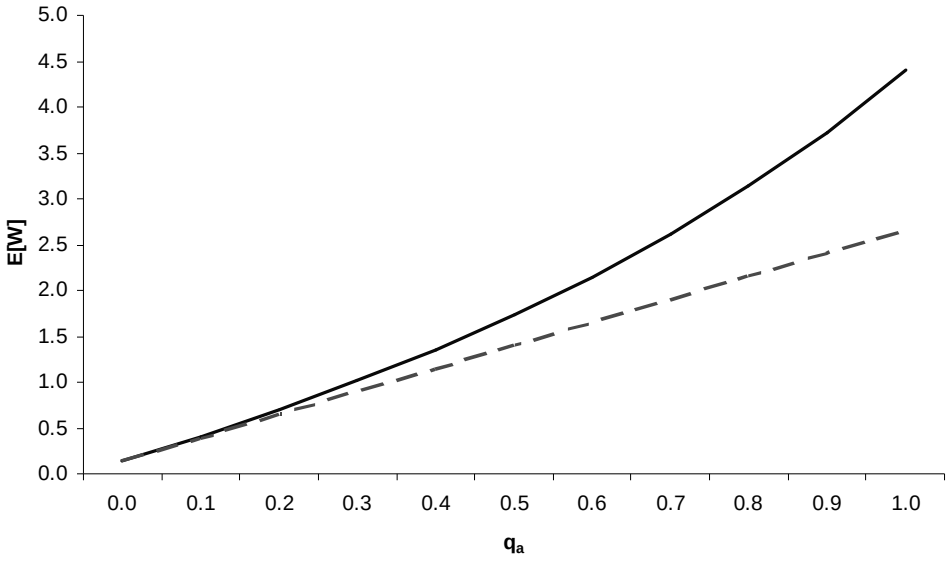

Figure 4.10 Mean sojourn time for $\lambda = 6$: approximated (- - -), simulated (—).

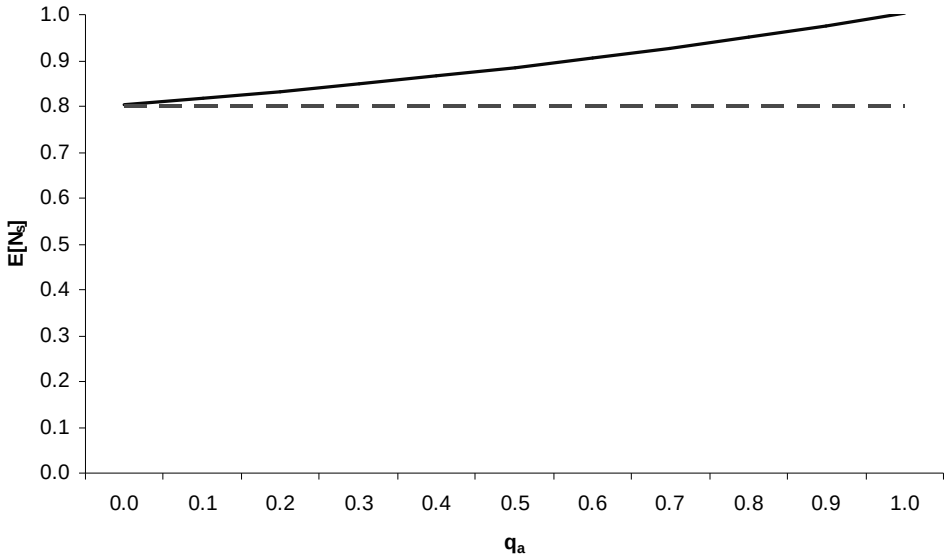

Figure 4.11 Mean number of customers at the servers for $\lambda = 6$: approximated (- - -), simulated (—).

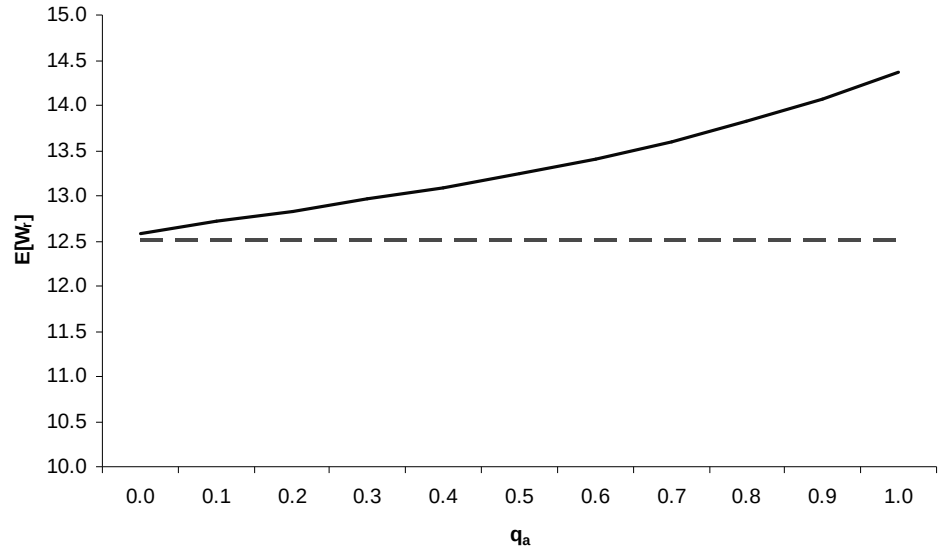

Figure 4.12 Mean time in orbit for $\lambda = 6$: approximated (- - -), simulated (—).

