Office of Graduate Studies

Dissertation / Thesis Approval Form

This form is for use by all doctoral and master's students with a dissertation/thesis requirement. Please print clearly as the library will bind a copy of this form with each copy of the dissertation/thesis. All doctoral dissertations must conform to university format requirements, which is the responsibility of the student and supervising professor. Students should obtain a copy of the Thesis Manual located on the library website.

Dissertation/Thesis Title: Preemption control of multi-class loss networks

Author: Zhen Zhao

This dissertation/thesis is hereby accepted and approved.

Signatures:

Examining Committee

Chair _____

Members _____

_____

_____

_____

_____

Academic Advisor _____

Department Head _____

Office of Graduate Studies • 3141 Chestnut St. • Randell Hall 240 • Philadelphia, PA 19104
Tel.: 215-895-0366 • **Fax**: 215-895-0495 • **Email**: graduate@drexel.edu • **Web**: www.drexel.edu/provost/graduatestudies

Preemption control of multi-class loss networks

A Thesis

Submitted to the Faculty

of

Drexel University

by

Zhen Zhao

in partial fulfillment of the

requirements for the degree

of

Doctor of Philosophy

June 2011

Dedications

I would like to dedicate this Doctoral dissertation to my father, Prof. Zhao, Jiakui. There is no doubt in my mind that without his affection I could not even have majored in this field or have completed this process.

Acknowledgments

Sukrit Dasgupta helped me with my arrival in the U.S. and also in this project. He did a lot of work on our first simulator. Joshua Goldberg wrote most of the second simulator. He gave me many useful advices and good tips. Bryan Williams and I cooperated writing the simulator that was used to study the use of preemption and adaptation together, resulting in my first publication in this field. Anbu Elancheziyan worked with me together for almost 7 years. His professional experience with the hardware and experiments were greatly helpful. Dr. de Oliveira and Dr. Weber are not only my advisors but also more like friends. I learned how to do research, write papers, present our work, prepare research proposals, etc., from them. I also got lots of help from both of them in my research and life. Dr. Dandekar's' and Dr. Sethu's feedback on my proposal were very critical to the continuation of my work and development of this dissertation. Moreover, as their TA, I learned a lot of from them with respect to teaching. This research is strongly grounded on the usage of mathematical knowledge obtained from many mathematical papers and books, which could not have been feasible to me without taking Dr. Schmutz's courses. I also owe thanks to Dr. Petropulu for her suggestions and recommendations. I do appreciate all of these and other friends' favor.

Table of Contents

List of Tables

List of Figures

Abstract

Preemption control of multi-class loss networks

Zhen Zhao

Jaudelice Cavalcante de Oliveira, Ph.D. and Steven Weber, Ph.D.

This thesis addresses the analysis and optimization of preemption in multi-class loss networks. Preemption, admission control and rate adaptation, are control mechanisms that enable loss network operators to provide quality of service (QoS) guarantees for admitted calls. This research includes two parts: *i*) performance characterization of a two parallel link loss network servicing multiple classes of calls under a specific preemption and admission policy, and *ii*) preemption and admission control policy analysis for a single loss link servicing two classes of calls.

In Part I, we consider a two parallel link multi-class loss network, where a call may preempt, if necessary, any calls with lower priorities and may in turn be preempted by any calls with higher priorities. The preemption policy permits both preemption from a preferred link to a backup link if possible, and eviction from either link if necessary. Our contributions in this part include: *i*) characterizing the rates of each class causing preemption of active lower priority calls, and the rates of each class being preempted by an arriving higher priority call in Erlang-B functions when all classes share a common service rate; *ii*) simple expressions of these preemption rates through uniform asymptotic approximation; and *iii*) asymptotic approximation of these preemption rates using nearly completely decomposable (NCD) Markov chain techniques when classes have individual service rates.

After analyzing the performance of a typical policy, we would also like to study various policies. In Part II, we analyze different preemption and admission control policies for a two-class loss link where per-class revenue is earned per unit time for each active call, and an instantaneous preemption cost is incurred whenever the preemption mechanism is employed. Our contributions in this part include: *i*) showing that under reasonable reward models, if we always preempt when the link is full, then it is better not to preempt at non-full states; *ii*) a sufficient condition under which the average

revenue of optimal preemption policy without admission control exceeds that of optimal admission control policy without preemption, which are established via policy improvement theorems from stochastic dynamic programming.

Chapter 1: Introduction

1.1 Preemption control of multi-class loss networks

Communication networks may be divided into two categories: *i*) best-effort networks and *ii*) circuit-switched (loss) networks, see Fig. 1.1. The current Internet is an example of a best-effort network, while the traditional phone network is an example of a loss network. It is well-known that best-effort networks are appropriate for the traditional data service but do not guarantee the adequate performance of real-time traffic. On the other hand, circuit-switched networks are appropriate for real-time traffic like voice calls and streaming media, but the allocation of resources is inefficient for bursty connections. A loss network is defined as a collection of links (each capable of multiplexing a finite number of concurrent calls) servicing a set of routes (each route consisting of a set of links), where a control mechanism determines whether or not to admit each arriving call on each route [1].

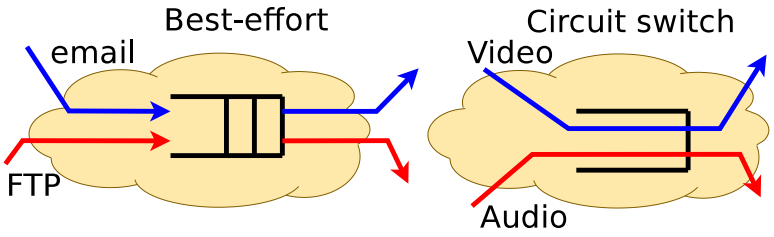


Figure 1.1: Best-effort networks are packet switched, do not reserve resources for connections, and do not employ admission control. Circuit switched networks are connection oriented, with each admitted call reserving the use of one or more dedicated circuits, and as such admission control is required.

In recent years, real-time (inelastic) network traffic (e.g., voice, streaming audio and video) comprises a rapidly growing fraction of network traffic. Real-time traffic is fundamentally different than non-real-time (elastic) traffic (e.g., data transfer applications like web and email) in that satisfactory application performance requires the network provide quality of service (QoS) guarantees that such traffic will receive a minimum bit rate with bounded delay variation (jitter). The rising prevalence of real-time traffic and fact that loss networks are the appropriate communication network architecture to offer the QoS guarantees that such traffic requires motivates the study of loss networks.

Loss networks offering multiple service classes are capable of discriminating among different connection requests, see Fig. 1.2. Multi-class loss networks service multiple classes of calls, where classes often indicate call priority, and call priority often reflects the ordering of reward paid to the network for each admitted call. In the general case arrival rate, service rate, and call rate/size (the number of circuits on each link of the route consumed by a call of that class) are class specific. The importance of multi-class service discrimination arises from the widely heterogeneous nature of loss network traffic, ranging from casual entertainment (e.g., YouTube videos) to emergency services (e.g., 911 calls). This importance motivates our study of multi-class loss networks.

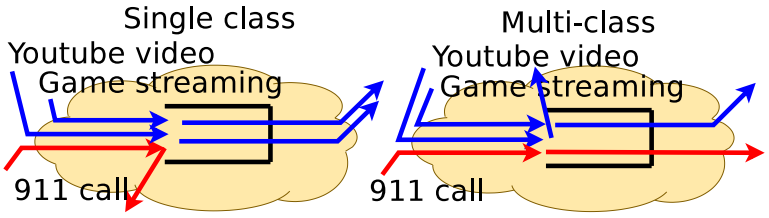


Figure 1.2: Multi-class loss networks are capable of service discrimination whereas single class networks loss networks are not.

The facts that loss links have a finite amount of circuits and that admitted calls reserve link resources requires that loss links employ some form of control to limit resource consumption. There are three popular control mechanisms for multi-class loss networks: admission control, preemption control, and rate adaptation. See Fig. 1.3. The most widely used control mechanism in loss networks is admission control. An admission control policy specifies whether or not to admit an arriving call of a given class on a given route as a function of the number of active calls of each class on each link.

A second control mechanism for loss networks is rate adaptation, where active calls may be asked to dynamically adjust their resource consumption in response to changes in the instantaneous link load. Which calls are asked to change their rates and by how much, are specified by the adaptation policy. Streaming media are good candidates for rate adaptation: typically there is a range between the minimum acceptable media quality and the maximally desirable media quality, with a commensurate range in media encoding rates. Dynamic encoding for rate adaptation enables active calls to respond to changes in network congestion.

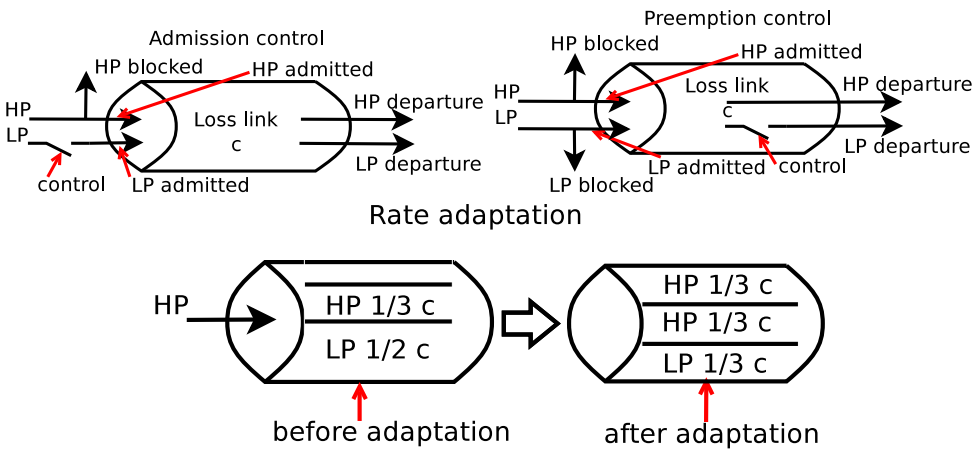


Figure 1.3: Control mechanisms for multi-class loss networks include admission control, preemption and rate adaptation.

A third control mechanism for multi-class loss networks is preemption, where an arriving call may be admitted by possibly preempting an active call of lower priority. The preemption policy is typically a function of the number of active calls of each priority level, which we call the state of the link. The preemption policy specifies whether to *i*) block, *ii*) admit without preemption, or *iii*) admit with preemption an arriving call of each possible class as a function of the state. The preemption policy enables service differentiation in that the blocking probability is typically smaller for higher priority calls, but incurs the cost that lower priority calls may find themselves admitted then preeempted before their intended call termination time. Preempted calls may be rerouted or terminated depending upon the network policy and resource availability. In short, preemption may be used to assure that high priority calls are routed along favorable paths. The use of preemption policies for inelastic traffic (voice, video, *etc.*) in loss networks has gained attention in recent years as a flexible and effective control mechanism to dynamically allocate network resources among competing traffic classes with different priorities. This thesis addresses the performance analysis and policy design of preemptive multi-class loss links in a wired network, see Fig. 1.4.

1.2 Thesis contributions

The contributions of this thesis are summarized in two main chapters. Chapter 2 investigates the performance of a two parallel link loss network servicing multiple service classes under a specified

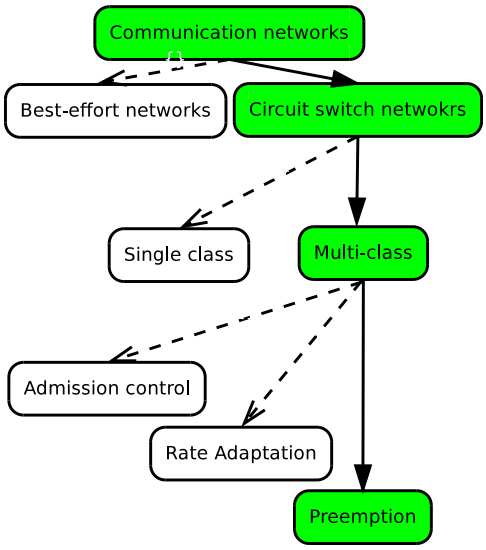


Figure 1.4: Preemption control of multi-class loss networks and related context.

preemption policy. Although blocking probabilities for multi-class non-preemptive loss networks (and their associated heavy traffic limits) have been known for a long time, this work is the first to successfully analyze the blocking rates and preemption rates under preemption. Contributions include:

- characterization of the preemption rates/probabilities for each of K preemptive classes with homogeneous service rates on a two parallel link network characterization of the admission, blocking, and departure rates is also provided;
- asymptotic expressions for the preemption rates for each of K preemptive classes with homogeneous service rates;
- approximation of preemption rates for classes with heterogeneous service rates under specific time scale separation in the arrival and service rates.

Chapter 3 studies the joint use of admission and preemption control for a two class loss link under a general revenue model incorporating per-class arrival and departure revenue, preemption costs, as well as per-class holding revenue rates. Admission control policies for loss links and networks have been widely studied for more than two decades. To the best of our knowledge, this is the first work

addressing preemption and admission control. Contributions include:

- if preemption is always done when the link is full then any additional preemptions from non-full states decrease revenue;
- a sufficient condition for the superiority of optimal preemption without admission control over optimal threshold-based admission control without preemption control.

Results of Chapter. 2 were published in [2] with preliminary conference versions appearing in [3;4]. Results of Chapter 3 have been submitted for publication [5] with a preliminary conference version appearing in [6]. Our initial results on the joint use of preemption and rate adaptation policies was published in [7]. Other related results on preemption controls in generic network topologies under different bandwidth constraint models are under review in [8].

1.3 Thesis outline

This thesis is organized as follows: Chapter 2 present our research on the performance analysis of a typical preemption policy in a multi-class loss network, as illustrated in Fig. 1.5. A detailed introduction of preemption performance analysis is presented in Section 2.1. Section 2.2 summarizes related work of preemption analysis. Section 2.3 defines the model, notation, and performance metrics. Sections 2.4 and 2.5 discuss the case when all classes have the same mean duration (homogeneous service rates); Section 2.4 addresses the finite capacity case, and 2.5 addresses the asymptotic many small users regime. Section 2.6 discuss the case when call duration means are class-dependent (heterogeneous service rates). The analytical results are shown to agree with simulation results in Section 2.7, and Section 2.8 offers a conclusion.

After analyzing the performance of a typical preemption policy, a few questions immediately follow: what about other preemption policies? what is the best preemption policy? which control mechanism is better: preemption or admission control? We answer some of these questions and also explain why we cannot answer the remaining ones in Chapter 3. We describe our system model in §3.3, as illustrated in Fig.1.6. The two primary findings discussed above are presented in the two subsections of §3.4. Numerical results are discussed in §3.5 with a detailed analysis of the simple

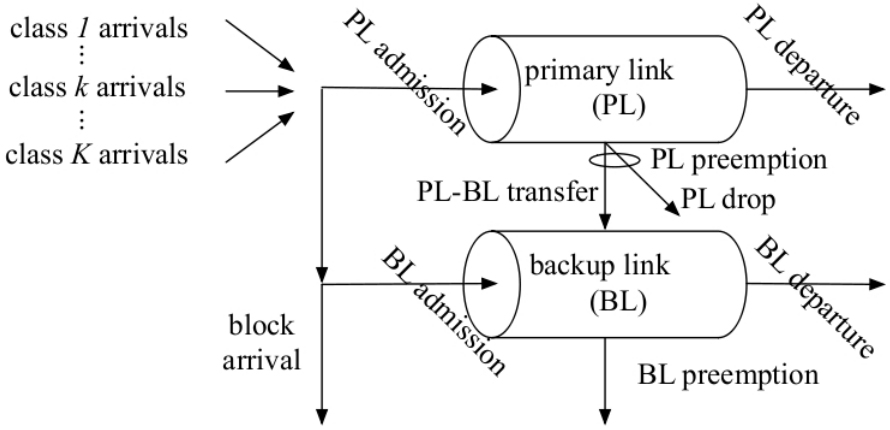


Figure 1.5: Chapter 2: Performance analysis of preemption rates/probabilities of a multi-class loss network with fixed admission and preemption policy.

case of a link supporting a single circuit ($c = 1$) in §3.5.1, and numerical performance plots of larger systems ($c > 1$) in §3.5.2. We give a brief summary of related work in §3.2 and a short conclusion in §3.6.

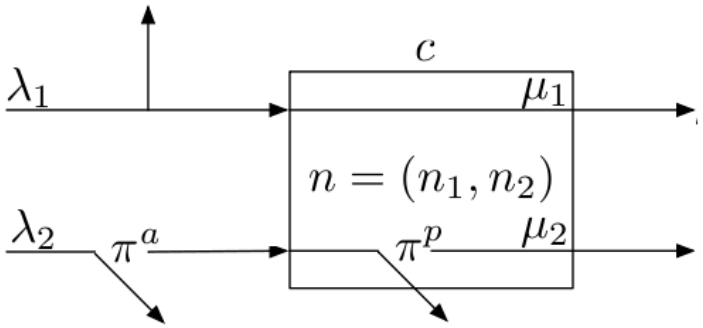


Figure 1.6: Chapter 3: Policy analysis of admission and preemption controls of a two-class loss link.

Summary of our research, limitations and future work is given in Chapter 4. Proofs of several lemmas are placed in appendices following the references.

Chapter 2: Preemption rates for a parallel link loss network

We consider a two parallel link network supporting K call classes, where a class k call may preempt if necessary any calls of classes $k+1,\ldots,K$, and may in turn be preempted by any calls of class $1,\ldots,k-1$. The two links are a (preferred) primary link (PL) and a backup link (BL). The preemption policy permits both preemption from the PL to the BL (a transfer) if possible, and eviction from either link if necessary. We characterize the rates of an arriving class k call causing preemption of an active lower priority call, and of an active class k call being preempted by an arriving higher priority call. When all classes share a common service rate, we express the preemption rates for each class in terms of the Erlang-B blocking probability equation. Simple expressions for the preemption rates are obtained in the heavy traffic limit. When classes have individual service rates, we approximate the preemption rates for each class using nearly completely decomposable (NCD) Markov chain techniques. The accuracy of the approximation improves with increasing timescale separation between classes.

2.1 Introduction

The use of preemption policies for inelastic traffic (voice, video, *etc.*) in loss networks has gained attention in recent years as a flexible and effective control mechanism to dynamically allocate network resources among competing traffic classes with different priorities. The motivation behind the use of preemption policies is the desire to provide differentiated quality of service (QoS) to the various classes. Consider the simple case of multiple classes of unit rate (size) sharing a single loss link. Without preemption, the service quality of the link is the common blocking probability for all classes. With preemption, the blocking probability is class dependent: higher priority classes have smaller blocking probabilities at the expense of higher preemption probabilities for lower priority classes.

Applications of preemption. Preemption policies have been widely employed in the context of Multi-Protocol Label Switching (MPLS) capable networks, where the *preemption attribute* deter-

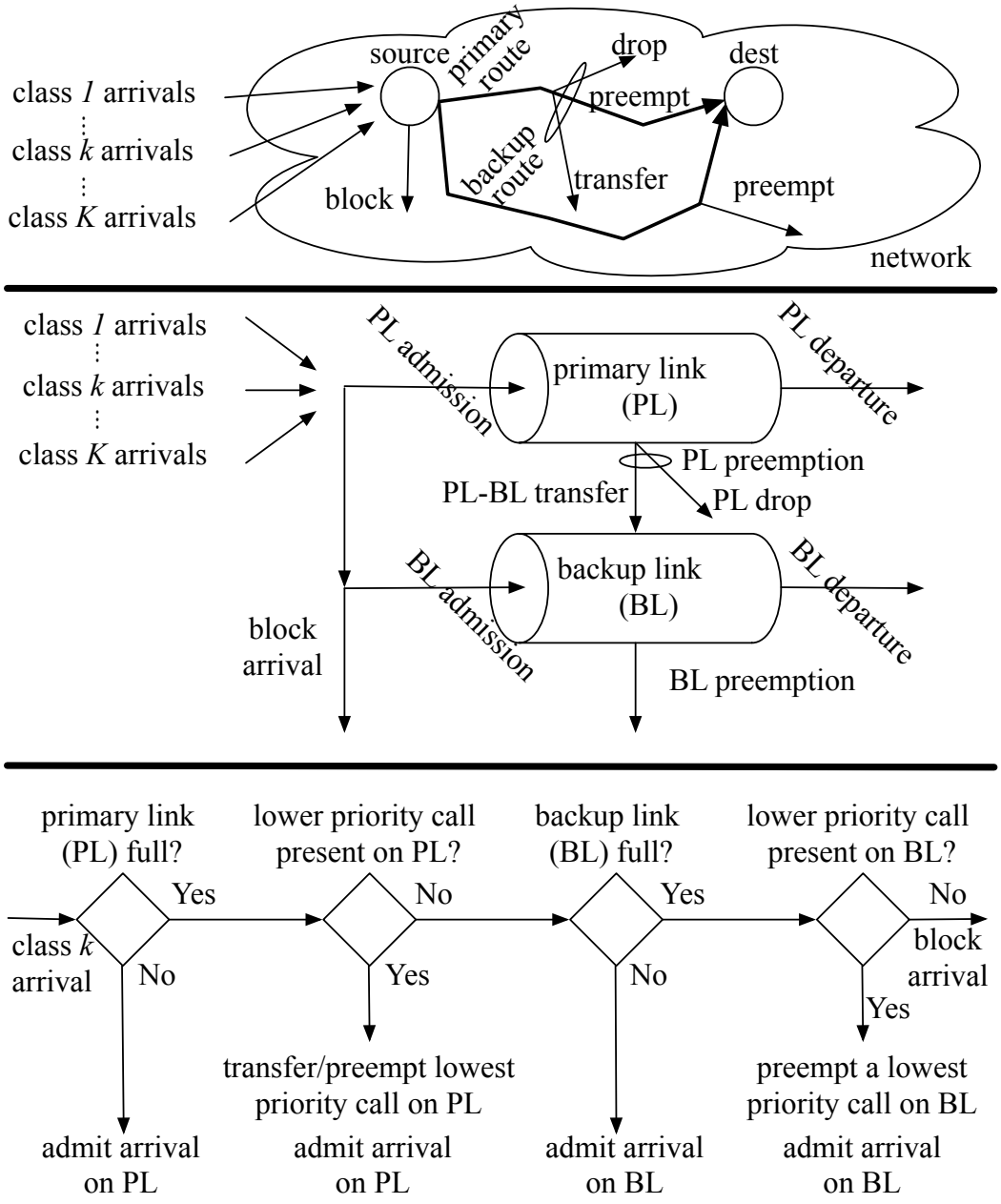


Figure 2.1: Top: a source destination pair connected by primary and backup routes in a multi-class loss network with preemption. On the primary route, preemptions consist of both transfers (to the backup route), and drops (from the system). **Middle:** a first order model of the above two route topology is a two parallel link topology. **Bottom:** new call admission decision flowchart.

mines whether a Label Switched Path (LSP) with a certain *priority attribute* can preempt another LSP with a lower *priority attribute* from a given path, when there is a competition for available resources. The preempted LSP may then be rerouted (if possible) or terminated (if necessary) [9]. Preemption has also been employed in wireless medium access control (MAC) protocols, where transmissions associated with inelastic applications, like voice or media traffic, have preemptive access priority over transmissions associated with elastic applications, like web and email traffic. In cellular networks, preemption has been employed to allow GSM (Global System for Mobile communications) voice calls to preempt GPRS (General Packet Radio Service) data packets [10]. In cognitive radio, the aim is to improve the efficiency of spectrum utilization by allowing users to adapt their spectrum consumption based on spatio-temporal availability. It has been proposed to allow "primary" users preemptive priority over "secondary users", ensuring that primary users have access to the resource when they require, and that secondary users have access to the resource when it is available [11]. Although preemption is gainfully employed in a wide variety of applications, the canonical application motivating this paper is the case of wired multi-service loss networks with primary and backup routes.

Model justification. The motivation for our two parallel link network is the common case of two disjoint routes servicing a source destination pair in a multi-class preemptive loss[1] network (see top of Fig. 2.1). Often one route (the primary) is preferable to the other (backup), possibly due to the shorter distance, shorter hop count, lower average congestion, *etc.* of the primary relative to the backup. Alternately, the backup route may only be used in case the primary link fails (fault restoration) or suffers an overflow (excessive congestion). As is natural in an overflow context, our model allows transfers of calls from the primary link to the backup link (but not vice-versa). A first order approximation of the two disjoint route network is the two parallel links topology (see middle of Fig. 2.1). This approximation ignores the cross traffic in the two routes topology, as well as the variation in link capacity on each route. Nonetheless, the model captures many essential features of the two route loss model: route preference, multi-class arrivals, admissions, blocked calls, transfers,

[1]Loss links/networks/systems are so called because they have no queueing: requests arriving to find all servers/circuits busy are lost (blocked) instead of queued.

preemptions, and departures.

Admission and preemption. Call[2] classes are distinguished by their arrival rate and their priority. Class 1 calls are highest priority, and class K calls are lowest priority. For simplicity we assume all calls are unit rate (size), *i.e.*, each call consumes one circuit, and we measure link capacity in terms of the number of available circuits. We consider the case when the service rate is common to all classes (homogeneous), as well as the case where the service rate is class dependent (heterogeneous).

The admission and preemption mechanism for the two parallel link topology is as follows (see bottom of Fig. 2.1). All arriving calls initially seek admission on the primary link. If an arriving class k call finds the primary link not full, it is admitted there. If the primary link is full but there are one or more lower priority active calls, then the class k call is admitted on the primary link, and a randomly selected call of the lowest active class is preempted. This preempted call of class $j > k$ is transferred to the backup link if either the backup link is not full, or if it is full but there are one or more active calls of lower priority $i > j$. In the latter case, a randomly selected call of the lowest active priority class on the backup link is preempted from the backup link to accommodate the class j call's transfer from the primary to the backup link. If the call preempted from the primary link cannot be transferred to the backup link, then it is preempted from the network (dropped).

If the arriving class k call cannot be admitted on the primary link (because the link is filled with calls of priority $1, \ldots, k$), then it seeks admission on the backup link. If an arriving class k call finds the backup link not full, it is admitted there. If the backup link is full but there are one or more lower priority active calls, then the class k call is admitted on the backup link, and a randomly selected call of the lowest active class is preempted. This preempted call is preempted from the network (dropped). If the arriving class k call cannot be admitted on the backup link (because the link is filled with calls of priority $1, \ldots, k$), then it is blocked from the system.

Performance metrics. We define three preemption probabilities/rates for each class: i) the probability/rate that an arriving class i call preempts a certain active class $k > i$ call, ii) the

[2]We use the generic term calls throughout the article, which could be replaced by LSPs, connections, circuits, *etc.*

probability/rate that a class k call arrival preempts any active lower priority call for admission, and *iii*) the probability/rate that an active class k call is preempted by an arriving higher priority call.

Summary of results. Although blocking probabilities for multi-class *non-preemptive* loss networks (and their associated heavy traffic limits) have been known for a long time, this work is the first to successfully analyze a multi-class *preemptive* loss network. This analysis is significant on account of the fact that numerical techniques are unlikely to be successful for even moderate sized systems, e.g., the number of states even for a single link with c circuits serving K classes grows like $O(c^K)$. This contribution is significant on account of the increasing use of preemptive policies in a wide variety of networking contexts. Specific contributions include:

- Characterization of the preemption rates/probabilities for each of K preemptive classes with homogeneous service rates on a two parallel link network in terms of Erlang-B blocking probabilities. Characterization of the admission, blocking, and departure rates is also provided.

- Asymptotic expressions for the preemption rates for each of K preemptive classes with homogeneous service rates using uniform asymptotic approximations for the Erlang-B blocking probabilities. This approximation is appropriate for the "many small users" regime, where both arrival rates and link capacities are large, and includes the heavy traffic limit as a special case. Sensitivities of the heavy traffic preemption rate expressions to the arrival rates are discussed.

- Application of the Markov chain concepts of lumpable, nearly completely decomposable (NCD), and time-scale separation to the preemption model. This leads to an approximation of the preemption rates for each of K preemptive classes with heterogeneous service rates under a specific time-scale separation regime.

Outline. The rest of this paper is organized as follows. Section 2.2 summarizes related work. Section 2.3 defines the model, notation, and performance metrics. Sections 2.4 and 2.5 discuss the case when all classes have the same mean duration (homogeneous service rates); Section 2.4 addresses the finite capacity case, and 2.5 addresses the asymptotic many small users regime. Section 2.6

discuss the case when call duration means are class-dependent (heterogeneous service rates). The analytical results are shown to agree with simulation results in Section 2.7, and Section 2.8 offers a conclusion.

2.2 Related Work

We divide our discussion of related work on preemption into two parts. The first part discusses related work on proposed preemption policies, both optimal and heuristic. Although much of this work discusses the important issue of the computational complexity of the proposed policies, in general this body of work contains very little in the way of performance analysis. The second part concentrates on performance analysis of a single link with preemptive priority.

2.2.1 Proposed preemption policies

The 1992 paper by Garay and Gopal addressed the call preemption problem in communication networks[12], showing that the problem of selecting a connection for preemption in order to minimize the number of preempted connections or minimize the amount of preempted bandwidth is NP-complete. They propose heuristics for a centralized network framework that are shown to perform reasonably well relative to the optimal solution. Extending Garay and Gopal's work, in 1997 Peyravian and Kshemkalyani proposed decentralized network connection preemption algorithms[13] that optimize three fixed criteria in a given order of importance: number of connections, bandwidth, and priority.

After these two seminal works, many of the subsequent proposed preemption policies have been described in the context of a Differentiated Services (DiffServ) aware MPLS scenario, *e.g.*,[14–20], discussed below. In particular, the decentralized policies in[13] are the basis for our earlier work on flexible and adaptive preemption policies[14]. Here, an order of importance for the considered criteria is not fixed, but can be configured by the network provider according to the network's best interest. In[15], Sung-eok *et al.* propose a centralized connection preemption algorithm that optimizes the preemption criteria in a fixed order different from[13]. In[16], Tong *et al.* present an algorithm that jointly considers both bandwidth allocation and preemption.

Stanisic and Devetsikiotis propose simple preemption policies based on random selection; this

dramatically reduces the time needed to select a set of connections to be preempted [17]. Both Blanchy *et al.* [18] and Yu *et al.* [19] focus on preemption–aware routing algorithms. In particular, a path is selected by minimizing the number of connections (LSPs) that require preemption. The routing algorithm therefore tries to minimize the occurrence of preemption events and thereby minimize the need for rerouting. Recently, Vieira and Guardieiro implemented de Oliveira's preemption policies in [14] using fuzzy logic and genetic algorithms in an MPLS testbed [20].

2.2.2 Analysis of single links with preemption

Ours is the first analytical treatment of the performance of a preemptive network with multiple loss links. Related work studies either a *single link* servicing multi-class elastic (*e.g.*, email, web) or inelastic (*e.g.*, voice, video) traffic *with preemption*, or a *general network* servicing multi-class traffic (elastic or inelastic) *without preemption*. The text by Ross [21] covers non-preemptive loss networks (for inelastic traffic), while the text by Srikant [22] covers non-preemptive best-effort networks (for elastic traffic). Below, we restrict our attention to work on preemption modeling.

Preemptive systems can be dichotomized into preemption with delay and preemption with loss. Preemption with delay means preempted calls are "put on hold", and queued until their service resumes or restarts. Preemption with loss means that preempted calls are removed, this can mean either transfer or eviction. Preemption with delay is usually modeled by an $M/G/c$ queue (infinite queueing), while preemption with loss is usually modeled by an $M/G/c/c$ queue (no queueing).

Preemption with delay. The earliest analysis of preemption is in the context of preemption with delay. In fact, the first paper published on priority queueing with preemption is from 1958, by White and Christie [23]. In this paper, White and Christie analyze the average queue length and the average time in system for a preemptive resume and repeat policy. They also study a "breakdown" system where the preemptive server is prone to failure (vacations). Miller [24] uses matrix-geometric methods to compute steady state probabilities for an $M/M/1$ priority queue, modeling a single link servicing elastic traffic with preemption. Buzen and Bondi [25] published an article in 1983 studying a network of $M/M/c$ queues with preemptive resume policies. Their results are focused on moments in a preemptive–delay network. Ngo and Lee published a short note in 1990 [26] on a single $M/M/c$

queue with preemptive priority, extending [24]. The work in [24] is further generalized by Cho and Un [27], who provide an analysis of a combined preemptive/nonpreemptive priority *M/G/1* queue. There are many other papers in the queueing literature on preemption with delay; these analyses are of limited relevance to our work since our focus is on preemption with loss.

Preemption with loss. The above articles analyze the performance of a preemption system with delay. Unfortunately, the more prevalent use of preemption policies (*e.g.*, MPLS) is to drop (as in the loss model), rather than postpone (as in the delay model) the preempted calls. There is some existing work on preemption with loss, but all such work is either analysis of a single link, or has a numerical/computational focus for multiple parallel links. The earliest performance analyses of a preemption policy in a loss context are by Helly [28] and Burke [29], both from 1962. These short papers present the framework for employing the Erlang B blocking probability equation on a single link with preemption. These two papers served as an inspiration for our results in §2.4. After that, the literature appears to be silent until 1980 when Calabrese *et al.* [30] published an analysis of a voice network of multiple parallel links with preemption. Their paper includes a discussion of a variety of different preemption policies, which they term "ruthless" and "friendly." This model combines the two preemption policies with the *estimated* probability that a high priority call returns to the original link after searching all alternate links and finding them blocked. Although this paper studies multiple parallel links, the focus is on algorithms for computation of the performance metrics, along with numerical approximations of the optimal solution. In contrast, our work focuses on closed-form performance expressions. Moreover, [30] is essentially a "soft" preemption model, where high-priority calls only preempt low-priority calls if each of the routes is full, whereas our "hard" preemption model allows high-priority calls to preempt low-priority calls if the primary link is full, regardless of the status of the backup link. In 1980, Fischer [30] discussed the blocking and preemption probabilities of two priority classes with different service times in a single preemptive loss link. In that paper, due to the difficulty in solving the steady state equations, the author analyzed three special cases of the solution: *i*) *M/M/1/1*, *ii*) *M/M/c/c* with ratio of class 2 to class 1 mean holding time tends to 0 and *iii*) *M/M/c/c* with ratio of class 2 to class 1 mean holding time tends to ∞.

2.3 Model, notation, and performance metrics

The two parallel link network is illustrated in Fig. 2.2. The notation used in the paper is given in Table 2.1. Quantities for the PL are denoted by superscript p, and quantities for the BL are denoted by superscript b. Quantities for the two links viewed as a whole are denoted by superscript s (for system). We will often discuss a generic single isolated link (IL); quantities for an IL are without superscript. There are K classes, numbered $1, \ldots, K$, where class 1 is of highest priority and class K is of lowest priority. Arriving class k calls have preemptive priority over all active calls of lower priority class $k+1, \ldots, K$. Moreover, active class k calls are subject to possible preemption by arriving higher priority calls of class $1, \ldots, k-1$. The admission and preemption policy is described in §2.1.

