Queueing Models for Capacity Changes in Cellular Networks

A Thesis Submitted to the
College of Graduate Studies and Research
in Partial Fulfillment of the Requirements
for the degree of Doctor of Philosophy
in the Department of Mathematics and Statistics
University of Saskatchewan
Saskatoon

By

Qingxiang Yan

Permission to Use

In presenting this thesis in partial fulfilment of the requirements for a Postgraduate degree from the University of Saskatchewan, I agree that the Libraries of this University may make it freely available for inspection. I further agree that permission for copying of this thesis in any manner, in whole or in part, for scholarly purposes may be granted by the professor or professors who supervised my thesis work or, in their absence, by the Head of the Department or the Dean of the College in which my thesis work was done. It is understood that any copying or publication or use of this thesis or parts thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to the University of Saskatchewan in any scholarly use which may be made of any material in my thesis.

Requests for permission to copy or to make other use of material in this thesis in whole or part should be addressed to:

Head of the Department of Mathematics and Statistics
University of Saskatchewan
Saskatoon, Saskatchewan
Canada
S7N 5E6

Abstract

With the rapid development of cellular communication techniques, many recent studies have focused on improving the quality of service (QoS) in cellular networks. One characteristic of the systems in cellular networks, which can have direct impact on the system QoS, is the fluctuation of the system capacity. In this thesis, the QoS of systems with capacity fluctuations is studied from two perspectives: (1) priority queueing systems with preemption, and (2) the $M/M/{\sim}C/{\sim}C$ system.

In the first part, we propose two models with controlled preemption and analyze their performance in the context of a single reference cell that supports two kinds of traffic (new calls and handoff calls). The formulae for calculating the performance measures of interest (i.e., handoff call blocking probability, new call blocking and dropping probabilities) are developed, and the procedures for solving optimization problems for the optimal number of channels required for each proposed model are established. The proposed controlled preemption models are then compared to existing non-preemption and full preemption models from the following three perspectives: (1) channel utilization, (2) low priority call (i.e., new calls) performance, and (3) flexibility to meet various constraints. The results show that the proposed controlled preemption models are the best models overall.

In the second part, the loss system with stochastic capacity, denoted by $M/M/{\sim}C/{\sim}C$, is analyzed using the Markov regenerative process (MRGP) method. Three different distributions of capacity interchange times (exponential, gamma, and Pareto) and three different capacity variation patterns (skip-free, distance-based, and uniform-based) are considered. Analytic expressions are derived to calculate call blocking and dropping probabilities and are verified by call level simulations. Finally, numerical examples are provided to determine the impact of different distributions of capacity interchange times and different capacity variation patterns on system performance.

Acknowledgements

I am grateful to a number of people for their time and support in the completion of this thesis. Principal thanks and appreciation are due to my supervisor, Dr. Raj Srinivasan, for his constant encouragement and valuable guidance in every stage in writing this thesis. Without his professional insights and impressive kindness and patience, this thesis would not have reached its present form. I would also like to extend a sincere thank you to my committee members, Dr. Mik Bickis, Dr. Chris Soteros and Dr. Gordon A. Sparks, for their participation and for providing me with valuable suggestions throughout. I would also like to thank Dr. Myron Hlynka for being my external examiner. His insightful comments were much appreciated. I would further like to express my gratitude to Guichang Zhang for his expertise and generous support, and as well as Zhengrong Li for his encouragement, which is necessary to complete this endeavor.

Finally, I am eternally grateful to my beloved parents, Junde Yan and Sumin He, for their loving considerations and great confidence in me all through the years. They have always helped me out of difficulties and supported me whenever I need. I would especially like to thank and dedicate this thesis to my wife, Jia Yu for her infinite patience, understanding, support and encouragement throughout the process. Her attitude of hard working has inspired me all along. I could never have made it this far without my family's perpetual love.

This thesis is dedicated to

my parents Junde Yan and Sumin He, my wife Jia Yu, and my son Caden Yan.

Contents

List of Tables

List of Figures

List of Symbols

$\boldsymbol{G}$	The infinitesimal generator of a given Markov chain	26
λ_c	The average capacity-change rate for the $M/M/{\sim}C/{\sim}C$ queueing system	141
λ_1	Arrival rate for low priority traffic, e.g., new calls	25
λ_2	Arrival rate for high priority traffic, e.g., handoff calls	25
μ_1	Call departure rate for low priority traffic, e.g., new calls	25
μ_2	Call departure rate for high priority traffic, e.g., handoff calls	25
μ_c	The mean capacity interchange time for the $M/M/{\sim}C/{\sim}C$ queueing system	141
Ω_b^N	The set of blocking states for new calls	30
Ω_d^N	A set containing all the states that can initiate call dropping transitions	30
$\Omega_{g=1}$	The state space of the first guard channel model when $g = 1$	35
Ω	State space of a giving queueing system	26
$\overline{Q}$	Mean queue length of the $M/M/c/c$ queue at equilibrium	34
ρ_k	The offered load for the new call ($k = 1$) or for the handoff call ($k = 2$)	25
$C_{(i,j)}^r$	Coefficient in front of the r^{th} boundary point for state (i, j)	37
$EB(\rho, k)$	The Erlang B formula with offered load ρ and number of channels k	35
g	Number of guard channels in the system	25
g^*	The optimal number of guard channels	64
n	Total number of channels in the system	25
n^*	The optimal number of channels	65

n_c^*	When g is fixed, the number of total channels where the local maxima of P_d^N occurs	81
P_0^{hb}	The constraint on handoff call blocking probability	63
P_0^{Nb}	The constraint on new call blocking probability	68
P_0^{Nd}	The constraint on new call dropping probability	68
P_0^{NL}	The constraint on new call loss (blocking and dropping combined) probability	65
P_b^h	Handoff call blocking probability	29
P_b^N	New call blocking probability	29
P_d^N	New call dropping probability	29
P_L^N	New call loss (blocking and dropping combined) probability.	31
R	The ratio of handoff blocking constraint (P_0^{hb}) to new call loss constraint(P_0^{NL})	102
$(n^{*\prime}, g^{*\prime})$	The solution to optimization problem $\mathbf{O_2'}$	85
$\mathbf{O_1}$	The optimization problem to find the optimal number of guard channels when the total number of channels is fixed	63
$\mathbf{O_2}$	The optimization problem to find both the optimal total number of channels and the optimal number of guard channels	65
AMPS	Advanced mobile phone system	5
AUTOVON	The automatic voice network	20
BDP	Birth and death process	13
BS	Base station	2
CAC	Call admission control	21

CDMA	Code division multiple access	2
CDPD	Cellular digital packet data	14
DCRS	Dynamic channel reservation scheme	10
DTMC	Discrete time Markov chain	125
EC	Effective capacity	34
FCFS	First-come first-served	8
FDMA	Frequency division multiple access	5
FIFO	First-in first-out	9
FSS	Fully shared scheme	8
GCS	Guard channel scheme	9
GoS	Grade of service	109
GPRS	General packet radio service	7
HQS	Handoff queueing scheme	9
HT	The widely used fixed guard channel model, which is a non-preemption model	69
LTE	Long term evolution	2
M1	The first guard channel model	25
M2	The second guard channel model	69
MBPS	Measurement based prioritization scheme	9
MGCS	Multi-guard channel scheme	11
MRGP	Markov regenerative process	19

MS	Mobile station	2
MTSO	Mobile telephone switching office	2
MWCP	Model(s) with controlled preemotion	100
MWFP	Model(s) with full preemption	100
MWNP	Model(s) with no preemption	100
MWP	Model(s) with preemption; this term includes both controlled preemption models and fully preemption models	100
OM	The original model, also known as the fully preemptive model	24
PASTA	Poisson arrivals see time averages.	72
PHZ	Pre handover zone	10
PSTN	Public switched telephone network	2
QoS	Quality of service	1
RSS	Received signal strength	6
SMP	Semi-Markov process	125
TDMA	Time division multiple access	2

Chapter 1

Introduction

Cellular communication has experienced an explosive growth in the past two decades. Today, millions of people around the world are using cellular phones as their major communication tools. Such rapid development in cellular communication has stimulated interest in studying and improving the quality of service (QoS) in cellular communication networks. One limitation of cellular networks is the unpredictability of available network capacity due to channel breakdown, channel reservation, and channel preemption. Therefore, the study of QoS of systems with fluctuating capacities becomes necessary and meaningful. The study in this thesis investigates this topic from two perspectives: (1) priority queueing systems with preemption, and (2) the $M/M/{\sim}C/{\sim}C$ system, which is a variant of the traditional $M/M/C/C$ system with fluctuating capacity.

A priority queueing system is a queueing system that serves customers of different priority levels. Most often, the services received by high-priority customers are guaranteed by allowing high-priority customers to preempt low-priority customers when the system is congested. As a result, the amount of system resources (servers) available to low-priority customers is greatly affected by the demand from high-priority costumers. In the first part of the thesis, the study focuses on priority queueing systems in the context of cellular communications, where two kinds of traffic are considered: handoff traffic (high-priority) and new traffic (low priority). In the second part of the thesis, the $M/M/{\sim}C/{\sim}C$ system, first introduced by Luo and Williamson [32], is used to directly model systems whose capacity can vary stochastically over time.

In this chapter, an introduction to cellular networks, and an overview of handoff techniques in cellular networks from the aspects of handoff initiation, handoff types, handoff decision and prioritization schemes, are provided. The traditional $M/M/C/C$ loss system and the Erlang B formula are introduced. Then recent studies on systems with fluctuating capacity are reviewed. The chapter concludes with a description of the scope of the thesis.

1.1 Introduction to cellular networks

The cellular network is currently in its fourth generation. The first generation used analogue communications. To accommodate more cellular phone subscribers and increase the network capacity, digital TDMA (time division multiple access) and CDMA (code division multiple access) technologies were developed in the second generation. The third generation provided users with high-speed packet-switching data transmission in addition to circuit-switching data transmission. The fourth and current generation provides mobile ultra broadband Internet access. Two 4G candidate systems are commercially deployed: the Mobile WiMax standard (first in South Korea in 2006) and the first-release Long Term Evolution (LTE) standard (in Oslo, Norway and Stockholm, Sweden since 2009).

What exactly is a cellular network? Zhang and Stojmenovic [64] provided a detailed introduction to cellular networks. A cellular network provides cell phones or mobile stations (MSs) with wireless access to the public switched telephone network (PSTN). In modern wireless communications, the service coverage area of a cellular network is divided into many small areas, or cells, each of which is served by a base station (BS). The BS is connected to the mobile telephone switching office (MTSO), which is also known as the mobile switching center. The MTSO is in charge of a cluster of BSs and is connected to the PSTN. The wireless connection between base and mobile stations allows mobile devices such as cellphones to communicate with wire-line phones in the PSTN (Figure 1.1).

One critical problem in cellular communication is the limited amount of frequency spectrum that can be allocated for cellular communication. The solution to this problem is the frequency reuse concept. As the coverage area is divided into cells, each cell is assigned a group of frequency bands or channels. To avoid radio cochannel interference, the group of channels assigned to one cell must be different from those assigned to its neighbouring cells. However, the same group of channels can be assigned to two cells if the cells are far enough from each other that the radio cochannel interference between them is limited to a tolerable

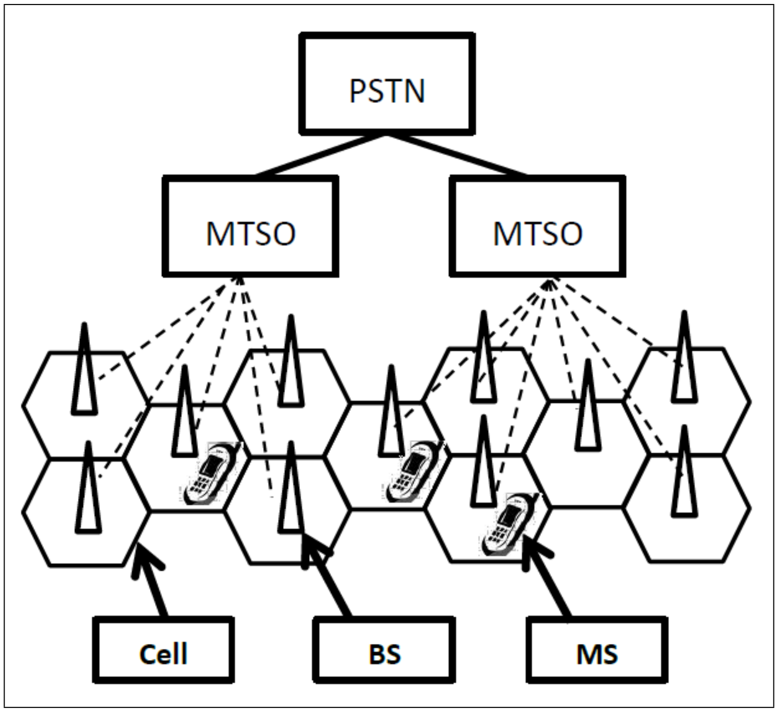


Figure 1.1: Typical structure of a cellular network

level. Typically, a reuse factor of seven is adopted meaning that seven neighbouring cells are grouped together to form a cluster. The total available channels are divided into seven groups, each of which is assigned to a cell within the cluster. The groups of channels can then be reused in other clusters of cells (Figure 1.2). Assuming there are N channels allocated to a cellular network that consists of C cells, $CN/7$ channels are available in the cellular network for concurrent use when the reuse factor is seven. However, because of the explosive growth of mobile phone subscribers, the current network capacity might not be enough, even with frequency reuse. Black [5] and Rappaport [43] proposed a cell splitting technique to increase the network capacity without new frequency spectrum allocation. The idea was to use several low power transmitters instead of one powerful transmitter and split an original cell into several (typically four) smaller cells. After cell-splitting, the cellular network that was originally covered by C cells is now covered by $4C$ smaller cells and, has the new capacity of $4CN/7$. In practice, not all cells are split into smaller cells and cells of different sizes (e.g., pico, micro, and macro cells) can coexist in a single cellular network. Another technique to increase the network capacity is sectoring [5, 43]. In sectoring, the cell size remains the same, but a cell is divided into several sectors by using directional antennas at the BS instead of a

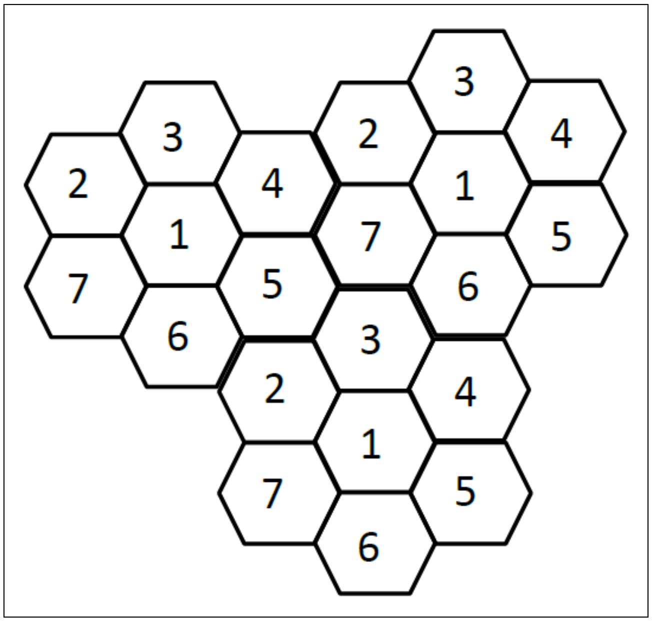


Figure 1.2: Channel reuse: the total available channels are divided into seven groups, each of which is assigned to a cell. The cells marked with the same number have the same group of channels assigned to them and the cells marked with different numbers have different groups of channels assigned to them.

single omnidirectional antenna. Typically a cell is divided into six 60° sectors. By dividing a cell into smaller sectors and applying the frequency reuse technique on these sectors, the frequency reuse factor is reduced and the total network capacity is increased.

The channels assigned to a cell can be divided into voice channels and control channels. A voice channel is used for an actual conversation and a control channel is used to set up the conversation. Both voice and control channels are further divided into forward (downlink) and reverse (uplink). A forward channel carries traffic from the BS to the MS and a reverse channel carries traffic from the MS to the BS. Multiple access methods are used to help MSs located in the cell to share the available channels.

To make a call from an MS, a request must be sent to the MTSO via a reverse control channel in its current cell. Once the request is granted by the MTSO, two voice channels (one for sending voice and the other one for receiving voice) will be assigned to the MS for making the call. Making a call to an MS is more complicated than making a call from an MS. To make a call to an MS, the call must be first routed to the MTSO in charge. Then the MTSO in charge needs to locate the cell of the target MS through location management.

Once the MTSO knows which cell the MS is in, two voice channels from that cell are then assigned to the MS to complete the call.

If an MS moves out of the MTSO where the MS is originally subscribed for wireless services (also known as the home MTSO), it is roaming. A roaming MS can receive services (such as making calls, receiving calls, connecting to the internet) only after it has been registered in the visited MTSO with information authenticated against the information kept in the home MTSO.

Within a given cell covered by a BS, there are multiple MSs that need to communicate with the BS simultaneously. Multiple MSs share the air interface in an orderly manner through multiple access methods. Three popular multiple access methods are frequency division multiple access (FDMA), time division multiple access (TDMA), and code division multiple access (CDMA). FDMA divides the frequency spectrum assigned to the BS into several frequency bands, or channels, that are well separated and do not interfere with each other. This method of FDMA is used in the Advanced Mobile Phone System (AMPS) [5, 6]. In an FDMA cellular network, typically about 45 MSs within a cell can communicate with the BS simultaneously. TDMA is usually built alongside FDMA and allows multiple MSs to share the same channel by chopping time into time slots of equal length. MSs take their turns using the shared channel with only one MS being allowed to use the shared channel in each time slot. Therefore, although the channel is shared, no interference can arise among the sharing MSs because only one MS can use the channel at a given time. Because MSs using TDMA cannot use a channel continuously, transmitting voice is a potential challenge. Fortunately, an ordinary human being can stand a delay of 20 milliseconds (ms). A more advanced way to implement TDMA is through dynamic TDMA which uses a scheduling algorithm to dynamically reserve a variable number of time slots to accommodate variable bit-rate data streams based on the traffic demand of each data stream. In the CDMA approach, each MS is assigned a unique sequence code to modulate its signal. CDMA is a spread spectrum multiple access technique, as each MS's signal is spread over the entire bandwidth by the unique sequence code assigned to it. At the receiver, that same unique code is used to recover the signal. Although the radio channel is shared, no interference can arise because the sequence codes used by the sharing MSs are orthogonal. The signal received by the BS from each MS must be at the same transmitted power; to achieve this, a few bits in the forward control channel are reserved for power control. The BS uses these bits to instruct each MS to adjust its output power level to guarantee that all signals received by the BS have the same strength. For more details regarding how CDMA encodes and decodes refer to Stallings [48].

1.2 Introduction to the handoff phenomenon and guard channel schemes

Handoff is a new phenomenon which arises with the development of wireless communications. Cellular systems divide a geographic area into small cells such that the same radio frequency can be reused in cells that are certain distance away. Smaller cells can help the system achieve higher system capacity but also increases the possibility that an active MS might move from cell to cell during an ongoing call. When an MS is engaged in a call, it is using two channels in its current cell. When the MS moves out of the boundary of the current cell and enters a neighbouring cell, it needs to acquire two channels from the neighbouring cell to keep the ongoing call alive. The process of transferring a call from one cell to a neighbouring cell is called a handoff.

1.2.1 Handoff Initiation

Handoff initiation is the process of requesting a handoff. Four main handoff initiation techniques mentioned in Ekiz et al. [13], Marichamy et al. [33], Pollini [39] will be examined. All techniques are based on the received signal strength (RSS) from the current cell (RSS_c in Figure 1.3) and from a neighbouring cell (RSS_n in Figure 1.3). As a result of signal propagation, the RSS becomes weaker as the MS moves towards the boundary of its current cell and becomes stronger as it crosses the boundary and enters a neighbouring cell. The received signal is averaged over time using an averaging window to remove momentary fading due to geographical and environmental factors [39, 55]. The signal strength threshold S_{min} in Figure 1.3 is called the "*receiver threshold*". The receiver threshold is the minimum acceptable RSS for call continuation [55]. If a moving MS fails to acquire channels from the neighbouring cell and RSS_c drops below the receive threshold, the ongoing call is dropped.

The first handoff initiation technique is purely based on the RSSs of an MS. The RSSs are measured over time and the MS will be transferred to the BS with the strongest signal. In Figure 1.3 at time T_1, the RSS from the neighbouring cell starts to exceed the RSS from the current cell and a handoff is initiated. This technique is simple to implement but its downside is obvious: due to signal fluctuations, several unnecessary handoffs can occur while the RSS from its current cell is still strong enough to serve the call (i.e., stronger than S_{min}). These unnecessary handoffs are known as *ping-pong* effects and will cause an increase in forced termination probability. A good handoff technique should minimize such effect.

The second handoff initiation technique is called *relative signal strength with threshold*. It is similar to the first handoff initiation technique but a threshold (S_1 in Figure 1.3) is implemented to reduce the ping-pong effect. A handoff is initiated only if RSS_c (the current cell's RSS) is lower than the threshold and RSS_n (the neighbouring cell's RSS) is stronger than RSS_c. The handoff is initiated at time T_2.

The third handoff initiation technique is *relative signal strength with hysteresis*. This technique uses a predetermined hysteresis value (h in Figure 1.3). A handoff is initiated (at time T_3) when RSS_n exceeds RSS_c by the hysteresis value h.

Relative signal strength with hysteresis threshold combines both the threshold technique and the hysteresis technique. A handoff is initiated when RSS_c is below a threshold (which could be chosen between S_1 and S_{min}) and RSS_n is stronger than RSS_c by the hysteresis value h. The handoff initiation can occur between T_3 and T_4.

1.2.2 Hard handoff and soft handoff

A handoff can be *hard* or *soft*. The hard handoff occurs when the radio frequency channel in use from the current channel is released first and a new channel from the neighbouring cell is acquired later. Because of the time gap between channel release and channel acquisition there is a service interruption when this type of handoff occurs. Hard handoffs are common to systems using TDMA and FDMA such as General Packet Radio Service (GPRS) [28].

The soft handoff is a feature of systems that use CDMA standards, where an MS can

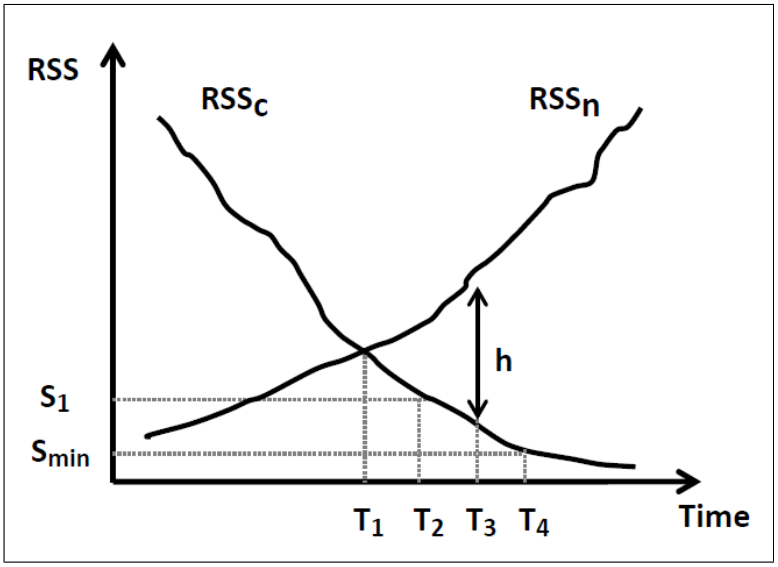


Figure 1.3: An illustration of handoff initiation techniques

simultaneously be connected to two or more BSs during a call. When an MS is engaged in a call a BS is added in when the RSS from this BS exceeds a given threshold and removed when RSS drops below certain threshold for a given amount of time [13]. The addition or removal of a BS during an active call causes soft handoff. There is no service interruption during a soft handoff.

1.2.3 Handoff channel-assignment schemes

If we consider a reference cell and a neighbouring cell, two types of calls can be distinguished. A *handoff call* is defined as a call that is in progress in a neighbouring cell needs to be transferred and continued in the reference cell because of the movement of an MS. In contrast, a *new call* is a call that originates in the reference cell. In this section different handoff channel-assignment schemes will be reviewed.

The simplest channel assignment scheme is the fully shared scheme (FSS) in which all available channels are fully shared by both handoff calls and new calls. Handoff calls and new calls are treated equally and are served on a first-come first-served (FCFS) basis. If all channels are busy upon the arrival of an incoming call, the incoming call will be blocked. The FSS is widely used in current cellular networks because of its simplicity [64]. In addition, the FSS has the advantage of maximizing the utilization of wireless channels as opposed to the guard-channels schemes (which will be introduced later). The disadvantage of the FSS is the potentially high blocking rate of handoff calls.

Since it is generally less desirable to terminate an ongoing handoff call than to block a new call, recent research on channel-assignment schemes has focused on reducing the loss probability of handoff calls. Many prioritization schemes have been proposed [33, 55, 56, 57]. One such scheme is the handoff queueing scheme (HQS, [55]). As discussed earlier, when a call moves into the reference cell from a neighbouring cell, it will be terminated if it fails to acquire a new channel from the reference cell and the RSS from its original cell (the neighbouring cell) drops below the receiver threshold. The HQS is feasible because there is a time difference between the time of handoff initiation and the time when the RSS reaches the receiver threshold. When a handoff call has requested to be transferred into the reference cell but all channels in the reference cell are occupied, instead of terminating it immediately, this handoff call is placed in the line of calls that are waiting for channel release in the reference cell. When the RSS drops below the receiver threshold the call will be lost. A new call can be admitted into the cell only if there is no handoff call waiting in the queue and there is at least one free channel in the BS. The HQS reduces the loss probability of handoff-calls while increasing the blocking probability of new calls. A timer based handoff priority scheme is proposed in Marichamy et al. [33] in which, when a channel is released, a timer starts and this channel will be reserved for handoff use for a certain amount of time. If no handoff call arrives during that period of time and the timer expires, the channel can be assigned to new or handoff calls on a FCFS basis. In Tekinay and Jabbari [56] Measurement Based Prioritization Scheme (MBPS) was introduced. The priority of a handoff call waiting in the queue changes dynamically based on the RSS from its cell. The calls with RSS close to the receiver threshold have higher priority than calls with higher RSSs. This scheme produces better results than the first-in first-out (FIFO) queueing scheme.

Another widely adopted type of channel assignment scheme that prioritizes the handoff call is the guard channel scheme (GCS). In Harine et al. [17] a basic GCS is introduced in which, a predetermined number of channels in the reference cell are reserved exclusively for handoff calls. The remaining channels, called the normal channels, are shared by handoff calls and new calls. Both handoff calls and new calls use the normal channels first. When all the normal channels are occupied, incoming new calls will be blocked but incoming handoff calls can still be admitted into the cell if there is at least one idle guard channel. The loss probability of handoff calls improves with an increase in the number of guard channels. However, the new call blocking probability increases and the total utilization of channels decreases, as idle guard channels can not be used by new calls. In Kim et al. [24], a dynamic channel reservation scheme (DCRS) based on mobility is proposed to increase the total channel utilization without increasing the loss probability of handoff calls. In the DCRS, normal channels are still shared by new calls and handoff calls. However, the guard channels, although reserved for handoff calls, can also be used by new calls whose request probability depends on the mobility of calls. The mobility of calls in the reference cell is defined as the ratio of the handoff arrival rate to the new call arrival rate. If there are no arrivals of handoff calls, the request probability is one, and the guard channels will be used by new calls. If there are no arrival of new calls, the request probability is zero and the guard channels will be used by handoff calls. If the mobility is greater than one, i.e., the arrival rate of handoff calls is larger than that of new calls, the request probability is decreased quickly so the handoff calls can use the guard channels. If the mobility is less than one, i.e., the arrival rate of handoff calls is less than that of new calls, the request probability is decreased slowly so new calls have the opportunity to use idle guard channels. In this way, handoff call performance is guaranteed and the blocking probability of new calls is reduced. There are other methods to determine the number of guard channels dynamically. In Agrawal et al. [1] the number of guard channels is determined dynamically by the use of neighbouring BSs. Each BS periodically determines the number of MSs in a prehandover zone (PHZ)—a small area next to the handoff zone that contains users who will possibly request handoff soon—and reports that number to an adjacent BS. The adjacent BS then reserves that number of channels as guard channels in its own PHZ. In Zhang and Liu [65] an adaptive algorithm to assign the number of guard channels is proposed. When the dropping probability of handoff calls exceeds a predetermined threshold the number of guard channels is increased to reduce the likelihood of a handoff call being lost.

As the demand for mobile multimedia services (such as voice, data, and video) has increased since the third-generation of cellular networks, multimedia based guard channel schemes are necessary. In Wang et al. [59], real-time and nonreal-time traffic are considered. Traffic calls are categorized into four different types: real-time and nonreal-time new calls, and real-time and nonreal-time handoff calls. Accordingly, the channels in each cell are divided into three parts: one for real-time calls, one for nonreal-time calls only, and one for handoff calls that cannot be serviced in the first two parts. In the third group, several channels are reserved exclusively for real-time handoff calls. In addition, a real-time handoff call has the right to preempt nonreal-time calls if no channels are available; the interrupted nonreal-time is redirected to a queue. Hwang et al. [21] has proposed a multiguard channel scheme (MGCS) that can be used in cellular networks with multiclass traffic. In this model, different channel thresholds are set for different types of calls. A certain type of traffic can be admitted to the cell only if the number of busy channels is less than the channel threshold set for its type. This model extends the GCS for single class traffic to multi-class traffic. In Somagari and Pati [47] an adaptive MGCS for multi-class traffic is proposed to ensure the QoS for multimedia wireless cellular networks and to minimize the dropping of handoff calls. Although a different number of guard channels are reserved for the handoff calls of different traffic classes, handoff calls in a class with low priority can access the guard channels of of the handoff calls in the next higher class with a certain probability determined by the mobility of calls and channel occupancy.

1.3 Queueing models for single cell

Server capacity is the primary determinant of system performance. In conventional queueing system environments, such as call centers, the physical server capacity is usually a fixed quantity. These systems have been well-studied for many years. In other queueing systems, the available system capacity can vary unpredictably over time. Many examples of stochastic capacity systems appear in the context of wireless transmission. For example, in a reservation-based system with multiple priority levels, high priority traffic such as voice may take precedence over data traffic. As a result, the system capacity available for low priority traffic changes over time based on the demand of high priority traffic [50]. Another simple example would be high-performance computing centers, where the failure or the removal of computing nodes from the system can result in the loss of jobs from the system, and having an impact on the blocking rate or queueing delay seen by other jobs [51]. The rapid development of computer networks and mobile technologies has increased the interests in systems in which the server capacity changes over time. In such systems call losses can be due to:

- Call blocking: This refers to the scenarios when an *incoming* call fails to acquire a channel and is rejected from the cell.
- Call dropping: This refers to the scenarios when an *ongoing* call is terminated prematurely and forced to leave the cell and never returns.

In the following section, the standard $M/M/C/C$ model and the well known Erlang B formula are reviewed. Then a variant of the $M/M/C/C$ model with stochastic capacity, denoted by $M/M/\sim C/\sim C$ ([32]) is introduced.

1.3.1 The $M/M/C/C$ model and the Erlang B formula

The $M/M/C/C$ model, also known as the Erlang loss model was first used to model call centers at the beginning of the 20th century and it can also be used to model a single-cell (the reference cell) in a cellular network. Assume that there are C channels available to serve calls made from wireless subscribers. All calls are homogeneous (in the sense that each of them can be served by any one of the channels) and arrive at the reference cell according to a homogeneous Poisson process with rate λ. The channels are also assumed to be homogeneous and the service time for each call follows i.i.d. exponential distribution with rate μ (which is also known as the departure rate for each call). It is further assumed that the traffic arrival process is independent of the traffic departure process. Arriving calls are served according to FCFS discipline and since there is no waiting room, when all channels are busy new calls will be blocked. Such a Markovian queueing model has been well studied in literature Kleinrock [25, 26].

The queue length process $\{Q(t), t \geq 0\}$ of this system is a finite birth and death process

(BDP) with state space $\{i | i = 0, 1, 2, ..., C\}$. The birth rate of state i is given by

$$\lambda_i = \begin{cases} \lambda & \text{if } 0 \leq i < C \\ 0 & \text{otherwise} \end{cases} \tag{1.1}$$

and the death rate of state i is $\mu_i = i\mu$, $i = 0, 1, ...C$. Its infinitesimal generator $\mathbf{G}$ can be written as:

$$\mathbf{G} = \begin{bmatrix} -\lambda & \lambda & & & & & & \\ \mu & -\lambda-\mu & \lambda & & & & & \\ & 2\mu & -\lambda-2\mu & \lambda & \cdots & & & \\ & & & & \cdots & & & \\ & & & & \cdots & (k-1)\mu & -\lambda-(k-1)\mu & \lambda \\ & & & & & & k\mu & -k\mu \end{bmatrix}.$$

The structure of the matrix G shows that the queue length process $\{Q(t), t \geq 0\}$ is irreducible and hence the stationary distribution π exists and is unique. Since the stationary distribution must satisfy $\boldsymbol{\pi G}$=$\mathbf{0}$ and $\sum_{i=0}^{C} \pi_i = 1$, by solving a system of equations we have:

$$\begin{aligned} \pi_0 &= \left(\sum_{n=0}^{C} \frac{\rho^n}{n!} \right)^{-1}, \\ \pi_i &= \pi_0 \cdot \frac{\rho^i}{i!} \text{ for all } i = 1, ..., C, \end{aligned}$$

where $\rho = \lambda/\mu$ is the total offered load, a measure of demand made on the system, which is dimensionless but given a unit called erlangs. The main performance measure for this model is the probability that all channels are busy and the cell is unable to accept new call requests, that is, the call blocking probability which is given by

$$\pi_C = \left(\sum_{n=0}^{C} \frac{\rho^n}{n!} \right)^{-1} \cdot \frac{\rho^C}{C!} = EB(\rho, C). \tag{1.2}$$

This formula is known as Erlang's loss formula or the Erlang B formula and is widely used in systems where the server capacity is a constant over time.

1.3.2 Stochastic capacity and the $M/M/\tilde{C}/\tilde{C}$ model

Background

The variation in capacity with time, known as the "stochastic capacity", arises frequently in the context of wireless networks. The main reasons that lead to stochastic capacity are:

- *Server failure and repair activities:* In wireless communications, channels that carry voice or data traffic can fail. Failed channels will be repaired after some time. The system capacity decreases when channel failure occurs and increases when failed channels are repaired. Since the failure activities usually occur unpredictably and the repair times are random variables, system capacity is changing stochastically [58].
- *Different priority levels for different traffic:* In wireless networks, voice traffic usually takes precedence over data traffic; an example is the cellular digital packet data (CDPD) system analyzed by Massey and Srinivasan [34]. As a result, the system capacity for low priority traffic (i.e., the number of channels that could be used to transmit data traffic) varies with time based on high priority traffic (voice traffic) demands.
- *The time-varying characteristics of the wireless propagation environment:* This phenomenon applies to wireless LANs and CDMA systems. The system capacity of CDMA systems has a complex nature. Gilhousen et al. [16] predicted that properly augmented and power-controlled multiple-cell CDMA promises a significant increase in current cellular capacity. Shen and Ji [46] showed that user bandwidth demand, transmission capability and outage requirement have significant impact on CDMA network capacity. In Wu and Williamson [61], Wu Y. and Williamson C. found that increased variability in data call arrival decreases the system capacity, whereas increased variability in data

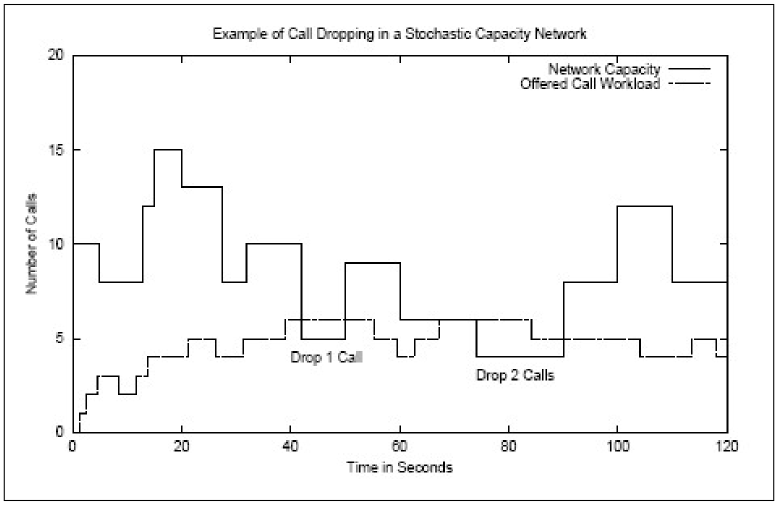


Figure 1.4: A sample path of stochastic capacity process

call holding times increases the system capacity.

Mechanism and impacts of stochastic capacity

In 2005, Sun and Williamson [50] performed a series of call-level simulations to study the performance of different call dropping policies in stochastic capacity network. The simulations assumes that the system has an overall average capacity for carrying n simultaneously ongoing calls, but the capacity varies randomly with time. An example of their stochastic network capacity model is shown in Figure 1.4. The horizontal axis represents time and the solid line portrays the available system capacity at each instant in time. The capacity changes are modeled as events that occur at specified points in time. The system capacity always has a non-negative integer value, but capacity changes can occur at arbitrary points in continuous time.

Four characteristics of the stochastic capacity process are:

- *Frequency of capacity changes:* If the frequency of capacity changes, f, is specified, then the capacity changes exactly every $1/f$ seconds. We say that capacity varies deterministically with time.
- *Distributions of interchange times:* The distribution used for the elapsed time between network capacity changes. Deterministic, exponential, and self-Similar models are used in their simulations. The deterministic model has a capacity-change event every T

seconds. The exponential model has capacity change events at random times following a Poisson process. The time between capacity-change events is exponentially distributed with a mean of T seconds. The self-similar model assumes that capacity-change events occur in bursts, similar to a self-similar (fractal) process. The mean time between capacity-change events is T seconds.

- *Distribution of the capacity itself:* This is used to generate the exact network capacity at each capacity-changing instant. The mean of this distribution should match the long-term average of n calls, while the variance affects the magnitude of capacity fluctuations that can occur. A normal distribution is used to facilitate control of both the mean and the variance of the system capacity.
- *Correlation structure in the capacity time series process:* Independent and identically distributed samples as well as self-similar processes are considered. In the self-similar model, the capacity values constitute a self-similar process, with short-range and long range correlations. In the random model, the same capacity trace is shuffled into a random order to remove short-range and long-range correlations.

Other important model specifications are:

- *Call workload:* New calls arrive according to a specified arrival process: Poisson or self-similar process. Each call has a specified holding time, drawn from a specified distribution (exponential or Pareto).
- *Call dropping Policies:* 9 dropping polices in five categories are considered. They are randomized (*Random*), arrival-based (*Last-In-First-Out*, *First-In-First-Out*), departure-based (*EarliestDeparture*, *LatestDeparture*), duration-based (*ShortestDuration policy*, *LongestDuration*) and completion-based (*LeastCompleted*, *MostCompleted*) policies.

The two inputs provided to the simulation are a call workload file and a network capacity file. The call workload file is a time-ordered sequence of call arrival events. Each call specifies its source node, destination node, arrival time, and duration. Each call requires one unit of network capacity. Workload files are generated using the call workload models indicated in Table 1.1. Each workload file contains 100,000 calls. The network capacity file is a time-ordered sequence of capacity-change events. Capacity files are generated using the models and parameters indicated in Table 1.2. Each capacity file contains 10,000 capacity-change

Table 1.1: Table 1. Call level workload parameters

Parameter		Level
Stochastic Traffic	Arrival Process	Poisson, Self-similar
	Holding Time	Exponential, Pareto
Call Arrival Rate (calls/sec)		0.1. . . 1.0. . . 6.0
Mean Call Holding Time (sec)		30

Table 1.2: Table 2. Network capacity parameter settings in call-level simulations

Parameter		Levels
Mean Time between Capacity Changes (sec)		10, 15, 30, 60, 120
Stochastic Capacity	Capacity Change Time	Deterministic, Exponential, Self-Similar
	Capacity Change Value	Normal
Capacity Values (calls)	Mean	40
	Standard Deviation	2, 5

events. In some simulations, only the initial portion of the capacity file is needed, depending on the frequency of capacity changes.

Each call dropping policy is provided with the same workload and capacity files, so that they each handle the same traffic demands under the same network conditions. Differences observed in the call level performance reflect differences in call dropping policies used.

The primary performance measures are call blocking probability and call dropping probability. The results (described below) of the simulations shed light on the impact of stochastic capacity:

- *Frequency of capacity changes.* As the time between capacity changes increases, the call blocking rates for all policies asymptotically converge toward the same value, and the call dropping rate asymptotically approaches 0. This result is expected, as low-frequency changes approximate a static network, for which the Erlang B blocking formula can be directly applied. If capacity changes are infrequent, few calls need to be dropped. The performance differences between dropping policies are more pronounced when there is a high frequency of capacity changes in the network. This result makes sense since high-frequency changes imply more call dropping episodes, and thus greater opportunity for distinctions among policies. The differences among policies manifest themselves more clearly in the call blocking performance than in the call dropping performance. Because all policies dropped about the same number of calls, carefully choosing which calls are dropped can significantly benefit the call blocking performance. The relationship between call blocking rate and frequency of capacity changes is not monotonic. For some policies, the call blocking rate decreases as capacity changes become less frequent, whereas for other policies, the behavior is nonmonotonic.

- *Variability of capacity changes.* The capacity values are drawn from a normal distribution with a mean of 40 calls and two different standard deviations 2 and 5. The higher-variability capacity model has a higher call blocking rate and a higher call dropping rate. The separation between dropping policies is more pronounced with higher capacity variability. These results show that for networks with high-frequency or high-variability capacity changes, the call dropping policy can have a large impact on call blocking performance.
- *Time of capacity changes.* The distribution of interchange time has a small impact on the call blocking performance, but a larger impact on the call dropping performance.
- *Correlation of capacity changes.* Results showed that correlations in the capacity-change process are beneficial. Random (uncorrelated) models can have large fluctuations in network capacity at any time scale, whereas correlated models produce more gradual changes in capacity.

$M/M/\sim C/\sim C$ queueing system

In Luo and Williamson [32], a variant of the M/M/C/C loss system with fluctuating server capacity was introduced. The new system is denoted by $M/M/\sim C/\sim C$. Similar to the $M/M/C/C$ loss system, the call interarrival time and service time follow independent exponential distributions. However, the system capacity (i.e., the number of available channels) follows a stochastic process and can vary with time (as is indicated by the tilde in front of system capacity C). Therefore, C can be considered to be the maximum capacity of this system. If the capacity interchange times are i.i.d. and follow exponential distributions, a two-dimensional Markov chain can be used to model this system. If capacity interchange times are i.i.d. but follow a general distribution, a Markov regenerative process (MRGP) method is used in Luo and Williamson [32] to analyze this model.

1.4 Scope of the thesis

For simplicity, we consider a pure loss cellular network with homogeneous cells for which a specific number of channels is permanently assigned to each cell, and our attention is focused on a single cell (the reference cell). First, two guard channel schemes with partial/controlled preemption are proposed. The inspiration comes from the widely used fixed guard channel scheme [17] as well as literatures on loss systems with preemption [18, 44, 66]. Although the proposed schemes will be discussed in the context of a loss system comprising handoff calls and new calls, the schemes can be considered generally as partially preemptive schemes for priority queueing system as well. Then, MRGP method will be reviewed and used to analyze the $M/M/\sim C/\sim C$ system. The main performance measures are calculated and the impact of stochastic capacity on these performance measures are assessed through numerical examples.

Related work

Based on the prioritizing schemes used, priority queueing systems can be classified into non-preemptive and preemptive priority queueing systems. Most of the handoff guard channel schemes are non-preemptive systems [11, 17, 37, 42, 63]. In Li et al. [29], Wang et al. [59], guard channel schemes with preemption are proposed in which nonreal-time traffic can be interrupted by real-time traffic. The interrupted traffic is redirected to a queue to wait for free channels instead of being dropped. Therefore they are delay systems rather then loss systems.

Related work on priority queueing systems can be divided into two categories: preemption policies and performance analysis of preemptive queueing systems. Garay and Gopal [15] investigated problems that relate to making the best decision on which call to preempt and proposed heuristics for a centralized network framework which performed well relative to the optimal solution. Then Peyravian and Kshemkalyani [38] presented a simulation study of preemption in a general connection-oriented network setting and developed two optimal connection preemption selection algorithms that operate in a decentralized network that optimized the criteria of (i) the number of connections to be preempted, (ii) the bandwidth to be preempted, and (iii) the priority of connections to be preempted, in different orders. Sung et al. [53] proposed a centralized connection preemption algorithm that optimized the preemption criteria in an order different from Peyravian and Kshemkalyani [38]; Sung's algorithm minimized the number of preempted and rerouted sessions. Stanisic and Devetsikiotis [49] analyzed two simple and efficient preemption policies with random selection which dramatically sped up the process of selecting a set of connections to be preempted.

Preemptive queueing systems can be classified into two groups: preemption with delay and preemption with loss. Preemption with delay is usually modelled by an $M/G/C$ queue (a system with infinite queueing). White and Christie [60] was the first to define and studied preemptive priority in a single server system with Poisson arrivals. This group also studied the case in which the preemptive server was prone to breakdown. In Miller [35], a matrix-geometric method was used to derive the recursive computational formulas for the steady state distributions of $M/M/1$ priority queues with two classes of customers. Buzen and Bondi [7] studied the mean response time of each priority level in a multiserver $M/M/m$ preemptive-delay network with multiple priority classes. In Cho and Un [10], the authors proposed a combined preemptive/non-preemptive priority discipline using preemptive-resume and preemptive-repeat-identical policies. Recently, Lian and Zhao [30] studied a two-stage $M/G/1$ queue with discretionary priority. These analyses are of limited relevance to the investigation of this thesis, which is a study of preemption with loss, because the more prevalent use of preemption policies is to drop, rather than postpone the preempted call.

The earliest work on the performance analyses of a preemptive loss system dates back to 1962, when Helly [18] used the Erlang B formula on a single cell to present a preemption framework. In 1980, Calabrese et al. [8] studied the automatic voice network (AUTOVON) with two classes of traffic, wherein the class 1 traffic can preempt class 2 traffic when the network is fully occupied. Two preemption disciplines (ruthless and the friendly) were considered. Also in 1980, Fischer [14] considered an $M/M/s/s$ preemptive system that carried two classes of customers with unequal service times. Due to the difficulty in obtaining the closed form steady state equations, the author analysed three special cases instead: (i) $s = 1$, (ii) the ratio of class 2 mean holding time to class 1 mean holding time approaches 0, and (iii) the ratio of class 2 mean holding time to class 1 mean holding time approaches infinity. In Zhao et al. [66], a two parallel link (i.e., a primary link and a backup link) network supporting K call classes was considered, where a class k call can preempt if necessary and calls of classes $k + 1, ..., K$ can in turn be preempted by any call of class $1, ..., k - 1$. The preemption rates were obtained in the heavy traffic limit. All the studies mentioned above considered systems with full preemption in which lower priority calls could be preempted by higher priority calls when necessary. To my knowledge, partial (or controlled) preemption was first introduced by in Zhou and Beard [67] and then in Zhou and Beard [68]. In their model, high priority calls (i.e., emergency calls) can only preempt low priority calls (i.e., public calls) when the number of active high priority calls in the system is within a threshold. Their scheme was similar to the first guard channel scheme that is proposed in this thesis. However, their scheme is not exactly the same as our scheme, and they focused on comparing the channel occupancy of their scheme with other emergency call admission control (CAC) schemes whereas we concentrated on studying the call loss probabilities and comparing them with fully preemptive and non-preemptive schemes.

Literature on the application of MRGP in queueing systems and recent studies on the $M/M/\sim C/\sim C$ system were examined. The MRGP model has been shown to capture the behavior of real systems with both exponentially and non-exponentially distributed event times and has been used to study non-Markovian queueing systems for years. In 1995, Logothetis et al. [31] surveyed the MRGP literature and adopted different solution techniques in their transient analyses. Dharmaraja et al. [12], used an MRGP model to calculate numerically new call and handoff call blocking probabilities with general (nonexponential) interarrival time distributions. Wu and Williamson [61] investigated the capacity of multiservice CDMA networks supporting voice and non-Poisson data traffic based on an MRGP model and showed that the variability of the data call arrival process adversely affected the system capacity. In 2007, Sun and Williamson [51] carried out some preliminary studies on queueing system with stochastic capacity based on MRGP. In 2008, the notation for a loss system with stochastic capacity, $M/M/{\sim}C/{\sim}C$, was introduced by Luo and Williamson [32]. They used the MRGP method to analyze the performance of an $M/M/{\sim}C/{\sim}C$ system for which capacity interchange times followed nonexponential distributions. In their model, the capacity could change only one unit at a time so the capacity process was skip-free. In our study, three different types of capacity variation are considered.

Thesis outline and contributions

In Chapters 2 and 3, we propose two guard channel models with controlled preemption. We restrict our attention to a relatively simple scenario: a single reference cell is considered, wherein two types of traffic are supported, one with higher priority than the other. In this context, the high priority traffic is the handoff call and the low priority traffic is the new call. In the proposed models, low priority traffic can access guard-channels but can be preempted by high priority traffic when necessary. Preempted calls are dropped and removed from the system. Assume that each call occupies one channel. The arrival processes for low and high priority traffic are independent Poisson processes (with rate λ_1 and λ_2, respectively) and the service times for both traffic types follow independent exponential distributions (with rate μ_1 and μ_2, respectively). The system as a whole can be viewed as a controlled preemptive $M/M/C/C$ system serving two types of traffic.

In Chapter 2, our first guard channel model with controlled preemption is proposed. The model is based on the full preemptive scheme but sets a limit on the maximum number of ongoing high priority calls (handoff calls) allowed in the system. The goal is to protect low priority calls (new calls) while maintaining the performance of high priority calls at a satisfactory level. Three performance measures are of interest: low priority call blocking and dropping probabilities and the high priority call blocking probability. Two approximate methods and two analytic methods are discussed and their performances are compared. Four special cases are investigated as inspired by Fischer [14]. At the end, two optimization problems are solved for which, an optimal number of total channels and/or guard channels can be determined based on predetermined call performance thresholds.

In Chapter 3, our second guard channel model with controlled preemption is developed and analyzed. This model also utilizes controlled preemption and is based on the fixed guard channel model in Harine et al. [17]. Closed form formulae for three performance measures are derived for homogeneous service rates of both low and high priority calls. The closed form solution is verified using call level simulations. Algorithms for solving two optimization problems are introduced. Finally, the optimal number of channels required to meet certain performance constraints are compared between our model and the fixed guard channel model studied in Harine et al. [17]. The results show that the channel utilization of our model is superior to that of the fixed guard channel model.

In Chapter 4, we compare the performance of four models: (i) the fixed guard channel model from Harine et al. [17], a model without preemption, (ii) our first guard channel model, a model with controlled preemption, iii) our second guard channel model, also a model with controlled preemption and, (iv) the model with full preemption studied in [18, 44, 66]. The models are compared according to: (a) channel utilization, which is reflected by the minimum number of channels required to meet certain constraints on call loss, (b) low priority call (i.e., new call) performance when the constraint for the performance of high priority call is met, and (c) flexibility to meet various constraints. The results show that each model possesses a unique advantage that depends on the traffic parameters. However, models with controlled preemption (i.e., the two new models proposed in the thesis) manifested the best overall performance.

In Chapter 5, the theory of MRGP is reviewed and applied to model our first guard channel model. Then the loss system with stochastic capacity, i.e., the $M/M/{\sim}C/{\sim}C$ system is discussed and the MRGP method is used to solve this system. Three different distributions of capacity interchange times (exponential, gamma, and Pareto), and three different capacity variation patterns (skip-free, uniform-based, and distance-based variations) are considered when constructing the MRGP model. Analytic results are verified by simulations and numerical experiments are carried out to study the impact of the characteristics of capacity-change (the distribution of capacity interchange times and the capacity variation pattern) on call loss probabilities.