Table 4.8 Mean orbit size and sojourn time as a function of q_f.

q_f		Lower CI Limit	Midpoint	Upper CI Limit	Approximation	Abs. Diff.
0.0	$E[R]$	0.44212	0.44345	0.44477	0.42307	0.02038
	$E[W]$	0.38392	0.38459	0.38526	0.37144	0.01315
0.1	$E[R]$	0.44594	0.44719	0.44844	0.42640	0.02079
	$E[W]$	0.38589	0.38651	0.38713	0.37310	0.01341
0.2	$E[R]$	0.44935	0.45051	0.45167	0.42974	0.02077
	$E[W]$	0.38765	0.38821	0.38877	0.37477	0.01344
0.3	$E[R]$	0.45323	0.45445	0.45567	0.43307	0.02138
	$E[W]$	0.38956	0.39019	0.39081	0.37644	0.01375
0.4	$E[R]$	0.45688	0.45803	0.45918	0.43640	0.02163
	$E[W]$	0.39142	0.39201	0.39260	0.37810	0.01391
0.5	$E[R]$	0.46042	0.46157	0.46272	0.43974	0.02183
	$E[W]$	0.39328	0.39384	0.39440	0.37977	0.01407
0.6	$E[R]$	0.46377	0.46503	0.46629	0.44307	0.02196
	$E[W]$	0.39490	0.39553	0.39615	0.38144	0.01409
0.7	$E[R]$	0.46668	0.46793	0.46918	0.44640	0.02153
	$E[W]$	0.39640	0.39705	0.39769	0.38310	0.01395
0.8	$E[R]$	0.47128	0.47260	0.47392	0.44973	0.02287
	$E[W]$	0.39874	0.39941	0.40007	0.38477	0.01464
0.9	$E[R]$	0.47490	0.47601	0.47712	0.45307	0.02294
	$E[W]$	0.40057	0.40115	0.40172	0.38644	0.01471
1.0	$E[R]$	0.47828	0.47957	0.48086	0.45640	0.02317
	$E[W]$	0.40234	0.40297	0.40361	0.38810	0.01487

Table 4.9 Mean number of customers at the servers and time in orbit as a function of q_f.

q_f		Lower CI Limit	Midpoint	Upper CI Limit	Approximation	Abs. Diff.
0.0	$E[N_s]$	0.32961	0.32974	0.32986	0.31980	0.00994
	$E[W_r]$	10.53275	10.55070	10.56865	10.42307	0.12763
0.1	$E[N_s]$	0.32967	0.32978	0.32990	0.31980	0.00998
	$E[W_r]$	10.52989	10.54828	10.56666	10.42307	0.12521
0.2	$E[N_s]$	0.32971	0.32983	0.32995	0.31980	0.01003
	$E[W_r]$	10.52782	10.54445	10.56108	10.42307	0.12138
0.3	$E[N_s]$	0.32978	0.32991	0.33003	0.31980	0.01011
	$E[W_r]$	10.53358	10.55190	10.57022	10.42307	0.12883
0.4	$E[N_s]$	0.32983	0.32995	0.33007	0.31980	0.01015
	$E[W_r]$	10.52474	10.54353	10.56231	10.42307	0.12045
0.5	$E[N_s]$	0.32984	0.32996	0.33008	0.31980	0.01016
	$E[W_r]$	10.52572	10.54373	10.56173	10.42307	0.12065
0.6	$E[N_s]$	0.32997	0.33010	0.33022	0.31980	0.01030
	$E[W_r]$	10.52460	10.54415	10.56370	10.42307	0.12108
0.7	$E[N_s]$	0.32996	0.33009	0.33021	0.31980	0.01029
	$E[W_r]$	10.52551	10.54498	10.56444	10.42307	0.12190
0.8	$E[N_s]$	0.33004	0.33017	0.33029	0.31980	0.01037
	$E[W_r]$	10.52866	10.54865	10.56864	10.42307	0.12558
0.9	$E[N_s]$	0.33007	0.33020	0.33033	0.31980	0.01040
	$E[W_r]$	10.52817	10.54733	10.56648	10.42307	0.12425
1.0	$E[N_s]$	0.33010	0.33021	0.33033	0.31980	0.01041
	$E[W_r]$	10.52275	10.54275	10.56275	10.42307	0.11968