Table 2.1: Mathematical notation.

c^p	capacity of PL (# of available circuits)
c^b	capacity of BL (# of available circuits)
$c^s = c^p + c^b$	total capacity of system
λ_k	arrival rate of class k
$\Lambda_k = \sum_{i=1}^{k} \lambda_i$	cumulative arrival rate of classes 1 to k
r	arrival rate scaling parameter (§2.7)
μ_k	service rate of class k
μ_k^{-1}	mean duration of a class k call
$\rho_k = \lambda_k/\mu_k$	offered load of class k
$R_k = \sum_{i=1}^{k} \rho_i$	cumulative offered load of classes 1 to k
n_k	# of class k calls on IL
n_k^p	# of class k calls on PL
n_k^s	# of class k calls in system
$\mathbf{n} = (n_1, \ldots, n_K)$	# of calls of each class on IL

Continued on next page

Table 2.1 – continued from previous page

$\mathbf{n}^p = (n_1^p, \ldots, n_K^p)$	# of calls of each class on PL
$\mathbf{n}^s = (n_1^s, \ldots, n_K^s)$	# of calls of each class in system
$N_k = \sum_{i=1}^{k} n_i$	# of calls of classes 1 through k on link
$\bar{N}_k = \sum_{i=k+1}^{K} n_i$	# of calls of classes $k+1$ through K on link
$N_k^p = \sum_{i=1}^{k} n_i^p$	# of calls of classes 1 through k on PL
$\bar{N}_k^p = \sum_{i=k+1}^{K} n_i^p$	# of calls of classes $k+1$ through K on PL
$N_k^s = \sum_{i=1}^{k} n_i^s$	# of calls of classes 1 through k in the system
$\bar{N}_k^s = \sum_{i=k+1}^{K} n_i^s$	# of calls of classes $k+1$ through K in system
$E(\rho, c)$	Erlang B blocking probability of an $M/M/c/c$ queue with offered load ρ.
A_k	admission rate of class k calls on IL
A_k^p	admission rate of class k calls on PL
A_k^b	admission rate of class k calls on BL
$A_k^s = A_k^p + A_k^b$	admission rate of class k calls into system
B_k	blocking rate of class k calls on IL
B_k^p	blocking rate of class k calls on PL
B_k^b	blocking rate of class k calls on BL
$B_k^s = B_k^b$	blocking rate of class k calls from system
D_k	departure rate of class k calls from IL
D_k^p	departure rate of class k calls from PL
D_k^b	departure rate of class k calls from BL
$D_k^s = D_k^p + D_k^b$	departure rate of class k calls from system
$P_{i,k} = Q_{k,i}$	rate that arriving class i calls preempt active class k calls on IL
$P_{i,k}^p = Q_{k,i}^p$	rate that arriving class i calls preempt

Continued on next page

Table 2.1 – continued from previous page

	active class k calls on PL
$P^s_{i,k} = Q^s_{k,i}$	rate that arriving class i calls preempt active class k calls from system
T_k	rate of class k arrivals causing transfers from PL to BL
O_k	rate of class k arrivals causing drops from PL
P_k	rate of class k arrivals causing preemptions from IL
$P^p_k = T_k + O_k$	rate of class k arrivals causing preemptions from PL
P^b_k	rate of class k arrivals causing preemptions from BL
$P^s_k = O_k + P^b_k$	rate of class k arrivals causing preemptions from system
V_k	rate of class k calls transferred from PL to BL
W_k	rate of class k calls dropped from PL
Q_k	rate of class k calls preempted from IL
$Q^p_k = V_k + W_k$	rate of class k calls preempted from PL
Q^b_k	rate of class k calls preempted from BL
$Q^s_k = W_k + Q^b_k$	rate of class k calls preempted from system

Notation. As evident from Table 2.1, the paper employs extensive notation; we have striven to make this notation as intuitive and consistent as possible. The following conventions are employed:

- *Letter mnemonics*: admitted calls (A), blocked calls (B), capacity (C), departures (D), Erlang blocking probability (E), class (K), number of active calls (N), drops (O, W), preemptions (P, Q), system (S), transfers (T, V).
- *Preemptions.* Rates for causing preemption are denoted by P, rates for being preempted are denoted by Q.
- *Preemptions, transfers, and drops.* Preemptions (denoted by P, Q) from the PL are either transfers or drops. Transfers (denoted by V, T) are calls that move from the PL to the BL.

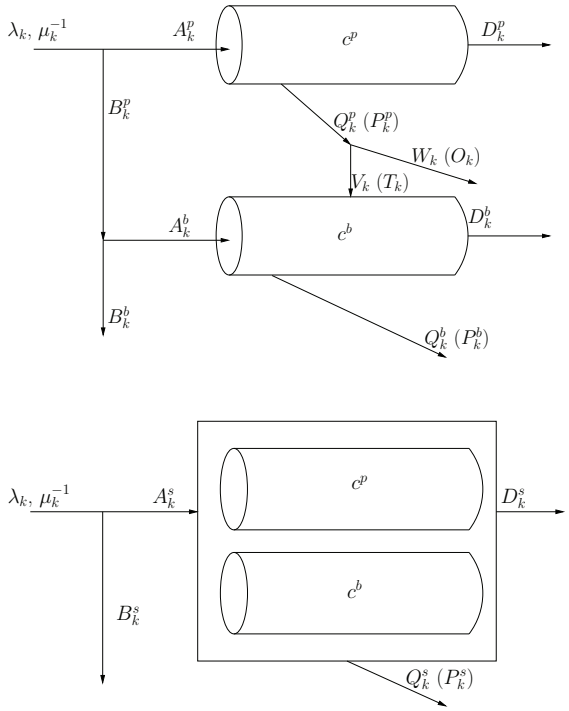


Figure 2.2: Class k calls arrive at rate λ_k, with mean duration μ_k^{-1}. **Top: link view.** A_k^p, A_k^b are the PL and BL admission rates, B_k^p, B_k^b are the primary and backup blocking rates, and D_k^p, D_k^b are the primary and backup departure rates. Arriving class k calls cause preemption on the primary and backup links at rates P_k^p, P_k^b respectively, and active class k calls are preempted from the primary and backup links at rates Q_k^p, Q_k^b respectively. Preemptions of active class k calls from the PL are divided into transfers (V_k) and drops (W_k), *i.e.*, $Q_k^p = V_k + W_k$. Preemptions caused by arriving class k calls are also divided into transfers (T_k) and drops (O_k), *i.e.*, $P_k^p = T_k + O_k$. Rate conservation ensures that all admitted calls either depart or are preempted, *i.e.*, $A_k^p = D_k^p + Q_k^p$, but no such conservation law need hold for calls causing preemption, *i.e.*, it need not be true that $A_k^p = D_k^p + P_k^p$. **Bottom: system view.** Consider the two links together as a single "system" with capacity $c^s = c^p + c^b$. A_k^s is the system admission rate, B_k^s is the system blocking rate, D_k^s is the system departure rate, and P_k^s, Q_k^s are the system preemption rates.

Drops (denoted by W, O) are calls that leave the network, from either link.

- *Primary, backup, system.* PL, BL, and system quantities are indicated by a superscript p, b, and s respectively. Unadorned quantities often refer to an IL.
- *Probabilities and rates.* The probability of an event, say F, is denoted by $\mathbb{P}(F)$; the rate at which the event occurs is denoted by $\mathbb{R}(F)$.
- *Arrivals, service rates, offered loads.* Consistent with queueing conventions, λ denotes arrival rates, μ denotes service rates, and $\rho = \lambda/\mu$ denotes offered loads. Cumulative arrival rates

are denoted by Λ, *i.e.*, $\Lambda_k = \lambda_1 + \cdots + \lambda_k$. Cumulative offered loads are denoted by R, *i.e.*, $R_k = \rho_1 + \cdots + \rho_k$.

The other notation in Table 2.1 will be introduced as needed.

Primitive quantities. The primitive quantities needed to specify the model are *i*) the link capacities, *ii*) the arrival rates, and *ii*) the service rates. All other quantities are derived from these three. The capacities of the two links are c^p, c^b, measured in number of circuits. All calls are of unit rate (size), *i.e.*, each call consumes a single circuit. The total system capacity is $c^s = c^p + c^b$ circuits. The symbol c will be used to denote the capacity of an IL, or in §2.5, the index in a sequence of ILs of increasing capacity. We assume arrivals for each class form an independent Poisson process of rate λ_k. Class k call durations are independent and exponentially distributed with rate μ_k, and hence mean μ_k^{-1}. In §2.4 and §2.5 we assume all classes share the same service rate, $\mu_k = \mu$ (homogeneous service rates); §2.6 addresses heterogeneous service rates.

Performance metrics. We focus on three performance metrics, computed for each class k, see Fig. 2.2.

1. P_k^p (P_k^b) is the rate that arriving class k calls are admitted on the PL (BL) by preempting an active call of class $k+1, \ldots, K$. The primary preemption rate consists of both transfers (T_k) and dropped calls (O_k): $P_k^p = T_k + O_k$. The quantity $P_k^s = O_k + P_k^b$ is the rate at which calls are preempted from the system (from either link).

2. $P_{i,k}^p = Q_{k,i}^p$ ($P_{i,k}^b = Q_{k,i}^b$) is the rate that arriving class i calls preempt active class $k > i$ calls on the PL (BL).

3. Q_k^p (Q_k^b) is the rate at which active calls of class k are preempted from the PL (BL) by an arriving call of class $1, \ldots, k-1$. The primary preemption rate consists of both transfers (V_k) and dropped calls (W_k), so $Q_k^p = V_k + W_k$. The quantity $Q_k^s = W_k + Q_k^b$ is the rate at which calls are preempted from the system (from either link).

2.4 Homogeneous service rates: finite capacity

Throughout this section and the next we assume that all calls have *homogeneous* independent and exponentially distributed service rates $\mu_k = \mu$. We use the Erlang-B blocking probability equation to obtain exact expressions for our performance metrics as a function of the offered loads and link capacities.

2.4.1 A Markov chain for a single preemptive link

Before studying the two parallel link model, we first study an isolated link (IL) serving a multi-class preemptive offered load, modeled as a modified $M/M/c/c$ queue.

Definition 1. Consider a single link $M/M/c/c$ queue with c circuits serving a multi-class preemptive offered load with independent Poisson arrival processes of rates $\lambda_1 \ldots, \lambda_K$, and independent and exponentially distributed service times with rate μ. Let $\{\mathbf{n}(t)\}$ be the random occupancy process giving the number of active calls of each class at each time t, where $\mathbf{n}(t) = (n_1(t), \ldots, n_K(t))$. The state space for the occupancy process is

$$\mathcal{S} = \{\mathbf{n} \geq \mathbf{0} : n_1 + \cdots + n_K \leq c\}. \tag{2.1}$$

The allowed transitions for each class $k = 1, \ldots, K$ are:

$$\begin{array}{lll} \mathbf{n} \to \mathbf{n} + \mathbf{e}_k & \text{if} & n_1 + \cdots + n_K < c \\ \mathbf{n} \to \mathbf{n} + \mathbf{e}_k - \mathbf{e}_l & \text{if} & n_1 + \cdots + n_K = c,\ l = \max\{j > k : n_j > 0\} \\ \mathbf{n} \to \mathbf{n} - \mathbf{e}_k & \text{if} & n_k > 0 \end{array} , \tag{2.2}$$

where $\mathbf{e}_k$ is a K-vector of all zeros except a one in position k.

The transition $\mathbf{n} \to \mathbf{n} + \mathbf{e}_k$ corresponds to a class k call arriving to find the system not full. The transition $\mathbf{n} \to \mathbf{n} - \mathbf{e}_k$ corresponds to the departure of a class k call. The transition $\mathbf{n} \to \mathbf{n} + \mathbf{e}_k - \mathbf{e}_l$ corresponds to a class k call arriving to find the link full, necessitating a preemption of a class l call, where l is the lowest priority class (of lower priority than k) with one or more active calls. The

admission is only allowed if the class l is well-defined.

Theorem 1. The random occupancy process $\{\mathbf{n}(t)\}$ is a continuous time Markov chain (CTMC).

The proof is found in the appendix, and in fact shows that the process $\{\mathbf{n}(t)\}$ is Markov for the more general case of heterogeneous service rates.

2.4.2 Markov chains for the PL and the system

We next consider the two parallel link model.

Definition 2. Let $\{\mathbf{n}^p(t)\}$, $\{\mathbf{n}^b(t)\}$, and $\{\mathbf{n}^s(t)\}$, be the random occupancy process giving the number of active calls of each class at each time t on the PL, BL, and in the system as a whole, where

$$\begin{aligned} \mathbf{n}^p(t) &= (n_1^p(t), \ldots, n_K^p(t)), \\ \mathbf{n}^b(t) &= (n_1^b(t), \ldots, n_K^b(t)), \\ \mathbf{n}^s(t) &= (n_1^s(t), \ldots, n_K^s(t)). \end{aligned} \tag{2.3}$$

By construction, $n_k^p(t) + n_k^b(t) = n_k^s(t)$ for each time t and each class $k = 1, \ldots, K$. The state spaces are

$$\begin{aligned} \mathcal{S}^p &= \{\mathbf{n}^p \geq \mathbf{0} : n_1^p + \cdots + n_K^p \leq c^p\}, \\ \mathcal{S}^b &= \{\mathbf{n}^b \geq \mathbf{0} : n_1^b + \cdots + n_K^b \leq c^b\}, \\ \mathcal{S}^s &= \{\mathbf{n}^s \geq \mathbf{0} : n_1^s + \cdots + n_K^s \leq c^s\}. \end{aligned} \tag{2.4}$$

The primary and backup links are "coupled" in two ways: arrivals blocked from admission on the PL seek admission on the BL, and calls preempted from the PL seek transfer to the BL. Although the state evolution of the BL very much depends upon the state of the PL, the inverse does not hold: *the evolution of the PL state is independent of the BL state.* Moreover, the state evolution of the system as a whole is independent of how the active calls are distributed between the primary

and backup links. A key observation is the fact that the PL and the system as a whole can each be studied as *isolated single link Markovian systems* with capacities c^p and c^s respectively, and, modulo the difference in capacity, their dynamics are in fact the same. Hence, for a generic L parallel link case, the basic two "couple" methods are the same and so on are the limitations of our results. Thus, we focus on only two parallel link case in this paper.

Theorem 2. The occupancy processes of the PL $\{\mathbf{n}^p(t)\}$ and of the system $\{\mathbf{n}^s(t)\}$ are statistically equivalent to an IL serving a multi-class preemptive offered load, with capacities c^p and c^s respectively. Both processes are Markovian.

Proof. Consider the PL. Its evolution is independent of the BL, *i.e.*, calls gain admission, depart, and are preempted from the PL independent of the state of the BL. The dynamics of the admission, departure, and preemption are exactly those transitions specified in Definition 1. Only the capacity c^p determines these dynamics, the capacity c^b is irrelevant. Consider the system as a whole. Its evolution depends upon the state of the primary and backup links only through the aggregate state, *i.e.*, calls gain admission, depart, and are preempted from the system as a function of the *total* number of calls of each class in the system, regardless of how they are divided among the two links. Again, the dynamics are exactly those specified in Definition 1. Only the capacity c^s determines these dynamics, the individual capacities c^p, c^b are irrelevant. □

2.4.3 Lumpability

General concepts. The key reason that the homogeneous service rates case is tractable is because the corresponding link level and system level Markov chains are *lumpable* under a partition aligned with the preemption rate performance metric. Quoting from Dayar and Stewart [31]:

> Lumpability is a property of some Markov chains which, if conditions are met, may be used to reduce a large state space to a smaller one. The idea is to find a partition of the original state space such that, when the states in each partition are combined to form a single state, the resulting Markov chain described by the combined states has equivalent behavior to the original chain, only at a coarser level of detail. ... It is mostly useful

in systems which require the computation of performance measures dependent on the coarser analysis specified by the lumped chain.

The following definition of lumpability for a CTMC is adapted to our notation from Ball and Yeo[32]. See Kemeny and Snell[33] for the discrete time Markov chain (DTMC) definition.

Definition 3. Lumpability *(Ball and Yeo[32]).* Consider a continuous time Markov chain $\{x(t)\}$ with state space $\mathcal{S}$, and initial distribution $\mathbf{p} = (p(s),\ s \in \mathcal{S})$. For a given state space partition $(\mathcal{S}_m,\ m = 1, \ldots, M)$ define the lumped process $\{y(t)\}$ with initial distribution $\mathbf{r} = (r(m),\ m = 1, \ldots, M)$ given by $r(m) = \mathbb{P}(x(0) \in \mathcal{S}_m)$ and transition probabilities given by

$$\mathbb{P}(y(t_n) = m_n \| y(t_{n-1}) = m_{n-1}, \ldots, y(t_0) = m_0) = \tag{2.5}$$
$$\mathbb{P}(x(t_n) \in \mathcal{S}_{m_n} \| x(t_{n-1}) \in \mathcal{S}_{m_{n-1}}, \ldots, x(t_0) \in \mathcal{S}_{m_0}),$$

for each n, for each set of times $t_0 < \cdots < t_n$, and for each sequence $m_0, \ldots, m_n \in [M]^{n+1}$. Then $\{x(t)\}$ is said to be *lumpable* with respect to $(\mathcal{S}_m,\ m = 1, \ldots, M)$ if, for each initial distribution $\mathbf{p}$, the process $\{y(t)\}$ is a Markov chain and the transition probabilities do not depend upon $\mathbf{p}$.

The fundamental theorem of lumpability is given below; note that some authors, *e.g.*,[31], use the following theorem as a *definition* of lumpability. See also[33] for the analogous theorem for DTMC.

Theorem 3. *(Ball and Yeo[32])* A necessary and sufficient condition for a Markov chain $\{x(t)\}$ on $\mathcal{S}$ to be lumpable with respect to a partition $(\mathcal{S}_m,\ m = 1, \ldots, M)$ is that its rate matrix $\mathbf{Q}$ satisfies the lumpability condition for each pair $\mathcal{S}_m, \mathcal{S}_{m'}$:

$$\sum_{s' \in \mathcal{S}_{m'}} q_{s,s'} = r_{m,m'},\ s \in \mathcal{S}_m \tag{2.6}$$

The $M \times M$ matrix $\mathbf{R}$ with entries $r_{m,m'}$ is the rate matrix for the lumped chain, $\{y(t)\}$.

The lumpability condition asserts that for any two subsets in the state space partition, the transition rate from a state in the first subset into the second subset is in fact the same for all states in the first subset.

Application to preemption. It is evident that the preemption probabilities for class k, *i.e.*, P_k and Q_k, do not depend on each class individually, but instead on the two groups of classes: $\{1,\ldots,k\}$ and $\{k+1,\ldots,K\}$. Informally, the only significance of a class j call to a class k call is whether $j<k$ or $j>k$, not the value of j itself. This observation allows us to collapse the set of classes from the perspective of a class k call from K to 2. The performance metrics of interest for class k are obtainable from the simplified process $\{(N_k(t),\bar{N}_k(t))\}$, where $N_k=n_1+\cdots n_k$ and $\bar{N}_k=n_{k+1}+\cdots+n_K$. This aggregation gives the total number of calls of classes $1,\ldots,k$ and the total number of calls of classes $k+1,\ldots,K$. This observation suggests the *split k aggregate occupancy partition*, defined below, and we show that the random occupancy process $\{\mathbf{n}(t)\}$ for an IL serving a multi-class preemptive load is lumpable under this partition. This partition is of use because it is aligned with the three performance metrics, allowing an effective state space reduction.

Definition 4. The *split k aggregate occupancy partition* (saop k) of $\mathcal{S}$ (defined in Definition 1) is

$$\mathcal{S}^{\text{saop},k}_{m,m'}=\{\mathbf{n}\in\mathcal{S}:N_k=m,\ \bar{N}_k=m'\},\tag{2.7}$$

for each aggregate occupancy pair (m,m') such that $m+m'\le c$, and for a specified class $k=1,\ldots,K$.

Several occupancy processes are lumpable under this partition.

Theorem 4. The occupancy processes $\{\mathbf{n}(t)\}$, $\{\mathbf{n}^p(t)\}$, and $\{\mathbf{n}^s(t)\}$ (for an IL, the PL, and the system, respectively) serving a multi-class preemptive offered load are lumpable under the split k aggregate occupancy partition for each $k=1,\ldots,K$. The corresponding lumped processes $\{(N_k(t),\bar{N}_k(t))\}$, $\{(N^p_k(t),\bar{N}^p_k(t))\}$, $\{(N^s_k(t),\bar{N}^s_k(t))\}$ are Markov.

The proof is found in the appendix. The lumpability of the state process is perhaps surprising on account of the fact that an active call may be preempted and replaced by a new call of higher priority, possibly without triggering a state change in the lumped process. This is attributable to the fact that the service rates are homogeneous and exponential, and thus the memoryless property ensures that "restarting" an active call of class i via preemption with a new call of class $j<i$ will

not affect the dynamics of the process from the perspective of a class k call that is either $k < i \wedge j$ or $k > i \vee j$.

2.4.4 Blocking and preemption on a single link

This subsection assumes an IL with capacity c servicing a multi-class preemptive offered load, see Fig. 2.3. By Theorem 2, the performance probabilities and rates for the single IL may be applied to the PL and the system by replacing the capacity c with c^p and c^s respectively.

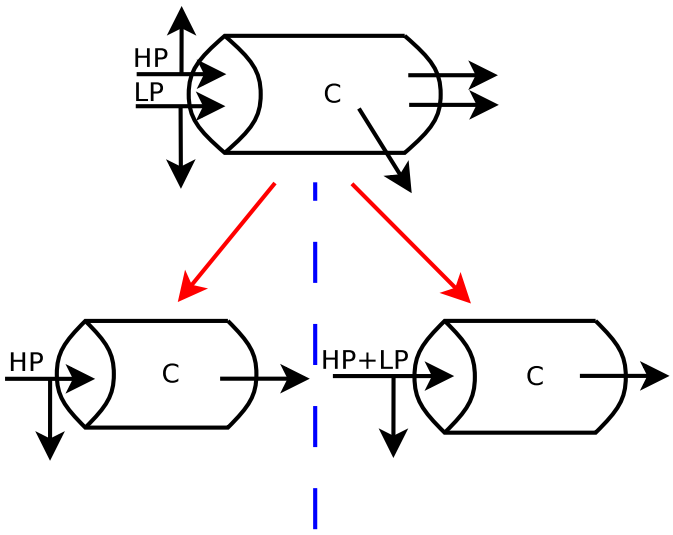


Figure 2.3: An example of an IL with capacity c servicing a two class preemptive offered load.

Fix some class k of interest. Let $\mathcal{T}_k = \{(N_k, \bar{N}_k) : N_k + \bar{N}_k \leq c\}$ be the state space for the lumped process obtained under the split k aggregate occupancy partition. See Fig. 2.4. The set of states where the link is full is:

$$\mathcal{T}_k^{\text{full}} = \{(N_k, \bar{N}_k) : N_k + \bar{N}_k = c\} \subseteq \mathcal{T}_k. \tag{2.8}$$

A class k arrival that finds the link full (*i.e.*, $N_k + \bar{N}_k = c$) results in either the arriving call being

blocked (if $\bar{N}_k = 0$), or the arriving call being admitted but causing a preemption (if $\bar{N}_k > 0$):

$$\begin{aligned} \mathcal{T}_k^{\text{block}} &= \{(N_k, \bar{N}_k) : N_k = c\}, \\ \mathcal{T}_k^{\text{preempt}} &= \{(N_k, \bar{N}_k) : N_k + \bar{N}_k = c, \bar{N}_k > 0\}, \\ \mathcal{T}_k^{\text{full}} &= \mathcal{T}_k^{\text{block}} \cup \mathcal{T}_k^{\text{preempt}}. \end{aligned} \tag{2.9}$$

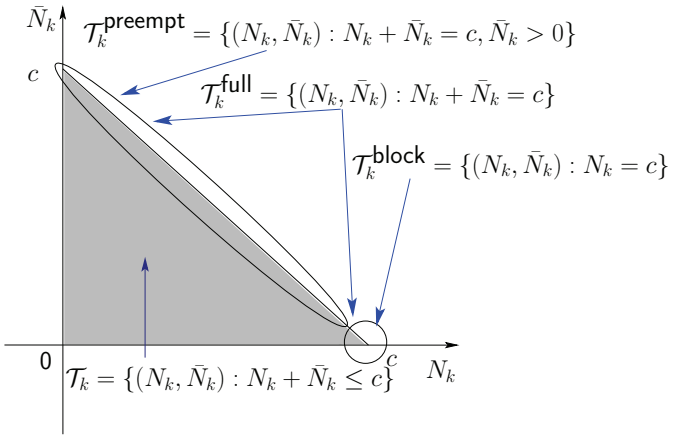


Figure 2.4: The state space $\mathcal{T}_k$ for an IL of capacity c. The x-axis (N_k) is the cumulative number of calls with higher or equal priority classes $1, \ldots, k$; the y-axis ($\bar{N}_k$) is the cumulative number of calls with lower priority classes $k+1, \ldots, K$. The set of states where the link is full ($N_k + \bar{N}_k = c$) may be decomposed into the blocking state ($\bar{N}_k = 0$) and the preempting states ($\bar{N}_k > 0$).

Define $E(\rho, c)$ as the Erlang-B blocking probability of a single-class $M/M/c/c$ queue with offered load ρ. If $\{n(t)\}$ is the Markov chain for the queue occupancy, then $E(\rho, c) = \mathbb{P}(n(t) = c)$, where t is either an arbitrary time or an arrival time.

By "Poisson arrivals see time averages" (PASTA), the probabilities that a typical class k arrival finds the system in a state in the sets $\mathcal{T}_k^{\text{full}}, \mathcal{T}_k^{\text{block}}, \mathcal{T}_k^{\text{preempt}}$ are found by summing the invariant distribution of $\{(N_k(t), \bar{N}_k(t))\}$, the Markov chain over the states comprising the event. The invariant distribution for this system is difficult to express in closed form, but the probabilities of the events of interest are fortunately more tractable. In particular, the probability that the link is full (with calls of any class) is $E(R_K, c)$, where $R_K = \rho_1 + \cdots + \rho_K$ is the cumulative offered load across all classes. Moreover, the probability that the link is filled with calls of classes $1, \ldots, k$ is $E(R_k, c)$. This

allows us to find the probability that an arriving class k call causes a preemption on the link:

$$\begin{aligned} \mathbb{P}((N_k(t), \bar{N}_k(t)) \in \mathcal{T}_k^{\text{full}}) &= E(R_K, c), \\ \mathbb{P}((N_k(t), \bar{N}_k(t)) \in \mathcal{T}_k^{\text{block}}) &= E(R_k, c), \\ \mathbb{P}((N_k(t), \bar{N}_k(t)) \in \mathcal{T}_k^{\text{preempt}}) &= E(R_K, c) - E(R_k, c). \end{aligned} \tag{2.10}$$

We now use this basic observation to obtain expressions for the preemption probabilities and rates.

Three priority classes ($K = 3$) on a single link. We first cover the case of $K = 3$ priority classes on a single link to build intuition. The case $K = 2$ on a single link is addressed by Helly [28]. It is vital to make a clear distinction between probabilities (of preemption, admission, blocking, *etc.*) and the corresponding rates. The probability of an event, say F, is denoted by $\mathbb{P}(F)$, the rate at which the event occurs is denoted by $\mathbb{R}(F)$. By (2.10), the *probability* that the link is full with at least one call of class 2 or 3 is $E(R_3, c) - E(R_1, c)$. In any such state, an arrival by a call of class 1 will cause a preemption of a call of class 3 (if any are present), else of class 2. By PASTA, the *rate* at which class 1 arrivals preempt active calls of class 2 or 3 is

$$\mathbb{R}(1 \text{ preempts}) = \mathbb{R}(1 \text{ preempts 2 or 3}) = \lambda_1(E(R_3, c) - E(R_1, c)). \tag{2.11}$$

This should be read as: "the rate that class 1 preempts is the rate that class 1 arrives, times the probability that the link is in a state requiring preemption". Similarly, the probability that the link is full with at least one call of class 3 is $E(R_3, c) - E(R_2, c)$. The rate at which calls of classes 1 and 2 preempt calls of class 3 is then

$$\begin{aligned} \mathbb{R}(3 \text{ is preempted}) &= \mathbb{R}(1 \text{ or } 2 \text{ preempt } 3) = \Lambda_2(E(R_3, c) - E(R_2, c)), \\ \mathbb{R}(1 \text{ preempts } 3) &= \lambda_1(E(R_3, c) - E(R_2, c)), \\ \mathbb{R}(2 \text{ preempts}) &= \mathbb{R}(2 \text{ preempts } 3) = \lambda_2(E(R_3, c) - E(R_2, c)). \end{aligned} \tag{2.12}$$

The rate that class 1 preempts class 2 is obtained from (2.11) and (2.12) using rate conservation:

$$\begin{aligned} \mathbb{R}(2 \text{ is preempted}) &= \mathbb{R}(1 \text{ preempts } 2 \text{ or } 3) - \mathbb{R}(1 \text{ preempts } 3) \\ &= \lambda_1(E(R_2, c) - E(R_1, c)). \end{aligned} \tag{2.13}$$

Note that (2.13) is independent of class 3's arrival rate, λ_3. This is because class 3 calls have no bearing on the admission and occupancy processes of class 2 calls. The probability of class 2 being preempted by class 1 is obtained from (2.13) using the ratio of rates:

$$\mathbb{P}(2 \text{ is preempted}) = \frac{\lambda_1}{\lambda_2}(E(R_2, c) - E(R_1, c)). \tag{2.14}$$

This probability is to be understood as a customer average: the fraction of class 2 arrivals that find themselves preempted is the long-run number/rate of preemptions over the long-run number/rate of class 2 arrivals.

K **priority classes on a single link.** Generalizing the result for $k = 3$, the probability that the link is full with at least one call of class $k+1$ through K is $E(R_K, c) - E(R_k, c)$, and the rate at which arrivals of classes $k < K$ preempt active calls of classes $k+1, \ldots, K$ is:

$$P_k = \mathbb{R}(k \text{ preempts}) = \lambda_k(E(R_K, c) - E(R_k, c)). \tag{2.15}$$

The *probability* of a class k arrival preempting an active call from the PL is obtained by taking the ratio of rates of (2.15) over the class k arrival rate:

$$\mathbb{P}(k \text{ preempts}) = E(R_K, c) - E(R_k, c). \tag{2.16}$$

The rate at which arriving class $i < K$ arrivals preempt active class $k > i$ calls is denoted by both

$P_{i,k} = Q_{k,i}$:

$$\begin{aligned} P_{i,k} &= \mathbb{R}(i \text{ preempts } k) = Q_{k,i} = \mathbb{R}(k \text{ preempted by } i) \\ &= \mathbb{R}(i \text{ preempts } k, \ldots, K) - \mathbb{R}(i \text{ preempts } k+1, \ldots, K) \\ &= \lambda_i(E(R_K, c) - E(R_{k-1}, c)) - \lambda_i(E(R_K, c) - E(R_k, c)) \\ &= \lambda_i(E(R_k, c) - E(R_{k-1}, c)). \end{aligned} \tag{2.17}$$

The rate at which active class $k > 1$ calls are preempted by arriving calls is

$$\begin{aligned} Q_k &= \mathbb{R}(k \text{ preempted}) = \mathbb{R}(k \text{ preempted by } 1, \ldots, k-1) \\ &= \sum_{i=1}^{k-1} \mathbb{R}(k \text{ preempted by } i) = \Lambda_{k-1}(E(R_k, c) - E(R_{k-1}, c)). \end{aligned} \tag{2.18}$$

The probability of a class k call being preempted is the ratio of rates of (2.18) over the class k arrival rate:

$$\mathbb{P}(k \text{ is preempted}) = \frac{\Lambda_{k-1}}{\lambda_k}(E(R_k, c) - E(R_{k-1}, c)). \tag{2.19}$$

2.4.5 Blocking and preemption on the PL and system

Applying Theorem 2 to the above analysis yields the following theorem on the performance of the PL and the system, see Fig. 2.5.

Theorem 5. The preemption rates P_k^p (P_k^s), $P_{i,k}^p = Q_{k,i}^p$ ($P_{i,k}^s = Q_{k,i}^s$) and Q_k^p (Q_k^s) for each (k, i) on the PL (system) are given by (2.15), (2.17) and (2.18) respectively by replacing c with c^p (c^s).

The following theorem employs the rate conservation equations from Fig. 2.2 to find departure, admission and blocking rates for the PL, BL and the system.

Theorem 6. Rate conservation in Fig. 2.2 ensures:

$$D_k^p = A_k^p - Q_k^p,\ D_k^s = A_k^s - Q_k^s,\ D_k^b = D_k^s - D_k^p. \tag{2.20}$$

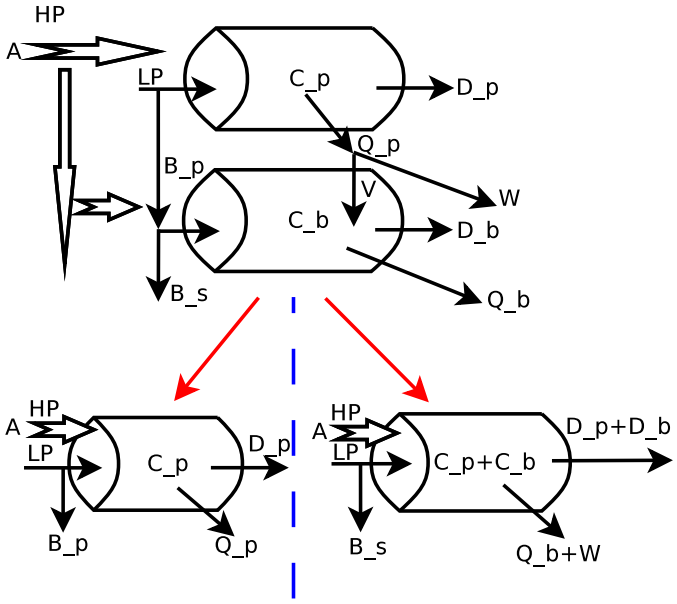


Figure 2.5: The two parallel links are viewd the PL and the system as a whole.

The admission / blocking rates on the PL, BL, and system are:

$$
\begin{aligned}
B_k^p &= \lambda_k E(R_k, c^p), \qquad (2.21)\\
A_k^p &= \lambda_k \bar{E}(R_k, c^p),\\
B_k^s = B_k^b &= \lambda_k E(R_k, c^s),\\
A_k^s &= \lambda_k \bar{E}(R_k, c^s),\\
A_k^b &= B_k^p - B_k^b = \lambda_k \left(E(R_k, c^p) - E(R_k, c^s)\right).
\end{aligned}
$$

The departure rates from the PL, BL, and system are:

$$
\begin{aligned}
D_k^p &= \lambda_k \bar{E}(R_k, c^p) - \Lambda_{k-1}(E(R_k, c^p) - E(R_{k-1}, c^p)), \qquad (2.22)\\
D_k^s &= \lambda_k \bar{E}(R_k, c^s) - \Lambda_{k-1}(E(R_k, c^s) - E(R_{k-1}, c^s)),\\
D_k^b &= \left(\lambda_k \bar{E}(R_k, c^s) - \Lambda_{k-1}(E(R_k, c^s) - E(R_{k-1}, c^s))\right)\\
&- \left(\lambda_k \bar{E}(R_k, c^p) - \Lambda_{k-1}(E(R_k, c^p) - E(R_{k-1}, c^p))\right).
\end{aligned}
$$

The departure rates are obtained by substituting the expressions in (2.18) into the rate conservation equations (2.20).

2.4.6 Limitations of Erlang-B analysis

Several of the rates in Fig. 2.2 have now been characterized in terms of the model primitives: P_k^p, Q_k^p, P_k^s, Q_k^s (Theorem 5); B_k^p, A_k^p, B_k^s, A_k^s, B_k^b, A_k^b, D_k^p, D_k^s and D_k^b (Theorem 6). It remains to compute V_k, W_k, Q_k^b and T_k, O_k, P_k^b. Rate conservation equations from Fig. 2.2 yield three equations for the three unknowns V_k, W_k, Q_k^b:

$$\begin{aligned} V_k + W_k &= A_k^p - D_k^p = Q_k^p, \\ W_k + Q_k^b &= A_k^s - D_k^s = Q_k^s, \\ Q_k^b - V_k &= A_k^b - D_k^b. \end{aligned} \tag{2.23}$$

The three equations are rate conservation expressions for the PL, the system, and the BL respectively. These three equations are linearly dependent: adding the first and the third yields the second. It is therefore not possible to use this system of equations to obtain expressions for all three quantities V_k, W_k, Q_k^b. The three unknowns T_k, O_k, P_k^b face a similar limitation with two independent equations:

$$T_k + O_k = P_k^b, \quad O_k + P_k^b = P_k^s. \tag{2.24}$$

The lack of a full rank system of equations in both cases stems from a fundamental limitation of the model. There are two systems we can analyze using the Erlang approach: the PL by itself (the link view) and the two links together (the system view). We can not analyze the BL by itself since its state depends on that of the PL. The basic problem is that we cannot determine the probability that a call preempted from the PL will be transferred to the BL versus being dropped. Determining this probability would require a characterization of the stationary occupancy distribution of the BL. This stationary distribution is difficult to obtain because the arrivals to the BL are not Poisson, on account of both the overflow admissions and the transfers [34]. Put differently, even in the simplest

case of $K = 2$ priority classes, only four elementary events may be expressed in terms of Erlang-B probabilities:

- Class 1 blocking on PL: $\{n_1^p = c^p\}$,
- Class 1 blocking from the system: $\{n_1^s = c^s\}$,
- Class 2 blocking on PL: $\{n_1^p + n_2^p = c^p\}$,
- Class 2 blocking from the system: $\{n_1^s + n_2^s = c^s\}$.

All the admission, preemption, and departure events are combinations of these four elementary events. Determining the probability of a transfer from the PL to the BL, however, requires computing the probability of

$$\{n_1^p + n_2^p = c^p,\ n_2^p > 0,\ n_1^b + n_2^b < c^b\}. \tag{2.25}$$

That is, a transfer requires *i*) the PL is filled, *ii*) there are one or more preemptable class 2 calls on the PL, and *iii*) the BL is not filled. This type of event is not expressible as a combination of the above four elementary events.