Chapter 2

First Guard Channel Model

2.1 Motivations and model description

2.1.1 Motivations

As Chapter 1 explains, most of the priority queueing systems use the full preemption scheme in which high-priority traffic have priority over low-priority traffic on **all** the channels in the system. High-priority traffic can access all the channels and can preempt low-priority traffic to accommodate itself whenever the system is full. A loss system with full preemption will be called the *original model* (which will be called the OM model hereafter) . In such an original model, the performance measure (i.e., the blocking probability) of high-priority traffic remains unaffected by the low-priority traffic. However, the low-priority traffic consequently suffers unnecessary losses. As a straightforward example, consider the performance measures for the following situation:

1) There are a total of 15 channels in the system.

2) The high-priority traffic and low-priority traffic arrives according to Poison processes with rate 5 and 2, respectively.

3) The call holding times for both types of traffic follow the same exponential distribution with rate 1.

If one employs the OM model, the blocking probability for high-priority traffic is 0.016%, and the loss probability (blocking probability and dropping probability combined) for low-priority traffic is 1.1%. Assume that the performance thresholds for high-priority traffic and low-priority traffic are 0.5% and 1%, respectively, then the performance measure of high-priority traffic (0.016%) is much lower than its threshold 0.5% while that of low-priority traffic

(1.1%) exceeds its threshold 1%. Therefore, a model that could offer an easy adjustment to balance performance measures of high-priority and low-priority traffic, is desirable.

2.1.2 Model description

The model to consider is a single cell with a limited number of channels n, wherein g of them are set up as guard channels. There are two kinds of traffic: low-priority traffic (i.e., *new calls*) and high-priority traffic (i.e., *handoff calls*). New calls can access all the n channels, and handoff calls can access only the g guard channels. The arrival processes for both new calls and handoff calls are assumed to be independent Poisson processes with rates λ_1 and λ_2, respectively. The service time for new calls and handoff calls are assumed to follow independent exponential distributions with rates μ_1 and μ_2, respectively. Let $\rho_k = \lambda_k/\mu_k$, $k = 1, 2$ be the offered load for the new call (when $k = 1$) or for the handoff call (when $k = 2$). The call admission procedure is as follows: when a new call arrives and at least one idle channel is in the cell, the new call will be accepted, and a channel will be assigned to it. If there are no idle channels available, the new call will get blocked. When a handoff call arrives, it is admitted provided that there are at most $g - 1$ ongoing handoff calls in the cell. If there are already g ongoing handoff calls in the cell, then this incoming handoff call will get blocked from this cell. When a handoff call is admitted although all channels are busy, the system will choose an ongoing new call (according to some call dropping policy, random dropping policy by default) to drop in order to free a channel to accommodate the admitted handoff call. Therefore, handoff calls can preempt new calls only when the number of ongoing handoff calls currently in the system is less than g. As we can see, the system capacity for new calls, denoted by i, is changing stochastically with time and is depending on the arrival and departure events of the handoff call. Based on this model description, it is not hard to see that there can be at most g ongoing handoff calls simultaneously in the system, and therefore i could take value in $c, c + 1, \cdots, n$, where $c = n - g$. This model is considered as the first guard channel model (or the M1 model, a term which will be used interchangeably). The major difference between the M1 model and the original model is that in the M1 model, any high-priority traffic can access only the g guard channels and will have priority over low-priority traffic on these g guard channels, instead of on all the n channels.

To be more specific, the original model is a special case of the M1 model when $g = n$. Next, we will construct a stochastic model to solve the M1 model.

2.2 The composite model method and performance metrics

2.2.1 The composite model method

The system states of the M1 model could be described by $\Omega = \{(i,j) | c \leq i \leq n, j \leq i\}$, where i denotes the system capacity for new calls, and j is the number of ongoing new calls currently in the system. The number of ongoing handoff calls currently within the system is given by $n - i$ (which could be considered as the number of channels currently NOT available to new calls). In this section, a composite model constructed using a two-dimensional Markov chain is employed to model this system. Later in Section 2.3.1 a two-level hierarchical model will be built in order to consider the availability model and performance model separately. The state transition diagram of the composite model is shown in Figure 2.1. Note that the system capacity i indicates that there are $n - i$ ongoing handoff calls in the system. Also notice that there can be no transition from state (i,j) to $(i-1, j-1)$ when $i \neq j$, since a handoff call can only preempt an ongoing new call when all channels are busy (i.e., when $i = j$). Based on the transition diagram, the system can be modeled as a homogeneous irreducible continuous time Markov chain with $(c + n + 2)(n - c + 1)/2$ states.

After ordering all the states lexicographically as $\{(c,0), ..., (c,c), (c+1,0), ..., (c+1, c+1), ...(n,0), ...(n,n)\}$, the infinitesimal generator $\boldsymbol{G}$ can be written as:

$$\boldsymbol{G} = \begin{vmatrix} \boldsymbol{L}^{(c)} & \boldsymbol{F}^{(c)} & & & \cdots & & & \\ \boldsymbol{B}^{(c+1)} & \boldsymbol{L}^{(c+1)} & \boldsymbol{F}^{(c+1)} & & \cdots & & & \\ & \boldsymbol{B}^{(c+2)} & \boldsymbol{L}^{(c+2)} & \boldsymbol{F}^{(c+2)} & \cdots & & & \\ & & & & \cdots & & & \\ & & & & & \boldsymbol{B}^{(n-1)} & \boldsymbol{L}^{(n-1)} & \boldsymbol{F}^{(n-1)} \\ & & & & & & \boldsymbol{B}^{(n)} & \boldsymbol{L}^{(n)} \end{vmatrix} \tag{2.1}$$

where $\boldsymbol{B}^{(i)}$ is an $i \times (i-1)$ matrix and has the following form:

$$\boldsymbol{B}^{(i)} = \begin{vmatrix} \lambda_2 & & & & \\ & \lambda_2 & & & \\ & & \lambda_2 & & \\ & & & \ddots & \\ & & & & \lambda_2 \\ & & & & \lambda_2 \end{vmatrix}, \tag{2.2}$$

$\boldsymbol{L}^{(i)}$ is an $i \times i$ matrix and

$$\boldsymbol{L}^{(c)} = \begin{vmatrix} Diag & \lambda_1 & & & & \\ \mu_1 & Diag & \lambda_1 & & & \\ & 2\mu_1 & Diag & \lambda_1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & (c-1)\mu_1 & Diag & \lambda_1 \\ & & & & c\mu_1 & Diag \end{vmatrix}, \tag{2.3}$$

$$\boldsymbol{L}^{(i)} = \begin{vmatrix} Diag & \lambda_1 & & & & \\ \mu_1 & Diag & \lambda_1 & & & \\ & 2\mu_1 & Diag & \lambda_1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & (i-1)\mu_1 & Diag & \lambda_1 \\ & & & & i\mu_1 & Diag \end{vmatrix}, (c < i < n), \tag{2.4}$$

$$\boldsymbol{L}^{(n)} = \begin{vmatrix} Diag & \lambda_1 & & & & \\ \mu_1 & Diag & \lambda_1 & & & \\ & 2\mu_1 & Diag & \lambda_1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & (n-1)\mu_1 & Diag & \lambda_1 \\ & & & & n\mu_1 & Diag \end{vmatrix} \tag{2.5}$$

where $Diag$ is the diagonal element of the infinitesimal generator in the given row, which is the negative sum of all the remaining elements in the same row.

$\boldsymbol{F}^{(i)}$ is an $i \times (i+1)$ matrix and has form:

$$\boldsymbol{F}^{(i)} = \begin{vmatrix} (n-i)\mu_2 & & & & & \\ & (n-i)\mu_2 & & & & \\ & & (n-i)\mu_2 & & & \\ & & & \ddots & & \\ & & & & (n-i)\mu_2 & 0 \end{vmatrix}. \tag{2.6}$$

Note that $\boldsymbol{G}$ is not a block tridiagonal matrix since $\boldsymbol{B}^{(i)}$ and $\boldsymbol{F}^{(i)}$ are not square matrices. The steady state distribution $\boldsymbol{\pi}$ can be obtained by solving $\boldsymbol{\pi G} = 0$.

2.2.2 Performance metrics

The performance measures of interest are handoff call blocking probability (P_b^h), new call blocking probability (P_b^N), and new call dropping probability (P_d^N). The calculation of these performance measures will be presented in this section. First of all, although handoff calls can access only the g guard channels, they have priority over new calls on these g channels. When the number of ongoing handoff calls is less than g and an incoming handoff call sees all channels busy, instead of being blocked, it can reserve a channel for itself by dropping an ongoing new call. The blocking probability of handoff calls is not affected by the presence of new calls. Therefore, handoff calls in this model can be represented by an $M/M/g/g$ loss system, and the blocking probability can be calculated by the well-known Erlang B formula as:

$$P_b^h = EB(\frac{\lambda_2}{\mu_2}, g). \tag{2.7}$$

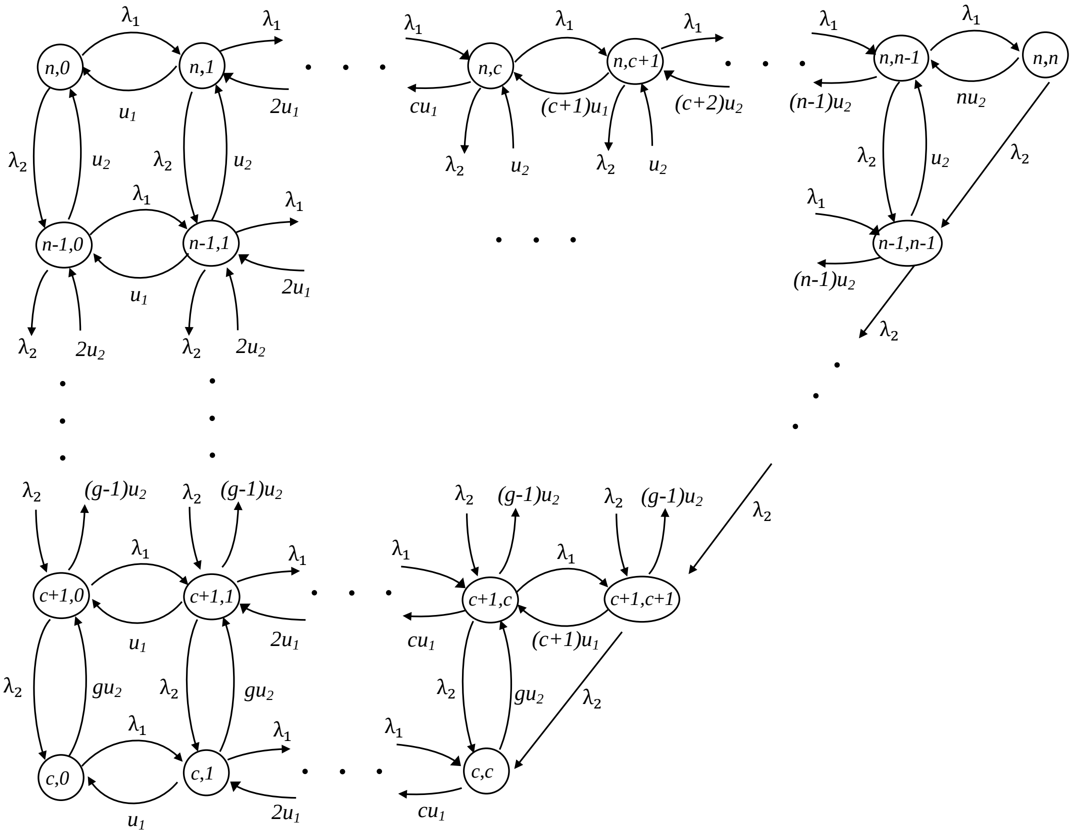


Figure 2.1: State transition diagram of the M1 model

To calculate the new call blocking probability P_b^N, let us define Ω_b^N as the set containing all the blocking states for new calls: $\Omega_b^N = \{(i,j) | i = j,\ (i,j) \in \Omega\}$. Then there exists

$$P_b^N = \sum_{(i,j)\in\Omega_b^N} \pi_{(i,j)}. \tag{2.8}$$

where $\pi_{(i,j)}$ is the steady state probability for state (i,j).

To calculate the new call dropping probability P_d^N, we define Ω_d^N as a set containing all the states that can initiate call dropping transitions. A call dropping transition is a transition that leads to a call dropping event. Assume that the system is currently in state (i,j), then a dropping event can occur only when both of the following two conditions are satisfied:

1) The system is currently full and the number of ongoing handoff calls is less than g (which could be represented by $i = j$ and $i \neq n - g$), and

2) a handoff call arrives and is admitted by dropping a new call. The system transits to

state $(i-1, j-1)$.

Subsequently, all the states that satisfy the first condition can initiate call dropping transitions and should be included in Ω_d^N, i.e., $\Omega_d^N = \{(i,j) | i = j \text{ and } i \neq n-g,\ (i,j) \in \Omega\}$. The new call dropping probability follows:

$$\begin{aligned} P_d^N &= \frac{\{\text{The number of new calls being dropped per unit time}\}}{\{\text{The number of incoming new calls per unit time}\}} \\ &= \frac{\lambda_2 \sum_{(i,j)\in\Omega_d^N} \pi(i,j)}{\lambda_1}. \end{aligned} \tag{2.9}$$

Combining the blocking and dropping probabilities of new calls together produces the overall loss probability of new calls:

$$P_L^N = P_b^N + P_d^N. \tag{2.10}$$

Let us revisit the example presented at the beginning of this chapter and describe it with the notations just developed: $n = 15$, $\lambda_1 = 2$, $\lambda_2 = 5$, and $\mu_1 = \mu_2 = 1$. When using the original model, we have $P_b^h \approx 0.016\%$ and $P_L^N \approx 1.1\%$ while the thresholds for P_b^h and P_L^N are 0.5% and 1%, respectively. Now the first guard channel model is employed to calculate the performance measures through the composite model method. Set $g = 12$ will then lead to $P_b^h \approx 0.34\%$ and $P_L^N \approx 0.73\%$. Both of them are now below their thresholds. Furthermore, the new call dropping probability P_d^N is reduced by 41% (from 0.79% to 0.46%).

Since the solution of the composite model would become intractable when the number of channels n is large[1], other methods that can handle large n's are called for. In Section 2.3, two approximate approaches will be introduced to effectively approximate new call blocking probability (P_b^N) and new call dropping probability (P_d^N)[2]. Then, a recursive method is presented in Section 2.4 as a numerical alternative to the composite method. A comparison among all available methods will be carried out in Section 2.5 where the performance of numerical methods and approximate methods are compared to simulation results.

[1] A desktop with Intel(R) i5 processor, 8GB ram, and Windows 7 32bit installed can calculate up to about 120 channels.

[2] There is no need to "approximate" handoff call blocking probability, as it can be calculated exactly by the Erlang B formula. The numerically stable method introduced in the Appendix can be used when the number of channels is large.

2.3 Approximate methods

2.3.1 Hierarchical model method

In the first approximate method, a two-level hierarchical model is constructed to estimate the performance measures of the M1 model. Such methodology is well-known in the field of performance modelling. Trivedi et al. (2003) chose this method to approximate the blocking probability of a pure loss system with server break-downs and repairs [58]. With appropriate modification this method can be also be applied to the M1 model.

The hierarchical model is composed of an upper level model which is an availability model, and a sequence of lower-level performance models. The availability model is essentially one that describes the stochastic evolution of the system capacity for new calls. Each state i of the availability model is a possible value of system capacity for new calls and is assigned a reward rate which is derived from the lower-level performance model with the same system capacity. Recall that in the M1 model the system capacity for new calls, i, is equal to the difference between n and the number of ongoing handoff calls in the system; therefore i can take value in $n - g$, $n - g + 1$, ..., n. Since the performance measure of interest here is the new call blocking probability, the reward rate assigned to state i of the availability model should be the blocking probability derived from the performance model with capacity i. The transition diagram of the capacity model, which accounts for the capacity evolution (for new calls) in the M1 model, is presented in Figure 2.2.

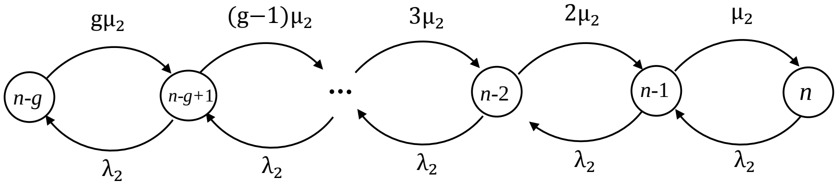


Figure 2.2: Hierarchical model approach: state transition diagram of the capacity model

This is a homogeneous skip-free continuous-time Markov chain. Its steady state probabilities are given by

$$
\begin{aligned}
\pi_{n-g} &= \left(\sum_{k=n-g}^{n} \frac{\frac{g!}{(n-k)!} \cdot \mu_2^{k-(n-g)}}{\lambda_2^{k-(n-g)}} \right)^{-1} \\
\pi_i &= \pi_{n-g} \cdot \frac{\frac{g!}{(n-i)!} \cdot \mu_2^{i-(n-g)}}{\lambda_2^{i-(n-g)}}, \quad \text{for } i = n-g+1, ..., n-1, n.
\end{aligned}
\tag{2.11}
$$

Now, consider the performance model with system capacity i. The state transition diagram is shown in Figure 2.3.

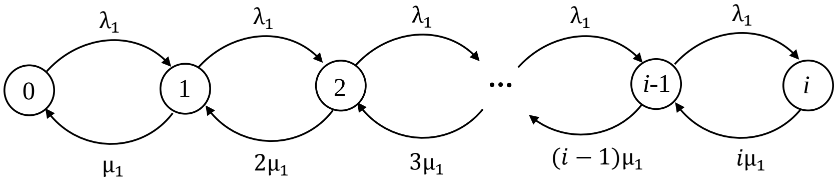


Figure 2.3: Hierarchical model approach: state diagram of performance model when system capacity is i

This is an $M/M/i/i$ queueing system, and its steady state probabilities are given by

$$
\begin{aligned}
v_0 &= \left(\sum_{k=0}^{i} \frac{\rho_1^k}{k!} \right)^{-1} \\
v_j &= v_0 \cdot \frac{\rho_1^j}{j!}, \quad \text{for } j = 1, 2, \ldots i.
\end{aligned}
\tag{2.12}
$$

Then, the blocking probability for new calls of this queueing system, denoted by $P_b(i)$, can be calculated by the Erlang B formula as:

$$
P_b(i) = v_i = \frac{\rho_1^i}{i!} \left(\sum_{k=0}^{i} \frac{\rho_1^k}{k!} \right)^{-1}. \tag{2.13}
$$

Now let us consider $P_b(i)$ as the reward rate assigned to the state i of the capacity model. The total new call blocking probability of the M1 model can be approximately computed as the expected reward rate when the system is at equilibrium and is given by

$$\begin{aligned} P_b^N &\approx \sum_{i=n-g}^{n} P_b(i)\pi_i \\ &= \sum_{i=n-g}^{n} \left[\frac{\rho_1^i}{i!} \left(\sum_{k=0}^{i} \frac{\rho_1^k}{k!} \right)^{-1} \cdot \frac{\frac{g!}{(n-i)!} \cdot u_2^{i-(n-g)}}{\lambda_2^{i-(n-g)}} \left(\sum_{k=n-g}^{n} \frac{\frac{g!}{(n-k)!} \cdot u_2^{k-(n-g)}}{\lambda_2^{k-(n-g)}} \right)^{-1} \right]. \end{aligned} \tag{2.14}$$

Since we are essentially using $P_b(i)\pi_i$ to approximate the steady state probability of state (i,i) in the composite model where $i = n-g, n-g+1, ...n$, the dropping probability can then be approximated according to Equation 2.9:

$$P_d^N \approx \frac{\lambda_2}{\lambda_1} \sum_{i=n-g+1}^{n} P_b(i)\pi_i. \tag{2.15}$$

2.3.2 Effective capacity method

The method of effective capacity is a simple yet efficient approach for estimating the new call blocking probability of our first guard channel model. The effective capacity (EC) can be interpreted as the average number of channels available for new calls after the system achieves equilibrium. Since handoff calls in the M1 model can occupy at most g channels simultaneously, and can be modeled as an $M/M/g/g$ loss system, it is fairly straightforward to demonstrate that the effective capacity for new calls can be calculated as

$$EC = n - g + (\text{average number of free channels in the } M/M/g/g \text{ system}). \tag{2.16}$$

The average number of free channels in $M/M/g/g$ is given by $g - \overline{Q}$ where $\overline{Q}$ is the mean queue length of the $M/M/g/g$ queueing system:

$$\overline{Q} = \textstyle\sum_{i=0}^{g} i\pi_i. \tag{2.17}$$

where π_i in the above equation is the steady state probabilities of state i in the $M/M/g/g$

queueing system. Therefore, we have

$$\begin{aligned} EC &= n - g + g - \overline{Q} \\ &= (n - \overline{Q}) \text{ and rounded to the nearest integer.} \end{aligned} \tag{2.18}$$

Now the new call blocking probability (P_b^N) of the M1 model can be approximated with the loss system $M/M/EC/EC$:

$$P_b^N \approx EB(\frac{\lambda_1}{\mu_1}, EC). \tag{2.19}$$

This method cannot be utilized, though, to approximate dropping probability for new calls.

2.4 The recursive method

This section introduces a recursive technique that can be used to solve the steady state probabilities of the M1 model. Herzog et al. [19] first employed such a technique to analyze a wide class of queueing systems whose interarrival and service times were described by multidimensional Markovian processes. Then, Alam and Mani [3] tested a similar recursive technique to study the steady state probabilities of a multi-server, first-come, first-served queueing system which alternates between two modes of system operation. In the following sections, we first develop a recursive approach for the case when $g = 1$ of the M1 model, and then extend it to the general cases when $g > 1$.

2.4.1 When $g = 1$

Figure 2.4 shows the state transition diagram of the M1 model when $g = 1$. The state space is defined as $\Omega_{g=1} = \{(i,j) | i = n \text{ or } n-1, 0 \leq j \leq i\}$, where j denotes the number of new calls in the system and i denotes the system capacity for new calls. $P_{(i,j)}$ represent the steady state probability of state (i,j) (where i can be $n-1$ or n); then, the balance equations for this system can be written as

- When $j = 0$:

$$(\lambda_1 + \lambda_2) P_{(n,0)} = \mu_1 P_{(n,1)} + \mu_2 P_{(n-1,0)}, \tag{2.20}$$

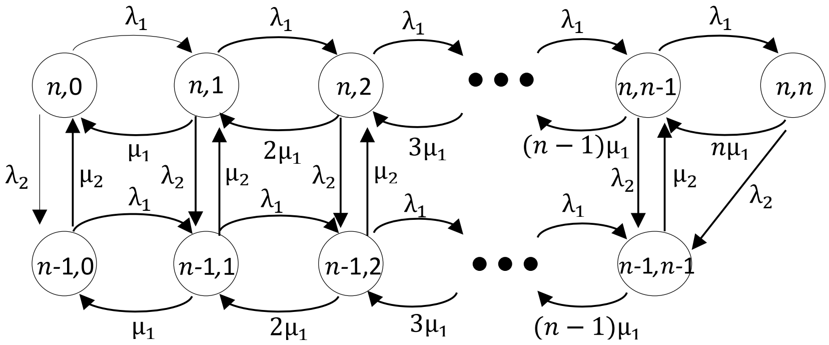


Figure 2.4: State transition diagram of the M1 model when $g = 1$

$$(\lambda_1 + \mu_2)P_{(n-1,0)} \;=\; \mu_1 P_{(n-1,1)} + \lambda_2 P_{(n,0)}. \tag{2.21}$$

- When $j = 1, 2, ..., n - 2$:

$$\begin{aligned}(\lambda_1 + \lambda_2 + j\mu_1)P_{(n,j)} &= \lambda_1 P_{(n,j-1)} \\ &\quad + (j+1)\mu_1 P_{(n,j+1)} + \mu_2 P_{(n-1,j)}, \qquad (2.22) \\ (\lambda_1 + \mu_2 + j\mu_1)P_{(n-1,j)} &= \lambda_1 P_{(n-1,j-1)} \\ &\quad + (j+1)\mu_1 P_{(n-1,j+1)} + \lambda_2 P_{(n,j)}. \qquad (2.23)\end{aligned}$$

- When $j = n - 1$:

$$\begin{aligned}(\lambda_1 + \lambda_2 + (n-1)\mu_1)P_{(n,n-1)} &= \lambda_1 P_{(n,n-2)} \\ &\quad + n\mu_1 P_{(n,n)} + \mu_2 P_{(n-1,n-1)}, \qquad (2.24) \\ (\mu_2 + (n-1)\mu_1)P_{(n-1,n-1)} &= \lambda_2 P_{(n,n)} \\ &\quad + \lambda_2 P_{(n,n-1)} + \lambda_1 P_{(n-1,n-2)}. \qquad (2.25)\end{aligned}$$

- When $j = n$:

$$(\lambda_2 + n\mu_1)P_{(n,n)} = \lambda_1 P_{(n,n-1)}. \tag{2.26}$$

The general idea of this recursive technique is to define a subset of state probabilities as boundary points. Then the next step is to express the rest state probabilities in terms of these boundary points and at last, to solve a reduced system of equations for these boundary points. In this case, we choose $P_{(n,0)}$ and $P_{(n-1,0)}$ as boundary points[3] and the procedures are outlined as follows.

Step 1: model reduction

Write all remaining state probabilities in terms of the chosen boundary points, i.e

$$P_{(i,j)} = C^1_{(i,j)} P_{(n,0)} + C^2_{(i,j)} P_{(n-1,0)} \tag{2.27}$$

where $(i,j) \in \Omega_{g=1}$. $C^1_{(i,j)}$ and $C^2_{(i,j)}$ are the unknown coefficients of the two boundary points $P_{(n,0)}$ and $P_{(n-1,0)}$ for state (i,j), respectively[4]. To successfully construct the reduced system of two equations with two unknowns (that is, the two boundary points), we need to first determine all the coefficients $C^r_{(i,j)}$ where $r \in \{1,2\}$. This step establish the recursive relations of $C^r_{(i,j)}$'s by way of Equations 2.20 - 2.26.

- Initial values I: Because the two boundary points can also be expressed in term of themselves as:

$$P_{(n,0)} = 1 \cdot P_{(n,0)} - 0 \cdot P_{(n-1,0)} \tag{2.28}$$

and

$$P_{(n-1,0)} = 0 \cdot P_{(n,0)} - 1 \cdot P_{(n-1,0)}, \tag{2.29}$$

we obtain the coefficients when $j = 0$ as:

$$C^1_{(n,0)} = 1, \;\; C^2_{(n,0)} = 0, \;\; C^1_{(n-1,0)} = 0, \;\; C^2_{(n-1,0)} = 1. \tag{2.30}$$

- Initial values II:

[3] The choice of boundary points is not unique. For example, using $P_{(n,n)}$ and $P_{(n-1,n-1)}$ as boundary points also works, but will lead to slightly different recursive formulae than those that are presented in this section.

[4] More specifically, the superscript 1 corresponds to the first boundary point $P_{(n,0)}$ and the superscript 2 corresponds to the second boundary point $P_{(n-1,0)}$

– when $i = n$, according to Equation 2.20, there follows

$$P_{(n,1)} = \frac{\lambda_1 + \lambda_2}{\mu_1} P_{(n,0)} - \frac{\mu_2}{\mu_1} P_{(n-1,0)}, \tag{2.31}$$

from which the coefficients can be extracted:

$$C^1_{(n,1)} = \frac{\lambda_1 + \lambda_2}{\mu_1}, \quad C^2_{(n,1)} = -\frac{\mu_2}{\mu_1}. \tag{2.32}$$

– When $i = n - 1$, so that now according to Equation 2.21, we have

$$P_{(n-1,1)} = -\frac{\lambda_2}{\mu_1} P_{(n,0)} + \frac{\lambda_1 + \mu_2}{\mu_1} P_{(n-1,0)}, \tag{2.33}$$

from which we can obtain the following the coefficients:

$$C^1_{(n-1,1)} = -\frac{\lambda_2}{\mu_1}, \quad C^2_{(n-1,1)} = \frac{\lambda_1 + \mu_2}{\mu_1}. \tag{2.34}$$

- When $j \in \{1, 2, ..., n - 1\}$ and $i = n$, by Equation 2.22 and 2.24

$$\begin{aligned} P_{(n,j+1)} = & \frac{(\lambda_1 + \lambda_2 + j\mu_1)}{(j+1)\mu_1} P_{(n,j)} \\ & - \frac{\lambda_1}{(j+1)\mu_1} P_{(n,j-1)} - \frac{\mu_2}{(j+1)\mu_1} P_{(n-1,j)}. \end{aligned} \tag{2.35}$$

By rewriting all the state probabilities in the above equation in terms of the boundary points with corresponding coefficients, we have

$$\begin{aligned} & C^1_{(n,j+1)} P_{(n,0)} + C^2_{(n,j+1)} P_{(n-1,0)} \\ & = \frac{(\lambda_1 + \lambda_2 + j\mu_1)}{(j+1)\mu_1} (C^1_{(n,j)} P_{(n,0)} + C^2_{(n,j)} P_{(n-1,0)}) \\ & - \frac{\lambda_1}{(j+1)\mu_1} (C^1_{(n,j-1)} P_{(n,0)} + C^2_{(n,j-1)} P_{(n-1,0)}) \\ & - \frac{\mu_2}{(j+1)\mu_1} (C^1_{(n-1,j)} P_{(n,0)} + C^2_{(n-1,j)} P_{(n-1,0)}). \end{aligned} \tag{2.36}$$

Consequently, it is not difficult to derive the following recursive relationship by equating the coefficients of like terms:

$$\begin{aligned}C^r_{(n,j+1)} &= \frac{(\lambda_1+\lambda_2+j\mu_1)}{(j+1)\mu_1}C^r_{(n,j)}\\ &\quad-\frac{\lambda_1}{(j+1)\mu_1}C^r_{(n,j-1)}-\frac{\mu_2}{(j+1)\mu_1}C^r_{(n-1,j)}\end{aligned} \tag{2.37}$$

where $r = 1$ or 2.

- When $j \in \{1, 2, ..., n-2\}$ and $i = n-1$, by Equation 2.23, the following situation exists:

$$\begin{aligned}P_{(n-1,j+1)} &= \frac{(\lambda_1+\mu_2+j\mu_1)}{(j+1)\mu_1}P_{(n-1,j)}\\ &\quad-\frac{\lambda_1}{(j+1)\mu_1}P_{(n-1,j-1)}-\frac{\lambda_2}{(j+1)\mu_1}P_{(n,j)}.\end{aligned} \tag{2.38}$$

Again, let us rewrite all the state probabilities in the above equation in terms of the boundary points with their corresponding coefficients:

$$\begin{aligned}&C^1_{(n-1,j+1)}P_{(n,0)}+C^2_{(n-1,j+1)}P_{(n-1,0)}\\ &=\frac{(\lambda_1+\mu_2+j\mu_1)}{(j+1)\mu_1}(C^1_{(n-1,j)}P_{(n,0)}+C^2_{(n-1,j)}P_{(n-1,0)})\\ &-\frac{\lambda_1}{(j+1)\mu_1}(C^1_{(n-1,j-1)}P_{(n,0)}+C^2_{(n-1,j-1)}P_{(n-1,0)})\\ &-\frac{\lambda_2}{(j+1)\mu_1}(C^1_{(n,j)}P_{(n,0)}+C^2_{(n,j)}P_{(n-1,0)}),\end{aligned} \tag{2.39}$$

and it is followed by a recursive relationship:

$$\begin{aligned}C^r_{(n-1,j+1)} &= \frac{(\lambda_1+\mu_2+j\mu_1)}{(j+1)\mu_1}C^r_{(n-1,j)}\\ &\quad-\frac{\lambda_1}{(j+1)\mu_1}C^r_{(n-1,j-1)}-\frac{\lambda_2}{(j+1)\mu_1}C^r_{(n,j)}\end{aligned} \tag{2.40}$$

where $r = 1$ or 2.

- Using the recursive relationship presented in Equations 2.37 and 2.40 together with the initial values listed in Equations 2.32 and 2.34, one may determine all the unknown coefficients $C^r_{(i,j)}$, where $r = 1, 2$ and $(i, j) \in \Omega_{g=1}$.

Step 2: compute steady state probabilities

In this step we construct and solve a system of two equations of unknown boundary points $P_{(n,0)}$ and $P_{(n-1,0)}$.

- The first equation: the first equation can be obtained by rewriting Equation 2.26 in terms of the two boundary points and combining like terms as follows:

$$\begin{aligned}
&(\lambda_2 + n\mu_1)P_{(n,n)} = \lambda_1 P_{(n,n-1)} \\
\Rightarrow &(\lambda_2 + n\mu_1)(C^1_{(n,n)}P_{(n,0)} + C^2_{(n,n)}P_{(n-1,0)}) = \lambda_1(C^1_{(n,n-1)}P_{(n,0)} + C^2_{(n,n-1)}P_{(n-1,0)}) \\
\Rightarrow &[(\lambda_2 + n\mu_1)C^1_{(n,n)} - \lambda_1 C^1_{(n,n-1)}]P_{(n,0)} = [-(\lambda_2 + n\mu_1)C^2_{(n,n)} + \lambda_1 C^2_{(n,n-1)}]P_{(n-1,0)} \\
\Rightarrow &P_{(n,0)} = \frac{-(\lambda_2 + n\mu_1)C^2_{(n,n)} + \lambda_1 C^2_{(n,n-1)}}{(\lambda_2 + n\mu_1)C^1_{(n,n)} - \lambda_1 C^1_{(n,n-1)}} P_{(n-1,0)}. \qquad (2.41)
\end{aligned}$$

- The second equation: The second equation represents the normalizing condition which demands that all steady state probabilities should sum up to 1:

$$\begin{aligned}
&\sum_{(i,j)\in\Omega_{g=1}} P_{(i,j)} = 1 \\
\Rightarrow &\sum_{(i,j)\in\Omega_{g=1}} (C^1_{(i,j)}P_{(n,0)} + C^2_{(i,j)}P_{(n-1,0)}) = 1 \\
\Rightarrow &P_{(n,0)} \cdot \sum_{(i,j)\in\Omega_{g=1}} C^1_{(i,j)} + P_{(n-1,0)} \cdot \sum_{(i,j)\in\Omega_{g=1}} C^2_{(i,j)} = 1. \qquad (2.42)
\end{aligned}$$

- The solution: By solving Equation 2.41 and 2.42 simultaneously, we have

$$P_{(n,0)} = \frac{1}{\sum_{(i,j)\in\Omega_{g=1}} C^1_{(i,j)} + h \cdot \sum_{(i,j)\in\Omega_{g=1}} C^2_{(i,j)}} \qquad (2.43)$$

  where

$$h = \frac{(\lambda_2 + n\mu_1)C^1_{(n,n)} - \lambda_1 C^1_{(n,n-1)}}{-(\lambda_2 + n\mu_1)C^2_{(n,n)} + \lambda_1 C^2_{(n,n-1)}}, \qquad (2.44)$$

  and then $P_{(n-1,0)}$ can be calculated using its relationship with $P_{(n,0)}$ as follows:

$$P_{(n-1,0)} = hP_{(n,0)}. \qquad (2.45)$$

Step 3: compute performance measures

After the boundary points are obtained, all the remaining state probabilities can also be calculated by substituting the coefficients and the boundary points into Equation 2.27. The performance measures, (i.e., handoff call blocking probability, new call blocking probability and new call dropping probability) then can be calculated by the formulae presented in Section 2.2.2.

2.4.2 Solution verification with some special cases

In order to gain a further insight into this system, as well as to check the recursive solutions developed in the last section, we now consider four special cases.

In **case I**, we set $\lambda_2 = \mu_2 = 0$ and let the model interpretation for this case be that handoff calls are completely removed from the system, and the new calls are the only kind of traffic that can still access the system. Since there is no higher priority traffic to compete with, new calls can use all n channels, and our system reduces to an $M/M/n/n$ loss system. Now, let us re-examine the formulae derived in the last section by setting $\lambda_2 = 0$ and $\mu_2 = 0$. Note that $P_{(n,0)}$ and $P_{(n-1,0)}$ remains as the boundary points. And, the condition must exist that $P_{(n-1,0)} = 0$ for $j = 0, 1, ..., n-1$ because when there is no handoff call, the system capacity for new calls should always be n.

- Initial values: Set $\lambda_2 = 0$ and $\mu_2 = 0$ in Equations 2.32 and 2.34, and there follows

$$C^1_{(n,1)} = \frac{\lambda_1}{\mu_1}, \tag{2.46}$$

$$C^2_{(n,1)} = 0, \tag{2.47}$$

$$C^1_{(n-1,1)} = 0, \text{ and} \tag{2.48}$$

$$C^2_{(n-1,1)} = \frac{\lambda_1}{\mu_1}. \tag{2.49}$$

- Recursive relations:
  - When $j = 1, 2, ..., n-1$ and $i = n$, substitute zero for λ_2 and μ_2 in Equation 2.37

to produce

$$\begin{aligned} C^r_{(n,j+1)} = {} & \frac{(\lambda_1 + j\mu_1)}{(j+1)\mu_1} C^r_{(n,j)} \\ & - \frac{\lambda_1}{(j+1)\mu_1} C^r_{(n,j-1)} - 0, \ r = 1, 2. \end{aligned} \tag{2.50}$$

Since $C^2_{(n,0)}$ (refer to Equation 2.30) and $C^2_{(n,1)}$ are both equal to zero, it is easy to see that all $C^2_{(n,j)}$'s are zero. Now, by taking a closer look at Equation 2.50, one may use it to calculate the first few values of $C^1_{(n,j)}$:

$$C^1_{(n,0)} = 1 \text{ (refer to Equation 2.30)} \tag{2.51}$$

$$C^1_{(n,1)} = \frac{\lambda_1}{\mu_1} \tag{2.52}$$

$$C^1_{(n,2)} = \frac{\lambda_1 + \mu_1}{2\mu_1} \cdot \frac{\lambda_1}{\mu_1} - \frac{\lambda_1}{2\mu_1} \cdot 1 = \frac{\lambda_1^2}{2\mu_1^2} = \left(\frac{\lambda_1}{\mu_1}\right)^2 \Bigg/ 2! \tag{2.53}$$

$$C^1_{(n,3)} = \frac{\lambda_1 + \mu_1}{2\mu_1} \cdot \frac{\lambda_1^2}{2\mu_1^2} - \frac{\lambda_1}{2\mu_1} \cdot \frac{\lambda_1}{\mu_1} = \frac{\lambda_1^3}{6\mu_1^3} = \left(\frac{\lambda_1}{\mu_1}\right)^3 \Bigg/ 3!. \tag{2.54}$$

Next, assume that

$$C^1_{(n,j)} = \left(\frac{\lambda_1}{\mu_1}\right)^j \Bigg/ j!, \text{ for } j = 0, 1, 2, ..., n, \tag{2.55}$$

and prove it through mathematical induction on j.

Proof. Assume that

$$C^1_{(n,j-1)} = \left(\frac{\lambda_1}{\mu_1}\right)^{j-1} \Bigg/ (j-1)!, \text{where } j = 1, 2, ..., n, \tag{2.56}$$

is true, then by Equation 2.50, the calculation is

$$\begin{aligned} C^1_{(n,j)} &= \frac{\lambda_1 + \mu_1}{j\mu_1} \cdot \frac{\lambda_1^{j-1}}{(j-1)!\mu_1^{j-1}} - \frac{\lambda_1}{j\mu_1} \cdot \frac{\lambda_1^{j-2}}{(j-1)!\mu_1^{j-2}} \\ &= \left(\frac{\lambda_1}{\mu_1}\right)^j \Bigg/ j!. \end{aligned} \tag{2.57}$$

Therefore, Equation 2.55 holds for $j = 0, 1, 2, ..., n$.

$\blacksquare$

- when $j = 1, 2, ..., n-1$ and $i = n-1$, by setting $\lambda_2 = 0$ and $\mu_2 = 0$ in Equation 2.40, we obtain

$$\begin{aligned} C^r_{(n-1,j+1)} = & \frac{(\lambda_1 + j\mu_1)}{(j+1)\mu_1} C^r_{(n-1,j)} \\ & - \frac{\lambda_1}{(j+1)\mu_1} C^r_{(n-1,j-1)} - 0, \ r = 1, 2. \end{aligned} \tag{2.58}$$

Again, all $C^1_{(n-1,j)}$'s are zero because $C^1_{(n-1,0)}$ and $C^1_{(n-1,1)}$ are both zero. Combining this situation with the fact that $P_{(n-1,0)} = 0$, it is not difficult to see from the following equation:

$$P_{(n-1,j)} = C^1_{(n-1,j)} P_{(n,0)} + C^2_{(n-1,j)} P_{(n-1,0)} \tag{2.59}$$

that $P_{(n-1,j)}$'s $(j = 0, 1, 2..., n-1)$ are all equal to zero.

- Solutions: Since $P_{(n-1,0)} = 0$, we have only one unknown variable $P_{(n,0)}$, and it is ready to be determined by the normalizing condition as follows:

$$\begin{aligned} & \sum_{(i,j) \in \Omega_{g=1}} P_{(i,j)} = 1 \\ \Rightarrow & \sum_{(i,j) \in \Omega_{g=1}} (C^1_{(i,j)} P_{(n,0)} + C^2_{(i,j)} P_{(n-1,0)}) = 1 \\ \Rightarrow & \sum_{(i,j) \in \Omega_{g=1}} C^1_{(i,j)} P_{(n,0)} = 1 \quad (\text{because } P_{(n-1,0)} = 0) \\ \Rightarrow & \sum_{j=0}^{n} C^1_{(n,j)} P_{(n,0)} = 1 \quad (\text{because } C^1_{(n-1,j)} = 0, \ j = 1, 2, ..., n-1) \\ \Rightarrow & P_{(n,0)} \sum_{j=0}^{n} C^1_{(n,j)} = 1 \\ \Rightarrow & P_{(n,0)} = \left(\sum_{j=0}^{n} \left[\left(\frac{\lambda_1}{\mu_1} \right)^j \Big/ j! \right] \right)^{-1} \quad (\text{by Equation 2.55}). \end{aligned} \tag{2.60}$$

Therefore, all the remaining state probabilities can be obtained:

$$P_{(i,j)} = \begin{cases} P_{(n,0)} \cdot \left(\frac{\lambda_1}{\mu_1}\right)^j \Big/ j!, & \text{when } i = n; \\ 0, & \text{when } i = n-1. \end{cases} \tag{2.61}$$

This result confirms the intuitive interpretation made at the beginning of this section: when $\lambda_2 = 0$ and $\mu_2 = 0$, the system is an $M/M/n/n$ loss system.

Now let us define

$$\rho_1 = \lambda_1/\mu_1, \quad \rho_2 = \lambda_2/\mu_2, \quad \alpha = \mu_1/\mu_2. \tag{2.62}$$

To reparametrize the recursive formulae using α, ρ_1 and ρ_2, we divide the original parameters (that is, λ_1, λ_2, μ_1 and μ_2) by μ_2 and establish the rules of correspondence for parameter conversion in Table 2.1:

Table 2.1: Parameter conversion table

Previous Parameter		New Parameter
λ_1	→	$\alpha\rho_1$
λ_2	→	ρ_2
μ_1	→	α
μ_2	→	1

- Now, the initial values in (2.30), (2.32) and (2.34) can be written in terms of the new parameters as

$$\begin{array}{llll} C^1_{(n,0)} = 1 & C^2_{(n,0)} = 0 & C^1_{(n,1)} = \dfrac{\alpha\rho_1 + \rho_2}{\alpha} & C^2_{(n,1)} = -\dfrac{1}{\alpha} \\ C^1_{(n-1,0)} = 0 & C^2_{(n-1,0)} = 1 & C^1_{(n-1,1)} = -\dfrac{\rho_2}{\alpha} & C^2_{(n-1,1)} = \dfrac{\alpha\rho_1 + 1}{\alpha}. \end{array} \tag{2.63}$$

- When $j = 1, 2, ..., n-2$, the recursive relationships among coefficients (Equation 2.37

and 2.40) can also be expressed with the new set of parameters as

$$\begin{aligned} C^r_{(n,j+1)} = & \frac{(\alpha\rho_1 + \rho_2 + j\alpha)}{(j+1)\alpha} C^r_{(n,j)} \\ & - \frac{\rho_1}{(j+1)} C^r_{(n,j-1)} - \frac{1}{(j+1)\alpha} C^r_{(n-1,j)} \end{aligned} \tag{2.64}$$

and

$$\begin{aligned} C^r_{(n-1,j+1)} = & \frac{(\alpha\rho_1 + 1 + j\alpha)}{(j+1)\alpha} C^r_{(n-1,j)} \\ & - \frac{\rho_1}{(j+1)} C^r_{(n-1,j-1)} - \frac{\rho_2}{(j+1)\alpha} C^r_{(n,j)}. . \end{aligned} \tag{2.65}$$

- Then the system of two equations for solving the two boundary points (as displayed in (2.41) and (2.42)) are apparent:

$$\begin{cases} P_{(n,0)} = \frac{-(\rho_2+n\alpha)C^2_{(n,n)}+\alpha\rho_1 C^2_{(n,n-1)}}{(\rho_2+n\alpha)C^1_{(n,n)}-\alpha\rho_1 C^1_{(n,n-1)}} P_{(n-1,0)} \\ P_{(n,0)} \cdot \sum_{(i,j)\in\Omega_{g=1}} C^1_{(i,j)} + P_{(n-1,0)} \cdot \sum_{(i,j)\in\Omega_{g=1}} C^2_{(i,j)} = 1 \end{cases}, \tag{2.66}$$

and the expression of $P_{(n,0)}$ can be obtained by

$$P_{(n,0)} = \frac{1}{\sum_{(i,j)\in\Omega_{g=1}} C^1_{(i,j)} + h \cdot \sum_{(i,j)\in\Omega_{g=1}} C^2_{(i,j)}} \tag{2.67}$$

where

$$h = \frac{(\rho_2 + n\alpha)C^1_{(n,n)} - \alpha\rho_1 C^1_{(n,n-1)}}{-(\rho_2 + n\alpha)C^2_{(n,n)} + \alpha\rho_1 C^2_{(n,n-1)}}. \tag{2.68}$$

In **case II**, let α approach infinity (∞). The initial values in (2.63) become

$$\begin{array}{llll} \lim\limits_{\alpha\to\infty} C^1_{(n,0)} = 1 & \lim\limits_{\alpha\to\infty} C^2_{(n,0)} = 0 & \lim\limits_{\alpha\to\infty} C^1_{(n,1)} = \rho_1 & \lim\limits_{\alpha\to\infty} C^2_{(n,1)} = 0 \\ \lim\limits_{\alpha\to\infty} C^1_{(n-1,0)} = 0 & \lim\limits_{\alpha\to\infty} C^2_{(n-1,0)} = 1 & \lim\limits_{\alpha\to\infty} C^1_{(n-1,1)} = 0 & \lim\limits_{\alpha\to\infty} C^2_{(n-1,1)} = \rho_1. \end{array} \tag{2.69}$$

Equations 2.64 and 2.65 become the next step:

$$\begin{aligned}\lim_{\alpha\to\infty} C^r_{(n,j+1)} = {} & \frac{(\rho_1 + j)}{(j+1)} \lim_{\alpha\to\infty} C^r_{(n,j)} \\ & - \frac{\rho_1}{(j+1)} \lim_{\alpha\to\infty} C^r_{(n,j-1)} - 0,\end{aligned} \tag{2.70}$$

and

$$\begin{aligned}\lim_{\alpha\to\infty} C^r_{(n-1,j+1)} = {} & \frac{(\rho_1 + j)}{(j+1)} \lim_{\alpha\to\infty} C^r_{(n-1,j)} \\ & - \frac{\rho_1}{(j+1)} \lim_{\alpha\to\infty} C^r_{(n-1,j-1)} - 0.\end{aligned} \tag{2.71}$$

Next, let us use mathematical induction to identify and prove patterns of the coefficients as α approaches ∞.

Based on the initial values listed in Equation 2.69, we are able to calculate the first few coefficients through Equation 2.70 and 2.71:

- When $j = 1$, we have

$$\lim_{\alpha\to\infty} C^1_{(n,2)} = \frac{\rho_1 + 1}{2}\rho_1 - \frac{\rho_1}{1+1} \cdot 1 = \frac{\rho_1^2}{2}, \tag{2.72}$$

$$\lim_{\alpha\to\infty} C^2_{(n,2)} = 0, \tag{2.73}$$

$$\lim_{\alpha\to\infty} C^1_{(n-1,2)} = 0, \tag{2.74}$$

$$\lim_{\alpha\to\infty} C^2_{(n-1,2)} = \frac{\rho_1^2}{2}. \tag{2.75}$$

- When $j = 2$, we have

$$\lim_{\alpha\to\infty} C^1_{(n,3)} = \frac{\rho_1 + 1}{3} \cdot \frac{1}{2}\rho_1^2 - \frac{\rho_1}{3} \cdot \rho_1 = \frac{\rho_1^3}{3!}, \tag{2.76}$$

$$\lim_{\alpha\to\infty} C^2_{(n,3)} = 0, \tag{2.77}$$

$$\lim_{\alpha\to\infty} C^1_{(n-1,3)} = 0, \tag{2.78}$$

$$\lim_{\alpha\to\infty} C^2_{(n-1,3)} = \frac{\rho_1^3}{3!}. \tag{2.79}$$

Since the first few values of $\lim_{\alpha\to\infty} C^1_{(n-1,j)}$ and $\lim_{\alpha\to\infty} C^2_{(n,j)}$ are all zeroes, it is certain that

$$\lim_{\alpha\to\infty} C^2_{(n,j+1)} = 0, \quad \text{for all } j = 0, 1, 2, 3, ..., n-1, \tag{2.80}$$

$$\lim_{\alpha\to\infty} C^1_{(n-1,j+1)} = 0, \quad \text{for all } j = 0, 1, 2, 3, ..., n-2, \tag{2.81}$$

Next, let us make the following conjectures about the coefficients $C^1_{(n,j+1)}$ and $C^2_{(n-1,j+1)}$:

$$\lim_{\alpha\to\infty} C^1_{(n,j+1)} = \frac{\rho_1^{j+1}}{(j+1)!}, \quad j = 0, 1, 2, ..., n-1, \text{ and} \tag{2.82}$$

$$\lim_{\alpha\to\infty} C^2_{(n-1,j+1)} = \frac{\rho_1^{j+1}}{(j+1)!}, \quad j = 0, 1, 2, 3, ..., n-2, \tag{2.83}$$

and prove them with mathematical induction.

Proof. Assume that

$$\lim_{\alpha\to\infty} C^1_{(n,j)} = \frac{\rho_1^{j}}{(j)!} \tag{2.84}$$

and

$$\lim_{\alpha\to\infty} C^1_{(n,j-1)} = \frac{\rho_1^{j-1}}{(j-1)!} \tag{2.85}$$

hold for $j = 0, 1, 2, ..., n-1$. Then we have

$$\begin{aligned}\lim_{\alpha\to\infty} C^1_{(n,j+1)} &= \frac{\rho_1 + j}{j+1} \cdot \frac{\rho_1^{j}}{(j)!} - \frac{\rho_1}{j+1} \cdot \frac{\rho_1^{j-1}}{(j-1)!} \\ &= \frac{\rho_1^{j+1}}{(j+1)!},\end{aligned} \tag{2.86}$$

which proves Equation 2.82. Similarly, Equation 2.83 can also be proven without difficulty.