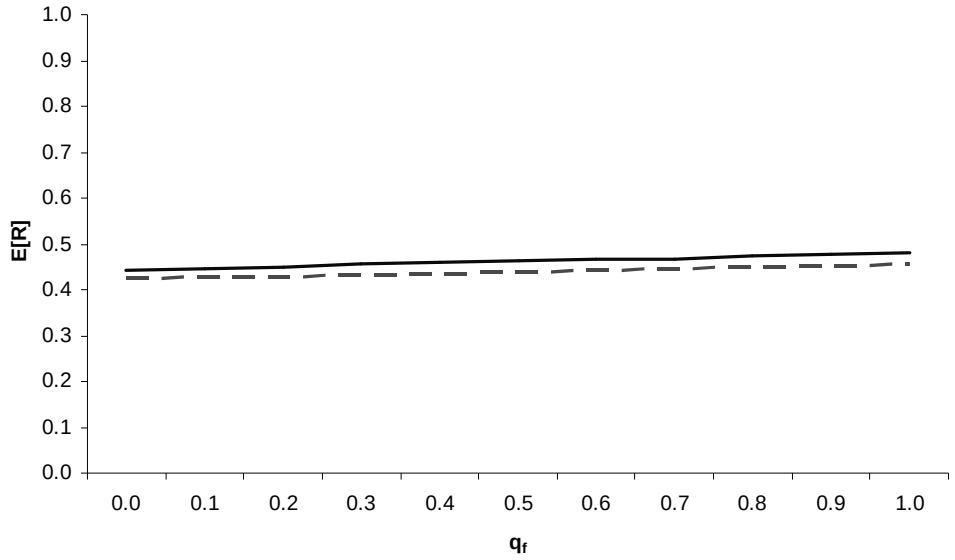

Figure 4.13 Mean orbit length for varying values of q_f: approximated (- - -), simulated (—).

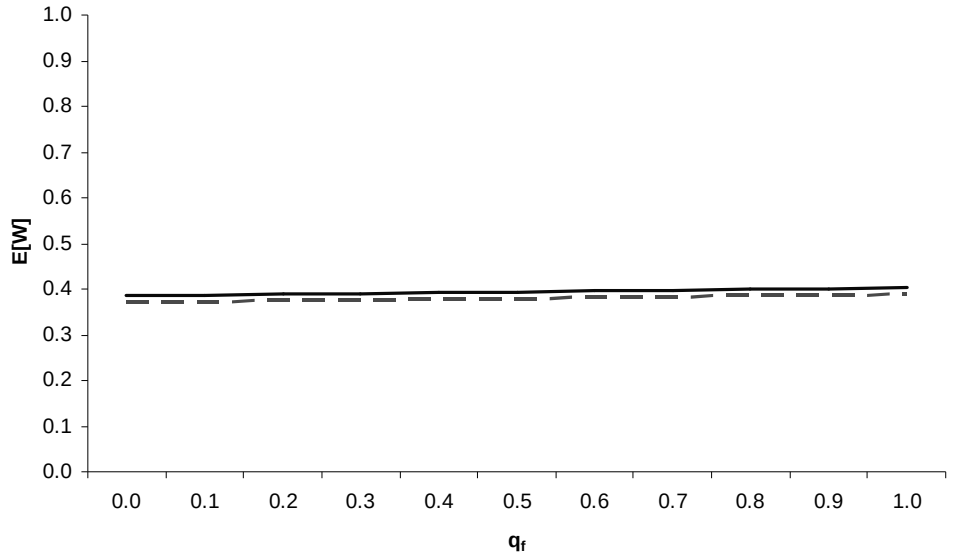

Figure 4.14 Mean sojourn time for varying values of q_f: approximated (- - -), simulated (—).

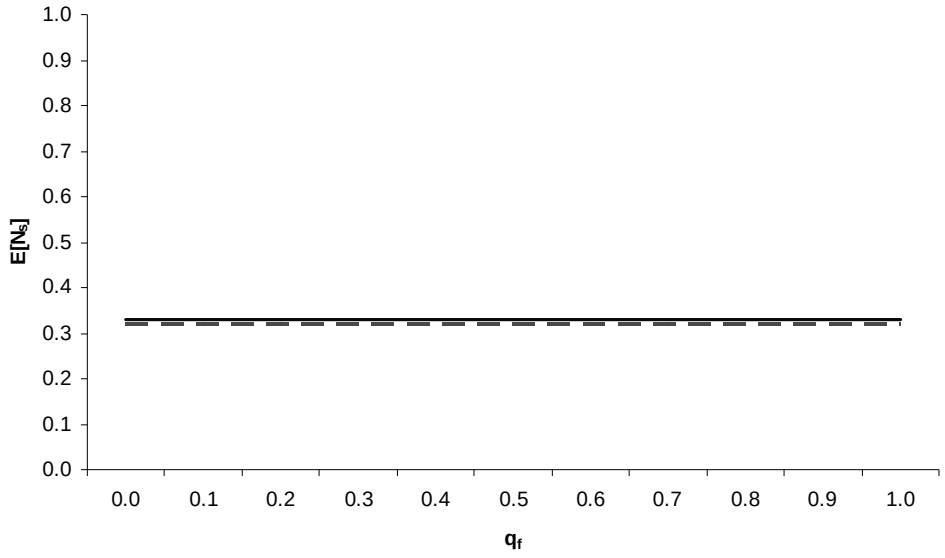

Figure 4.15 Mean number of customers at the servers for varying values of q_f: approximated (- - -), simulated (—).