2.5 Homogeneous service rates: the many small users regime

In this section we obtain simplified expressions for the preemption rates P_k (2.15), $P_{i,k} = Q_{k,i}$ (2.17), Q_k (2.18) in the "many small users" regime, obtained by letting the arrival rates and link capacity scale linearly to infinity. In particular, consider a sequence of links[3], indexed by c, where link c has capacity c and the arrival rates are $\lambda_k^{(c)} = \lambda_k^* c$ and the offered loads are $\rho_k^{(c)} = \frac{\lambda_k^*}{\mu} c$ for each $k = 1, \ldots, K$. The common service rate is held constant: μ is independent of k and c. This is called the many small users regime because for c large the arrival rates and capacities are large enough to ensure a large number of active calls (users), where each call consumes a negligibly small fraction of the link capacity.

[3]The phrase "sequence of links" should not be misconstrued as a network of parallel links. There is only one IL, we are simply scaling its capacity and offered load.

2.5.1 Single class link in the many small users regime

Define $\rho^* = \frac{\lambda^*}{\mu} = \frac{\rho^{(c)}}{c}$ as the normalized offered load. The underloaded, critically loaded, and overloaded regimes then correspond to the cases when $\rho^* < 1$, $\rho^* = 1$, and $\rho^* > 1$, respectively. In the critically loaded case, it is conventional to define $\rho^* = \rho^{*,(c)} = 1 \pm \frac{\delta}{\sqrt{c}}$, so that $\rho^{(c)} = c \pm \delta\sqrt{c}$. Intuitively, one may think of δ as the number of standard deviations of the offered Poisson load above or below the link capacity.

The technical literature on asymptotic blocking probability approximations in the many small users regime is quite large (see, *e.g.*,[21] and the references therein); for our purposes the results by Mitra and Morrison[35] are sufficient. This paper is appealing as it is the first to develop a *uniform asymptotic approximations* (UAA) for the Erlang blocking probability in the many small users regime, *i.e.*, an expression for the blocking probability with a relative error (in this case, $O(1/c)$), valid for *all* values of ρ^*.[4] Earlier results give expressions subject to some restriction on ρ^*. The following Theorem 7 is obtained in a straightforward way from[35] by either taking the Poisson limit in Propositions 5.2, 5.3, and 5.4, or by specializing Proposition 5.5 to the three regimes for ρ^*.

Theorem 7. *(Mitra and Morrison[35]).* Consider a sequence of loss links indexed by c, where link c has capacity c and the offered load is $\rho^{(c)} = \rho^* c$. A uniform asymptotic approximation for the blocking probability is:

$$E(\rho^{(c)}, c) = \begin{cases} \frac{\mathrm{e}^{-c\left(\rho^* - 1 - \log \rho^*\right)}}{\sqrt{2\pi c}} \left(1 + O\left(\frac{1}{c}\right)\right), & \rho^* < 1 \\ \sqrt{\frac{2}{\pi}} \frac{1}{\sqrt{c} \pm \frac{\delta}{2}} \frac{\mathrm{e}^{-\frac{\delta^2}{2}}}{\mathrm{Erfc}\left(\pm\frac{\delta}{\sqrt{2}}\right)} + O\left(\frac{1}{c}\right), & \rho^* = 1 \pm \frac{\delta}{\sqrt{c}} \\ 1 - \frac{1}{\rho^*} + O\left(\frac{1}{c}\right), & \rho^* > 1 \end{cases} . \tag{2.26}$$

Here, $\mathrm{Erfc}(x)$ is the complimentary error function. Theorem 7 is stated with a relative error of $O(1/c)$ for the underloaded case ($\rho^* < 1$), and with absolute error of $O(1/c)$ for the critical and overloaded cases. The fact that $\rho^* - 1 - \log \rho^* > 0$ means the blocking probability in the underloaded case is exponentially small, $E(\rho^{(c)}, c) = O(c^{-1/2}\mathrm{e}^{-c})$ for $\rho^* < 1$. Absolute error is

[4] Actually[35] develops expressions for multiple classes, where each class has a distinct arrival rate *and* each class consumes a distinct number of circuits per call. For our needs it is sufficient to specialize their result to a single class, where each call of that class consumes a single circuit, *i.e.*, the classical $M/M/c/c$ queue.

preferable to relative error for our needs because the preemption probabilities require us to compute the *difference* of two Erlang blocking probabilities.

2.5.2 Preemptive link in the many small users regime

The corollary below gives a UAA for the *difference* of two Erlang blocking probabilities in the many small users regime.

Corollary 1. Consider the same setup as Theorem 7, and let ρ_1^*, ρ_2^* represent two normalized offered loads. Require $\rho_1^* > \rho_2^*$ and $\delta_1 < \delta_2$. Then: $E(\rho_1^{(c)}, c) - E(\rho_2^{(c)}, c) =$

$$\begin{cases} \frac{1}{\sqrt{2\pi c}} \left(\mathrm{e}^{-c(\rho_1^* - 1 - \log \rho_1^*)} - \mathrm{e}^{-c(\rho_2^* - 1 - \log \rho_2^*)} \right) \left(1 + O\left(\frac{1}{c}\right) \right), \\ \qquad \rho_2^* < \rho_1^* < 1, \quad \text{(region (1))} \\ \sqrt{\frac{2}{\pi}} \frac{1}{\sqrt{c} \pm \frac{\delta_1}{2}} \frac{\mathrm{e}^{-\frac{\delta_1^2}{2}}}{\mathrm{Erfc}\left(\pm \frac{\delta_1}{\sqrt{2}} \right)} + O\left(\frac{1}{c}\right), \\ \qquad \rho_2^* < 1, \rho_1^* = 1 \pm \frac{\delta_1}{\sqrt{c}}, \quad \text{(region (2))} \\ 1 - \frac{1}{\rho_1^*} + O\left(\frac{1}{c}\right), \\ \qquad \rho_2^* < 1 < \rho_1^*, \quad \text{(region (3))} \\ \sqrt{\frac{2}{\pi}} \left(\frac{1}{\sqrt{c} \pm \frac{\delta_1}{2}} \frac{\mathrm{e}^{-\frac{\delta_1^2}{2}}}{\mathrm{Erfc}\left(\pm \frac{\delta_1}{\sqrt{2}} \right)} - \frac{1}{\sqrt{c} \pm \frac{\delta_2}{2}} \frac{\mathrm{e}^{-\frac{\delta_2^2}{2}}}{\mathrm{Erfc}\left(\pm \frac{\delta_2}{\sqrt{2}} \right)} \right) + O\left(\frac{1}{c}\right), \\ \qquad \rho_2^* = 1 \pm \frac{\delta_2}{\sqrt{c}}, \rho_1^* = 1 \pm \frac{\delta_1}{\sqrt{c}}, \quad \text{(region (4))} \\ 1 - \frac{1}{\rho_1^*} - \sqrt{\frac{2}{\pi}} \frac{1}{\sqrt{c} \pm \frac{\delta_2}{2}} \frac{\mathrm{e}^{-\frac{\delta_2^2}{2}}}{\mathrm{Erfc}\left(\pm \frac{\delta_2}{\sqrt{2}} \right)} + O\left(\frac{1}{c}\right), \\ \qquad \rho_2^* = 1 \pm \frac{\delta_2}{\sqrt{c}}, 1 < \rho_1^*, \quad \text{(region (5))} \\ \frac{1}{\rho_2^*} - \frac{1}{\rho_1^*} + O\left(\frac{1}{c}\right), \\ \qquad 1 < \rho_2^* < \rho_1^*, \quad \text{(region (6))} \end{cases} \tag{2.27}$$

The quantities ρ_1^*, ρ_2^* represent two generic normalized offered loads; they should not be thought of as class–specific. The six regions mentioned in the corollary are sketched in Fig. 2.6. They correspond to (1) both loads $1, 2$ underloaded, (2) load 2 is underloaded and load 1 is critically loaded, (3) load 2 is underloaded and load 1 is overloaded, (4) both loads $1, 2$ critically loaded, (5) load 2 is critically loaded and load 1 is overloaded, and (6) both loads $1, 2$ overloaded. With this corollary it is straightforward to obtain the many small users UAA for the single IL preemption

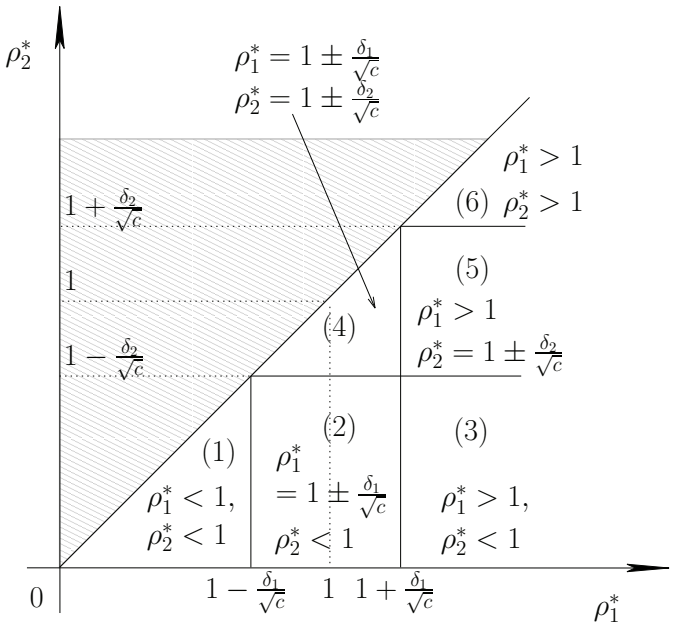


Figure 2.6: Illustration of the six regimes of $\rho_1^* > \rho_2^*$. The six regions are addressed in Corollary 1.

rates, given in the following theorem. Let $R_k^{(c)} = \rho_1^{(c)} + \cdots + \rho_k^{(c)}$, and $\Lambda_k^* = \lambda_1^* + \cdots + \lambda_k^*$.

Theorem 8. UAA for the preemption rates P_k (2.15), $P_{i,k} = Q_{k,i}$ (2.17), Q_k (2.18) in the many small users regime are:

$$\begin{aligned} \frac{P_k^{(c)}}{c} &= \lambda_k^* \left(E(R_K^{(c)}, c) - E(R_k^{(c)}, c) \right), k < K, \\ \frac{P_{i,k}^{(c)}}{c} &= \lambda_i^* \left(E(R_k^{(c)}, c) - E(R_{k-1}^{(c)}, c) \right), i < k, \\ \frac{Q_k^{(c)}}{c} &= \Lambda_{k-1}^* \left(E(R_k^{(c)}, c) - E(R_{k-1}^{(c)}, c) \right), k > 1. \end{aligned} \tag{2.28}$$

where differences of Erlang blocking probabilities are given by Corollary 1. In the "both underloaded" case the absolute error is $O(c^{-1/2}\mathrm{e}^{-c})$; otherwise the absolute error is $O(1/c)$.

By Theorem 2, the above Theorem can be specialized to give the many small users approximation for the PL and system view preemption rates by replacing c with c^p and c^s respectively. Consider a particular system of interest with fixed capacities c^p, c^b and $c^s = c^p + c^b$, fixed arrival rates $\lambda_1, \ldots, \lambda_K$, and fixed service rate μ. For the PL use Theorem 8 with $\lambda_k^* = \frac{\lambda_k}{c^p}$ for $k = 1, \ldots, K$, and for the

system use Theorem 8 with $\lambda_k^* = \frac{\lambda_k}{c^s}$ for $k = 1, \ldots, K$.

2.5.3 Preemptive overloaded link in the many small users regime

It is insightful to show explicitly the preemption rates for the overloaded regime, where the expressions are simplified:

Corollary 2. If $\Lambda_k^* > \mu$ then the expression for $P_k^{(c)}$ is obtained from the "both overloaded" regime ($1 < \rho_2^* < \rho_1^*$, region (6)) in Corollary 1, yielding:

$$\frac{P_k^{(c)}}{\mu c} = \frac{\lambda_k^*}{\Lambda_k^*} \times \frac{\Lambda_K^* - \Lambda_k^*}{\Lambda_K^*} + O\left(\frac{1}{c}\right), k < K. \tag{2.29}$$

If, moreover, $\Lambda_{k-1}^* > \mu$, then the expressions for $P_{i,k}^{(c)}$ and $Q_k^{(c)}$ are similarly obtained from the "both overloaded" regime in Corollary 1, yielding:

$$\begin{aligned} \frac{P_{i,k}^{(c)}}{\mu c} &= \frac{\lambda_i^*}{\Lambda_{k-1}^*} \times \frac{\lambda_k^*}{\Lambda_k^*} + O\left(\frac{1}{c}\right), i < k, \\ \frac{Q_k^{(c)}}{\mu c} &= \frac{\lambda_k^*}{\Lambda_k^*} + O\left(\frac{1}{c}\right), k > 1. \end{aligned} \tag{2.30}$$

The expressions in Corollary 2 admit a natural interpretation. In particular, in the "both overloaded" regime, $P_k^{(c)}$, the rate that class k calls cause preemption (of calls of class $k+1, \ldots, K$) is proportional to the product of two rate ratios: *i*) the rate of class k arrivals over the rate of arrivals that can preempt $k+1, \ldots, K$, and *ii*) the rate of arrivals of calls that k can preempt over the overall rate of arrivals. An analogous interpretation holds for $P_{i,k}^{(c)}$. Finally, note that, as expected, $P_{i,k}^{(c)}$ and $Q_k^{(c)}$ are independent of $\lambda_{k+1}, \ldots, \lambda_K$.

2.5.4 Sensitivities of a preemptive overloaded link in the many small users regime

It is insightful to study the sensitivities of the expressions in Corollary 2 on the parameters λ_i, λ_k.

Corollary 3. The sensitivities of the preemption rates in Corollary 2 are:

$$\begin{aligned}
\frac{1}{\mu c}\frac{\partial P_k^{(c)}}{\partial \lambda_k^*} &= \frac{(\Lambda_K^* - \Lambda_k^*)((\Lambda_k^* - \lambda_k^*)(\Lambda_K^* - \lambda_k^*) - (\lambda_k^*)^2)}{(\Lambda_k^*)^2(\Lambda_K^*)^2}, \\
\frac{1}{\mu c}\frac{\partial P_{i,k}^{(c)}}{\partial \lambda_i^*} &= \frac{\lambda_k^*((\Lambda_{k-1}^* - \lambda_i^*)(\Lambda_k^* - \lambda_i^*) - (\lambda_i^*)^2)}{(\Lambda_{k-1}^*)^2(\Lambda_k^*)^2}, \\
\frac{1}{\mu c}\frac{\partial P_{i,k}^{(c)}}{\partial \lambda_k^*} &= \frac{\lambda_i^*}{(\Lambda_k^*)^2}, \\
\frac{1}{\mu c}\frac{\partial Q_k^{(c)}}{\partial \lambda_k^*} &= \frac{\Lambda_{k-1}^*}{(\Lambda_k^*)^2}.
\end{aligned} \tag{2.31}$$

Holding each λ_l^* aside from λ_k^* constant, define

$$\lambda_k^{*,\max} = \sqrt{(\Lambda_k^* - \lambda_k^*)(\Lambda_K^* - \lambda_k^*)}. \tag{2.32}$$

Corollary 3 asserts that in the "both overloaded" regime, $P_k^{(c)}$ is monotone increasing in λ_k^* for $\lambda_k^* < \lambda_k^{*,\max}$, and monotone decreasing for $\lambda_k^* > \lambda_k^{*,\max}$. Intuitively, increasing λ_k at first increases the amount of preempted load of classes $k+1, \ldots, K$, but eventually the class k traffic is preventing the admission of traffic of classes $k+1, \ldots, K$, which in turn reduces the preemption rate. The expression for $\frac{1}{\mu c}\frac{\partial P_{i,k}^{(c)}}{\partial \lambda_i^*}$ has a similar form to that of $\frac{1}{\mu c}\frac{\partial P_k^{(c)}}{\partial \lambda_k^*}$. Holding each λ_l^* aside from λ_i^* constant, define

$$\lambda_{i,k}^{*,\max} = \sqrt{(\Lambda_{k-1}^* - \lambda_i^*)(\Lambda_k^* - \lambda_i^*)}. \tag{2.33}$$

The corollary further asserts that in the "both overloaded" regime, $P_{i,k}^{(c)}$ is monotone increasing in λ_i^* for $\lambda_i^* < \lambda_{i,k}^{*,\max}$, and monotone decreasing for $\lambda_i^* > \lambda_{i,k}^{*,\max}$. The same intuition applies to explain the result, *mutatis mutandis*. Next, the corollary asserts $P_{i,k}$ is monotone increasing in λ_k, while $\frac{1}{\mu c}\frac{\partial P_{i,k}^{(c)}}{\partial \lambda_k^*}$ goes to zero. Increasing λ_k^* increases $P_{i,k}^{(c)}$ because the system is more likely to be full, with a non-decreasing fraction of the link occupied by calls of class k. This increase is subject to a law of diminishing returns, however, since the fraction of the link occupied by calls of class k is limited by the fixed fraction occupied by calls of class $1, \ldots, k-1$. Note that the sensitivity is independent of $\lambda_{k+1}^*, \ldots, \lambda_K^*$. Finally, the corollary asserts an almost identical relationship for $Q_k^{(c)}$. The same

intuition applies to explain the result, *mutatis mutandis*.

2.6 Heterogeneous service rates

In this section we discuss the reasons why the heterogeneous service rates case is in general intractable. We then discuss an approximate solution valid in a time-scale separation regime. Throughout this section we focus on an IL serving a multi-class preemptive load. The results are specialized to the PL and system CTMCs by application of Theorem 2.

2.6.1 Lumpability

The primary reason for the intractability of the heterogeneous service rates case is the fact that the CTMC $\{\mathbf{n}(t)\}$ is not lumpable under a partition aligned with the performance metrics of interest. It is lumpable under other partitions, but, as will be shown, these other partitions are only valuable in obtaining approximate expressions, valid under certain time-scale separation assumptions.

Occupancy partitions. We introduce two occupancy partitions for the IL Markov chain $\{\mathbf{n}(t)\}$.

Definition 5. The *aggregate occupancy partition* (aop) of $\mathcal{S}$ is $\mathcal{S}_m^{\text{aop}} = \{\mathbf{n} \in \mathcal{S} : N_K = m\}$ for each $m = 1, \ldots, c$.

Definition 6. The *priority k occupancy partition* (pop-k) of $\mathcal{S}$ is $\mathcal{S}_m^{\text{pop},k} = \{\mathbf{n} \in \mathcal{S} : n_k = m\}$, for each occupancy level $m = 0, \ldots, c$ and some priority level $k = 1, \ldots, K$.

The *aop*, *pop*-1, and *pop*-2 partitions are shown in Fig. 2.7 for the case of an IL with capacity $c = 2$ and $K = 2$ priority classes. The following theorem identifies when the Markov chain $\{\mathbf{n}(t)\}$ is lumpable over these partitions.

Theorem 9. The CTMC $\{\mathbf{n}(t)\}$ is:

1. Lumpable under the *aop* with *homogeneous* service rates, and is Markovian across subsets.
2. *Not* lumpable under the *aop* with *heterogeneous* service rates, and therefore not Markovian across subsets.
3. Lumpable under the *pop*-1 with homogeneous or heterogeneous service rates, and so Markovian across subsets.

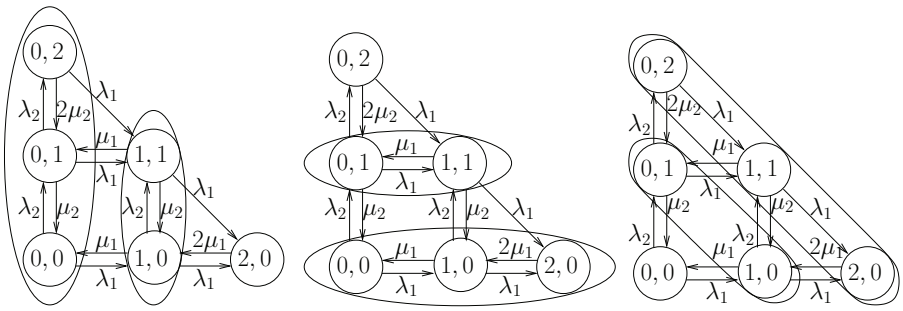


Figure 2.7: Illustration of the occupancy partitions for $c = 2$ and $K = 2$. Each state shown represents an occupancy of $\mathbf{n} = (n_1, n_2)$. **Left:** the priority 1 occupancy partition, **middle:** the priority 2 occupancy partition, **right**: the aggregate occupancy partition.

4. *Not* lumpable under the *pop-k* (for $k > 1$) with homogeneous or heterogeneous service rates, and therefore not Markovian across subsets.

Proof. Recall the definition of the transition matrix of the lumped chain given in Theorem 3. It is clear that $R_{m,m'} = 0$ for all m, m' such that $\|m - m'\| > 1$ for all partitions under consideration. Hence it suffices to consider transitions from occupancy level m to $m+1$ and to $m-1$. Let $\mathbf{n} \in \mathcal{S}_m$ be a state in occupancy level m.

1. The transition rate from $\mathbf{n}$ in aggregate occupancy level $m > 0$ to aggregate occupancy level $m - 1$ is

$$\sum_{\mathbf{n}' \in \mathcal{S}^{\text{aop}}_{m-1}} q_{\mathbf{n},\mathbf{n}'} = n_1\mu + \cdots + n_K\mu = m\mu, \tag{2.34}$$

and the transition rate from $\mathbf{n}$ in aggregate occupancy level $m < c$ to aggregate occupancy level $m + 1$ is

$$\sum_{\mathbf{n}' \in \mathcal{S}^{\text{aop}}_{m+1}} q_{\mathbf{n},\mathbf{n}'} = \lambda_1 + \cdots + \lambda_K = \Lambda_K. \tag{2.35}$$

In both cases the transition rate is independent of the state $\mathbf{n}$.

2. The transition rate from $\mathbf{n}$ in aggregate occupancy level $m > 0$ to aggregate occupancy level $m - 1$ is:

$$\sum_{\mathbf{n}' \in \mathcal{S}^{\text{aop}}_{m-1}} q_{\mathbf{n},\mathbf{n}'} = n_1\mu_1 + \cdots + n_K\mu_K. \tag{2.36}$$

The transition rate depends upon the state $\mathbf{n}$.

3. The transition rate from $\mathbf{n}$ in priority 1 occupancy level $m > 0$ to priority 1 occupancy level $m-1$ is:

$$\sum_{\mathbf{n}' \in \mathcal{S}_{m-1}^{\text{pop},1}} q_{\mathbf{n},\mathbf{n}'} = n_1\mu_1 = m\mu_1, \tag{2.37}$$

and the transition rate from $\mathbf{n}$ in priority 1 occupancy level $m < c$ to priority 1 occupancy level $m+1$ is

$$\sum_{\mathbf{n}' \in \mathcal{S}_{m+1}^{\text{pop},1}} q_{\mathbf{n},\mathbf{n}'} = \lambda_1. \tag{2.38}$$

In both cases the transition rate is independent of the state $\mathbf{n}$.

4. The transition rate from $\mathbf{n}$ in priority $k > 1$ occupancy level $m > 0$ to priority k occupancy level $m+1$ is:

$$\sum_{\mathbf{n}' \in \mathcal{S}_{m+1}^{\text{pop},k}} q_{\mathbf{n},\mathbf{n}'} = \lambda_k \mathbf{1}_{n_1+\cdots+n_k<c}. \tag{2.39}$$

The transition rate depends upon the state $\mathbf{n}$.

□

The key reason why the chain is lumpable under the *aop* is that *preemptions do not change the aggregate occupancy level*. It is also worth noting that the CTMC is lumpable under *pop*-1 precisely because class 1 has preemptive priority over all other calls. The multi-class model where priorities are not preemptive is not lumpable under *pop*-1.

The *aop* is a valuable partition for the preemption model because the preemption probability can be expressed in terms of the probability of being in aggregate occupancy level c. Unfortunately, as we have seen, *aop* is only lumpable under homogeneous service rates. The *pop*-1 is appealing as it is lumpable under heterogeneous service rates, but this is of less value than *aop* because the partition does not map easily to the performance metrics of interest, *i.e.*, the preemption probabilities and rates. Nonetheless, the *pop*-1 is still of value in computing performance, especially when a time-scale separation holds among the various classes.

2.6.2 Decomposability and time-scale separation

Whereas lumpability refers to a partition where the transition across subsets is not state-dependent, decomposability refers to a partition where the transition rate across subsets is zero, *i.e.*, the chain is reducible. Thus decomposability is a special case of lumpability. Both lumpable and decomposable may be relaxed to quasi-lumpable (QL) and nearly completely decomposable (NCD), respectively. A CTMC is said to be ϵ *quasi-lumpable* if $\mathbf{Q}$ can be decomposed as $\mathbf{Q} = \mathbf{Q}^{-} + \mathbf{Q}^{\epsilon}$ where $\mathbf{Q}^{-}$ is lumpable and the largest element in $\mathbf{Q}^{\epsilon}$ has absolute value no larger than ϵ. A CTMC is said to be *nearly completely decomposable* if the states may be arranged into blocks such that $\mathbf{Q} = \mathbf{Q}^{+} + \mathbf{Q}^{\delta}$, where $\mathbf{Q}^{+}$ is block diagonal, and the norm of the off-diagonal transition rates, $\|\mathbf{Q}^{\delta}\|$ is the degree of coupling. The intuition for QL is that "most" transitions across subsets are state-independent, and the intuition for NCD is that "most" transitions are within (rather than across) subsets. Just as decomposability implies lumpability, Dayar and Stewart have shown that NCD implies QL, but the inverse need not hold [31]. In other words, NCD is a stronger condition than QL. This is natural since QL asserts the transitions across the subsets have a simple form, whereas NCD asserts the transitions across the subsets may be effectively ignored.

The previous subsection identified the priority 1 occupancy partition as lumpable, but pointed out that this by itself is of limited value since the partition does not map easily to the computation of the performance metrics of interest, *i.e.*, the preemption rates and probabilities. We now establish that the priority 1 occupancy partition is NCD under a time-scale separation among classes. A thorough discussion of time-scale separation for *discrete* time Markov chains is given in the book by Yin and Zhang [36]; Reiman and Schmitt use time-scale separation for a multi-class *non-preemptive* load on a loss link [37]. We now establish that the *pop*-1 is NCD under a time-scale separation among classes.

Definition 7. The arrival rates and service rates obey a *high-slow low-fast (*hslf*) time-scale separation* if

$$\lambda_1 \ll \cdots \ll \lambda_K, \quad \mu_1 \ll \cdots \ll \mu_K. \tag{2.40}$$

They obey a *high-fast low-slow (*hfls*) time-scale separation* if

$$\lambda_1 \gg \cdots \gg \lambda_K, \quad \mu_1 \gg \cdots \gg \mu_K. \tag{2.41}$$

Intuitively, under both types of time-scale separation there are K timescales, one for each class k. Under the *hslf* time-scale separation the highest priority class is on the slowest time scale, and the lowest priority class is on the fastest time scale; the reverse is true under the *hfls* time-scale separation. The degree of time-scale separation is not specified. Although there is an extensive body of literature quantifying the degree of coupling among the subsets in an NCD analysis, to our knowledge all such quantification is performed for DTMCs (see the book [36] for time-scale separation for DTMCs), and these metrics do not carry over naturally to CTMCs. [5] See Dayar and Stewart [31] and Meyer [38] and the references therein. As will be shown, the degree of time-scale separation is directly proportional to the degree of coupling. Lacking a precise measure for the latter, it is inessential to specify the former. The impact of the degree of time-scale separation on the performance approximation is studied in §2.7. Our approach is similar in spirit to Reiman and Schmitt, who use time-scale separation for a continuous time model of a multi-class *non-preemptive* load on a loss link [37].

Theorem 10. The CTMC $\{\mathbf{n}(t)\}$ is NCD under the *pop*-1 and the *hslf* time-scale separation assumption. It is not NCD under the *pop*-1 and the *hfls* time-scale separation assumption. It is also not NCD under any *pop-k* with $k > 1$, for either time-scale separation assumption.

Proof. Order the states in $\mathcal{S}$ by their priority 1 occupancy level: all states in $\mathcal{S}_0^{\text{pop},1}$ are listed before all states in $\mathcal{S}_1^{\text{pop},1}$, and so on. Because the only transitions across subsets are from priority 1 occupancy level m to $m-1$ and $m+1$, it follows that the infinitesimal generator matrix $\mathbf{Q}$ may be

[5] It is straightforward to show that the embedded discrete time (jump) chain associated with the CTMC of interest in this work may demonstrate a high degree of coupling, even though the CTMC is decomposable. Thus computing the degree of coupling for the embedded DTMC is an unreliable measure of the degree of coupling of the original CTMC.

written in block tridiagonal form:

$$\mathbf{Q} = \begin{bmatrix} \mathbf{Q}_0 & \mathbf{A}_0 & & & & & \\ \mathbf{D}_1 & \mathbf{Q}_1 & \mathbf{A}_1 & & & & \\ & & \ddots & & & & \\ & & & \mathbf{D}_{c-1} & \mathbf{Q}_{c-1} & \mathbf{A}_{c-1} & \\ & & & & \mathbf{D}_c & \mathbf{Q}_c & \end{bmatrix}. \tag{2.42}$$

Here $\mathbf{Q}_m$ contains the rates of all transitions among states $\mathbf{n}, \mathbf{n}' \in \mathcal{S}_m^{\text{pop},1}$, and is of dimensions $\|\mathcal{S}_m^{\text{pop},1}\| \times \|\mathcal{S}_m^{\text{pop},1}\|$, for each $m = 0, \ldots, c$. The *arrival* matrix $\mathbf{A}_m$ contains the rates of all transitions among states $\mathbf{n} \in \mathcal{S}_m^{\text{pop},1}$ and $\mathbf{n}' \in \mathcal{S}_{m+1}^{\text{pop},1}$, and is of dimensions $\|\mathcal{S}_m^{\text{pop},1}\| \times \|\mathcal{S}_{m+1}^{\text{pop},1}\|$, for each $m = 0, \ldots, c-1$. Similarly, the *departure* matrix $\mathbf{D}_m$ contains the rates of all transitions among states $\mathbf{n} \in \mathcal{S}_m^{\text{pop},1}$ and $\mathbf{n}' \in \mathcal{S}_{m-1}^{\text{pop},1}$, and is of dimensions $\|\mathcal{S}_m^{\text{pop},1}\| \times \|\mathcal{S}_{m-1}^{\text{pop},1}\|$, for each $m = 1, \ldots, c$. The elements of $\mathbf{A}_m$ are

$$\mathbf{A}_m(\mathbf{n}, \mathbf{n}') = \begin{cases} \lambda_1, & (\mathbf{n}' = \mathbf{n} + \mathbf{e}_1,\ n_1 + \cdots + n_K < c) \\ & \text{or } (\mathbf{n}' = \mathbf{n} + \mathbf{e}_1 - \mathbf{e}_k,\ n_1 + \cdots + n_K = c, \\ & k = \max\{j > 1 : n_j > 0\}) \\ 0, & \text{else} \end{cases}. \tag{2.43}$$

In words, the only transitions from $\mathcal{S}_m^{\text{pop},1}$ to $\mathcal{S}_{m+1}^{\text{pop},1}$ are at rate λ_1, and these occur if either the link is not full, or is full and the transition marks the preemption of the lowest priority active stream. The elements of $\mathbf{D}_m$ are

$$\mathbf{D}_m(\mathbf{n}, \mathbf{n}') = \begin{cases} m\mu_1, & \mathbf{n}' = \mathbf{n} - \mathbf{e}_1 \\ 0, & \text{else} \end{cases}. \tag{2.44}$$

The expressions for $\mathbf{A}_m, \mathbf{D}_m$ make clear that they depend solely upon λ_1, μ_1, and not upon λ_k, μ_k for $k > 1$. This suffices to ensure that the CTMC under the *pop*-1 is NCD provided the *hslf* time-scale separation assumption is valid. The rest of the proof is similar. □

The tri-diagonal form of the rate matrix for an IL under *pop*-1 and *pop*-2 for the special case

of $c = 2$ and $K = 2$ is shown in Table 2.2. The off-diagonal elements make clear that the *pop*-1 is NCD under the *hslf* separation assumption, and that the *pop*-2 is not NCD under either time-scale separation assumption.

Table 2.2: Rate matrix for a single link under the *pop*-1 (left) and *pop*-2 (right) for the special case of $c = 2$ and $K = 2$.

	0,0	0,1	0,2	1,0	1,1	2,0
0,0		λ_2	‖	λ_1		
0,1	μ_2		λ_2‖		λ_1	
0,2		$2\mu_2$	‖		λ_1	
	—	—	—	—	—	
1,0	μ_1		‖		λ_2‖	λ_1
1,1		μ_1	‖	μ_2	‖	λ_1
				—	—	—
2,0				$2\mu_1$	‖	

	0,0	1,0	2,0	0,1	1,1	0,2
0,0		λ_1	‖	λ_2		
1,0	μ_1		λ_1‖		λ_2	
2,0		$2\mu_1$	‖			
	—	—	—	—	—	
0,1	μ_2		‖		λ_1‖	λ_2
1,1		μ_2	λ_1‖	μ_1	‖	
				—	—	—
0,2				$2\mu_2$	λ_1‖	

2.6.3 Time-scale separation and preemption rates

The previous subsection established that the CTMC $\{\mathbf{n}(t)\}$ is NCD under the *pop*-1, provided the *hslf* time-scale separation holds. We now show how this decomposition lends itself to an approximate computation of the preemption rates.

Definition 8. For each call class $k = 1, \ldots, K$:

- Let $\pi_k(m, c)$ denote the probability that a single-class $M/M/c/c$ queue with offered load ρ_k is in state m. Thus, $E(\rho_k, c) = \pi_k(c, c)$.
- Let $\phi_k(m, c)$ denote the probability that a queue with k classes labeled $1, \ldots, k$, with preemptive offered loads $\rho_1, \ldots, \rho_k$ and c circuits is in a state with $N_k = m$.
- Let $\psi_k(m, c)$ denote the probability that a queue with $K - k$ classes labeled $k + 1, \ldots, K$, with preemptive offered loads $\rho_{k+1}, \ldots, \rho_K$ and c circuits is in a state with $\bar{N}_k = m$ (recall $\bar{N}_k = n_{k+1} + \cdots n_K$).

Lemma 1. Under the *hslf* time-scale separation assumption, the functions ϕ_k, ψ_k admit the following

recursive approximation for each $k = 1, \ldots, K-1$:

$$\begin{aligned} \phi_k(m,c) &\approx \sum_{l=0}^{m} \phi_{k-1}(l,c)\pi_k(m-l,c-l), \\ \psi_k(m,c) &\approx \sum_{l=0}^{m} \pi_{k+1}(l,c)\psi_{k+1}(m-l,c-l), \end{aligned} \tag{2.45}$$

with the base case being

$$\phi_1(m,c) \approx \pi_1(m,c),\ \psi_{K-1}(m,c) \approx \pi_K(m,c), \tag{2.46}$$

and $\psi_K(m,c) = 0$. The quality of the approximation improves as the degree of time-scale separation increases.

Proof. Consider $\phi_k(m,c)$. Condition on $N_{k-1} = l$ for $l = 0, \ldots, m$. Then, by time-scale separation, class k arrivals behave approximately the same as class k arrivals on a single class queue with $c-l$ circuits available. The probability that $N_k = m$ conditioned on $N_{k-1} = l$ is then the probability that there are $m-l$ class k calls in this single class queue, $\pi_k(m-l, c-l)$. The probability of $N_{k-1} = l$ is then $\phi_{k-1}(l,c)$. Next, consider $\psi_k(m,c)$. Class $k+1$ is unaffected by classes $k+2, \ldots, K$, and thus the probability that $n_{k+1} = l$ is given by $\pi_{k+1}(l,c)$. Conditioned on $n_{k+1} = l$, by time-scale separation, the arrivals of classes $k+2, \ldots, K$ effectively see a queue with $c-l$ circuits available. Thus, the probability that $\bar{N}_k = m$ conditioned on $n_{k+1} = l$ is given by $\psi_{k+1}(m-l, c-l)$. □

The following Theorem gives an approximation for the preemption rates under the *hslf* time-scale separation.