■

Then from Equation 2.66, the boundary points are obtained as α approaches infinity:

$$\begin{aligned}\lim_{\alpha\to\infty} P_{(n,0)} &= \frac{-(\rho_2 + n\alpha)\lim_{\alpha\to\infty} C^2_{(n,n)} + \alpha\rho_1 \lim_{\alpha\to\infty} C^2_{(n,n-1)}}{(\rho_2 + n\alpha)\lim_{\alpha\to\infty} C^1_{(n,n)} - \alpha\rho_1 \lim_{\alpha\to\infty} C^1_{(n,n-1)}} \lim_{\alpha\to\infty} P_{(n-1,0)} \\ &= \frac{-(\rho_2 + n\alpha)\cdot 0 + \alpha\rho_1 \cdot 0}{(\rho_2 + n\alpha)\frac{\rho_1^n}{n!} - \alpha\rho_1 \frac{\rho_1^{n-1}}{(n-1)!}} \lim_{\alpha\to\infty} P_{(n-1,0)}\end{aligned}$$

$$= 0. \tag{2.87}$$

Then $\lim_{\alpha\to\infty} P_{(n-1,0)}$ can be directly calculated as follows:

$$\sum_{(i,j)\in\Omega_{g=1}} \lim_{\alpha\to\infty} P_{(i,j)} = 1 \tag{2.88}$$

$$\Rightarrow \sum_{(i,j)\in\Omega_{g=1}} \lim_{\alpha\to\infty} \left(C^1_{(i,j)} P_{(n,0)} + C^2_{(i,j)} P_{(n-1,0)} \right) = 1$$

$$\Rightarrow \left(\lim_{\alpha\to\infty} P_{(n-1,0)} \right) \sum_{j=0}^{n-1} \lim_{\alpha\to\infty} C^2_{(n-1,j)} = 1$$

(because $\lim\limits_{\alpha\to\infty} P_{(n,0)}$ and all $\lim\limits_{\alpha\to\infty} C^1_{(n-1,j)}$'s are zeroes)

$$\Rightarrow \lim_{\alpha\to\infty} P_{(n-1,0)} = \left(\sum_{j=0}^{n-1} \left[\left(\frac{\lambda_1}{\mu_1} \right)^j \Big/ j! \right] \right)^{-1}. \tag{2.89}$$

Since $\lim_{\alpha\to\infty} P_{(n,0)}$ and all $\lim_{\alpha\to\infty} C^2_{(n,g)}$ are zeroes, we have all $\lim_{\alpha\to\infty} P_{(n,j)}$'s are also zeroes. All the remaining non zero steady state probabilities as α approaches infinity are

$$\lim_{\alpha\to\infty} P_{(n-1,j)} = \frac{\rho_1^j}{j!} \cdot \left(\sum_{j=0}^{n-1} \left[\left(\frac{\lambda_1}{\mu_1} \right)^j \Big/ j! \right] \right)^{-1}, \ j = 1, 2, ...n-1, \tag{2.90}$$

which indicates that the system is reduced to an $M/M/n{-}1/n{-}1$ loss system as α approaches infinity. This conclusion is expected, since when α approaches infinity the mean service time for handoff calls becomes infinitely long by comparison to that of new calls. As a result, the probability of the system having one ongoing handoff call is approaching 1, and it is almost certain that the capacity for new calls is $n-1$.

Remark 1 *We have also noticed another pattern of the coefficients, that is:*

$$C^r_{(n,j)} + C^r_{(n-1,j)} = \frac{\rho_1^j}{j!}, \ r = 1, 2 \ \textit{and} \ j = 0, 1, 2, ..., n-1. \tag{2.91}$$

The proof can be found in the Appendix.

In **case III**, we treat $\rho_1 = 0$ and (for simplicity) set $\alpha = 1$. The initial values and the recursive relations for the coefficients are re-established, and patterns of coefficients are found to facilitate in solving for the steady state probabilities.

By setting $\alpha = 1$ and $\rho_1 = 0$ in (2.63) we obtain the initial values for this case:

$$\begin{array}{llll} C^1_{(n,0)} = 1 & C^2_{(n,0)} = 0 & C^1_{(n,1)} = \rho_2 & C^2_{(n,1)} = -1 \\ C^1_{(n-1,0)} = 0 & C^2_{(n-1,0)} = 1 & C^1_{(n-1,1)} = -\rho_2 & C^2_{(n-1,1)} = 1. \end{array} \tag{2.92}$$

The recursive relations for coefficients can also be established by setting $\alpha = 1$ and $\rho_1 = 0$ in Equation 2.64 and 2.65. For $j = 1, 2, ..., n-2$ We have

$$C^r_{(n,j+1)} = \frac{(\rho_2 + j)}{(j+1)} C^r_{(n,j)} - \frac{1}{j+1} C^r_{(n-1,j)} \tag{2.93}$$

and

$$C^r_{(n-1,j+1)} = C^r_{(n-1,j)} - \frac{\rho_2}{j+1} C^r_{(n,j)}. \tag{2.94}$$

Equation 2.66 - 2.68 can also be reduced to

$$P_{(n-1,0)} = hP_{(n,0)} \tag{2.95}$$

$$P_{(n,0)} = \frac{1}{\sum_{(i,j)\in\Omega_{g=1}} C^1_{(i,j)} + h \sum_{(i,j)\in\Omega_{g=1}} C^2_{(i,j)}} \tag{2.96}$$

where

$$h = \frac{(\rho_2 + n\alpha)C^1_{(n,n)} - \alpha\rho_1 C^1_{(n,n-1)}}{-(\rho_2 + n\alpha)C^2_{(n,n)} + \alpha\rho_1 C^2_{(n,n-1)}} = -\frac{C^1_{(n,n)}}{C^2_{(n,n)}}. \tag{2.97}$$

The patterns of coefficients that are useful in explicitly calculating all the coefficients are established as follows:

- based on the initial values listed in (2.92), the calculation is

$$\frac{C^1_{(n,1)}}{C^2_{(n,1)}} = -\rho_2 \quad \text{and} \quad \frac{C^1_{(n-1,1)}}{C^2_{(n-1,1)}} = -\rho_2$$
$$\Rightarrow C^1_{(i,1)} = -\rho_2 C^2_{(i,1)}, \quad i = n-1, \text{ or } n. \tag{2.98}$$

- When $j = 1$, by Equations 2.93 and 2.94 we have

$$C^1_{(n,2)} = \frac{\rho_2 + 1}{2}\rho_2 - \frac{1}{2}(-\rho_2) = \frac{1}{2}\rho_2^2 + \rho_2 \tag{2.99}$$

$$C^2_{(n,2)} = \frac{\rho_2 + 1}{2} \cdot (-1) - \frac{1}{2} \cdot (1) = -\frac{1}{2}\rho_2 - 1$$

$$\Rightarrow \frac{C^1_{(n,2)}}{C^2_{(n,2)}} = -\rho_2 \tag{2.100}$$

and

$$C^1_{(n-1,2)} = C^1_{(n-1,1)} - \frac{\rho_2}{2}C^1_{(n,1)} = -\rho_2 - \frac{\rho_2^2}{2} \cdot \rho_2 \tag{2.101}$$

$$C^2_{(n-1,2)} = C^2_{(n-1,1)} - \frac{\rho_2}{2}C^2_{(n,1)} = 1 - \frac{\rho_2}{2} \cdot (-1)$$

$$\Rightarrow \frac{C^1_{(n-1,2)}}{C^2_{(n-1,2)}} = -\rho_2. \tag{2.102}$$

Again, there follows

$$C^1_{(i,2)} = -\rho_2 C^2_{(i,2)}, \ i = n-1, n. \tag{2.103}$$

- Assume that $C^1_{(i,j-1)} = -\rho_2 C^2_{(i,j-1)}$ for $i = n-1$ or n, and $j = 1, 2, 3, ..., n$, then

$$\begin{aligned} C^1_{(n,j)} &= \frac{\rho_2 + j}{j+1}C^1_{(n,j-1)} - \frac{1}{j+1}C^1_{(n-1,j-1)} \\ &= \frac{\rho_2 + j}{j+1}\left(-\rho_2 C^2_{(n,j-1)}\right) - \frac{1}{j+1}\left(-\rho_2 C^2_{(n-1,j-1)}\right) \\ &= -\rho_2\left(\frac{\rho_2 + j}{j+1}C^2_{(n,j-1)} - \frac{1}{j+1}C^2_{(n-1,j-1)}\right) \\ &\overset{(2.93)}{=} -\rho_2 C^2_{(n,j)}, \quad j = 2, 3, ..., n \end{aligned} \tag{2.104}$$

and

$$\begin{aligned} C^1_{(n-1,j)} &= C^1_{(n-1,j-1)} - \frac{\rho_2}{j+1}C^1_{(n,j-1)} \\ &= -\rho_2 C^2_{(n-1,j-1)} - \frac{\rho_2}{j+1}(-\rho_2 C^2_{(n,j-1)}) \\ &= -\rho_2\left(C^2_{(n-1,j-1)} - \frac{\rho_2}{j+1}C^2_{(n,j-1)}\right) \\ &\overset{(2.94)}{=} -\rho_2 C^2_{(n-1,j)}, \quad j = 1, 2, 3, ..., n-1. \end{aligned} \tag{2.105}$$

Therefore, we have

$$C^1_{(n,j)} = -\rho_2 C^2_{(n,j)}, \quad \text{for all } j = 1, 2, ..., n, \tag{2.106}$$

and

$$C^1_{(n-1,j)} = -\rho_2 C^2_{(n-1,j)}, \quad \text{for all } j = 1, 2, 3, ..., n-1. \tag{2.107}$$

To summarize, the following relationship between $C^1_{(i,j)}$ and $C^2_{(i,j)}$ has been observed and proven:

$$C^1_{(i,j)} = -\rho_2 C^2_{(i,j)}, \quad i = n-1 \text{ or } n, \text{ and } j = 1, 2, ..., i. \tag{2.108}$$

Now, using Equation 2.108 together with the fact that

$$h = -\frac{C^1_{(n,n)}}{C^2_{(n,n)}} = \rho_2, \tag{2.109}$$

Equation 2.95 can be simplified and the boundary points $P_{(n-1,0)}$ and $P_{(n,0)}$ can be obtained as

$$\begin{aligned} P_{(n,0)} &= \frac{1}{\sum_{(i,j)\in\Omega_{g=1}} C^1_{(i,j)} + h\sum_{(i,j)\in\Omega_{g=1}} C^2_{(i,j)}} && (2.110)\\ &= \frac{1}{C^1_{(n,0)} + C^1_{(n-1,0)} + \sum_{j\neq 0} C^1_{(i,j)} + h(C^2_{(n,0)} + C^2_{(n-1,0)}) + h\sum_{j\neq 0} C^2_{(i,j)}}\\ &= \frac{1}{1 + 0 + \sum_{j\neq 0} C^1_{(i,j)} + \rho_2(0+1) + \rho_2\sum_{j\neq 0} C^2_{(i,j)}}\\ &= \frac{1}{1 + \sum_{j\neq 0} C^1_{(i,j)} + \rho_2 - \sum_{j\neq 0}(-\rho_2 C^2_{(i,j)})}\\ &= \frac{1}{1 + \sum_{j\neq 0} C^1_{(i,j)} + \rho_2 - \sum_{j\neq 0} C^1_{(i,j)}}\\ &= \frac{1}{1+\rho_2}, && (2.111) \end{aligned}$$

and

$$P_{(n-1,0)} = hP_{(n,0)} = \frac{\rho_2}{1+\rho_2}. \tag{2.112}$$

Therefore, the remaining steady state probabilities are

$$\begin{aligned}
P_{(i,j)} &= C^1_{(i,j)}P_{(n,0)} + C^2_{(i,j)}P_{(n-1,0)} \\
&= C^1_{(i,j)}P_{(n,0)} + C^2_{(i,j)}\rho_2 P_{(n,0)} \\
&= P_{(n,0)}(C^1_{(i,j)} + \rho_2 C^2_{(i,j)}) \\
&= 0, \quad \text{for } j = 1, 2, ..., i. \qquad (2.113)
\end{aligned}$$

Therefore, only the two boundary points $P_{(n,0)}$ and $P_{(n-1,0)}$ are non zero. By further examining their values, one discover that they match the limiting distribution of an $M/M/1/1$ loss system. This result can be deduced readily from the model settings: $\rho_1 = 0$ implies $\lambda_1 = 0$, which means that the incoming stream of new calls is shut down, and only handoff calls can access the system. As a result, the system reduces to an $M/M/1/1$ loss system serving only handoff calls.

In the last case, **case IV**, keep $\alpha = 1$ but set $\rho_2 = 0$; this will reduce the system to an $M/M/n/n$ loss system because in this case, the incoming stream of handoff is shut down; only new calls can access the system. The settings in **case IV** describe the same situation as in **case I** by a different parametrization; and it is expected to produce the same results. This case can also be verified by examining the steady state probabilities with the recursive method.

By setting $\alpha = 1$ and $\rho_2 = 0$ in Equation 2.63, we obtain the initial values of the coefficients for **case IV**:

$$\begin{array}{llll}
C^1_{(n,0)} = 1 & C^2_{(n,0)} = 0 & C^1_{(n,1)} = \rho_1 & C^2_{(n,1)} = -1 \\
C^1_{(n-1,0)} = 0 & C^2_{(n-1,0)} = 1 & C^1_{(n-1,1)} = 0 & C^2_{(n-1,1)} = \rho_1 + 1.
\end{array} \qquad (2.114)$$

The recursive relations for coefficients can also be established by setting $\alpha = 1$ and $\rho_2 = 0$ in Equations 2.64 and 2.65. This setting produces

$$\begin{aligned}
C^r_{(n,j+1)} &= \frac{\rho_1 + j}{j+1} C^r_{(n,j)} \\
&- \frac{\rho_1}{j+1} C^r_{(n,j-1)} - \frac{1}{j+1} C^r_{(n-1,j)}
\end{aligned} \qquad (2.115)$$

and

$$
\begin{aligned}
C^r_{(n-1,j+1)} = \frac{\rho_1 + 1 + j}{j+1} C^r_{(n-1,j)} \\
- \frac{\rho_1}{j+1} C^r_{(n-1,j-1)}.
\end{aligned}
\tag{2.116}
$$

Again, we need to find useful patterns among coefficients in order to explicitly calculate all the coefficients:

- When $r = 1$:
  1. When the capacity for new calls is $n-1$, since $C^1_{(n-1,0)} = C^1_{(n-1,1)} = 0$, from (2.116) it is easy to see that

$$
C^1_{(n-1,j)} = 0, \quad \text{for all } j = 1, 2, ..., n-1. \tag{2.117}
$$

  2. When the capacity for new calls is n, recall from initial values listed in (2.114) and Equation 2.116 that

$$
\begin{aligned}
C^1_{(n,0)} &= 1 \\
C^1_{(n,1)} &= \rho_1 \\
C^1_{(n,2)} &= \frac{\rho_1^2}{2}.
\end{aligned}
$$

  By mathematical induction, we can prove that for $j = 0, 1, 2, ..., n$,

$$
C^1_{(n,j)} = \frac{\rho_1^j}{j!}. \tag{2.118}
$$

- When $r = 2$:
  1. When the capacity for new call is $n-1$, by initial values listed in (2.114) together with Equation 2.116, we have:

$$
\begin{aligned}
C^2_{(n-1,0)} &= 1 \\
C^2_{(n-1,1)} &= \rho_1 + 1 \\
C^2_{(n-1,2)} &= \frac{\rho_1 + 1 + 1}{2} C^2_{(n-1,1)} - \frac{\rho_1}{2} C^2_{(n-1,0)}
\end{aligned}
$$

$$= \frac{1}{2}\rho_1^2 + \rho_1 + 1,$$

and again by mathematical induction, we can prove that for $j = 0, 1, 2, ..., n-1$,

$$C^2_{(n-1,j)} = \sum_{k=0}^{j} \frac{\rho_1^k}{k!}. \tag{2.119}$$

2. When the capacity for new calls is n, using the equality in Equation 2.91, we have for $j = 0, 1, 2, ..., n-1$ that

$$C^2_{(n,j)} = \frac{\rho_1^j}{j!} - C^2_{(n-1,j)} = -\sum_{k=0}^{j-1} \frac{\rho_1^k}{k!}. \tag{2.120}$$

Now, the boundary points $P_{(n-1,0)}$ and $P_{(n,0)}$ are ready to be derived. Because

$$h = \frac{(\rho_2 + n\alpha)C^1_{(n,n)} - \alpha\rho_1 C^1_{(n,n-1)}}{-(\rho_2 + n\alpha)C^2_{(n,n)} + \alpha\rho_1 C^2_{(n,n-1)}} \quad \text{by (2.44)} \tag{2.121}$$

$$= \frac{nC^1_{(n,n)} - \rho_1 C^1_{(n,n-1)}}{-nC^2_{(n,n)} + \rho_1 C^2_{(n,n-1)}} \quad \text{(because } \rho_2 = 0, \alpha = 1)$$

$$= \frac{n\frac{\rho_1^n}{n!} - \rho_1 \frac{\rho_1^{n-1}}{(n-1)!}}{-n\left(-\sum_{k=0}^{n-1} \frac{\rho_1^k}{k!}\right) + \rho_1\left(-\sum_{k=0}^{n-2} \frac{\rho_1^k}{k!}\right)}$$

$$= 0, \tag{2.122}$$

we have

$$P_{(n-1,0)} = hP_{(n,0)} \quad \text{by (2.45)}$$

$$= 0, \tag{2.123}$$

and

$$P_{(n,0)} = \frac{1}{\sum_{(i,j)\in\Omega_{g=1}} C^1_{(i,j)} + h \cdot \sum_{(i,j)\in\Omega_{g=1}} C^2_{(i,j)}} \quad \text{by (2.43)}$$

$$= \frac{1}{\sum_{(i,j)\in\Omega_{g=1}} C^1_{(i,j)}}$$
$$= \frac{1}{\sum_{j=0}^{n} \frac{\rho_1^j}{j!}}.$$

Because $P_{(n,0)}$ only depends on n, all the js in the above expression are replaced by k to avoid confusion. Then $P_{(n,0)}$ can be written as:

$$P_{(n,0)} = \frac{1}{\sum_{k=0}^{n} \frac{\rho_1^k}{k!}}. \tag{2.124}$$

The remaining steady state probabilities are as follows:

$$\begin{aligned} P_{(n,j)} &= C^1_{(n,j)}P_{(n,0)} + C^2_{(n,j)}P_{(n-1,0)} \\ &= \frac{\frac{\rho_1^j}{j!}}{\sum_{k=0}^{n} \frac{\rho_1^k}{k!}}, \quad \text{for } j = 1,2,...,n \qquad (2.125) \\ P_{(n-1,j)} &= C^1_{(n-1,j)}P_{(n,0)} + C^2_{(n-1,j)}P_{(n-1,0)} \\ &= 0, \quad \text{for } j = 1,2,...,n-1, \qquad (2.126) \end{aligned}$$

which indicates that the system in case IV is indeed an $M/M/n/n$ loss system that serves only new calls.

2.4.3 When $g > 1$

This section extends the recursive solution from $g = 1$ to the more general case $g > 1$. To facilitate the transition from $g = 1$ to $g > 1$, let us divide all the state probabilities into 5 groups as shown in Figure 2.5. Then, the balance equations can be established below.

- Boundary points: $P_{(i,0)}$ where $i = n-g, n-g+1, ...n$,
  - When $i = n$:

$$(\lambda_1 + \lambda_2)P_{(n,0)} = \mu_1 P_{(n,1)} + \mu_2 P_{(n-1,0)}. \tag{2.127}$$

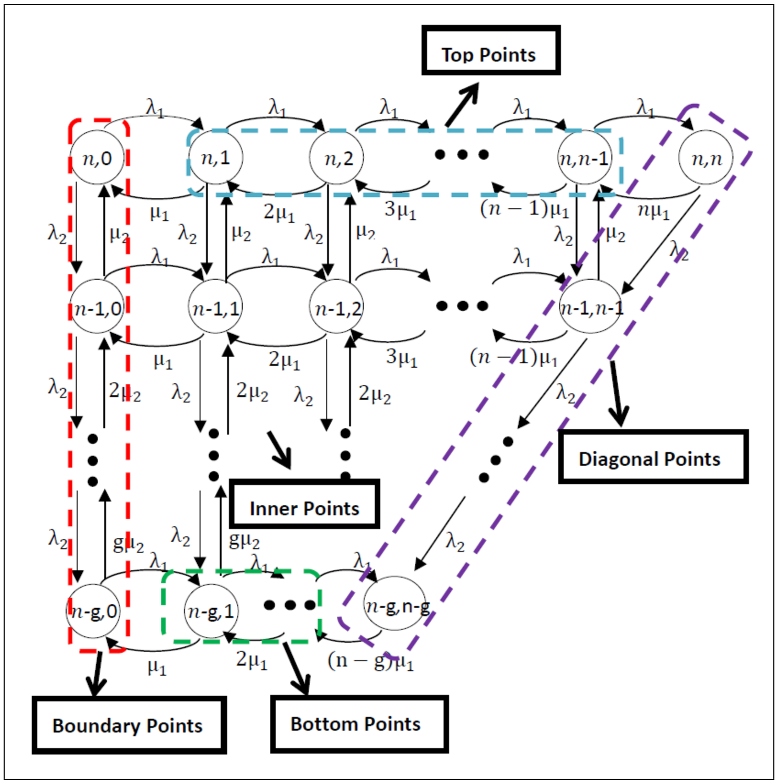


Figure 2.5: Grouping of states of the M1 model when $g > 1$.

– When $i = n - g$:

$$(\lambda_1 + g\mu_2)P_{(n-g,0)} = \mu_1 P_{(n-g,1)} + \lambda._2 P_{(n-g+1,0)} \tag{2.128}$$

– When $n - g < i < n$:

$$(\lambda_1 + \lambda_2 + (n-i)\mu_2)P_{(i,0)} = \mu_1 P_{(i,1)} + \lambda_2 P_{(i+1,0)} + (n-i+1)\mu_2 P_{(i-1,0)}. \tag{2.129}$$

- Top points: $P_{(n,j)}$ where $j = 1, 2, ..., n-1$,

$$(\lambda_1 + \lambda_2 + j\mu_1)P_{(n,j)} = \lambda_1 P_{(n,j-1)} + \mu_2 P_{(n-1,j)} + (j+1)\mu_1 P_{(n,j+1)}. \tag{2.130}$$

- Bottom points: $P_{(n-g,j)}$ where $j = 1, 2, ...n-g-1$,

$$(g\mu_2 + \lambda_1 + j\mu_1)P_{(n-g,j)} = \lambda_1 P_{(n-g,j-1)} + \lambda_2 P_{(n-g+1,j)} + (j+1)\mu_1 P_{(n-g,j+1)}. \tag{2.131}$$

- Diagonal points: $P_{(i,j)}$ where $i = j$ and $i = n-g, n-g+1, ..., n$,
  - When $i = j = n$:
$$(n\mu_1 + \lambda_2)P_{(n,n)} = \lambda_1 P_{(n,n-1)}. \tag{2.132}$$
  - When $i = j = n-g$:
$$(g\mu_2 + (n-g)\mu_1)P_{(n-g,n-g)} = \lambda_1 P_{(n-g,n-g-1)} + \lambda_2(P_{(n-g+1,n-g)} + P_{(n-g+1,n-g+1)}). \tag{2.133}$$
  - When $n-g < i = j < n$:
$$(\lambda_2 + i\mu_1 + (n-i)\mu_2)P_{(i,i)} = \lambda_1 P_{(i,i-1)} + \lambda_2(P_{(i+1,i)} + P_{(i+1,i+1)}). \tag{2.134}$$

- Inner points: $P_{(i,j)}$ where $i = n-g+1, n-g+2, .., n-1$ and $0 < j < i$,
$$(\lambda_1+\lambda_2+j\mu_1+(n-i)\mu_2)P_{(i,j)} = \lambda_1 P_{(i,j-1)}+\lambda_2 P_{(i+1,j)}+(j+1)\mu_1 P_{(i,j+1)}+(n-i+1)\mu_2 P_{(i-1,j)}. \tag{2.135}$$

By defining $P_{(i,j)} = 0$ whenever (i,j) is not a valid state (i.e., $(i,j) \notin \Omega$), one finds that some of the above balance equations can be combined together and as a result, the 5 groups of steady state probabilities can also be combined accordingly into 3 sets:

- Set A: Boundary points (with $i = n-g+1, ..., n$) + top points + inner points. The following balance equation is obtained by combining (2.127), (2.129), (2.130) and (2.135) together.
$$(\lambda_1+\lambda_2+j\mu_1+(n-i)\mu_2)P_{(i,j)} = \lambda_1 P_{(i,j-1)}+\lambda_2 P_{(i+1,j)}+(j+1)\mu_1 P_{(i,j+1)}+(n-i+1)\mu_2 P_{(i-1,j)} \tag{2.136}$$
  for $j = 0, 1, 2, ..., i-1$.

- Set B: Boundary points (with $i = n-g$) + bottom points. The following balance equation is obtained by combining (2.128) and (2.131) together.
$$(\lambda_1 + g\mu_2 + j\mu_1)P_{(n-g,j)} = \lambda_1 P_{(n-g,j-1)} + \lambda_2 P_{(n-g+1,j)} + (j+1)\mu_1 P_{(n-g,j+1)} \tag{2.137}$$

for $j = 0, 1, 2, ..., n - g - 1$.

- Set C: all the diagonal points.
  - When $i = n - g + 1, n - g + 2, ..., n$ (by combining (2.132) and (2.134) together):

$$(\lambda_2 + i\mu_1 + (n - i)\mu_2)P_{(i,i)} = \lambda_1 P_{(i,i-1)} + \lambda_2(P_{(i+1,i)} + P_{(i+1,i+1)}). \tag{2.138}$$

  - When $i = n - g$, refer to Equation 2.133.

Since the steady state probabilities of states $(n, 0), (n-1, 0), ...$ and $(n-g, 0)$ are serving as boundary points, by defining $C^r_{(i,j)}$ as the coefficient of the steady state probability of the boundary point $(r, 0)$, that is, $P_{(r,0)}$, all the remaining steady state probabilities could be written in terms of the boundary points as:

$$P_{(i,j)} = C^n_{(i,j)} P_{(n,0)} + C^{n-1}_{(i,j)} P_{(n-1,0)} + ... + C^{n-g}_{(i,j)} P_{(n-g,0)}, \tag{2.139}$$

and all the coefficients $C^r_{(i,j)}$, where $r = n - g, n - g + 1, ..., n$ and $(i, j) \in \Omega$, still need to be determined. The recursive formulae for $P_{(i,j)}$ can be established first using Equation 2.136 and 2.137:

$$\begin{aligned} P_{(i,j+1)} = {} & \frac{(\lambda_1 + \lambda_2 + j\mu_1 + (n - i)\mu_2)}{(j + 1)\mu_1} P_{(i,j)} - \frac{\lambda_1}{(j + 1)\mu_1} P_{(i,j-1)} - \frac{\lambda_2}{(j + 1)\mu_1} P_{(i+1,j)} \\ & - \frac{(n - i + 1)\mu_2}{(j + 1)\mu_1} P_{(i-1,j)} \end{aligned} \tag{2.140}$$

for $i = n - g + 1, n - g + 2, ..., n$ and $j = 0, 1, 2, ..., i - 1$, and

$$P_{(n-g,j+1)} = \frac{(\lambda_1 + g\mu_2 + j\mu_1)}{(j + 1)\mu_1} P_{(n-g,j)} - \frac{\lambda_1}{(j + 1)\mu_1} P_{(n-g,j-1)} - \frac{\lambda_2}{(j + 1)\mu_1} P_{(n-g+1,j)} \tag{2.141}$$

for $j = 0, 1, 2, ..., n - g - 1$.

The recursive formulae for the coefficients can then be obtained:

- For $i = n - g + 1, n - g + 2, ..., n$ and $j = 0, 1, 2, ..., i - 1$, by Equation 2.140, we have

$$\begin{aligned} C^r_{(i,j+1)} = & \frac{(\lambda_1 + \lambda_2 + j\mu_1 + (n-i)\mu_2)}{(j+1)\mu_1} C^r_{(i,j)} - \frac{\lambda_1}{(j+1)\mu_1} C^r_{(i,j-1)} - \frac{\lambda_2}{(j+1)\mu_1} C^r_{(i+1,j)} \\ & - \frac{(n-i+1)\mu_2}{(j+1)\mu_1} C^r_{(i-1,j)}. \end{aligned} \tag{2.142}$$

- For $i = n - g$ and $j = 0, 1, 2, ..., n - g - 1$, by Equation 2.141, we have

$$C^r_{(n-g,j+1)} = \frac{(\lambda_1 + g\mu_2 + j\mu_1)}{(j+1)\mu_1} C^r_{(n-g,j)} - \frac{\lambda_1}{(j+1)\mu_1} C^r_{(n-g,j-1)} - \frac{\lambda_2}{(j+1)\mu_1} C^r_{(n-g+1,j)}. \tag{2.143}$$

- In order to calculate all the coefficients, start with the initial values where $j = 0$ or 1:
  - When $j = 0$:

$$C^r_{(i,0)} = \begin{cases} 1 & \text{if } r = i \\ 0 & \text{Otherwise} \end{cases}. \tag{2.144}$$

  - When $j = 1$: Set $j = 0$ in Equation 2.140 to bring about:

$$P_{(i,1)} = \frac{(\lambda_1 + \lambda_2 + (n-i)\mu_2)}{\mu_1} P_{(i,0)} - \frac{\lambda_2}{\mu_1} P_{(i+1,0)} - \frac{(n-i+1)\mu_2}{\mu_1} P_{(i-1,0)} \tag{2.145}$$

for $i = n - g + 1, n - g + 2, ..., n$.

Then, the corresponding initial values are as shown below:

$$C^r_{(i,1)} = \begin{cases} \frac{\lambda_1+\lambda_2+(n-i)\mu_2}{\mu_1} & \text{when } r = i \\ -\frac{\lambda_2}{\mu_1} & \text{when } r = i+1 \\ -\frac{(n-i+1)\mu_2}{\mu_1} & \text{when } r = i-1 \\ 0 & \text{otherwise} \end{cases}$$

for all $i = n-g+1, n-g+2, ..., n$.

Next, we set $j = 0$ in Equation 2.141 to obtain the initial values of the coefficients for $P_{(n-g,1)}$:

$$P_{(n-g,1)} = \frac{(\lambda_1 + g\mu_2)}{\mu_1} P_{(n-g,0)} - \frac{\lambda_2}{\mu_1} P_{(n-g+1,0)} \tag{2.146}$$

and the corresponding initial values of coefficients are produced:

$$C^r_{(i,1)} = \begin{cases} \frac{\lambda_1+g\mu_2}{\mu_1} & \text{when } r = n-g \\ -\frac{\lambda_2}{\mu_1} & \text{when } r = n-g+1 \\ 0 & \text{otherwise} \end{cases} .$$

The remaining coefficients can be calculated recursively using Equation 2.142 and 2.143 together with the initial values just obtained. Finally, after calculating all the coefficients, we need to solve a system of $g+1$ equations for the $g+1$ boundary points. The first g equations can be obtained from Equation 2.138:

$$(\lambda_2 + i\mu_1 + (n-i)\mu_2)P_{(i,i)} = \lambda_1 P_{(i,i-1)} + \lambda_2(P_{(i+1,i)} + P_{(i+1,i+1)}) \tag{2.147}$$

$$\Longrightarrow (\lambda_2 + i\mu_1 + (n-i)\mu_2) \sum_{r=n-g}^{n} P_{(r,0)} C^r_{(i,i)} = \lambda_1 \sum_{r=n-g}^{n} P_{(r,0)} C^r_{(i,i-1)}$$

$$+ \lambda_2(\sum_{r=n-g}^{n} P_{(r,0)} C^r_{(i+1,i)} + \sum_{r=n-g}^{n} P_{(r,0)} C^r_{(i+1,i+1)})$$

$$\Longrightarrow \sum_{r=n-g}^{n} \left(P_{(r,0)} \left[(\lambda_2 + i\mu_1 + (n-i)\mu_2) \cdot C^r_{(i,i)} - \lambda_1 C^r_{(i,i-1)} - \lambda_2 (C^r_{(i+1,i)} + C^r_{(i+1,i+1)}) \right] \right) = 0 \tag{2.148}$$

where $i = n - g + 1, n - g + 2, ..., n$. The last equation is the normalizing condition:

$$\sum_{(i,j)\in\Omega} P_{(i,j)} = 1 \tag{2.149}$$

$$\Longrightarrow \sum_{(i,j)\in\Omega} \left(\sum_{r=n-g}^{n} P_{(r,0)} C^r_{(i,j)} \right) = 1 \tag{2.150}$$

$$\Longrightarrow \sum_{r=n-g}^{n} \left[P_{(r,0)} \left(\sum_{(i,j)\in\Omega} C^r_{(i,j)} \right) \right] = 1. \tag{2.151}$$

Numerical methods can serve to solve this system of equations for boundary points. Subsequently, all the remaining steady state probabilities can be computed with Equation 2.139. As illustrated, the advantage of this recursive method over the composite model method introduced earlier is that when using the recursive method, we need only to solve a system of $g + 1$ equations instead of solving a system of $(2n - g + 2)(g + 1)/2$ equations while using the composite model approach. Therefore, systems with large n that is intractable with the composite model are now solvable using the recursive method.

2.5 Numerical examples

In this chapter, four methods have been introduced to calculate the performance measures for the M1 model. In the following experiment, all the four methods—as well as call-level simulations—will be carried out to compute the blocking and dropping probabilities for new calls[5] in the M1 model so that the results will be compared. The parameters are chosen as

[5] The blocking probability for handoff calls in the M1 model can be computed exactly by the Erlang B formula and is not included for method comparison

follows: λ_1 varies from 1 to 40; $\lambda_2 = \lambda_1/2$, $\mu_1 = \mu_2 = 1$, $n = 20$ and $g = 10$. Figure 2.6 demonstrates that the two numerical methods (i.e., the composite model method and the recursive method) matched very well with the simulation results. The approximate methods would overestimate new call blocking and dropping probabilities; the discrepancy becomes more significant as λ_1 increases. The CPU time for each method running on an i5-2500K 3.30GHz CPU was recorded and listed in Table 2.2. It is clear that the two approximate methods took only negligible amount of CPU time. The composite model method and the recursive method were running with the Multiprecision computing toolbox in Matlab to greatly boost their accuracy, but as a trade-off, they took much more CPU time to run.

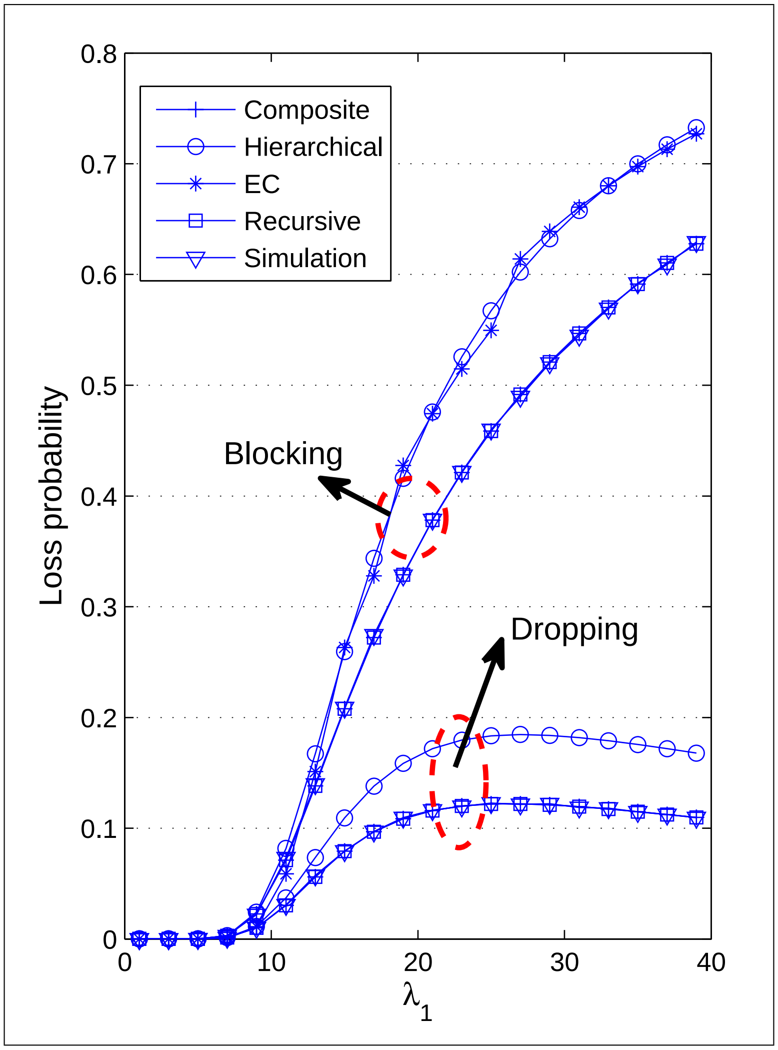


Figure 2.6: Comparison of 5 different methods for calculating the blocking or/and dropping probabilities for new calls in the M1 model. Please note that the EC method is not able to approximate dropping probabilities.

Table 2.2: CPU run time (in seconds) of different methods for solving the M1 model

Methods	Composite	Hierarchical	ENC	Recursive	Simulation
CPU time	1561.1	0.0442	0.0156	473.8842	613.3959

2.6 Optimization problems

To put the M1 model in practice, we would like to minimize its performance measures: handoff call blocking probability (P_b^h), new call blocking probability(P_b^N), and new call dropping probability (P_d^N). This is a multi-objective optimization problem [22], and the decision variables are the number of guard channels g and the number of total channels n. There are several different ways to set up the optimization problem, and we adopt the ways that were presented in Harine et al. [17] and consider the following two representative optimization problems.

2.6.1 Optimal number of guard channels

$\mathbf{O_1}$: Given λ_1, λ_2, μ_1, μ_2 and n, determine the optimal integer value of g so as to

$$\text{minimize } P_b^N \text{ and } P_d^N \text{ such that } P_b^h(g) \leq P_0^{hb},$$

where P_0^{hb} is a constraint imposed on the handoff call blocking probability P_b^h.

In order to solve this optimization problem, outlining the following properties of the performance metrics becomes a necessary step:

Properties

1. The handoff call blocking probability $P_b^h = EB(\rho_2, g)$, according to the property of the Erlang-B formula, is a decreasing function of g, i.e., $P_b^h(\rho_2, g) < P_b^h(\rho_2, g-1)$. Proof can be found in Harine et al. [17].
2. The new call blocking probability P_b^N is a decreasing function of n (when holding g fixed) and an increasing function of g (when holding n fixed). When g is fixed, a system with smaller n will provide fewer channels for new calls to use; hence, the higher the

new call blocking probability will be. When n is fixed, a system with more guard channels will allow more handoff calls to stay in the system at the same time. As a result, it will reduce the number of channels available for new calls and increase its blocking probability.

3. The new call dropping probability, P_d^N, is a decreasing function of n (when holding g fixed) and an increasing function of g (when holding n fixed). When g is fixed, a smaller n will increase the chance of new calls to use guard channels with the risk of being preempted by handoff calls later. Therefore, the dropping probability for new calls increases. When n is fixed, a larger g will also increase the chance of new calls occupying guard channels with the risk of being preempted by handoff calls later. Again, the dropping probability for new call increases.

The first property tells us that if g^* is the smallest value of g that satisfies $P_b^h(g^*) \leq P_0^{hb}$, then any g in $\{g^*+1, g^*+2, ..., n\}$ would also satisfy $P_b^h(g) \leq P_0^{hb}$. The second and the third properties suggest that when n is fixed, both P_d^N and P_b^N increase as g increases. Therefore, among all the possible values of g that satisfy $P_b^h(g) \leq P_0^{hb}$, the smallest one we defined earlier, g^*, will also minimize P_d^N and P_b^N at the same time. Thus, the optimal value of g can be obtained by using a simple one-dimensional search over the range $\{0, 1, 2, ..., n\}$ for g^* such that

$$g^* = \min\{g | P_b^h(g) \leq P_0^{hb}\}. \tag{2.152}$$

As an illustration, we set $\lambda_1 = \lambda_2 = 20$, $\mu_1 = \mu_2 = 1$ and $n = 60$ in the following examples, summarizing the optimal number of guard channels for different constraints of handoff call blocking probability in Table 2.3. As we can see, the optimal number of guard channels, g^*, increases as P_0^{hb} becomes stricter. When $P_0^{hb} = 10^{-6}$, $3/4$ of the channels are employed as guard channels. Also note that as g^* increases, both the blocking and dropping probabilities for new calls increase as well.

Table 2.3: Results of optimization problem $\mathbf{O_1}$ for the M1 model

P_0^{hb}	g^*	P_b^h	P_b^N	P_d^N
10^{-2}	30	0.0085	3.6643×10^{-4}	2.6134×10^{-4}
10^{-3}	35	6.8593×10^{-4}	6.1756×10^{-4}	5.5628×10^{-4}
10^{-4}	39	5.5554×10^{-5}	6.7247×10^{-4}	6.5805×10^{-4}
10^{-5}	42	6.4520×10^{-6}	6.7865×10^{-4}	6.7584×10^{-4}
10^{-6}	45	6.0625×10^{-7}	6.7940×10^{-4}	6.7904×10^{-4}

2.6.2 Optimal number of guard channels and total channels

$\mathbf{O_2}$: Given λ_1, λ_2, μ_1 and μ_2, determine the optimal integer values of n and g so as to

$$\text{minimize } n \text{ such that } \begin{cases} P_b^h(g) \le P_0^{hb} \\ \\ P_b^N(n,g) + P_d^N(n,g) \le P_0^{NL}. \end{cases}$$

In this optimization problem, we impose a constraint not only on the handoff call blocking probability but also on the new call loss probability (i.e., $P_L^N(n,g) = P_b^N(n,g) + P_d^N(n,g)$). To solve this optimization problem, first plot the contours of $P_b^h(g)$ and $P_L^N(n,g)$ in the first quadrant of the (n,g) plane in Figure 2.7. The region above the contour line $P_b^h(g) = P_0^{hb}$ and below line $n = g$ will satisfy the constraint $P_b^h(g) \le P_0^{hb}$. The region to the right of contour line $P_L^N = P_0^{NL}$ and below line $n = g$ will satisfy the other constraint $P_L^N(n,g) \le P_0^{NL}$. Therefore the feasible region for this optimization problem is the shaded region F, as shown in Figure 2.7, and the solution (n^*, g^*) is just the intersection point of the two contours. To locate this solution point (n^*, g^*) the first need is to find the smallest number of guard channels, g^*, that satisfies $P_b^h(g^*) \le P_0^{hb}$. Then, we fix $g = g^*$. From properties 2 and 3 we know that both $P_b^N(n,g)$ and $P_d^N(n,g)$ are decreasing functions of n when g is fixed, so the sum of these two functions is also a decreasing function of n when g is fixed. Consequently, the optimal number of channels is just the smallest n that satisfies $P_L^N(n,g^*) \le P_0^{NL}$. We have

$$g^* = \min\{g | P_b^h(g) \le P_0^{hb}\} \tag{2.153}$$

and then

$$n^* = \min\{n | n \ge g^*, P_L^N(n, g^*) \le P_0^{NL}\}. \tag{2.154}$$

Note that if the contour $P_L^N(n,g) = P_0^{NL}$ starts from line $g = 0$ and reaches line $n = g$ at some point (g', g') without intersecting with the other contour $P_b^h(g) = P_0^{hb}$, which could happen when the contour $P_b^h(g) = P_0^{hb}$ is above the point (g', g'), then the optimal solution is just (g^*, g^*).

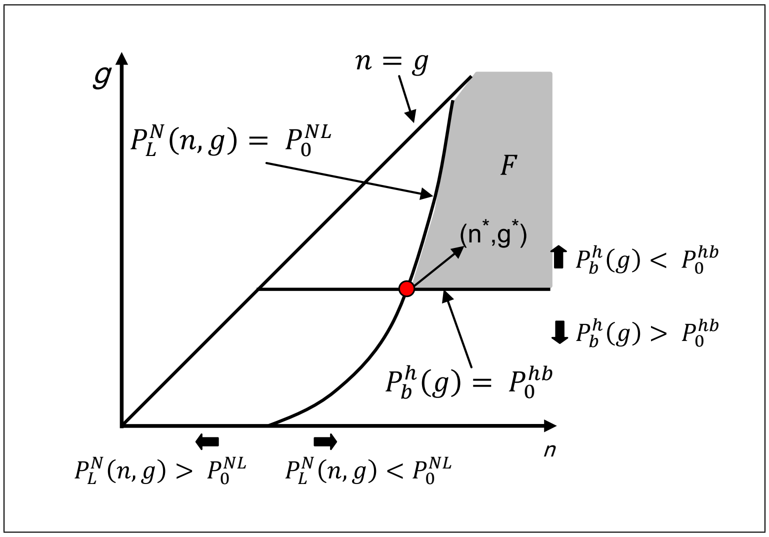


Figure 2.7: Optimization problem $\mathbf{O_2}$ for the M1 model

We provide some numerical examples of optimization problem $\mathbf{O_2}$ with the same traffic parameters as those chosen for illustrating optimization problem $\mathbf{O_1}$ (i.e., $\lambda_1 = \lambda_2 = 20$, $\mu_1 = \mu_2 = 1$). As shown in Table 2.4, both n^* and g^* increase as P_0^{hb} and P_0^{NL} become stricter and stricter. When these results are compared with the results of optimization problem $\mathbf{O_1}$ in Table 2.3, we note that g^*'s for the same P_0^{hb} are exactly the same: i.e., g^* is not affected by the presence of P_0^{NL} and depends only on P_0^{hb}.

It is also possible to set separate constraints for $P_b^N(n,g)$ and $P_d^N(n,g)$ and form the following optimization problem:

Table 2.4: Results of optimization problem $\mathbf{O_2}$ for the M1 model

P_0^{hb}	P_0^{NL}	n^*	g^*	P_b^h	P_L^N
10^{-3}	10^{-2}	55	35	6.8593×10^{-4}	0.0084
10^{-4}	10^{-3}	61	39	5.5554×10^{-5}	8.6622×10^{-4}
10^{-5}	10^{-4}	66	42	6.4520×10^{-6}	8.3133×10^{-5}
10^{-6}	10^{-5}	70	45	6.0625×10^{-7}	9.7555×10^{-6}

$\mathbf{O_3}$: Given λ_1, λ_2, μ_1 and μ_2, determine the optimal integer values of n and g so as to

$$\text{minimize } n \text{ such that } \begin{cases} P_b^h(n,g) \leq P_0^{hb} \\ P_b^N(n,g) \leq P_0^{Nb} \\ P_d^N(n,g) \leq P_0^{Nd} \end{cases} .$$

The situations are depicted in Figure 2.8. The region(s) to the right of contour $P_b^N(n,g) = P_0^{Nb}$ (or $P_d^N(n,g) = P_0^{Nd}$) and below line $n = g$ will satisfy $P_b^N(n,g) \leq P_0^{Nb}$ (or $P_d^N(n,g) \leq P_0^{Nd}$). Therefore, the feasible region of this optimization problem is the shaded region F in the figure. The procedure for finding the optimal solution is similar to the procedure developed for optimization problem $\mathbf{O_2}$. We first find the optimal number of guard channels g^* as defined in Equation 2.153. Then, starting at $n = g^*$, we search for n_b^* and n_d^* such that

$$n_b^* = \min\{n|n \geq g^*, P_b^N(n,g^*) \leq P_0^{Nb}\}. \tag{2.155}$$

and

$$n_d^* = \min\{n|n \geq g^*, P_d^N(n,g^*) \leq P_0^{Nd}\}. \tag{2.156}$$

The optimal number of channels n^* follows:

$$n^* = \max\{n_b^*, n_d^*\} \tag{2.157}$$

To provide some numerical examples as illustrations, we again take $\lambda_1 = \lambda_2 = 20$ and

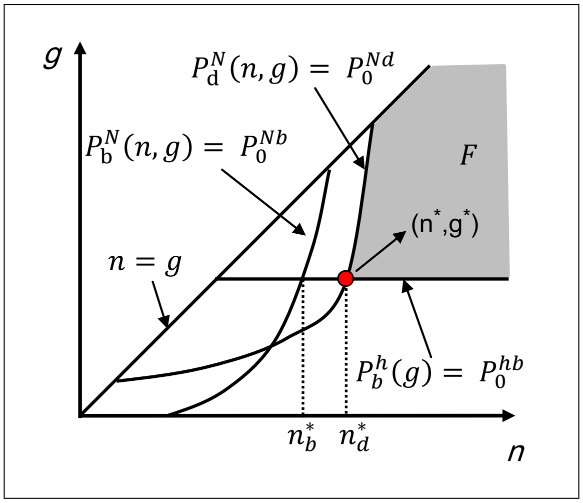


Figure 2.8: Optimization problem $\mathbf{O_3}$. Note that the relative position of contour $P_b^N(n,g) = P_0^{Nb}$ and $P_d^N(n,g) = P_0^{Nd}$ depends on the choices of thresholds P_0^{Nb} and P_0^{Nd}.

$\mu_1 = \mu_2 = 1$. As an easy way to break down the constraint on new call loss (P_0^{NL}) into blocking constraint (P_0^{Nb}) and dropping constraint (P_0^{Nd}), simply set $P_0^{Nb} = P_0^{Nd} = P_0^{NL}/2$. The results are summarized in Table 2.5. After comparing the results with those in Table 2.4, we discovered that, for the given traffic parameters (λ_1, λ_2, μ_1, and μ_2), the breakdown of the constraint on new call loss does not affect the optimal solutions n^* and g^*.

Table 2.5: Results of optimization problem $\mathbf{O_3}$ for the M1 model

P_0^{hb}	P_0^{Nb}	P_0^{Nd}	n^*	g^*	P_b^h	P_b^N	P_d^N
10^{-3}	0.5×10^{-2}	0.5×10^{-2}	55	35	0.00068593	0.00427811	0.00407770
10^{-4}	0.5×10^{-3}	0.5×10^{-3}	61	39	0.00005555	0.00043895	0.00042726
10^{-5}	0.5×10^{-4}	0.5×10^{-4}	66	42	0.00000645	0.00004199	0.00004114
10^{-6}	0.5×10^{-5}	0.5×10^{-5}	70	45	0.00000061	0.00000491	0.00000485

Chapter 3
Second Guard Channel Model

3.1 Motivation and model introduction

In Chapter 2, we developed the M1 model where the high-priority traffic (i.e., handoff calls) is only allowed to simultaneously occupy a pre-determined number of channels (i.e., guard channels). However high-priority traffic is guaranteed access to guard channels because it has priority over low-priority traffic (i.e., new calls).

In this chapter we introduce another guard channel model with controlled preemption where high-priority traffic is allowed to access guard channels and normal channels. In this model, high-priority and low-priority traffic first compete with each other for the normal channels according to the FCFS discipline. Incoming calls (both high-priority and low-priority) can access guard channels only after all the normal channels are occupied. High-priority traffic has priority over low-priority traffic and can preempt low-priority traffic only on these guard channels and only when the system is full. In 2001, Harine et al. [17] introduced a fixed guard channel model (hereafter referred to as HT's model). In HT's model, high-priority traffic can access all the channels (i.e., both guard channels and the normal channels) and the low-priority traffic is only allowed to access normal channels. This is a non-preemption model because high-priority traffic cannot preempt low-priority traffic under any circumstances. Our second guard channel model (hereafter referred to as the M2 model) is a modified version of HT's model: low-priority traffic can borrow idle guard channels when all normal channels are occupied, but with the risk of being dropped by incoming high-priority traffic when the system is full.

This chapter includes the same notations used in Chapter 2. New calls and handoff calls arrives according to independent Poisson processes with rates λ_1 and λ_2, respectively.

Service times for new calls and handoff calls are assumed to follow independent exponential distributions with rates μ_1 and μ_2, respectively. A total of n channels are in the system, and the number of guard channels is g. In HT's model, the system always tries to reserve g channels for handoff calls. Therefore, when the number of available channels (i.e., idle channels in the system) for handoff calls is less than or equal to g, no new calls are admitted. In the M2 model, new calls can be admitted into the system even when the number of "available channels"[1] for handoff calls is less than or equal to g. However, new calls that are admitted when the number of available channels for handoff calls is less than or equal to g are marked as preemptable calls and can later be preempted by handoff calls when necessary. Channels that are occupied by preemptable new calls are also available for handoff calls.