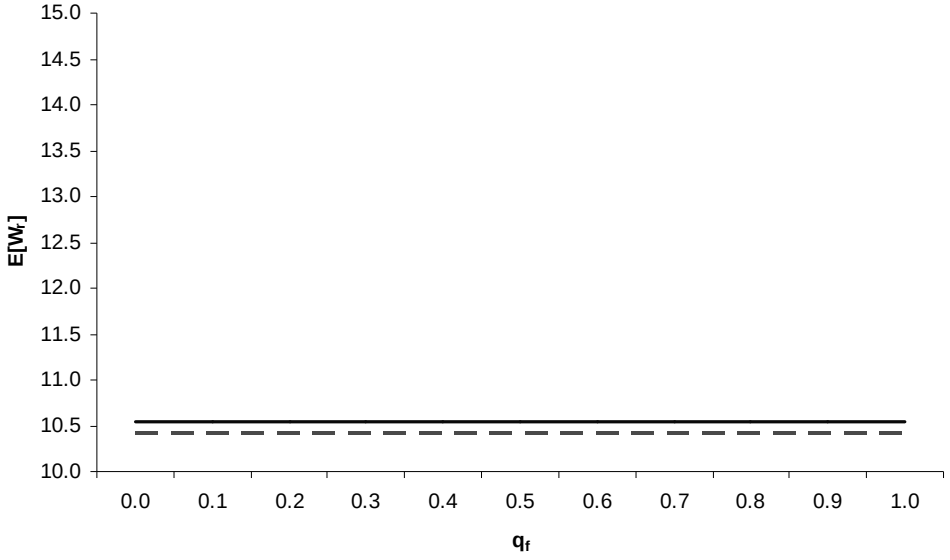

Figure 4.16 Mean time in orbit for varying values of q_f: approximated (- - -), simulated (—).

4.4 Summary of Results

The phase-merging approximation performs well for the case $\lambda = 2$ as the absolute difference for each of the four measures remains low for all values of q_a. Note that the mean orbit length remains below one as q_a increases. This implies that the rates of transition that correspond to retrial successes are relatively low which satisfies the assumption that rates within levels of the orbit must be greater than those between levels. For each value of q_a, the approximations for $E[N_s]$ and $E[W_r]$ remained constant due to their strong dependence on the service rate, μ, which remained constant for all experiments. The simulation results showed that there was an extremely gradual increase in $E[N_s]$ for increasing values of q_a, but $E[W_r]$ essentially remained constant.

For the case $\lambda = 4$, we notice that once the value of q_a exceeds 0.4, the approximation performs poorly with respect to $E[R]$ and $E[W]$. This can be explained in two ways. First, the rate λq_a, which corresponds to a retrial orbit entry due to blocking upon arrival, approaches λ as $q_a \rightarrow 1$. Second, the retrial orbit grows in size as q_a increases, forcing the retrial success transition rates to become large. Both scenarios increase the flows between levels, threatening to violate the assumption that must hold for accurate approximations. With regards to $E[N_s]$ and $E[W_r]$, for $\lambda = 4$, we again observe that the approximations remain constant for all values of q_a. Furthermore, the simulated values of $E[N_s]$ and $E[W_r]$ exhibit a gradual increase, however the approximation remains effective for most values of q_a.

We see in the case of $\lambda = 6$ the same phenomenon as $\lambda = 4$, only that the approximations for the four measures worsen once q_a is greater than 0.2. We also note a more substantial growth in the simulated results for $E[N_s]$ and $E[W_r]$ with the approximation only effective for the lower values of q_a. For all values of λ tested, we conclude that q_a has a very limited effect on $E[N_s]$ and $E[W_r]$. Varying the service rate, μ, is likely to have a more pronounced impact on the two measures.

Figures 4.13 through 4.16 indicate that the phase-merging algorithm is reasonably effective in approximating $E[R]$, $E[W]$, $E[N_s]$ and $E[W_r]$ for all values of q_f when q_a is fixed at 0.5. Both the simulation results and approximations for $E[R]$ and $E[W]$ increase in a linear fashion as q_f increases, although the growth is very gradual. This is due to the fact that the assumed failure rate was small ($\xi = 0.01$). In contrast, the simulation results and approximations for $E[N_s]$ and $E[W_r]$ remain constant for all values of q_f. This is explained by their dependence on the service rate which was constant in all of the experiments. We conclude that for low rates of failure, the method of approximation is good, so long as the transition intensities within orbit levels are significantly greater than those between levels.

Examining Figures 4.1, 4.2, 4.5, 4.6, 4.9 and 4.10, we notice that the mean orbit length and sojourn time grow exponentially while the approximations exhibit linear growth as q_a increases. This effect is more pronounced in the cases $\lambda = 4$ and $\lambda = 6$. We conclude, therefore, that the method of approximation is effective when q_a is relatively small (less than 0.5) for the cases when μ is not significantly greater than λ. When μ is significantly greater than λ, the approximation is effective for most values of q_a.

Ultimately, systems that exhibit a low number of customers in the retrial orbit while in the steady-state can be effectively analyzed by the phase-merging algorithm. Although an alternative approach is to simply simulate the system, the time required to run a simulation can be substantial. Depending on the parameters chosen for each experiment, the simulation experiments ran for up to 90 minutes while the approximation method produced effective results in less than a second, provided the assumptions are met. Thus, the phase-merging algorithm can be used to quickly and efficiently estimate the various queueing performance measures of interest. These measures can potentially be used to optimally design, staff, operate or maintain these types of unreliable multi-server retrial queueing systems.