Theorem 11. Assume the *hslf* time-scale separation holds. The rate that arriving calls of class $k < K$ preempt active calls of classes $k+1, \ldots, K$ is

$$P_k \approx \tilde{P}_k = \lambda_k \sum_{m=0}^{c-1} \phi_k(m,c)\psi_k(c-m, c-m). \tag{2.47}$$

The rate that class $i < K$ arrivals preempt active class $k > i$ calls is approximately

$$\begin{aligned} \tilde{P}_{i,k} = \tilde{Q}_{k,i} &= \lambda_i \sum_{m=0}^{c-1} (\phi_{k-1}(m,c)\psi_{k-1}(c-m,c-m) \\ &\quad - \phi_k(m,c)\psi_k(c-m,c-m)) \,. \end{aligned} \tag{2.48}$$

The rate that active class $k > 1$ calls are preempted is

$$\begin{aligned} Q_k \approx \tilde{Q}_k &= \Lambda_{k-1} \sum_{m=0}^{c-1} (\phi_{k-1}(m,c)\psi_{k-1}(c-m,c-m) \\ &\quad - \phi_k(m,c)\psi_k(c-m,c-m)) \,. \end{aligned} \tag{2.49}$$

The rates that arriving calls of class $k = 1$ are blocked (B_1) and admitted (A_1) are exactly

$$B_1 = \lambda_1 \phi_1(c,c) = \lambda_1 \pi_1(c,c) = \lambda_1 E(\rho_1, c), \quad A_1 = \lambda_1 (1 - E(\rho_1, c)). \tag{2.50}$$

The rates that arriving calls of class $k > 1$ are blocked (B_k) and admitted (A_k) are approximately

$$B_k \approx \tilde{B}_k = \lambda_k \phi_k(c,c), \quad A_k \approx \tilde{A}_k = \lambda_k (1 - \phi_k(c,c)). \tag{2.51}$$

The rate that class $k = 1$ calls depart is exactly $D_1 = A_1$; the rate that class $k > 1$ calls depart is

$$D_k \approx \tilde{D}_k = \tilde{A}_k - \tilde{Q}_k. \tag{2.52}$$

The preemption rates (P_k^p, $P_{i,k}^p = Q_{i,k}^p$, Q_k^p or P_k^s, $P_{i,k}^s = Q_{k,i}^s$, Q_k^s), blocking and admission rates (B_k^p, A_k^p or B_k^s, A_k^s), and departure rates (D_k^p or D_k^s) are found by replacing c with c^p or c^s in (2.47 – 2.52).

Proof. Consider P_k. The probability that an arriving class k call finds the system in a state requiring preemption is the probability that $(N_k, \bar{N}_k) = (m, c-m)$ for some $m = 0, \ldots, c-1$. By time-scale separation, $\mathbb{P}(N_k = m) = \phi_k(m,c)$, and $\mathbb{P}(\bar{N}_k = c-m) = \psi_k(c-m, c-m)$. The remaining cases

yield to similar argument. □

Example: $K = 2$**.** Consider the case of $K = 2$ classes under the *hslf* time-scale separation. Using Theorem 11, the rate that class 1 arrivals preempt active class 2 calls is approximately

$$\tilde{P}_1 = \tilde{P}_{1,2} = \tilde{Q}_{2,1} = \tilde{Q}_2 = \lambda_1 \sum_{m=0}^{c-1} \pi_1(m,c)\pi_2(c-m,c-m). \tag{2.53}$$

Moreover, the class 1 blocking rate is $B_1 = \lambda_1\pi_1(c,c) = E(\rho_1,c)$, while the class 2 blocking rate is approximately

$$\tilde{B}_2 = \lambda_2 \sum_{m=0}^{c} \pi_1(m,c)\pi_2(c-m,c-m). \tag{2.54}$$

Finally, the class 1 departure rate is $D_1 = A_1 = \lambda_1 - B_1$, while the class 2 departure rate is approximately $\tilde{D}_2 = \lambda_2 - \tilde{B}_2 - \tilde{Q}_2$.

Example: $K = 3$**.** Consider the case of $K = 3$ classes under the high-slow low-fast time-scale separation. The arrival transitions are illustrated in Figure 2.8. Using Theorem 11 the rate of preemption caused by class 1 arrivals is approximately

$$\tilde{P}_1 = \lambda_1 \sum_{m=0}^{c-1} \sum_{l=0}^{c-m} \pi_1(m,c)\pi_2(l,c-m)\pi_3(c-(m+l),c-(m+l)), \tag{2.55}$$

while the rate of preemption caused by class 2 arrivals is approximately

$$\tilde{P}_2 = \tilde{P}_{23} = \lambda_2 \sum_{m=0}^{c-1} \sum_{l=0}^{m} \pi_1(l,c)\pi_2(m-l,c-l)\pi_3(c-m,c-m). \tag{2.56}$$

The rate of preemptions of class 2 calls by class 1 arrivals is approximately

$$\begin{aligned} \tilde{P}_{12} \quad = \quad & \lambda_1 \sum_{m=0}^{c-1} \Big(\sum_{l=0}^{c-m} \pi_1(m,c)\pi_2(l,c-m) \times \\ & \pi_3(c-(m+l),c-(m+l)) \\ & - \sum_{l=0}^{m} \pi_1(l,c)\pi_2(m-l,c-l)\pi_3(c-m,c-m) \Big), \end{aligned} \tag{2.57}$$

while the rate of preemptions of class 3 calls by class 1 arrivals is approximately

$$\tilde{P}_{13} = \lambda_1 \sum_{m=0}^{c-1} \sum_{l=0}^{m} \pi_1(l,c)\pi_2(m-l,c-l)\pi_3(c-m,c-m). \tag{2.58}$$

The rate at which active class 2 calls are preempted is approximately

$$\begin{aligned} \tilde{Q}_2 &= \lambda_1 \sum_{m=0}^{c-1} \Big(\sum_{l=0}^{c-m} \pi_1(m,c)\pi_2(l,c-m) \times \\ &\quad \pi_3(c-(m+l),c-(m+l)) - \\ &\quad \sum_{l=0}^{m} \pi_1(l,c)\pi_2(m-l,c-l)\pi_3(c-m,c-m) \Big), \end{aligned} \tag{2.59}$$

while the rate at which active class 3 calls are preempted is approximately

$$\tilde{Q}_3 = \Lambda_2 \sum_{m=0}^{c-1} \sum_{l=0}^{m} \pi_1(l,c)\pi_2(m-l,c-l)\pi_3(c-m,c-m). \tag{2.60}$$

Moreover, the class 1 blocking rate is $B_1 = \lambda_1 \pi_1(c,c) = E(\rho_1, c)$, while the class 2 and 3 blocking rates are approximately

$$\begin{aligned} \tilde{B}_2 &= \lambda_2 \sum_{m=0}^{c} \pi_1(m,c)\pi_2(c-m,c-m), \\ \tilde{B}_3 &= \lambda_3 \sum_{m=0}^{c} \sum_{l=0}^{m} \pi_1(l,c)\pi_2(m-l,c-l)\pi_3(c-m,c-m). \end{aligned} \tag{2.61}$$

Finally, the class 1 departure rate is $D_1 = A_1 = \lambda_1 - B_1$, while the class 2 and 3 departure rates are approximately

$$\tilde{D}_2 = \lambda_2 - \tilde{B}_2 - \tilde{Q}_2, \quad \tilde{D}_3 = \lambda_3 - \tilde{B}_3 - \tilde{Q}_3. \tag{2.62}$$

Comparison of approximate and exact results. To give some intuition for the relationship between the actual preemption rates and the approximations in Theorem 11, we compute P_1 for the case of *arbitrary* heterogeneous service rates for the simple case of $K = 2$ and $c \in \{1, 2\}$.

Consider first when the link can support a single call: $c = 1$. Global balance on the states

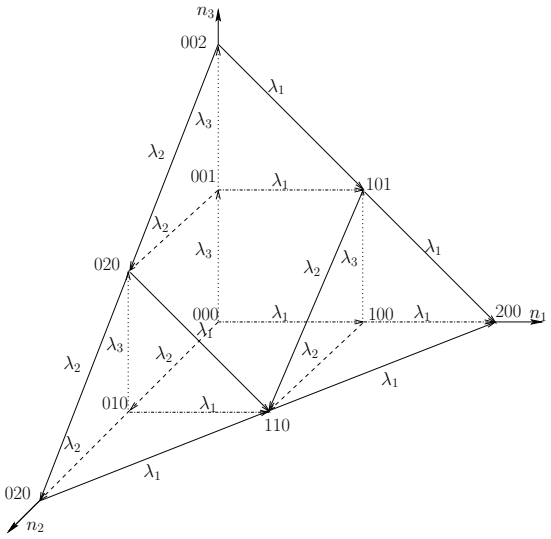


Figure 2.8: State transition diagram for a link with three service classes ($K = 3$) and a capacity of two calls ($c = 2$). Downward transitions (departures) are not shown for clarity.

$\mathcal{S} = \{(0,0),(0,1),(1,0)\}$ yields the invariant distribution, say $\hat{\pi}(0,0), \hat{\pi}(0,1), \hat{\pi}(1,0)$, from which we find

$$P_1 = \lambda_1 \hat{\pi}(0,1) = \lambda_1 \frac{\rho_2}{(\rho_1+1)(\rho_1(\mu_1/\mu_2)+\rho_2+1)}. \tag{2.63}$$

Theorem 11 gives

$$\tilde{P}_1 = \lambda_1 \pi_1(0,1)\pi_2(1,1) = \lambda_1 \frac{1}{1+\rho_1}\frac{\rho_2}{1+\rho_2}. \tag{2.64}$$

The ratio

$$\frac{\tilde{P}_1}{P_1} = 1 + \frac{\lambda_1}{\lambda_2+\mu_2} \tag{2.65}$$

illustrates that the approximation nears 1 as the time-scale separation becomes large. In fact, the ratio illustrates that, at least for the case when $c = 1$, the approximation is valid if either λ_2 is large or μ_2 is large.

Consider next when the link can support two calls: $c = 2$. Solving the global balance equations yields the invariant distribution, say

$$\hat{\pi}(0,0),\ \hat{\pi}(0,1),\ \hat{\pi}(0,2),\ \hat{\pi}(1,0),\ \hat{\pi}(1,1),\ \hat{\pi}(2,0), \tag{2.66}$$

which permits computation of

$$P_1 = \lambda_1 \left(\hat{\pi}(0,2) + \hat{\pi}(1,1)\right) = \lambda_1 \frac{\imath(\rho_1, \rho_2, (\mu_1/\mu_2))}{\jmath(\rho_1, \rho_2, (\mu_1/\mu_2))}, \tag{2.67}$$

where

$$\begin{aligned}
&\imath(\rho_1, \rho_2, (\mu_1/\mu_2)) \\
&\quad = \quad 2\rho_2\Big[\rho_1^2(\rho_1+1)(\mu_1/\mu_2)^2 \\
&\quad + \quad (2\rho_1^2\rho_2 + 3\rho_1^2 + 3\rho_1\rho_2 + 2\rho_1 + \rho_2)(\mu_1/\mu_2) \\
&\quad + \quad (\rho_1\rho_2^2 + \rho_2^2 + 2\rho_1\rho_2 + \rho_2 + 2\rho_1)\Big], \\
&\jmath(\rho_1, \rho_2, (\mu_1/\mu_2)) \\
&\quad = \quad (\rho_1^2 + 2\rho_1 + 2)\Big[\rho_1^3(\mu_1/\mu_2)^3 + 2\rho_1^2(\rho_2+2)(\mu_1/\mu_2)^2 \\
&\quad + \quad (\rho_2^2 + \rho_1\rho_2 + 2\rho_2 + \rho_1 + 2)(\mu_1/\mu_2) \\
&\quad + \quad (\rho_2+1)(\rho_2^2 + 2\rho_2 + 2)\Big].
\end{aligned} \tag{2.68}$$

Theorem 11 gives

$$\begin{aligned}
\tilde{P}_1 &= \lambda_1\Big(\pi_1(0,2)\pi_2(2,2) + \pi_1(1,2)\pi_2(1,1)\Big) \\
&= \lambda_1\left(\frac{1}{1+\rho_1+\rho_1^2/2} \cdot \frac{\rho_2^2/2}{1+\rho_2+\rho_2^2/2}\right. \\
&\quad + \left.\frac{\rho_1}{1+\rho_1+\rho_1^2/2} \cdot \frac{\rho_2}{1+\rho_2}\right).
\end{aligned} \tag{2.69}$$

Simple algebra shows that dropping all terms with $\frac{\mu_1}{\mu_2}$ in P_1 yields $\tilde{P}_1$, showing that the approximation becomes exact in the limit of the time scale separation. It warrants mention that in principle the exact preemption rates can always be computed, for arbitrary K, c, by obtaining the invariant distribution and summing over the appropriate states. This approach is infeasible, however, due to the exponential growth in the size of the state space in both K and c.

2.7 Numerical and simulation results

In this section we present plots of the preemption rates versus the arrival rates. We have written a preemption network simulator in Java; the simulation results support the obtained exact numerical results, as well as illustrate the regimes where the asymptotic approximations are valid. Throughout this section our independent variable is the **arrival rate scaling parameter**, r. In particular, we will grow λ_k linearly in r, *i.e.*, $\lambda_k = a_k r$, for specified $a_1, \ldots, a_K$. For r small the links are underloaded, and for r large the links are overloaded.

2.7.1 Single link: homogeneous service rates

Consider a single IL with $c = 100$, $K = 2$, arrival rates λ_1, λ_2 (to be varied), and $\mu_1 = \mu_2 = 1$ (homogeneous service rates). **Fig. 2.9** contains three plots of preemption probabilities versus r. The preemption probabilities are obtained from the rate expressions by dividing by the appropriate arrival rate: the preemption probability for class 1 is P_1/λ_1, and the preemption probability for class 2 is Q_2/λ_2. In each case the probability is to be understood as a "customer" average, *e.g.*, P_1/λ_1 is the fraction of arriving class 1 calls that cause a preemption. Further, each curve on each plot is actually a superposition of simulation results, exact numerical results (from §2.4), and approximate numerical results (from §2.5). Theorem 8 is used for the numerical approximations for the first three plots, Corollary 2 is used for the numerical approximation for the bottom plot.

The **top** of Fig. 2.9 presents P_1/λ_1 where $\lambda_1 = r$, and λ_2 is varied among $10r$, r, and $0.1r$. In each case the preemption probability is seen to be increasing, reach a maximum very near to $\lambda_1 = \rho_1 = c_1 = 100$, and then be convex decreasing. The initial increase is because increasing λ_1 moves the link from an underloaded regime to an overloaded regime: the number of preemptions increases as the system "fills up". The subsequent decrease is because as λ_1 continues to increase, it is increasingly likely that *all* circuits are occupied by class 1 calls, and thus arriving class 1 calls are blocked, rather than admitted by preempting a class 2 call. P_1/λ_1 is increasing as λ_2 increases from $0.1\lambda_1$ to λ_1 to $10\lambda_1$: a higher λ_2 means there are more class 1 arrivals that preempt class 2 calls.

The **middle** plot in Fig. 2.9 presents Q_2/λ_2 where $\lambda_2 = r$, and λ_1 is varied among $10r$, r, and

$0.1r$. The same shape of initial increasing and subsequent convex decreasing is retained, as is the fact that the maximum preemption probability occurs when $\lambda_1 = \rho_1 = c_1 = 100$. For the three curves, these maxima occur at $\lambda_2 = 10, 100, 1000$ respectively.

The **bottom** plot in Fig. 2.9 presents P_1/λ_1 and Q_2/λ_2 where the three scalings of the arrival rates are $(\lambda_1, \lambda_2) \in \{(r, r), (r, 10r), (10r, r)\}$. The numerical approximation used in this plot is the "both overloaded" expression from Corollary 2. The approximation is seen to be increasingly accurate as r increases, but the inaccuracy is visible for r small. As is expected, the preemption probability curves are increasing in the ratio of class 2 to class 1 calls. It bears mention that although the preemption rates for the two classes P_1, Q_2 are the same for $K = 2$, *i.e.*, $P_1 = Q_2$, the preemption probabilities P_1/λ_1 and Q_2/λ_2 are distinct.

2.7.2 Single link: heterogeneous service rates

Consider a single IL with $c = 100$ servicing $K = 2$ classes with two different settings for $\lambda_1, \lambda_2, \mu_1, \mu_2$:

scaling	λ_1	λ_2	μ_1	μ_2	ρ_1	ρ_2
hfls	$10r$	r	10	1	r	r
hslf	$0.1r$	$10r$	0.1	10	r	r

The first scaling corresponds to a *hfls* time-scale separation, and the second to a *hslf* time-scale separation. Note that the offered loads for the two classes are equal for both scalings, *i.e.*, $\rho_1 = \rho_2$. The fact that $\mu_1 \neq \mu_2$ means we have heterogeneous service rates. **Fig. 2.10** presents numerical and simulation results of the preemption probability P_1/λ_1 and the blocking probabilities B_1/λ_1 and B_2/λ_2 versus r. The **top** figure demonstrates the inaccuracy of the NCD approximations in the *hfls* scaling, while the **bottom** figure shows the NCD approximation to be accurate in the *hslf* scaling.

2.7.3 Two parallel links: homogeneous service rates

The two parallel links have $c^p = c^b = 100$ servicing $K = 2$ service classes with $\lambda_1 = \lambda_2$ (to be varied), and $\mu_1 = \mu_2 = 1$ (homogeneous service rates). **Fig. 2.11** presents several of the rates from

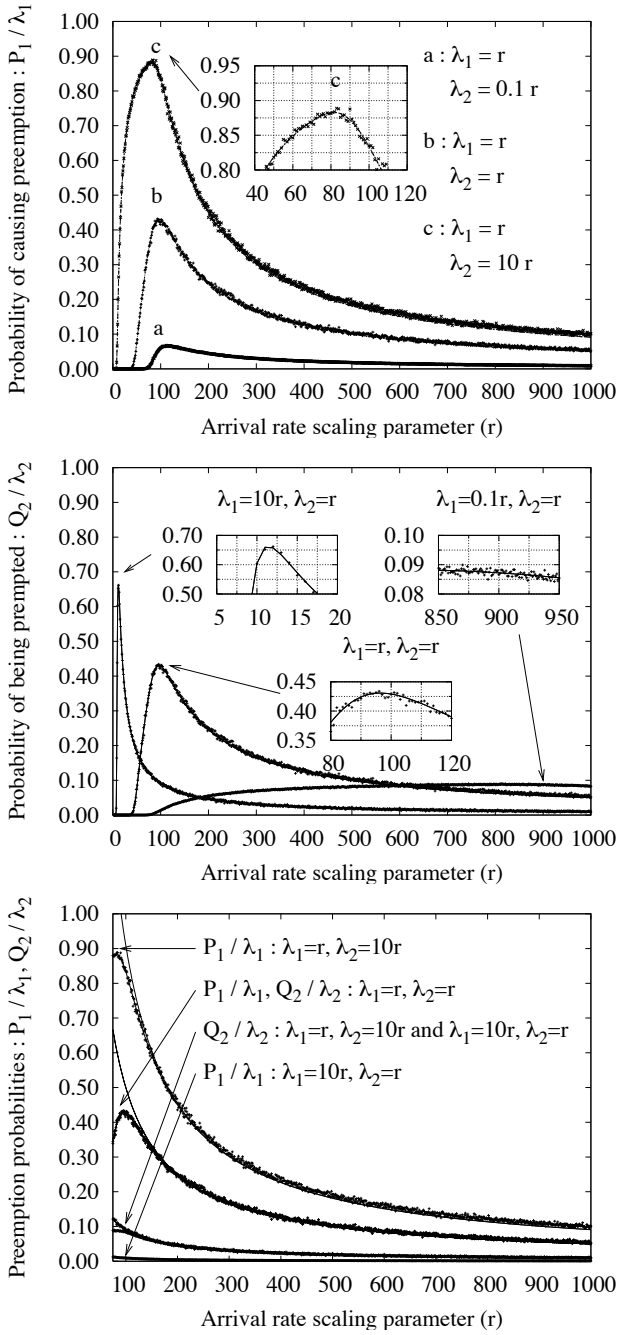


Figure 2.9: Single link with $K = 2$ classes and homogeneous service rates. Preemption probabilities versus the arrival rate scaling parameter r. **Top:** P_1/λ_1 versus r; **Middle:** Q_2/λ_2 versus r; **Bottom:** P_1/λ_1 and Q_2/λ_2 versus r.

Fig. 2.2 versus the common arrival rate $\lambda_1 = \lambda_2$.

The **top** plot shows the class 2 departure rates from the PL, D_2^p, and from the system, D_2^s. The numerical results are taken from §2.4.5 (Theorem 6) and are seen to match exactly with the

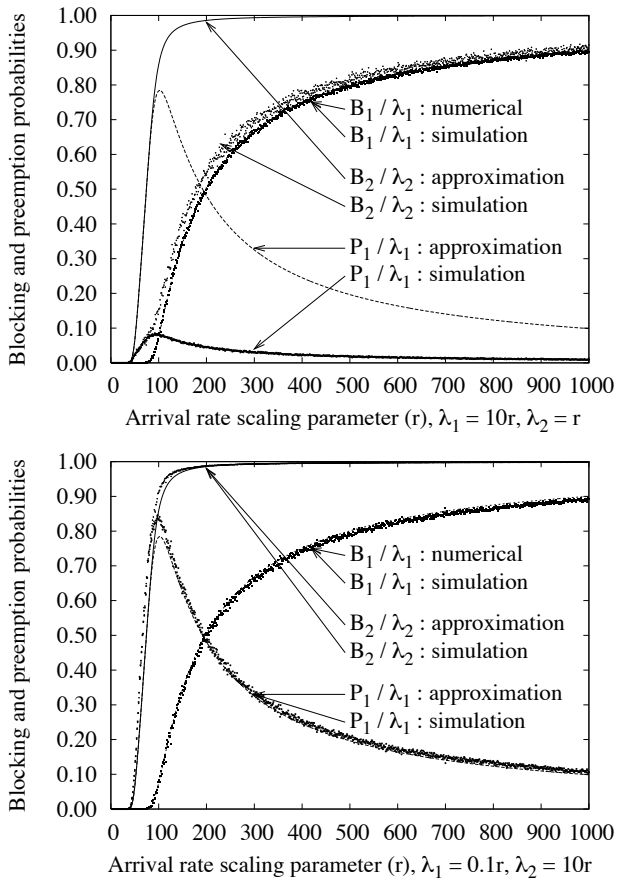


Figure 2.10: Single link with $K = 2$ classes and heterogeneous service rates. The preemption probability P_1/λ_1 and blocking probabilities B_1/λ_1 and B_2/λ_2 versus the arrival rate parameter r. **Top:** *hfsl* time scale separation, $\lambda_1 10\lambda_2$, $\mu_1 = 10\mu_2$. **Bottom:** *hslf* time-scale separation, $\lambda_2 = 100\lambda_1$, $\mu_2 = 100\mu_1$.

simulation results. The departure rate of class 2 calls from the PL is seen to increase until $\lambda_1 + \lambda_2 = \rho_1 + \rho_2 = c^p = 100$, and then decrease; this increase is attributable to the increasing arrival rate of class 2 calls to an underloaded system. Above $\lambda_1 + \lambda_2 = 100$, the PL is typically filled up, and thus class 2 calls are likely to be either blocked from admission on the PL, or preempted to the secondary link; this explains the decreasing departure rate from the PL for $\lambda_1 + \lambda_2 > 100$. The system departure rate is seen to increase steadily until $\lambda_1 + \lambda_2 = \rho_1 + \rho_2 = c^s = 200$. Again, this increase is attributable to the increasing rate of admissions of class 2 calls to an underloaded system. Above $\lambda_1 + \lambda_2 = 200$, the system is typically filled up, and thus class 2 calls are likely to be either blocked from admission into the system or preempted from the system; this explains the decreasing

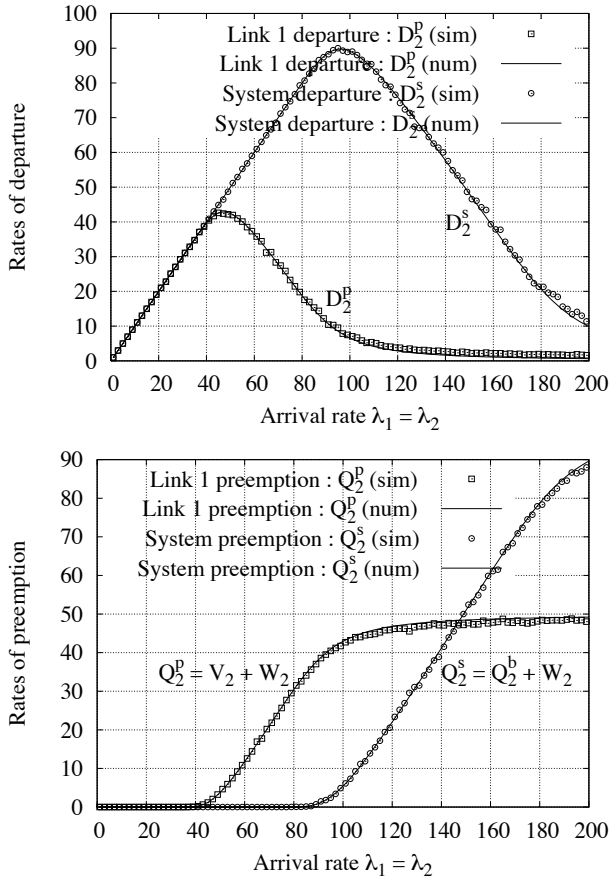


Figure 2.11: Two parallel links with $K = 2$ classes and homogeneous service rates. All plots are versus the common arrival rate $\lambda_1 = \lambda_2$ with $c^p = c^b = 100$. **Top:** departure rates of class 2 calls from the PL (D_2^p) and from the system (D_2^s). **Bottom:** rate of preemption from the PL (both transfers and drops) and the rate of preemption from the system (from both the primary and BL).

departure rate from the system for $\lambda_1 + \lambda_2 > 200$.

The **bottom** plot shows the rate at which class 2 calls are preempted from the PL, Q_2^p, and from the system, Q_2^s. The numerical results are taken from §2.4.5 (Theorem 5) and are seen to match exactly with the simulation results. The preemption rate from the PL, Q_2^p, is seen to be near zero for $\lambda_1 + \lambda_2 = \rho_1 + \rho_2 < c_1 = 100$; this is because the PL is underloaded and there is no need for preemption. From $\lambda_1 + \lambda_2 = 100$ to $\lambda_1 + \lambda_2 = c^s = 200$ the PL preemption rate steadily increases. This is attributable to the fact that the PL is overloaded, and it is increasingly likely that any class 2 call admitted to the PL will be preempted. Above $\lambda_1 + \lambda_2 = 200$, the rate levels out, but the rate of preemption from the system rises rapidly. The rate of preemption from the system is near zero

for $\lambda_1 + \lambda_2 < c^s = 200$ since the system is underloaded.

2.8 Conclusions and future work

We have studied the performance of a two parallel link network with transfers servicing K classes with preemptive priority. The performance metrics are the rate that arriving calls preempt lower priority active calls, and the rate that active calls are preempted by higher priority arriving calls. For homogeneous service rates we obtained exact expressions by lumping the Markov chain, and using the fact that the PL and the system as a whole can be treated as ILs. The exact results are in terms of Erlang-B blocking probabilities. These expressions simplify in an asymptotic many small users regime, appropriate for large arrival rates and large capacities. For heterogeneous service rates with a time scale separation, we provided an efficiently computable approximation for preemption rates using NCD methods for Markov chains.

There are a number of open directions for extending this work, including relaxation of the assumed Poisson arrivals and exponential service times, and extension to $L > 2$ parallel links. Using the same system view approach, one can extend the results to a network with L parallel links. However, again only coupled metrics (system metrics) are computable, given that the same restriction applies to finding the individual preemption/blocking probabilities within the individual parallel links.

Chapter 3: Preemption and admission control of a two-class loss link

Although admission control policies for reward maximization in multi-class loss networks have been well studied, much less attention has been given to preemption control policies, where active low priority calls may be preempted upon high priority arrivals. We compare admission and preemption control of a two class loss link under a reward model incorporating per class arrival and departure rewards, preemption costs, and per class holding reward rates. Main results include *i*) if preemption is always done when the link is full then additional preemptions from non full states decrease reward, and *ii*) a sufficient condition for the superiority of optimal preemption without admission control over optimal threshold-based admission control without preemption control. Results are established by policy improvement theorems from stochastic dynamic programming. We provide numerical results consonant with our analyses that illustrate parameter regimes with various sensitivities of performance to control: *i*) optimal joint preemption and admission control outperforms admission control and preemption control alone, *ii*) either admission or preemption control achieves same performance as optimal joint control, *iii*) no control performs significantly worse than admission or preemption control alone, and *iv*) no control performs same as either control alone but worse than optimal joint control.

3.1 Introduction

A loss network is a collection of links (each capable of multiplexing a finite number of concurrent calls) servicing a set of routes (each route consisting of a set of links), where a control mechanism determines whether or not to admit each arriving call on each route [1]. There is no queueing in loss networks – call requests are either admitted and begin service immediately or they are blocked. Multi-class loss networks service multiple classes of calls, where classes often indicate call priority, and call priority often reflects the ordering of reward paid to the network for each admitted call. In the general case arrival rate, service rate, and call rate/size (the number of circuits on each link of

the route consumed by a call of that class) are class specific.

In this paper we consider the simplest non-trivial case of a multi-class loss network: a single loss link servicing two classes of calls, where all calls have a common rate (size), i.e., each call occupies a single circuit. We further assume arrivals for each class of calls form independent Poisson processes with class-dependent rates, and service times for all calls are independent and exponentially distributed with class-dependent means. It follows from these assumptions that the state vector is the number of active calls of each class, and the state evolution is that of a continuous time Markov chain on a finite state space. Throughout the paper we adopt the convention that high (low) priority calls are denoted as class 1 (2) respectively.

We consider two forms of control: admission control and preemption control. An admission control policy specifies whether or not to admit an arriving call of a given class on a given route as a function of the number of active calls of each class on each link. In our setting of two call classes on a single link this translates to a decision of whether or not admit class 2 calls in each state where the link is not full. A preemption control policy, on the other hand, specifies whether or not to admit an arriving call on a given route by preempting one or more lower priority calls occupying circuits on links comprising the arriving call's route. Preemption is in fact an integral control mechanism within the proposed Differentiated Services Internet architecture (DiffServ) [39]. Fig. 3.1 illustrates the design space of admission control and preemption control policies for the setting of two call classes on a single link. We use the framework of Markov decision processes (MDP) to demonstrate expected per stage reward orderings between two competing control policies. The literature on preemption is smaller than on admission control, and in our opinion there has been insufficient attention given to *i*) the optimal preemption policy and its performance, and *ii*) the relative and joint performance of optimal admission and optimal preemption control policies. Although the results in this paper fall short of answering these questions, our motivation in writing this paper is to initiate their investigation.

The design of optimal admission control policies for multi-class loss networks is difficult due to the following tension: admitting a low priority call guarantees a certain reward – more so than would

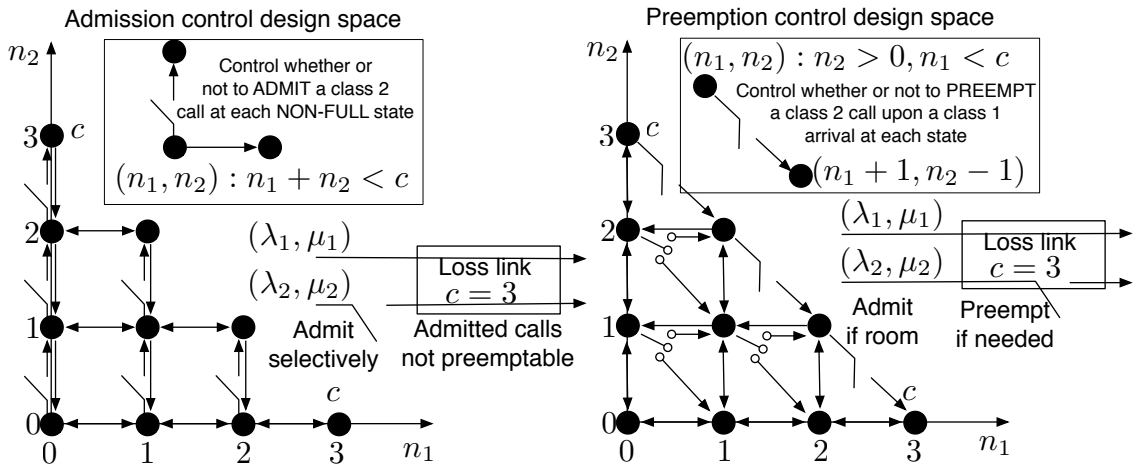


Figure 3.1: The state transition diagrams for the continuous time Markov chains describing the evolution of the number of active calls of each class (n_1, n_2) on a loss link with capacity $c = 3$. **Left:** the admission control policy specifies whether to admit class 2 calls in non full states. **Right:** the preemption control policy specifies whether to admit class 1 by preempting class 2 calls.

be obtained were the circuits used to support the call to remain idle – but the admission occupies circuits that might preclude even greater reward by admission of a potential future arrival of a high priority call. Even in our restricted setting of two call classes on a single link the optimal admission control policy is unknown. In fact, the set of all possible admission control policies is often too large and unwieldy to yield to analysis. The admission control problem is an instance of a stochastic knapsack problem, where the knapsack size is represented by the capacities of the network links, the state is the number of calls of each class, and the arrival streams of items of various types are the arriving multi-class calls [21]. Instead of seeking to optimize over all admission control policies, much of the admission control literature focuses on optimizing over the set of coordinate convex admission policies [40]. When restricted to this set, it is known that the optimal policy for two call classes is of threshold type [41]. In particular, in our setting the set of coordinate convex admission control policies is the set of policies where a class 2 call is admitted as long as the number of such calls lies below a threshold (Prop. 1). It is then relatively straightforward to numerically identify the optimal threshold (say, by bisection search). The design of optimal preemption control policies for multi-class loss networks is difficult due to the fact that admitting a high priority call via preemption

guarantees a higher instantaneous reward rate than was achieved by the preempted low priority call, but this gain is offset by the cost of preempting the low priority call.

In this work we adopt a flexible yet tractable reward framework that incorporates *i*) instantaneous per class reward amounts for admitted, departing, and preempted calls, and *ii*) class and state dependent reward rates (paid per unit time) for active calls. Our primary findings are summarized as follows:

1. The use of preemption at non full states is shown to be strictly sub-optimal for all "reasonable" reward models provided preemption is always performed at all full states (Prop. 2 in §3.4.1). The stronger statement that policy performance always improves by removing preemptions at non full states is shown to be false by a simple counter-example. We conjecture that the optimal preemption policy *is* achieved without preempting at non full states (Conjectures 1 and 2), but we do not have a proof of this claim.

2. Given a choice between admission control and preemption control, we provide a sufficient condition (Prop. 3 in §3.4.2) for the superiority of *i*) a policy that preempts at all full states (and never preempts at non full states) but does not use admission control over *ii*) a policy that does not employ preemption but chooses an optimal admission control strategy over the class of class 2 threshold policies (the set of coordinate convex policies).

3. For a fixed admission and control policy, the space of all possible reward models may be partitioned based on the asymptotic expected time-average reward rate. We characterize these classes (Prop. 4 in §3.4.3) and give a condition for two reward models to be reward equivalent (Prop. 5 in §3.4.3).

We provide extensive numerical results consonant with our analyses. Although our analytical work in this paper does not address the combined use of admission control and preemption control, our numerical results illustrate intriguing phenomena in various parameter regimes including *i*) optimal joint admission control and preemption control outperforms either one alone, and *ii*) the use of either one alone performs the same as no control (Fig. 3.13). These numerical results demonstrate

the interesting design space of joint admission control and preemption control.

This paper is organized as follows. We give a brief summary of related work in §3.2 and describe our system model in §3.3. The three primary findings discussed above are presented in the three subsections of §3.4. Numerical results are discussed in §3.5 with results for a link capable of holding a single call in §3.5.1, and for larger capacity links in §3.5.2. §3.6 offers a brief conclusion. Proofs of several technical lemmas are placed in appendices following the references.

3.2 Related work

We briefly discuss related work on admission control and preemption control. The discussion is restricted to loss networks as the literature for more general queueing networks is too large to be covered here and is of limited relevance.

Admission control. The optimization and performance analysis of admission control mechanisms for multi-class loss networks are well understood, see, e.g., the survey paper by Kelly [1] and the book by Ross [21]. Ross and Tsang modeled the control of multi-class loss networks as a stochastic knapsack problem [41] and characterized the optimal coordinate convex admission control policy. The notion of coordinate convex sets was introduced by Aein [42;43]. Computation of blocking probabilities in multi-class loss networks with coordinate convex admission control policies was addressed by Kaufman [40] who utilized the product form of the steady state occupancy distribution. Loss networks served by finite populations (Engset models) were investigated by Foschini and Gopinath [44]. More recently work in admission control has addressed *i*) statistical quality of service (e.g., [45] and many others), *ii*) measurement based decisions (e.g., [46;47] and many others), and *iii*) micro–cellular policies for cellular wireless networks (e.g., [48] and many others) where there is the additional concern of handoff (e.g., [49] and many others).