The call admission control process of the M2 model, therefore, can be summarized as follows: An incoming new call will be admitted if there is a free channel. After the number of available channels (including all the idle channels and the channels that are occupied by preemptable new calls) for handoff calls is less than or equal to g, new calls that are admitted will be preemptable. An incoming new call will be blocked and lost if all channels are busy. An incoming handoff call can access any free channel and when all channels are busy, it can preempt a preemptable new call. A handoff call will only be blocked if there are no available channels for it.

3.2 Analysis with homogeneous service rate

In this section we consider the case where both handoff calls and new calls have homogeneous service rates, (i.e., $\mu_1 = \mu_2 = \mu$), and we develop closed-form expressions for all three performance measures of interest. Because the performance measures for the case of heterogeneous service rates are generally intractable and not mentioned in Harine et al. [17], they are beyond the scope of this chapter.

[1] The available channels for handoff calls in the M2 model include not only idle channels, but also busy guard channels that are occupied with new calls.

3.2.1 Closed form performance metrics

Blocking probability of handoff calls

Handoff calls have priority over new calls only on guard channels. Although new calls can access idle guard channels, they get preempted by incoming handoff calls if all channels are busy. Therefore, although we have relaxed the call admission control protocol for new calls by allowing them to occupy idle guard channels, the blocking probability of handoff calls will not be affected and is same as was presented in HT's model:

$$P_b^h(n,g) = \frac{\frac{\rho^{n-g}}{n!}\rho_2^g}{\sum_{m=0}^{n-g-1}\frac{\rho^m}{m!} + \sum_{m=n-g}^{n}\frac{\rho^{n-g}}{m!}\rho_1^{m-(n-g)}}, \tag{3.1}$$

where $\rho = \lambda_1/\mu + \lambda_2/\mu$, which is the total offered load in the system, and $\rho_2 = \lambda_2/\mu$, which is the offered load for handoff calls.

Blocking probability and dropping probability of new calls

If we do not allow handoff calls to preempt new calls on guard channels, then according to the FCFS discipline, both handoff and new calls will have to compete equally for idle channels. The system thus becomes a fully shared multiserver pure loss system, supporting two types of traffic. Given that the total offered load is $\rho = \lambda_1/\mu + \lambda_2/\mu$, the common blocking probability for both traffic is given by the Erlang B formula $EB(\rho, n)$. Note that when a handoff call preempts a new call and takes over its channel, the total number of busy channels remains unchanged (i.e., remains equal to n). Moreover, the service time distribution of the channel taken over by the handoff call also remains unchanged due to the assumption of exponential service times with common rate μ [18]. Therefore, preemption does not affect the blocking probability of new calls and the blocking probability of new calls remains equal to $EB(\rho, n)$. We have

$$P_b^N(n,g) = EB(\rho,n) = \frac{\rho^n}{n!}\left(\sum_{k=0}^{n}\frac{\rho^k}{k!}\right)^{-1}. \tag{3.2}$$

As Equation 3.2 shows, after we allow handoff calls to preempt those new calls that are occupying the guard channels when the system is full, the blocking probability of handoff calls decreases from $EB(\rho, n)$ to $P_b^h(n, g)$. This decrease implies (by PASTA[2] property) that in unit time, the average number of new calls being preempted by handoff calls is $\lambda_2 \left[EB(\rho, n) - P_b^h(n, g) \right]$. Therefore, the dropping probability of new calls can be calculated as

$$P_d^N(n, g) = \frac{\lambda_2 \left[EB(\rho, n) - P_b^h(n, g) \right]}{\lambda_1}. \tag{3.3}$$

Next, consider two special cases of the M2 model: when we set $g = n$ and when we set $g = 0$.

In the first case, we set $g = n$, then all channels are guard channels and when the system is full, incoming handoff calls can preempt new calls on any channel. By setting $g = n$ in Equation 3.1, we obtain the blocking probability of handoff calls:

$$\begin{aligned} P_b^h(n, g) &= \frac{\frac{\rho^0}{n!} \rho_2^n}{\sum_{m=0}^{-1} \frac{\rho^m}{m!} + \sum_{m=0}^{n} \frac{\rho^0}{m!} \rho_2^m} \\ &= \frac{\frac{\rho_2^n}{n!}}{\sum_{m=0}^{n} \frac{\rho_2^m}{m!}}, \end{aligned} \tag{3.4}$$

which reduces to the Erlang B formula $EB(\rho_2, n)$. This tells us that in the M2 model, the handoff calls when $g = n$ can be modeled by an $M/M/n/n$ queueing system. The blocking probability of new calls, $P_b^N(n, g)$, will remain unchanged because it does not depend on the value of g. Therefore, the dropping probability of new calls when $g = n$ is

$$P_d^N(n, g) = \frac{\lambda_2 \left[EB(\rho, n) - EB(\rho_2, n) \right]}{\lambda_1}. \tag{3.5}$$

Note that when $g = n$, the M1 and M2 models are essentially the same (that is, they both become the OM model); hence, their performance measures should also match. In Table 3.1, all three performance measures are calculated for the M1 and M2 models for different values of n. We have assumed $\lambda_1 = 30$, $\lambda_2 = 6$, $\mu = 1$, and $g = n$. The performance measures of the M1 model were calculated by the composite model method introduced in Section 2.2 and performance measures of the M2 models were calculated by the Equations 3.1, 3.2, and 3.3.

[2] PASTA is the acronym for Poisson Arrivals See Time Averages.

As predicted, both models indeed produced the same results.

Table 3.1: Performance measures for the M1 and M2 models when $g = n$

		M1 model			M2 model		
n	g	P_b^h	P_b^N	P_d^N	P_b^h	P_b^N	P_d^N
10	10	4.31%	73.19%	13.78%	4.31%	73.19%	13.78%
20	20	3.72×10^{-4}%	47.26%	9.45%	3.72×10^{-4}%	47.26%	9.45%
30	30	2.07×10^{-10}%	23.66%	4.73%	2.07×10^{-10}%	23.66%	4.73%
40	40	4.06×10^{-18}%	6.54%	1.31%	4.06×10^{-18}%	6.54%	1.31%
50	50	6.59×10^{-27}%	0.50%	0.10%	6.59×10^{-27}%	0.50%	0.10%

In the second case, we set $g = 0$, the absence of guard channels turns the M2 model into a multiserver pure loss system that is fully shared by two classes of traffic. The blocking probabilities of both handoff and new calls can be obtained by setting $g = 0$ in Equations 3.1 and 3.2. As expected, both blocking probabilities equal to $EB(\rho, n)$. Thus according to Equation 3.3, the dropping probability of new calls is zero because $P_b^h(n, g) = EB(\rho, n)$ and the numerator of the left-hand side is zero. This is also as expected because when $g = 0$, all channels are normal channels where both traffic are treated equally and no preemption can occur.

Numerical aspects

In Harine et al. [17], recursive formulae to conveniently calculate the Erlang B formula and the handoff call blocking probability of large n's can be found as

$$EB(\rho, k) = \frac{\frac{\rho}{k}EB(\rho, k-1)}{1 + \frac{\rho}{k}EB(\rho, k-1)} \tag{3.6}$$

and

$$P_b^h(n_1 + k, k) = \frac{P_b^h(n_1 + (k-1), k-1)}{\frac{n_1+k}{\rho_2} + P_b^h(n_1 + (k-1), k-1)}, \tag{3.7}$$

where $k = 1, 2, 3, ...$ and $P_b^h(n_1, 0) = EB(\rho, n_1)$.

3.2.2 Closed form solution versus simulation

In this section, call-level simulations are used to validate the closed form solutions. The input file provided to the simulation is a time-ordered sequence of call records. Each call record specifies its type (new call or handoff call), arrival time, departure time, and unique identification number. This file contains exactly $20,000$ handoff calls, and the number of new calls is approximately equal to $20000\lambda_1/\lambda_2$. Assume that n and μ are fixed to be 10 and 1, respectively. We varied the other three parameters (namely λ_1, λ_2 and g) and obtained 9 different parameter combinations. We performed 10 simulation runs using each parameter combination and then performed T-tests to determine if the mean performance measures generated by simulations are significantly different from those calculated by the closed form formula developed in the previous section. Table 3.2 lists detailed results and the significant results of T-tests (when the p-value threshold is 0.05). Because the T-test resulted in only one case where the difference is statistically significant, we conclude that the closed form solutions are well supported by the simulation results.

3.2.3 Properties of performance measures

This section examines the properties of the three performance measures of interest: the blocking probability of handoff calls, the blocking probability of new calls, and the dropping probability of new calls. By the recursive formulae presented at the end of Section 3.2.1, a set of properties is ready to be proven for $P_b^N(n,g)$ and $P_b^h(n,g)$. (For details refer to Harine et al. [17]). The dropping probability of new calls, $P_d^N(n,g)$, is a new performance measure introduced by the M2 model and its properties are also investigated.

Properties of $P_b^N(n,g)$

1. When n is fixed, $P_b^N(n,g)$ remains constant for different g because it does not depend on g. So when $g \leq n$, the following is true:

$$P_b^N(n,g) = P_b^N(n,g-1). \tag{3.8}$$

Table 3.2: Validate the closed form solution by call-level simulations. Assuming $\mu = 1$ and $n = 10$. The single asterisk sign indicates that the preceding performance measure is significantly different (when the p-value threshold is 0.05) from the corresponding mean performance measure generated by 10 simulation runs.

Parameters			Simulation results			Closed form solution		
λ_1	λ_2	g	P_b^h	P_b^N	P_d^N	P_b^h	P_b^N	P_d^N
5	1	1	0.0075	0.0434	0.0071	0.0075	0.0431	0.0071
5	1	2	0.0014	0.0436	0.0081	0.0013	0.0431	0.0084*
5	1	3	0.0002	0.0429	0.0087	0.0003	0.0431	0.0086
10	2	1	0.0674	0.3036	0.0468	0.0672	0.3019	0.0469
10	2	2	0.0173	0.3014	0.0565	0.0169	0.3019	0.0570
10	2	3	0.0050	0.3018	0.0592	0.0047	0.3019	0.0594
15	3	1	0.1396	0.4927	0.0707	0.1397	0.4935	0.0708
15	3	2	0.0465	0.4926	0.0890	0.0470	0.4935	0.0893
15	3	3	0.0173	0.4932	0.0946	0.0178	0.4935	0.0951

2. When g is fixed, $P_b^N(n, g)$ is a decreasing function of n. So when $n \geq g$, the following is true:

$$P_b^N(n, g) < P_b^N(n - 1, g). \tag{3.9}$$

3. When both n and g can vary, $P_b^N(n, g)$ decreases when both n and g decrease. So for all pairs of n and g satisfy $g \leq n$, the following is true:

$$P_b^N(n, g) > P_b^N(n - 1, g - 1). \tag{3.10}$$

Properties of $P_b^h(n, g)$

1. When n is fixed, $P_b^h(n, g)$ is a decreasing function of g. So when $g \leq n$, the following is true:

$$P_b^h(n, g) < P_b^h(n, g - 1). \tag{3.11}$$

2. When g is fixed, $P_b^h(n,g)$ is also a decreasing function of n. So when $n \geq g$, the following is true:

$$P_b^h(n,g) < P_b^h(n-1,g). \tag{3.12}$$

3. When both n and g can vary, $P_b^h(n,g)$ increases when both g and n decrease. So for all pairs of n and g satisfy $g \leq n$, the following is true:

$$P_b^h(n,g) < P_b^h(n-1,g-1). \tag{3.13}$$

Properties of $P_d^N(n,g)$

The properties of $P_d^N(n,g)$ can also be derived from Equation 3.3 and from the properties of $P_b^N(n,g)$ and $P_b^h(n,g)$ presented above.

1. When n is fixed: From the properties presented in (3.8) and (3.11) and the formula for calculating $P_d^N(n,g)$, we have for all $g \leq n$ that

$$P_d^N(n,g) > P_d^N(n,g-1). \tag{3.14}$$

2. When g is fixed: From the properties presented in (3.9) and (3.12), $EB(\rho,n)$ and $P_b^h(n,g)$ both decrease as n increases; the behavior of their difference $(EB(\rho,n) - P_b^h(n,g))$ is uncertain, which leads to the possibly complicated behavior of the new call dropping probability, $P_d^N(n,g)$. Intuitively, when g is fixed and n increases, we would expect the dropping probability of new calls to decrease; because when more channels are available for both types of traffic, new calls are less likely to have to borrow idle guard channels and then be dropped later. Furthermore, it is easy to prove that the limiting value of $P_d^N(n,g)$ is zero when n approaches infinity. Because both $EB(\rho,n)$ and $P_b^h(n,g)$ are blocking probabilities, they should approach zero when there are infinite number of channels:

$$\lim_{n\to\infty} EB(\rho,n) = \lim_{n\to\infty} P_b^h(n,g) = 0, \tag{3.15}$$

and according to the formula for calculating $P_d^N(n,g)$ (Equation 3.3), there follows

$$\begin{aligned}\lim_{n\to\infty} P_d^N(n,g) &= \frac{\lambda_2\left[\lim_{n\to\infty} EB(\rho,n) - \lim_{n\to\infty} P_b^h(n,g)\right]}{\lambda_1} \\ &= 0. \end{aligned} \tag{3.16}$$

After intuitively studying the behavior of $P_d^N(n,g)$ when g is fixed, we should also analytically study the shape of the curve of $P_d^N(n,g)$ when g is fixed; we do this by investigating its first order partial derivative with respect to n, namely, $\partial(P_d^N(n,g))\big/\partial n$. However, before we can compute this first order partial derivative of $P_d^N(n,g)$, we need to extend the domains of $EB(\rho,n)$ and $P_b^h(n,g)$ from non-negative integers to non-negative real numbers, and then the first order partial derivative follows for all $n \geq g, n \in R$ as:

$$\frac{\partial}{\partial n}(P_d^N(n,g)) = \frac{\lambda_2}{\lambda_1}\left[\frac{\partial}{\partial n}EB(\rho,n) - \frac{\partial}{\partial n}P_b^h(n,g)\right]. \tag{3.17}$$

The analytic extension of $EB(\rho,n)$ and $P_b^h(n,g)$ can be obtained by Gamma functions. As shown in Syski [54], the Erlang-B formula can be extended as follows:

$$\begin{aligned} EB(\rho,n) &= \frac{\rho^n/n!}{\sum_{m=0}^{n}\rho^m/m!} \\ &= \frac{\rho^n e^{-\rho}}{n!e^{-\rho}\sum_{m=0}^{n}\rho^m/m!} \\ &= \frac{\rho^n e^{-\rho}}{\Gamma(n+1,\rho)} \\ &= \frac{\rho^n e^{-\rho}}{\int_\rho^\infty t^n e^{-t}dt} \\ &= \left(\rho^{-n}e^{\rho}\int_\rho^\infty t^n e^{-t}dt\right)^{-1} \text{ (Let } x = t-\rho \text{)} \\ &= \left(\int_0^\infty \left(\frac{x}{\rho}+1\right)^n e^{-x}dx\right)^{-1} \text{ (Then let } z = \frac{x}{\rho}\text{)} \\ &= \left(\rho\int_0^\infty (z+1)^n e^{-\rho z}dz\right)^{-1}, \end{aligned} \tag{3.18}$$

where $\Gamma(n+1,\rho)$ denotes the upper incomplete gamma function. Furthermore, the first order partial derivative of $EB(\rho, n)$ w.r.t. n can be found in Jagerman [23]:

$$\frac{\partial}{\partial n} EB(\rho, n) = -EB^2(\rho, n) \cdot \rho \cdot \int_0^\infty \ln(z+1) \cdot (z+1)^n e^{-\rho z} dz. \tag{3.19}$$

The analytic extension of $P_b^h(n, g)$ can be obtained by Gamma functions in the same manner. The first step is to replace $n!$ by $\Gamma(n+1)$ in Equation 3.1:

$$\begin{aligned} P_b^h(n, g) &= \frac{\frac{\rho^{n-g}}{n!} \rho_2^g}{\sum_{m=0}^{n-g-1} \frac{\rho^m}{m!} + \sum_{m=n-g}^{n} \frac{\rho^{n-g}}{m!} \rho_2^{m-(n-g)}} \\ &= \frac{\rho^{n-g} \rho_2^g / \Gamma(n+1)}{\sum_{m=0}^{n-g-1} \frac{\rho^m}{m!} + \frac{\rho^{n-g}}{\rho_2^{n-g}} \sum_{m=n-g}^{n} \frac{\rho_1^m}{m!}}. \end{aligned} \tag{3.20}$$

Then the two terms in the denominator can also be expressed with Gamma functions as follows:

$$\begin{aligned} \sum_{m=0}^{n-g-1} \frac{\rho^m}{m!} &= \frac{e^\rho}{(n-g-1)!} (n-g-1)! e^{-\rho} \sum_{m=0}^{n-g-1} \frac{\rho^m}{m!} \\ &= \frac{e^\rho}{(n-g-1)!} \Gamma(n-g, \rho) \\ &= \frac{e^\rho}{(n-g-1)!} \int_\rho^\infty t^{n-g-1} e^{-t} dt \quad (\text{let } x = t - \rho) \\ &= \frac{\int_0^\infty (x+\rho)^{n-g-1} e^{-x} dx}{\Gamma(n-g)} \\ &= \frac{\int_0^\infty (x+\rho)^{n-g-1} e^{-x} dx}{\int_0^\infty x^{n-g-1} e^{-x} dx} \end{aligned} \tag{3.21}$$

and

$$\begin{aligned} \sum_{m=n-g}^{n} \frac{\rho_2^m}{m!} &= \sum_{m=0}^{n} \frac{\rho_2^m}{m!} - \sum_{m=0}^{n-g-1} \frac{\rho_2^m}{m!} \\ &= \frac{e^{\rho_2}}{n!} \Gamma(n+1, \rho_2) - \frac{e^{\rho_2}}{(n-g-1)!} \Gamma(n-g, \rho_2) \\ &= \frac{\int_0^\infty (x+\rho_2)^n e^{-x} dx}{\Gamma(n+1)} - \frac{\int_0^\infty (x+\rho_2)^{n-g-1} e^{-x} dx}{\Gamma(n-g)} \\ &= \frac{\int_0^\infty (x+\rho_2)^n e^{-x} dx}{\int_0^\infty x^n e^{-x} dx} - \frac{\int_0^\infty (x+\rho_2)^{n-g-1} e^{-x} dx}{\int_0^\infty x^{n-g-1} e^{-x} dx}. \end{aligned} \tag{3.22}$$

Then it follows from Equation 3.20 that

$$P_b^h(n,g) = \frac{f(n,g)}{h(n,g)}, \tag{3.23}$$

where

$$\begin{aligned}
f(n,g) &= \frac{\rho^{n-g}\rho_2^g}{\Gamma(n+1)} \\
&= \frac{\rho^{n-g}\rho_2^g}{\int_0^\infty x^n e^{-x} dx} && (3.24) \\
h(n,g) &= h_1(n,g) + \left(\frac{\rho}{\rho_2}\right)^{n-g} (h_2(n,g) - h_3(n,g)) && (3.25) \\
h_1(n,g) &= \frac{\int_0^\infty (x+\rho)^{n-g-1} e^{-x} dx}{\int_0^\infty x^{n-g-1} e^{-x} dx} && (3.26) \\
h_2(n,g) &= \frac{\int_0^\infty (x+\rho_2)^{n} e^{-x} dx}{\int_0^\infty x^{n} e^{-x} dx} && (3.27) \\
h_3(n,g) &= \frac{\int_0^\infty (x+\rho_2)^{n-g-1} e^{-x} dx}{\int_0^\infty x^{n-g-1} e^{-x} dx}. && (3.28)
\end{aligned}$$

Now we are ready to compute the first order partial derivative of $P_b^h(n,g)$ w.r.t. n. By the quotient rule of differentiation[3], we have

$$\frac{\partial}{\partial n} P_b^h(n,g) = \frac{h(n,g)\frac{\partial}{\partial n} f(n,g) - f(n,g)\frac{\partial}{\partial n} h(n,g)}{(h(n,g))^2}, \tag{3.29}$$

where

$$\begin{aligned}
\frac{\partial}{\partial n} f(n,g) &= \rho_2^g \frac{\ln\rho \cdot \rho^{n-g} \int_0^\infty x^n e^{-x} dx - \rho^{n-g}\rho_2^g \int_0^\infty x^n e^{-x} (\ln x) dx}{\left(\int_0^\infty x^n e^{-x} dx\right)^2} && (3.30) \\
\frac{\partial}{\partial n} h(n,g) &= \frac{\partial}{\partial n} h_1(n,g) + \left(\frac{\rho}{\rho_2}\right)^{n-g} \ln\left(\frac{\rho}{\rho_2}\right) (h_2(n,g) - h_3(n,g)) \\
&\quad + \left(\frac{\rho}{\rho_2}\right)^{n-g} \left(\frac{\partial}{\partial n} h_2(n,g) - \frac{\partial}{\partial n} h_3(n,g)\right) && (3.31)
\end{aligned}$$

[3] When choosing MAPLE 16 to numerically compute this partial derivative, we found that the use of logarithmic differentiation would greatly improve the accuracy of the results.

$$\frac{\partial}{\partial n}h_1(n,g) = \frac{\int_0^\infty (x+\rho)^{n-g-1}\ln(x+\rho)e^{-x}dx \cdot \int_0^\infty x^{n-g-1}e^{-x}dx}{\left(\int_0^\infty x^{n-g-1}e^{-x}dx\right)^2} - \frac{\int_0^\infty (x+\rho)^{n-g-1}e^{-x}dx \cdot \int_0^\infty x^{n-g-1}e^{-x}(\ln x)\,dx}{\left(\int_0^\infty x^{n-g-1}e^{-x}dx\right)^2} \tag{3.32}$$

$$\frac{\partial}{\partial n}h_2(n,g) = \frac{\int_0^\infty (x+\rho_2)^{n}\ln(x+\rho_2)e^{-x}dx \cdot \int_0^\infty x^{n}e^{-x}dx}{\left(\int_0^\infty x^{n}e^{-x}dx\right)^2} - \frac{\int_0^\infty (x+\rho_2)^{n}e^{-x}dx \cdot \int_0^\infty x^{n}e^{-x}(\ln x)\,dx}{\left(\int_0^\infty x^{n}e^{-x}dx\right)^2} \tag{3.33}$$

$$\frac{\partial}{\partial n}h_3(n,g) = \frac{\int_0^\infty (x+\rho_2)^{n-g-1}\ln(x+\rho_2)e^{-x}dx \cdot \int_0^\infty x^{n-g-1}e^{-x}dx}{\left(\int_0^\infty x^{n-g-1}e^{-x}dx\right)^2} - \frac{\int_0^\infty (x+\rho_2)^{n-g-1}e^{-x}dx \cdot \int_0^\infty x^{n-g-1}e^{-x}(\ln x)\,dx}{\left(\int_0^\infty x^{n-g-1}e^{-x}dx\right)^2}. \tag{3.34}$$

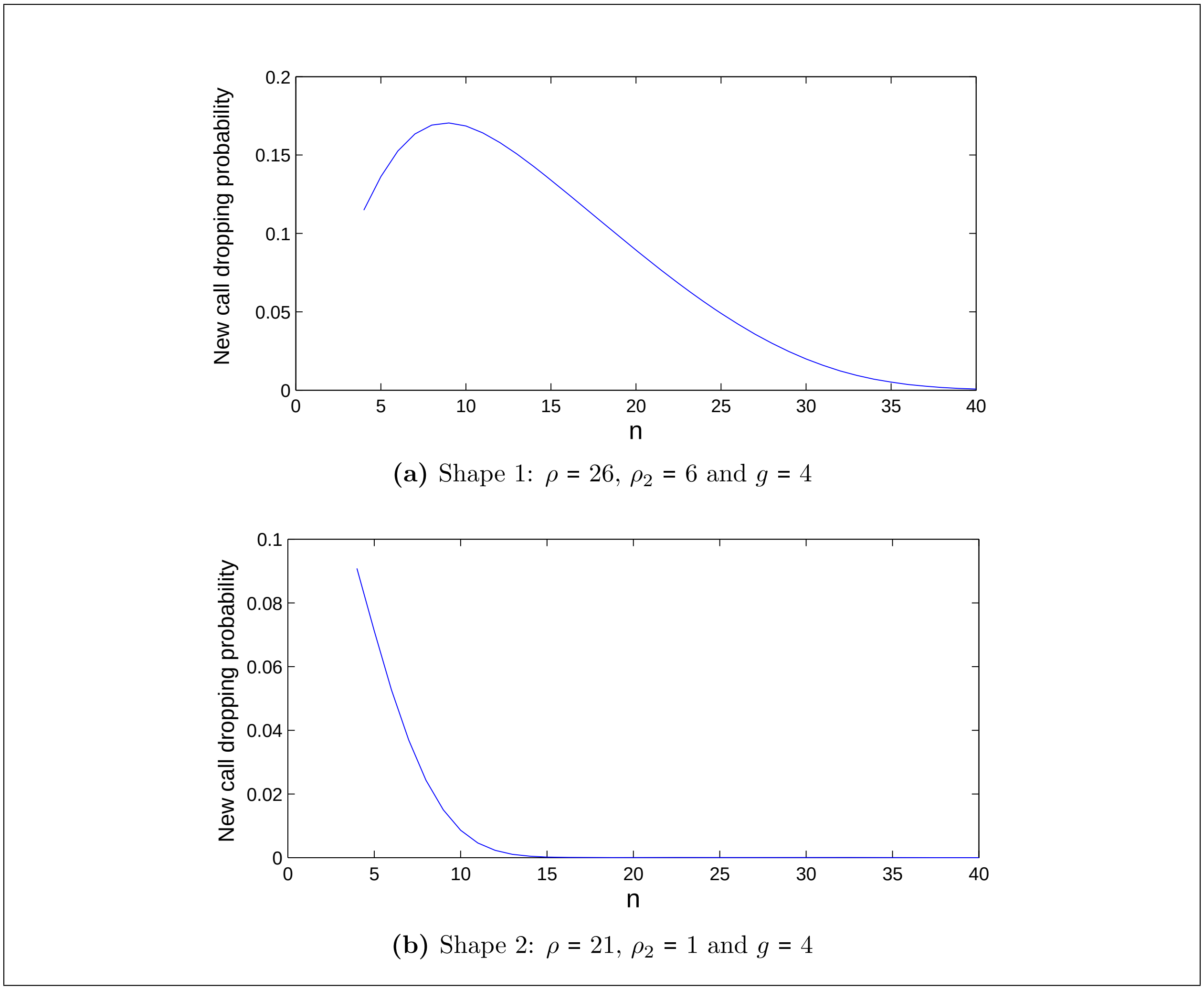
Now we have the expressions for both $\partial EB(\rho,n)/\partial n$ and $\partial P_b^h(n,g)/\partial n$. Equation 3.17 can then be used to obtain $\partial P_d^N(n,g)/\partial n$.

Although the expression of $\partial P_d^N(n,g)/\partial n$ is too complicated to be analytically tractable, we can still use it to numerically study the behavior of $P_d^N(n,g)$ when g is fixed. After extensive numerical calculations we have observed two possible patterns of $P_d^N(n,g)$. Following are two representative examples selected to illustrate these two patterns. In both examples we fixed g to be 4. **Example 1:** We set the total offered load (ρ) and the offered load for handoff call (ρ_2) to be 26 and 6, respectively; we then vary n from 4 to 40. As shown in Table 3.3, the first order partial derivative $\partial P_d^N(n,g)/\partial n$ is positive and decreases until n reaches 9, indicating that the curve of $P_d^N(n,g)$ is increasing and concave downward on the interval of $[4,9]$. Starting at $n = 9$, $\partial P_d^N(n,g)/\partial n$ becomes negative and keeps decreasing, indicating that the curve of $P_d^N(n,g)$ is now decreasing, however it is still concave downward. The concavity of $P_d^N(n,g)$ changes from downward to upward near $n = 17$ as the value of $\partial P_d^N(n,g)/\partial n$ starts to increase. **Example 2:** We set ρ and ρ_2 to be 21 and 1, respectively. Note that the offered load for handoff call is now less than the number of guard channels in this case. As we can see, the value of $\partial P_d^N(n,g)/\partial n$ is always negative, indicating that $P_d^N(n,g)$ is always decreasing. However, the concavity of $P_d^N(n,g)$ changes from concave downward to upward near $n = 7$ as $\partial P_d^N(n,g)/\partial n$ changes from decreasing to increasing. Table 3.3 lists the values of $P_d^N(n,g)$ for both examples, and Figure 3.1 plots the curves of $P_d^N(n,g)$ for both examples.

To conclude, $P_d^N(n,g)$ is not necessarily a strictly decreasing function of n when g is held constant. It could first increase until it reaches a local maxima and then decrease and approach zero as n increases (the top plot in Figure 3.1), or it could start at its local maxima and decrease as n increases (the bottom plot in Figure 3.1). We use n_c^* to denote the number of total channels where the local maxima occurs.

Table 3.3: Numerical examples to study the behavior of $\partial P_d^N(n,g)/\partial n$ and $P_d^N(n,g)$.

Example 1: A=26, A_1=6, g=4			Example 2: A=21, A_1=1, g=4		
n	$\partial P_d^N(n,g)/\partial n$	$P_d^N(n,g)$	n	$\partial P_d^N(n,g)/\partial n$	$P_d^N(n,g)$
4	NA	0.11488	4	NA	0.04020
5	0.06325	0.13620	5	-0.03940	0.03851
6	0.04537	0.15253	6	-0.04251	0.03644
7	0.02725	0.16340	7	-0.04312	0.03429
8	0.01124	0.16910	8	-0.04304	0.03214
9	-0.00155	0.17047	9	-0.04269	0.02999
10	-0.01109	0.16849	10	-0.04220	0.02787
11	-0.01790	0.16408	11	-0.04160	0.02578
12	-0.02260	0.15796	12	-0.04089	0.02371
13	-0.02576	0.15068	13	-0.04007	0.02169
14	-0.02783	0.14261	14	-0.03911	0.01971
15	-0.02910	0.13406	15	-0.03801	0.01778
16	-0.02981	0.12521	16	-0.03674	0.01591
17	-0.03009	0.11622	17	-0.03569	0.01411
18	-0.03005	0.10719	18	-0.03367	0.01238
19	-0.02975	0.09821	19	-0.03180	0.01075
20	-0.02922	0.08936	20	-0.02976	0.00921
25	-0.02379	0.04905	25	-0.02501	0.00332
30	-0.01461	0.01990	30	-0.00377	0.00068
35	-0.00549	0.00513	35	-0.00073	0.00007
40	-0.00113	0.00075	40	-0.00005	0.00000



(a) Shape 1: $\rho = 26$, $\rho_2 = 6$ and $g = 4$

(b) Shape 2: $\rho = 21$, $\rho_2 = 1$ and $g = 4$

Figure 3.1: Two possible shapes of the curve of P_d^N when g is fixed.

3.2.4 Optimization problems

In practical situations, we would like to minimize each of the three performance measures: new call blocking probability, new call dropping probabilities, and handoff call blocking probability. Hence we have a multi-objective optimization problem [22]. The decision variables are the number of guard channels (g) and the total number of channels (n). Several different methods are available for setting up the optimization problem for multiple objectives. In this section, we choose one of the three performance measures as the objective function to be minimized and impose constraints on the other two. Two representative optimization problems will be considered below.

Optimal number of guard channels

In the first optimization problem, we fix the total number of channels n and search for the optimal number of guard channels g^* to achieve all the objectives. Note that when the total number of channels n is fixed, the blocking probability for new calls, i.e., $P_b^N(n,g)$, is also determined and thus is not considered as an objective for this optimization problem. The two remaining performance measures, i.e., handoff call blocking probability ($P_b^h(n,g)$) and new call dropping probability ($P_d^N(n,g)$), are candidates can be considered as objective functions. Since handoff calls are of high priority, it is more reasonable to set a hard constraint on handoff call blocking probability than on new call dropping probability so that handoff calls perform satisfactorily. Therefore, we chose new call dropping probability as the objective function and formed the following optimization problem:

$\mathbf{O_1}$: Given ρ, ρ_2 and n, determine the optimal integer value of g so as to

$$\text{minimize } P_d^N(n,g) \text{ s.t. } P_b^h(n,g) \leq P_0^{hb}. \tag{3.35}$$

To solve this optimization problem we need to use the first property of $P_b^N(n,g)$ (Equation 3.8) and the first property of $P_d^N(n,g)$ (Equation 3.14). Because $P_b^h(n,g)$ is a decreasing function of g when n is fixed, we first determine the smallest value of g ($\leq n$), denoted by g_0, such that $P_b^h(n,g) \leq P_0^{hb}$. If we fail to find such g_0 then this optimization problem has no feasible solution for the given parameters. If such g_0 exists, then by the first property of $P_d^N(n,g)$, such g_0 minimizes $P_d^N(n,g)$ and is the optimal number of guard channels.

Table 3.4 provides numerical examples where we use $\rho = 80$, $\rho_2 = 40$, $\mu = 1$, and $n = 100$. Optimization results for HT's model are also listed for comparison. As the results suggest, both models require the same number of optimal guard channels and therefore produce the same handoff call blocking probabilities. However, when using the M2 model, the new call loss probability can be reduced between 40% and 90%.

Table 3.4: Optimization problem $\mathbf{O}_1$: numerical examples. This table lists results from the M2 model and HT's model. P_L^N is the total loss probability of new call, which is P_b^N for HT's model and the sum of P_b^N and P_d^N for the M2 model.

Constraint	M2 Model			HT's Model		
P_0^{hb}	g^*	P_b^h	P_L^N	g^*	P_b^h	P_L^N
10^{-2}	0	0.003992	0.003992	0	0.003992	0.003992
10^{-3}	3	0.000504	0.007490	3	0.000504	0.012528
10^{-4}	6	0.000065	0.007920	6	0.000065	0.023195
10^{-5}	9	0.000008	0.007976	9	0.000008	0.038967
10^{-6}	13	0.00000058	0.007983	13	0.00000058	0.069839

Optimal number of channels

In this second optimization problem, we want to find the optimal n so as to meet the constraints imposed on the three performance measures. We formed the following optimization problem: $\mathbf{O_2}$: Given ρ and ρ_2, determine the optimal integer value of g and n so as to

$$\text{minimize } n \text{ s.t.} \begin{cases} P_b^N(n,g) \leq P_0^{Nb} \\ P_b^h(n,g) \leq P_0^{hb} \\ P_d^N(n,g) \leq P_0^{Nd} \end{cases} . \tag{3.36}$$

From Equation 3.2 we know that $P_b^N(n,g)$ depends only on n and is independent of g. In order to reduce the complexity of this problem, we can handle the constraint $P_b^N(n,g) \leq P_0^{Nb}$ separately from the other two constraints by solving two independent subordinate optimization problems and then combine their results to obtain the final optimal solution (n^*, g^*). Therefore the procedure for solving optimization problem $\mathbf{O_2}$ can be broken down into three steps.

Procedure for solving optimization problem $\boldsymbol{O_2}$

In the **first step** we solve the first subordinate optimization problem in which we only consider the constraint for new call blocking probability $P_b^N(n,g) \leq P_0^{Nb}$. The smallest n that satisfies $P_b^N(n_{Nb},g) \leq P_0^{Nb}$, denoted by n_{Nb}, can be found. The optimal n for optimization problem $\mathbf{O_2}$ should be at least n_{Nb}.

In the **second step** we need to solve the second subordinate optimization problem $\mathbf{O_2'}$ defined as

$$\text{minimize } n \text{ s.t. } \begin{cases} P_b^h(n,g) \leq P_0^{hb} \\ P_d^N(n,g) \leq P_0^{Nd} \end{cases}. \tag{3.37}$$

The procedure for solving optimization problem $\mathbf{O_2'}$ is more complicated and will be introduced later in this section. For now let us assume that $(n^{*\prime}, g^{*\prime})$ is the solution.

In the **third step** we find the final solution (n^*, g^*) to the optimization problem $\mathbf{O_2}$. Of all three performance measures, only $P_d^N(n,g)$ can be a nonmonotonic function of n when g is fixed (see the second property of $P_d^N(n,g)$); therefore, we need to determine the value of n , denoted by n_c^*, at which $P_d^N(n, g^{*\prime})$ reaches its local maxima. We can then determine the optimal solution to optimization problem $\mathbf{O_2}$ based on the relationship among n_{Nb}, $n^{*\prime}$, and n_c^* as follows:

1. If $n^{*\prime} \geq n_{Nb}$, then the solution to $\mathbf{O_2}$ is just $(n^{*\prime}, g^{*\prime})$;
2. If $n_c^* < n^{*\prime} < n_{Nb}$, then we have $P_b^h(n_{Nb}, g^{*\prime}) < P_b^h(n^{*\prime}, g^{*\prime}) \leq P_0^{hb}$and $P_d^N(n_{Nb}, g^{*\prime}) < P_d^N(n^{*\prime}, g^{*\prime}) \leq P_0^{Nd}$. Therefore the optimal number of channels is just n_{Nb} and the solution to $\mathbf{O}_2$ is $(n_{Nb}, g^{*\prime})$.
3. If $n^{*\prime} < n_c^* < n_{Nb}$ or $n^{*\prime} < n_{Nb} < n_c^*$, we have $P_b^h(n_{Nb}, g^{*\prime}) < P_b^h(n^{*\prime}, g^{*\prime}) \leq P_0^{hb}$ but the relationship between $P_d^N(n_{Nb}, g^{*\prime})$ and P_0^{Nd} is not certain. In this case, we have to start with n_{Nb} and search for the optimal combination (n^*, g^*) that satisfies

$$\begin{cases} n^* \geq n_{Nb} \\ P_d^N(n^*, g^*) \leq P_0^{Nd} \\ P_b^h(n^*, g^*) \leq P_0^{hb} \end{cases} \tag{3.38}$$

The procedure for searching for the solution (n^*, g^*) for this scenario is as follows:

```
**Step 1**: Set $n = n_{Nb}$
**Step 2**: Find the smallest $g$ that satisfies $P_b^h(n, g) \leq P_0^{hb}$
**Step 3**:
**if** $P_d^N(n, g) \leq P_0^{Nd}$ **then**
    return $(n, g)$ as solution to $\mathbf{O_2}$
**else**
    $n = n + 1$ and goto Step 2
**end if**
```

Procedure for solving optimization problem $\boldsymbol{O_2'}$

In order to solve the optimization problem $\mathbf{O_2'}$, we consider the region that is in the first quadrant of the (n, g) plane and below line $n = g$. First, we examine the contour diagram of both $P_b^h(n, g)$ and $P_d^N(n, g)$. As Figures 3.2 and 3.3 indicate, the contour curves of $P_d^N(n, g)$ have three different patterns (Figure 3.3b - 3.3d) and the contour curves of $P_b^h(n, g)$ all follow the same pattern (Figure 3.3a). These patterns of contour curves can be verified by the properties of $P_b^h(n, g)$ and $P_d^N(n, g)$ presented in Section 3.2.3. Therefore, for the problem at hand, we need to first determine the pattern to which the contour curve of $P_d^N(n, g) = P_0^{Nd}$ belongs by the following procedure:

1) If $P_d^N(1, 1) > P_0^{Nd}$, then it follows pattern 3

2) If $P_d^N(1, 1) \leq P_0^{Nd}$, then it follows either pattern 1 or pattern 2. In either case, the contour curve intersects the $n = g$ line twice and therefore it has two points of intersection, which can be denoted by c (the point on the left) and d (the point on the right), respectively. Let n_d^1 (or n_d^2) be the n-coordinate associated with point c (or d). Then we can distinguish pattern 2 from pattern 1 by checking if there exists a number $n_0 \in [n_d^1, n_d^2]$ such that $P_d^N(n_0, 1) > P_0^{Nd}$. If such n_0 can be found, then the contour curve follows pattern 2. Otherwise it follows pattern 1.

After the pattern of contour curve $P_d^N(n, g) = P_0^{Nd}$ is determined, it is time to consider contour curve $P_b^h(n, g) = P_0^{hb}$ and to look for the feasible region that satisfies both $P_d^N(n, g) \leq P_0^{Nd}$ and $P_b^h(n, g) \leq P_0^{hb}$.

Pattern 3

The following investigates when contour curve $P_d^N(n,g) = P_0^{Nd}$ follows pattern 3 (see Figure 3.4). Three different cases will each lead to different solutions to optimization problem $\mathbf{O_2'}$.

- **Case a1**: Contour curve $P_b^h(n,g) = P_0^{hb}$ is to the left of contour curve $P_d^N(n,g) = P_0^{Nd}$ and they do not intersect, as shown in Figure 3.4a. The region to the right of contour curve $P_d^N(n,g) = P_0^{Nd}$ and below line $n = g$, as well as all the points on its boundaries (i.e. line $n = g$, contour curve $P_d^N(n,g) = P_0^{Nd}$ and line $g = 0$) will satisfy the constraint $P_d^N(n,g) \leq P_0^{Nd}$. Note that all points on line $g = 0$ will also satisfy constraint $P_d^N(n,g) \leq P_0^{Nd}$ because $P_d^N(n,0)$ is always zero. The region to the right of contour curve $P_b^h(n,g) = P_0^{hb}$ and bounded by line $n = g$ and $g = 0$ will satisfy $P_b^h(n,g) \leq P_0^{hb}$. Clearly, the optimal solution to optimization problem $\mathbf{O_2'}$ is one of the following two points, whichever has a smaller n coordinate: (1) the point on line $g = 0$ whose n coordinate is the smallest possible value of n such that $P_b^h(n,0) \leq P_0^{hb}$, or (2) the point labelled as f on line $g = 1$ whose n coordinate is denoted by n_h', which will be introduced shortly. These two candidates of optimal solution to $\mathbf{O_2'}$ are indicated in Figure 3.4a as P_1'.
- **Case a2**: Contour curve $P_b^h(n,g) = P_0^{hb}$ intersects contour curve $P_d^N(n,g) = P_0^{Nd}$ at point $\mathrm{P}_2(n_2, g_2)$, as shown in Figure 3.4b. As indicated in Figure 3.4b, the two contour curves divide the area under line $n = g$ into four regions, namely,
  - R_1: in which all (n,g) satisfy $P_b^h(n,g) > P_0^{hb}$ and $P_d^N(n,g) < P_0^{Nd}$.
  - R_2: in which all (n,g) satisfy $P_b^h(n,g) > P_0^{hb}$ and $P_d^N(n,g) > P_0^{Nd}$.
  - R_3: in which all (n,g) satisfy $P_b^h(n,g) < P_0^{hb}$ and $P_d^N(n,g) > P_0^{Nd}$.
  - R_4: in which all (n,g) satisfy $P_b^h(n,g) < P_0^{hb}$ and $P_d^N(n,g) < P_0^{Nd}$.

  Because region R_1 and R_4 satisfy $P_d^N(n,g) \leq P_0^{Nd}$, and region R_3 and R_4 satisfy $P_b^h(n,g) \leq P_0^{hb}$, the feasible region is then region R_4. As shown in Figure3.4b, in region R_4, the point labeled as P_2 has the smallest n coordinate and because $n^{*\prime}$ and $g^{*\prime}$ must be integers, the solution $(n^{*\prime}, g^{*\prime})$ should be the point that is in region R_4 and has the shortest distance to point P_2 among all the points in R_4. Algorithm $\mathbf{P}_2$, which will be introduced shortly, can be used to find the solution $(n^{*\prime}, g^{*\prime})$.

- **Case a3**: Contour curve $P_b^h(n,g) = P_0^{hb}$ is to the right of contour curve $P_d^N(n,g) = P_0^{Nd}$ and they do not intersect with each other, as shown in Figure 3.4c. The only active constraint for this case is $P_b^h(n,g) \le P_0^{hb}$, as the entire region to the right of contour curve $P_b^h(n,g) = P_0^{hb}$ and below line $n = g$ will satisfy both $P_b^h(n,g) \le P_0^{hb}$ and $P_d^N(n,g) \le P_0^{Nd}$. The optimal number of channels $n^{*\prime}$ would be the smallest n such that $P_b^h(n,n) \le P_0^{hb}$, and we have $g^{*\prime} = n^{*\prime}$. This solution is labeled as P_3 in Figure 3.4c.

The above introduced the three different cases of the optimization problem $\mathbf{O_2'}$ when contour curve $P_d^N(n,g) = P_0^{Nd}$ follows pattern 3. The following is the procedure for determining which of the above three cases can be applied given a set of parameters:

1) Find point a, which is the intersection point of line $n = g$ and contour curve $P_b^h(n,g) = P_0^{hb}$. Starting with $n = 1$ and set $g = n$, search for the smallest integer n, denoted by n_h, such that $P_b^h(n_h, n_h) \le P_0^{hb}$.

2) Find point d, which is the intersection point of line $n = g$ and contour curve $P_d^N(n,g) = P_0^{Nd}$. Starting with $n = 1$ and set $g = n$, search for the smallest integer n, denoted by n_d, such that $P_d^N(n,n) \le P_0^{Nd}$.

3) If $n_h \ge n_d$, hence case **a3** applies.

4) If $n_h < n_d$, fix g to be 1, let n vary, and search for the smallest integer n_h' (which is associated with point f in Figure 3.4), such that $P_b^h(n_h', 1) \le P_0^{hb}$; also search for the smallest integer n_d', such that $P_d^N(n_d', 1) \le P_0^{Nd}$. Then if $n_h' < n_d'$, the two contours do not intersect, hence case **a1** applies. On the other hand, if $n_h' \ge n_d'$ is the case, it implies that the two contours do intersect hence case **a2**.

The following algorithm $\mathbf{P}_2$ can be used to find the optimal solution $(n^{*\prime}, g^{*\prime})$ when the two contours intersect (i.e., case **a2**). This algorithm is based on the bisection technique introduced in Harine et al. [17], but it implements a different search pattern.

Algorithm $\boldsymbol{P}_2$

```
**Part A**: Set $g$ to be 1. First determine $n_h^{'}$, the smallest value of $n$ such that $P_b^h(n_h^{'}, 1) \le P_0^{hb}$. Then determine $n_d^{'}$, the smallest value of $n$ such that $P_d^N(n_d^{'}, 1) \le P_0^{Nd}$.
$g \leftarrow 0$
$N_{\max} \leftarrow n_h^{'}$
$N_{\min} \leftarrow n_d^{'}$
Step 1:
if $N_{\max} - N_{\min} > 1$ then
    $N_{\text{mid}} \leftarrow (N_{\max} + N_{\min})/2$
    $N \leftarrow N_{\text{mid}}$
else $g \leftarrow g + 1$ and then use Part B to find the solution
end if
Step 2: $g \leftarrow g + 1$
if $g > N$ then
    go to Part B
else calculate $P_b^h(N, g)$ and $P_d^N(N, g)$
    if $(N, g) \in R_1$ then
        go to Step 2
    else if  $(N, g) \in R_2$ then
        $N_{\min} \leftarrow N_{\text{mid}}$
        $g \leftarrow g - 1$;
        go to Step 1
    else if $(N, g) \in R_3$ then
        Use Part B to find the solution
    else                          → This is the case when $(N, g) \in R_4$
        $N_{\max} \leftarrow N_{\text{mid}}$
        $g \leftarrow g - 1$
        go to Step 1
    end if
end if
```

```
End Part A

Part B:
for $N \leftarrow N_{\min}$ to $N_{\max}$ by 1 do
    for $g' \leftarrow g - 1$ to $g + 1$ by 1 do
        Find the smallest $N$ s.t. $P_b^h(N, g') \leq P_0^{hb}$ and $P_d^N(N, g') \leq P_0^{Nd}$ and break
    end for
end for
$n^{*'} \leftarrow N$
$g^{*'} \leftarrow g'$
End Part B
```

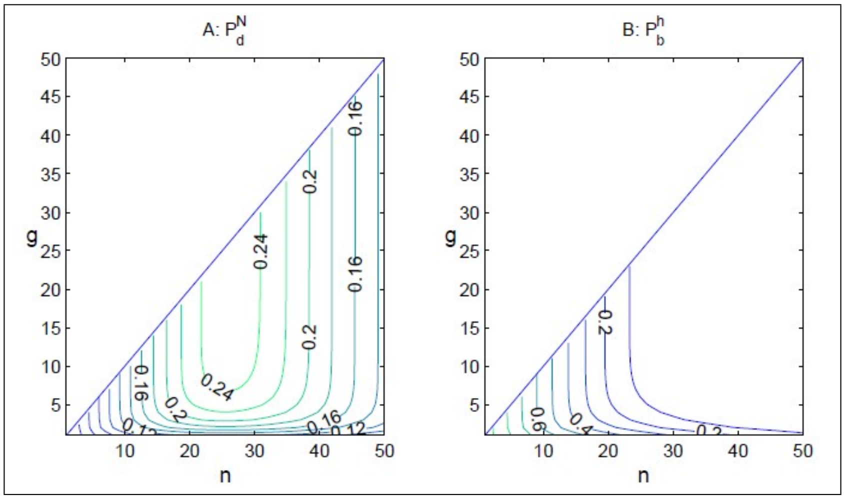
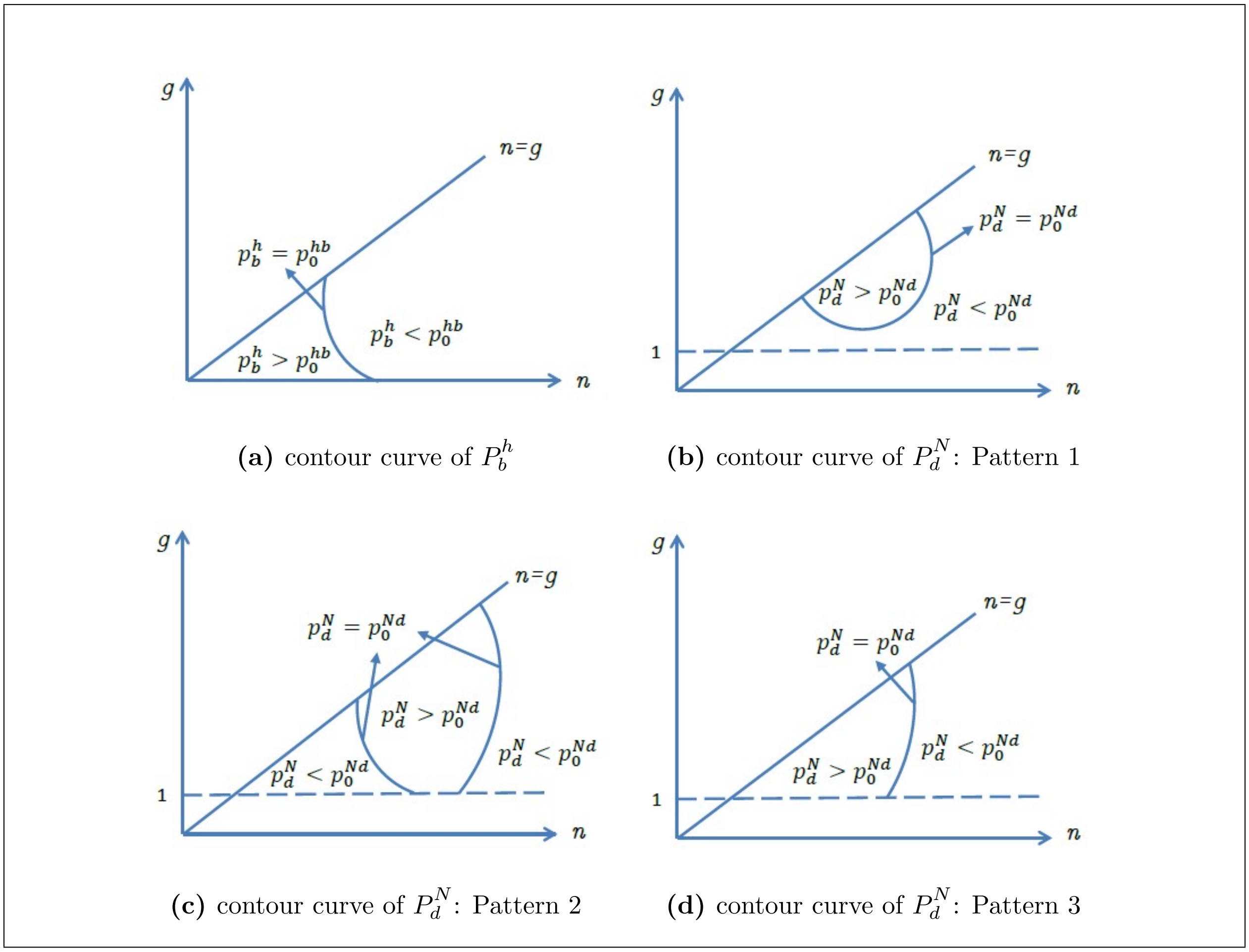


Figure 3.2: Sample contour curves of P_d^N and P_b^h. A): contour curves of P_d^N. B): Contour curves of P_b^h. Both are generated by the same parameters ($\lambda_1 = 49$, $\lambda_2 = 21$, $\mu = 1$)

Pattern 2

Next, we investigate when contour curve $P_d^N(n, g) = P_0^{Nd}$ follows pattern 2 (see Figure 3.5). In order to solve optimization problem $\mathbf{O_2'}$ when contour curve $P_d^N(n, g) = P_0^{Nd}$ belongs to pattern 2, we need to first establish several important values of n associated with points a through f labeled in Figure 3.5a - 3.5b.