5. Conclusions and Future Research

The primary aim of this research was to provide a formal analysis of the unreliable M/M/2 retrial queueing system. A review of the existing retrial queueing literature revealed that very few results exist for unreliable multi-server retrial queues. Under the assumptions of our model, it was shown that deriving the joint steady-state probability distribution of the number of customers in the orbit and status of the servers via a direct analytical approach is extremely difficult. This complexity is due to transitions that correspond to changes in the number of customers in the orbit (i.e. retrial successes, blocking upon first arrival, or being preempted by a server failure). Therefore, we resorted to an approximate analysis to obtain the joint steady-state distribution of the number of customers in the retrial orbit and the status of the servers in the unreliable M/M/2 retrial queue.

Applying a phase-merging algorithm due to Korolyuk and Korolyuk [23] and Courtois [13], it was found that the aggregated model is analogous to an M/M/∞ queue. Solving the balance equations for the aggregated model, we showed that the steady-state orbit length is approximately Poisson distributed. Using this result, we approximated the joint probability distribution of the number of customers in the orbit and the status of the servers. This enabled us to derive approximate expressions for the steady-state mean orbit length, mean number of customers in service, mean number of customers in the system, the mean system sojourn time, and the mean orbit sojourn time. In lieu of an exact benchmark, the accuracy of these approximations was assessed using a discrete-event simulation model. The results indicated that, under moderate assumptions (i.e. the transition intensities that flow between states within a given level of the orbit must be significantly greater than those intensities that flow between orbit levels), the algorithm produces effective approximations. However, if the assumptions are violated, the method may perform very poorly. For the approximation to be useful, q_a should not exceed 0.5 when

μ is not significantly greater than λ. In contrast, when μ is significantly greater than λ, the method is effective for most values of q_a. For systems exhibiting these characteristics, the phase-merging approximation will be of value.

It can be argued that the necessary assumptions for the phase-merging algorithm are extremely restrictive. However, to the best of our knowledge, only three other works have attempted to derive results for the unreliable multi-server retrial queue. In this sense, we feel that the model, with its assumptions, does contribute significantly to our understanding of the dynamics of unreliable, multi-server retrial queues.

With regards to future research, a formal stability analysis of the system will provide additional insight into the dynamics of the model. Once this is accomplished, an extension of this work to the more general case of unreliable M/M/c retrial queueing systems with $c > 2$ should be considered. Matrix-analytic methods may be used if it can be shown that the infinitesimal generator matrix of $\{(R(t), X(t)) : t \geq 0\}$ possesses a quasi-birth-death structure. A multitude of queueing variants can be considered in this model, including the case of general service time distributions, balking, reneging, feedback or a FCFS discipline for the retrial orbit. A version of the model that does not allow for loss could also be of great value in applications where customers do not have the option of departing the system. The model may also be extended to a network of unreliable multi-server retrial queues in which the phase-merging algorithm can be applied.

Bibliography

1. Abramov, V. M. (2006). Analysis of multiserver retrial queueing system: A martingale approach and an algorithm of solution. *Annals of Operations Research*, **141**, 19-50.

2. Aissani, A. (1988). On the M/G/1/1 queueing system with repeated orders and unreliable server. *Journal of Technology*, **6**, 98-123 (in French).

3. Aissani, A. (1993). Unreliable queueing with repeated orders. *Microelectronics and Reliability*, **33** (14), 2093-2106.

4. Aissani, A. (1994). A retrial queue with redundancy and unreliable server. *Queueing Systems*, **17** (3-4), 431-449.

5. Aissani, A. and J. R. Artalejo (1998). On the single server retrial queue subject to breakdowns. *Queueing Systems*, **30** (3-4), 309-321.

6. Almasi, B., J. Roszik and J. Sztrik (2005). Homogeneous finite-source retrial queues with server subject to breakdowns and repairs. *Mathematical and Computer Modelling*, **42**, 673-682.

7. Artalejo, J. R. (1994). New results in retrial queueing systems with breakdowns of the servers. *Statistica Neerlandica*, **48** (1) 23-36.

8. Artalejo, J. R. (1997). Analysis of an M/G/1 queue with constant repeated attempts and server vacations. *Computers and Operations Research*, **24** (6), 493-504.

9. Artalejo, J. R. (1999). Accessible Bibliography on Retrial Queues. *Mathematical and Computer Modelling*, **30**, 1-6.

10. Avi-Itzhak B. and P. Naor (1963). Some queueing problems with the service station subject to breakdowns. *Operations Research*, **11**, 303-320.

11. Clos, C. (1948). An aspect of the dialing behaviour of subscribers and its effect on the trunk plant. *Bell Systems Technical Journal*, **27**, 424-445.

12. Cohen, J. W. (1957). Basic problems of telephone traffic theory and the influence of repeated calls. *Philips Telecommunication Review*, **18** (2), 49-101.

13. Courtois, P. J. (1975). Decomposability, instabilities, and saturation in multiprogramming systems. *Communications of the ACM*, **18** (7), 371-377.