Preemption control. Garay and Gopal's seminal 1992 paper showed that optimal preemption decisions (minimizing either the number of preempted connections or the amount of preempted bandwidth) in circuit–switched networks is an NP-complete problem [12], and suggested several (centralized) approximations with associated guarantees relative to the optimal solution. Decentralized preemption algorithms were presented in [13]. Many preemption policies have been described in the

context of a Differentiated Services (DiffServ) aware multi–protocol label switching (MPLS) scenario, e.g.,[14–20;50], but few of these works offer analytical results. Instead these papers present preemption policies that *i*) are flexible and adaptive[14;50], *ii*) jointly allocate bandwidth and make preemption decisions[16], *iii*) reduce algorithm computation time[17], *iv*) integrate preemption with routing[18;19], *v*) employ fuzzy logic and genetic algorithms[20]. The earliest performance analyses of a preemption policy in a loss context are by Helly[28] and Burke[29], both from 1962. After that, the literature appears to be silent until 1980 when Calabrese *et al.*[30] published an analysis of a voice network of multiple parallel links with preemption. More recently, our prior work[2] analyzed the performance of a preemption policy for two parallel loss links serving multiple call classes.

3.3 Model

As shown in Fig. 3.2, we study two classes of call arrivals serviced by a single loss link, where the link employs selective admission control for class 2 arrivals, and may preempt active class 2 calls upon class 1 arrivals. Class-specific quantities are indicated by a subscript $k \in \{1, 2\}$.

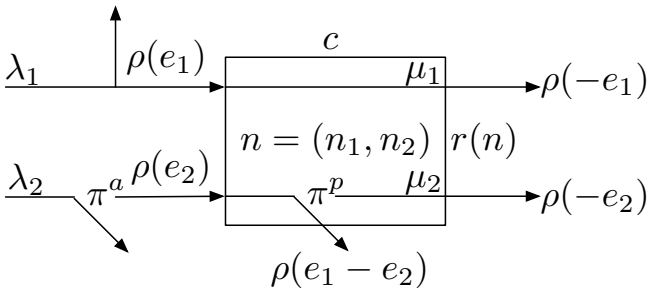


Figure 3.2: A loss link of capacity c circuits servicing two classes of traffic. The per class arrival processes are Poisson with rates λ_1, λ_2, and the per class service times are independent and exponentially distributed with rates μ_1, μ_2. The state of the system is $n = (n_1, n_2)$ with n_i the number of active class i calls. The admission control policy π^a selectively admits arriving class 2 calls while the preemption control policy π^p selectively preempts active class 2 calls upon class 1 arrivals. The reward model consists of a state-dependent reward per unit time $r(n)$ and an instantaneous reward upon state transitions: $\rho(e_1), \rho(e_2)$ for class 1 (2) admissions, $\rho(-e_1), \rho(-e_2)$ for class 1 (2) departures, and $\rho(e_1 - e_2)$ for a preemption.

3.3.1 Arrival rates, service rates, link capacity, and system state

Assumption 1. *All calls of both classes request a common bandwidth/rate, i.e., all calls are of common "size", which we define as a unit of capacity.*

The link capacity $c \in \mathbb{N}$ is the maximum number of calls it can hold.

Assumption 2. *Each class $k \in \{1, 2\}$ generates an independent Poisson arrival process with rate λ_k and the corresponding service times are independent and identically distributed (iid) exponential random variables with mean μ_k^{-1}.*

Due to the memoryless property corresponding to the assumed arrivals and service time distributions, it follows that the relevant system state is the number of calls of each class active on the link, i.e., $n \equiv (n_1, n_2)$, with n_k the number of active calls of class $k \in \{1, 2\}$. The state space is denoted $\mathcal{N} \equiv \{n \in \mathbb{Z}_+^2 : n_1 + n_2 \leq c\}$. Note that the choices of admission and preemption control policies, described below, may limit the set of accessible states in the state space to a strict subset of $\mathcal{N}$. We say the link is *full* for any state n with $n_1 + n_2 = c$. Let $N \equiv |\mathcal{N}|$ be the number of states.

3.3.2 Admission and preemption control policies and spaces

We consider two control mechanisms: *i*) admission control and *ii*) preemption control. Admission control determines whether to admit arriving class 2 calls, and preemption control determines whether to preempt a class 2 call upon a class 1 admission. The following assumption reflects the assumed priority of class 1 over class 2 calls.

Assumption 3. *Class* 1 *calls are always admitted when the link is not full. Class* 1 *calls are never preempted. Class* 2 *calls are never admitted when the link is full.*

These assumptions leave open the following possibilities. First, class 1 calls may be admitted by preempting a class 2 call even if the link is not full, provided one or more class 2 calls is active. Second, class 2 arrivals requesting admission may be blocked even if the link is not full. The focus of this paper is on the proper use of these controls. We restrict our attention to deterministic (as opposed to randomized) and stationary (the chosen action is not time dependent) decisions. Although we allow an admission or preemption control decision to depend upon the system state n, we do not allow the decision to depend upon any other variables, including, e.g., *i*) the duration of the arriving call, *ii*) the time of arrival, *iii*) recent call arrivals or completions or preemptions. Note that admission (preemption) control policies dictate the course of action upon class 2 (1) arrivals,

respectively.

Definition 1. *An* admission control policy *is a function* $\pi^a : \mathcal{N} \to \{0, 1\}$*, where*

$$\pi^a(n) \equiv \begin{cases} 1, & \textit{admit} \\ 0, & \textit{block} \end{cases} \quad \textit{arriving class 2 call when the link state is } n, \tag{3.1}$$

for $n \in \mathcal{N}$*. Feasibility requires* $\pi^a(n) = 0$ *for all* n *such that* $n_1 + n_2 = c$*.*

Definition 2. *A* preemption control policy *is a function* $\pi^p : \mathcal{N} \to \{0, 1\}$*, where*

$$\pi^p(n) \equiv \begin{cases} 1, & \textit{preempt} \\ 0, & \textit{don't preempt} \end{cases} \quad \textit{class 2 call when class 1 arrival sees state } n, \tag{3.2}$$

for $n \in \mathcal{N}$*. Feasibility requires* $\pi^p(n) = 0$ *for all* n *such that* $n_2 = 0$*.*

We call $\pi \equiv (\pi(n), n \in \mathcal{N})$ a control policy, and the specific value $\pi(n)$ a control decision. We consider various ways to restrict the feasible controls in each state through the use of *decision spaces*.

Definition 3. *A* decision space *is a function* $\mathcal{S} : \mathcal{N} \to \{\{0\}, \{1\}, \{0, 1\}\}$*, where*

$$\mathcal{S}(n) \equiv \begin{cases} \{0\} & \Rightarrow \pi(n) = 0 \\ \{1\} & \Rightarrow \pi(n) = 1 \\ \{0, 1\} & \Rightarrow \pi(n) \in \{0, 1\} \end{cases} \quad , n \in \mathcal{N}. \tag{3.3}$$

To clarify, the statement $\mathcal{S}(n) = \{0, 1\} \Rightarrow \pi(n) \in \{0, 1\}$ means the decision space $\mathcal{S}$ imposes no restriction on the choice of decision for any control policy within that space at state n. A control policy π is *permissible* under decision space $\mathcal{S}$ if $\pi(n) \in \mathcal{S}(n)$ for each $n \in \mathcal{N}$. A decision space $\mathcal{S}$ has an associated set of policies, defined below.

Definition 4. *A* control policy space Π *is a collection of control policies* π*. A control policy space associated with a decision space* $\mathcal{S}$ *is the set of control policies* π *permissible under* $\mathcal{S}$*, i.e.,*

$$\Pi_{\mathcal{S}} \equiv \{\pi : \pi(n) \in \mathcal{S}(n),\ n \in \mathcal{N}\}. \tag{3.4}$$

The above discussion of a general decision space $\mathcal{S}$ and control space Π applies to both admission control $(\pi^a, \mathcal{S}^a, \Pi^a)$ and preemption control $(\pi^p, \mathcal{S}^p, \Pi^p)$. We now give several examples of feasible admission and preemption control policies and spaces that will be of interest in this paper.

Admission control policies and spaces

- *Complete sharing (cs) policy:* admit a class 2 call whenever there is room, i.e., π^a_{cs} has elements $\pi^a_{\text{cs}}(n) \equiv \mathbf{1}_{n_1+n_2<c}$, where $\mathbf{1}_A$ is the indicator function for event A.

- *Threshold (th) admission control policy:* admit a class 2 call if $n_1 + n_2 < c$ and $n_2 < \tau$, for $\tau \in \{0, \ldots, c\}$. That is, $\pi^a_\tau(n) \equiv \mathbf{1}_{n_1+n_2<c, n_2<\tau}$. Complete sharing is a special case of this policy with $\tau = c$, while $\tau = 0$ corresponds to never admitting a class 2 call. Let $\Pi^a_{\text{th}} \equiv \{\pi^a_0, \ldots, \pi^a_c\}$ denote the collection of possible threshold policies.

- *Maximum flexibility (mf) admission control decision space:* imposes no restriction on the choice of admission control policy at any state:

$$\mathcal{S}^a_{\text{mf}}(n) \equiv \begin{cases} \{0,1\}, & n_1 + n_2 < c \\ \{0\}, & \text{else} \end{cases}, \; n \in \mathcal{N}. \tag{3.5}$$

  The corresponding control policy space is $\Pi^a_{\text{mf}} \equiv \Pi_{\mathcal{S}^a_{\text{mf}}}$.

- *Coordinate convex (cc) control policy space.* A coordinate convex admission control policy has an associated set of achievable states, say $\Omega \subseteq \mathcal{N}$, satisfying *i)* $n \in \Omega$ with $n_k > 0$ implies $n - e_k \in \Omega$ for $k \in \{1, 2\}$, and *ii)* accept a class k call in $n \in \mathcal{N}$ iff $n + e_k \in \Omega$ [40;41;43]. Let Π^a_{cc} be the control policy space of coordinate convex control policies.

Proposition 1. *Under Ass. 1 and 3, the coordinate convex admission control policy space equals the threshold admission control policy space:* $\Pi^a_{\text{cc}} = \Pi^a_{\text{th}}$.

Proof. It is simple to verify that a threshold policy is coordinate convex. It remains to show that a coordinate convex policy is a threshold policy, or, equivalently, a non-threshold policy is not coordinate convex. Let Ω be the set of achievable states of a coordinate convex policy – we will show

that admissions under a non-threshold policy violate the rules for Ω. A non-threshold policy must have two distinct states in one of the following two scenarios. If no two such states exist then the policy is of threshold type. See Fig. 3.3.

- Consider states n, n' be with $n_2 < n'_2$, $n_1 + n_2 < c$, and $n'_1 + n'_2 < c$ such that $\pi^a(n) = 0$ and $\pi^a(n') = 1$. Suppose $n_1 \leq n'_1$ (Fig. 3.3 left). Observe that $i)$ $n' \in \Omega$ and $ii)$ $n + e_2 \notin \Omega$ (due to $\pi^a(n) = 0$). But such a set is not coordinate convex by repeated application of the requirement $n' \in \Omega$ with $n'_k > 0$ implies $n' - e_k \in \Omega$. Suppose instead $n_1 > n'_1$ (Fig. 3.3 middle). Observe that $i)$ $\pi^a(n) = 0$ so that $n + e_2 = (n_1, n_2 + 1) \notin \Omega$. However, by repeating application of the requirement $n' \in \Omega$ with $n'_2 > 0$ implies $n' - e_2 \in \Omega$, we obtain $(n'_1, n_2 + 1) \in \Omega$. Notice $n'_1 < n_1,\ n'_1 + n_2 + 1 < n_1 + n_2 + 1 \leq c$, and by assumption 3, we admit class 1 calls from state $(n'_1, n_2 + 1)$ till occupancy sum equals c, which means $n + e_2 \in \Omega$, which is a contradiction.

- Consider states n, n' with $n_2 = n'_2$, $n_1 < n'_1$, and $n'_1 + n'_2 < c$ such that $\pi^a(n) + \pi^a(n') = 1$ (Fig. 3.3 right). If $\pi^a(n) = 0, \pi^a(n') = 1$, then $n' + e_2 \in \Omega$ and $n + e_2 \notin \Omega$. Repeating application of the requirement $n' + e_2 \in \Omega$ with $n'_1 > 0$ implies $n' + e_2 - e_1 \in \Omega$, we obtain $n + e_2 \in \Omega$, which is a contradiction. If $\pi^a(n) = 1, \pi^a(n') = 0$, then $n + e_2 \in \Omega$ and $n' + e_2 \notin \Omega$. However, Assumption 3 implies $n' + e_2 \in \Omega$ due to $n + e_2 \in \Omega,\ n_1 < n'_1$, which is a contradiction.

□

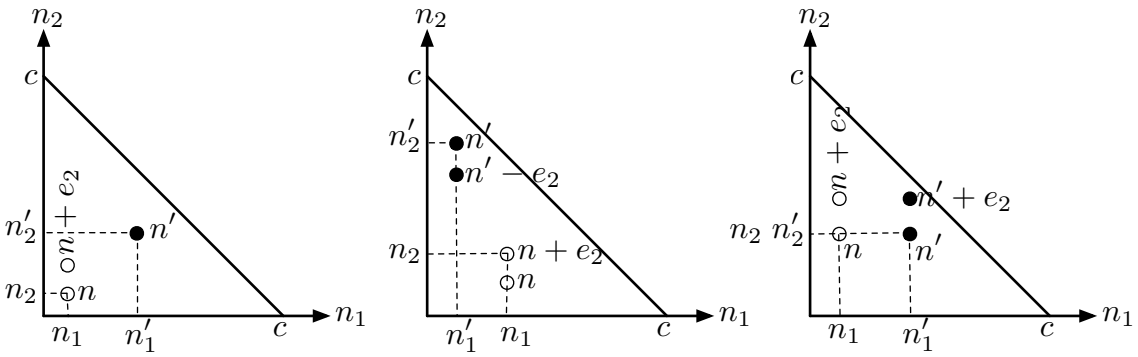


Figure 3.3: Three cases of non-threshold policies discussed in the proof of Prop. 1.

We have the following relationships among the admission control policies and spaces:

$$\pi^a_{\rm cs} \in \Pi^a_{\rm th} = \Pi^a_{\rm cc} \subseteq \Pi^a_{\rm mf}. \tag{3.6}$$

Thus for any performance objective, say g, that depends upon the admission control policy π^a we have:

$$g_{\pi^a_{\rm cs}} \leq \max_{\pi \in \Pi^a_{\rm th}} g_\pi = \max_{\pi \in \Pi^a_{\rm cc}} g_\pi \leq \max_{\pi \in \Pi^a_{\rm mf}} g_\pi. \tag{3.7}$$

The restriction to coordinate convex policies (equivalently, here, threshold policies) may preclude achieving the overall optimal reward rate. For example, it would appear quite natural to consider a class of "sum rate threshold policies" (none of which is coordinate convex) where $\pi^a(n) = 1$ for $n_1 + n_2 < \tau \leq c$ for some $\tau \in [c]$. It is worth noting, however, that restriction to coordinate convex policies is common in the loss network admission control literature [40;41;43]. See Remark 1 later in this section.

Preemption control policies and spaces

- *Never preempt (np) control policy:* $\pi^p_{\rm np}(n) = 0$ for all n.

- *Always when full (awf) decision space:* $\mathcal{S}^p_{\rm awf}$ requires preemption when the link is full: $\mathcal{S}^p_{\rm awf}(n) = \{1\}$ for $n : n_1 + n_2 = c$ and $n_2 > 0$. Optionally preempt when the link is not full: $\mathcal{S}^p_{\rm awf}(n) = \{0, 1\}$ for $n : n_1 + n_2 < c$ and $n_2 > 0$. Denote the control policy space by $\Pi^p_{\rm awf} \equiv \Pi^p_{\mathcal{S}^p_{\rm awf}}$. Let $\pi^p_{\rm awf}$ denote the control policy that always preempts when full but never when not full, note $\pi^p_{\rm awf} \in \Pi^p_{\rm awf}$.

- *Only if full (oif) decision space:* only allow preemption when the link is full:

$$\mathcal{S}^p_{\rm oif}(n) \equiv \begin{cases} \{0,1\}, & n_2 > 0, n_1 + n_2 = c \\ \{0\}, & \text{else} \end{cases} \tag{3.8}$$

  The corresponding control policy space is $\Pi^p_{\rm oif} \equiv \Pi_{\mathcal{S}^p_{\rm oif}}$. Note $\pi^p_{\rm awf} \in \Pi^p_{\rm oif}$.

- *Maximum flexibility (mf) preemption decision space:* imposes no restriction on the choice of

preemption control policy at any state:

$$\mathcal{S}^p_{\rm mf}(n) \equiv \begin{cases} \{0,1\}, & n_2 > 0 \\ \{0\}, & \text{else} \end{cases} \tag{3.9}$$

The corresponding control policy space is $\Pi^p_{\rm mf} \equiv \Pi_{\mathcal{S}^p_{\rm mf}}$.

The following relationships hold among preemption control policies and spaces:

$$\begin{array}{rcl} \{\pi^p_{\rm np}, \pi^p_{\rm awf}\} & \in & \Pi^p_{\rm oif} \\ \pi^p_{\rm awf} & \in & \Pi^p_{\rm awf} \end{array} \Bigg\} \subseteq \Pi^p_{\rm mf}, \tag{3.10}$$

For any performance objective, say g, that depends upon the preemption control policy π^p we have:

$$\begin{array}{rcl} \{g_{\pi^p_{\rm np}}, g_{\pi^p_{\rm awf}}\} & \leq & \max_{\pi \in \Pi^p_{\rm oif}} g_\pi \\ g_{\pi^p_{\rm awf}} & \leq & \max_{\pi \in \Pi^p_{\rm awf}} g_\pi \end{array} \Bigg\} \leq \max_{\pi \in \Pi^p_{\rm mf}} g_\pi. \tag{3.11}$$

where the stacking of the $\pi^p_{\rm np}, \pi^p_{\rm awf}$ policies indicates no ordering exists between them. From here on, the notation $\pi \equiv (\pi^a, \pi^p)$ denotes an admission control and preemption control policy pair.

3.3.3 Reward models

We consider a general reward model, where rewards may be both positive (revenue) and negative (costs). The system accrues reward in two distinct ways: *i*) the function $r : \mathcal{N} \to \mathbb{R}$ gives the rate of reward accrued per unit time $r(n)$ by the system in each state n, and *ii*) the function $R : \mathcal{N} \times \mathcal{N} \to \mathbb{R}$ gives the amount of reward earned $R(n, n')$ when the system transitions from state n to state n'. We often consider R as an $N \times N$ matrix with entries

$$R[n, n'] \equiv \begin{cases} R(n, n'), & n \neq n' \\ 0, & n = n' \end{cases}. \tag{3.12}$$

We emphasize that r is reward rate and R is a reward amount.

There are at most five possible state transitions $n \to n'$ for $n, n' \in \mathcal{N}$: two possible arrival

transitions, two possible departure transitions, and a combined arrival/departure for preemption. Let $e_1 \equiv (1,0)$ and $e_2 \equiv (0,1)$ denote the two unit vectors.

$$\begin{array}{rcll} n' & = & n + e_1 & \text{class 1 admission without preemption} \\ n' & = & n + e_1 - e_2 & \text{class 1 admission with preemption of a class 2 call} \\ n & = & n + e_2 & \text{class 2 admission} \\ n' & = & n - e_k & \text{class } k \text{ departure, } k \in \{1,2\} \end{array} \tag{3.13}$$

Our reward model allows for transition rewards to depend upon the type of transition but not the state of the transition.

Definition 5. *A reward model (r, ρ) consists of a state–dependent reward rate function $r : \mathcal{N} \to \mathbb{R}$ and a five tuple for transition rewards*

$$\rho \equiv (\rho(e_1), \rho(e_2), \rho(-e_1), \rho(-e_2), \rho(e_1 - e_2)) \in \mathbb{R}_+^4 \times \mathbb{R} \tag{3.14}$$

where $\rho(u) \equiv R(n, n+u)$ is the transition reward for a transition of "type" u. Note the arrival and departure reward $\rho(\pm e_k)$ for $k \in \{1,2\}$ are non-negative while the preemption reward $\rho(e_1 - e_2)$ is unrestricted in sign.

Assumption 4. *The state-dependent reward rate function is nondecreasing with respect to the partial order on $\mathcal{N}$: if $n \leq n'$ then $r(n) \leq r(n')$, where $n \leq n'$ means $n_k \leq n'_k$ for $k \in \{1,2\}$.*

We will employ a linear reward rate function in our numerical results in §3.5, $r(n) = r_1 n_1 + r_2 n_2$ for $r_1 \geq r_2 \geq 0$, which may be interpreted as each active class k call pays at rate r_k per unit time. Transition reward examples include:

ρ	$\rho(e_1)$	$\rho(e_2)$	$\rho(-e_1)$	$\rho(-e_2)$	$\rho(e_1 - e_2)$
$\rho_{(a)}$	0	0	0	0	$-\rho^p$
$\rho_{(b)}$	0	0	ρ_1^d	ρ_2^d	$-\rho^p$
$\rho_{(c)}$	ρ_1^a	ρ_2^a	0	0	$\rho_1^a - \rho^p$

(3.15)

1. $\rho_{(a)}$ corresponds to no admission or departure reward but a preemption cost $-\rho^p < 0$ for all preempted class 2 calls.

2. $\rho_{(b)}$ corresponds to reward ρ_k^d earned upon each class k call completion, and $-\rho^p$ for all preempted class 2 calls.

3. $\rho_{(c)}$ corresponds to reward ρ_k^a earned upon each class k call admission, and $-\rho^p$ for all preempted class 2 calls.

Note that our reward model includes a preemption cost $-\rho^p$ but no blocking cost. The justification for this model is twofold: *i*) One one hand, it is not natural to impose a charge for blocking a call since the customer did not receive any service. On the other hand, customers need to be rewarded in situations where their ongoing service is interrupted due to the arrival of a higher priority customers' call. An everyday example of this is the case of flight overbooking, where customers that have booked their tickets need to be compensated when their reservation is revoked. *ii*) Service disruption is naturally seen as worse than blocking.

3.3.4 Markov decision process and dynamic programming formulation

The system state evolves as a continuous time Markov chain $\tilde{N}_\pi = \{\tilde{N}_\pi(t)\}$ on the state space $\mathcal{N}$. The statistics of the process are determined by λ_1, λ_2, μ_1, μ_2, and c and the choice of an admission and preemption control policy pair $\pi = (\pi^a, \pi^p)$. Let $q_\pi(n, n')$ denote the rate of transition $n \to n'$ under policy π:

$$\begin{array}{rcll} q_\pi(n, n+e_1) & = & \lambda_1(1-\pi^p(n)), & n_1+n_2<c \\ q_\pi(n, n+e_1-e_2) & = & \lambda_1\pi^p(n), & n_2>0 \\ q_\pi(n, n+e_2) & = & \lambda_2\pi^a(n), & n_1+n_2<c \\ q_\pi(n, n-e_k) & = & \mu_k n_k, & n_k>0, \qquad k\in\{1,2\} \end{array} \quad , \; n \in \mathcal{N}. \tag{3.16}$$

All other transition rates are zero. The transition rates are illustrated in Fig. 3.4 (left) for the case $c = 2$. The asymptotic time-average reward rate for the CTMC $\{\tilde{N}_\pi(t)\}$ is

$$\tilde{g}_\pi = \lim_{T\to\infty} \frac{1}{T} \left[\int_0^T r(\tilde{N}_\pi(t))\mathrm{d}t + \sum_{j:t_j \le T} R(\tilde{N}_\pi(t_j), \tilde{N}_\pi(t_j^+)) \right], \tag{3.17}$$

where $\{t_j\}$ is the (countably infinite) collection of state transition times.

Remark 1. *The class of threshold policies is optimal in terms of maximizing $\tilde{g}_\pi$ over the class of coordinate convex admission control policies when Assumptions 1 and 3 are relaxed and a linear reward rate model (no transition rewards) is employed*[41]1.

Uniformization

Uniformization is a standard technique for converting a piecewise constant continuous time Markov chain (CTMC) into an equivalent discrete time Markov chain (DTMC) [51] (§5.1). We select the uniform transition rate $\gamma = \lambda_1 + \lambda_2 + (\mu_1 + \mu_2)c$ to exceed the total transition rate from any state $n \in \mathcal{N}$. The uniformized chain has transition probabilities given by normalizing the transition rates by the uniform transition rate, and adding self-loops to ensure the transition probabilities sum to one:

$$\begin{aligned} q_\pi(n,n') > 0 \quad &\Rightarrow \quad p_\pi(n,n') = q_\pi(n,n')/\gamma \\ \sum_{n'\in\Gamma_\pi(n)} q_\pi(n,n') < \gamma \quad &\Rightarrow \quad p_\pi(n,n) = 1 - \sum_{n'\in\Gamma_\pi(n)} p_\pi(n,n'), \end{aligned} \tag{3.18}$$

for each $n, n' \in \mathcal{N}$. Here $\Gamma_\pi(n) = \{n' \neq n : q_\pi(n,n') > 0\}$ are the neighboring states of n under π. All other transition probabilities are zero. The transition probabilities are illustrated in Fig. 3.4 (right) for the case $c = 2$. Recall that a Markov chain is unichain under a policy if there exists a state that can be reached from any other state ([52] p. 165). The CTMC $\tilde{N}_\pi$ and the DTMC N_π are both unichain for all policies since the origin $o = (0,0) \in \mathcal{N}$ can always be reached from any

[1] For $r(n) = r_1 n_1 + r_2 n_2$ with $r_1 \ge r_2$ and $R(n,n') = 0$ for all $n, n' \in \mathcal{N}$ the optimal expected reward rate over all coordinate convex admission control policies $\tilde{g}^*_{\mathrm{cc}} = \max_{\pi\in\Pi^a_{\mathrm{cc}}} \tilde{g}_\pi$ is obtained by a threshold admission control policy limiting the number of class 2 calls (see last sentence in first column of p. 743 in [41] and note for our model we have $b_1 = b_2 = 1$).

other state under any policy π by simply having all active calls depart before the next arrival. The unichain property is required for several results in §3.4.

After uniformization of the CTMC $\{\tilde{N}_\pi(t)\}$ to an equivalent DTMC $\{N_\pi[i]\}$ as described above, the asymptotic reward per stage is

$$g_\pi = \lim_{I\to\infty} \frac{1}{I} \sum_{i=0}^{I-1} \left(\frac{1}{\gamma} r(N_\pi[i]) + R(N_\pi[i], N_\pi[i+1]) \right). \tag{3.19}$$

Note that (3.17) has units of dollars per unit time while (3.19) has units of dollars per stage. The duration of a stage in seconds is one over the uniform transition rate, $1/\gamma$, and so it follows that the reward per unit time of the DTMC is γg_π. The uniformization of the reward rate function $r(n)$ per unit time in the CTMC to $r(n)/\gamma$ per unit stage in the DTMC ensures $\tilde{g}_\pi = \gamma g_\pi$. Having uniformized, we henceforth restrict our attention to the DTMC $\{N_\pi[i]\}$.

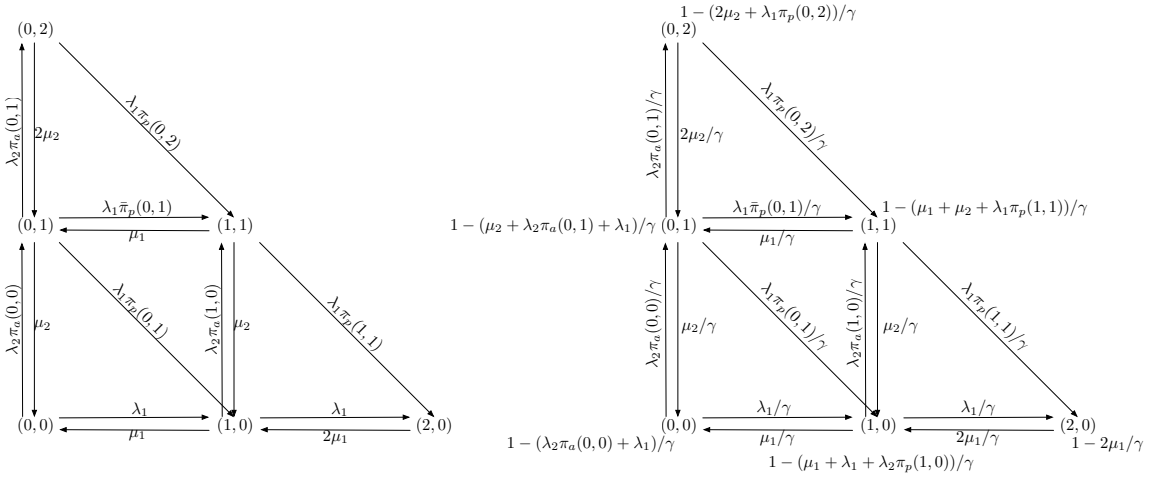


Figure 3.4: Left: the continuous time Markov chain for $c = 2$ for a generic admission and preemption policy pair (π^a, π^p): state transitions are marked with transition rates. **Right:** the uniformized discrete time Markov chain with uniform transition rate γ: states are marked with self-loop transition probabilities and state transitions are marked with transition probabilities. The notation $\bar{\pi}$ denotes complement: $\bar{\pi} = 1 - \pi$.

Infinite horizon average cost discrete time Markov decision process

Our dynamic programming problem is of the infinite horizon average reward discrete time class. For a given policy π, with $i > 0$ stages to go from state $n \in \mathcal{N}$, the expected cumulative reward over the

remaining i stages is

$$v_\pi(n,i) = \frac{1}{\gamma} r(n) + \sum_{n' \in \Gamma_\pi(n) \cup \{n\}} \left(R(n,n') + v_\pi(n', i-1)\right) p_\pi(n,n'),\ n \in \mathcal{N},\ i > 0. \tag{3.20}$$

Note $R(n,n) = 0$ for all $n \in \mathcal{N}$. The initial conditions are $v_\pi(n,0) = 0$ for all $n \in \mathcal{N}$. For each policy π the associated DTMC $\{N_\pi[i]\}$ is *aperiodic, positive recurrent and irreducible.* Aperiodicity and positive recurrence ensure $\lim_{i\to\infty} \frac{1}{i} v(n,i) < \infty$ exists while irreducibility ensures the limit is unique and independent of the starting state n. In particular, define $m_\pi[i] \equiv \min_{n\in\mathcal{N}}\{v_\pi(n,i) - v_\pi(n,i-1)\}$ and $M_\pi[i] \equiv \max_{n\in\mathcal{N}}\{v_\pi(n,i) - v_\pi(n,i-1)\}$. Then

$$\lim_{i\to\infty} m_\pi[i] = \lim_{i\to\infty} M_\pi[i] = g_\pi \tag{3.21}$$

and g_π is independent of the initial state (see p. 210 in [52]). Thus the asymptotic expected reward per stage, as i gets large, is

$$g_\pi = \lim_{i\to\infty} \frac{1}{i} v_\pi(n,i),\ n \in \mathcal{N}. \tag{3.22}$$

Define $h_\pi \equiv (h_\pi(n), n \in \mathcal{N})$ as the asymptotic reward increment per stage at each state; these increments obey

$$h_\pi(n') - h_\pi(n) = \lim_{i\to\infty} v_\pi(n',i) - v_\pi(n,i),\ n, n' \in \mathcal{N}. \tag{3.23}$$

Without loss of generality we may pick a special state, say the origin $o \equiv (0,0) \in \mathcal{N}$, and set $h(o) = 0$. Define

$$\begin{aligned} \Delta_{v_\pi}(n,i) &\equiv v_\pi(n,i) - v_\pi(n,i-1) \\ \Delta_{v_\pi}(n',n,i-1) &\equiv v_\pi(n',i-1) - v_\pi(n,i-1) \\ \Delta_{h_\pi}(n',n) &\equiv h_\pi(n') - h_\pi(n). \end{aligned} \tag{3.24}$$

For each $n \in \mathcal{N}$ and $i > 0$ we can subtract $v_\pi(n, i-1)$ from each side of (3.20) and take the limit as $i \to \infty$.

$$\begin{aligned}
\Delta_{v_\pi}(n,i) &= \frac{1}{\gamma} r(n) + \sum_{n' \in \Gamma_\pi(n) \cup \{n\}} \left(R(n,n') + \Delta_{v_\pi}(n', n, i-1)\right) p_\pi(n,n') \\
\lim_{i\to\infty} \Delta_{v_\pi}(n,i) &= \frac{1}{\gamma} r(n) + \sum_{n' \in \Gamma_\pi(n) \cup \{n\}} \Big(R(n,n') + \lim_{i\to\infty}\left(\Delta_{v_\pi}(n', n, i-1)\right)\Big) p_\pi(n,n') \\
g_\pi &= \frac{1}{\gamma} r(n) + \sum_{n' \in \Gamma_\pi(n)} \left(R(n,n') + \Delta_{h_\pi}(n', n)\right) p_\pi(n,n') \qquad (3.25)
\end{aligned}$$

The asymptotic reward per stage g_π under policy π can be obtained by solving the N equations in (3.25) along with $h_\pi(o) = 0$ for the $N+1$ variables g_π, h_π.

Given a decision space $\mathcal{S}$, the optimal policy π^* over the class of policies $\Pi_\mathcal{S}$ permissible under $\mathcal{S}$ is given by the solution to the Bellman equation:

$$g_{\pi^*} = \frac{1}{\gamma} r(n) + \max_{\pi(n) \in \mathcal{S}(n)} \left\{ \sum_{n' \in \Gamma_\pi(n)} \left(R(n,n') + h_\pi(n') - h_\pi(n)\right) p_\pi(n,n') \right\}, \; n \in \mathcal{N}. \qquad (3.26)$$

We will not address optimal policies in our analysis (§3.4), however, we will do so in our numerical results (§3.5).

3.4 Model analysis

In this section we present the three results discussed in the Introduction: *i*) preempting always when full implies not preempting when not full is optimal (§3.4.1), *ii*) a sufficient condition for the superiority of preempting always when full (with no admission control) over optimal coordinate convex admission control (with no preemption control) (§3.4.2), and *iii*) a characterization of reward models that are equivalent in terms of per stage reward under a given policy (§3.4.3).

3.4.1 Preempting always when full implies not preempting when not full

In this subsection we study the performance of preemption policies with a focus on establishing conditions under which it is optimal to not preempt when the link is not full. Prop. 2 below states that, for "reasonable" reward models, if we always preempt when the link is full then the average per

Table 3.1: Model parameters ($k \in \{1, 2\}$)

c	link capacity
λ_k	arrival rate of class k calls
μ_k	service rate of class k calls
n_k	number of active class k calls
$n = (n_1, n_2)$	link state
$\mathcal{N} = \{n : n_1 + n_2 \leq c\}$	link state space
$N = \lvert\mathcal{N}\rvert$	number of states
$\pi^a(n)$	admission decision for an arriving class 2 call in state n
$\pi^p(n)$	preemption decision for an arriving class 1 call in state n
π^a	admission control policy
π^p	preemption control policy
$\pi = (\pi^a, \pi^p)$	an admission and preemption policy pair
$\mathcal{S}(n)$	allowable decisions at state n
$\mathcal{S}$	decision space
Π	control policy space, a collection of control policies
$\Pi_{\mathcal{S}}$	collection of control policies permissible under $\mathcal{S}$
$r(n)$	reward rate at state n
$R(n, n')$	reward accrued upon transition $n \to n'$
$\rho(u) = R(n, n + u)$	reward accrued for a transition of "type" u
ρ	a transition reward tuple specifying reward for each of the five transition types
(r, R) or (r, ρ)	reward model
$v_\pi(n, i)$	expected time-cumulative reward under π with i steps to go from state n
$h_\pi(n)$	asymptotic reward increment per stage at state n under π
g_π	long-term time-average expected reward per stage under π

stage reward is increased by removing a preemption from a non full state. By extension, Corollary 1 states that if we always preempt when full then it is optimal to not preempt at any non full state. One might conjecture the stronger statement that for "reasonable" reward models the performance of any preemption policy is improved by removing preemptions at non full states. We show this conjecture is false by a simple counter-example. Finally, we make two conjectures regarding optimal preemption: for "reasonable" reward models *i*) the optimal preemption control only preempts when full (Conjecture 1), and *ii*) optimal preemption control is of threshold type (Conjecture 2). Our numerical results suggest these results to be true but we are unable to prove them.