(a) contour curve of P_b^h

(b) contour curve of P_d^N: Pattern 1

(c) contour curve of P_d^N: Pattern 2

(d) contour curve of P_d^N: Pattern 3

Figure 3.3: Different patterns for contour curves of P_b^h and P_d^N.

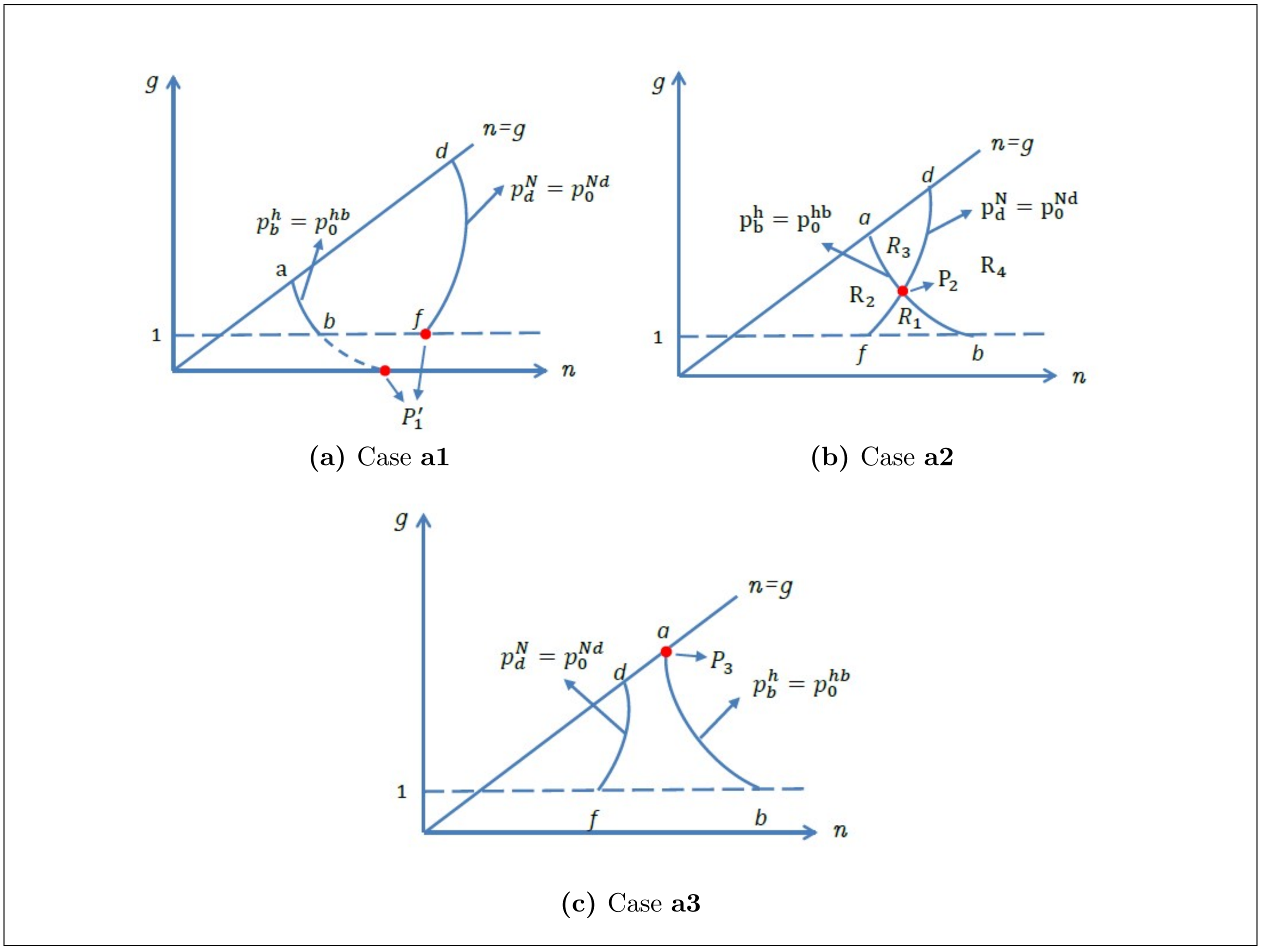
Point a: Set $g = n$. Starting with $n = 1$, search for the smallest integer n, denoted by n_h, such that $P_b^h(n_h, n_h) \le P_0^{hb}$.

Point b: Set $g = 1$. Starting with $n = 1$, search for the smallest integer n, denoted by $n_h^{'}$, such that $P_b^h(n_h^{'}, 1) \le P_0^{hb}$.

Point c: Set $g = n$. Starting with $n = 1$, search for the smallest integer n, denoted by n_d^1, such that $P_d^N(n_d^1, n_d^1) \le P_0^{Nd}$ and $P_d^N(n_d^1 + 1, n_d^1 + 1) > P_0^{Nd}$.

Point d: Set $g = n$. Starting with $n = 1$, search for the smallest integer n, denoted by n_d^2, such that $P_d^N(n_d^2 - 1, n_d^2 - 1) > P_0^{Nd}$ and $P_d^N(n_d^2, n_d^2) \le P_0^{Nd}$.

Point e: Set $g = 1$. Starting with $n = 1$, search for the smallest integer n, denoted by $n_d^{1'}$, such that $P_d^N(n_d^{1'}, 1) \le P_0^{Nd}$ and $P_d^N(n_d^{1'} + 1, 1) > P_0^{Nd}$.



(a) Case **a1**

(b) Case **a2**

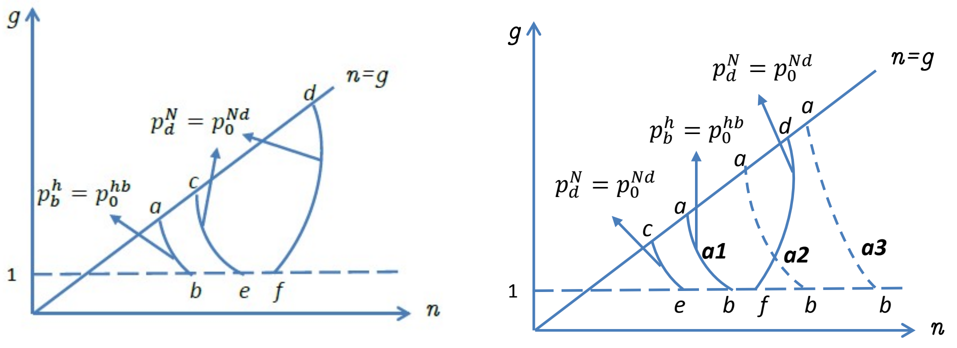
(c) Case **a3**

Figure 3.4: Different scenarios when the contour curve of P_d^N belongs to pattern 3.

Point f: Set $g = 1$. Starting with $n = 1$, search for the smallest integer n, denoted by $n_d^{2\prime}$, such that $P_d^N(n_d^{2\prime} - 1, 1) > P_0^{Nd}$ and $P_d^N(n_d^{2\prime}, 1) \leq P_0^{Nd}$.

Now according to the relative positions of these 6 points we can break the optimization problem $\mathbf{O_2'}$ down into 2 different cases:

- **Case b1**: As shown in Figure 3.5a, point a is to the left of point c. This can be determined by checking if $n_h \leq n_d^1$ is true. For this case, the region below line $n = g$ and to the left of the left branch of contour curve $P_d^N(n, g) = P_0^{Nd}$ (curve $\widetilde{ce}$), or to the right of the right branch of contour curve $P_d^N(n, g) = P_0^{Nd}$ (curve $\widetilde{df}$), including all the points on line $g = 0$ will satisfy $P_d^N(n, g) \leq P_0^{Nd}$. The region below line $n = g$ and to the right of contour curve $P_b^h(n, g) = P_0^{hb}$ will satisfy $P_b^h(n, g) \leq P_0^{hb}$. Therefore because only integer values of $n^{*\prime}$ and $g^{*\prime}$ are allowed, the optimal solution $(n^{*\prime}$,$g^{*\prime})$



(a) Case **b1** **(b)** Case **b2**

Figure 3.5: Different scenarios when the contour curve of P_d^N belongs to pattern 2.

should be point (n_h, n_h).

- **Case b2**: This case happens when $n_h > n_d^1$, that is, when point a is to the right of point c (Figure 3.5b). Since the region to the left of contour curve $P_b^h(n,g) = P_0^{hb}$ will violate the constraint $P_b^h(n,g) \leq P_0^{hb}$, we can ignore the left branch of contour curve $P_d^N(n,g) = P_0^{Nd}$ and focus on its right branch (curve $\widetilde{df}$) together with contour curve $P_b^h(n,g) = P_0^{hb}$. The situations in this case are exactly same as the situations when contour curve $P_d^N(n,g) = P_0^{Nd}$ follows pattern 1:

  1) If $n_h \geq n_d^2$, then point a is to the right of point d and the situation is same as case **a3.**

  2) If $n_h < n_d^2$ and $n_h' < n_d^{2\prime}$, then point a is to the left of point d and point b is to the left of point f. Therefore curves $\widetilde{ab}$ and $\widetilde{df}$ do not intersect and the situation is same as case **a1**.

  3) If $n_h < n_d^2$ and $n_h' \geq n_d^{2\prime}$, then curves $\widetilde{ab}$ and $\widetilde{df}$ do intersect and the situation is same as case **a2**.

Pattern 1

Finally we investigate when contour curve $P_d^N(n,g) = P_0^{Nd}$ follows **pattern 1** (see Figure 3.6). As illustrated in Figure 3.6, the left and right branches of contour curve $P_d^N(n,g) = P_0^{Nd}$

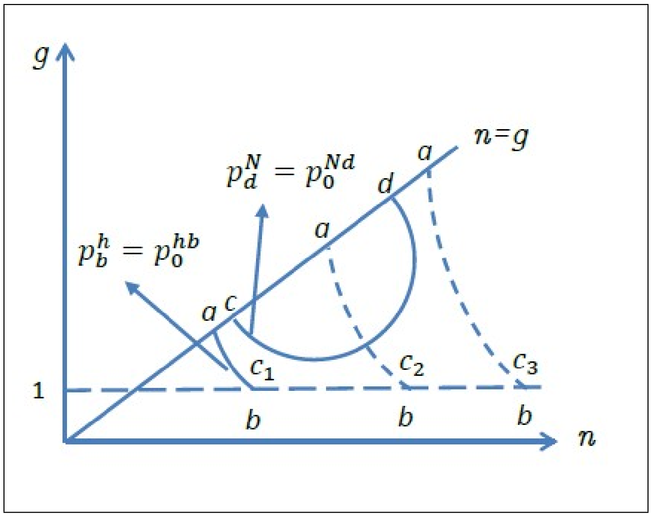


Figure 3.6: Different scenarios when the contour curve of P_d^N belongs to pattern 1.

connect with each other and form a single curve. First, the n values associated with points a to d can be obtained as introduced on Page 90. Then we have three different cases:

- **Case c1**: When $n_h \leq n_d^1$, point a is to the left of point c. Similar to case **b1**, the optimal solution ($n^{*'}$,$g^{*'}$) is just (n_h, n_h).
- **Case c2**: When $n_d^1 < n_h < n_d^2$, point a is in between point c and d. Similar to case **a2**, the optimal solution can be found by algorithm $\mathbf{P}_2$.
- **Case c3**: When $n_h \geq n_d^2$, point a is to the right of point d. Similar to case **a3**, the optimal solution ($n^{*'}$,$g^{*'}$) is just (n_h, n_h).

To summarize, complete procedures have been established to find the optimal solution ($n^{*'}$, $g^{*'}$) to optimization problem $\mathbf{O_2'}$ when the contour curve $P_d^N(n, g) = P_0^{Nd}$ follows each of the three patterns.

Numerical experiments

In this section we use numerical experiments to compare the optimal number of channels required by the M2 model and HT's model. Our goal is to answer the following questions:

1. Is there a difference between the two models?
2. If there is a difference, then how does one model differ from the other?

First, as a numerical illustration, we take the same parameters as those used in Harine et al. [17]: $\lambda_1 = \lambda_2 = 40$, $\mu = 1$, and we calculate the solutions to optimization problem

$\mathbf{O_2}$. Various constraints on handoff call blocking (P_0^{hb}) and new call loss (P_0^{NL}) are used and the results are summarized in Table 3.5. Note that the new call loss in the M2 model has two components—new call blocking and new call dropping—for HT's model the new call loss only includes new call blocking. In the optimization problem $\mathbf{O_2}$ we set separate constraints for each of the two components, i.e., P_0^{Nb} (constraint for new call blocking) and P_0^{Nd} (constraint for new call dropping). In order to compare the M2 model to HT's model, we define $P_0^{NL} = P_0^{Nb} + P_0^{Nd}$. In this numerical illustration, we simply set $P_0^{Nb} = P_0^{Nd} = P_0^{NL}/2$. As shown in Table 3.5, the differences between the optimal number of channels (n^*) required by these two models are insignificant.

Table 3.5: Optimization problem $\mathbf{O}_2$: An illustration. This table lists optimal n's and g's required to meet various performance constraints by the M2 model and HT's model . P_L^N is the total loss probability of new call, which for the M2 model equals the sum of P_b^N and P_d^N, and for HT's model equals P_b^N.

Constraints		M2 Model				HT's Model			
P_0^{hb}	P_0^{NL}	n^*	g^*	P_b^h	P_L^N	n^*	g^*	P_b^h	P_L^N
10^{-3}	10^{-2}	100	3	0.00050421	0.0074798	101	2	0.00079145	0.0077859
10^{-4}	10^{-3}	108	3	0.00005835	0.0008741	109	2	0.00008556	0.0009482
10^{-5}	10^{-4}	115	3	0.00000553	0.0000829	116	2	0.00000763	0.0000933
10^{-6}	10^{-5}	121	3	0.00000052	0.0000079	122	2	0.00000069	0.0000091

A series of numerical experiments were then conducted to thoroughly investigate the difference between the number of optimal channels (n^*) required by HT's model and by the M2 model to meet various call performance constraints. In particular, we were interested in studying the relative difference (in percentage) in n^* between the two models. The relative difference (in percentage) with respect to the HT's model is defined as:

$$D = \frac{n_{M2}^* - n_{HT}^*}{n_{HT}^*} \times 100\%, \tag{3.39}$$

where n_{M2}^* and n_{HT}^* are the optimal number of channels required by the M2 model and HT's model, respectively. Therefore a negative value of D indicates that the M2 model requires fewer channels than HT's model to meet the given call performance constraints. Without loss of generality, we set $\mu = 1$. The remaining parameters and their levels are shown in Table 3.6, as are a set of constraints used for setting up the optimization problems. Two points are worth noting: (1) The constraint on handoff call blocking (P_0^{hb}) is always less than the constraint on new call loss (P_0^{NL}) because handoff calls are high priority traffic and need to be protected with a stricter constraint; (2) When breaking down the constraint on new call loss (P_0^{NL}) into two components—i.e., constraints on new call blocking (P_0^{Nb}) and dropping (P_0^{Nd})—we set P_0^{Nb} to be 50%, 75%, or 90% of P_0^{NL}. Note that P_0^{Nb} is at least 50% of P_0^{NL} because we want to ensure that $P_0^{Nd} \leq P_0^{Nb}$. The reasoning behind this is that dropping an ongoing call is more serious than blocking an incoming call [62] and therefore it is better to have a stricter constraint for call dropping than for call blocking.

Table 3.6: Experiment parameters and levels used to compare the M2 model and HT's model

Parameter	Level(s)
λ_1	20
λ_2	1, 1.5, 2, 4, 10, 20, 40, 80, 100, 200
P_0^{hb}	10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}
P_0^{NL}	$> P_0^{hb}$ and $= 10^{-k}$, $k = 1, 2, ..., 5$
P_0^{Nb}	$50\% \times P_0^{NL}$, $75\% \times P_0^{NL}$, $90\% \times P_0^{NL}$

Based on all the parameters listed in Table 3.6, there are a total of 450 different combinations [4]of parameters. For each combination, we solve optimization problem $\mathbf{O_2}$ for optimal number of channels (n^*) for both the M2 model and HT's model. The procedures developed in this chapter will be used for the M2 model, and the procedures presented in Harine et al. [17] will be adopted for HT's model. The relative difference in percentage, D, is calculated for each parameter combination by Equation 3.39. The results suggest that about 67% of all D values are negative; this means that about 67% of the time, employing the M2 model will reduce the optimal number of channels required. Focusing on the more significant differences (where $|D| \geq 5\%$), we found that out of 450 parameter combinations, 130 (about 30%) pro-

[4] Note that the value of P_0^{NL} must be greater than the value of P_0^{hb}, so there are only 450 combinations instead of 750.

duced D values that satisfied $|D| \geq 5\%$ and 123 (about 95%) were negative. These statistics indicate that choosing the M2 model over HT's model will notably reduce the number of channels required. Therefore, the answer to the first question—is these a difference between the two models?—is a definitive, "YES."

To address the question about how the models differ, we first investigate the effect of relative offered load (the ratio of λ_1/μ to λ_2/μ, or just λ_1/λ_2) on the relative difference D. Figure 3.8 presents boxplots of D grouped by different values of λ_1/λ_2. No obvious pattern has been found from these plots except in Figure 3.7B (when $P_0^{Nb} = 50\% \times P_0^{NL}$), where the median of D first decreases as λ_1/λ_2 increases until $\lambda_1 = \lambda_2$, and then it increases and approaches zero as λ_1/λ_2 increases. To conclude, the median of D does not seem to be affected significantly by the value of λ_1/λ_2 and is almost always less than zero, suggesting that the M2 model generally requires fewer channels than HT's model.

Next, we examine the effect of the ratio of constraints (P_0^{hb}/P_0^{NL}) on D. As explained before, the call performance constraint on high-priority traffic (handoff calls) should be stricter than that on low-priority traffic (new calls). Therefore, we set the ratio P_0^{hb}/P_0^{NL} to be less than one and let it vary from 10^{-5} to 10^{-1}. Figure 3.8 presents the boxplots of D grouped by different values of P_0^{hb}/P_0^{NL}. The pattern exhibited is consistent across all the subplots in the figure: the relative difference D increases as P_0^{hb}/P_0^{NL} increases. Therefore, in channel utilization, where the call loss constraint for high-priority traffic is much stricter than for low-priority traffic, the M2 model offers a substantial advantage over HT's model. On average, employing the M2 model when $P_0^{hb}/P_0^{NL} = 10^{-5}$ reduces the total number of channels required by 10%.

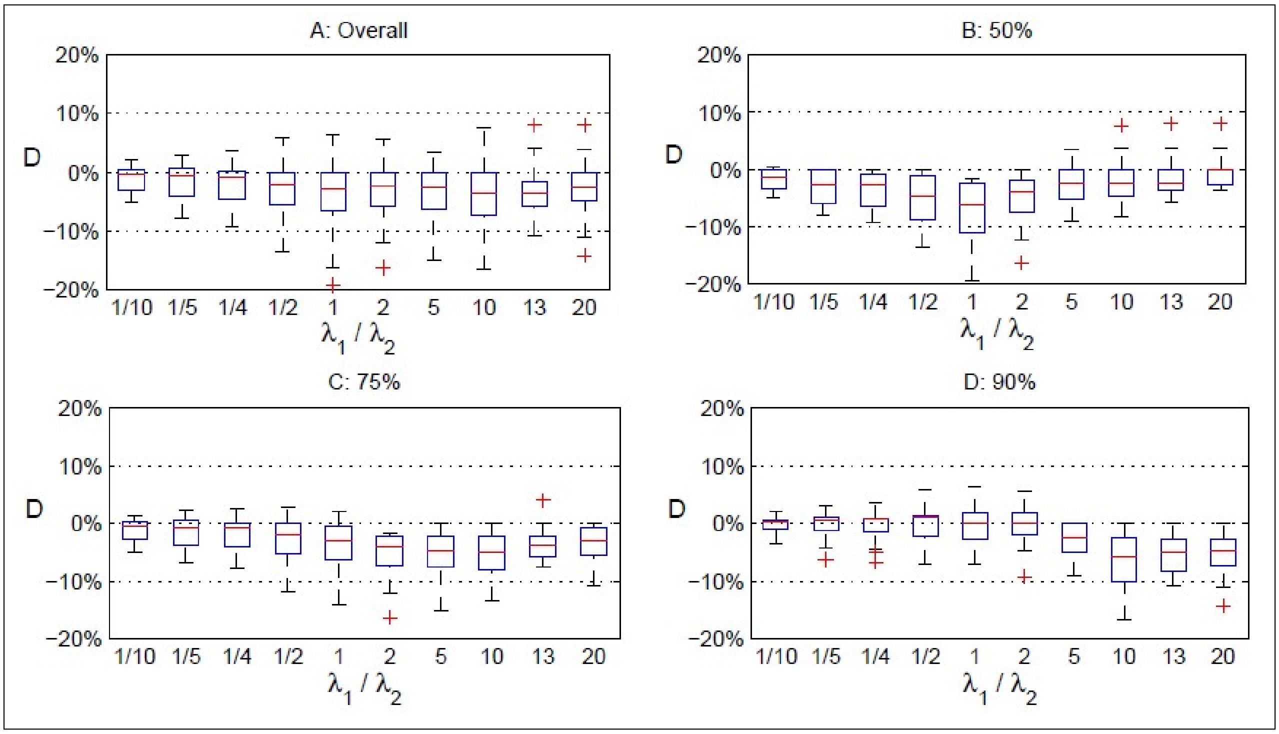


Figure 3.7: Boxplots of relative difference D at different values of λ_1/λ_2. A) All data points are included. B) When the *new call* blocking constraint P_0^{Nb} is 50% of P_0^{NL}. C) When the *new call* blocking constraint P_0^{Nb} is 75% of P_0^{NL}. D) When the *new call* blocking constraint P_0^{Nb} is 90% of P_0^{NL}.

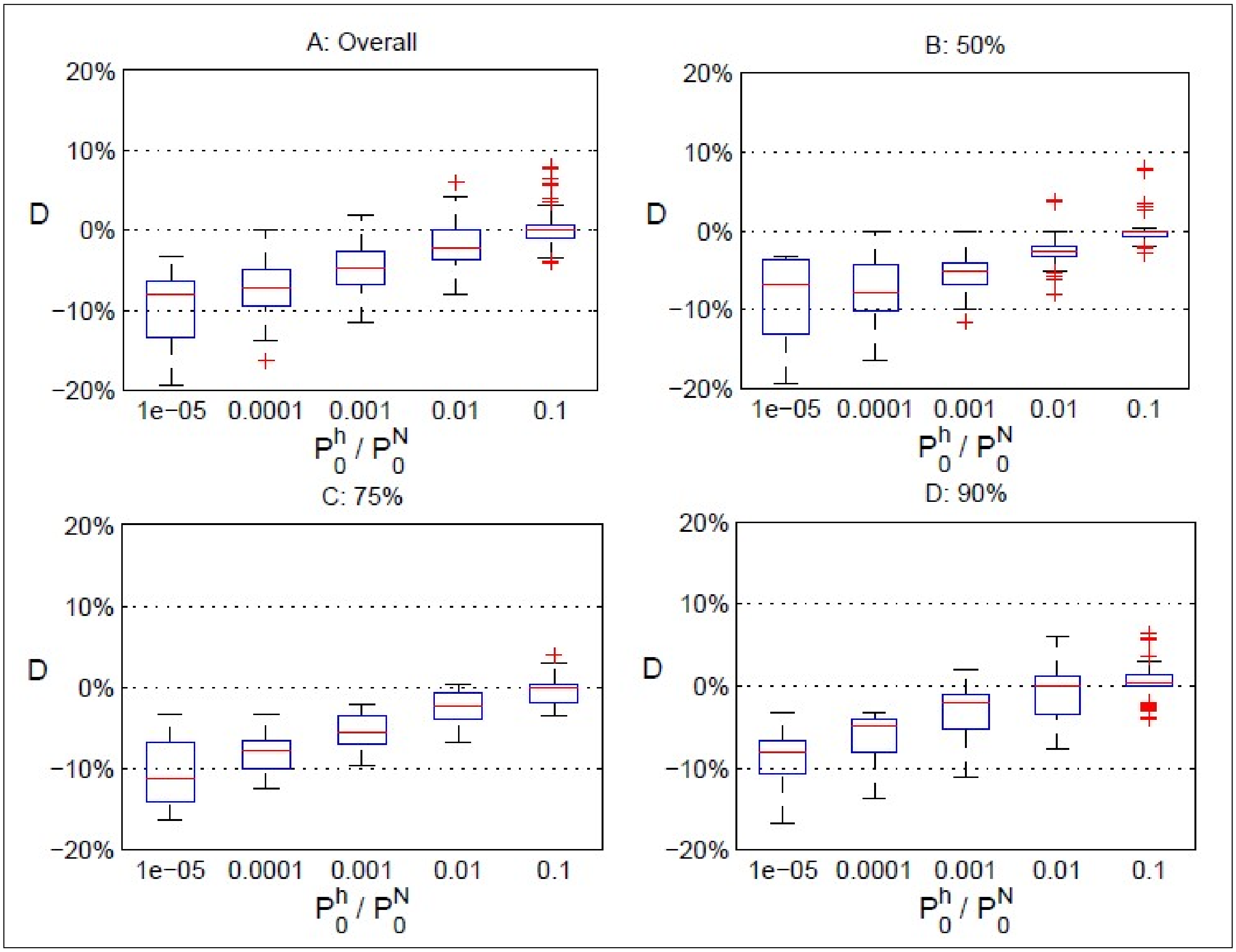


Figure 3.8: Boxplots of relative difference D at different values of P_0^{hb}/P_0^{NL}. A) All data points are included. B) When the *new call* blocking constraint P_0^{Nb} is 50% of P_0^{NL}. C) When the *new call* blocking constraint P_0^{Nb} is 75% of P_0^{NL}. D) When the *new call* blocking constraint P_0^{Nb} is 90% of P_0^{NL}.

Chapter 4

Comparison of Models

We have introduced two new guard channel models: the first guard channel model (the M1 model) and the second guard channel model (the M2 model). Both are controlled preemption models: high priority traffic (handoff calls) can preempt low priority traffic (new calls) on a subset of the total channels (guard channels). It is important to know how these two new models compare to the existing guard channel models: HT's model, a non-preemption model, and the OM model, a full preemption model. In this chapter we use extensive numerical studies to compare the following models:

1. Model(s) with full preemption (MWFP): OM model
2. Model(s) with controlled preemption (MWCP): M1 model, M2 model
3. Model(s) with no preemption (MWNP): HT's model

Note that both MWCP and MWFP can also be considered as model(s) with preemption (MWP).

In this chapter, we compare three characteristics of the models:

1. Their optimal number of channels required to meet a set of pre-determined constraints on call loss.
2. Their new call performances after the total number of channels is fixed and predetermined handoff call blocking constraint is met, and
3. Their ability to meet performance constraints in various traffic environments.

4.1 Optimal number of channels

The optimal number of channels (n^*) is defined as the minimum number of channels required for a model to meet the given constraints on both handoff call and new call performances.

In Section 3.2.4 we compared the optimal number of channels required by the M2 and HT's models. In this section we conduct similar experiments but compare all four models: the M1 model, the M2 model, HT's model, and the OM model. The model parameters and their levels used in the experiments are summarized in Table 4.1.

Table 4.1: Experiment parameters and levels for Experiment 1

Parameter	Level(s)
λ_1	10
μ	1
λ_2	1, 2, 5, 10, 20, 30, 40, 50
P_0^{hb}	10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}
P_0^{NL}	$= k \times P_0^{hb}$ and $k = 1, 5, 10, 10^2, 10^3, 10^4, 10^5$
P_0^{Nb}	$50\% \times P_0^{NL}$, $75\% \times P_0^{NL}$, $90\% \times P_0^{NL}$

For a given set of parameters, n^* for the M1 model can be calculated by solving optimization problem $\mathbf{O}_3$, which was introduced in Section 2.6.2. For the M2 model, n^* can be calculated by solving optimization problem $\mathbf{O}_2$, which was introduced in Section 3.2.4; and for HT's model, n^* can be calculated by the procedures described in Harine et al. [17]. The OM model is a special case of the M1 model, where the number of guard channels g is always equal to the total number of channels n. Therefore, n^* for the OM model can be determined by starting at $n = 1$, searching for the smallest n until all given constraints are met. After repeating these calculations for each model to determine n^* for all possible combinations of parameters, we make the following pairwise comparisons between models:

1. M2 versus M1: We compare the two MWCP to each other;
2. M2 versus OM and M1 versus OM: We compare each MWCP to the MWFP;
3. M2 versus HT's and M1 versus HT's: We compare each MWCP to the MWNP.

For each pairwise comparison and each combination of model parameters, the relative difference (in percentage) D can be calculated as:

$$D = \frac{n_1^* - n_2^*}{n_2^*} \times 100\%, \tag{4.1}$$

where n_1^* and n_2^* stand for the optimal number of channels required for the first and the second model involved in the pairwise comparison, respectively. Therefore, a positive D indicates that the first model requires more channels to meet the given constraints than the second model requires. For example, in the comparison "M2 versus M1", the M2 model is the first model in the comparison and the M1 model is the second model; therefore, in order to calculate the relative difference D, n_1^* is the optimal number of channels for the M2 model, and n_2^* is the optimal number of channels for the M1 model.

The following four parameters/quantities may have an effect on the model performance. They are:

1. Mobility, which is defined as λ_2/λ_1, is the ratio of total handoff call arrival rate to the total new call arrival rate [45]. *High mobility* is when mobility is greater than 1 and *low mobility* is when mobility is less than 1.
2. Constraint for handoff blocking probability, i.e., P_0^{hb}. The smaller the value of P_0^{hb}, the stricter the constraint is.
3. Ratio of constraints, denoted by R, is defined as P_0^{hb}/P_0^{NL}, where P_0^{hb} is the constraint for handoff blocking and P_0^{NL} is constraint for new call loss.
4. P_0^{Nb} percentage, the ratio of new call blocking constraint to new call loss constraint expressed as a percentage, is defined as

$$\frac{P_0^{Nb}}{P_0^{NL}} \times 100\%. \tag{4.2}$$

   We also have $P_0^{NL} = P_0^{Nb} + P_0^{Nd}$. Therefore, when P_0^{NL} (new call loss constraint) is given, a higher P_0^{Nb} (new call blocking constraint) implies a lower P_0^{Nd} (new call dropping constraint), and vice versa. Since dropping an ongoing call is less desirable than blocking an incoming call, we want $P_0^{Nd} \leq P_0^{Nb}$; therefore, P_0^{Nb} percentage is set to be at least 50%.

We compared all models at different mobilities. The results are plotted in Figure 4.1. When comparing the M2 model to the M1 model, the difference between their required optimal number of channels becomes noticeable when mobility is > 1. We also noticed that the M2 model almost always uses fewer channels than the M1 model.

When comparing the M1 and M2 models to HT's model, the difference D in the optimal number of channels required by the models changes as mobility changes. The medians of D are about 5% (for both M1 versus HT and M2 versus HT) at the lowest mobility (0.1) and approach 0% as mobility increases to 5. Also, the difference is more significant at low mobilities (≤ 1) than at high mobilities (> 1).

However, this pattern is reversed when comparing the M1 and M2 models to the OM model. As shown in Figure 4.1, at lower mobilities, there are nearly no differences in the optimal number of channels required between the M1 and OM models (or between the M2 and OM models); because most othen, $D = 0$. However, the results differ at higher mobilities. Also, out data suggest that the M1 and M2 models never require more channels than the OM model. Therefore, we can conclude that the M1 and M2 models use fewer channels than HT's model at low mobility and use fewer channels than the OM model at higher mobility.

To explain the conclusion drawn from Figure 4.1, we should understand how the prioritizing mechanism works for each model. MWP allow new calls to access idle guard channels, while MWNP prohibit such call admission. Therefore, it is intuitively easy to see that MWP (i.e., the M1, M2 and OM models) use channels more efficiently. Such advantage of MWP over MWNP (i.e., HT's model) is more significant at low mobilities, when the offered load of new call is higher than the offered load of handoff call. Because new call traffic dominates handoff call traffic, an incoming new call is more likely to be blocked in MWNP, where there are idle guard channels. Such blocking would not happen in MWP. Therefore, MWNP require more channels than MWP to compensate for inefficiency in channel utilization. As a result, the difference in number of channels required by HT's model versus the M1 or M2 models is more significant at low mobilities: HT's model requires more channels than the M1 or the M2 model. The advantage of allowing new calls to access idle guard channels, however, decreases as mobility increases. This is because when the offered load of handoff call is higher than new calls, the possibility of a guard channel being idle is low; even when a new call finds an idle guard channel, it has a high probability of being preempted by an incoming handoff call and terminated prematurely. Therefore, the difference in the number of channels required by HT's model and the M1 or M2 models is less significant at higher mobilities.

On the other hand, when MWCP (the M1 and M2 models) are compared to the MWFP (the OM model), the difference is more significant (where the OM model uses more channels) at higher mobilities than at lower mobilities. When the offered load of handoff call is higher than that of new calls, the effect of full preemption on new call performance is more substantial because new calls are more likely to get preempted by handoff calls; therefore, to meet the performance constraints (in particular, the constraint on call dropping) for new calls, the OM model requires more channels than the M1 and M2 models. This effect diminishes as mobility decreases, until the difference between the models is negligible.

In Figure 4.2, models are compared at different ratios of constraints, R. We first examined the difference between the M1 and M2 models. The first boxplot in Figure 4.2 suggests that the M2 model uses fewer channels when R is greater than 10^{-4}, especially when R is equal to 0.1 and 0.2 (which implies that P_0^{hb} is more similar to P_0^{NL}). This is because with the M2 model, when P_0^{hb} and P_0^{NL} have the same order of magnitude, most of the channels are set up as normal channels (i.e., non-guard channels), which are fully shared by both types of traffic. At the same time only a few channels are guard channels to ensure that the constraint for handoff calls, which is a little stricter than the constraint for new calls, can also be met. Such set up is more efficient than the set up in the M1 model, where the non-guard channels can only be accessed by new calls. However, when R is small and P_0^{hb} is several orders of magnitude less than P_0^{NL}, most channels in both the M1 and M2 models are guard channels and therefore, the difference between them is negligible.

We also compared the M1 and M2 models to HT's model. The patterns shown in the fourth and fifth boxplots in Figure 4.2 suggest that at low values of R, the M1 and M2 models use significantly fewer (10% to 20%) channels than HT's model. This is expected because the goal of preemption is to prioritize handoff calls. MWP use channels more efficiently when the constraint for handoff call blocking is much stricter than the constraint for new call loss. However, the value of relative difference, D, approaches zero as R increases, indicating that MWCP and MWNP do not significantly differ when similar constraints are adopted for handoff calls and new calls.

As shown in Figure 4.2, when the M1 and M2 models are compared to the OM model, the difference in optimal number of channels required is minimal for small values of R but is more significant for larger values of R. Because the MWFP (the OM model) priorities low handoff call blocking over performance of new calls, it functions best when the constraint on handoff call blocking is several orders of magnitude stricter than constraint for new call loss. However, when the constraints for handoff call blocking and new call loss are comparable, the MWFP cannot control the new call loss by adjusting the number of guard channels. Because of this inflexibility, MWFP has to use more channels than MWCP in order to simultaneously meet the similar constraints on handoff call blocking and new call loss.

As shown in Figure 4.3, when all models are compared at different constraints for handoff call blocking (i.e., P_0^{hb}), a similar pattern merges: when P_0^{hb} is strict ($< 10^{-3}$), MWCP (the M1 and M2 models) use fewer channels than MWNP (HT's model). This advantage decreases as P_0^{hb} increases, and the HT's requires less channel than the M1 model when $P_0^{hb} \geq 10^{-2}$.

In Figure 4.4, models are compared at different P_0^{Nb} percentages. In the first plot, the M1 and M2 models are compared, and the boxplot shows that the M2 model uses fewer channels at higher P_0^{Nb} percentages (75% and 90%). When comparing the M1 and M2 models to HT's model, the median of relative differences D with respect to the HT's model is below zero (indicating that MWCP use fewer channels than MWNP) when P_0^{Nb} percentage is at 50% (i.e., $P_0^{Nb} = P_0^{Nd}$) and increases as P_0^{Nb} percentage approaches 90%. When comparing the M1 and M2 models to the OM model, the values of relative differences D with respect to the OM model (as shown in the second and third boxplots) are always below zero, indicating that MWCP always require few channels than MWFP. These patterns are closely related to the principle of preemption. The performance of high-priority traffic (i.e., handoff call) is guaranteed by allowing high-priority traffic to preempt low-priority traffic (i.e., new call); therefore, because of the increased number of new call dropping events due to preemption, the performance of low-priority traffic is deteriorated. When we impose a separate performance constraint on new call dropping, the advantage of MWP (OM, M1 and M2 models) over MWNP (HT's model) decreases as the constraint becomes stricter (i.e., when P_0^{Nb} percentage increases). Similarly, MWCP (M1 and M2 models) also outperform full MWFP (OM model) at higher P_0^{Nb} percentage.

4.2 New call performance

In the second experiment we compare the new call loss probability of different guard channel models. Three models (HT's model, the M1 model, and the M2 model) are compared with each other by the method outlined below.

1. We consider a reference cell containing $n = 50$ channels, and
2. for a given constraint on handoff call blocking, we calculate the optimal number of guard channels required for each model in order to meet this constraint (i.e., solving the optimization problem $\mathbf{O}_1$).
3. We then use the number of optimal guard channels calculated in Step 2 to calculate the new call loss probability between different models.

Note that the OM model is excluded from this comparison because when the total number of channels n is fixed, the new call loss probability (P_L^N) for the OM model will always be less than or equal to that of the M1 model. This is because the OM model is a special case of the M1 model where all the channels are guard channels. When the number of guard channels is less than the number of total channels (i.e., $g < n$, the more general case of the M1 model), P_b^N and P_d^N will both decrease; therefore, provided that n is fixed, the M1 model always has fewer new call losses than the OM model. Accordingly, we choose not to include the OM model in our comparison.

The new call performance are compared under two considerations: (1) There is no additional penalty for new call droppings, and (2) There is an additional penalty imposed on new call droppings. In the first case, the new call dropping and blocking are treated equally, that is, the new call loss probability can be calculated as the sum of new call blocking and new call dropping probabilities:

$$P_L^N = P_b^N + P_d^N.$$

In the second case, however, the new call dropping is more serious than the new call blocking as an additional penalty is imposed on new call dropping. The calculation of new call loss probability for this case will be introduced in Section 4.2.

When there is no additional penalty for new call dropping

The values of model parameters used in this experiment are summarized in Table 4.2.

Table 4.2: Experiment parameters and levels for Experiment 2-1

Parameter	Level(s)
n	50
λ_1	Vary from 10 to 100
μ	1
Mobility	0.1, 0.5,1, 2, 3
P_0^{hb}	10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}

The following pairwise comparisons between models will be carried out in this experiment:

1. M2 versus M1: the two MWCP are compared with each other;
2. M2 versus HT's and M1 versus HT's: each MWCP is compared to MWNP.

For each of the pairwise comparison, the relative difference in new call loss probability with respect to the second model involved in the comparison (i.e., the model being compared to), denoted by L, is calculated for each combination of model parameter as:

$$L = \frac{P_{L1}^N - P_{L2}^N}{P_{L2}^N} \times 100\%, \tag{4.3}$$

where P_{L1}^N is the new call loss probability of the first model involved in the comparison and P_{L2}^N is the new call loss probability of the second model involved in the comparison. As we can see, a negative value of L indicates that the new call loss probability of the first model is lower than that of the second model.

Figure 4.5 shows the results of comparing models at different levels of mobility. When comparing the M2 model to the M1 model, mobility does not appear to affect the relative performance of new calls in an obvious pattern. We found that $L = 0$, indicating that two models perform similarly, occurs at most parameter combinations at lower mobility, and $L < 0$, indicating that the new call loss probability for the M2 model is less than the new call loss probability for the M1 model, occurs at higher mobility. When the M1 and M2 models are compared to HT's model, however, an obvious pattern merges in both the second and third boxplots: The new call loss of MWCP (the M1 and M2 models) is much lower (up to 100% lower) than that of MWNP (HT's model), especially at higher mobility. Also the new call loss of the M2 model is always less than or equal to that of HT's model. The dramatic improvement in new call performance when using MWCP is as expected, because these models allow new calls to use idle guard channels. When the total number of channels n is fixed, as mobility increases, so does the number of guard channels required to meet the constraint on handoff call blocking. Therefore, the use of MWCP significantly improves new call performance (that is, decreases new call loss probability).

Figure 4.6 presents the comparison results at different constraints of handoff call blocking, P_0^{hb}. Again, the difference in new call loss between the M1 and M2 models is insignificant (refer to the first boxplot). However, when the M1 and M2 models are compared to HT's model, a trend clearly emerges in the second and third boxplots: The medians of L are always negative and increase as P_0^{hb} increases, indicating that MWCP (the M1 and M2 models) produce fewer new call losses than does MWNP (HT's model). MWCP reduces new call loss even more substantially (up to 100%) when the constraint for handoff call blocking is stricter (i.e., when $P_0^{hb} < 0.01$); however, this effect diminishes as P_0^{hb} increases, and it becomes negligible when $P_0^{hb} = 0.01$. The reason for this trend is that MWCP use channels more efficiently then does MWNP: MWCP allow new calls to access idle guard channels to reduce new call blocking probability. This efficiency increases as the number of guard channels increases. When stricter handoff blocking constraint is applied, more guard channels are required for models to meet this constraint, and as a result, substantial reductions in new call loss is likely to be observed for MWCP.

When there is an additional penalty for new call dropping

In the previous section, we calculate the new call loss probability as:

$$P_L^N = P_b^N + P_d^N.$$

This is a special case of the Grade of Service (GoS) cost function, where there is no additional penalty for dropping a call. Although sophisticated cost functions for new calls have been proposed in Barcelo [4], in practice, a simple weighted average is useful for most design purpose. Such a function should reflect the penalty of the call dropping over the call blocking probability [45]. The GoS of new call can then be defined as:

$$\text{GoS of new calls} = P_b^N + W \times P_d^N, \tag{4.4}$$

where $W \geq 1$ and can be considered as the additional penalty imposed on new call dropping. Note that smaller values of GoS are associated with better new call performance, and larger values of GoS are associated with worse new call performance. In the previous experiment we set $W = 1$, i.e., no additional penalty for new call dropping. In the following experiment we want to impose a penalty for new call dropping (i.e., $W > 1$) to investigate how different values of the penalty affect the GoS of new calls for different models.

The traffic parameters used in this experiment are taken from Salamah [45]. The total number of channels is fixed to be 50. Constraint on handoff call blocking, P_0^{hb}, can vary from 10^{-5} to 10^{-2} (see Table 4.3). Given a set of traffic parameters, a penalty W and a constraint on handoff call blocking P_0^{hb}, one can obtain the number of guard channels required by each model to meet P_0^{hb} by solving optimization problem $\mathbf{O}_1$. Equation 4.4 can then be used to obtain each model's GoS of new calls.

Table 4.3: Experiment parameters and levels for Experiment 2-2

Parameter	Level(s)
n	50
μ	1/180
Total offered load[1]	$0.6 \times n = 30$(moderate to high load)
Mobility	0.5(low), 1(moderate), 2(high)
P_0^{hb}	10^{-2},10^{-3}, 10^{-4}, 10^{-5}

[1] The total offered load is defined as $\lambda_1/\mu + \lambda_2/\mu$

In Figure 4.7 through 4.10, GoS of new calls is plotted against penalty W under four different constraints for handoff call blocking. In each plot, the GoS of new calls for different models are compared at different mobilities. For MWNP (HT's model), since the new call dropping probability is zero (because preemption is not allowed), its GoS is not affected by different values of W and therefore stays constant against W. On the other hand, for MWCP (the M1 and M2 models), their GoS curves in the plot are so close to each other that they almost always overlap. Furthermore, the GoS curves of the M1 and M2 models increase with W; and when $W = 1$, they start below the corresponding GoS curve of HT's model, indicating that the new call performance of MWCP is better than that of MWNP. This confirms our results from the previous experiment, that when there is no additional penalty imposed on new call dropping, MWCP outperform MWNP in new call performance.

Looking at each plot and studying the effect of mobility on the comparison results between MWCP and MWNP, it is interesting to know at what penalty the new call performance of MWCP starts to become worse than that of MWNP, i.e., at what value of W the GoS curves of MWCP surpass the corresponding GoS curve of MWNP. It is apparent that the GoS curves of MWCP are more likely to surpass the corresponding GoS curve of HT's model at low penalties when mobility is low, or at high penalties when mobility is high. For example, in Figure 4.9 the constraint for handoff call blocking is 10^{-4}. Notice that the GoS curves of MWCP (square dotted and square dashed lines) at mobility 0.5 surpass the corresponding GoS curve of MWNP model (square solid line) at penalty 6; however, when mobility is 2, the GoS curves of MWCP (asterisk dotted and asterisk square lines) do not surpass the corresponding GoS curve of MWNP (asterisk solid line), not even at penalty 10. Similar patterns can also be observed when we study the effect of P_0^{hb} across all four figures: the GoS curves of MWCP are more likely to surpass those of MWNP at low penalty when P_0^{hb} is loose, or at high penalty when P_0^{hb} is strict. Note that when P_0^{hb} is 10^{-5}, the GoS curves of MWCP never surpass those of MWNP (within the area of Figure 4.10) at both low (i.e., 0.5) and moderate (i.e., 1) mobilities. To conclude, MWCP (the M1 and M2 models) can tolerate high penalty on new call dropping and still outperform MWNP in new call performance when the mobility is high and/or when the constraint on handoff call blocking is strict.

4.3 Ability to meet constraints

Another topic that we explored was how the different models adapt to various traffic conditions (i.e., various combinations of λ_1, μ, and mobility) and how they meet predetermined constraints on call loss. In the next experiment, we use numerical examples to compare the abilities of each model to meet constraints on call loss. The model parameters and their levels used in this experiment are summarized in Table 4.4.

Table 4.4: Experiment parameters and levels for Experiment 3

Parameter	Level(s)
n	50
λ_1	vary from 1 to 50
μ	1
Mobility	0.1, 0.5, 1, 2, 3, 4, 5
P_0^{hb}	10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}
P_0^{NL}	$= k \times P_0^{hb}$ and $k = 2, 5, 10, 50, 10^2, 10^3, 10^4, 10^5$

For each of the parameter combinations, we examine each of the four models and determine if the model can meet a given set of constraints on call loss (P_0^{hb} and P_0^{NL}). Note that we do not set separate constraints for new call dropping and blocking probabilities. We only set one constraint for the total new call loss. The total new call loss is simply the sum of new call blocking and new call dropping (i.e., no penalty is imposed on call dropping). The following method is used to determine whether a certain model can meet given constraints:

1. For HT's model, the M1 model, and the M2 model, we set $n = 50$ and search exhaustively for the smallest number of guard channels g ($\leq n$) such that both the given constraints are met. If such g can be found, then the given constraints can be met; otherwise the constraints cannot be met when there are only 50 channels.
2. For the OM model, since all the channels are guard channels, we only need to set $n = 50$ and check whether all the given constraints can be met.

We summarized the results by calculating the constraint met percentage for each model, which is defined as the percentage of the total sets of constraints that can be met by each model. Figure 4.11 to 4.15 plot constraint met percentages against λ_1, mobility, P_0^{hb}, P_0^{NL}, and P_0^{NL}/P_0^{hb}. Note that we only recorded where the total offered load is less than or equal to n; this is because when the total offered load is greater than n, the systems are overloaded and all models performed equally poorly in meeting given call loss constraints.

In Figure 4.11, constraint met percentages for each model are plotted against different values of new call arrival rate λ_1. The trend exhibited in the figure is as expected: the constraint met percentage decreases as λ_1 increases for all models. This is because increases in the new call arrival rate cause an increase in total offered load, making it difficult for models to meet given constraints. As shown in Figure 4.11, the three MWP (the M1 model, the M2 model, and the OM model) have almost the same performance, and the performance of the MWNP (HT's model) is close to those of the MWP at smaller λ_1 but worsens as λ_1 increases.

In Figure 4.12, constraint met percentages of each models are plotted against mobility. As we can see, the constraint met percentages are high at low mobility and decrease as mobility increases. The explanation is straightforward: high mobility implies high handoff call offered load (when call service time is holding constant), and therefore only relatively low constraint met percentage can be achieved. When we look at individual models, the M1 and the M2 models are slightly better than the OM model. The constraint met percentage of HT's model is about 2% to 4% less than those of the M1 and M2 models.

Figures 4.13 and 4.14 illustrate the effects of constraints for handoff blocking (P_0^{hb}) and new call loss (P_0^{NL}) on constraint met percentages. Stricter constraints correspond to lower constraint met percentage (as low as 45% to 55%), and looser constraints correspond to higher constraint met percentage (up to 85% to 95%). When we compare the performance between models, Figure 4.13 presents an interesting pattern. The gap between MWP and HT's model is about 5% at extremely strict P_0^{hb} (i.e., 10^{-6}), and the gap closes when P_0^{hb} becomes loose.

Finally, Figure 4.15 plots the constraint met percentages against P_0^{hb}/P_0^{NL}, the ratio of constraints defined in Section 4.1. The pattern exhibited in this plot suggests that for all models, the constraint met percentages are higher when P_0^{hb} is several orders less in magnitude than P_0^{NL} (i.e., when P_0^{hb}/P_0^{NL} is close to zero), and decrease to about 60% when both P_0^{NL} and P_0^{hb} have the same order of magnitude ($P_0^{hb}/P_0^{NL} = 0.5$). Also, when $P_0^{hb}/P_0^{NL} \leq 10^{-2}$, the difference in constraint met percentages between MWP and MWNP is more significant (i.e., MWNP meets about 10% fewer constraints than MWP).

To conclude, MWP always outperform MWNP (HT's model) in meeting constraints under various traffic conditions. Among the three MWP, MWCP (the M1 and M2 models) perform slightly better than MWFP (the OM model).