14. Cox, D. R. (1955). The analysis of non-Markovian stochastic processes by the inclusion of supplemental variables. *Proc. Camb. Phil. Soc.*, **51**, 433-441.

15. Djellab, N. V. (2002). On the M/G/1 retrial queue subjected to breakdowns. *RAIRO Operations Research*, **36** (4), 299-310.

16. Falin, G. (1990). A survey of retrial queues. *Queueing Systems*, **7** (2) 127-168.

17. Falin, G. and J. G. C. Templeton (1997). *Retrial Queues*. Chapman & Hall, London.

18. Gharbi, Nawel and M. Ioualalen (2006). GSPN analysis of retrial systems with servers breakdowns and repairs. *Applied Mathematics and Computation*, **174**, 1151-1168.

19. Gray, W. J., P. P. Wang, and M. Scott (2004). A queueing model with multiple types of server breakdowns. *Opsearch*, **39** (5-6), 281-295.

20. Hanschke, T. (1987). Explicit formulas for the characteristics of the M/M/2/2 queue with repeated attempts. *Journal of Applied Probability*, **24**, 486-494.

21. Keilson, J., H. Cozzolino, and A. Young (1968) A service system with unfilled requests repeated. *Operations Research*, **16** (6), 1126-1137.

22. Kendall, D. G. (1953). Stochastic processes occurring in the theory of queues and their analysis by the method of the imbedded Markov chain. *Annals of Mathematical Statistics*, **24**, 338-354.

23. Korolyuk, V. S. and V. V. Korolyuk (1999) *Stochastic Models of Systems*. Kluwer Academic Publishers, Boston.

24. Kulkarni, V. G. (1983). On queueing systems with retrials. *Journal of Applied Probability*, **20** (2), 380-389.

25. Kulkarni, V. G. and B. D. Choi (1990). Retrial queues with server subject to breakdowns and repairs. *Queueing Systems*, **7** (2), 191-208.

26. Kulkarni, V. G. and H. M. Liang (1997). Retrial queues revisited, in: *Frontiers in Queueing*, ed. J.H. Dshalalow (CRC Press, Boca Raton, FL, 1997), 19-34.

27. Kumar, B. K., S. P. Madheswari, and A. Vijayakumar (2002). The M/G/1 retrial queue with feedback and starting failures. *Applied Mathematical Modelling*, **26** (11), 1057-1075.

28. Li, Q., Y. Ying, and Y. Q. Zhao (2006). A BMAP/G/1 retrial queue with a server subject to breakdowns and repairs. *Annals of Operations Research*, **141** (1), 233-270.

29. Li, H. and T. Yang (1995). A single-server retrial queue with server vacations and a finite number of input sources. *European Journal of Operations Research*, **85** (1), 149-160.

30. Li, H. and Y. Q. Zhao (2005). A retrial queue with a constant retrial rate, server breakdowns and impatient customers. *Stochastic Models*, **21** (2-3), 531-550.

31. Li, W., D. Shi, and X. Chao (1997). Reliability analysis of M/G/1 queueing systems with server breakdowns and vactions. *Journal of Applied Probability*, **34** (2), 546-555.

32. Miller, J. O. and Honabarger, J. B. (2006). Modeling and measuring network centric warfare (NCW) with the system effectiveness analysis simulation (SEAS), in: *11th ICCRTS Coalition Command and Control in the Networked Era*, Cambridge, UK.

33. Mitrany, I. L. and B. Avi-Itzhak (1968). A many-server queue with service interruptions. *Operations Research*, **16**, 628-638.

34. Neuts, M. and D. Lucantoni (1979). A Markovian queue with N servers subject to breakdowns and repairs. *Management Science*, **25** (9), 849-861.

35. Roszik, J. and J. Sztrik (2004). Performance analysis of finite-source retrial queues with non-reliable heterogeneous servers, in: *Proceedings of the XXIV Seminar on Stability Problems of Stochastic Models*, Jurmala, Latvia.

36. Sherman, N. (2006). *Analysis and Control of Unreliable, Single-server Retrial Queues with Infinite-capacity Orbit and Normal Queue.* Ph.D. Dissertation, Air Force Institute of Technology.

37. Sherman, N. and J. Kharoufeh (2006). An M/M/1 retrial queue with unreliable server. *Operations Research Letters*, **34**, 697-705.

38. Sztrik, J. and T. Gal (1990). A recursive solution of a queueing model for a multi-terminal system subject to breakdowns. *Performance Evaluation*, **11** (1), 1-7.

39. Tang, Y. H. (1997). A single-server M/G/1 queueing system subject to breakdowns-some reliability and queueing problems. *Microelectronics and Reliability*, **37** (2), 315-321.

40. Thiruvengadam, K. (1963). Queueing with breakdowns. *Operations Research*, **11**, 62-71.

41. Wang, J., J. Cao, and Q. Li (2001). Reliability analysis of the retrial queue with server breakdowns and repairs. *Queueing Systems*, **38** (4), 363-380.

42. Wang, J. (2006). Reliability analysis of M/G/1 queues with general retrial times and server breakdowns. *Progress in Natural Science*, **16** (5), 464-473.

43. White, H. and L. S. Christie (1958). Queueing with breakdowns. *Operations Research*, **6**, 79-95.

44. Wilkinson, R. I. (1956). Theories for toll traffic engineering in the USA. *Bell Systems Technical Journal*, **35** (2), 421-507.