Recall that upon a preemption state transition $n \to n + e_1 - e_2$ a reward $R(n, n + e_1 - e_2) = \rho(e_1 - e_2)$ is accrued. Without loss in generality we define the "preemption penalty" $f_p \in \mathbb{R}$ as

$$\rho(e_1 - e_2) = \rho(e_1) - \rho(e_2) - f_p. \tag{3.27}$$

The intuition behind this definition is that a preemption *i*) admits a class 1 call that would have earned reward $\rho(e_1)$ upon admission without preemption, *ii*) preempts an active class 2 call that presumably paid $\rho(e_2)$ for its admission and this reward is now forfeit, and *iii*) requires the link pay a penalty f_p to the preempted customer. This intuition merely adds a helpful interpretation to (3.27), but the interpretation is not necessary for (3.27) to be valid. The phrase "preemption penalty" suggests we usually think of f_p as nonnegative, but it need not be. This penalty serves as the disincentive for preemption and compensates for the disruption in service experienced by the preempted customer. We define a "reasonable" reward model as one for which $f_p \geq 0$.

Proposition 2. *Let $\pi = (\pi^a, \pi^p)$ be a policy pair with $\pi^a \in \Pi^a_{\text{cc}} = \Pi^a_{\text{th}}$ and $\pi^p \in \Pi^p_{\text{awf}}$. If there exists a non full state under which a call is preempted (i.e., there exists $\hat{n} \in \mathcal{N}$ with $\hat{n}_1 + \hat{n}_2 < c$ and $\hat{n}_2 > 0$ for which $\pi^p(\hat{n}) = 1$) then form the policy $\bar{\pi} = (\pi^a, \bar{\pi}^p)$ which behaves identically to π on all states $n \neq \hat{n}$ but does not preempt at $\hat{n}$:*

$$\bar{\pi}^p(n) = \begin{cases} \pi^p(n), & n \neq \hat{n} \\ 0, & n = \hat{n} \end{cases}. \tag{3.28}$$

Define $\Delta r(\hat{n}) \equiv r(\hat{n}) - r(\hat{n} - e_2) \geq 0$ *(by Assumption 4). Let* f_p *be as in* (3.27). *If*

$$f_p \geq \begin{cases} A_1 \equiv \frac{-(\Delta r(\hat{n}) + \rho(-e_2)\mu_2) + \lambda_2 \rho(e_2)}{\gamma + \lambda_1} - \rho(e_2), & \lambda_2 \rho(e_2) \leq \rho(-e_2)\mu_2 + \Delta r(\hat{n}) \\ A_2 \equiv \frac{-(\Delta r(\hat{n}) + \rho(-e_2)\mu_2) + \lambda_2 \rho(e_2)}{\gamma - \lambda_1} - \rho(e_2), & \textit{else} \end{cases} \tag{3.29}$$

then the expected per stage reward under $\bar{\pi}$ *equals or exceeds that under* π*:* $g_{\bar{\pi}} \geq g_\pi$. *If*

$$f_p \leq P \equiv -\frac{\Delta r(\hat{n}) + \rho(-e_2)\mu_2}{\gamma - \lambda_1} - \rho(e_2) \tag{3.30}$$

then the expected per stage reward under π *equals or exceeds that under* $\bar{\pi}$*:* $g_\pi \geq g_{\bar{\pi}}$. *Further,* $P < \min\{A_1, A_2\} \leq \max\{A_1, A_2\} \leq 0$.

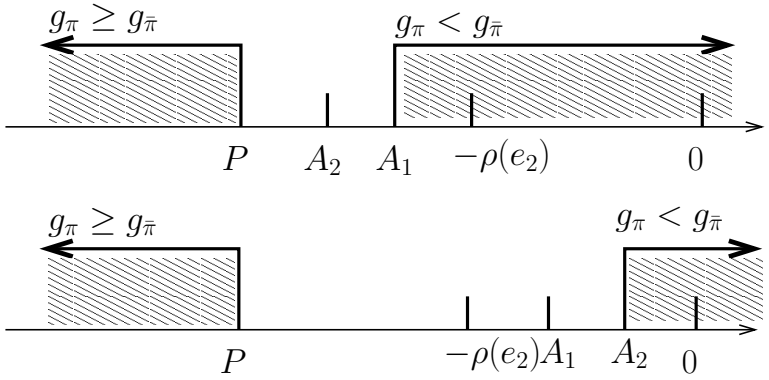


Figure 3.5: Illustration of Prop. 2. The axes are f_p where f_p is defined in (3.27). For $f_p \geq \max\{A_1, A_2\}$ the removal of the preemption at the non full state $\hat{n}$ results in an increase in expected reward. Only for $f_p < P$ (effectively rewarding preemption) does removing the preemption at $\hat{n}$ reduce expected reward. **Top:** $P < A_2 < A_1 < -\rho(e_2) \leq 0$. **Bottom:** $P < -\rho(e_2) < A_1 < A_2 < 0$.

Prop. 2 is illustrated in Fig. 3.5. Note the proposition does not order $g_\pi, g_{\bar{\pi}}$ for $f_p \in [P, \max\{A_1, A_2\}]$. Note $f_p < P$ requires a rather "unreasonable" reward model with both *i)* f_p and *ii)* either $\rho(-e_2)$ and/or $\rho(e_2)$ sufficiently negative. Before we prove Prop. 2 we state two technical lemmas; both proofs are in the Appendix.

Lemma 1. *Let* $\pi = (\pi^a, \pi^p)$ *be a policy pair with* $\pi^a \in \Pi^a_{\text{cc}} = \Pi^a_{\text{th}}$ *and* $\pi^p \in \Pi^p_{\text{awf}}$. *If* (3.29) *holds then*

$$\rho(e_1) - \rho(e_1 - e_2) + h_\pi(n) - h_\pi(n - e_2) \geq 0 \ \forall n \in \mathcal{N} : n_1 > 0, n_2 > 0. \tag{3.31}$$

Lemma 2. *Let $\pi = (\pi^a, \pi^p)$ be a policy pair with $\pi^a \in \Pi^a_{\rm cc} = \Pi^a_{\rm th}$ and $\pi^p \in \Pi^p_{\rm awf}$. If (3.30) holds then*

$$\rho(e_1) - \rho(e_1 - e_2) + h_\pi(n) - h_\pi(n - e_2) \leq 0 \ \forall n \in \mathcal{N} : n_1 > 0, n_2 > 0. \tag{3.32}$$

The sufficient condition (3.29) is not quite intuitive. Note however, that the condition is always satisfied when *i)* $\rho(e_1) \geq \rho(e_2) + \rho(e_1 - e_2)$, *ii)* $\rho(e_1) \geq \rho(e_1 - e_2)$, and *iii)* $\rho(-e_2) \geq 0$. In words, *i)* the reward from a class 1 admission ($\rho(e_1)$) exceeds the sum of rewards from a class 2 admission ($\rho(e_2)$) followed by a preemption ($\rho(e_1 - e_2)$), *ii)* the admission reward for class 1 exceeds preemption reward, and *iii)* class 2 departure reward is nonnegative. Using (3.27), Condition *i)* is equivalent to $f_p \geq 0$, as illustrated in Fig. 3.6. Given condition *i)* is true, conditions *ii)* and *iii)* hold provided we make the natural assumption that non-preemptive admissions and departures generate nonnegative reward.

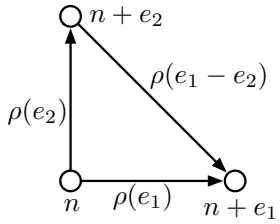


Figure 3.6: Assuming $f_p \geq 0$ for f_p defined in (3.27) is equivalent to $\rho(e_1) \geq \rho(e_2) + \rho(e_1 - e_2)$, i.e., it is better to admit a class 1 call without preemption than to admit a class 2 call but then preempt it to admit a class 1 call.

Proof. of Proposition 2. Let $g_\pi, g_{\bar\pi}$ be the expected per stage reward under $\pi, \bar\pi$, respectively, and let $h_\pi, h_{\bar\pi}$ be the relative rewards under $\pi, \bar\pi$, respectively. By Theorem 3.1 in [52] (g_π, h_π) satisfy the $N + 1$ equations:

$$r(n) + \sum_{n' \in \Gamma_\pi(n)} p_\pi(n, n') R(n, n') - g_\pi + \sum_{n' \in \Gamma_\pi(n)} p_\pi(n, n') h_\pi(n') = (1 - p_\pi(n, n)) h_\pi(n) \tag{3.33}$$

for each $n \in \mathcal{N}$ and $h(o) = 0$ for $o = (0, 0) \in \mathcal{N}$ the origin. Note (3.33) is a rearrangement of (3.25).

The proof that (3.29) is sufficient for $g_{\bar\pi} \geq g_\pi$ is very similar to the proof that (3.30) is sufficient for $g_\pi \geq g_{\bar\pi}$, hence we only present the proof of the former. By the policy improvement theorem for

stochastic dynamic programming (e.g., Theorem 3.2 in [52]) a sufficient condition for $g_{\bar{\pi}} \geq g_\pi$ is that (g_π, h_π) obey the N inequalities:

$$r(n) - g_\pi + \sum_{n' \in \Gamma_{\bar{\pi}}(n)} p_{\bar{\pi}}(n, n')(R(n, n') + h_\pi(n')) + p_{\bar{\pi}}(n, n) h_\pi(n) \geq h_\pi(n) \tag{3.34}$$

for each $n \in \mathcal{N}$. The neighborhood of $\hat{n}$ changes slightly under the two policies. Under π we have $\hat{n} + e_1 - e_2 \in \Gamma_\pi(\hat{n})$ and $\hat{n} + e_1 \notin \Gamma_\pi(\hat{n})$ (preempt in $\hat{n}$ under π), while under $\bar{\pi}$ we have $\hat{n} + e_1 - e_2 \notin \Gamma_{\bar{\pi}}(\hat{n})$ and $\hat{n} + e_1 \in \Gamma_\pi(\hat{n})$ (don't preempt in $\hat{n}$ under $\bar{\pi}$). This change is also reflected in the transition probabilities p from $\hat{n}$:

$$\begin{array}{rclrcl} p_\pi(\hat{n}, \hat{n} + e_1) & = & 0 & p_{\bar{\pi}}(\hat{n}, \hat{n} + e_1) & = & \lambda_1 \\ p_\pi(\hat{n}, \hat{n} + e_1 - e_2) & = & \lambda_1 & p_{\bar{\pi}}(\hat{n}, \hat{n} + e_1 - e_2) & = & 0 \end{array} \tag{3.35}$$

All transition probabilities and neighborhoods under the two policies are identical aside from the change in the neighborhood of $\hat{n}$ and the change in the transition probabilities in (3.35). Observe that (3.34) therefore holds with equality for all $n \in \mathcal{N} \setminus \{\hat{n}\}$. It remains to establish (3.34) for $n = \hat{n}$. The LHS of (3.34) for $n = \hat{n}$ is:

$$r(\hat{n}) + \sum_{n' \in \Gamma_{\bar{\pi}}(\hat{n})} p_{\bar{\pi}}(\hat{n}, n') R(\hat{n}, n') - g_\pi + \sum_{n' \in \Gamma_{\bar{\pi}}(\hat{n})} p_{\bar{\pi}}(\hat{n}, n') h_\pi(n'). \tag{3.36}$$

Adding and subtracting $p_\pi(\hat{n}, \hat{n} + e_1 - e_2) R(\hat{n}, \hat{n} + e_1 - e_2)$ and $p_\pi(\hat{n}, \hat{n} + e_1 - e_2) h_\pi(\hat{n} + e_1 - e_2)$ allows:

$$\begin{array}{l} r(\hat{n}) + \sum_{n' \in \Gamma_\pi(\hat{n})} p_\pi(\hat{n}, n') R(\hat{n}, n') - g_\pi + \sum_{n' \in \Gamma_\pi(\hat{n})} p_\pi(\hat{n}, n') h_\pi(n') \quad + \\ p_{\bar{\pi}}(\hat{n}, \hat{n} + e_1) R(\hat{n}, \hat{n} + e_1) - p_\pi(\hat{n}, \hat{n} + e_1 - e_2) R(\hat{n}, \hat{n} + e_1 - e_2) \quad + \\ \qquad p_{\bar{\pi}}(\hat{n}, \hat{n} + e_1) h_\pi(\hat{n} + e_1) - p_\pi(\hat{n}, \hat{n} + e_1 - e_2) h_\pi(\hat{n} + e_1 - e_2) \end{array} \tag{3.37}$$

Substitute the RHS of (3.33) (with $n = \hat{n}$) for the first line in (3.37) and use (3.35) to obtain:

$$(1 - p_\pi(\hat{n}, \hat{n}))h_\pi(\hat{n}) + \tag{3.38}$$
$$\lambda_1 \left(R(\hat{n}, \hat{n} + e_1) - R(\hat{n}, \hat{n} + e_1 - e_2) + h_\pi(\hat{n} + e_1) - h_\pi(\hat{n} + e_1 - e_2)\right).$$

Noting $p_\pi(\hat{n}, \hat{n}) = p_{\bar{\pi}}(\hat{n}, \hat{n})$ it follows that the theorem is proved provided we can establish

$$R(\hat{n}, \hat{n} + e_1) - R(\hat{n}, \hat{n} + e_1 - e_2) + h_\pi(\hat{n} + e_1) - h_\pi(\hat{n} + e_1 - e_2) \geq 0. \tag{3.39}$$

That is

$$\rho(e_1) - \rho(e_1 - e_2) + h_\pi(\hat{n} + e_1) - h_\pi(\hat{n} + e_1 - e_2) \geq 0. \tag{3.40}$$

But this is true by assumption: by Lemma 1, (3.29) implies (3.31) holds for all n and thus it holds for $n = \hat{n} + e_1$. □

The following Corollary states that always preempting at full states ensures it is optimal to never preempt at non full states, for all "reasonable" reward models. The corollary follows immediately from repeated application of Prop. 2.

Corollary 1. *Let $\pi = (\pi^a, \pi^p)$ be a policy pair with $\pi^a \in \Pi^a_{\text{cc}} = \Pi^a_{\text{th}}$ and $\pi^p \in \Pi^p_{\text{awf}}$. Form the policy $\bar{\pi} = (\pi^a, \pi^p_{\text{awf}})$ which has the same class 2 admission policy π^a but always and only preempts when full (no preemption in non full states). If (3.29) always holds then the expected per stage reward under $\bar{\pi}$ equals or exceeds that under π: $g_{\bar{\pi}} \geq g_\pi$.*

Given Corollary 1, one might conjecture that under "reasonable" reward models any policy $\pi = (\pi^a, \pi^p)$ with a preemption at a non full state ($\pi(\hat{n}) = 1$ for $\hat{n} \in \mathcal{N}$ with $\hat{n}_1 + \hat{n}_2 < c$ and $\hat{n}_2 > 0$) will be improved by removing the preemption at $\hat{n}$. That is, form $\bar{\pi} = (\pi^a, \bar{\pi}^p)$ with $\bar{\pi}^p = \pi^p$ for all $n \in \mathcal{N}$ except $\bar{\pi}^p(\hat{n}) = 0$, and conjecture $g_{\bar{\pi}} \geq g_\pi$. The following counter-example shows this conjecture is false even for $c = 2$.

Consider a link with capacity $c = 2$ circuits, linear per state rewards $r(n) = r_1 n_1 + r_2 n_2$, no arrival or departure rewards ($\rho(\pm e_k) = 0$ for $k \in \{1, 2\}$), complete sharing admission control (π^a_{cs}),

and parameter values:

$$\begin{array}{cccccccc} \lambda_1 & \lambda_2 & \mu_1 & \mu_2 & r_1 & r_2 & \rho(e_1 - e_2) \\ \hline 10 & 3 & 5 & 1 & 10 & 1 & -0.1 \end{array}. \tag{3.41}$$

With these parameters we have an offered load of $\alpha_1 = \lambda_1/\mu_1 = 2$ and $\alpha_2 = \lambda_2/\mu_2 = 3$, where class 1 calls accrue reward at a rate of 10 per second, while class 2 calls accrue reward at a much lower rate of 1 per second. Preemptions are relatively inexpensive, incurring a cost of 0.1. As illustrated in Fig. 3.7, there are three possible preemptive states ($(0,1)$, $(0,2)$, and $(1,1)$) and thus eight possible preemption policies. The expected reward per stage for the eight policies is shown in the figure. The claim that performance improves by removing preemptions at non full states is disproved by observing $g_{\pi^p_{(c)}} > g_{\pi^p_{(d)}}$ and $g_{\pi^p_{(e)}} > g_{\pi^p_{(g)}}$. The superiority of $\pi^p_{(a)}$ over $\pi^p_{(b)}$ is guaranteed by Prop. 2 and Corollary 1. The optimality of $\pi^p_{(a)}$ is consistent with Conjecture 1.

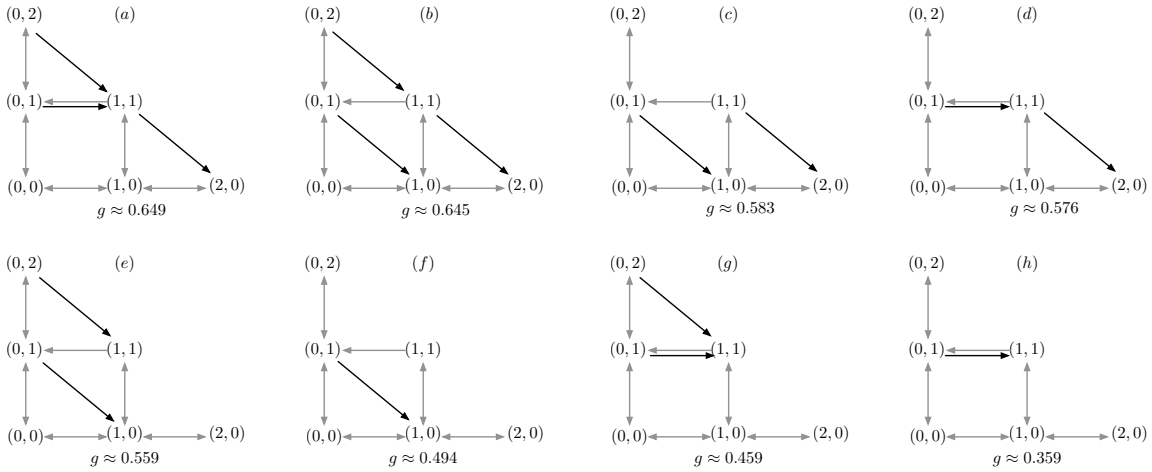


Figure 3.7: The eight possible preemption policies for a loss link with $c = 2$ circuits, ordered by largest to smallest expected per stage reward g.

Although expected per stage reward is not "monotone increasing in removed non full preemptions", we nonetheless make the following conjectures based on our numerical experiments with preemption policies.

Conjecture 1. *Under "reasonable" reward models, the optimal preemption policy does not preempt at non full states:*

$$\max_{\pi \in \Pi^p_{\text{oif}}} g_\pi = \max_{\pi \in \Pi^p_{\text{mf}}} g_\pi. \tag{3.42}$$

The second conjecture is stronger than the first: we conjecture that not only does it suffice to consider preemption only at full states, but that the optimal preemption policy is of threshold type. Define the following two preemption control policy spaces.

- *Threshold when full, type 1 (thf1) control policy:* preempt iff the link is full and $n_2 \leq \tau$: $\pi^p_{\leq\tau}(n) = \mathbf{1}_{0<n_2\leq\tau, n_1+n_2=c}$. For $\tau = 0$ we never preempt, for $\tau = c$ we always preempt when full. Denote the control policy space as $\Pi^p_{\text{thf1}} = \{\pi^p_{\leq 0}, \ldots, \pi^p_{\leq c}\}$.

- *Threshold when full, type 2 (thf2) control policy:* preempt iff the link is full and $n_2 > \tau$: $\pi^p_{>\tau}(n) = \mathbf{1}_{n_2>\tau, n_1+n_2=c}$. For $\tau = 0$ we always preempt when full and for $\tau = c$ we never preempt. Denote the control policy space as $\Pi^p_{\text{thf2}} = \{\pi^p_{>0}, \ldots, \pi^p_{>c}\}$.

Conjecture 2. *Under "reasonable" reward models, the optimal preemption policy does not preempt at non full states and is of threshold type:*

$$\max_{\pi^p \in \Pi^p_{\text{thf1}} \cup \Pi^p_{\text{thf2}}} g_\pi = \max_{\pi^p \in \Pi^p_{\text{mf}}} g_\pi. \tag{3.43}$$

The conjectures, if true, have significant implications for the size of the set of potentially optimal preemption control policies as a function of the link capacity c. Without these conjectures we must potentially search over $2^{\Theta(c^2)}$ policies, as there are $|\mathcal{N}| = \Theta(c^2)$ states and most of them may feasibly employ preemption. If Conjecture 1 is true then we need only consider $\Theta(2^c)$ possible policies, since there are $\Theta(c)$ full states in $\mathcal{N}$ where $n_1 + n_2 = c$ and $n_2 > 0$. If Conjecture 2 is true then we need only consider $\Theta(c)$ possible policies, since there are $\Theta(c)$ possible thresholds.

3.4.2 Admission control vs. preemption control

In this subsection we compare the relative value of admission control vs. preemption control. In particular, we consider a policy π with optimal admission control over the class of threshold / coordinate convex admission control policies but no preemption control, and a policy $\bar{\pi}$ with no admission control but a preemption policy of always and only preempting when full. Prop. 3 provides a sufficient condition under which preemption control outperforms admission control, i.e., $g_{\bar{\pi}} \geq g_\pi$.

This same condition naturally also guarantees that optimal preemption control will outperform optimal admission control (Corollary 2). Recall f_p defined in (3.27).

Proposition 3. *Let $\pi = (\pi^a_*, \pi^p_{\rm np})$ be a policy of optimal admission control without preemption control, where $\pi^a_* = \arg\max_{\pi^a \in \Pi^a_{\rm th}} g_{\pi^a}$. Let $\bar{\pi} = (\pi^a_{\rm cs}, \pi^p_{\rm awf})$ be the policy that always and only preempts when the link is full but does not employ admission control. If*

$$f_p \leq \frac{1}{\lambda_1}\left(r(n) - r(n-e_2) + \rho(-e_2)\mu_2 + (\gamma - \lambda_1 - \lambda_2)\rho(e_2)\right), \tag{3.44}$$

then the expected per stage reward under $\bar{\pi}$ equals or exceeds that under π: $g_{\bar{\pi}} \geq g_\pi$.

We first provide a technical lemma needed in the proof of Prop. 3; the proof of the Lemma is found in the Appendix.

Lemma 3. *Let $\bar{\pi} = (\pi^a_{\rm cs}, \pi^p_{\rm awf})$ be a policy that does not employ admission control but always and only preempts when the link is full. If* (3.44) *holds then*

$$v_{\bar{\pi}}(n,i) - v_{\bar{\pi}}(n-e_2,i) \geq -\rho(e_2)\ \forall n \in \mathcal{N} : n_2 > 0,\ \forall i > 0. \tag{3.45}$$

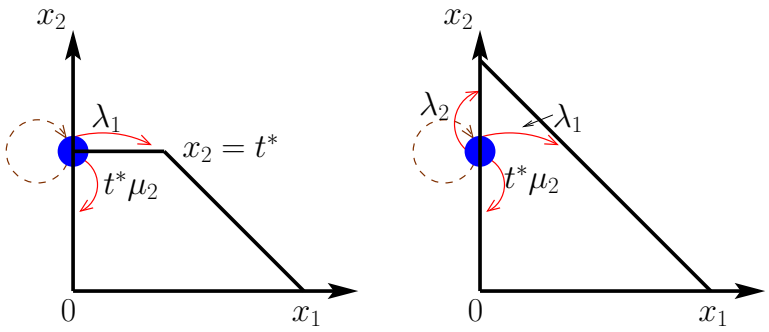


Figure 3.8: Illustration of the transitions from state $n = (0, t^*)$ under π (left) and $\bar{\pi}$ (right).

Proof. of Proposition 3. Let $g_\pi, g_{\bar{\pi}}$ be the expected per stage reward under $\pi, \bar{\pi}$, respectively. From (3.22), we see it suffices to prove that, for some starting state $n \in \mathcal{N}$, the asymptotic reward per stage under $\bar{\pi}$ exceeds that under π starting in some state n, since this limit is not state dependent

(i.e., if it holds for some state n then it holds for all states n):

$$g_{\bar{\pi}} = \lim_{i\to\infty} \frac{1}{i} v_{\bar{\pi}}(n,i) \geq \lim_{i\to\infty} \frac{1}{i} v_{\pi}(n,i) = g_{\pi}. \tag{3.46}$$

A sufficient condition for the above inequality is that for the particular $n \in \mathcal{N}$ in (3.46) we have:

$$v_{\bar{\pi}}(n,i) \geq v_{\pi}(n,i), \ \forall i. \tag{3.47}$$

From Prop. 1 the optimal threshold/coordinate convex admission control policy is a class-2 threshold $t^* \in \{0,\ldots,c\}$. Fix $n = (0,t^*)$, see Fig. 3.8. The expected cumulative reward over the remaining i stages starting from state n under policies π and $\bar{\pi}$ is, respectively

$$\begin{aligned}
\gamma v_{\pi}(n,i+1) &= r(n) + \lambda_1 R(n,n+e_1) + t^*\mu_2 R(n,n-e_2) \\
&+ \lambda_1 v_{\pi}(n+e_1,i) + t^*\mu_2 v_{\pi}(n-e_2,i) \\
&+ (c\mu_1 + (c-t^*)\mu_2 + \lambda_2) v_{\pi}(n,i) \\
\gamma v_{\bar{\pi}}(n,i+1) &= r(n) + \lambda_1 R(n,n+e_1) + t^*\mu_2 R(n,n-e_2) + \lambda_2 R(n,n+e_2) \\
&+ \lambda_1 v_{\bar{\pi}}(n+e_1,i) + \lambda_2 v_{\bar{\pi}}(n+e_2,i) + t^*\mu_2 v_{\bar{\pi}}(n-e_2,i) \\
&+ (c\mu_1 + (c-t^*)\mu_2) v_{\bar{\pi}}(n,i).
\end{aligned} \tag{3.48}$$

From the above two equations, the difference between the expected cumulative reward over the remaining i stages starting from state n of the corresponding policies $\pi, \bar{\pi}$ is

$$\begin{aligned}
\gamma(v_{\bar{\pi}}(n,i+1) - v_{\pi}(n,i+1)) &= \lambda_2\rho(e_2) + \lambda_2(v_{\bar{\pi}}(n+e_2,i) - v_{\bar{\pi}}(n,i)) \\
&+ t^*\mu_2(v_{\bar{\pi}}(n-e_2,i) - v_{\pi}(n-e_2,i)) \\
&+ \lambda_1(v_{\bar{\pi}}(n+e_1,i) - v_{\pi}(n+e_1,i)) \\
&+ (\lambda_2 + c\mu_1 + (c-t^*)\mu_2)(v_{\bar{\pi}}(n,i) - v_{\pi}(n,i)).
\end{aligned} \tag{3.49}$$

We show by induction on i that (3.44) is sufficient to guarantee $v_{\bar{\pi}}(n,i) - v_{\pi}(n,i) \geq 0$ for all i. We

first establish the base case: $i = 0$. By definition $v_\pi(n, 0) = v_{\bar{\pi}}(n, 0) = 0$ for all $n \in \mathcal{N}$ (see below (3.20)). Using (3.49) for $i = 0$ gives the difference for $i = 1$ as

$$\frac{\gamma}{\lambda_2}\left(v_{\bar{\pi}}(n, 1) - v_\pi(n, 1)\right) = \rho(e_2) \geq 0. \tag{3.50}$$

For the induction step, suppose $v_{\bar{\pi}}(n, j) - v_\pi(n, j) \geq 0$ for $n = (0, t^*)$ and for all $j = 1, \ldots, i$. Using (3.49) and the induction hypothesis yields the following inequality for $j = i + 1$:

$$\frac{\gamma}{\lambda_2}\left(v_{\bar{\pi}}(n, i+1) - v_\pi(n, i+1)\right) \geq \rho(e_2) + v_{\bar{\pi}}(n + e_2, i) - v_{\bar{\pi}}(n, i). \tag{3.51}$$

The required conclusion holds provided

$$v_{\bar{\pi}}(n + e_2, i) - v_{\bar{\pi}}(n, i) \geq -\rho(e_2). \tag{3.52}$$

By assumption (3.44) holds, and by Lemma 3 this ensures (3.45) holds for all $n \in \mathcal{N}$ with $n_2 > 0$, which ensures (3.52) for $n = (0, t^*)$. □

The following corollary observes that the sufficient condition (3.44) in Prop. 3 trivially implies the superiority of *optimal* preemption control over optimal admission control. This is immediate from the fact that optimal preemption control outperforms any particular preemption control.

Corollary 2. *Let π be as in Prop. 3. Let $\pi' = (\pi^a_{\text{cs}}, \pi^p_{\text{oif}^*})$ be a policy that has optimal preemption control over full states but has no admission control: $\pi^p_{\text{oif}^*} = \arg\max_{\pi \in \Pi^p_{\text{oif}}} g_\pi$. If (3.44) holds then $g_{\pi'} \geq g_\pi$.*

It is natural to inquire if we might not also obtain a sufficient condition for the superiority of optimal admission control over preempting when full, i.e., a condition for $g_\pi \geq g_{\bar{\pi}}$. Although certainly such a condition exists, our proof technique does not cover this case. The following corollary states this formally; the details are presented in the Appendix.

Corollary 3. *Let $\pi, \bar{\pi}$ be as in Prop. 3. The proof methodology of Prop. 3 fails to yield a sufficient*

condition for $g_\pi \geq g_{\bar{\pi}}$.

3.4.3 Reward model equivalence

Our third set of results characterizes the class of reward models (r, R) (cf. §3.3.3) equivalent in terms of expected per stage reward g_π under a given policy π. In this subsection we write $g_\pi(r, R)$ to emphasize reward model dependence. Let $\mathcal{R} = \{(r, R)\}$ be the space of all reward models. Each policy π determines a partition of $\mathcal{R}$ into equivalence classes.

Definition 6. *The class of reward models* (r, R) *with expected per stage reward* $\xi \in \mathbb{R}$ *under* π *is defined as*

$$\mathcal{R}_\pi(\xi) \equiv \{(r, R) \in \mathcal{R} : g_\pi(r, R) = \xi\}. \tag{3.53}$$

Definition 7. *Two reward models* (r, R) *and* (r', R') *are expected per stage reward equivalent under* π *when*

$$g_\pi(r, R) \equiv g_\pi(r', R'). \tag{3.54}$$

The following two propositions characterize the equivalence class $\mathcal{R}_\pi(\xi)$ and the relationship between expected per stage reward equivalent reward models (r, R) and (r', R') respectively. These propositions are immediate consequences of Theorem 3.1 in [52], as discussed in the proofs.

Proposition 4. *For a policy* π *and* $\xi \in \mathbb{R}$*, the expected per stage reward model equivalence class* $\mathcal{R}_\pi(\xi)$ *is given by*

$$\mathcal{R}_\pi(\xi) = \left\{ (r, R) \in \mathcal{R} : \exists h' \in \mathbb{R}^{N-1} : P_\pi \begin{bmatrix} 0 \\ h' \end{bmatrix} = (\xi - P_\pi \circ R)\mathbf{1} - \frac{1}{\gamma} r \right\}, \tag{3.55}$$

where P_π *is an* $N \times N$ *stochastic (singular) matrix with elements*

$$P_\pi[n, n'] = \begin{cases} -(1 - p_\pi(n, n)), & n' = n \\ p_\pi(n, n'), & n' \in \Gamma(n) \\ 0, & \textit{else} \end{cases}, \tag{3.56}$$

$\mathbf{1}$ *is the* N*-vector of all ones,* R *is viewed as an* $N \times N$ *matrix as in* (3.12)*, and* $P_\pi \circ R$ *is the* $N \times N$

element-wise product matrix with element (n, n') *given by* $P_\pi[n, n']R[n, n']$.

Proposition 5. *Two reward models* (r_1, R_1) *and* (r_2, R_2) *are expected per stage reward equivalent under* π *if there exist* $h_1', h_2' \in \mathbb{R}^{N-1}$ *such that*

$$P_\pi \begin{bmatrix} 0 \\ h_1' \end{bmatrix} + (P_\pi \circ R_1)\mathbf{1} + \frac{1}{\gamma} r_1 = P_\pi \begin{bmatrix} 0 \\ h_2' \end{bmatrix} + (P_\pi \circ R_2)\mathbf{1} + \frac{1}{\gamma} r_2. \tag{3.57}$$

Proof. of Prop. 4. Fix a policy π and $\xi \in \mathbb{R}$. Rearrange (3.25) as

$$\sum_{n' \in \Gamma(n)} h(n')p(n, n') - (1 - p(n, n))h(n) = \xi - \sum_{n' \in \Gamma(n)} R(n, n')p(n, n') - \frac{1}{\gamma} r(n) \tag{3.58}$$

for each $n \in \mathcal{N}$, where we suppress the policy dependence notation since the policy is fixed, and substitute the required $g_\pi = \xi$. We can write the N equations in (3.58) more compactly as

$$Ph = (\xi - (R \circ P))\mathbf{1} - \frac{1}{\gamma} r. \tag{3.59}$$

Note that P is stochastic (each row is nonnegative and sums to zero) by (3.18):

$$\begin{aligned} (P\mathbf{1})(n) &= -1 + p(n, n) + \sum_{n' \in \Gamma(n)} p(n, n') \\ &= -1 + \left(1 - \sum_{n' \in \Gamma(n)} p(n, n')\right) + \sum_{n' \in \Gamma(n)} p(n, n') = 0, \ n \in \mathcal{N}. \end{aligned} \tag{3.60}$$

A stochastic matrix is singular: the N columns are dependent since their sum is the zero vector, hence the matrix is not of full rank and thus not invertible. By Theorem 3.1 in [52], the system of $N+1$ linear equations (the N equations in (3.59) along with $h(o) = 0$ for $o = (0, 0)$ the origin) with $N+1$ unknowns (g and $h = (h(n), n \in \mathcal{N})$) has a unique solution. Further, the solution is such that $g = g_\pi$ is the reward per stage and $h(n) = h_\pi(n)$ for $n \in \mathcal{N}$ is the relative reward under π. The equation in (3.55) simply combines (3.59) with $h(o) = 0$ to yield N equations with $N - 1$ unknowns $h' = (h(n), n \in \mathcal{N} \setminus \{o\})$. □

Proof. of Prop. 5. Following the discussion of the proof of Prop. 4 and using Definition 7, the two reward models (r_1, R_1) and (r_2, R_2) are by expected per stage reward equivalent if their corresponding expected per stage rewards g_1 and g_2 are the same. Solving the equation in (3.55) for ξ for both models and equating them gives (3.57). □

We illustrate reward equivalence for the $c = 1$ case in Remark 2 in §3.5.1.

3.5 Numerical results

This section presents numerical results for a link capable of holding a single call ($c = 1$, §3.5.1), and for larger capacity links ($c \geq 1$, §3.5.2). Similar to the analysis of the optimal admission control in [53], the time index i in this section is the step index of the discrete system in the simulation, which means the elapsed time from the start of the simulation. This is different from the "time to go" definition used in the theoretical analysis, but this reversal is common in dynamic programming [51].

From (3.21) we know the expected per step reward starting form any state converges to a constant as $i \to \infty$. For a given $\epsilon > 0$ our numerical termination condition is to stop at iteration i such that the growth $v(n, i) - v(n, i-1)$ in cumulative reward is relatively constant for all states, i.e.,

$$\max_{n \in \mathcal{N}} \{v(n, i) - v(n, i-1)\} - \min_{n \in \mathcal{N}} \{v(n, i) - v(n, i-1)\} < \epsilon \cdot \min_{n \in \mathcal{N}} \{v(n, i) - v_(n, i-1)\}. \tag{3.61}$$

When the simulation terminates at time index $i = I$, the expected per stage reward is calculated from any $n \in \mathcal{N}$ as:

$$g = v(n, I) - v(n, I-1), \ n \in \mathcal{N}. \tag{3.62}$$

3.5.1 Results for a single circuit link

We consider the simplest case of a link with a single circuit ($c = 1$). The following results illustrate the interplay between admission control and preemption control is non-trivial even for this simplest of models. We give explicit expressions for the expected per stage reward g in terms of the arrival rates λ_1, λ_2, the service rates μ_1, μ_2, and the reward model (r, ρ). For $c > 1$ these expressions are unwieldy. State transition diagrams for the Markov process $\{N[i]\}$ when $c = 1$ are shown in Fig. 3.9

for the three "primitive policies" of (a) forced preemption (left), (b) complete sharing (middle), and (c) forced admission control (right). Note that the optimal preemption control policy π_*^p is found by selecting between complete sharing (b) and forced preemption (a), while the optimal admission control policy π_*^a is found by selecting between complete sharing (b) and forced admission control (c). The optimal joint admission control and preemption policy π_* is found by selecting the maximum between all three cases (a, b, c).