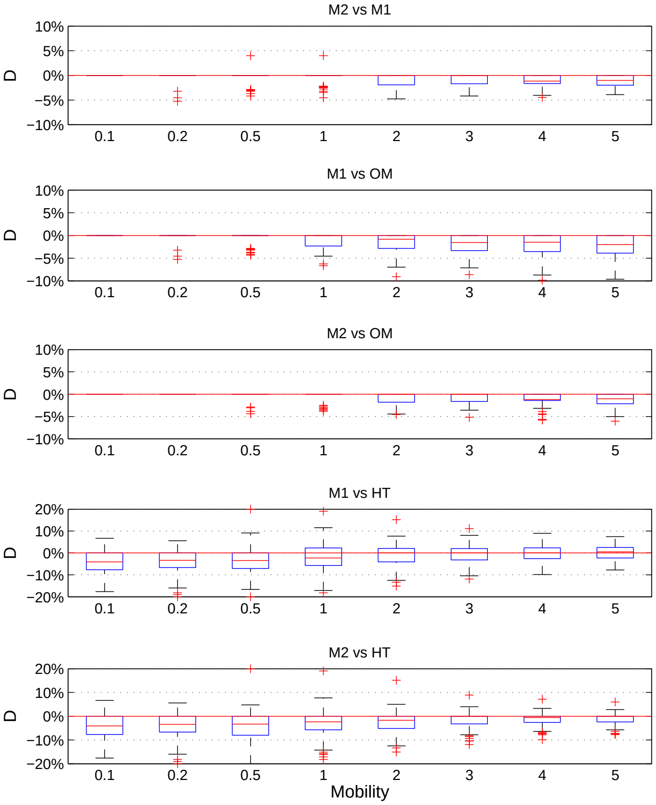


Figure 4.1: Experiment 1: Boxplots of relative differences. Models are compared at different mobilities

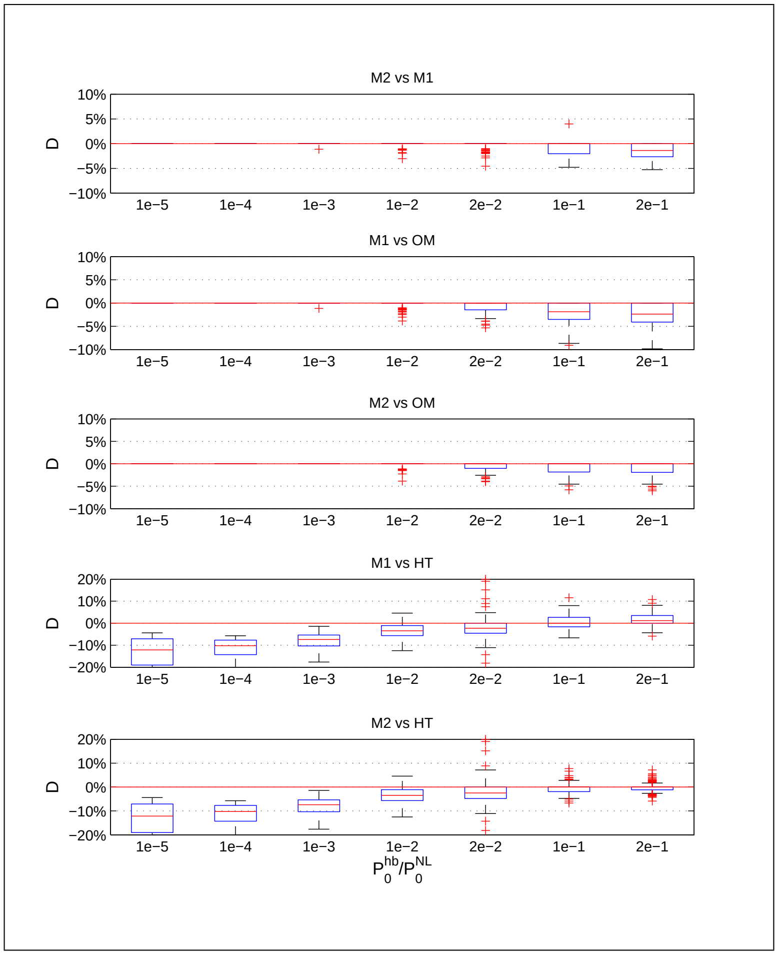


Figure 4.2: Experiment 1: Boxplots of relative differences. Models are compared at different ratios of constraints.

4.4 Conclusions

In this chapter, we divided four guard channel models into three groups: MWNP (HT's model), MWCP (the M1 and M2 models) and MWFP (the OM model). We compared three characteristics of these groups: (1) their optimal number of channels required to meet a set of pre-determined constraints on call loss, (2) new call performances after the total number of channels is fixed and predetermined handoff call blocking constraint is met, and (3) their ability to meet performance constraints in various traffic environments. From our extensive numerical experiments, we conclude that MWCP (the M1 and M2 models) are overall the best models for maximizing channel utilization and balancing the performance between high-priority and low-priority traffic. When compared to MWNP (HT's model), MWCP are

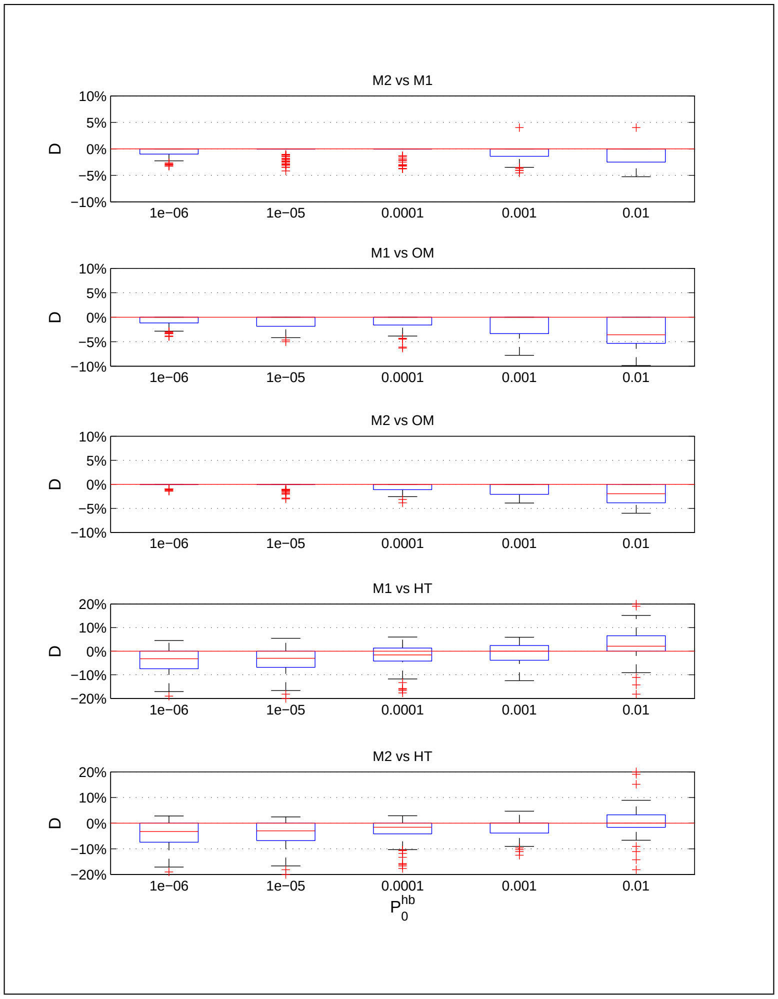


Figure 4.3: Experiment 1: Boxplots of relative differences. Models are compared at different handoff call blocking constraints.

more efficient in channel utilization and, therefore, often produce much lower new call loss, especially when the mobility is high or the when constraint on handoff call blocking is strict. When compared to MWFP (the OM model), MWCP also exhibit the advantages of efficient channel utilization because they can adjust the number of guard channels according to the given offered load and the given constraints on call losses. Our conclusion are well supported by the first experiment in which MWCP almost always required fewer channels than MWFP. Such flexibility also helps MWCP to achieve lower new call loss probability than MWFP. Finally, when the two MWCP, the M1 and M2 models, are compared to each other, the M2 model slightly outperformed the M1 model in most of the experiments, but the difference is not significant.

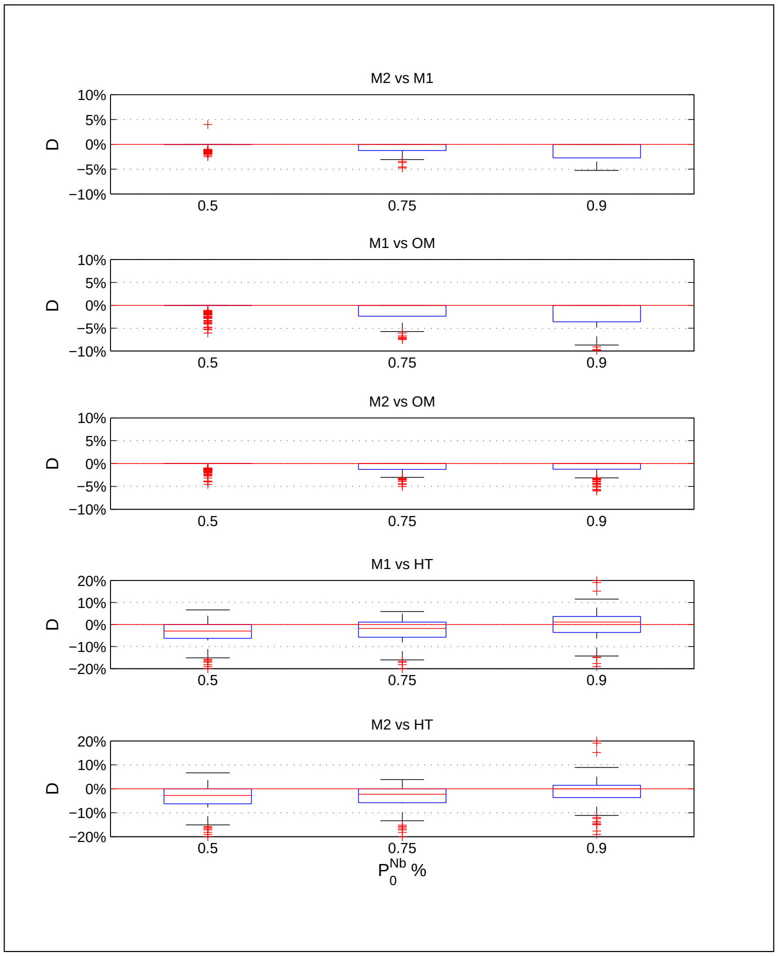


Figure 4.4: Experiment 1: Boxplots of relative differences. Models are compared at different P_0^{Nb} percentages.

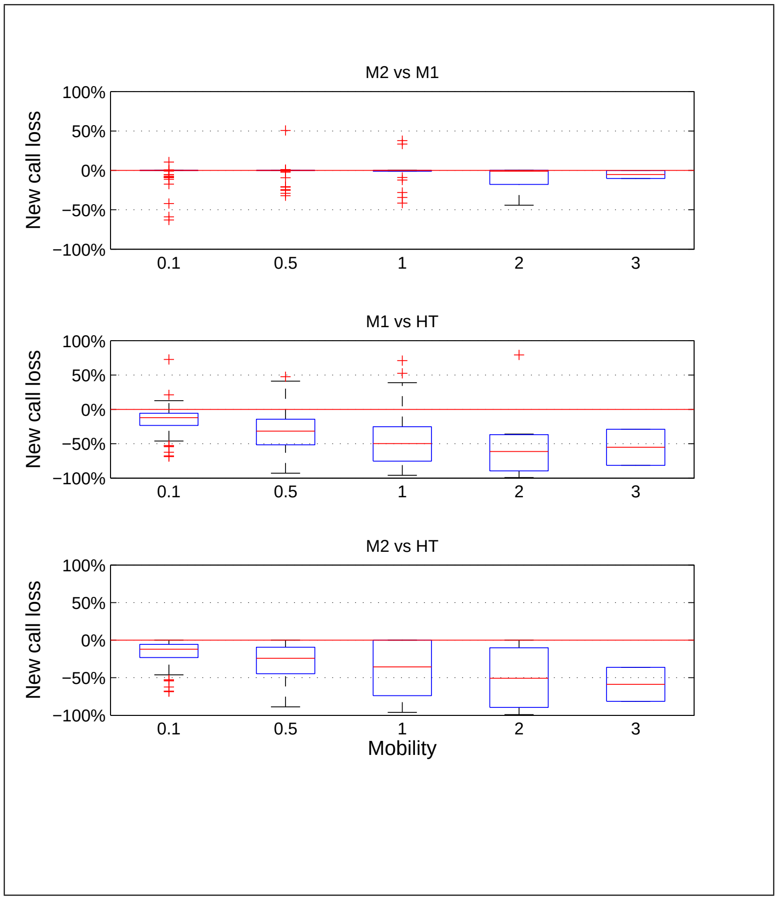


Figure 4.5: Experiment 2-1: Boxplots of relative differences. Models are compared at different mobilities.

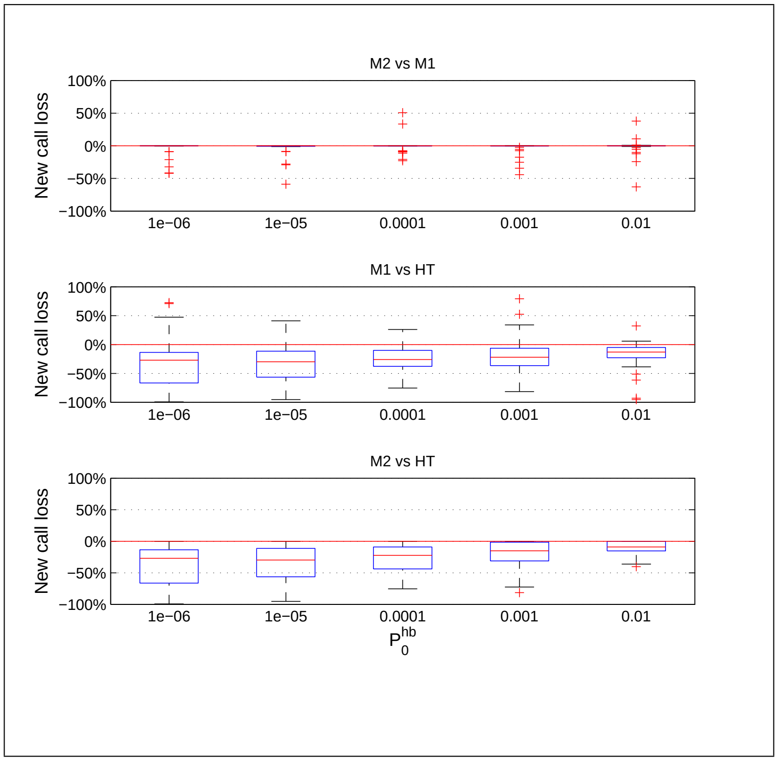


Figure 4.6: Experiment 2-1: Boxplots of relative differences. Models are compared at different handoff call blocking constraints.

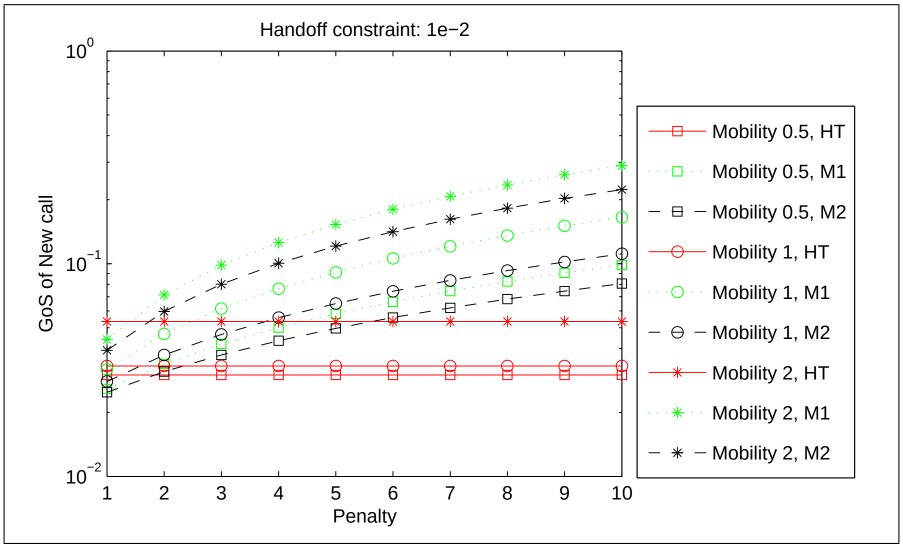


Figure 4.7: Experiment 2-2: GoS of new calls for different models when $P_0^{hb} = 10^{-2}$.

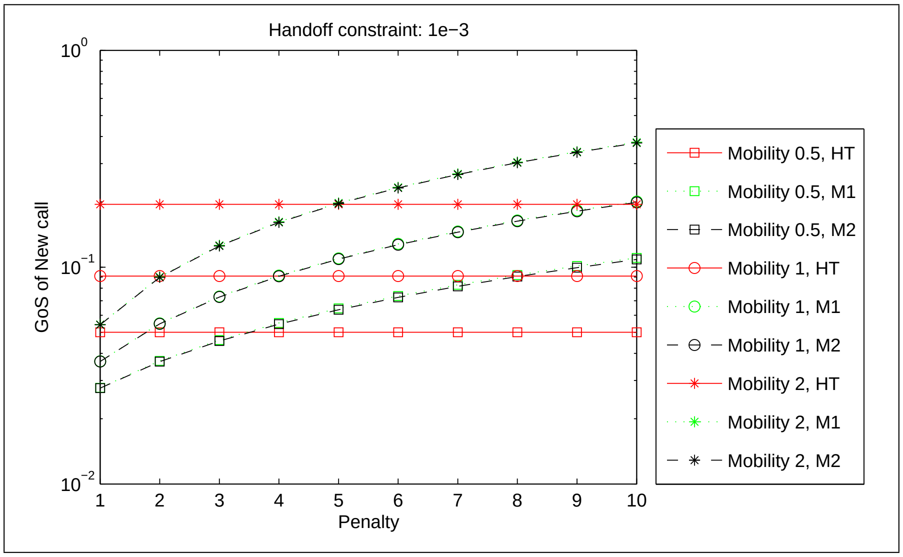


Figure 4.8: Experiment 2-2: GoS of new calls for different models when $P_0^{hb} = 10^{-3}$.

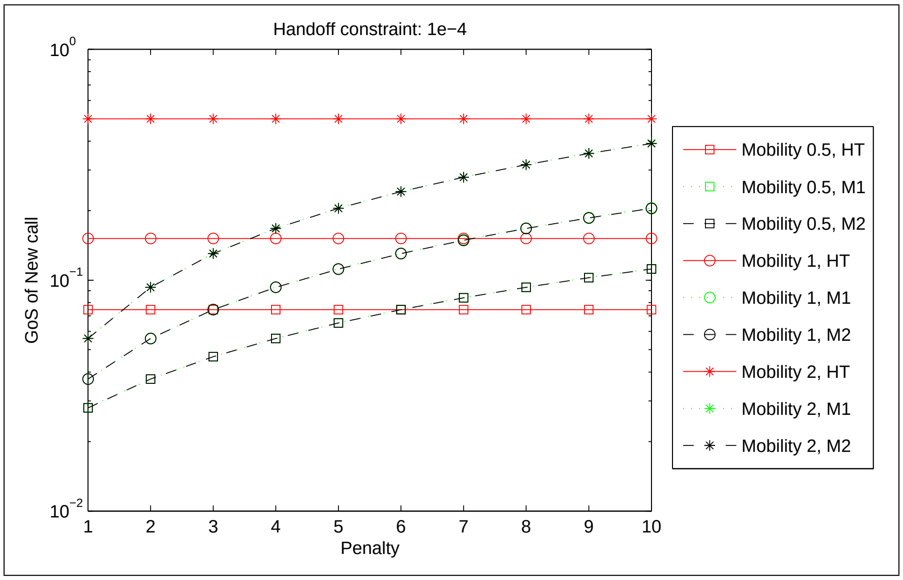


Figure 4.9: Experiment 2-2: GoS of new calls for different models when $P_0^{hb} = 10^{-4}$.

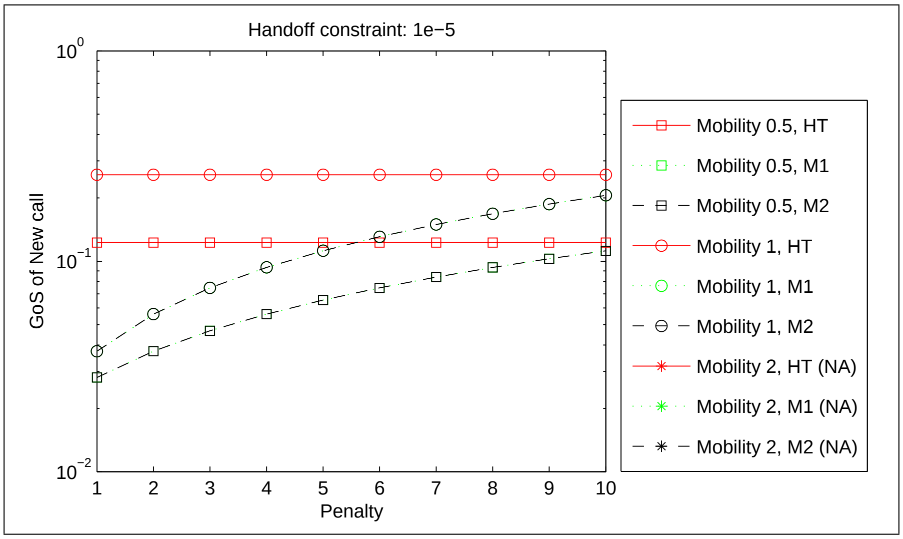


Figure 4.10: Experiment 2-2: GoS of new calls for different models when $P_0^{hb} = 10^{-5}$.

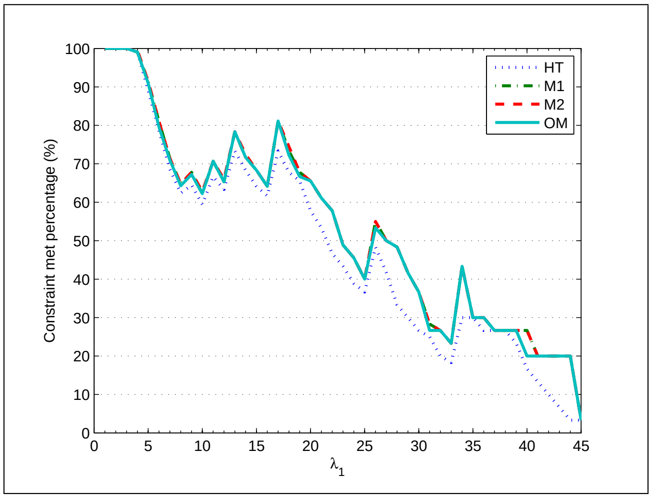


Figure 4.11: Experiment 3: Plot of constraint met percentage against λ_1.

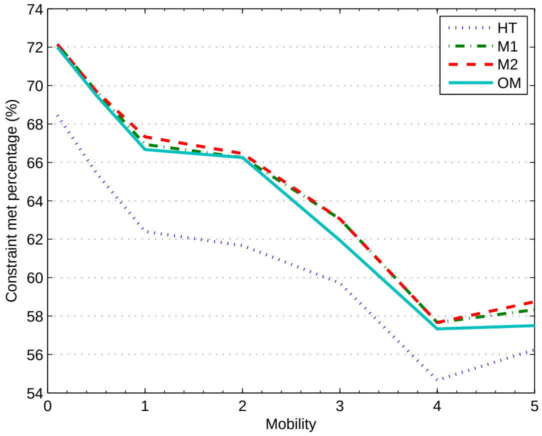


Figure 4.12: Experiment 3: Plot of constraint met percentage against mobility.

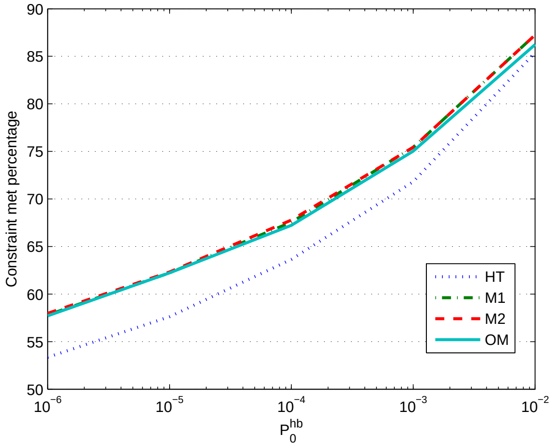


Figure 4.13: Experiment 3: Plot of constraint met percentage against handoff call blocking constraint, P_0^{hb}.

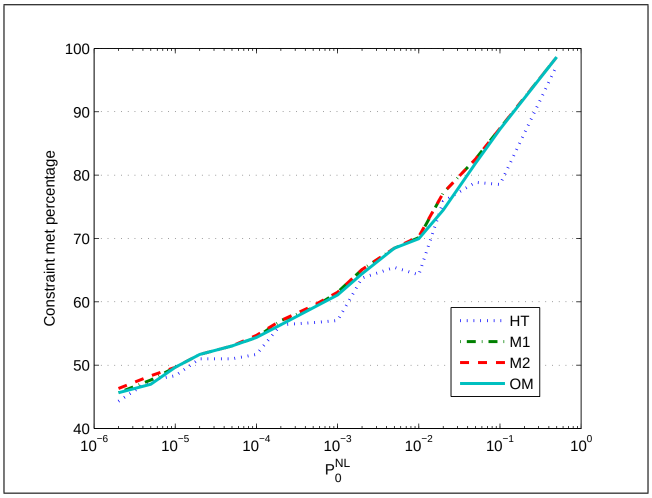


Figure 4.14: Experiment 3: Plot of constraint met percentage against new call loss constraint, P_0^{NL}.

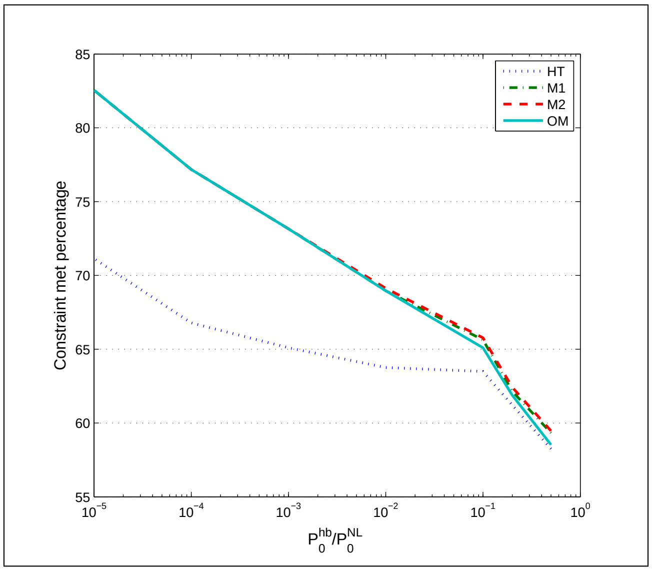


Figure 4.15: Experiment 3: Plot of constraint met percentage against ratio of constraints, P_0^{hb}/P_0^{NL}.

Chapter 5

The $M/M/{\sim}C/{\sim}C$ Queueing System and Markov Regenerative Process

5.1 Introduction

In most systems, especially in traditional call centres, the server capacity is a constant value over time. However, in cellular communication the system capacity usually can vary over time due to many complicated factors in the networks, such as channel failure, frequency borrowing, competition among different classes of traffic, channel preemption and so forth. In the guard channel models analyzed in the previous chapters, the system capacity for the low priority traffic (new calls) varies due to the existence of the high priority traffic (handoff calls). The performance metrics, that is, the blocking and dropping probabilities could be calculated by numerical or analytic methods. In this chapter we analyze the more general case of systems with stochastic capacity: the $M/M/{\sim}C/{\sim}C$ queueing system, which was first studied by Sun et al. [52] and then by Luo and Williamson [32]. In both cases the literature show that the Markov regenerative process (MRGP) can be used to model the $M/M/{\sim}C/{\sim}C$ system where the capacity variation process is a skip-free process; but no explicit formulae for calculating the dropping probability are developed. In [32], three different distributions of capacity interchange times are considered but no explicit formulae are presented for calculating the steady state probabilities of the MRGP under different distributions. In this chapter, we first review the theory of MRGP. Then, as an illustration, the MRGP method is applied to the M1 model where $n = 1$ and $g = 1$. Finally, the MRGP method is applied to the $M/M/{\sim}C/{\sim}C$ queueing system where three different distributions of capacity interchange times (exponential, Pareto and gamma) and three capacity variation types (skip-free, uniform-based, and distance-based) are discussed. Explicit formulae are derived for calculating the steady state probabilities as well as the dropping probability. The analytic solutions using the derived explicit formulae are verified by simulation results.

5.2 Review of Markov regenerative theory

Markov regenerative theory is used to analyze Semi-Markovian queueing systems. This type of queueing systems is characterized by having an MRGP as its queue length process, which indicates that the sojourn time of each state is not necessarily exponentially distributed. In this section, the theory of MRGP is briefly reviewed.

Definitions and theorems in this section follow Kulkarni (2010). Consider a stochastic process wherein there exist time points where the process satisfies the Markov property. These time points are referred to as regeneration points. In an MRGP the stochastic evolution between two successive regeneration points depends only on the state at regeneration, not on the evolution before regeneration. Furthermore, due to the time homogeneity of the embedded Markov renewal process, the evolution of the MRGP becomes a probabilistic replica after each regeneration. As a consequence, all memory other than the state must be reset at a regeneration point. The concepts of MRGP are given below.

Definition 2 *Markov renewal sequence. A sequence of bivariate random variables* $\{(Y_n, S_n), n \geq 0\}$ *is called a Markov renewal sequence if*

(i) $S_0 = 0, S_{n+1} \geq S_n; Y_n \in \{0, 1, 2, ...\}$ and

(ii) for all $n \geq 0$,

$$\begin{aligned}
&P\{Y_{n+1} = j, S_{n+1} - S_n \leq t | Y_n = i, S_n, ..., Y_0, S_0\} \\
=&P\{Y_{n+1} = j, S_{n+1} - S_n \leq t | Y_n = i\} \\
&\text{(Markov property)} \\
=&P\{Y_1 = j, S_1 \leq t | Y_0 = i\} \\
&\text{(Time homogeneity).}
\end{aligned}$$

Definition 3 *Semi-Markov process. Let* $\{(Y_n, S_n), n \geq 0\}$ *be a Markov renewal sequence and* $N(t) = \sup\{n \geq 0 : S_n \leq t\}$*. Let*

$$X(t) = Y_{N(t)}, \quad t \geq 0.$$

The stochastic process $\{X(t), t \geq 0\}$ *is called a semi-Markov process (SMP).*

Definition 4 *Markov regenerative process. A stochastic process* $\{Z(t), t \geq 0\}$ *on its discrete state space,* Ω*, is called an MRGP if there exists a Markov renewal sequence* $\{(Y_n, S_n), n \geq 0\}$ *of random variables such that all conditional finite dimensional distributions of* $\{Z(S_n+t), t \geq 0\}$ *given* $\{Z(u), 0 \leq u \leq S_n, Y_n = i\}$ *are the same as those of* $\{Z(t), t \geq 0\}$ *given* $Y_0 = i$*,* $i \in \Omega' \subset \Omega$.

Note that the above definition implies that in this case $\{Z(S_n^+), n \geq 0\}$ or $\{Z(S_n^-), n \geq 0\}$ is an embedded discrete time Markov chain (DTMC) or just the embedded Markov chain in $\{Z(t), t \geq 0\}$, and also that S_n is a stopping time (regeneration points) of $\{Z(t), t \geq 0\}$. As a special case, the definition implies that

$$\begin{aligned} &P\{Z(S_n + t) = j | Z(u), 0 \leq u \leq S_n, Y_n = i\} \\ =&P\{Z(t) = j | Y_0 = i\}. \end{aligned}$$

We denote the conditional probability $P\{Y_1 = j, S_1 \leq t | Y_0 = i\}$ by $K_{ij}(t)$, $i, j \in \Omega'$. The matrix $\mathbf{K}(t) = [K_{ij}(t)]$ is called the *global kernel* of the Markov renewal sequence. Define $E_{i,j}(t)$, where $i, j \in \{0, 1, 2, ...\}$, as follows:

$$E_{i,j}(t) = P\{Z(t) = j, S_1 > t | Y_0 = i\}.$$

Then the matrix $\mathbf{E}(t) = [E_{i,j}(t)]$ describes the behavior of the MRGP between two transition epochs of the embedded Markov chain, that is, over the time interval $[0, S_1)$. We call the matrix $\mathbf{E}(t)$ the *local kernel.*

To study the limiting behavior of the MRGP, we need to define three variables:

- Vector $\mathbf{m}$, where $m_i = E(S_1 | Y_0 = i)$, is the mean time the embedded Markov chain spends on state i.

- $\alpha_{i,j} = E[\text{time spends by the system in state } j \text{ during } (0, S_1)|Y_0 = i] = \int_0^\infty E_{i,j}(t)dt$ is the mean time the MRGP spends in state j between two successive regeneration instants, given that it stayed in state i after the last regeneration. Note that $m_i = \sum_j \alpha_{i,j}$.
- $\mathbf{v}$ is the steady state probability vector of the embedded Markov chain:

$$\mathbf{v} = \mathbf{vP}, \quad \sum_{k \in \Omega'} v_k = 1 \tag{5.1}$$

where $\mathbf{P} = \mathbf{K}(\infty)$ is the one-step transition probability matrix of the embedded Markov chain.

The following theorem describes the limiting behavior of MRGPs:

Theorem 5 *Let $\{Z(t), t \geq 0\}$ be an MRGP on Ω with Markov renewal sequence $\{(Y_n, S_n), n \geq 0\}$ with kernel $\boldsymbol{K}(\cdot)$. Let $N(t)$ denotes the total number of state changes by time t, i.e. $N(t) = \sup\{n \geq 0 : S_n \leq t\}$. Suppose that*

(i) the sample paths of $\{Z(t), t \geq 0\}$ are right continuous with left limits,

(ii) the SMP $\{Y_{N(t)}, t \geq 0\}$ is irreducible, aperiodic, and positive recurrent,

(iii) v is the positive solution to Equation 5.1.

Then the steady state probability of the MRGP is given by

$$\pi_j = \lim_{t \to \infty} P\{Z(t) = j\} = \frac{1}{\mathbf{vm}} \sum_{k \in \Omega'} v_k \alpha_{k,j} \tag{5.2}$$

where $\mathbf{vm} = \sum_{i \in \Omega'} v_i m_i$.

For more details and examples see Kulkarni [27].

5.3 Application to the first guard channel model

As illustrated in this section, the theory of MRGP was used to analyze the M1 model presented in Chapter 2. This model was re-examined as a composition of a series of traffic models and a capacity model. The traffic model focuses on activities of new calls and the capacity model, on the other hand, accounts for the capacity fluctuation to new calls caused by handoff call. Note that the M1 model serves as a special case where every event is Markovian. The use of MRGP to model non-Markovian queueing systems are presented in later sections.

5.3.1 General procedure

Traffic models

We use $N(t)$ to denote the number of new calls in the system at time t (or, in other words, the system occupancy of new calls at time t). A new call arrives according to a Poisson process with rate λ_1 and spends an amount of time in the system according to an exponential distribution with rate μ_1. Given that the system capacity for new calls is i, the stochastic process $\{N(t), t \geq 0\}$ is a homogeneous continuous time Markov chain and can be solved as an $M/M/i/i$ queueing system. The steady state probabilities are given by

$$P(N = k) = \frac{\left(\frac{\lambda_1}{\mu_1}\right)^k \Big/ k!}{\sum_{l=0}^{i} \left(\frac{\lambda_1}{\mu_1}\right)^l \Big/ l!}, \quad k = 0, 1, 2, ..., i. \tag{5.3}$$

The capacity model

Let $C(t)$ be the system capacity for new calls at time t and S_i be the time of i^{th} capacity-change. The stochastic process of the capacity model can be denoted by $\{C(t), t \geq 0\}$, where $C(t) = n - g$, $n - g + 1$, ... n. The system capacity is determined by the difference between n, the maximum number of channels in the system, and the number of handoff calls in the system. The time between two consecutive capacity changes is just the time between two consecutive handoff-call-events (arrival or completion) and its distribution is state dependent. Because both handoff call interarrival times and completion times are independently exponentially distributed, capacity interchange times also follow exponential distributions. Let $E_i(t)$ denote the distribution function of capacity interchange times given that the capacity is at state i. Then $E_i(t)$ is an exponential distribution function with rate $\lambda^{(i)}$. Refer to Figure 2.2 as the state transition diagram of the capacity model. For instance, when the capacity is at $n-g$, the capacity interchange time follows an exponential distribution $E_{n-g}(t)$ with rate $\lambda^{(n-g)} = g\mu_2$. It is not difficult to obtain the expression for $\lambda^{(i)}$:

$$\lambda^{(i)} = (n-i)\mu_2 + I(i \neq n-g) \cdot \lambda_2 \tag{5.4}$$

where

$$I(i \neq j) = \begin{cases} 1 & \text{if } i \neq j \\ \\ 0 & \text{if } i = j \end{cases} . \tag{5.5}$$

Let $H_{i,j}$ be the probability at which the capacity will change from i to j at the capacity-change instant:

$$H_{i,j} = P\{C(S_1) = j | C(S_0) = i\}. \tag{5.6}$$

Based on the state transition diagram it is not hard to see that $H_{n-g,n-g+1} = H_{n,n-1} = 1$. The capacity process $\{C(t), t \geq 0\}$ is a finite birth and death process. Note that the stochastic variation of the system capacity is independent of the traffic variation of new calls.

The composite model

Let the stochastic process $\{(C(t), N(t)), t \geq 0, N(t) \leq C(t)\}$ represent the traffic-capacity composite process with state space $\Omega = \{(i,k)|n-g \leq i \leq n, k \leq i\}$. It can be proven that this stochastic process is indeed a Markov regenerative process.

Proof. Let Y_i be the i^{th} sojourn time of the capacity process $\{C(t), t \geq 0\}$, then let $S_n = \sum_{i=1}^{n} Y_i$. Let N_n be the number of new calls in the system immediately after the nth capacity change and C_n be the system capacity at that time. Then $\{(C_n, N_n), S_n)\}$ is a Markov-renewal sequence with kernel $\mathbf{K}(t)$ (which will be discussed later) because

$$\begin{aligned} & P\{(C_{n+1,}N_{n+1}) = (i,j), S_{n+1} - S_n \leq x | (C_n, N_n) = (k,l), S_n, (C_{n-1}, N_{n-1}), \\ & S_{n-1}, (C_{n-2}, N_{n-2}), S_{n-2}, ...(C_0, N_0), S_0\} \\ & = P\{(C_{n+1}, N_{n+1}) = (i,j), S_{n+1} - S_n \leq x | (C_n, N_n) = (k,l)\} \qquad (5.7) \\ & = P\{(C_1, N_1) = (i,j), S_1 \leq x | (C_0, N_0) = (k,l)\}, \end{aligned}$$

and the process $\{(C(t), N(t)), t \geq 0\}$ is an MRGP because $\{(C(t+S_n), N(t+S_n)), t \geq 0\}$ given $\{(C(u), N(u)), 0 \leq u \leq S_n, (C(S_n), N(S_n)) = (k,l)\}$ is stochastically identical to $\{(C(t), N(t)), t \geq 0\}$ given $(C(0), N(0)) = (k,l)$. In other words, $\{(C(t+S_n), N(t+S_n)), t \geq 0\}$ depends on $\{(C(u), N(u)), 0 \leq u \leq S_n, (C(S_n), N(S_n)) = (k,l)\}$ only through $(C(S_n), N(S_n))$. The state space for this MRGP is Ω. ■

Expressions for global kernel and local kernel

The global kernel The entries of the global kernel $\mathbf{K}(\infty) = \lim_{t\to\infty} \mathbf{K}(t)$ are given by

$$\begin{aligned} K_{(i,k),(j,l)}(\infty) &= \lim_{t\to\infty} K_{(i,k),(j,l)}(t) \\ &= \lim_{t\to\infty} \Pr\{(C(S_1), N(S_1)) = (j,l), S_1 \leq t | (C(S_0), N(S_0)) = (i,k)\}. \end{aligned} \tag{5.8}$$

As a matter of fact, the state transition described by $K_{(i,k),(j,l)}(t)$ is a two-step transition. The first step transition is the evolution of the MRGP between two consecutive Markov regeneration epochs (the capacity-change epochs) and the second step transition is caused by the change of capacity. Assume that the system is at state (i,k) immediately after the most recent capacity-change. Then the system will first transit from state (i,k) to state (i,l) immediately before the next capacity change and then because of the change of capacity the system will then transit from state (i,l) to state (j,l). Note that if the new capacity j is less than l then call dropping occurs and the new state after capacity change is (j,j). For those state transitions that are invalid, for instance, where capacity changes more than one unit at a time or does not change at all, $K_{(i,k),(j,l)}(t)$ equals to zero. The evolution of the MRGP between the Markov regeneration epochs can be described by the infinitesimal generator $\mathbf{Q}_i$ of the subordinated CTMC, where the subscript i indicates the current system capacity before the next capacity-change epoch. Let $P^i_{k,l}(t) = P_{(i,k),(i,l)}(t)$ be the probability that the subordinated CTMC will be in state (i,l) at time t given that it was in state (i,k) initially, and $P^i_{k,l}(t)$ can be obtained by solving

$$\frac{d\mathbf{P}^i(t)}{dt} = \mathbf{P}^i(t)\mathbf{Q}_i. \tag{5.9}$$

Since the subordinated CTMC in this model is actually a finite birth and death process, its transient solution is given by ([44])

$$P_{k,l}^{i}(t) = \frac{\rho^{l}/l!}{\sum_{m=0}^{i}\rho^{m}/m!} + \frac{i!}{l!}\rho^{i-k}\cdot\sum_{r=1}^{i}\frac{D_k(x_r)D_l(x_r)}{x_r D_i(x_r)D_i^{\prime}(x_r+1)}e^{x_r\mu t}, \tag{5.10}$$

where

$$\rho = \lambda/\mu,$$

$$D_n(x) = \begin{cases} 1, & n = 0 \\ x+\rho, & n = 1 \\ (x+\rho+n-1)D_{n-1}(x) - (n-1)\rho D_{n-2}(x), & n = 2,3,\cdots,C \end{cases}, \tag{5.11}$$

and X_r, $r = 1,2,\cdots,i$, are the roots of $D_i(x+1) = 0$. Then the expressions of the entries in $\mathbf{K}(\infty) = \{K_{(i,k),(j,l)}(\infty)\}$ can be obtained. Since system capacity for new calls, $C(t)$ can only vary one unit at a time (i.e., it is skip-free), all the state-transitions of this MRGP can be classified into two cases:

- Case 1: Transitions may involve new call dropping. Call dropping would only occur when the capacity decreases at the capacity-change epoch and the system is full immediately after that. Therefore case 1 happens when

$$i = j+1,\ j = l \text{ and } i \neq n-g, \tag{5.12}$$

and the entry of the global kernel can be expressed as

$$\begin{aligned} K_{(i,k),(j,l)}(\infty) &= \lim_{t\to\infty} K_{(i,k),(j,l)}(t) \\ &= \lim_{t\to\infty}\left(\int_0^t P_{k,l}^{i}(x)\cdot H_{i,j}dE_i(x) + \int_0^t P_{k,i}^{i}(x)\cdot H_{i,j}dE_i(x)\right) \\ &= \int_0^\infty P_{k,l}^{i}(x)\cdot H_{i,j}dE_i(x) + \int_0^\infty P_{k,i}^{i}(x)\cdot H_{i,j}dE_i(x). \end{aligned} \tag{5.13}$$

Note that the first integral of the right-hand side of the above equation describes the scenario when no dropping occurs because the number of new calls in the system immediately before and after the capacity-change are both equal to l. However, the second integral accounts for the scenario when a new call is dropped: the number of new calls in the system is i $(= j + 1)$ immediately before the capacity-change and j immediately after the capacity-change.

- Case 2: Transitions that would not involve new call dropping. The indicator of such transitions is that the system is not full immediately after the capacity-change epoch, that is, $j > l$. The kernel entries of such transitions can be expressed as

$$\begin{aligned} K_{(i,k),(j,l)}(\infty) &= \lim_{t\to\infty} K_{(i,k),(j,l)}(t) \\ &= \lim_{t\to\infty} \int_0^t P^i_{k,l}(x) \cdot H_{i,j} dE_i(x) \\ &= \int_0^\infty P^i_{k,l}(x) \cdot H_{i,j} dE_i(x). \end{aligned} \tag{5.14}$$

To summarize, we have

$$K_{(i,k),(j,l)}(\infty) = \begin{cases} \int_0^\infty P^i_{k,l}(x) \cdot H_{i,j} dE_i(x) & \text{when } i = j+1,\ j = l, \\ + \int_0^\infty P^i_{k,i}(x) \cdot H_{i,j} dE_i(x) & \text{and } i \neq n-g. \\ \int_0^\infty P^i_{k,l}(x) \cdot H_{i,j} dE_i(x) & \text{when } j > l, j = i \pm 1. \\ 0 & \text{otherwise.} \end{cases} \tag{5.15}$$

The integrals in $\mathbf{K}(\infty)$ can be expressed as

$$\begin{aligned} &\int_0^\infty P^i_{k,l}(x) \cdot H_{i,j} dE_i(x) \\ &= \int_0^\infty \left[\frac{\rho^l/l!}{\sum_{m=0}^i \rho^m/m!} + \frac{i!}{l!}\rho^{i-k} \cdot \sum_{r=1}^{i} \frac{D_k(x_r)D_l(x_r)}{x_r D_i(x_r) D'_i(x_r+1)} e^{x_r \mu x} \right] \cdot H_{ij} dE_i(x) \\ &= \int_0^\infty \frac{\rho^l/l!}{\sum_{m=0}^i \rho^m/m!} H_{ij} dE_i(x) + \int_0^\infty \frac{i!}{l!}\rho^{i-k} \cdot \sum_{r=1}^{i} \left(\frac{D_k(x_r)D_l(x_r)}{x_r D_i(x_r) D'_i(x_r+1)} e^{x_r \mu x} H_{ij} dE_i(x) \right) \end{aligned}$$

$$= \frac{\rho^l/l!}{\sum_{m=0}^{i} \rho^m/m!} H_{ij} \int_0^\infty dE_i(x) + \frac{i!}{l!}\rho^{i-k} \cdot \sum_{r=1}^{i} \left(\frac{D_k(x_r)D_l(x_r)}{x_r D_i(x_r) D_i^!(x_r+1)} H_{ij} \int_0^\infty e^{x_r \mu x} dE_i(x) \right). \tag{5.16}$$

Let

$$f_i(x)dx = dE_i(x) = \lambda^{(i)} e^{-\lambda^{(i)}x} dx, \tag{5.17}$$

and since

$$\int_0^\infty dE_i(x) = 1 \tag{5.18}$$

$$\begin{aligned} \int_0^\infty e^{x_r \mu x} dE_i(x) &= \int_0^\infty e^{x_r \mu x} f_i(x) dx \\ &= \int_0^\infty e^{x_r \mu x} \lambda^{(i)} e^{-\lambda^{(i)}x} dx \\ &= \frac{\lambda^{(i)}}{\lambda^{(i)} - x_r \mu}, \end{aligned} \tag{5.19}$$

we have

$$\begin{aligned} (5.16) = & \frac{\rho^l/l!}{\sum_{m=0}^{i} \rho^m/m!} H_{ij} \\ & + \frac{i!}{l!}\rho^{i-k} \cdot H_{ij} \sum_{r=1}^{i} \left(\frac{D_k(x_r)D_l(x_r)}{x_r D_i(x_r) D_i^!(x_r+1)} \cdot \frac{\lambda^{(i)}}{\lambda^{(i)} - x_r \mu} \right). \end{aligned} \tag{5.20}$$

The local kernel The local kernel matrix $\mathbf{E}(t) = [E_{(i,k),(j,l)}(t)]$ describes the behavior of the embedded BDP between two consecutive capacity-change epochs. Define

$$\begin{aligned} E_{(i,k),(j,l)}(t) &= P\left\{(C(t), N(t)) = (j,l), t \le S_1 | \left(C\left(S_0\right), N\left(S_0\right)\right) = (i,k)\right\} \\ &= \begin{cases} P_{k,l}^{i}(t)(1 - E_i(t)) & \text{when } i = j, \ (i,k) \text{ and } (j,l) \in \Omega, \\ 0 & \text{otherwise.} \end{cases} \end{aligned} \tag{5.21}$$

Since $E_i(t)$ is the cumulative distribution function of an exponential distribution with rate $\lambda^{(i)}$, its complementary cumulative distribution function can be expressed as

$$E_i^c(t) = 1 - E_i(t) = e^{-\lambda^{(i)}t}. \tag{5.22}$$

Equation 5.21 becomes

$$E_{(i,k),(j,l)}(t) = \begin{cases} P_{k,l}^i(t)e^{-\lambda^{(i)}t} & \text{when } i = j,\ (i,k) \text{ and } (j,l) \in \Omega \\ \\ 0 & \text{otherwise} \end{cases}. \tag{5.23}$$

Then in order to study the limiting behavior of this MRGP, we need to compute

$$a_{(i,k),(j,l)} = \int_0^\infty E_{(i,k),(j,l)}(t)dt, \tag{5.24}$$

which is the mean time this MRGP spends in state (j,l) between two consecutive capacity-change epochs given that the system starts at state (i,k) immediately after the last capacity-change. When $i = j$, (i,k) and $(j,l) \in \Omega$, we have

$$\begin{aligned} a_{(i,k),(j,l)} &= \int_0^\infty P_{k,l}^i(t)e^{-\lambda^{(i)}t}dt \\ &= \int_0^\infty \left[\frac{\rho^l/l!}{\sum_{m=0}^i \rho^m/m!} + \frac{i!}{l!}\rho^{i-k} \cdot \sum_{r=1}^i \left(\frac{D_k(x_r)D_l(x_r)}{x_r D_i(x_r) D_i'(x_r+1)} e^{x_r \mu t} \right) \right] e^{-\lambda^{(i)}t}dt \\ &= \frac{\rho^l/l!}{\sum_{m=0}^i \rho^m/m!} \int_0^\infty e^{-\lambda^{(i)}t}dt + \frac{i!}{l!}\rho^{i-k} \sum_{r=1}^i \left(\frac{D_k(x_r)D_l(x_r)}{x_r D_i(x_r) D_i'(x_r+1)} \int_0^\infty e^{x_r \mu t} e^{-\lambda^{(i)}t}dt \right) \\ &= \frac{1}{\lambda^{(i)}} \left(\frac{\rho^l/l!}{\sum_{m=0}^i \rho^m/m!} \right) + \frac{i!}{l!}\rho^{i-k} \sum_{r=1}^i \left(\frac{D_k(x_r)D_l(x_r)}{x_r D_i(x_r) D_i'(x_r+1)} \cdot \frac{1}{\lambda^{(i)} - x_r \mu} \right). \end{aligned} \tag{5.25}$$

Performance measures

The steady state distribution of this MRGP can then be calculated using Equation 5.2:

$$\begin{aligned}\pi_{(i,k)} &= \lim_{t\to\infty} P\left\{C(t) = i, N(t) = k\right\} \\ &= \frac{\sum_{(j,l)\in\Omega} v_{(j,l)} a_{(j,l)(i,k)}}{\sum_{(j,l)\in\Omega} v_{(j,l)} \beta_{(j,l)}},\end{aligned} \tag{5.26}$$

where $\beta_{(j,l)} = \sum_{(m,r)\in\Omega} a_{(j,l)(m,r)}$ and $\mathbf{v} = [v_{(j,l)}]$ are the solution of

$$\mathbf{v} = \mathbf{v}\mathbf{K}(\infty) \text{ and } \sum v_{(j,l)} = 1. \tag{5.27}$$

From the steady state distribution, the following performance measures can be easily obtained:

$$\begin{aligned}&\text{New call blocking probabilty: } P_b^N(n,g) = \textstyle\sum_{i=k} \pi_{(i,k)}, \\ &\text{New call dropping probability: } P_d^N(n,g) = \tfrac{\lambda_2}{\lambda_1} \textstyle\sum_{i=k, i\neq n-g} \pi_{(i,k)}, \text{ and} \\ &\text{Handoff call blocking probability: } P_b^h(n,g) = \textstyle\sum_{i=n-g} \pi_{(i,k)}.\end{aligned} \tag{5.28}$$

5.3.2 A simple case when $n = 1$ and $g = 1$

In this section we solve a special case of the first guard channel model where $n = 1$ and $g = 1$ using the MRGP procedure. The state space of this system is $\Omega = \{(0,0), (1,0), (1,1)\}$.