45. Wu, X., P. Brill, M. Hlynka, and J. Wang (2005). An M/G/1 retrial queue with balking and retrials during service. *International Journal of Operational Research*, **1** (1-2), 30-51.

46. Xueming, Y. and W. Li (2003). Availability analysis of the queueing system GI/PH/1 with server breakdowns. *Journal of Systems Science and Complexity*, **16** (2), 177-183.

47. Yang, T. and J. G. C. Templeton (1987). A survey on retrial queues. *Queueing Systems*, **2** (3), 203-233.

48. Yang, T. and H. Li (1994). The M/G/1 retrial queue with the server subject to starting failures. *Queueing Systems*, **16** (1-2), 83-96.

49. Yuan, X. and W. Lei (2003). Availability analysis of the queueing system GI/PH/1 with server breakdowns. *Journal of Systems Science and Complexity*, **16** (2), 177-183.

Appendix A. MATLAB® Code: Reliable M/M/2 Case

```
%************************************************************************
%AUTHOR: Lt Brian P Crawford
%        AFIT/ENS/GOR07-M
%        March 2007
%This function calculates the sum of a hypergeometric series given four
%parameters: a, b, c, and x.  It outputs the sum to a variable called Fnew.
%This sum is further used to calculate the mean queue length of a reliable
%M/M/2 retrial queue.  This function is called within another function
%titled MeanQueueLength as a means to perform the calculation.
%************************************************************************
function [Fnew] = Hypergeometric(a,b,c,x)

Fnew=0; %Initilization values
Fold=1;
j=1;

while abs(Fnew-Fold) > 10^-9 %since this is an infinite sum this condition
    %will cause the loop to stop once the difference of sums obtained in
    %consecutive iterations is less than 10^-9

    Fold=Fnew;
    product=1;

    %Note that MATLAB will not allow an indexing to begin with zero
    for k=1:j-1;
        product=product*(((a+k-1)*(b+k-1))/(c+k-1)); %calculates the
        %product portion of the series
    end
    Fnew=Fnew+((x^(j-1))/(factorial(j-1)))*product; %calculates the sum
    j=j+1;
end

end
```

```
%************************************************************************
%AUTHOR: Lt Brian P Crawford
%        AFIT/ENS/GOR07-M
%        March 2007
%This function calculates the mean queue length of a reliable M/M/2 retrial
%queue given an arrival rate and a retrial rate.  The service rate is
%assumed to be exponential with mean 1.  This function calls another
%function titled Hypergeometric which returns a sum used in the calculation
%of the mean queue length.  Since the formula is complicated and rather
%involved variables are used to represent pieces of it and then it is put
%together under the variable N.  The blocking probability is also
%calculated.
%************************************************************************
function MeanQueueLength(lambda,theta)

a=(2*lambda+1+sqrt(4*lambda+1))/(2*theta);

b=(2*lambda+1-sqrt(4*lambda+1))/(2*theta);

c=(2+3*lambda+2*theta)/(2*theta);

[Fnew]=Hypergeometric(a,b,c,lambda/2);
A=Fnew;
[Fnew]=Hypergeometric(a+1,b+1,c+1,lambda/2);
B=Fnew;

%The expressions are extremely involved so they are broken up into pieces
%that represent numerators, denominators, etc.

g=(lambda^3)/(2+3*lambda+2*theta)*(B/A);

N1=(1+theta)/theta;
N2=lambda^3 + g*(lambda^2 - 2*lambda + 2);
N3=(2-lambda)*(2 + lambda + g);

R=N1*(N2/N3) %mean queue length

Bl=(lambda^2 + g*(lambda - 1))/(2 + lambda + g) %blocking probability
```