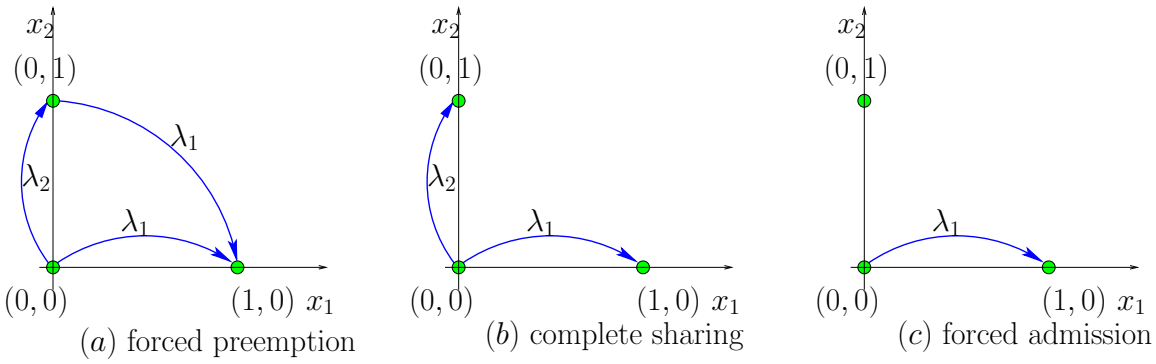


Figure 3.9: Markov process of a single-link network model with $c = 1$. Only the arrival transitions are shown for clarity.

Let g_a, g_b, g_c be the expected per stage reward under cases $(a), (b), (c)$, respectively in Fig. 3.9. Let h_a, h_b, h_c be the relative rewards under $(a), (b), (c)$, respectively. From (3.33) and Theorem 3.1 in [52], we obtain four equations for cases (a) and (b) and three equations for case (c). Note $\mathcal{N} = \{o, e_1, e_2\}$ with $o = (0,0)$, $e_1 = (1,0)$ and $e_2 = (0,1)$. Define $\lambda \equiv \lambda_1 + \lambda_2$ as the total arrival rate and $\alpha_k \equiv \frac{\lambda_k}{\mu_k}$ as the class k offered load, for $k \in \{1, 2\}$. The four equations of case (a) are:

$$\begin{aligned}
\lambda h_a(o) &= r(o) + \lambda_1 \rho(e_1) + \lambda_2 \rho(e_2) - \gamma g_a + \lambda_1 h_a(e_1) + \lambda_2 h_a(e_2) \\
\mu_1 h_a(e_1) &= r(e_1) + \mu_1 \rho(-e_1) - \gamma g_a + \mu_1 h_a(o) \\
\lambda h_a(e_2) &= r(e_2) + \mu_2 \rho(-e_2) + \lambda_1 \rho(e_1 - e_2) - \gamma g_a + \lambda_1 h_a(e_1) + \mu_2 h_a(o) \\
h_a(o) &= 0
\end{aligned} \tag{3.63}$$

where $\gamma = \lambda + \mu_1 + \mu_2$. The solutions of the above equations for case (a) and of the analogous

equations for cases (b) and (c) are

$$\begin{aligned}
\gamma g_a &= \frac{1}{1+\frac{\lambda_1}{\mu_1}}\left(\frac{\lambda_1+\mu_2}{\lambda+\mu_2}r(o)+\frac{\lambda_1}{\mu_1}r(e_1)+\frac{\lambda_2}{\lambda+\mu_2}r(e_2)+\frac{\lambda_1+\mu_2}{\lambda+\mu_2}\lambda_1\rho(e_1)+\right. \\
&\quad \left.\frac{\lambda_2+\mu_2}{\lambda+\mu_2}\lambda_2\rho(e_2)+\lambda_1\rho(-e_1)+\frac{\mu_2}{\lambda+\mu_2}\lambda_2\rho(-e_2)+\frac{\lambda_1}{\lambda+\mu_2}\lambda_2\rho(e_1-e_2)\right) \qquad (3.64) \\
\gamma g_b &= \frac{1}{1+\frac{\lambda_1}{\mu_1}+\frac{\lambda_2}{\mu_2}}\left(r(o)+\frac{\lambda_1}{\mu_1}r(e_1)+\frac{\lambda_2}{\mu_2}r(e_2)+\lambda_1\rho(e_1)+\lambda_2\rho(e_2)\right. \\
&\quad \left.+\lambda_1\rho(-e_1)+\lambda_2\rho(-e_2)\right) \qquad (3.65) \\
\gamma g_c &= \frac{1}{1+\frac{\lambda_1}{\mu_1}}\left(r(o)+\frac{\lambda_1}{\mu_1}r(e_1)+\lambda_1\rho(e_1)+\lambda_1\rho(-e_1)\right). \qquad (3.66)
\end{aligned}$$

These expressions are used below to establish an example of reward equivalence, and are plotted in Fig. 3.10.

Remark 2. *Recall that Prop. 5 gives a condition for two reward models to be expected per stage reward equivalent for a given policy. Consider two reward models* $(r_{(a)}, \rho_{(a)})$ *and* $(r_{(b)}, \rho_{(b)})$ *where* $r_{(a)}(n) = r_1 n_1 + r_2 n_2$ *and* $r_{(b)}(n) = 0$*, and*

	$\rho(e_1)$	$\rho(e_2)$	$\rho(-e_1)$	$\rho(-e_2)$	$\rho(e_1-e_2)$
$\rho_{(a)}$	0	0	0	0	0
$\rho_{(b)}$	0	0	r_1/μ_1	r_2/μ_2	0

(3.67)

Substituting $(r_{(a)}, \rho_{(a)})$ *and* $(r_{(b)}, \rho_{(b)})$ *into* (3.65) *for case* (b) *(complete sharing) gives*

$$\gamma g(r_{(a)}, \rho_{(a)}) = \gamma g(r_{(b)}, \rho_{(b)}) = \frac{\alpha_1 r_1 + \alpha_2 r_2}{1+\alpha_1+\alpha_2}. \qquad (3.68)$$

Thus these two reward models are expected per stage reward equivalent.

Throughout the rest of this subsection and §3.5.2 we use a reward model (r, ρ) with linear reward rates, no admission or departure transition rewards, and a preemption penalty:

$$r(n) = r_1 n_1 + r_2 n_2, \quad \begin{array}{ccccc} \rho(e_1) & \rho(e_2) & \rho(-e_1) & \rho(-e_2) & \rho(e_1-e_2) \\ \hline 0 & 0 & 0 & 0 & -f_p \end{array} \qquad (3.69)$$

where f_p is defined in (3.27). Under this reward model, the sufficient condition (3.44) in Prop. 3 becomes

$$f_p < \frac{r_2}{\lambda_1}. \tag{3.70}$$

We now present numerical results for the $c = 1$ case. In Fig. 3.10 we use the following parameters:

case	λ_1	λ_2	μ_1	μ_2	r_1	r_2	f_p
(a)	1.0	1.0	1.0	1.0		1.0	4.5
(b)	1.0	1.0	1.0	1.0		4.5	1.0
(c)		1.0	0.5	0.5	5.0	3.0	1.5
(d)		1.0	2.5	0.5	5.0	3.0	1.5

(3.71)

From plot (a), we see i) forced admission control is always better than forced preemption, ii) forced admission control is better than complete sharing for $r_1 > 2$ (approx.), and iii) forced preemption is better than complete sharing for $r_1 > 5.5$ (approx.). In plot (b) we have decreased the preemption cost from 4.5 to 1.0 and increased r_2 from 1.0 to 4.5. Now we see that forced preemption is best for $r_1 \geq 5.5$ (approx.) and is always better than forced admission control. This is guaranteed by Prop. 3 as the sufficient condition (3.70) under this reward model (3.69) is satisfied: $1.0 = f_p < r_2/\lambda_1 = 4.5$. In plots (c) and (d) we vary the class 1 arrival rate λ_1. Note in plot (c) we use $\mu_1 = 1.0$ while in plot (d) we use $\mu_1 = 2.5$, hence class 1 calls leave five times as quickly in the latter case. This smaller load of class 1 calls in plot (d) explains why full sharing is optimal for small to moderate λ_1. Note that all three policies converge to the same value in both plots as λ_1 grows large: when class 1 call load outstrips class 2 call load it doesn't matter what controls are applied. Of course there is always a slight benefit to using forced admission in the large λ_1 regime since the other two policies incur unnecessary costs by either admitting class 2 calls or admitting and then preempting class 2 calls.

3.5.2 Numerical results for multi circuit links

The previous subsections gave analysis and numerical investigation of the $c = 1$ case; we now give a numerical investigation of the $c > 1$ case. In this subsection we investigate four policies, see Fig. 3.11:

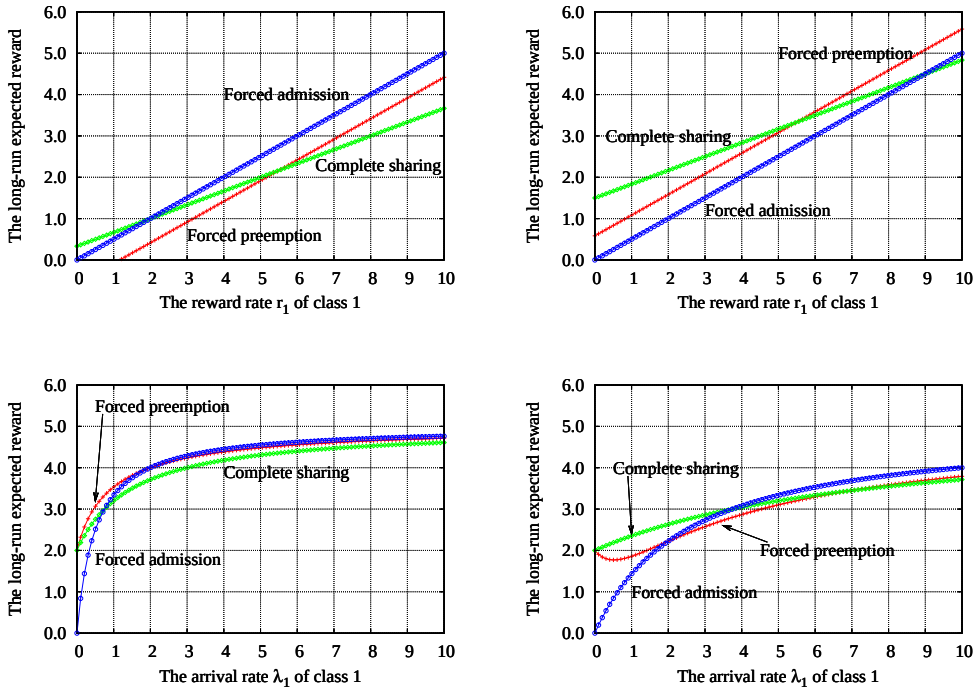


Figure 3.10: Plots for §3.5.1. The expected reward rate per unit time γg for $c = 1$ for the three policies in Fig. 3.9. The top plots vary r_1 while the bottom plots vary λ_1. Note that each of the three policies is optimal in certain parameter regimes.

- $(\pi^a_{\rm cs}, \pi^p_{\rm np})$ (CS in plots): no admission control, no preemption control.
- $(\pi^a_*, \pi^p_{\rm np})$ (oAC in plots): optimal admission control over class 2 threshold policies, no preemption control.
- $(\pi^a_{\rm cs}, \pi^p_{{\rm oif}*})$ (Pre in plots): no admission control, optimal preemption control over full states.
- $(\pi^a_*, \pi^p_{{\rm oif}*})$ (Pre+oAC in plots): optimal admission control over class 2 threshold policies and optimal preemption control over full states.

We offer plots that vary the following parameters: r_1 and f_p (§3.5.2), λ_1 and λ_2 (§3.5.2), and μ_1 and μ_2 (§3.5.2).

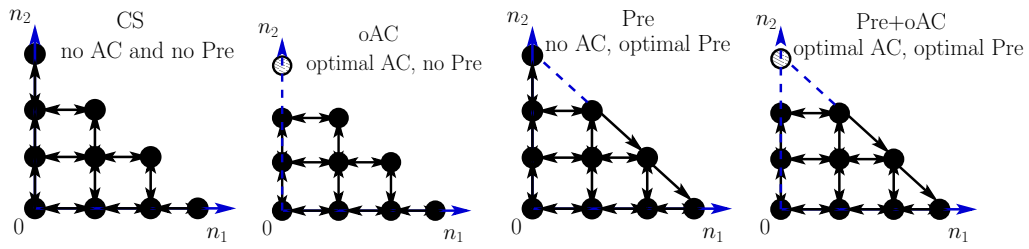


Figure 3.11: Figures for §3.5.2. Four policies are investigated in our numerical results.

Varying r_1 and f_p

Fig. 3.12 shows four cases where we vary r_1 (cases $(a), (b), (c)$) and f_p (case (d)). The parameters for the four cases are

case	c	λ_1	λ_2	μ_1	μ_2	r_1	r_2	f_p
(a)	6	0.1	1.0	1.0	0.1		1.0	0.5
(b)	50	1.0	5.0	0.2	0.1		1.0	0.5
(c)	50	3.0	4.0	0.1	0.1		1.0	0.5
(d)	6	1.0	5.0	1.0	0.1	2.0	1.0	

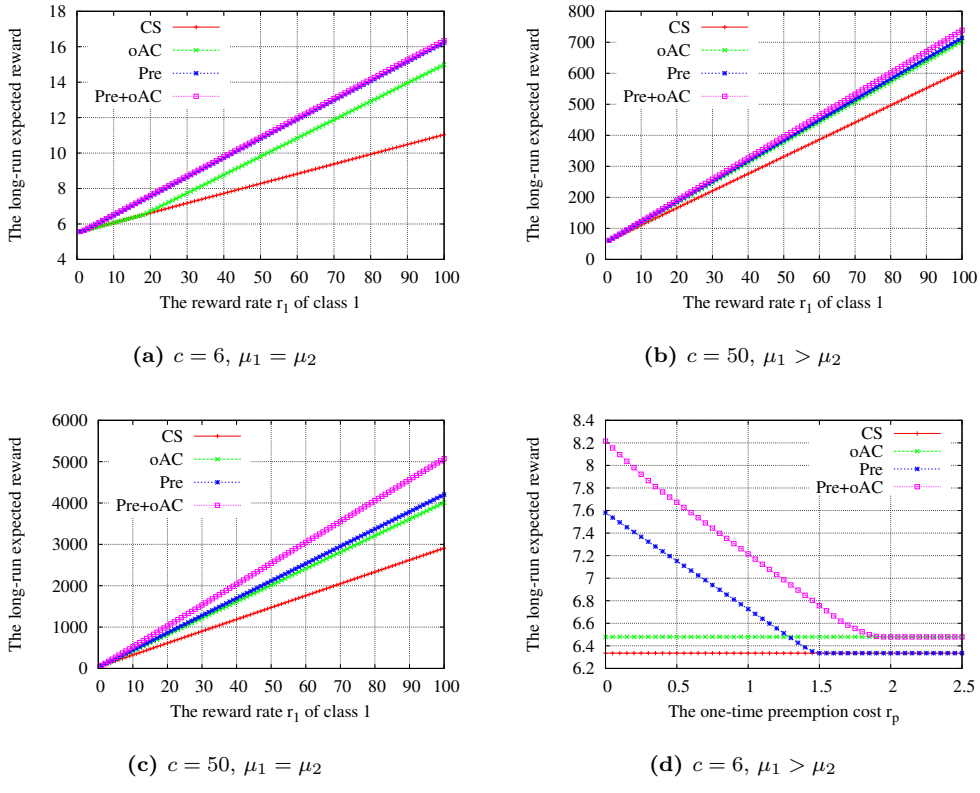
(3.72)

Cases $(a), (d)$ have link capacity $c = 6$; cases $(b), (c)$ have link capacity $c = 50$.

We first discuss cases $(a), (b), (c)$. In case (a) the offered load of class 2 calls is $\alpha_2 = \lambda_2/\mu_2 = 10$ while the offered load for class 1 calls is $\alpha_1 = \lambda_1/\mu_1 = 1/10$, and further the link capacity $c = 6$ is inadequate to handle the aggregate offered load. Complete sharing performs poorly since a control mechanism is required to ensure the class 1 calls are admitted even if the link is full. The preemption cost is low and hence optimal preemption performs almost as well as optimal joint control. In case (b) the offered loads are $\alpha_1 = 5$ and $\alpha_2 = 50$ and the link capacity is $c = 50$. Here both admission control alone and preemption control alone nearly achieve the performance of joint control, but the no control case performs poorly. In case (c) the offered loads are $\alpha_1 = 30$ and $\alpha_2 = 40$ and the link capacity is again $c = 50$. Now we observe that $i)$ joint control outperforms individual admission or preemption control, and $ii)$ both individual controls outperform no control. In all three cases we observe that for r_1 small (near $r_2 = 1$) the four policies perform identically, since there is little

distinction between class 1 and class 2 calls, and thus no need for control. We have also investigated performance as a function of r_2 and have obtained qualitatively similar results as for varying r_1; the reason being that what matters is the ratio r_1/r_2 and not their absolute values.

In case (d) the expected rewards received from $\pi^a_{\rm cs}$ and π^a_* are constant since there is no preemption in these two policies. π^p_* is better than $\pi^a_{\rm cs}$ and π^a_* when the preemption cost f_p is small. However, as f_p increases, the superiority of π^p_* decreases. The sufficient condition (3.70) for the superiority of preemption over optimal admission control for these parameters is $f_p < r_2/\lambda_1 = 1$ but we observe that in fact preemption is superior for $f_p < 1.25$ (approx.). Further note that for $f_p > 1.5$ (approx.) the cost of preemption is too high and the optimal preemption policy is in fact complete sharing (no preemption). Next note that optimal joint control outperforms both individual preemption and admission control for $f_p < 1.8$ (approx.) but performs identically with optimal admission control alone (no preemption) for $f_p > 1.8$.



(a) $c = 6$, $\mu_1 = \mu_2$ **(b)** $c = 50$, $\mu_1 > \mu_2$

(c) $c = 50$, $\mu_1 = \mu_2$ **(d)** $c = 6$, $\mu_1 > \mu_2$

Figure 3.12: Plots for §3.5.2. The expected reward rate per unit time γg for the four policies in §3.5.2. Cases $(a), (b), (c)$ vary r_1; case (d) varies f_p.

Varying λ_1 and λ_2

Fig. 3.13 shows four cases varying λ_1. The parameters for the four cases are

case	c	λ_1	λ_2	μ_1	μ_2	r_1	r_2	f_p
(a)	6		5.0	1.0	0.1	2.0	1.0	0.5
(b)	100		5.0	1.0	0.1	2.0	1.0	0.5
(c)	100		5.0	0.1	0.1	2.0	1.0	0.5
(d)	100		25.0	0.1	0.5	2.0	1.0	0.5

(3.73)

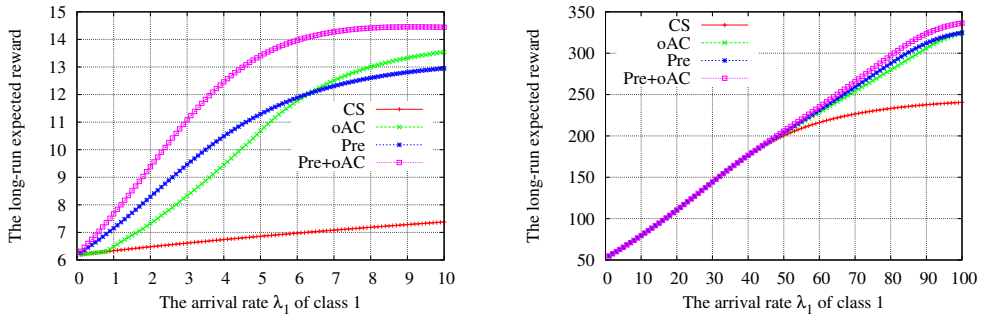
Case (a) shows dramatic differences in performance under the four policies. Condition $\lambda_1 < r_2/f_p = 2$ in (3.70) guarantees the superiority of preemption over admission control for this regime; its superiority extends to $\lambda_1 = 6$ (approx.). For cases $(b), (c), (d)$ observe no control is required for small to moderate λ_1 as the link is underloaded. At $\lambda_1 \geq 50$ (approx.) for case (b) and $\lambda_1 \geq 5$ (approx.) for cases $(c), (d)$, however, control is needed to achieve optimal performance. In case (b) either control suffices but some control is necessary: employing either optimal admission control or optimal preemption alone achieves a performance nearly equivalent to that under optimal joint control, but employing no control incurs a major performance hit. In case (c) both controls are required for optimal performance, but either control alone is significantly better than no control. In case (d) both controls are required for optimal performance, but either control alone is equivalent in performance to no control at all.

Fig. 3.14 shows performance as a function of λ_2; the two cases are:

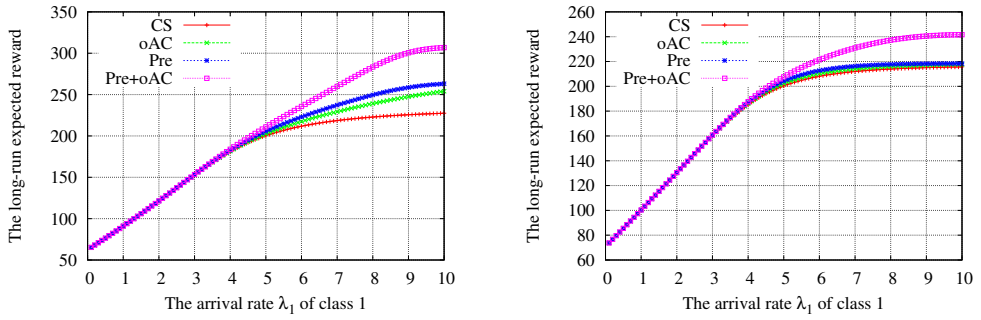
case	c	λ_1	λ_2	μ_1	μ_2	r_1	r_2	f_p
(a)	100	5.0		0.1	1.0	2.0	1.0	0.5
(b)	100	50.0		1.0	0.1	2.0	1.0	0.5

(3.74)

Note $i)$ the class 1 offered load is $\alpha_1 = 50$ and the class 2 offered load varies from 1 to 100 in both cases, and $ii)$ in case (a) $\mu_1 < \mu_2$ while in case (b) $\mu_1 > \mu_2$. No control is needed for small to moderate class 2 load (approx. $\alpha_2 = \lambda_2/\mu_2 = 50$) since the system is underloaded. Performance is much more



(a) $c = 6$ **(b)** $c = 100$, $\mu_1 > \mu_2$



(c) $c = 100$, $\mu_1 = \mu_2$ **(d)** $c = 100$, $\mu_1 < \mu_2$

Figure 3.13: Plots for §3.5.2. Case (a) has $c = 6$ while cases $(b), (c), (d)$ have $c = 100$, all vary λ_1 and show the expected reward rate per unit time γg for the four policies in §3.5.2. Cases $(b), (c), (d)$ represent $\mu_1 > \mu_2$, $\mu_1 = \mu_2$ and $\mu_1 < \mu_2$, respectively.

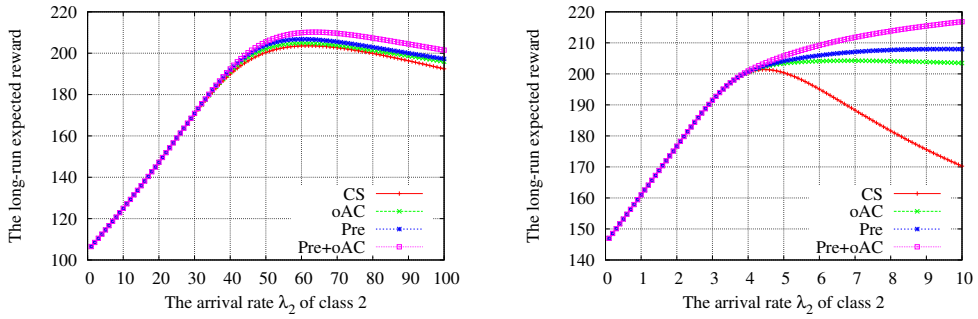
sensitive to control for case (b) than for case (a): all four controls have similar performance in (a) while the four controls have distinct performance in (b). The expected reward per unit time γg is not necessarily monotone increasing in λ_2. In particular, $\gamma = \lambda_1 + \lambda_2 + (\mu_1 + \mu_2)c$ is increasing in λ_2 but the expected reward per stage g may decrease in λ_2.

Varying μ_1 and μ_2

Fig. 3.15 shows performance as a function of μ_1 and μ_2 for two cases:

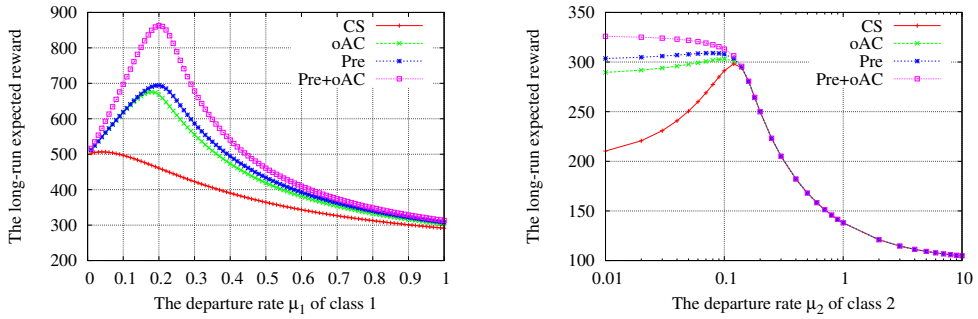
case	c	λ_1	λ_2	μ_1	μ_2	r_1	r_2	f_p
(a)	100	20.0	10.0		0.1	5.0	2.0	0.1
(b)	100	20.0	10.0	1.0		5.0	2.0	0.1

(3.75)



(a) $\mu_1 < \mu_2$ **(b)** $\mu_1 > \mu_2$

Figure 3.14: Plots for §3.5.2. Case (a) has $\mu_1 < \mu_2$; case (b) has $\mu_1 > \mu_2$. Both vary λ_2 and show the expected reward rate per unit time γg for the four policies in §3.5.2.



(a) varying μ_1 **(b)** varying μ_2

Figure 3.15: Plots for §3.5.2. Case (a) varies μ_1 and case (b) varies μ_2. Both cases show the expected reward rate per unit time γg for the four policies in §3.5.2.

Note preemptions are inexpensive ($f_p = 0.1$) and hence optimal preemption control outperforms optimal admission control, but both controls are required to achieve optimal performance. It is interesting that all four controls converge as μ_1 and μ_2 increase. In case (a) $((b))$ the convergence as μ_1 (μ_2) increases is explained by the decreasing fraction of class 1 (2) offered load $\alpha_1 = \lambda_1/\mu_1 \to 0$ ($\alpha_2 = \lambda_2/\mu_2 \to 0$), and in both cases the choice of control is irrelevant.

3.6 Conclusion

The design and performance analysis of joint admission control and preemption control mechanisms for a multi-class loss network is an important and largely unsolved problem. This paper presents results for the simplest non-trivial case of a single link serving two call classes. Further, our results

are restricted to a restriction of the preemption control design space (if always preempting when full then not preempting when not full is optimal), and a comparison of the performance of admission and preemption control. Our proof technique is an application of the policy improvement theorems from stochastic dynamic programming, and the associated induction technique works when the two policies being compared essentially differ at two states. This limitation is restrictive in light of the counter-example where we showed preemption performance is not monotone increasing as we remove preemptions at non-full states. Our numerical investigations have motivated our Conjectures 1 and 2 that it suffices to consider threshold policies that preempt only at full states, but our current proof techniques appear inadequate to prove them. Our numerical results evaluate the four policies of no control, optimal preemption, optimal admission, and optimal joint control, and illustrate a variety of qualitatively different sensitivities of performance to control. Analytical results on the joint use of preemption and admission control are of both practical and theoretical interest, and we hope this paper inspires others to work on this hard problem.

Chapter 4: Conclusion

This dissertation presents results on the performance analysis and policy design of preemption in the context of multi-class loss networks. The performance analysis in Chapter 2 is the first for a multi-class *preemptive* loss network, although blocking probabilities for multi-class *non-preemptive* loss networks (and their associated heavy traffic limits) have been known for a long time. We have studied the performance of a two parallel link network with transfers servicing K classes with preemptive priority. The performance metrics we study include the rate that arriving calls preempt lower priority active calls, and the rate that active calls are preempted by higher priority arriving calls. Contributions include:

- Exact expressions for each of K preemptive classes with homogeneous service rates on a two parallel link network in terms of Erlang-B blocking probabilities.
- Asymptotic approximation of these expressions appropriate for the "many small users" regime (large arrival rates and large link capacities).
- An efficiently computable approximation for preemption rates for each of K preemptive classes with heterogeneous service rates under a specific time-scale separation regime, obtained via associated lumpable, nearly completely decomposable (NCD) Markov chains.

The policy analysis of preemption and admission control policies for a two class loss link in Chapter 3 is the first work comparing the performance of preemption and admission control. Contributions include:

- If preemption is always employed when the link is full, we obtain sufficient conditions under which it is better/worse to preempt when the link is not full.
- A sufficient condition for the performance of a certain natural preemption policy (without admission control) to exceed that of optimal admission control.

There are many directions for extending this work, besides what has been mentioned in Chapter 2 and 3, such as the relaxation of the assumptions of the mathematical models. Pressing unanswered questions include:

1. For Chapter 2 our investigation is restricted to the special case of a two parallel link network. Other topologies are also of interest and merit investigation. Generic network topologies do not appear to be analytically tractable.

2. For Chapter 3, one extension is to compare the joint use of preemption and rate adaptation controls. Our preliminary numerical results [7] motivate this investigation.

Bibliography

[1] F. Kelly, "Loss Networks," *Annals of Applied Probability*, vol. 1, pp. 319–378, 1991.

[2] Z. Zhao, S. Weber, and J. C. de Oliveira, "Preemption rates for a parallel link loss network," *Performance Evaluation*, vol. 66, pp. 21–46, 2009.

[3] Z. Zhao, B. Willman, S. Weber, and J. de Oliveira, "Performance Analysis of a Parallel Link Network with Preemption," in *The 40th Annual Conferences on Information Sciences and Systems (CISS'06)*, Mar. 2006.

[4] Z. Zhao, S. Weber, and J. de Oliveira, "Heavy traffic analysis of a multi-class loss link with preemption," in *The 41st Annual Conferences on Information Sciences and Systems (CISS'07)*, Mar. 2007.

[5] Z. Zhao and S. Weber, "Preemption and admission control of a two class loss link," *Performance Evaluation*, 2011. submitted.

[6] Z. Zhao, S. Weber, and J. de Oliveira, "Admission control and preemption policy design of multi-class computer networks," in *Proceedings of the 44th Conference on Information Sciences and Systems (CISS)*, (Princeton, NJ), March 2010.

[7] S. Weber, J. C. de Oliveira, B. Willman, and Z. Zhao, "Combined Preemption and Adaptation in Next Generation Multiservice Networks," in *IEEE ICC 2006*, Jun. 2006.

[8] Z. Zhao, J. B. Goldberg, S. Weber, and J. C. de Oliveira, "Bandwidth constraint models: a performance study with adaptation and preemption on link failures in MPLS networks," *Computer Networks Journal (Elsevier)*, 2011. in preparation.

[9] J. de Oliveira, J. Vasseur, L. Chen, and C. Scoglio, "Label switched path (LSP) preemption policies for MPLS traffic engineering," Tech. Rep. RFC 4829, Internet Engineering Task Force (IETF), April 2007. `http://www.rfc-editor.org/rfc/rfc4829.txt`.

[10] M. Markowski and A. Sethi, "Fully distributed wireless transmission of heterogeneous real-time data," in *Proceedings of the IEEE Vehicular Technology Conference (VTC)*, vol. 2, (Ottawa, Ontario, Canada), pp. 1439–1442, May 1998.

[11] S. Haykin, "Cognitive radio: Brain-empowered wireless communications," *IEEE Journal on Selected Areas in Communications*, vol. 23, pp. 201–220, February 2005.

[12] J. Garay and I. Gopal, "Call preemption in communication networks," in *Proceedings of IEEE INFOCOM*, vol. 3, (Florence, Italy), pp. 1043–1050, May 1992.

[13] M. Peyravian and A. Kshemkalyani, "Connection preemption: issues, algorithms, and a simulation study," in *Proceedings of IEEE INFOCOM*, vol. 1, (Kobe, Japan), pp. 143–151, April 1997.

[14] J. de Oliveira, C. Scoglio, I. Akyildiz, and G. Uhl, "New preemption policies for DiffServ-aware traffic engineering to minimize rerouting in MPLS networks," *IEEE/ACM Transactions on Networking*, vol. 12, pp. 733–745, August 2004.

[15] J. Sung-eok, R. Abler, and A. Goulart, "The optimal connection preemption algorithm in a multi-class network," in *Proceedings of IEEE International Conference on Communications (ICC)*, vol. 4, (New York, NY), pp. 2294–2298, April 2002.

[16] S. Tong, D. Hoang, and O. Yang, "Bandwidth allocation and preemption for supporting differentiated-service-aware traffic engineering in multi-service networks," in *Proceedings of IEEE International Conference on Communications (ICC)*, vol. 2, (New York, NY), pp. 1305–1309, April 2002.

[17] V. Stanisic and M. Devetsikiotis, "A dynamic study of providing quality of service using preemption policies with random selection," in *Proceedings of IEEE International Conference on Communications (ICC)*, vol. 3, (Anchorage, AK), pp. 1543–1546, May 2003.

[18] F. Blanchy, L. Melon, and G. Leduc, "Routing in a MPLS network featuring preemption mechanisms," in *Proceedings of the International Conference on Telecommunications (ICT)*, vol. 1, (Tahiti, Papeete, French Polynesia), pp. 253–260, February 2003.

[19] K. Yu, L. Zhang, and H. Zhang, "A preemption-aware path selection algorithm for DiffServ/MPLS networks," in *Proceedings of the IEEE Workshop on IP Operations and Management*, (Beijing, China), pp. 129–133, October 2004.

[20] R. Vieira and P. Guardieiro, "A proposal and evaluation of a LSP preemption policy implemented with fuzzy logic and genetic algorithms in a DiffServ/MPLS test-bed," in *Proceedings of International Conference on Communications, Circuits and Systems*, vol. 1, (Hong Kong, China), pp. 109–114, May 2005.

[21] K. W. Ross, *Multiservice loss models for broadband communication networks*. Springer Verlag, 1995.

[22] R. Srikant, *The mathematics of Internet congestion control*. Boston, MA: Birkhäuser, 2003.

[23] H. White and L. Christie, "Queuing with preemptive priorities or with breakdown," *Operations Research*, vol. 6, pp. 79–95, Jan.–Feb. 1958.

[24] D. Miller, "Computation of steady-state probabilities for $M/M/1$ priority queues," *Operations Research*, vol. 29, pp. 945–959, Sep.–Oct. 1981.

[25] J. Buzen and A. Bondi, "The response times of priority classes under preemptive resume in $M/M/m$ queues," *Operations Research*, vol. 31, pp. 456–465, May–June 1983.

[26] B. Ngo and H. Lee, "Analysis of a pre-emptive priority $M/M/c$ model with two types of customers and restriction," *Electronics Letters*, vol. 26, pp. 1190–1192, July 1990.

[27] Y. Cho and C. Un, "Analysis of the $M/G/1$ queue under a combined preemptive/nonpreemptive priority discipline," *IEEE Transactions on Communications*, vol. 41, pp. 132–141, January 1993.

[28] W. Helly, "Two doctrines for the handling of two-priority traffic by a group of N servers," *Operations Research*, vol. 10, pp. 268–269, Mar.–Apr. 1962.

[29] P. Burke, "Priority traffic with at most one queueing class," *Operations Research*, vol. 10, pp. 567–569, Jul.–Aug. 1962.

[30] D. Calabrese, M. Fischer, B. Hoiem, and E. Kaiser, "Modeling a voice network with preemption," *IEEE Transactions on Communications*, vol. 28, pp. 22–27, January 1980.

[31] T. Dayar and W. Stewart, "Quasi lumpability, lower-bounding coupling matrices, and nearly completely decomposable Markov chains," *SIAM Journal of Matrix Analysis and its Applications*, vol. 18, pp. 482–498, April 1997.

[32] F. Ball and G. Yeo, "Lumpability and marginalisability for continuous-time Markov chains," *Journal of Applied Probability*, vol. 30, pp. 518–528, September 1993.

[33] J. Kemeny and J. Snell, *Finite Markov Chains*. New York, NY: Springer-Verlag, 1983.

[34] B. Schehrer, "On the calculation of overflow systems with a finite number of sources and full available groups," *IEEE Transactions on Communications*, vol. 26, pp. 75–82, Jan. 1978.

[35] D. Mitra and J. Morrison, "Erlang capacity and uniform approximations for shared unbuffered resources," *IEEE/ACM Transactions on Networking*, vol. 2, pp. 558–570, December 1994.