The capacity model

The capacity for new calls at any time t, $C(t)$, could be either 0 (1 handoff call) or 1 (no handoff call). The state transition diagram for the capacity model can be found in Figure

5.1a. The distributions of interchange times

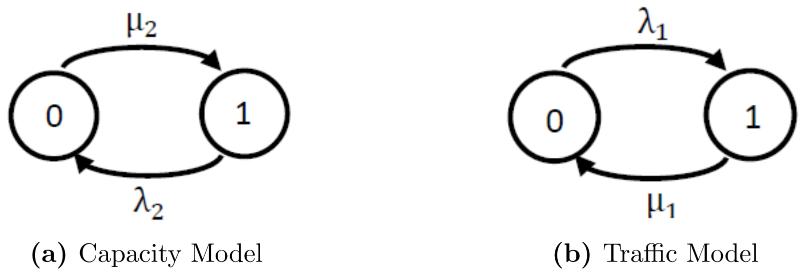
$$E_i(t) = \begin{cases} \text{exponential with rate } \mu_2, & \text{if the capacity is at 0,} \\ \text{exponential with rate } \lambda_2, & \text{if the capacity is at 1.} \end{cases} \tag{5.29}$$

The one step transition probabilities, H_{ij}, are also not difficult to obtain. From the transition diagram we have

$$H_{0,1} = 1 \text{ and } H_{1,0} = 1. \tag{5.30}$$

Traffic models

The traffic process $\{N(t), t \geq 0\}$ is a BDP. The number of states in this BDP depends on the current system capacity. When the capacity is at 0 the BDP has only one state and the transient probability $P^{(0)}_{0,0}(t) = 1$. When the capacity is at 1, the traffic process is then a two-state BDP (see Figure 5.1b) and its transient solution can be calculated using Equations 5.10 and 5.11. Firstly we calculate functions D_i:



(a) Capacity Model

(b) Traffic Model

Figure 5.1: State transition diagram of the capacity and the traffic model of the M1 model when $n = 1$ and $g = 1$

$$D_0(x) = 1 \tag{5.31}$$

$$D_1(x) = x + \rho \tag{5.32}$$

$$D_1^{\prime}(x) = 1 \tag{5.33}$$

$$D_1(x+1) = x+1+\rho \tag{5.34}$$

and the solution to $D_1(x+1) = 0$ is $x_1 = -(1+\rho)$. Then we have

$$\begin{aligned} D_0(x_1) &= 1 & (5.35)\\ D_1(x_1) &= -1 & (5.36)\\ D_1'(x_1+1) &= 1. & (5.37) \end{aligned}$$

Using Equation 5.10 all the four transient probabilities when the capacity is at 1 can be obtained:

$$\begin{aligned} P_{01}^1(t) &= \frac{\lambda_1}{\lambda_1+\mu_1} - \frac{\lambda_1}{\lambda_1+\mu_1}e^{-(\lambda_1+\mu_1)t} & (5.38)\\ P_{00}^1(t) &= \frac{\mu_1}{\lambda_1+\mu_1} + \frac{\lambda_1}{\lambda_1+\mu_1}e^{-(\lambda_1+\mu_1)t} & (5.39)\\ P_{10}^1(t) &= \frac{\mu_1}{\lambda_1+\mu_1} - \frac{\mu_1}{\lambda_1+\mu_1}e^{-(\lambda_1+\mu_1)t} & (5.40)\\ P_{11}^1(t) &= \frac{\lambda_1}{\lambda_1+\mu_1} + \frac{\mu_1}{\lambda_1+\mu_1}e^{-(\lambda_1+\mu_1)t}. & (5.41) \end{aligned}$$

Computing the global and local kernels

As stated in Section 5.3.1, the non-zero entries of the global kernel $K(\infty)$ can be divided into two categories:

- Case 1: New call dropping may occur, and two of them fall into this category:

$$\begin{aligned} K_{(1,0),(0,0)}(\infty) &= \int_0^\infty P_{01}^1(x)H_{10}dE_1(x) + \int_0^\infty P_{00}^1(x)H_{10}dE_1(x)\\ &= \int_0^\infty 1\cdot H_{10}dE_1(x)\\ &= 1 & (5.42)\\ K_{(1,1),(0,0)}(\infty) &= \int_0^\infty P_{11}^1(x)H_{10}dE_1(x) + \int_0^\infty P_{10}^1(x)H_{10}dE_1(x)\\ &= \int_0^\infty 1\cdot H_{10}dE_1(x)\\ &= 1. & (5.43) \end{aligned}$$

- Case 2: New call dropping cannot occur. There is only one entry of $\mathbf{K}(\infty)$ that belongs to this category:

$$\begin{aligned} K_{(0,0),(1,0)}(\infty) &= \int_0^\infty P_{00}^0(x) H_{01} dE_0(x) && (5.44) \\ &= 1. && (5.45) \end{aligned}$$

After ordering all the states lexicographically as $\{(0,0)\ (1,0)\ (1,1)\}$, the global kernel $\mathbf{K}(\infty)$ can be expressed as

$$\mathbf{K}(\infty) = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}. \qquad (5.46)$$

Solve

$$\mathbf{v} = \mathbf{v}\mathbf{K}(\infty) \text{ and } \sum v_{(j,l)} = 1 \qquad (5.47)$$

for $\mathbf{v}$ and we have

$$\begin{cases} v_{(0,0)} = \frac{1}{2} \\ v_{(1,0)} = \frac{1}{2} \\ v_{(1,1)} = 0 \end{cases} \qquad (5.48)$$

Following equation 5.23 the non-zero entries of local kernel $E(t)$ can be easily obtained:

$$\begin{aligned} E_{(1,0),(1,1)}(t) &= \left[\frac{\lambda_1}{\lambda_1 + \mu_1} - \frac{\lambda_1}{\lambda_1 + \mu_1} e^{-(\lambda_1+\mu_1)t} \right] \cdot e^{-\lambda_2} && (5.49) \\ E_{(1,1),(1,0)}(t) &= \left[\frac{\mu_1}{\lambda_1 + \mu_1} - \frac{\mu_1}{\lambda_1 + \mu_1} e^{-(\lambda_1+\mu_1)t} \right] \cdot e^{-\lambda_2} && (5.50) \\ E_{(1,0),(1,0)}(t) &= \left[\frac{\mu_1}{\lambda_1 + \mu_1} + \frac{\lambda_1}{\lambda_1 + \mu_1} e^{-(\lambda_1+\mu_1)t} \right] \cdot e^{-\lambda_2} && (5.51) \\ E_{(1,1),(1,1)}(t) &= \left[\frac{\lambda_1}{\lambda_1 + \mu_1} + \frac{\mu_1}{\lambda_1 + \mu_1} e^{-(\lambda_1+\mu_1)t} \right] \cdot e^{-\lambda_2} && (5.52) \\ E_{(0,0),(0,0)}(t) &= 1 \cdot e^{-\mu_2}. && (5.53) \end{aligned}$$

And therefore all the non-zero $a_{(i,k),(j,l)}$s can also be obtained:

$$\begin{aligned}
a_{(1,0),(1,1)} &= \int_0^\infty E_{(1,0),(1,1)}(t)dt = \frac{\lambda_1}{\lambda_2(\lambda_1+\mu_1)} - \frac{\lambda_1}{\lambda_1+\mu_1}\frac{1}{\lambda_1+\lambda_2+\mu_1} && (5.54)\\
a_{(1,1),(1,0)} &= \frac{\mu_1}{\lambda_2(\lambda_1+\mu_1)} - \frac{\mu_1}{\lambda_1+\mu_1}\frac{1}{\lambda_1+\lambda_2+\mu_1} && (5.55)\\
a_{(1,0),(1,0)} &= \frac{\mu_1}{\lambda_2(\lambda_1+\mu_1)} + \frac{\lambda_1}{\lambda_1+\mu_1}\frac{1}{\lambda_1+\lambda_2+\mu_1} && (5.56)\\
a_{(1,1),(1,1)} &= \frac{\lambda_1}{\lambda_2(\lambda_1+\mu_1)} + \frac{\mu_1}{\lambda_1+\mu_1}\frac{1}{\lambda_1+\lambda_2+\mu_1} && (5.57)\\
a_{(0,0),(0,0)} &= \frac{1}{\mu_2}. && (5.58)
\end{aligned}$$

Computing the steady state probabilities and performance measures

Now that we have all the ingredients and are ready to compute the steady state probabilities $\pi_{(i,k)}$. By Equation 5.26, there follows

$$\begin{aligned}
\pi_{(0,0)} &= \frac{\sum_{(j,l)\in\Omega} v_{(j,l)} a_{(j,l)(0,0)}}{\sum_{(j,l)\in\Omega} v_{(j,l)} \sum_{(m,r)\in\Omega} a_{(j,l)(m,r)}} = \frac{\lambda_2}{\lambda_2+\mu_2}\\
\pi_{(1,0)} &= \frac{\sum_{(j,l)\in\Omega} v_{(j,l)} a_{(j,l)(1,0)}}{\sum_{(j,l)\in\Omega} v_{(j,l)} \sum_{(m,r)\in\Omega} a_{(j,l)(m,r)}} = \frac{(\mu_1+\lambda_2)\mu_2}{(\lambda_1+\mu_1+\lambda_2)(\lambda_2+\mu_2)} \qquad (5.59)\\
\pi_{(1,1)} &= \frac{\sum_{(j,l)\in\Omega} v_{(j,l)} a_{(j,l)(1,1)}}{\sum_{(j,l)\in\Omega} v_{(j,l)} \sum_{(m,r)\in\Omega} a_{(j,l)(m,r)}} = \frac{\lambda_1\mu_2}{(\lambda_1+\mu_1+\lambda_2)(\lambda_2+\mu_2)}.
\end{aligned}$$

If these steady state probabilities were calculated using the composite model approach (Refer to Chapter 2), that is, by solving the following system of equations for $\pi_{(i,k)}$, the same formulae as presented in (5.59) could be obtained:

$$\begin{cases}
-\mu_2\pi_{(0,0)} + \lambda_2\pi_{(1,0)} + \lambda_2\pi_{(1,1)} = 0\\
-(\lambda_1+\lambda_2)\pi_{(1,0)} + \mu_2\pi_{(0,0)} + \mu_1\pi_{(1,1)} = 0 \quad \cdot\\
\pi_{(0,0)} + \pi_{(1,0)} + \pi_{(1,1)} = 1
\end{cases} \qquad (5.60)$$

The performance measures of this example follows Equation 5.28:

$$
\begin{aligned}
P_b^N(n,g) &= \pi_{(0,0)} + \pi_{(1,1)} = \frac{(\lambda_1+\mu_1+\lambda_2)\lambda_2+\lambda_1\mu_2}{(\lambda_1+\mu_1+\lambda_2)(\lambda_2+\mu_2)} && (5.61)\\
P_d^N(n,g) &= \frac{\lambda_2}{\lambda_1}\pi_{(1,1)} = \frac{\lambda_2\mu_2}{(\lambda_1+\mu_1+\lambda_2)(\lambda_2+\mu_2)} && (5.62)\\
P_b^h(n,g) &= \pi_{(0,0)} = \frac{\lambda_2}{\lambda_2+\mu_2}. && (5.63)
\end{aligned}
$$

Note that $P_b^h(n,g)$ can be rewritten as $\rho_2/(1+\rho_2)$ which is just the Erlang B formula $EB(\rho_2,1)$, which agrees with Equation 2.7.

Although closed form solutions can be found when n is small, for general n and g the steady state distribution of the M1 model is intractable and no closed form solutions have been found. In the next section the performance measures of the M1 model for larger n and g are computed numerically using both the MRGP method and the composite model method.

5.3.3 Numerical examples

In this section we verify the MRGP method by comparing its results (new call blocking and dropping probabilities) to the results calculated using the composite model approach (as described in Chapter 2). Both methods are implemented in Matlab in which the function `mldivide` is used to numerically solve a system of linear equations and the function `roots` is used to numerically find the roots of a polynomial equation. Let $n = 20$, $g = 10$, $\mu_1 = \mu_2 = 1$, $\lambda_1 = 10$ and the mobility can vary from 0.1 to 5. As Figure 5.2 suggests, the solutions calculated by the MRGP method and the composite model method agree with each other very well. However, the CPU time of running MRGP is almost 50 times the CPU time needed for applying the composite model method. Therefore, using the method of MRGP to solve Markovian queues is inefficient. In the next section we apply MRGP on the M/M/~C/~C system where capacity interchange times follow non exponential distribution and therefore cannot be analyzed using the composite model method.

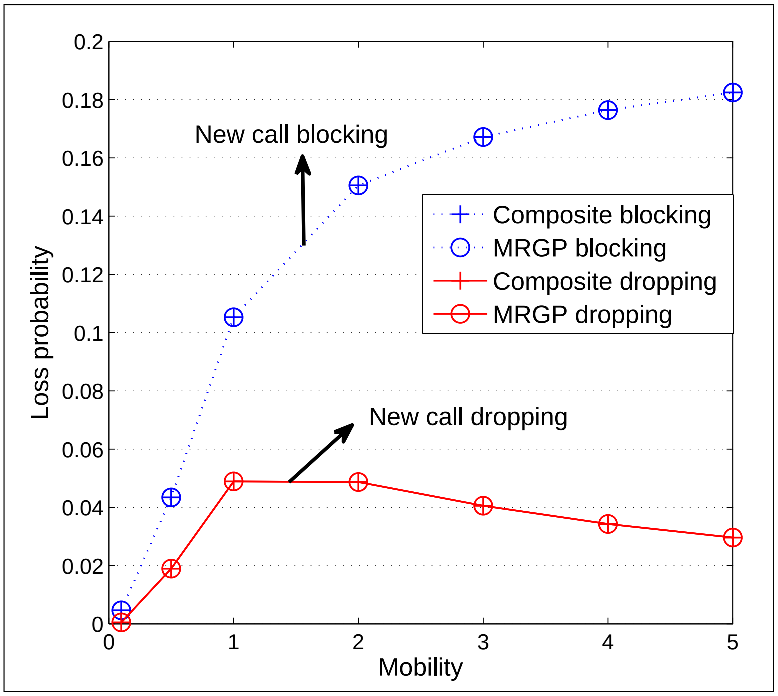


Figure 5.2: The M1 model: results for the MRGP method and the composite model method

5.4 Application to the $M/M/{\sim}C/{\sim}C$ queueing systems

The $M/M/{\sim}C/{\sim}C$ queueing system is a variant of the loss system $M/M/C/C$ where the system capacity may change over time. The maximum possible capacity for this system is C. As stated in Chapter 1, the system capacities of wireless networks may vary randomly with time for various reasons such as the channel status, the dynamics of protocols used for channel assignment, bandwidth allocation, rate control and mobility management [52]. In this section we analyze the $M/M/{\sim}C/{\sim}C$ queueing system. We assume that capacity interchange times are independent and identically distributed with a general cumulative distribution function $G(\cdot)$; we further assume that the value of the capacity can change from its current value i to a different value j drawn randomly from $\{j|0 \le j \le C, j \ne i\}$. Then we examine three specific distributions of capacity interchange times (exponential, gamma and Pareto) and three capacity variation patterns (skip-free, distance-based and uniform-based). Numerical and simulation results are presented to conclude this section.

5.4.1 Analytical model

Traffic models

The system occupancy at any time t is denoted by $N(t), t \geq 0$. The traffic arrives according to a Possion process with rate λ. The service times follow independent exponential distributions with common rate μ. Given that the system capacity is i, the stochastic process $\{N(t), t \geq 0\}$ is a homogeneous continuous time Markov chain (CTMC) with state space $\Omega_N : \{0, 1, \ldots, i\}$. This model can be analyzed as an $M/M/i/i$ queueing system. The steady state probabilities are given by

$$P(N = k) = \frac{(\frac{\lambda}{\mu})^k / k!}{\sum\limits_{l=0}^{i} (\frac{\lambda}{\mu})^l / l!}, \; k = 0, 1, \cdots, i. \tag{5.64}$$

The capacity model

The capacity variation model is again represented by the stochastic process $\{C(t), t \geq 0\}$ with state space $\Omega_C = \{0, 1, \ldots, C\}$, where C stands for the maximum possible capacity for this system. The random variable $C(t)$ represents the capacity of the system at time t. The time between two consecutive capacity changes is generally (non-exponentially) distributed with cumulative distribution function $G(\cdot)$, density function $g(\cdot)$ and mean μ_c. Define $\lambda_c = 1/\mu_c$ as the average capacity-change rate. At capacity-change time epochs, the system capacity may change from its current state to any other state in the state space. Note that capacity-change is independent of the traffic variation. When capacity drops at the capacity-change instant, empty channels will be dropped first to avoid unnecessary call droppings. The capacity process $\{C(t), t \geq 0\}$ can be modeled as a SMP: let us assume that it starts at the initial state C_0 at time $t = 0$ and stays on that state for a sojourn time of Y_1 before jumping to the next state, C_1. In general it stays in state $C_n, n \geq 0$ for a duration of Y_n and then jumps to the next state C_{n+1}. Then Y_is are i.i.d. with cumulative distribution function $G(\cdot)$. Let $\tau_i = \sum_{j=1}^{i} Y_j$, which is the i^{th} capacity-change instance. Let $J(t)$ be the number of capacity changes up to time t. Then the sequence $\{C_0, (C_n, Y_n), n \geq 1\}$ can be used to define the capacity process $\{C(t), t \geq 0\}$ by

$$C(t) = C_{J(t)}, t \geq 0. \tag{5.65}$$

And $\{C(t), t \geq 0\}$ is an SMP because the sequence $\{C_0, (C_n, Y_n), n \geq 1\}$ satisfies

$$\begin{aligned} &\Pr(C_{n+1} = j, Y_{n+1} \leq y | C_n = i, Y_n, C_{n-1}, Y_n, ..., C_1, Y_1, C_0) \\ &= \Pr(C_1 = j, Y \leq y | C_0 = i),\ i, j \in \Omega_C,\ n \geq 0. \end{aligned} \tag{5.66}$$

Assume that there are a finite number of capacity-changes during a finite time, then $\{C_n, n \geq 0\}$ is called the embedded DTMC in the SMP with transition probabilities

$$H_{i,j} = \Pr(C_{n+1} = j | C_n = i),\ i, j \in \Omega_C,\ n \geq 0. \tag{5.67}$$

The composite model

Let the stochastic process $\{(C(t), N(t))\}$ represents the traffic-capacity composite model. Define Ω to be the state space of this stochastic process and we have

$$\Omega = \{(i, k) | 0 \leq i \leq C, 0 \leq k \leq i\}. \tag{5.68}$$

Define $N'_n = N(\tau_n)$, which is the number of calls in the system at the n^{th} capacity-change instance. Then the sequence $\left\{(C_0, N'_0), \left((C_n, N'_n), Y_n\right), n \geq 1\right\}$ can be used to define the two dimensional process $\{(C(t), N'(t)), t \geq 0\}$ by

$$\left(C(t), N'(t)\right) = (C_{J(t)}, N'_{J(t)}),\ t \geq 0. \tag{5.69}$$

It is not difficult to see that this process is an SMP because it only possesses Markovian property at capacity-change epochs. Define the kernel of this SMP as follows:

$$\mathbf{K}(t) = [K_{(i,k),(j,l)}(t)]_{(i,k),(j,l) \in \Omega},\ t \geq 0, \tag{5.70}$$

where

$$K_{(i,k),(j,l)}(t) = \Pr\{(C_1, N_1') = (j,l), Y_1 \le t | (C_0, N_0') = (i,k)\}. \tag{5.71}$$

Define $\mathbf{v} = [v_{(i,k)}]_{(i,k)\in\Omega}$. Then an SMP is completely described by its kernel $\mathbf{K}(t)$ and the initial distribution

$$v_{(i,k)} = \Pr\Big(\big(C(0), N'(0)\big) = (i,k)\Big),\ (i,k) \in \Omega. \tag{5.72}$$

Clearly $\{(C_n, N_n'), n \ge 0\}$ is a DTMC (called the embedded DTMC in the SMP) with transition probabilities

$$K_{(i,k),(j,l)}(\infty) = \Pr\{(C_{n+1}, N_{n+1}') = (j,l) | (C_n, N_n') = (i,k)\},\ (i,k),(j,l) \in \Omega, \tag{5.73}$$

and the initial distribution of the embedded DTMC, $\mathbf{v}$, is a positive solution to solution to

$$\mathbf{v} = \mathbf{v}\mathbf{K}(\infty) \text{ and } \sum_{(i,k)\in\Omega} v_{(i,k)} = 1. \tag{5.74}$$

The Markov regenerative process (MRGP)

The following proof shows that the stochastic process for the composite model $\{(C(t), N(t))\}$ is indeed an MRGP.

Proof. As defined before, Y_n is the n^{th} sojourn time of the capacity process $\{C(t), t \ge 0\}$ and $\tau_n = \sum_{j=1}^{n} Y_j$ denote the n^{th} capacity jump epoch. Let N_n and C_n be the number of customs in the system and the system capacity, respectively, which are observed immediately after the n^{th} capacity-change (and insures right continuous with left limit at each capacity-change epoch). Then $\{((C(t), N(t)), \tau_n\}$ is a Markov-renewal sequence because

$$\begin{aligned}
&\Pr\{(C_{n+1}, N_{n+1}) = (i,j), \tau_{n+1} - \tau_n \le x | (C_n, N_n) = (k,l), \tau_n,\\
&(C_{n-1}, N_{n-1}), \tau_{n-1}, (C_{n-2}, N_{n-2}), \tau_{n-2}, \cdots, (C_0, N_0), \tau_0\}\\
&= \Pr\{(C_{n+1}, N_{n+1}) = (i,j), \tau_{n+1} - \tau_n \le x | (C_n, N_n) = (k,l)\}\\
&= \Pr\{(C_{n+1}, N_{n+1}) = (i,j), \tau_1 \le x | (C_0, N_0) = (k,l)\},
\end{aligned} \tag{5.75}$$

and the process $\{(C(t), N(t)), t \geq 0\}$ is a MRGP because $\{(C(t+\tau_n), N(t+\tau_n)), t \geq 0\}$ given $\{(C(u), N(u)), 0 \leq u \leq \tau_n, (C(\tau_n), N(\tau_n)) = (k,l)\}$ is stochastically identical to $\{(C(t), N(t)), t \geq 0\}$ given $\{C(0), N(0) = (k,l)\}$. In other words, $\{(C(t+\tau_n), N(t+\tau_n)), t \geq 0\}$ depends on $\{(C(u), N(u)), 0 \leq u < S_n, C(\tau_n), N(\tau_n)\}$ only through $(C(\tau_n), N(\tau_n))$. $\tau_n^{'}s$ are called the Markov regeneration epochs of this MRGP. The state space for this MRGP is Ω. ■

Expressions for global kernel and local kernel

The global kernel

We can write out the expression of the global kernel $\mathbf{K}(\infty)$ as follows:

$$\begin{aligned}
K(\infty) &= \lim_{t\to\infty} K_{(i,k),(j,l)}(t) \\
&= \lim_{t\to\infty} \begin{cases} \int_0^t P^i_{k,l}(x) H_{i,j} dG(x), & i \neq j, l < j \text{ and } l < i \\ \int_0^t P^i_{k,j}(x) H_{i,j} dG(x) \\ \quad + \sum\limits_{m=j+1}^{i} \int_0^t P^i_{k,m}(x) H_{i,j} dG(x) & i \neq j, l = j \text{ and } i > j \\ 0 & \text{otherwise} \end{cases} \\
&= \begin{cases} \int_0^\infty P^i_{k,l}(x) H_{i,j} dG(x), & i \neq j, l < j \text{ and } l < i \\ \sum\limits_{m=j}^{i} \int_0^\infty P^i_{k,m}(x) H_{i,j} dG(x) & i \neq j, l = j \text{ and } i > j \\ 0 & \text{otherwise.} \end{cases}
\end{aligned} \tag{5.76}$$

Denote $\hat{f}(s)$ as the Laplace transformation of function $f(t)$:

$$\hat{f}(s) = \mathcal{L}\{f(t)\} = \int_0^\infty e^{-st} f(t) dt. \tag{5.77}$$

After using (5.10) to substitute all the transition probabilities of the traffic model in (5.76), it is not difficult to show that when $l < j$ and $l < i$, the global kernel can be expressed as

$$\begin{aligned} K_{(i,k),(j,l)}(\infty) &= \frac{\rho^l/l!}{\sum_s \rho^s/s!} H_{i,j} \\ &+ \frac{i!}{l!}\rho^{i-k} H_{i,j} \cdot \sum_{r=1}^{i} \frac{D_k(x_r)D_l(x_r)}{x_r D_i(x_r) D_i^{\prime}(x_r+1)} \hat{g}(-x_r\mu), \end{aligned} \tag{5.78}$$

and when $l = j$ and $i > j$, the global kernel has entries

$$\begin{aligned} &K_{(i,k),(j,l)}(\infty) = H_{i,j} \\ &\cdot \sum_{m=j}^{i} \left(\frac{\rho^m/m!}{\sum_s \rho^s/s!} + \frac{i!}{m!}\rho^{i-k} \cdot \sum_{r=1}^{i} \frac{D_k(x_r)D_m(x_r)}{x_r D_i(x_r) D_i^{\prime}(x_r+1)} \hat{g}(-x_r\mu) \right), \end{aligned} \tag{5.79}$$

where $\hat{g}(\cdot)$ denotes the Laplace transform of the probability distribution function $g(\cdot)$.

The local kernel

The local kernel matrix $\mathbf{E}(t)$ describes the behavior of the process during the time between two consecutive capacity changes (starting from the state of the system immediately after the last capacity-change), and it can be written as

$$E_{(i,k),(j,l)}(t) = \begin{cases} P_{k,l}^{i}(t) \cdot (1 - G(t)) & (i,k),(i,l) \in \Omega \\ 0 & \text{Otherwise} \end{cases} \tag{5.80}$$

where $P_{k,l}^{i}(t)$ are defined in Equation 5.10.

To study the limiting behaviors of the MRGP we will need the following quantity:

$$a_{(i,k),(j,l)} = \int_0^\infty E_{(i,k),(j,l)}(t)dt = \int_0^\infty P_{k,l}^{i}(t) \cdot (1 - G(t))dt, \tag{5.81}$$

which is the mean time that the MRGP spends in state (j,l) between two successive capacity-change epochs when the system is initially in state (i,k) immediately after the last capacity-change. The closed form expression of $a_{(i,k),(j,l)}$ can also be obtained by expanding $P_{k,l}^{i}(t)$ in

(5.81) using (5.10):

$$\begin{aligned} a_{(i,k),(j,l)} &= \left(\frac{\rho^l/l!}{\sum_s \rho^s/s!}\right)\int_0^\infty (1-G(t))dt \\ &+ \frac{i!}{l!}\rho^{i-k}\cdot\sum_{r=1}^{i}\frac{D_k(x_r)D_l(x_r)}{x_r D_i(x_r)D_i'(x_r+1)}\hat{G}^c(-x_r\mu) \\ &= \mu_c\left(\frac{\rho^l/l!}{\sum_s \rho^s/s!}\right) + \frac{i!}{l!}\rho^{i-k}\cdot\sum_{r=1}^{i}\frac{D_k(x_r)D_l(x_r)}{x_r D_i(x_r)D_i'(x_r+1)}\hat{G}^c(-x_r\mu), \end{aligned} \tag{5.82}$$

where μ_c is the mean time between capacity changes or the mean of the probability density function $g(\cdot)$, and $\hat{G}^c(\cdot)$ is the Laplace transform of the complementary cumulative distribution function of $G(\cdot)$.

Steady state probabilities

The steady state probabilities, $\mathbf{v} = [v_{(i,k)}]$, of the embedded DTMC is a positive solution to

$$\mathbf{v} = \mathbf{v}\mathbf{K}(\infty) \text{ and } \sum_{(i,k)\in\Omega} v_{(i,k)} = 1. \tag{5.83}$$

The steady state probabilities $\boldsymbol{\pi}= [\pi_{(i,k)}]$ of the MRGP are calculated as follows:

$$\begin{aligned} \pi_{(i,k)} &= \lim_{t\to\infty} \Pr\{C(t) = i, N(t) = k\} \\ &= \frac{\sum_{(j,l)\in\Omega} V_{(j,l)}a_{(j,l),(i,k)}}{\sum_{(j,l)\in\Omega} V_{(j,l)}\beta_{(j,l)}}, \end{aligned} \tag{5.84}$$

where $\beta_{(j,l)} = \sum_{(m,r)\in\Omega} a_{(j,l),(m,r)}$.

The important quantities required to calculate the steady state probabilities are summarized below:

- $P^i_{k,j}(t)$: This is the transient solution to the subordinate CTMC when the system capacity is i. Formulae for computing them are presented in (5.10) and (5.11). Note that in (5.10) the roots X_rs to polynomial equations can be found using the Matlab function `roots`.

- $H_{i,j}$: This is the probability for the system capacity to change from i to j at the capacity-change epoch (as defined in (5.67)). Formulae for calculating $H_{i,j}$ for capacity variation patterns are developed in section: Capacity variation patterns.
- $\hat{g}(-x_r\mu)$ and $\hat{G}^c(-x_r\mu)$: These are the Laplace transforms of the probability density function $g(\cdot)$ and the complementary cumulative distribution function $G^c(\cdot)$ at $-x_r\mu$, respectively. These two quantities are required in the calculation of $K_{(i,k),(j,l)}(\infty)$ and $a_{(i,k),(j,l)}$. For different distributions of capacity interchange times, formulae for calculating $\hat{g}(-x_r\mu)$ and $\hat{G}^c(-x_r\mu)$ are presented in section: The distribution of capacity interchange times.
- **v**: This is the steady state distribution of the embedded DTMC. **v** can be obtained by solving the system of linear equations defined in (5.83) numerically using the Matlab function `mldivide`.

Performance measures

The performance measures of interest of the $M/M/{\sim}C/{\sim}C$ system are the blocking probability (denoted by P_b) and the dropping probability (denoted by P_d). The blocking probability P_b is the sum of all the steady state probabilities ($\pi_{(i,k)}$'s) where $i = k$. Following the idea presented in Equation 2.9 in Chapter 2, the dropping probability P_d can be calculated. However, the formula for calculating P_d is more complicated because there are more state transitions that can cause call dropping, and capacity interchange times are non-exponentially distributed. The dropping probability can be calculated as

$$\frac{1}{\lambda} \sum_{(i,k)\in\Omega_d} \pi_{(i,k)} \sum_{(j,l)\in\Omega_{(i,k)}} R_{(i,k),(j,l)} N^d_{(i,k),(j,l)} H_{ij}, \tag{5.85}$$

where:

- $R_{(i,k),(j,l)}$ is the expected transit rate for the system to transit from state (i,k) to state (j,l) for $i \neq j$. If the capacity interchange time is exponential then this quantity is a constant and equal to $1/\mu_c$. But when capacity interchange times are non exponential, this quantity is no longer a constant. It depends on the mean time the system will spend before entering state (i,k) since the last capacity-change.

- λ is the traffic arrival rate.
- Ω_d is a subset of the system state space and includes all the states that are the initial state of a state transition that involves dropping events. For instance, when this queueing system transits from state $(4,4)$ to state $(3,3)$, one dropping event happens, and therefore the initial state of this transition, $(4,4)$, will be included in Ω_d. In other words, Ω_d is a collection of all states that, when system is at one of these states, can lead to transitions involving dropping events. It is not hard to see that Ω_d should include all the states when the system is not empty (that is, these is at least one call in the system). We can write $\Omega_d = \{(i,k)|(i,k) \in \Omega_M, i \neq 0 \text{ and } k \neq 0\}$.
- $\Omega_{(i,k)}$ is a subset of the system state space Ω_M and includes all the terminal states of the transitions involving dropping event(s) initiated from state (i,k).
- $\pi_{(i,k)}$ is the steady state probability of state (i,k) calculated by Equation 5.84.
- $N^d_{(i,k),(j,l)}$ is the number of calls that have been dropped caused by the transition from state (i,k) to (j,l).
- H_{ij} is the probability (as defined in Equation 5.67) that the capacity will change to j at the capacity-change instant given that its current value is i.

However, difficulty arises when capacity interchange times are non-exponential. For non-exponential capacity interchange times (any that do not possess memoryless property), the expected time left until the next capacity-change epoch is not always equal to $1/\lambda_c$. As a matter of fact, the mean residual time for a given state depends on the amount of time that has already elapsed since the last capacity-change. Therefore, the transition rates ($R_{(i,k),(j,l)}$) in the above formula are difficult to obtain. In the following section we develop a new method to calculate dropping probabilities which can be easily applied to cases when capacity interchange times are non-exponential.

Calculating dropping probability for non-exponential cases

Since call dropping events can only occur at capacity-change epochs, they can be completely captured by the SMP $\{(C(t), N'(t)), t \geq 0\}$. Suppose that the system is in state (i,k) immediately after a capacity change epoch, and transits to state (j,l) at the second capacity-change epoch. The necessary and sufficient conditions for a dropping event to occur at the second capacity-change epoch are:

1. The system is full immediately after the second capacity-change epoch, that is,
$$j = l \tag{5.86}$$
and
2. There are more than l calls in the system immediately before the second capacity-change. Let (i, m) be the system state immediately before the second capacity-change, then we have
$$i > j \text{ and } m > l. \tag{5.87}$$

Therefore, when the states (i, k) and (j, l) satisfy:

$$\begin{cases} j = l \\ i > j \end{cases} \tag{5.88}$$

the probability for the system to transit from (i, k) to (j, l) via state (i, m) (where $m > l$, and the system stays on (i, m) until it transits to state (j, l) at the capacity-change instant) can be found as part of the expression of $\mathbf{K}(\infty)$ in Equation 5.76, which is

$$\int_0^\infty P^i_{k,m}(x) H_{i,j} dG(x) \tag{5.89}$$

and during such transition, $\lambda_c(m - j)$ calls are dropped, where λ_c is the average capacity-change rate. Therefore, the expected number of calls that are dropped when the system transits from (i, k) to (j, l) can be calculated as

$$\sum_{m=j+1}^{i} \left(\lambda_c(m - j) \int_0^\infty P^i_{k,m}(x) H_{i,j} dG(x) \right). \tag{5.90}$$

Given the initial distribution $\mathbf{v}$ of the embedded DTMC (which is, in fact, also the initial distribution of the SMP $\{(C(t), N'(t)), t \geq 0\}$), and the traffic arrival rate λ, the total dropping probability for the system can be obtained as

$$\frac{1}{\lambda}\cdot \sum_{\substack{\text{for all pairs of}\\ (i,k) \text{ and } (j,l)\\ \text{satisfy } j=l \text{ and } i>j}} \left[v_{(i,k)} \cdot \sum_{m=j+1}^{i} \left(\lambda_c (m-j) \int_0^\infty P^i_{k,m}(x) H_{i,j} dG(x) \right) \right]. \qquad (5.91)$$

The distribution of capacity interchange times

As studied in Luo and Williamson [32], Sun and Williamson [50, 51], Sun et al. [52], the characteristics of the capacity variation process can have a large impact on the performance measures such as call blocking and dropping probabilities. As stated in Luo and Williamson [32], we assume that the probability distribution function of capacity interchange times is $G(\cdot)$ with mean μ_c and consider three domains: $[0,\mu_c)$ (the "head"), $[\mu_c, 3\mu_c]$ (the "body"), and $(3\mu_c, \infty)$ (the "tail") respectively and then three different kinds of distribution functions are considered: gamma distribution (with the shape parameter less than 10), which has a larger density function at the "head", Pareto distribution with shape parameter $1 < a < 2$, which has a larger density function at the tail, and exponential distribution whose density is relatively evenly distributed over the three domains. We will use these three kinds of probability distribution functions as the distributions followed by capacity interchange times in the $M/M/\sim C/\sim C$ system. The focus is on developing formulae for $\hat{g}(-x_r\mu)$ and $\hat{G}^c(-x_r\mu)$ in Equation 5.79 and 5.82.

Exponential distribution

When the cumulative distribution function $G(\cdot)$ of capacity interchange times is an exponential distribution with mean μ_c, we have

$$g(x) = \frac{1}{\mu_c} e^{-\frac{x}{\mu_c}} \qquad (5.92)$$

$$G^c(x) = e^{-\frac{x}{\mu_c}}. \qquad (5.93)$$

The Laplace transform of the probability density function $g(x)$ at $-x_r\mu$ is

$$\hat{g}(-x_r\mu) = \int_0^\infty e^{x_r\mu x} \frac{1}{\mu_c} e^{-\frac{x}{\mu_c}} dx$$

$$= \frac{\frac{1}{\mu_c}}{\frac{1}{\mu_c} - x_r\mu}. \tag{5.94}$$

The Laplace transform of the complementary cumulative distribution function can be obtained using

$$\hat{G}^c(s) = \frac{1 - \hat{g}(s)}{s}, \tag{5.95}$$

and we have

$$\begin{aligned} \hat{G}^c(-x_r\mu) &= \frac{1}{s} - \frac{\hat{g}(-x_r\mu)}{s} \\ &= \frac{1}{\frac{1}{\mu_c} - x_r\mu}. \end{aligned} \tag{5.96}$$

Gamma distribution

Gamma distribution is a two-parameter family of continuous probability distributions. Let α be its shape parameter and β be the inverse scale parameter. If we only consider the case where α is a positive integer, then the distribution represents an Erlang distribution; that is, the sum of k independent exponentially distributed random variables, each of which has a mean of $1/\beta$. Based on this parametrization we can write its probability density function as well as the probability distribution function:

$$g(x,\alpha,\beta) = \beta^\alpha \frac{1}{\Gamma(\alpha)} x^{\alpha-1} e^{-x\beta} \tag{5.97}$$

$$G(x,\alpha,\beta) = \int_0^x \beta^\alpha \frac{1}{\Gamma(\alpha)} t^{\alpha-1} e^{-t\beta} dt. \tag{5.98}$$

The Laplace transform of $g(x,\alpha,\beta)$ can be found in Hogg and Craig [20] and the Laplace transform of $G^c(x,\alpha,\beta)$ can be obtained using Equation 5.95. Then $\hat{g}(-x_r\mu)$ and $\hat{G}^c(-x_r\mu)$ can be calculated as

$$\hat{g}(-x_r\mu) = \int_0^\infty e^{x_r\mu x} g(x,\alpha,\beta) dx$$

$$= \frac{\beta^{\alpha}}{(-x_r\mu + \beta)^{\alpha}} \tag{5.99}$$

$$\hat{G}^c(-x_r\mu) = \frac{1 - \hat{g}(-x_r\mu)}{-x_r\mu}$$

$$= \frac{1}{-x_r\mu} - \frac{\beta^{\alpha}}{(-x_r\mu)(\beta - x_r\mu)^{a}}. \tag{5.100}$$

Pareto distribution

Pareto distribution is also a two-parameter family of continuous probability distributions. Let x_m be the scale parameter and a the shape parameter and the probability density function is

$$g(x) = \begin{cases} \frac{ax_m^a}{x^{a+1}} & \text{for } x \geq x_m \\ 0 & \text{for } x < x_m \end{cases}. \tag{5.101}$$

A transformed probability density function of the Pareto distribution, which is widely used in modeling communication networks, is defined as

$$f(y) = \frac{ax_m^a}{(x_m + y)^{a+1}}, \ y \geq 0 \tag{5.102}$$

where $y = x - x_m$.[1] So we have

$$g(x) = f(x - x_m), \ x \geq x_m \tag{5.103}$$

and

$$\begin{aligned} G(x) &= \int_0^x g(t)dt \\ &= \int_{x_m}^x g(t)dt \\ &= \int_{x_m}^x f(t - x_m)dt \text{ (Let } u \ = t - x_m) \\ &= \int_0^{x - x_m} f(u)du \end{aligned}$$

[1] x is the original Pareto random variable used in Equation 5.101 and y is the transformed Pareto random variable used in Equation 5.102.

$$= F(x - x_m),\ x \geq x_m, \tag{5.104}$$

where $G(\cdot)$ and $F(\cdot)$ are the cumulative distribution function of $g(\cdot)$ and $f(\cdot)$, respectively. Then the explicit expressions (in terms of well-known functions) for the Laplace transform of the transformed Pareto distribution can be represented as follows (refer to Nadarajah and Kotz [36]):

$$\mathcal{L}\left\{f(x)\right\}(s) = \hat{f}(s) = a(x_m s)^{(a-1)/2} \cdot e^{x_m s/2} \cdot W_{-(a+1)/2,-a/2}(x_m s), \tag{5.105}$$

where $W_{\lambda,\mu}(x)$ stands for the Whittaker W function, which is defined as

$$\begin{aligned} W_{\lambda,\mu}(x) = {} & \frac{x^{\mu+1/2} e^{-x/2}}{\Gamma(\mu - \lambda + 1/2)} \\ & \times \int_0^\infty t^{\mu-\lambda-1/2} (1+t)^{\mu+\lambda-1/2} e^{-xt} dt, \end{aligned} \tag{5.106}$$

where

$$\Gamma(x) = \int_0^\infty t^{x-1} e^{-t} dt \tag{5.107}$$

is the gamma function. It then follows Equation 5.95 that the Laplace transform of its complementary cumulative probability distribution is

$$\begin{aligned} \mathcal{L}\left\{F^c(y)\right\}(s) &= \frac{1 - \hat{f}(s)}{s} \\ &= \frac{1 - a(x_m s)^{(a-1)/2} \cdot e^{x_m s/2} \cdot W_{-(a+1)/2,-a/2}(x_m s)}{s}. \end{aligned} \tag{5.108}$$

In order to obtain the Laplace transform of the original Pareto probability density function, we apply the following property of Laplace transform (The time delay property):

$$\mathcal{L}\left\{f(x - x_m) \cdot U(x - x_m)\right\}(s) = e^{-x_m s} \mathcal{L}\left\{f(x)\right\}(s), \tag{5.109}$$

where $U(x - x_m)$ is the unit step function defined as

$$\begin{cases} U(x - x_m) = 1, & x \geq x_m \\ U(x - x_m) = 0, & x < x_m \end{cases} \tag{5.110}$$

on Equation 5.105 and yields

$$\begin{aligned} \mathcal{L}\left\{g(x)\right\}(s) &= \mathcal{L}\left\{f(x - x_m) \cdot U(x - x_m)\right\}(s) \\ &= e^{-x_m s} \mathcal{L}\left\{f(x)\right\}(s) \\ &= e^{-x_m s} \cdot a(x_m s)^{(a-1)/2} \cdot e^{x_m s/2} \cdot W_{-(a+1)/2, -a/2}(x_m s) \\ &= a(x_m s)^{(a-1)/2} \cdot e^{(x_m/2 - x_m)s} \cdot W_{-(a+1)/2, -a/2}(x_m s) \\ &= a(x_m s)^{(a-1)/2} \cdot e^{-x_m s/2} \cdot W_{-(a+1)/2, -a/2}(x_m s). \end{aligned} \tag{5.111}$$

Then Equation 5.95 applies and we have

$$\begin{aligned} \mathcal{L}\left\{G^c(x)\right\}(s) &= \frac{1 - \hat{g}(s)}{s} \\ &= \frac{1 - a(x_m s)^{(a-1)/2} \cdot e^{-x_m s/2} \cdot W_{-(a+1)/2, -a/2}(x_m s)}{s}. \end{aligned} \tag{5.112}$$

At last we can obtain the explicit expressions of $\hat{g}(-x_r \mu)$ and $\hat{G}^c(-x_r \mu)$ by replacing s with $(-x_r \mu)$ in Equation 5.111 and 5.112.

Capacity variation patterns

The capacity value process[2], which can be represented by a DTMC $\{C_n, n \geq 0\}$ with state space $\{i | i = 0, 1, 2, ..., C\}$ (see Section 5.4.1), can also affect the performance of the system significantly [50, 51]. Three different kinds of capacity variation patterns[3] are introduced

[2] It is important to distinguish between the *capacity process* (which is a CTMC denoted by $\{C(t)\}$) and the *capacity value processs* (which is a DTMC denoted by $\{C_n\}$.)

[3] In Sun and Williamson [50, 51] the capacity value was drawn from a Normal distribution with predetermined mean and standard deviation. The difficulty of using a Normal distribution is that we are generating positive integers from a continuous probability function that is defined on $(-\infty, \infty)$. Therefore we propose some other techniques to generate capacity values at capacity-change instants.

and the corresponding transition probabilities H_{ij}s (defined by Equation 5.67) are derived. Then some numerical examples are provided to study the characteristics of the capacity value processes under different capacity variation patterns.

Skip-free variation

A skip-free variation is the type of variation where the capacity can only change one unit at a time. In Luo and Williamson [32] the capacity value process $\{C_n, n \geq 0\}$ is assumed to be a skip-free process. Given that the current capacity is i, and at any capacity-change instant, the capacity can increase by one with probability $f_{\uparrow}(i)$ or decrease by one with probability $f_{\downarrow}(i)$. It is certain that $f_{\uparrow}(i) + f_{\downarrow}(i) = 1$. Also $f_{\uparrow}(0) = 1$ and $f_{\downarrow}(C) = 1$ because C is the maximum capacity of the system. The expression of H_{ij} for a skip-free capacity-change process can be expressed as

$$H_{ij} = \begin{cases} f_{\uparrow}(i) & \text{if } j = i+1 \\ f_{\downarrow}(i) & \text{if } j = i-1 \\ 0 & \text{otherwise} \end{cases} . \tag{5.113}$$

The probabilities $f_{\uparrow}(i)$ and $f_{\downarrow}(i)$ can be functions of i or some predetermined values such as 0.5 and 0.5.

Distance-based variation

The second capacity variation pattern is the distance-based variation. The idea of distance-based variation is that we allow the capacity to transit from its current value to any other value in $\{0, 1, 2, ..., C\}$ to increase the variability of the capacity value process. However we want to control the variability in such a way that the capacity is more likely to change to a value that is close to its current value than to a value that is distance away. Therefore, the probability that the capacity will change from its current value i to another value j should depend on the distance (defined as $|i-j|$) between i and j: the larger the distance is, the smaller the probability would be. The following derivation of H_{ij} explains how the distance-based transition probabilities are calculated.

1. Assume that the current capacity is i. Then the distance from i to another value j ($j \in [0, C]$ and $j \neq i$) is defined as $d_{ij} = |i - j|$.
2. Because we would like the transition probability H_{ij} to be inversely related to the distance d_{ij}, we define $r_{ij} = 1/d_{ij}$.
3. To meet the normalization condition we multiply r_{ij} by the constant $\left(\sum_{i \neq j} r_{ij}\right)^{-1}$. Therefore

$$H_{ij} = \begin{cases} \frac{r_{ij}}{\sum_{i \neq j} r_{ij}} & \text{if } i \neq j \\ 0 & \text{if } i = j \end{cases} . \tag{5.114}$$

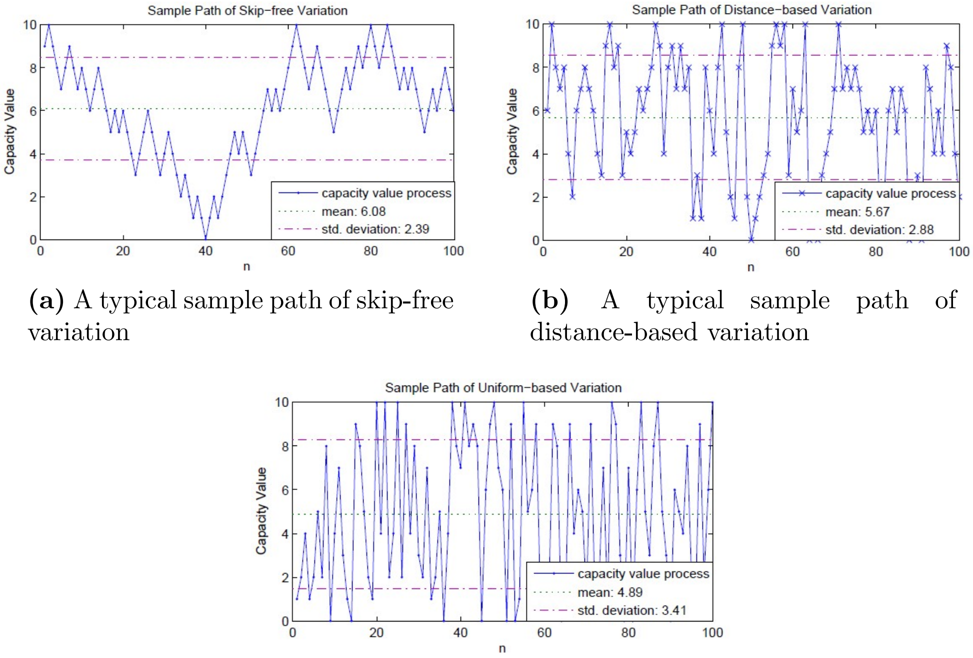
Uniform-based variation

In order to allow even more variability in the capacity value process we propose the uniform-based variation, where the system capacity can transit from its current value to other values in $0, 1, ...i - 1, i + 1, ..., C$ with equal probability. Therefore, at the capacity-change instant, the probability that the capacity will change from i to any other value in $0, 1, ...i - 1, i + 1, ..., C$ is always equal to $1/C$. The transition probabilities H_{ij} for this case is

$$H_{ij} = \begin{cases} 1/C & \text{if } i \neq j \\ 0 & \text{if } i = j \end{cases} . \tag{5.115}$$

Numerical examples of capacity variation patterns

In this section, numerical examples are used to study the characteristics of these three capacity variation patterns. Assume that $C = 10$ and the capacity value process starts at full capacity. For the skip-free variations we further assume that $f_{\uparrow}(i) = f_{\downarrow}(i) = 0.5$ when $i \neq 0$ or C. First, as plotted in Figure 5.3, a typical sample path consisting of 100 sample points was generated for each capacity variation patterns. As Figure 5.3 shows, the skip-free variation (Figure 5.3a) is characterized by a sample path with modest fluctuations. The sample path for the uniform-based variation fluctuates more dramatically. The variability of distance-based variation is in the middle. From the simulated autocorrelation plots in Figure 5.4 it is clear that the series of skip-free variations has the strongest autocorrelation; because lag-1 has high autocorrelation and slowly declines and goes towards negative autocorrelation. The series of distance-based variations has weaker autocorrelation than those of skip-free variations as the autocorrelation starts high at lag-1 but decreases quickly and reaches negative values at lag-5. The series of uniform-based variations is the most random series since almost all the autocorrelation lie within the confidence limits and there is no apparent pattern in the correlation.