Appendix B. MATLAB® Code: Unreliable M/M/2 Case

```
1
2 %************************************************************************
3 %AUTHOR: Lt Brian P Crawford
4 %        AFIT/ENS/GOR07-M
5 %        March 2007
6 %This function approximates the steady-state probabilities of the M/M/2
7 %retrial queue with servers subject to breakdowns and repairs.  User inputs
8 %arrival rate 'l', service rate 'm', failure rate 'x', repair rate 'a' and
9 %retrial rate 'theta' as well probabilities 'qa' and 'qf', where 'qa' is
10 %the probability an arriving customer stays in the system when both servers
11 %are inaccessible and 'qf' is the probability a customer preempted by a
12 %server failure remains in the system. For the input 'OrbitSize' a large
13 %integer (e.g. 1000) should be chosen. The function outputs the performance
14 %charateristics to include mean orbit length, N; expected number at the
15 %servers, Ns; long-run average number of customers in system, L; long-run
16 %average time waiting in the system, W; and the time spent in orbit.
17 %************************************************************************
18
19 function ClassProbs(l,m,x,a,theta,qa,qf,OrbitSize)
20
21 format long
22
23 Den = m^2*x^2*l+6*m^2*x^2*a+2*m^3*x^2+2*m^2*x^3+6*a^2*m*x*l ...
24 +2*m*a^3*l+4*l*a^2*m^2+l^2*a^3+3*l^2*a^2*m+l^3*a^2+2*x*l^2*a^2 ...
25 +4*m^3*x*a+6*a^2*m^2*x+2*m^2*a^3+2*m^3*a^2+2*m*x*l^2*a+6*m^2*x*l*a ...
26 +4*a*m*x^2*l;
27
28 %'A' corresponds to p_1|i
29 A=2*(a+l+x+m)*a^2*m^2/Den;
30
31 %'B' corresponds to p_2|i
32 B=2*(a+m+l+2*x)*a^2*m*l/Den;
33
34 %'C' corresponds to p_3|i
35 C=(a+m+l+2*x)*a^2*l^2/Den;
36
37 %'D' corresponds to p_4|i
38 D=2*(a+l+2*m+2*x)*a*m*x*l/Den;
39
40 %'E' corresponds to p_5|i
41 E=2*a*x*m^2*(l+2*a+2*m+2*x)/Den;
```

```
42
43 %'F' corresponds to p_6|i
44 F=m^2*x^2*(1+2*a+2*m+2*x)/Den;
45
46 %Stores the above probabilities into a vector
47 CondProb = [A B C D E F];
48
49 %Check to make sure the conditional probabilities sum to 1
50 Check=sum(CondProb)
51
52 %The 'birth' rate for the nonhomogeneous aggregated model
53 LambdaHat = x*qf*B + (l*qa+2*x*qf)*C + (l*qa+x*qf)*D + l*qa*F;
54 %The 'death' rate
55 ThetaHat = theta*(A + B + E);
56 %The level of the orbit is approximately Poisson distributed with this rate
57 Parameter = LambdaHat/ThetaHat;
58
59 %This loop creates an 'OrbitSize' X 6 matrix comprised of the approximate
60 %joint probabilities of the orbit size and status of the servers where the
61 %rows correspond to the orbit size and the columns consist of the
62 %probabilities that correspond to the status of the servers.
63 P=[];
64 for i=1:OrbitSize
65     for j=1:6
66        P(i,j) = CondProb(j)*poisspdf(i-1,Parameter);
67     end
68 end
69
70 P;
71
72 %A small section to verify a substantial portion of probability mass
73 CheckSum=[];
74 i=1;
75 for i=1:OrbitSize
76     CheckSum(i)=sum(P(i,:));
77 end
78 IsItOne=sum(CheckSum)
79
80 %Approximate Mean Orbit Length, E[R], calculation
81 prob=[];
82 index=[];
83 for i=1:OrbitSize
84     prob(i)=sum(P(i,:));
```

```
    index(i)=i-1;
end
R=index*prob'

%This loop approximates the probability that a server is free.  It is used
%in the approximation for total time spent in orbit
i=1;
ProbServerFree=[];
for i=1:OrbitSize
    ProbServerFree(i)=P(i,1)+P(i,2)+P(i,5);
end

%Approximate expected number at server
Ns=sum(P(:,2))+sum(P(:,4))+2*sum(P(:,3))

%Approximate long-run average number of customers in system
L=R + Ns

%Approximate long-run average time spent in system per customer
W=L/l

ProbServFree=sum(ProbServerFree)
%Approximate average time spent in orbit
Wr = (1/theta)*(1/ProbServFree)

%Probability a customer cannot gain access to the servers
BlockProb=1-ProbServFree
```

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704–0188

1. REPORT DATE *(DD–MM–YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From — To)*
23–03–2007	Master's Thesis	Mar 2006 — Mar 2007

4. TITLE AND SUBTITLE

Approximate Analysis of an Unreliable M/M/2 Retrial Queue

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Crawford, Brian P., 1 Lt, USAF

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way
WPAFB OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT/GOR/ENS/07-05

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

National Reconnaissance Office
205 West D. Avenue, Suite 348
Attn: William J. Comstock, Rm 43D19H
14675 Leed Road
Chantilly, VA 20151-1715
(703) 808-4436

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

Approval for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

This thesis considers the performance evaluation of an M/M/2 retrial queue for which both servers are subject to active and idle breakdowns. Customers may abandon service requests if they are blocked from service upon arrival, or if their service is interrupted by a server failure. Customers choosing to remain in the system enter a retrial orbit for a random amount of time before attempting to re-access an available server. We assume that each server has its own dedicated repair person, and repairs begin immediately following a failure. Interfailure times, repair times and times between retrials are exponentially distributed, and all processes are assumed to be mutually independent. Modeling the number of customers in the orbit and status of the servers as a continuous-time Markov chain, we employ a phase-merging algorithm to approximately analyze the limiting behavior. Subsequently, we derive approximate expressions for several congestion and delay measures. Using a benchmark simulation model, we assess the accuracy of the approximations and show that, when the algorithm assumptions are met, the approximation procedure yields favorable results. However, as the rate of abandonment for blocked arrivals decreases, the performance declines while the results are insensitive to the rate of abandonment of customers preempted by a server failure.

15. SUBJECT TERMS

Multi-server, retrial queue, unreliable server

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Jeffrey P. Kharoufeh, Phd, (ENS)
U	U	U	UU	84	19b. TELEPHONE NUMBER *(include area code)* (937) 785-3636 ext 4603; e-mail: Jeffrey.Kharoufeh@afit.edu