(a) A typical sample path of skip-free variation

(b) A typical sample path of distance-based variation

(c) A typical sample path of uniform-based variation

Figure 5.3: Typical sample paths of different types of capacity variation process

5.4.2 Numerical study

Analytical solution vs. simulated solution

In this section, call-level simulation was used to verify analytic solutions. The solutions based on the MRGP method were calculated in Matlab. Our experiments covered all 9 different combinations of the distribution of capacity interchange times and the capacity variation

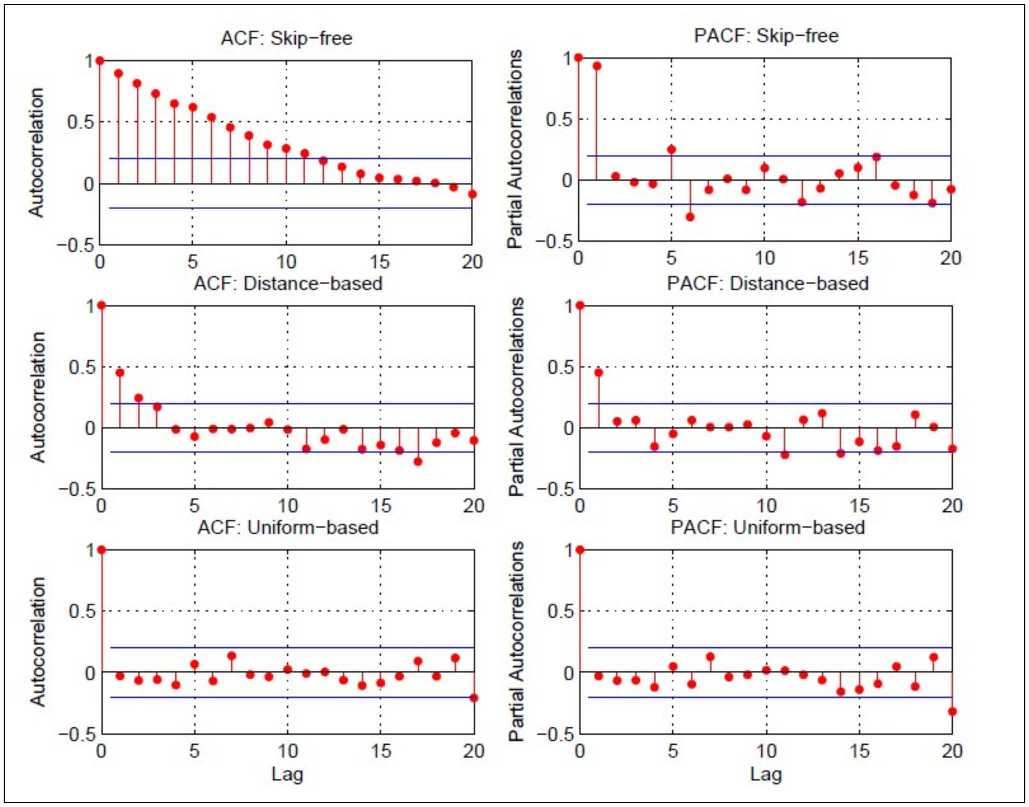


Figure 5.4: Simulated autocorrelation and partial autocorrelation function plots for three different types of variations

patterns. For each experiment, considering a reference cell with $n = 10$ total channels, 10 simulation runs were performed and then one sample T-tests were carried out to test the null hypothesis that the sample mean of each of the performance measures produced by the 10 simulation runs is equal to the solution calculated using the MRGP method. The model parameters used for our experiments are: $\lambda = 5$ and $\mu = 1$, and the capacity-change rate λ_c can vary from 0.1 to 10. Results displayed in Figure 5.5 suggest that the analytic solutions are well supported by simulated solutions. Detailed results are also provided in Tables A.1 - A.9 in Appendix.

The impact of the distributions of capacity interchange times and capacity variation patterns on performance metrics

We have introduced three different distributions of capacity interchange times as well as three different capacity variation patterns. Experiments are carried out to investigate their impact on performance metrics.

The performance metrics were calculated using the MRGP method developed in Section 5.4. We fixed the total number of channels, n, to be 10 and the call completion rate, μ, to be 1. For the skip-free variation we assume that $f_{\uparrow}(i) = 0.5$ for $0 < i < C$. The remaining model parameters can vary for different experiments.

We then conducted two experiments. The first experiment focused on the effect of offered load on performance metrics. In this experiment, the offered load (λ/μ) varied from 1 (low traffic load) to 20 (overload). The mean capacity-change rate $\lambda_c = 1$. The three distributions of capacity interchange times under study were: exponential ($\lambda = 1$), gamma ($\alpha = 10$, $\beta = 10$), and Pareto ($a = 1.2$, $x_m = 1/6$). Among which, the Pareto ($a = 1.2$, $x_m = 1/6$) distribution had the heaviest tail and the gamma($\alpha = 10$, $\beta = 10$) distribution had the lightest tail.

Figure 5.6a plots blocking probabilities against offered load, and a clear trend can be observed: The blocking probability increases as offered load increases for all types of capacity variation patterns. The distribution of capacity interchange times has little impact on the blocking probability; because under the same capacity variation pattern, the call blocking curves for different distributions of capacity interchange times overlay with each other.

The capacity variation patterns, on the other hand, have impact on the blocking probability to some extent: The capacity variation pattern with less dramatic fluctuations (skip-free) is able to produce lower blocking probability than capacity variation patterns with more dramatic fluctuations (distance-based and uniform-based) at lower offered load (< 5). However, the relationship is reversed at higher offered load (> 5): the distance-based and uniform-based variations produce the lower blocking probabilities and the skip-free variation produces the highest blocking probability. The reason is that the fluctuation of capacity has the effect of reducing call blocking probability by first clearing the system (when capacity decreases and ongoing calls are dropped) and then producing free channels for incoming calls (when the capacity increases). When the offered load is higher and when the capacity fluctuates more dramatically, this effect is more significant; therefore, capacity variation patterns with more dramatic fluctuations (distance-based and uniform-based) produce lower blocking probabilities than the capacity variation pattern with less dramatic fluctuations (skip-free).

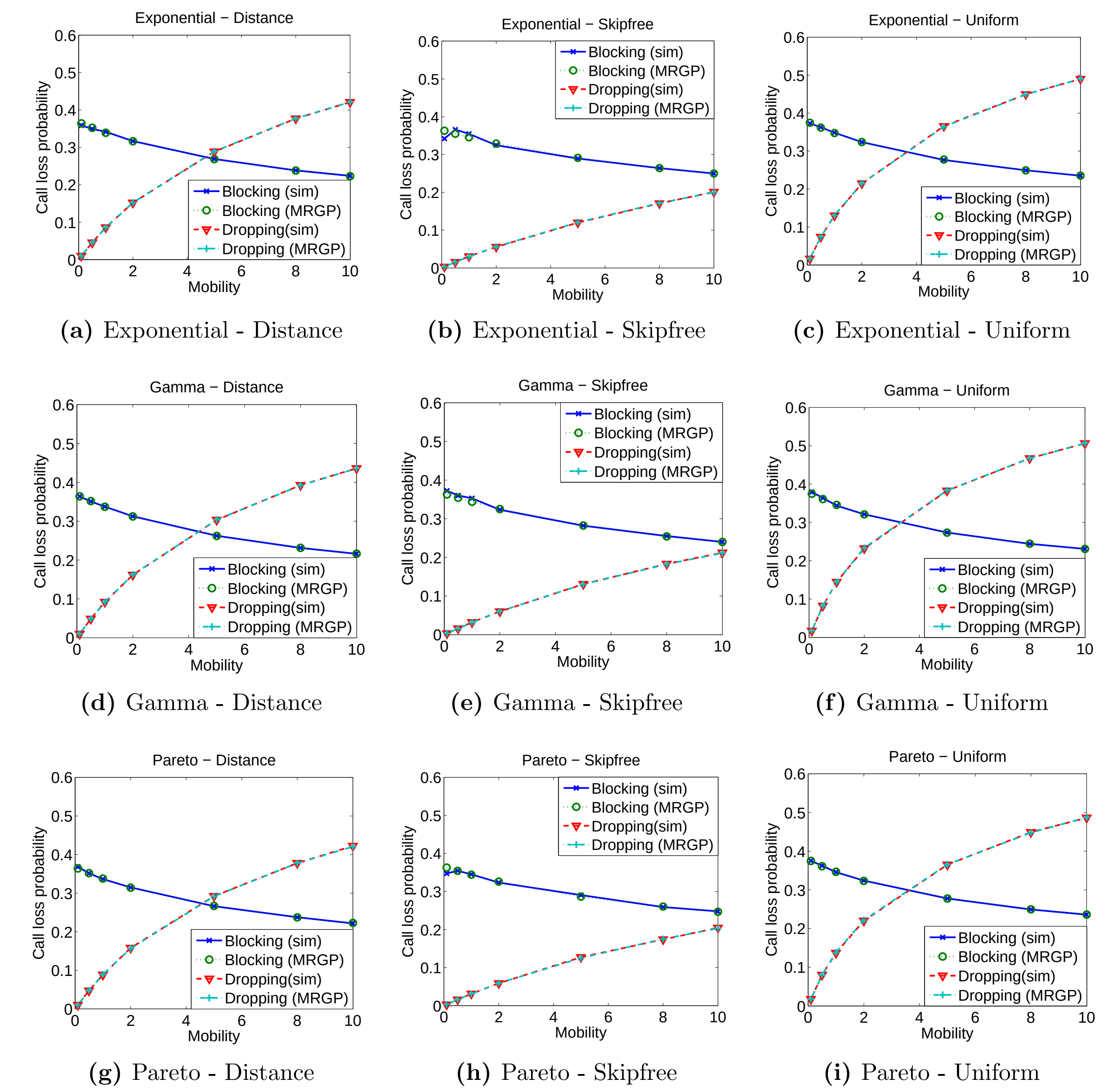
In Figure 5.6b, dropping probabilities are plotted against offered load, and several interesting patterns are displayed. First, the dropping probabilities of different capacity variation patterns are quite different: Uniform-based variation produces the highest dropping probability (because its capacity fluctuates the most dramatically). The skip-free variation achieves the lowest dropping probability (because its capacity fluctuates the least dramatically). The dropping probability produced by distance-based variation is in between. Second, for skip-free variation, dropping probability decreases as offered load increases. For uniform-based and distance-based variations, the dropping probabilities vary nonmonotonically as offered load increases. As a matter of fact, the dropping probabilities produced by uniform-based and distance-based variations increase first and start to decrease after reaching the maxima (which occurs at offered load = 6 Erlangs). At first glance it seems that our results contradicts to what was presented in Sun and Williamson [50], where the authors draw the conclusion through simulation studies that call dropping ratio should increase as offered load increases. However, after examining closely the parameters they were using, we found out that the ”contradiction” may be caused by insufficient data in their experiments. Note that the capacity variation process in Sun and Williamson [50] had a mean capacity of 40 while the offered load only varied from 20 to 60 (which is 150% of the mean capacity). In our experiment, with the mean capacity being about 5 for both uniform-based and distance-based variations, we let offered load vary from 1 to 20 (which is 400% of the mean capacity) and were able to detect the decreasing portions of the dropping probability curves. Third, for the most dramatic capacity variation pattern, that is, the uniform-based variation, we are able to see the difference in dropping probabilities between different distributions of capacity interchange times: for the gamma distribution (which has the lightest tail) we see the highest dropping probability, whereas for the Pareto distribution (which has the heaviest tail) we observe the lowest dropping probability. The dropping probability of the exponential distribution is in between.

In the second experiment we studied the effect of λ_f on performance metrics. λ_f, the relative scale of capacity fluctuation, was defined by Luo and Williamson [32] as the ratio of λ_c to μ. In this experiment we fixed n to be 10 and μ to be 1. We then chose a medium offered load of 5 and let λ_f varies from 0.05 to 20. When λ_f = 0.05 capacity-change events occur less frequently than call completion events, and when λ_f reaches 1, capacity-change events occur as frequently as call completion events. When λ_f is greater than 1, capacity-change events occur more frequently than call completion events and is expected to have more significant impact on call loss probabilities. This is well supported by the results displayed in Figures 5.7a and 5.7b: As λ_f increases, the differences in both call blocking and dropping probabilities between different distributions of capacity interchange times become more observable.

In Figure 5.7a, blocking probabilities are plotted against λ_f. Blocking probabilities decrease as λ_f increases. Also the distribution with heavier tail (Pareto) produces higher blocking probabilities than the distribution with lighter tail (gamma). Last but not the least, we notice that lower blocking probabilities are often associated with capacity variation patterns that fluctuate more dramatically (distance-based and uniform-based); and higher blocking probabilities are associated with the capacity variation pattern that fluctuates less dramatically (skip-free variation).

In Figure 5.7b, dropping probabilities are plotted against λ_f and opposite patterns to Figure 5.7a are displayed. First, the dropping probabilities increases as λ_f increases. Second, for the same capacity variation pattern, high dropping probabilities are observed for distributions associated with low blocking probabilities (gamma and exponential); and low dropping probabilities are observed for the distribution that is associated with high blocking probability (Pareto). Finally, capacity variation patterns that fluctuate more dramatically (distance and uniform-based) produce high dropping probability, whereas the capacity variation pattern that fluctuates modestly (skip-free) leads to low dropping probability, which is expected.

To summarize, distributions of capacity interchange times and capacity variation patterns have great impact on call blocking and dropping probabilities. For instance, distributions with a lighter tail, and capacity variation patterns with higher variability (i.e., can vary more dramatically at the capacity-change instant) can lead to higher dropping probabilities. Furthermore, high dropping probability is usually associated with low blocking probability. An intuitive explanation would be: When capacity decreases by a significant number of channels, it also terminates many ongoing calls; therefore, the system has more free channels to accommodate incoming calls when the capacity increases. The average time for calls to spend in the system is reduced due to call droppings. This effect is defined as the *dropping-induced speed-up effect* by Luo and Williamson in Sun and Williamson [50]. This speed-up effect is more significant when the change of capacity happens more frequently than the completion of calls, that is, when the rate of capacity-change is higher than the rate of call completion.

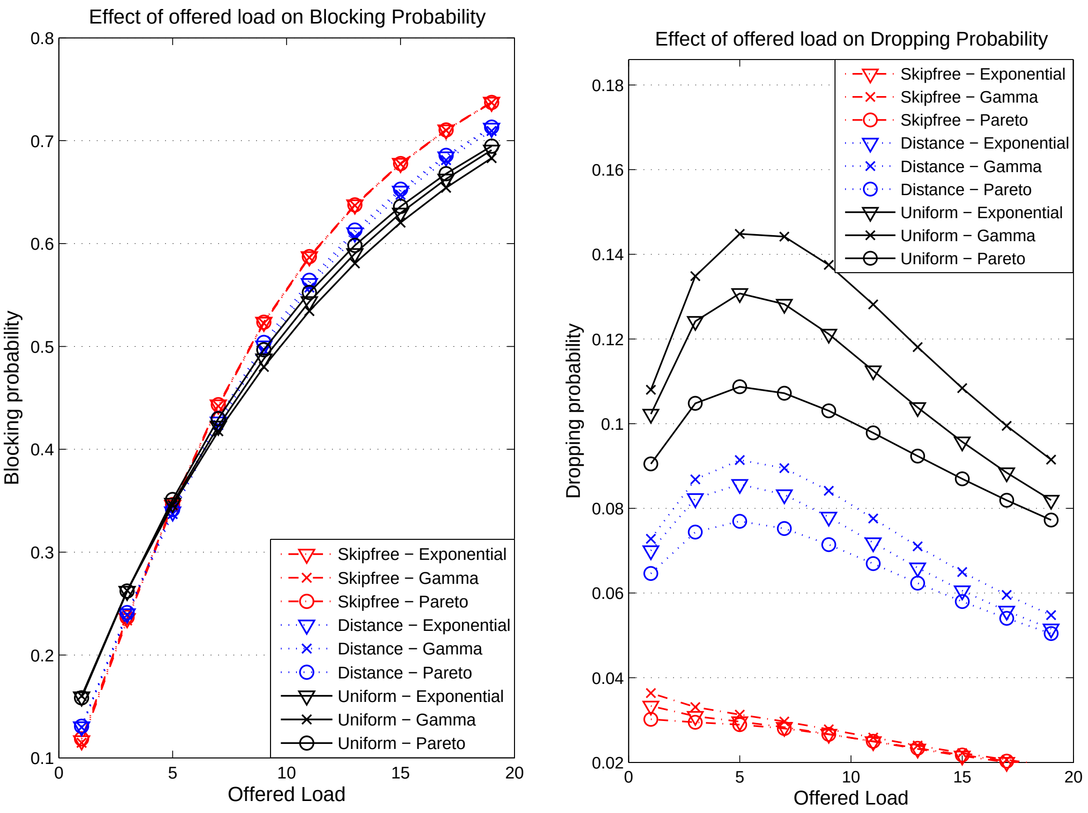


(a) Exponential - Distance **(b)** Exponential - Skipfree **(c)** Exponential - Uniform

(d) Gamma - Distance **(e)** Gamma - Skipfree **(f)** Gamma - Uniform

(g) Pareto - Distance **(h)** Pareto - Skipfree **(i)** Pareto - Uniform

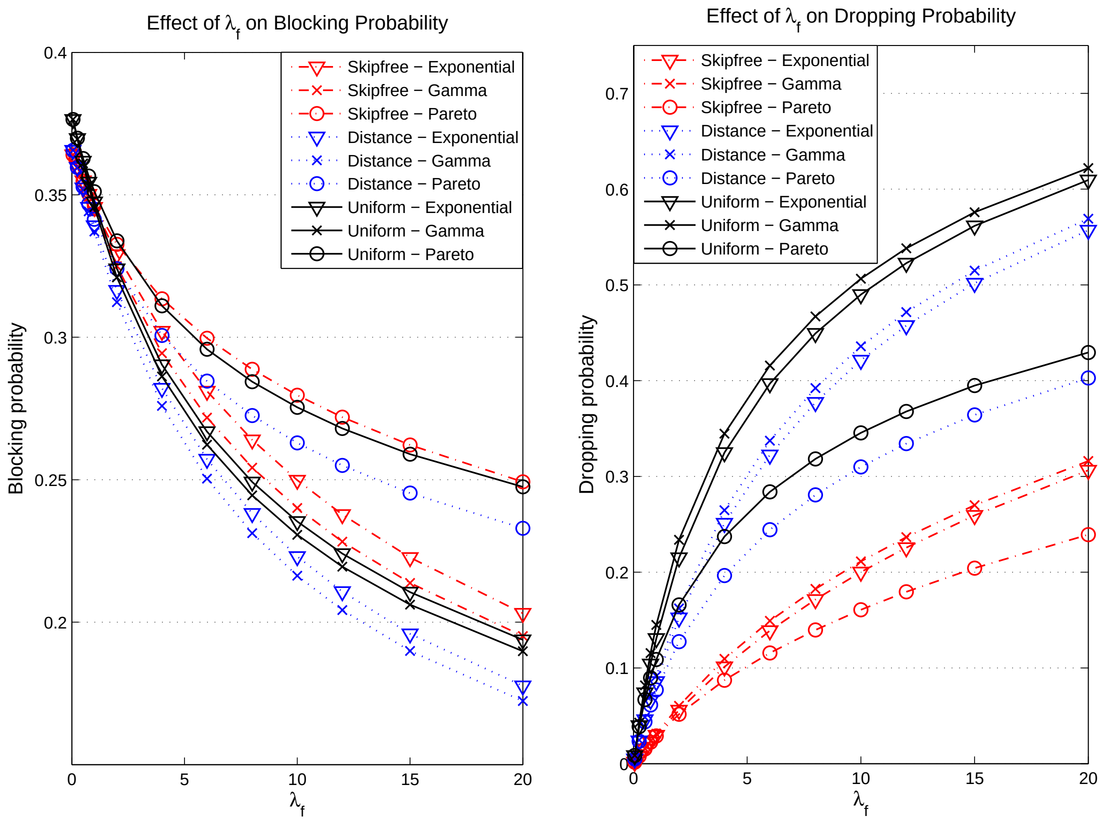
Figure 5.5: Verify the method of MRGP using simulations



(a) Blocking probability

(b) Dropping probability

Figure 5.6: Effect of offered load on call loss. λ varies from 1 to 20, $\mu = 1$, $\lambda_c = 1$ and $n = 10$. a): Blocking probabilities are plotted against offered load. b): Dropping probabilities are plotted against offered load.



(a) Blocking probability

(b) Dropping probability

Figure 5.7: Effect of λ_f on call loss. λ_f is defined as the ratio of mean capacity-change rate (λ_c) to call departure rate (μ). $\lambda = 5$, $\mu = 1$, $n = 10$ and λ_f varies from 0.05 to 20. a): Blocking probabilities are plotted against λ_f. b): Dropping probabilities are plotted against λ_f.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

The explosive growth of cellular networks has attracted many researchers to study the technology from various perspectives. An important characteristic of cellular networks is the stochastically fluctuating system capacity, which can have significant impact on system performance. Therefore, the study of systems with fluctuating capacity is of great interest. In this thesis, we first studied priority queueing systems by proposing and analyzing two guard channel models with controlled preemption. Then the $M/M/\tilde{}C/\tilde{}C$ system was directly analyzed using the MRGP method. Our contributions are described below:

- We developed two analytic methods and two approximate methods to analyse the performance of our first guard-channel model (the M1 model). The four methods were compared to simulation results. Approximate methods took negligible time to finish; but they overestimated the call loss probabilities at high offered load. On the other hand, two of the analytic solutions agreed very well with simulation results; but they took a substantial amount of time to run. Therefore, approximate methods can be used when the number of channels is large (i.e., $n \geq 120$) and/or the computational power is limited; otherwise, the analytic methods are recommended. Algorithms were developed to find an optimal number of total channels (n) and guard channels (g) to meet given call performance thresholds.
- For the second proposed guard channel model (the M2 model), closed expressions of the call loss probabilities were derived when call holding times for both traffic types are homogeneous. The property of the new call dropping probability was studied through the investigation of its first partial derivative. The results showed that when g was

fixed, the new call dropping probability can be a non-monotonic function of n (Figure 3.1). Then the contours of loss probabilities were examined and algorithms for solving optimization problems were developed based on the different patterns of the contour plot. In the last section of Chapter 3, the M2 model was compared to the HT's model and results showed that the M2 model required about 10% fewer channels on average to meet performance constraints when the performance constraint on high-priority traffic was much stricter than on low-priority traffic

- In Chapter 4, a series of numerical experiments were conducted to thoroughly compare four models at hand (M1, M2, OM, and HT's models). The following characteristics were compared between models: (i) channel utilization, (ii) low priority call (i.e., new call) performance, and (iii) flexibility to meet various constraints. The results suggested that the proposed controlled preemption models were the best models overall; this is because they use channels more efficiently than the non-preemption model, and they have more flexibility than fully preemption model.
- In Chapter 5 a loss system with stochastic capacity (the $M/M/\tilde{}C/\tilde{}C$ system) was studied using the MRGP method in which three different distributions (exponential, gamma, and Pareto) of the capacity interchange times and three different capacity variation patterns (skip-free, distance-based, and uniform-based) were considered. Explicit expressions for call blocking and dropping probabilities were obtained and were verified by call-level simulations. Further numerical results showed that the blocking probability in this system was affected by different distributions of capacity interchange times and capacity variation patterns. Especially when the ratio of mean capacity change rate to call departure change rate, λ_f, was greater then 2, the effects of different distributions and capacity variation patterns begin to aggregate: the gamma distribution (which has the lightest tail among all the three distributions under study) produced the lowest blocking probability; and the capacity variation patterns with more variability (distance-based and uniform-based variation) produced lower blocking probabilities than the capacity variation pattern with limited variability (skip-free variation).

6.2 Future work

Three projects that could extend the work done in this thesis and improve the technology of cellular networks are described.

1. Zhou and Beard [68] also proposed an MWCP which deals with a delay system that supports three classes of traffic: new calls from public users, handoff calls from public users, and the emergency calls (in order of low to high priority). In Zhou and Beard's model, when an incoming emergency call fails to find a free channel, and the number of active emergency calls is within a predetermined limit, the incoming emergency call can preempt an ongoing public call. The difference between Zhou and Beard's model and the M1 model is that in Zhou and Beard's model, although the high priority traffic (emergency calls) can only preempt low priority traffic (public calls) when the system is full and when the number of active emergency calls is within a predetermined limit (which is similar to the M1 model), the emergency calls can also access free channels when the system is not full even when the number of active emergency calls is over the limit, which is not allowed in the M1 model. It would be interesting to see how the call loss probabilities of Zhou's model compares to those of the M1 and M2 models.
2. Recently, studies that question the validity of the assumption of handoff arrival being Poissonian have appeared in the literature Chlebus and Ludwin [9], Rajaratnam and Takawira [40, 41]. The MRGP method could be used to analyze the M1 model with generally distributed handoff call interarrival times.
3. Recently, 4G cellular networks have started supporting high speed transmission of multimedia traffic, including video, audio, and text. One important extension to our proposed guard channel models would be to apply them in a system carrying multiclass traffic, in which each class of traffic originates from either the cell under study (new traffic) or from any one of the neighbouring cells of the reference cell (handoff traffic).

References

[1] P. Agrawal, D. K. Anvekar, and B. Narendran. Channel management policies for handovers in cellular networks. *Bell Labs Technical Journal*, 1(2):97–110, 1996.

[2] H. Akimaru and K. Kawashima. *Teletraffic: Theory and Applications.* Heidelberg, Germany: Springer-Verlag, 1993.

[3] M. Alam and V. Mani. Recursive solution technique in a multi-server bi-level queueing system with server breakdowns. *Reliability, IEEE Transactions on*, 38(4):416–421, 1989.

[4] F. Barcelo. Performance analysis of handoff resource allocation strategies through the state-dependent rejection scheme. *Wireless Communications, IEEE Transactions on*, 3 (3):900–909, 2004.

[5] U. Black. *Mobile and wireless networks.* Upper Saddle River, NJ: Prentice Hall, 1996.

[6] U. Black. *Second generation mobile and wireless networks.* Upper Saddle River, NJ: Prentice Hall, 1999.

[7] J. P. Buzen and A. B. Bondi. The response times of priority classes under preemptive resume in $M/M/m$ queues. *Operations Research*, 31(3):456–465, 1983.

[8] D. Calabrese, M. Fischer, B. Hoiem, and E. Kaiser. Modeling a voice network with preemption. *Communications, IEEE Transactions on*, 28(1):22–27, 1980.

[9] E. Chlebus and W. Ludwin. Is handoff traffic really poissonian? In *Proceedings of the Fourth IEEE International Conference on Universal Personal Communications*, pages 348–353, 1995.

[10] Y. Z. Cho and C. K. Un. Analysis of the $M/G/1$ queue under a combined preemptive/nonpreemptive priority discipline. *Communications, IEEE Transactions on*, 41(1): 132–141, 1993.

[11] H. Daehyoung and S. S. Rappaport. Traffic model and performance analysis for cellular mobile radio telephone systems with prioritized and nonprioritized handoff procedures. *Vehicular Technology, IEEE Transactions on*, 35(3):77–92, 1986.

[12] S. Dharmaraja, K. S. Trivedi, and D. Logothetis. Performance modeling of wireless networks with generally distributed handoff interarrival times. *Computer Communications*, 26(15):1747 – 1755, 2003.

[13] N. Ekiz, T. Salih, S. Kucukoner, and K. Fidanboylu. An overview of handoff techniques in cellular networks. *International Journal of Information Technology*, 2(3):132–136, 2005.

[14] M. J. Fischer. Priority loss systems-unequal holding times. *American Institute Industrial Engineers Transactions*, 12(1):47–53, 1980.

[15] J. A. Garay and I. S. Gopal. Call preemption in communication networks. In *Proceedings of INFOCOM '92: Eleventh Annual Joint Conference of the IEEE Computer and Communications Societies*, pages 1043–1050, 1992.

[16] K. S. Gilhousen, I. M. Jacobs, R. Padovani, A. J. Viterbi, Jr. Weaver, L. A., and III Wheatley, C. E. On the capacity of a cellular cdma system. *Vehicular Technology, IEEE Transactions on*, 40(2):303–312, 1991.

[17] G. Harine, R. Marie, R. Puigjaner, and K. Trivedi. Loss formulas and their application to optimization for cellular networks. *Vehicular Technology, IEEE Transactions on*, 50 (3):664–673, 2001.

[18] W. Helly. Letter to the editor - two doctrines for the handling of two-priority traffic by a group of n servers. *Operations Research*, 10(2):268–269, 1962.

[19] U. Herzog, L. Woo, and K. M. Chandy. Solution of queuing problems by a recursive technique. *IBM Journal of Research and Development*, 19(3):295–300, 1975.

[20] R. V. Hogg and A. T. Craig. *Introduction to Mathematical Statistics.* Pearson, 7th edition, 2012.

[21] B. Hwang, I. Hwang, and C. Shen. Performance evaluation of multi-guard channel schemes in broadband mobile networks. In *Proceedings of the International Conference on Mobile Technology, Applications, and Systems*, pages 19:1–19:7, 2008.

[22] J. P. Ignizio. *Goal Programming and Extensions.* Lexington books, 1976.

[23] D. L. Jagerman. Some properties of the erlang loss function. *Bell System Tech. J*, 53 (3):525–551, 1974.

[24] Y. Kim, D. Lee, B. Lee, Y. Kim, and B. Mukherjee. Dynamic channel reservation based on mobility in wireless atm networks. *Communications Magazine, IEEE*, 37(11):47–51, 1999.

[25] L. Kleinrock. *Queueing systems. volume 1: Theory.* Wiley-Interscience, 1975.

[26] L. Kleinrock. *Queueing systems: volume 2: computer applications.* Wiley-Interscience, 1976.

[27] V. G. Kulkarni. *Modeling and analysis of stochastic systems.* Boca Raton: Chapman & Hall, 2nd ed edition, 2010.

[28] A. E. Leu and B. L. Mark. Modeling and analysis of fast handoff algorithms for microcellular networks. pages 321–328, 2002.

[29] W. Li, C. Hang, and D. P. Agrawal. Performance analysis of handoff schemes with preemptive and nonpreemptive channel borrowing in integrated wireless cellular networks. *Wireless Communications, IEEE Transactions on*, 4(3):1222–1233, 2005.

[30] Z. Lian and N. Zhao. A two-stage $M/G/1$ queue with discretionary priority. In *Proceedings of the 2011 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, pages 1402–1406, 2011.

[31] D. Logothetis, K. S. Trivedi, and A. Puliafito. Markov regenerative models. In *Proceedings of the Computer Performance and Dependability Symposium*, pages 134–142, 1995.

[32] J. Luo and C. Williamson. Performance implications of fluctuating server capacity. *Comput. Commun.*, 31(16):3760–3770, 2008.

[33] P. Marichamy, S. Chakrabarti, and S. L. Maskara. Overview of handoff schemes in cellular mobile networks and their comparative performance evaluation. 3:1486–1490, 1999.

[34] W. A. Massey and R. Srinivasan. A packet delay analysis for cellular digital packet data. *Selected Areas in Communications, IEEE Journal on*, 15(7):1364–1372, 1997.

[35] D. R. Miller. Computation of steady-state probabilities for $M/M/1$ priority queues. *Operations Research*, 29(5):945–958, 1981.

[36] S. Nadarajah and S. Kotz. On the Laplace transform of the Pareto distribution. *Queueing Systems*, 54:243–244, 2006.

[37] S. Oh and D. Tcha. Prioritized channel assignment in a cellular radio network. *Communications, IEEE Transactions on*, 40(7):1259–1269, 1992. ISSN 0090-6778.

[38] M. Peyravian and A. D. Kshemkalyani. Connection preemption: issues, algorithms, and a simulation study. In *Proceedings of INFOCOM '97: Sixteenth Annual Joint Conference of the IEEE Computer and Communications Societies*, pages 143–151, 1997.

[39] G. P. Pollini. Trends in handover design. *Communications Magazine, IEEE*, 34(3): 82–90, 1996.

[40] M. Rajaratnam and F. Takawira. Hand-off traffic modelling in cellular networks. In *Proceedings of GLOBECOM '97: IEEE Global Telecommunications Conference*, pages 131–137, 1997.

[41] M. Rajaratnam and F. Takawira. Nonclassical traffic modeling and performance analysis of cellular mobile networks with and without channel reservation. *Vehicular Technology, IEEE Transactions on*, 49(3):817–834, 2000.

[42] P. Ramanathan, K. M. Sivalingam, P. Agrawal, and S. Kishore. Dynamic resource allocation schemes during handoff for mobile multimedia wireless networks. *Selected Areas in Communications, IEEE Journal on*, 17(7):1270–1283, 1999.

[43] T. S. Rappaport. *Wireless communications - Principles and practice.* Upper Saddle River, NJ: Prentice Hall, 2002.

[44] J. Riordan. *Stochastic service systems.* New York: Wiley, 1962.

[45] M. Salamah. An adaptive multi-guard channel scheme for multi-class traffic in cellular networks. pages 716–723, 2006.

[46] D. Shen and C. Ji. Capacity trade-offs for heterogeneous traffic over large scale fading channels in cdma networks. 6:1718–1722, 2001.

[47] S. R. Somagari and H. K. Pati. An analytical model for adaptive multi-guard channel scheme for multi-class traffic in cellular networks with reduced handoff drop probabilities. *Procedia Technology*, 6(0):690 – 697, 2012.

[48] W. Stallings. *Wireless communicaions and networks.* Upper Saddle River, NJ: Prentice Hall, 2002.

[49] V. Stanisic and M. Devetsikiotis. A dynamic study of providing quality of service using preemption policies with random selection. In *Proceedings of ICC '03: IEEE International Conference on Communications*, pages 1543–1546, 2003.

[50] H. Sun and C. Williamson. Simulation evaluation of call dropping policies for stochastic capacity networks. In *Proceedings of the SCS SPECTS*, 2005.

[51] H. Sun and C. Williamson. On effective capacity in time-varying wireless networks. In *Proceedings of the SCS SPECTS*, page 111, 2006.

[52] H. Sun, S. Dharmaraja, C. Williamson, and V. Jindal. An analytical model for wireless networks with stochastic capacity. In *Proceedings of the SCS SPECTS*, 2007.

[53] E. J. Sung, R. T. Abler, and A. E. Goulart. The optimal connection preemption algorithm in a multi-class network. In *Proceedings of ICC 2002: IEEE International Conference on Communications*, pages 2294–2298, 2002.

[54] R. Syski. *Congestion Theory in Telephone Systems.* Oliver and Boyd, 1960.

[55] S. Tekinay and B. Jabbari. Handover and channel assignment in mobile cellular networks. *Communications Magazine, IEEE*, 29(11):42–46, 1991.

[56] S. Tekinay and B. Jabbari. A measurement-based prioritization scheme for handovers in mobile cellular networks. *Selected Areas in Communications, IEEE Journal on*, 10 (8):1343–1350, 1992.

[57] N. D. Tripathi, J. H. Reed, and H. F. VanLandinoham. Handoff in cellular systems. *Personal Communications, IEEE*, 5(6):26–37, 1998.

[58] K. S. Trivedi, X. Ma, and S. Dharmaraja. Performability modelling of wireless communication systems. *International Journal of Communication Systems*, 16(6):561–577, 2003.

[59] J. Wang, Q. Zeng, and D.P. Agrawal. Performance analysis of a preemptive and priority reservation handoff scheme for integrated service-based wireless mobile networks. *Mobile Computing, IEEE Transactions on*, 2(1):65–75, 2003.

[60] H. White and L. S. Christie. Queuing with preemptive priorities or with breakdown. *Operations Research*, 6(1):79–95, 1958.

[61] Y. Wu and C. Williamson. Impact of data call characteristics on multi-service cdma system capacity. *Performance Evaluation*, 62:83–99, 2005.

[62] L. Yi, S. Mohan, and A. Noerpel. Queueing priority channel assignment strategies for pcs hand-off and initial access. *Vehicular Technology, IEEE Transactions on*, 43(3): 704–712, 1994.

[63] O. T. W. Yu and V. C. M. Leung. Adaptive resource allocation for prioritized call admission over an atm-based wireless pcn. *Selected Areas in Communications, IEEE Journal on*, 15(7):1208–1225, 1997.

[64] J. Zhang and I. Stojmenovic. *Handbook on Security*, volume 1, chapter 45, pages 654 – 663. Wiley, 2005.

[65] Y. Zhang and D. Liu. An adaptive algorithm for call admission control in wireless networks. 6:3628–3632, 2001.

[66] Z. Zhao, S. Weber, and J. C. de Oliveira. Preemption rates for a parallel link loss network. *Performance Evaluation*, 66(1):21 – 46, 2009.

[67] J. Zhou and C. C. Beard. Tunable preemption controls for a cellular emergency network. In *Proceedings of WCNC'2007: Wireless Communications and Networking Conference*, pages 3647–3652, 2007.

[68] J. Zhou and C. C. Beard. A controlled preemption scheme for emergency applications in cellular networks. *Vehicular Technology, IEEE Transactions on*, 58(7):3753–3764, 2009.

Appendix

A. Numerically stable methods for computing steady state probabilities (recursive methods for Erlang)

A recursion method proposed in [2] to avoid overflow problems when calculating Erlang B formula:

$$EB(A,k) = \frac{\frac{A}{k}EB(A,k-1)}{1+\frac{A}{k}EB(A,k-1)},\ k = 1,2,...,N \tag{A.1}$$

with $EB(A,0) = 1$.

B. Proof of the pattern of recursive solution to the 1st handoff model

Now Let us prove Equation 2.91:

$$C^r_{(n,j+1)} + C^r_{(n-1,j+1)} = \frac{\rho_1^{j+1}}{(j+1)!}, r = 1,2 \text{ and } j = 0,1,2,...,n-1.$$

Proof. The L.H.S of the above equation is

$$\begin{aligned}
& C^r_{(n,j+1)} + C^r_{(n-1,j+1)} \\
= & \frac{\alpha\rho_1 + \rho_2 + j\alpha}{(j+1)\alpha}C^r_{(n,j)} - \frac{\rho_1}{(j+1)}C^r_{(n,j-1)} - \frac{1}{(j+1)\alpha}C^r_{(n-1,j)} \\
& + \frac{\alpha\rho_1 + 1 + j\alpha}{(j+1)\alpha}C^r_{(n-1,j)} - \frac{\rho_1}{j+1}C^r_{(n-1,j-1)} - \frac{\rho_2}{(j+1)\alpha}C^r_{(n,j)} \\
= & \left(\frac{\alpha\rho_1 + \rho_2 + j\alpha}{(j+1)\alpha} - \frac{\rho_2}{(j+1)\alpha}\right)C^r_{(n,j)} - \frac{\rho_1}{(j+1)}C^r_{(n,j-1)} \\
& + \left(\frac{-1}{(j+1)\alpha} + \frac{\alpha\rho_1 + 1 + j\alpha}{(j+1)\alpha}\right)C^r_{(n-1,j)} - \frac{\rho_1}{j+1}C^r_{(n-1,j-1)}
\end{aligned}$$

$$
\begin{aligned}
&= \frac{\rho_1 + j}{(j+1)} C^r_{(n,j)} - \frac{\rho_1}{(j+1)} C^r_{(n,j-1)} + \frac{\rho_1 + j}{(j+1)} C^r_{(n-1,j)} - \frac{\rho_1}{j+1} C^r_{(n-1,j-1)} \\
&= \frac{\rho_1 + j}{(j+1)} \left(C^r_{(n,j)} + C^r_{(n-1,j)} \right) - \frac{\rho_1}{(j+1)} \left(C^r_{(n,j-1)} + C^r_{(n-1,j-1)} \right) \\
&= \frac{\rho_1 + j}{(j+1)} \frac{\rho_1^j}{j!} - \frac{\rho_1}{(j+1)} \frac{\rho_1^{j-1}}{(j-1)!} \\
&= \frac{\rho_1 + j}{(j+1)} \frac{\rho_1^j}{j!} - \frac{j}{(j+1)} \frac{\rho_1^j}{j!} \\
&= \frac{\rho_1^{j+1}}{(j+1)!}
\end{aligned}
$$

which is just the R.H.S. of the equation 2.91. ■

C. Laplace transform of Gamma CDF

The cumulative distribution function of Gamma distribution can be witten as

$$
\begin{aligned}
G(x, \alpha, \beta) &= \int_0^x g(t; \alpha, \beta) dt \\
&= \int_0^x \beta^\alpha \frac{1}{\Gamma(\alpha)} t^{\alpha-1} e^{-t\beta} dt \text{ (Let } y = \beta t) \\
&= \frac{1}{\Gamma(\alpha)} \int_0^{x\beta} y^{\alpha-1} e^{-y} dy \\
&= \frac{\gamma(a, x\beta)}{\Gamma(\alpha)} \qquad \text{(A.2)}
\end{aligned}
$$

where $\Gamma(\alpha)$ is the gamma function and

$$\gamma(a, x) = \int_0^x t^{\alpha-1} e^{-t} dt \qquad \text{(A.3)}$$

is the lower incomplete gamma function. The upper incomplete gamma function is given by:

$$\Gamma(a, x) = \Gamma(a) - \gamma(a, x) \qquad \text{(A.4)}$$

and its Laplace transform is

$$\mathcal{L}\{\Gamma(a, t)\}(s) = \Gamma(a) \frac{1 - (1+s)^{-a}}{s}. \qquad \text{(A.5)}$$

Take Laplace transform of Equation A.4 we have

$$\begin{aligned}
\mathcal{L}\{\Gamma(a,t\}(s) &= \mathcal{L}\{\Gamma(a)-\gamma(a,t)\}(s) \\
&= \mathcal{L}\{\Gamma(a)\}(s)-\mathcal{L}\{\gamma(a,t)\}(s) \\
&= \int_0^\infty e^{-st}\Gamma(a)dt-\mathcal{L}\{\gamma(a,t)\}(s) && \text{(A.6)} \\
\Rightarrow \Gamma(a)\frac{1-(1+s)^{-a}}{s} &= \frac{\Gamma(a)}{s}-\mathcal{L}\{\gamma(a,t)\}(s) && \text{(A.7)} \\
\mathcal{L}\{\gamma(a,t)\}(s) &= \frac{\Gamma(a)}{s}(1+s)^{-a} && \text{(A.8)}
\end{aligned}$$

Now we are ready to calculate the Laplace transform of the Gamma cumulative distribution function $G(x,\alpha,\beta)$:

$$\begin{aligned}
\mathcal{L}\{G(x,\alpha,\beta)\}(s) &= \mathcal{L}\{\frac{\gamma(a,x\beta)}{\Gamma(\alpha)}\}(s) \\
&= \int_0^\infty e^{-sx}\frac{\gamma(a,x\beta)}{\Gamma(\alpha)}dx \\
&= \frac{1}{\Gamma(\alpha)}\int_0^\infty e^{-sx}\gamma(a,x\beta)dx \\
&= \frac{1}{\Gamma(\alpha)}\mathcal{L}\left\{\gamma(a,x\beta)\right\}(s) \\
&= \frac{1}{\Gamma(\alpha)}\frac{1}{\beta}\mathcal{L}\left\{\gamma(a,x)\right\}(s/\beta) \\
&= \frac{1}{\Gamma(\alpha)}\frac{1}{\beta}\frac{\Gamma(a)}{s/\beta}(1+s/\beta)^{-a} \\
&= \frac{1}{s(1+s/\beta)^a} && \text{(A.9)}
\end{aligned}$$

D. Generalized Pareto distribution in Matlab

The probability density function of generalized Pareto distribution used in Matlab is defined as:

$$y=f(x|k,\delta,\theta)=\left(\frac{1}{\delta}\right)\left(1+k\frac{(x-\theta)}{\delta}\right)^{-1-\frac{1}{k}} \tag{A.10}$$

for $\theta < x$, when $k > 0$, or for $\theta < x < -\delta/k$ when $k < 0$. We want to reparameterize it and obtain the *pdf* of Pareto distribution defined in Equation 5.101.

Let $k > 0$ and $\theta = \delta/k$ we have

$$\begin{aligned} f(x|k,\delta,\delta/k) &= \left(\frac{1}{\delta}\right)\left(1 + k\frac{(x-\delta/k)}{\delta}\right)^{-1-\frac{1}{k}} \\ &= \left(\frac{1}{\delta}\right)\left(\frac{kx}{\delta}\right)^{-1-\frac{1}{k}} \\ &= \left(\frac{1}{\delta}\right)\left(\frac{\delta}{kx}\right)^{1+\frac{1}{k}} \\ &= \frac{\delta^{\frac{1}{k}}}{(kx)^{1+\frac{1}{k}}} \\ &= \frac{\delta^{\frac{1}{k}}}{k^{1+\frac{1}{k}}x^{1+\frac{1}{k}}} \\ &= \frac{\frac{1}{k}\left(\frac{\delta}{k}\right)^{\frac{1}{k}}}{x^{1+\frac{1}{k}}} \end{aligned} \tag{A.11}$$

Compare with Equation 5.101 and we should adopt the following reparameterization:

$$k = \frac{1}{\alpha} \tag{A.12}$$

$$\delta = kx_m = \frac{x_m}{\alpha} \tag{A.13}$$

E. Proof of the relationship between $\mathcal{L}\{f(x)\}(s)$ and $\mathcal{L}\{F^c(x)\}(s)$

If $f(x)$ is a probability density function defined on $[a,\infty)$ and $F^c(x)$ is the corresponding complementary cumulative density function, then we must have

$$\mathcal{L}\left\{F^c(x)\right\}(s) = \frac{1 - \mathcal{L}\{f(x)\}(s)}{s} \tag{A.14}$$

where $\mathcal{L}\{f(x)\}(s)$ stands for the Laplace transform of a function $f(x)$.

Proof. We start with writing out the L.H.S. of Equation A.14 explicitly as:

$$\begin{aligned} \mathcal{L}\left\{F^c(x)\right\}(s) &= \mathcal{L}\{1 - F(x)\}(s) \\ &= \mathcal{L}\{1\}(s) - \mathcal{L}\{F(x)\}(s) \\ &= \frac{1}{s} - \int_a^\infty e^{-st}F(t)dt \end{aligned}$$

$$
\begin{aligned}
&= \frac{1}{s} - \int_a^\infty e^{-st}\left(\int_a^t f(x)dx\right)dt \\
&= \frac{1}{s} - \left[\left(-\frac{1}{s}e^{-sy}\int_a^t f(x)dx\right)\Bigg|_a^\infty - \int_a^\infty \left(-\frac{1}{s}e^{-st}\right)f(t)dt\right] \\
&= \frac{1}{s} - 0 - \frac{1}{s}\int_a^\infty e^{-st}f(t)dt \\
&= \frac{1 - \mathcal{L}\left\{f(x)\right\}(s)}{s} \qquad \text{(A.15)}
\end{aligned}
$$

which is just the R.H.S. of Equation A.14. ■

F. Supplementary material

The following tables list detailed results of the numerical experiments for verifying the MRGP method using simulations presented in Section 5.4.2.

Table A.1: MRGP vs. simulation: Exponential - Distance

Model Parameters			Simulation results		MRGP solution	
λ	μ	λ_c	Blocking Prob.	Dropping Prob.	Blocking Prob.	Dropping Prob
5	1	0.1	0.3589	0.0097	0.3641	0.0097
5	1	0.5	0.3495	0.0452	0.3524	0.0459
5	1	1	0.3417	0.0859	0.3392	0.0857
5	1	2	0.3169	0.1520	0.3165	0.1522
5	1	5	0.2690	0.2888	0.2687	0.2893
5	1	8	0.2384	0.3770	0.2382	0.3773
5	1	10	0.2240	0.4208	0.2231	0.4213

A * indicates that the T-test is significant at level 0.05.

Table A.2: MRGP vs. simulation: Exponential - Skipfree

Model Parameters			Simulation results		MRGP solution	
λ	μ	λ_c	Blocking Prob.	Dropping Prob.	Blocking Prob.	Dropping Prob
5	1	0.1	0.3420	0.0030	0.3630	0.0031
5	1	0.5	0.3659	0.0156	0.3549	0.0153
5	1	1	0.3549	0.0305	0.3456	0.0296*
5	1	2	0.3246	0.0555	0.3290	0.0559
5	1	5	0.2895	0.1197	0.2910	0.1207
5	1	8	0.2640	0.1707	0.2640	0.1714
5	1	10	0.2498	0.2009	0.2498	0.2001

A * indicates that the T-test is significant at level 0.05.

Table A.3: MRGP vs. simulation: Exponential - Uniform

Model Parameters			Simulation results		MRGP solution	
λ	μ	λ_c	Blocking Prob.	Dropping Prob.	Blocking Prob.	Dropping Prob
5	1	0.1	0.3728	0.0162	0.3748	0.0167
5	1	0.5	0.3628	0.0739	0.3618	0.0740
5	1	1	0.3485	0.1300	0.3475	0.1307
5	1	2	0.3236	0.2147	0.3240	0.2151
5	1	5	0.2766	0.3656	0.2777	0.3641*
5	1	8	0.2494	0.4495	0.2492	0.4494
5	1	10	0.2352	0.4898	0.2354	0.4899

A * indicates that the T-test is significant at level 0.05.

Table A.4: MRGP vs. simulation: Gamma - Distance

Model Parameters				Simulation results		MRGP solution	
λ	μ	α	β	Blocking Prob.	Dropping Prob.	Blocking Prob.	Dropping Prob
5	1	10	1	0.3637	0.0097	0.3641	0.0099
5	1	10	5	0.3507	0.0486	0.3517	0.0484
5	1	10	10	0.3375	0.0921	0.3371	0.0914
5	1	10	20	0.3126	0.1617	0.3123	0.1622
5	1	10	50	0.2624	0.3033	0.2621	0.3037
5	1	10	80	0.2312	0.3927	0.2312	0.3922
5	1	10	100	0.2161	0.4361	0.2163	0.4359

A * indicates that the T-test is significant at level 0.05.

Table A.5: MRGP vs. simulation: Gamma - Skipfree

Model Parameters				Simulation results		MRGP solution	
λ	μ	α	β	Blocking Prob.	Dropping Prob.	Blocking Prob.	Dropping Prob
5	1	10	1	0.3724	0.0033	0.3630	0.0032
5	1	10	5	0.3601	0.0160	0.3545	0.0158
5	1	10	10	0.3530	0.0321	0.3441	0.0313
5	1	10	20	0.3232	0.0599	0.3250	0.0604
5	1	10	50	0.2819	0.1308	0.2823	0.1303
5	1	10	80	0.2552	0.1831	0.2542	0.1825
5	1	10	100	0.2400	0.2121	0.2401	0.2112

A * indicates that the T-test is significant at level 0.05.

Table A.6: MRGP vs. simulation: Gamma - Uniform

Model Parameters				Simulation results		MRGP solution	
λ	μ	α	β	Blocking Prob.	Dropping Prob.	Blocking Prob.	Dropping Prob
5	1	10	1	0.3794	0.0171	0.3747	0.0173
5	1	10	5	0.3638	0.0821	0.3610	0.0818
5	1	10	10	0.3439	0.1444	0.3457	0.1448
5	1	10	20	0.3211	0.2321	0.3210	0.2338*
5	1	10	50	0.2736	0.3827	0.2733	0.3834
5	1	10	80	0.2442	0.4672	0.2445	0.4670
5	1	10	100	0.2308	0.5062	0.2307	0.5065

A * indicates that the T-test is significant at level 0.05.

Table A.7: MRGP vs. simulation: Pareto - Distance

Model Parameters				Simulation results		MRGP solution	
λ	μ	a	x_m	Blocking Prob.	Dropping Prob.	Blocking Prob.	Dropping Prob
5	1	2	5	0.3686	0.0102	0.3641	0.0099
5	1	2	1	0.3515	0.0473	0.3518	0.0478
5	1	2	0.5	0.3365	0.0884	0.3377	0.0892
5	1	2	0.25	0.3149	0.1580	0.3142	0.1566
5	1	2	0.1	0.2662	0.2922	0.2669	0.2918
5	1	2	0.0625	0.2375	0.3779	0.2373	0.3776
5	1	2	0.05	0.2215	0.4223	0.2227*	0.4204*

A * indicates that the T-test is significant at level 0.05.

Table A.8: MRGP vs. simulation: Pareto - Skipfree

Model Parameters				Simulation results		MRGP solution	
λ	μ	a	x_m	Blocking Prob.	Dropping Prob.	Blocking Prob.	Dropping Prob
5	1	2	5	0.3474	0.0030	0.3630	0.0032
5	1	2	1	0.3544	0.0161	0.3545	0.0158
5	1	2	0.5	0.3452	0.0312	0.3443	0.0311
5	1	2	0.25	0.3233	0.0589	0.3260	0.0592
5	1	2	0.1	0.2906	0.1274	0.2866*	0.1254
5	1	2	0.0625	0.2590	0.1741	0.2604	0.1753
5	1	2	0.05	0.2479	0.2047	0.2469	0.2032

A * indicates that the T-test is significant at level 0.05.

Table A.9: MRGP vs. simulation: Pareto - Uniform

Model Parameters				Simulation results		MRGP solution	
λ	μ	a	x_m	Blocking Prob.	Dropping Prob.	Blocking Prob.	Dropping Prob
5	1	2	5	0.3751	0.0174	0.3747	0.0173
5	1	2	1	0.3626	0.0801	0.3612	0.0796
5	1	2	0.5	0.3457	0.1369	0.3466	0.1383
5	1	2	0.25	0.3239	0.2200	0.3231	0.2217
5	1	2	0.1	0.2777	0.3650	0.2775	0.3656
5	1	2	0.0625	0.2493	0.4490	0.2496	0.4478
5	1	2	0.05	0.2359	0.4864	0.2361	0.4872

A * indicates that the T-test is significant at level 0.05.