[36] G. Yin and Q. Zhang, *Discrete-time Markov chains: two-time-scale methods and applications*. New York, NY: Springer, 2005.

[37] M. Reiman and J. Schmitt, "Performance models of multirate traffic in various network implementations," in *The Fundamental Role of Teletraffic in the Evolution of Telecommunication Networks*, pp. 1217–1228, Elsevier, 1994.

[38] C. Meyer, "Stochastic complementation, uncoupling Markov chains, and the theory of nearly reducible systems," *SIAM Review*, vol. 31, pp. 240–272, June 1989.

[39] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss, *RFC2475 - An Architecture for Differentiated Services*. IETF Network Working Group, Dec. 1998.

[40] J. S. Kaufman, "Blocking in a shared resource environment," *IEEE Transactions on Communications*, vol. COM-29, Oct. 1981.

[41] K. W. Ross and D. H. K. Tsang, "The stochastic knapsack problem," *IEEE Transactions on Communications*, vol. 37, Jul. 1989.

[42] J. M. Aein and G. S. Kosevych, "Satellite capacity allocation," *Proc. IEEE*, vol. 65, pp. 332–342, Mar. 1977.

[43] J. M. Aein, "A multi-user-class blocked-calls-cleared demand access model," *IEEE Trans. Commun.*, vol. COM-26, Mar. 1978.

[44] G. J. Foschini, B. Gopinath, and J. F. Hayes, "Optium allocation of servers to two types of compting customers," *IEEE Transactions on Communications*, vol. COM-29, pp. 1051–1055, Jul. 1981.

[45] E. W. Knightly and N. B. Shroff, "Admission control for statistical QoS : theory and practice," *IEEE Network*, Mar. 1999.

[46] S. Jamin, S. J. Shenker, and P. B. Danzig, "Comparison of measurement-based admission control algorithms for controlled-load service," *INFOCOM' 97*, Apr. 1997.

[47] C. C. Wu and D. P. Bertsekas, "Admission control for wireless networks," tech. rep., MIT, Cambridge, MA, Dec. 2002.

[48] W. C. Y. Lee, "Smaller cells for greater performance," *IEEE Communications Magazine*, Nov. 1991.

[49] E. C. Posner and R. Guerin, "Traffic policies in cellular radio that minimize blocking of handoff calls," in *Proceedings of ITC*, vol. 11, (Kyoto, Japan), 1985.

[50] J. de Oliveira, C. Scoglio, I. Akyildiz, and G. Uhl, "A new preemption policy for DiffServ-aware traffic engineering to minimize rerouting," in *Proceedings of IEEE INFOCOM*, vol. 2, (New York, NY), pp. 695–704, June 2002.

[51] D. P. Bertsekas, *Dynamic programming and optimal control*. Athena Scientific, Belmont, Massachusetts, 2000.

[52] H. C. Tijms, *Stochastic modelling and analysis : a computational approach*. New York: Wiley, 1986.

[53] K. W. Ross and D. H. K. Tsang, "Optimal circuit access policies in an ISDN environment - A Markov decision approach," *IEEE Transactions on Communications*, vol. 37, pp. 934–939, Sep. 1989.

Appendix A: Appendix for Chapter 2

A.1 Proof of Theorem 1

Proof. Although Theorem 1 asserts that $\{\mathbf{n}(t)\}$ is a Markov process when the service rates are homogeneous, we will in fact show that the process is Markovian for the more general case of heterogeneous service rates. We first show that the time spent in each state is exponentially distributed with a rate that depends only upon the current state. Let the current state be $\mathbf{n}$, where we first assume that $n_1 + \cdots + n_K < c$. Let $A_k(s,t)$ denote the number of arrivals of each class k over $(s,t]$. Let the durations of the various active calls of class k be denoted $X_{k,1}, \ldots, X_{k,n_k}$. By the memoryless property, each call duration probabilistically restarts upon entering each new state. Let $T_{\mathbf{n}}$ be the random variable giving the time spent in state $\mathbf{n}$. Then

$$\begin{aligned} \mathbb{P}(T(\mathbf{n}) > t) &= \mathbb{P}(A_k(0,t) = 0,\ k = 1, \ldots, K, \text{ and} \\ &\qquad X_{k,i} > t,\ i = 1, \ldots, n_k,\ k = 1, \ldots, K), \\ &= \exp\left\{ -\left(\sum_{k=1}^{K} \lambda_k + n_k \mu_k \right) t \right\}, \end{aligned} \tag{A.1}$$

where the second equality holds by the independence and stationarity of the arrival processes and the independence of the call durations. Define $\tau(\mathbf{n}) = \sum_{k=1}^{K} \lambda_k + n_k \mu_k$; this allows $T(\mathbf{n}) \sim \text{Exp}(\tau(\mathbf{n}))$ for each such $\mathbf{n}$ with $n_1 + \cdots + n_K < c$.

Consider next the case where the state is $\mathbf{n}$ with $n_1 + \cdots + n_K = c$. For each class $k = 1, \ldots, K$, define $l(\mathbf{n}, k) = \max\{j > k : n_j > 0\}$ as the class of call that would be preempted from state $\mathbf{n}$ if a class k call arrived; leave $l(\mathbf{n}, k)$ undefined if no such j exists. Let $\mathcal{B}(\mathbf{n}) = \{k : l(\mathbf{n}, k) \text{ defined}\}$.

Then

$$\begin{aligned}\mathbb{P}(T(\mathbf{n}) > t) &= \mathbb{P}(A_k(0,t) = 0,\ k \in \mathcal{B}(\mathbf{n}),\ \text{and} \\ &\quad X_{k,i} > t,\ i = 1,\ldots,n_k,\ k = 1,\ldots,K), \\ &= \exp\left\{-\left(\sum_{k\in\mathcal{B}(\mathbf{n})}\lambda_k + \sum_{k=1}^{K} n_k\mu_k\right)t\right\}. \end{aligned} \tag{A.2}$$

Define $\tau'(\mathbf{n}) = \sum_{k\in\mathcal{B}(\mathbf{n})}\lambda_k + \sum_{k=1}^{K} n_k\mu_k$; this allows $T(\mathbf{n}) \sim \text{Exp}(\tau'(\mathbf{n}))$ for each $\mathbf{n}$ with $n_1 + \cdots + n_K = c$. We next show that the embedded jump chain is well-defined at each state. Let $p(\mathbf{n}, \mathbf{n}')$ be the probability of jumping from $\mathbf{n}$ to $\mathbf{n}'$ at a jump time. The results below exploit the well-known property that if $Z_1, \ldots, Z_n$ are independent exponential random variables with rates $\nu_1, \ldots, \nu_n$ then

$$\mathbb{P}(m = \arg\min\{Z_1, \ldots, Z_n\}) = \frac{\nu_m}{\nu_1 + \cdots + \nu_n}, \tag{A.3}$$

for each $m = 1, \ldots, n$. Applying this fact to a state in $\mathbf{n}$ with $n_1 + \cdots + n_K < c$ yields:

$$\begin{aligned} p(\mathbf{n}, \mathbf{n} + \mathbf{e}_k) &= \frac{\lambda_k}{\tau(\mathbf{n})},\ k = 1, \ldots, K \\ p(\mathbf{n}, \mathbf{n} - \mathbf{e}_k) &= \frac{n_k\mu_k\mathbf{1}_{n_k>0}}{\tau(\mathbf{n})},\ k = 1, \ldots, K, \\ p(\mathbf{n}, \mathbf{n}') &= 0, \quad \text{else.} \end{aligned} \tag{A.4}$$

Similarly, for a state $\mathbf{n}$ with $n_1 + \cdots + n_K = c$:

$$\begin{aligned} p(\mathbf{n}, \mathbf{n} + \mathbf{e}_k - \mathbf{e}_l) &= \frac{\lambda_k}{\tau'(\mathbf{n})},\ k \in \mathcal{B}(\mathbf{n}) \\ p(\mathbf{n}, \mathbf{n} - \mathbf{e}_k) &= \frac{n_k\mu_k\mathbf{1}_{n_k>0}}{\tau'(\mathbf{n})},\ k = 1, \ldots, K, \\ p(\mathbf{n}, \mathbf{n}') &= 0, \quad \text{else.} \end{aligned} \tag{A.5}$$

It is easily verified that $\sum_{\mathbf{n}'} p(\mathbf{n}, \mathbf{n}') = 1$ for each $\mathbf{n} \in \mathcal{S}$. This establishes the process $\{\mathbf{n}(t)\}$ is a Markov chain with rate matrix $\mathbf{Q}$ with entries $q(\mathbf{n}, \mathbf{n}') = p(\mathbf{n}, \mathbf{n}')\tau(\mathbf{n})$ for each $\mathbf{n} \neq \mathbf{n}'$. □

A.2 Proof of Theorem 4

Proof. Theorem 4 asserts that $\{\mathbf{n}(t)\}$ with homogeneous service rates is lumpable under the split k aggregate occupancy partition; we will show that lumpability holds iff the service rates are homogeneous. Fix k. Consider a pair $(N_k, \bar{N}_k)$ with $N_k + \bar{N}_k < c$. Then the only allowed transitions are

$$\begin{aligned}
(N_k, \bar{N}_k) \to \quad & (N_k - 1, \bar{N}_k), && N_k > 0 \\
(N_k, \bar{N}_k) \to \quad & (N_k, \bar{N}_k - 1), && \bar{N}_k > 0 \\
(N_k, \bar{N}_k) \to \quad & (N_k + 1, \bar{N}_k), && N_k + \bar{N}_k < c \\
(N_k, \bar{N}_k) \to \quad & (N_k, \bar{N}_k + 1), && N_k + \bar{N}_k < c \\
(N_k, \bar{N}_k) \to \quad & (N_k + 1, \bar{N}_k - 1), && N_k + \bar{N}_k = c \text{ and } \bar{N}_k > 0
\end{aligned} \tag{A.6}$$

These correspond to i) a departure of a call of class $1, \ldots, k$, ii) a departure of a call of class $k+1, \ldots, K$, iii) an arrival of a call of class $1, \ldots, k$ to a link that is not full, iv)) an arrival of a call of class $k+1, \ldots, K$ to a link that is not full, and v) an arrival of a call of class $1, \ldots, k$ to a link that is full, causing a preemption of a call of class $k+1, \ldots, K$. Consider some $\mathbf{n} \in \mathcal{S}_n^{\text{saop},k}$; the transition rate up and down is:

$$\begin{aligned}
\sum_{\mathbf{n}' \in \mathcal{S}^{\text{saop},k}_{N_k - 1, \bar{N}_k}} q(\mathbf{n}, \mathbf{n}') &= (n_1\mu_1 + \cdots n_k\mu_k)\, \mathbf{1}_{N_k > 0}, \\
\sum_{\mathbf{n}' \in \mathcal{S}^{\text{saop},k}_{N_k, \bar{N}_k - 1}} q(\mathbf{n}, \mathbf{n}') &= (n_{k+1}\mu_{k+1} + \cdots + n_K\mu_K)\, \mathbf{1}_{\bar{N}_k > 0}, \\
\sum_{\mathbf{n}' \in \mathcal{S}^{\text{saop},k}_{N_k + 1, \bar{N}_k}} q(\mathbf{n}, \mathbf{n}') &= \Lambda_k \mathbf{1}_{N_k + \bar{N}_k < c}, \\
\sum_{\mathbf{n}' \in \mathcal{S}^{\text{saop},k}_{N_k, \bar{N}_k + 1}} q(\mathbf{n}, \mathbf{n}') &= (\Lambda_K - \Lambda_k) \mathbf{1}_{N_k + \bar{N}_k < c}, \\
\sum_{\mathbf{n}' \in \mathcal{S}^{\text{saop},k}_{N_k + 1, \bar{N}_k - 1}} q(\mathbf{n}, \mathbf{n}') &= \Lambda_k \mathbf{1}_{N_k + \bar{N}_k = c, \bar{N}_k > 0}.
\end{aligned} \tag{A.7}$$

Clearly the process is lumpable and so Markovian under the partition iff the service rates are homogeneous. □

Appendix B: Appendix for Chapter 3

B.1 Proof of Lemma 1

Recall $v_\pi(n,i)$, defined in (3.20), with initial condition $v_\pi(n,0) = 0$ for all $n \in \mathcal{N}$, is the expected cumulative reward over the remaining i stages starting from state n, for $i > 0$ and $n \in \mathcal{N}$. We omit the dependence on π in the remainder of the proof which is assumed fixed. We rewrite $v(n,i)$ as

$$\gamma v(n,i) = r(n) + \sum_{n' \in \Gamma(n)} (R(n,n') + v(n',i-1))q(n,n') + v(n,i-1)(\gamma - Q(n)) \tag{B.1}$$

for $n \in \mathcal{N}$ and $i > 0$, where $Q(n) \equiv \sum_{n' \in \Gamma(n)} q(n,n') \leq \gamma$ is the aggregate transition rate out of state n in the CTMC. Define

$$\Delta(n,i) \equiv \gamma(v(n,i) - v(n-e_2,i)),\ n \in \mathcal{N}, i > 0. \tag{B.2}$$

We will establish $\Delta(n,i) \geq -\gamma(\rho(e_2) + f_p)$ for each $n \in \mathcal{N}$ and $i > 0$. This suffices to prove the desired result by letting $i \to \infty$ and using (3.23). Let $m \equiv n - e_2$. Then:

$$\begin{aligned}
\Delta(n,i) &= r(n) - r(m) && \text{(B.3)}\\
&+ \sum_{n' \in \Gamma(n)} R(n,n')q(n,n') - \sum_{m' \in \Gamma(m)} R(m,m')q(m,m') && \text{(B.4)}\\
&+ \sum_{n' \in \Gamma(n)} v(n',i-1)q(n,n') - \sum_{m' \in \Gamma(m)} v(m',i-1)q(m,m') && \text{(B.5)}\\
&+ v(n,i-1)(\gamma - Q(n)) - v(m,i-1)(\gamma - Q(m)). && \text{(B.6)}
\end{aligned}$$

There are 12 possibilities for the transitions out of the pair of states (n,m), illustrated in Fig. B.1. These are found by taking all 16 possibilities for the following 4 binary options and pruning 4

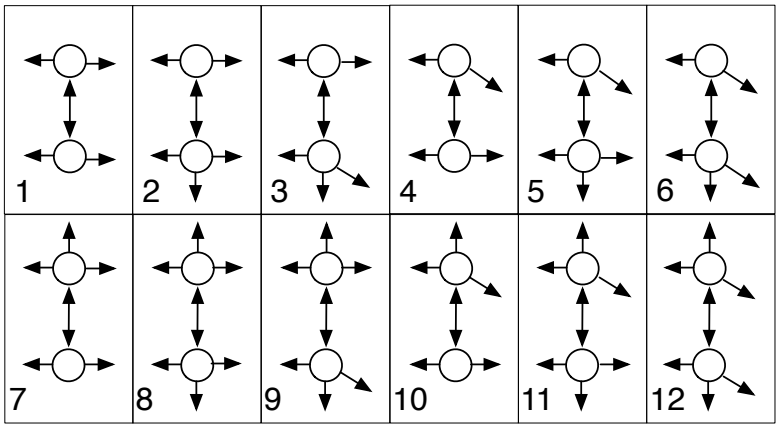


Figure B.1: The twelve possibilities for transitions out from two states $n, m = n - e_2 \in \mathcal{N}$ with $n_1 > 0, n_2 > 0$ under $\pi^p \in \Pi^p_{\text{awf}}$.

impossible cases:

$$\begin{array}{lllll} (a) & \pi^a(n) = 0 & \text{or} & \pi^a(n) = 1 \\ (b) & \pi^p(n) = 0 & \text{or} & \pi^p(n) = 1 \\ (c) & \pi^p(m) = 0 & \text{or} & \pi^p(m) = 1 \\ (d) & m_2 = 0 & \text{or} & m_2 > 0 \end{array}$$

The 4 impossible cases are for $m_2 = 0$ and $\pi^p(m) = 1$, i.e., m has zero class 2 calls but still allows preemption. Writing out $\Delta(n, i) - (r(n) - r(m))$ for the 12 cases gives Table B.1, after simplification, where we have used the shorthand notations $j \equiv i - 1$, $\lambda \equiv \lambda_1 + \lambda_2$, and $v_j(n) \equiv v_\pi(n, j)$. The last five columns in Table B.1 have the common characteristic that each entry is a difference of expected cumulative reward ($v_j(\cdot)$) values between two states where the two states have the same number of class 1 calls and the first state has one or more additional class 2 calls than the second state. We illustrate these manipulations for case #12 (the remaining 11 cases are analogous).

Table B.1: $\Delta(n,i) - (r(n) - r(m))$ in (B.3) for the twelve possibilities in Fig. B.1.

#	(B.4)	(B.5) (class 1 arrivals)	(B.5) (class 2 arrivals)
1	$\rho(-e_2)\mu_2 - \rho(e_2)\lambda_2$	$\lambda_1(v_j(n+e_1) - v_j(m+e_1))$	
2	$\rho(-e_2)\mu_2 - \rho(e_2)\lambda_2$	$\lambda_1(v_j(n+e_1) - v_j(m+e_1))$	
3	$\rho(-e_2)\mu_2 - \rho(e_2)\lambda_2 + (\rho(e_2)+f_p)\lambda_1$	$\lambda_1(v_j(n+e_1) - v_j(m+e_1-e_2))$	
4	$\rho(-e_2)\mu_2 - \rho(e_2)\lambda_2 - (\rho(e_2)+f_p)\lambda_1$		
5	$\rho(-e_2)\mu_2 - \rho(e_2)\lambda_2 - (\rho(e_2)+f_p)\lambda_1$		
6	$\rho(-e_2)\mu_2 - \rho(e_2)\lambda_2$	$\lambda_1(v_j(m+e_1) - v_j(m+e_1-e_2))$	
7	$\rho(-e_2)\mu_2$	$\lambda_1(v_j(n+e_1) - v_j(m+e_1))$	$\lambda_2(v_j(n+e_2) - v_j(n))$
8	$\rho(-e_2)\mu_2$	$\lambda_1(v_j(n+e_1) - v_j(m+e_1))$	$\lambda_2(v_j(n+e_2) - v_j(n))$
9	$\rho(-e_2)\mu_2 + (\rho(e_2)+f_p)\lambda_1$	$\lambda_1(v_j(n+e_1) - v_j(m+e_1-e_2))$	$\lambda_2(v_j(n+e_2) - v_j(n))$
10	$\rho(-e_2)\mu_2 - (\rho(e_2)+f_p)\lambda_1$		$\lambda_2(v_j(n+e_2) - v_j(n))$
11	$\rho(-e_2)\mu_2 - (\rho(e_2)+f_p)\lambda_1$		$\lambda_2(v_j(n+e_2) - v_j(n))$
12	$\rho(-e_2)\mu_2$	$\lambda_1(v_j(m+e_1) - v_j(m+e_1-e_2))$	$\lambda_2(v_j(n+e_2) - v_j(n))$

#	(B.5) (class 1 departures)	(B.5) (class 2 departures)	(B.6)
1	$n_1\mu_1(v_j(n-e_1) - v_j(m-e_1))$		$(\gamma - (\lambda + n_1\mu_1 + \mu_2))(v_j(n) - v_j(m))$
2	$n_1\mu_1(v_j(n-e_1) - v_j(m-e_1))$	$(n_2-1)\mu_2(v_j(m) - v_j(m-e_2))$	$(\gamma - (\lambda + n_1\mu_1 + n_2\mu_2))(v_j(n) - v_j(m))$
3	$n_1\mu_1(v_j(n-e_1) - v_j(m-e_1))$	$(n_2-1)\mu_2(v_j(m) - v_j(m-e_2))$	$(\gamma - (\lambda + n_1\mu_1 + n_2\mu_2))(v_j(n) - v_j(m))$
4	$n_1\mu_1(v_j(n-e_1) - v_j(m-e_1))$		$(\gamma - (\lambda + n_1\mu_1 + \mu_2))(v_j(n) - v_j(m))$
5	$n_1\mu_1(v_j(n-e_1) - v_j(m-e_1))$	$(n_2-1)\mu_2(v_j(m) - v_j(m-e_2))$	$(\gamma - (\lambda + n_1\mu_1 + n_2\mu_2))(v_j(n) - v_j(m))$
6	$n_1\mu_1(v_j(n-e_1) - v_j(m-e_1))$	$(n_2-1)\mu_2(v_j(m) - v_j(m-e_2))$	$(\gamma - (\lambda + n_1\mu_1 + n_2\mu_2))(v_j(n) - v_j(m))$
7	$n_1\mu_1(v_j(n-e_1) - v_j(m-e_1))$		$(\gamma - (\lambda + n_1\mu_1 + \mu_2))(v_j(n) - v_j(m))$
8	$n_1\mu_1(v_j(n-e_1) - v_j(m-e_1))$	$(n_2-1)\mu_2(v_j(m) - v_j(m-e_2))$	$(\gamma - (\lambda + n_1\mu_1 + n_2\mu_2))(v_j(n) - v_j(m))$
9	$n_1\mu_1(v_j(n-e_1) - v_j(m-e_1))$	$(n_2-1)\mu_2(v_j(m) - v_j(m-e_2))$	$(\gamma - (\lambda + n_1\mu_1 + n_2\mu_2))(v_j(n) - v_j(m))$
10	$n_1\mu_1(v_j(n-e_1) - v_j(m-e_1))$		$(\gamma - (\lambda + n_1\mu_1 + \mu_2))(v_j(n) - v_j(m))$
11	$n_1\mu_1(v_j(n-e_1) - v_j(m-e_1))$	$(n_2-1)\mu_2(v_j(m) - v_j(m-e_2))$	$(\gamma - (\lambda + n_1\mu_1 + n_2\mu_2))(v_j(n) - v_j(m))$
12	$n_1\mu_1(v_j(n-e_1) - v_j(m-e_1))$	$(n_2-1)\mu_2(v_j(m) - v_j(m-e_2))$	$(\gamma - (\lambda + n_1\mu_1 + n_2\mu_2))(v_j(n) - v_j(m))$

$$\Delta(n,i) - (r(n) - r(m)) =$$

$$\begin{aligned}
&(\lambda_1\rho(e_1 - e_2) + \lambda_2\rho(e_2) + n_1\mu_1\rho(-e_1) + n_2\mu_2\rho(-e_2)) \\
&-(\lambda_1\rho(e_1 - e_2) + \lambda_2\rho(e_2) + n_1\mu_1\rho(-e_1) + (n_2 - 1)\mu_2\rho(-e_2)) \\
&+(\lambda_1 v_j(m + e_1) + \lambda_2 v_j(n + e_2) + n_1\mu_1 v_j(n - e_1) + n_2\mu_2 v_j(m)) \\
&-(\lambda_1 v_j(m + e_1 - e_2) + \lambda_2 v_j(m + e_2) + n_1\mu_1 v_j(m - e_1) + (n_2 - 1)\mu_2 v_j(m - e_2)) \\
&+(\gamma - (\lambda + n_1\mu_1 + n_2\mu_2))v_j(n) - (\gamma - (\lambda + n_1\mu_1 + (n_2 - 1)\mu_2))v_j(m)
\end{aligned} \tag{B.7}$$

which may be rearranged as

$$\begin{aligned}\Delta(n,i)-(r(n)-r(m)) &= \rho(-e_2)\mu_2 \\ &\quad +\lambda_1(v_j(m+e_1)-v_j(m+e_1-e_2)) \\ &\quad +\lambda_2(v_j(n+e_2)-v_j(m+e_2)) \\ &\quad +n_1\mu_1(v_j(n-e_1)-v_j(m-e_1)) \\ &\quad +(n_2-1)\mu_2(v_j(m)-v_j(m-e_2)) \\ &\quad +(\gamma-(\lambda+n_1\mu_1+n_2\mu_2))(v_j(n)-v_j(m)) \qquad \text{(B.8)}\end{aligned}$$

The six terms in the six rows in (B.8) are the six column entries in Table B.1 for row #12.

We now prove $\Delta(n,i) \geq -\gamma(\rho(e_2)+f_p)$ for all $n \in \mathcal{N}$ with $n_2 > 0$ and all $i \geq 0$ by induction in the time index i. Fix the state n.

- When $f_p \geq -\rho(e_2)$, the smallest value in the first column in the top half of Table B.1 (corresponding to (B.4)) is for cases #4, #5. Requiring this value to exceed $-\gamma(\rho(e_2)+f_p)$ ensures all other entries in the first column in the top half of the table will also exceed that value. This requirement

$$r(n)-r(m)+\rho(-e_2)\mu_2-\rho(e_2)\lambda_2-(\rho(e_2)+f_p)\lambda_1 \geq -\gamma(\rho(e_2)+f_p) \qquad \text{(B.9)}$$

  is easily seen to be equivalent to

$$f_p \geq \begin{cases} A_2, & \text{if} \quad \rho(e_2)\lambda_2 > \rho(-e_2)\mu_2 + r(n) - r(m) \\ -\rho(e_2), & \text{if} \quad \rho(e_2)\lambda_2 \leq \rho(-e_2)\mu_2 + r(n) - r(m) \end{cases} \qquad \text{(B.10)}$$

- When $f_p < -\rho(e_2)$, the smallest value in the first column in the top half of Table B.1 (corresponding to (B.4)) is for cases #3. Requiring this value to exceed $-\gamma(\rho(e_2)+f_p)$ ensures all other entries in the first column in the top half of the table will also exceed that value. This

requirement

$$r(n) - r(m) + \rho(-e_2)\mu_2 - \rho(e_2)\lambda_2 + (\rho(e_2) + f_p)\lambda_1 \geq -\gamma(\rho(e_2) + f_p) \tag{B.11}$$

is impossible for $\rho(e_2)\lambda_2 > \rho(-e_2)\mu_2 + r(n) - r(m)$, and otherwise equivalent to

$$f_p \geq A_1, \text{ if } \quad \rho(e_2)\lambda_2 \leq \rho(-e_2)\mu_2 + r(n) - r(m). \tag{B.12}$$

Note (B.10) and (B.12) are the sufficient condition (3.29). Consider the base case of the induction, $i = 0$, where by definition we have $v_\pi(n, 0) = v_0(n) = 0$ and thus all entries in the remaining five columns of Table B.1 are zero. It follows that (3.29) is sufficient to establish the base case. Suppose now that the induction hypothesis holds for all times $0, \ldots, i - 1 = j$, we must show this ensures it holds for i. Assumption 4 guarantees (B.3) is nonnegative, (3.29) guarantees (B.4) exceeds $-\gamma(\rho(e_2) + f_p)$, the induction hypothesis and Table B.1 guarantees (B.5) and (B.6) are both nonnegative, and thus $\Delta(n, i) \geq -\gamma(\rho(e_2) + f_p)$.

B.2 Proof of Lemma 2

The proof is very similar to that of Lemma 1 and hence we only highlight the important differences. Recall $\Delta(n, i)$ defined in (B.3) and enumerated in Fig. B.1 and Table B.1. It now suffices to establish $\Delta(n, i) \leq -\gamma(\rho(e_2) + f_p)$ for each $n \in \mathcal{N}$ with $n_2 > 0$ and $i \geq 0$. We again use induction in the time index i. Fix the state n.

- When $f_p > -\rho(e_2)$, the largest value in the first column in the top half of Table B.1 (corresponding to (B.4)) is for case #9. Requiring this value plus $r(n) - r(m)$ not to exceed $-\gamma(\rho(e_2) + f_p)$ ensures all other entries in the first column in the top half of the table will also not exceed that value. This requirement

$$r(n) - r(m) + \rho(-e_2)\mu_2 + (\rho(e_2) + f_p)\lambda_1 \leq -\gamma(\rho(e_2) + f_p) \tag{B.13}$$

is impossible for $f_p > -\rho(e_2)$ since $r(n) - r(m)$, $\rho(-e_2)\mu_2$ and $f_p + \rho(e_2)$ are all nonnegative.

- When $f_p \leq -\rho(e_2)$, the largest value in the first column in the top half of Table B.1 (corresponding to (B.4)) is for cases #10 and #11. Requiring this value plus $r(n) - r(m)$ not to exceed $-\gamma(\rho(e_2) + f_p)$ ensures all other entries in the first column in the top half of the table will also not exceed that value. This requirement

$$r(n) - r(m) + \rho(-e_2)\mu_2 - (\rho(e_2) + f_p)\lambda_1 \leq -\gamma(\rho(e_2) + f_p) \tag{B.14}$$

is equivalent to (3.30).

The lemma follows by the same induction argument used at the end of the proof of Lemma 1.

B.3 Proof of Lemma 3

The proof is very similar to that of Lemma 1 and hence we only highlight the important differences. Recall the assumed policy $\bar{\pi} = (\pi^a_{\text{cs}}, \pi^p_{\text{awf}})$; we omit the dependence on $\bar{\pi}$ in the remainder of the proof. Recall $v(n, i)$ defined in (3.20), with initial condition $v(n, 0) = 0$ for all $n \in \mathcal{N}$. Recall $\Delta(n, i)$ defined in (B.2) and its expansion in (B.3) through (B.6): this expansion is valid for both Lemma 1 and this lemma. The lemma will be proved if we show $\Delta(n, i) \geq -\gamma\rho(e_2)$ for each $n \in \mathcal{N}$ with $n_2 > 0$ and $i \geq 0$. Let $m \equiv n - e_2$. There are twelve possibilities for the transitions out of the pair of states (n, m) with $n_2 > 0$, illustrated in Fig. B.2. Note the twelve possibilities for transitions out of (n, m) under Lemmas 1 and 3 are different: in Lemma 1 we know n has $n_1 > 0$ and $\pi^p \in \Pi^p_{\text{awf}}$ (always preempt when full and possibly when non-empty), whereas in Lemma 3 we know $\pi^p = \pi^p_{\text{awf}}$ (always and only preempt when full).

Writing out $\Delta(n, i) - (r(n) - r(m))$ for the 12 cases gives Table B.2, after simplification, where we have used the shorthand notations $j \equiv i - 1$, $\lambda \equiv \lambda_1 + \lambda_2$, and $v_j(n) \equiv v_{\bar{\pi}}(n, j)$ as was done in Table B.1. The last five columns in Table B.2 have the common characteristic that each entry is a difference of expected cumulative reward ($v_j(\cdot)$) values between two states where the two states have the same number of class 1 calls and the first state has one or more additional class 2 calls than the second state. We illustrate these manipulations for case #10 (the remaining 11 cases are

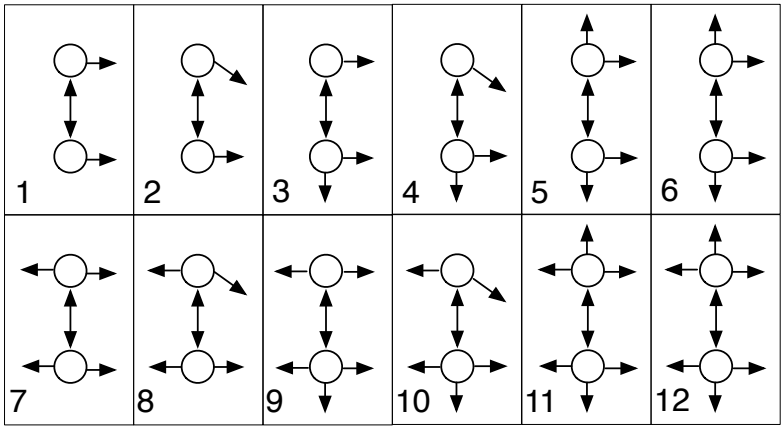


Figure B.2: The twelve possibilities for transitions out from two states $n, m = n - e_2 \in \mathcal{N}$ under $\bar{\pi} = (\pi^a_{\text{cs}}, \pi^p_{\text{awf}})$ with $n_2 > 0$.

analogous): $\Delta(n,i) - (r(n) - r(m)) =$

$$
\begin{aligned}
&(n_1\mu_1\rho(-e_1) + n_2\mu_2\rho(-e_2))) \\
&-(\lambda_1\rho(e_1) + \lambda_2\rho(e_2) + n_1\mu_1\rho(-e_1) + (n_2-1)\mu_2\rho(-e_2)) \\
&+(\lambda_1 \max\{v_j(m+e_1) + \rho(e_1-e_2), v_j(n)\} + n_1\mu_1 v_j(n-e_1) + n_2\mu_2 v_j(m)) \\
&-(\lambda_1 v_j(m+e_1) + \lambda_2 v_j(m+e_2) + n_1\mu_1 v_j(m-e_1) + (n_2-1)\mu_2 v_j(m-e_2)) \\
&+(\gamma - (\lambda_1 + n_1\mu_1 + n_2\mu_2))v_j(n) - (\gamma - (\lambda + n_1\mu_1 + (n_2-1)\mu_2))v_j(m)
\end{aligned}
\tag{B.15}
$$

Since $(\lambda_1 \max\{v_j(m+e_1) + \rho(e_1-e_2), v_j(n)\} \geq \lambda_1(v_j(m+e_1) + \rho(e_1-e_2))$, we obtain $\Delta(n,i) - (r(n) - r(m)) \geq$

$$
\begin{aligned}
&(\lambda_1\rho(e_1-e_2) + n_1\mu_1\rho(-e_1) + n_2\mu_2\rho(-e_2))) \\
&-(\lambda_1\rho(e_1) + \lambda_2\rho(e_2) + n_1\mu_1\rho(-e_1) + (n_2-1)\mu_2\rho(-e_2)) \\
&+(\lambda_1 v_j(m+e_1) + n_1\mu_1 v_j(n-e_1) + n_2\mu_2 v_j(m)) \\
&-(\lambda_1 v_j(m+e_1) + \lambda_2 v_j(n) + n_1\mu_1 v_j(m-e_1) + (n_2-1)\mu_2 v_j(m-e_2)) \\
&+(\gamma - (\lambda_1 + n_1\mu_1 + n_2\mu_2))v_j(n) - (\gamma - (\lambda + n_1\mu_1 + (n_2-1)\mu_2))v_j(m)
\end{aligned}
\tag{B.16}
$$

Table B.2: $\Delta(n,i) - (r(n) - r(m))$ in (B.3) for the twelve possibilities in Fig. B.2.

#	(B.4)	(B.5) (class 1 arrivals)	(B.5) (class 2 arrivals)
1	$\rho(-e_2)\mu_2 - \rho(e_2)\lambda_2$	$\lambda_1(v_j(n+e_1) - v_j(m+e_1))$	
2	$\rho(-e_2)\mu_2 - \rho(e_2)\lambda_2 - (\rho(e_2)+f_p)\lambda_1$		
3	$\rho(-e_2)\mu_2 - \rho(e_2)\lambda_2$	$\lambda_1(v_j(n+e_1) - v_j(m+e_1))$	
4	$\rho(-e_2)\mu_2 - \rho(e_2)\lambda_2 - (\rho(e_2)+f_p)\lambda_1$		
5	$\rho(-e_2)\mu_2$	$\lambda_1(v_j(n+e_1) - v_j(m+e_1))$	$\lambda_2(v_j(n+e_2) - v_j(n))$
6	$\rho(-e_2)\mu_2$	$\lambda_1(v_j(m+e_1) - v_j(m+e_1))$	$\lambda_2(v_j(n+e_2) - v_j(n))$
7	$\rho(-e_2)\mu_2 - \rho(e_2)\lambda_2$	$\lambda_1(v_j(n+e_1) - v_j(m+e_1))$	
8	$\rho(-e_2)\mu_2 - \rho(e_2)\lambda_2 - (\rho(e_2)+f_p)\lambda_1$		
9	$\rho(-e_2)\mu_2 - \rho(e_2)\lambda_2$	$\lambda_1(v_j(n+e_1) - v_j(m+e_1))$	
10	$\rho(-e_2)\mu_2 - \rho(e_2)\lambda_2 - (\rho(e_2)+f_p)\lambda_1$		
11	$\rho(-e_2)\mu_2$	$\lambda_1(v_j(n+e_1) - v_j(m+e_1))$	$\lambda_2(v_j(n+e_2) - v_j(n))$
12	$\rho(-e_2)\mu_2$	$\lambda_1(v_j(n+e_1) - v_j(m+e_1))$	$\lambda_2(v_j(n+e_2) - v_j(n))$

#	(B.5) (class 1 departures)	(B.5) (class 2 departures)	(B.6)
1			$(\gamma - (\lambda+\mu_2))(v_j(n) - v_j(m))$
2			$(\gamma - (\lambda+\mu_2))(v_j(n) - v_j(m))$
3		$(n_2-1)\mu_2(v_j(m) - v_j(m-e_2))$	$(\gamma - (\lambda+n_2\mu_2))(v_j(n) - v_j(m))$
4		$(n_2-1)\mu_2(v_j(m) - v_j(m-e_2))$	$(\gamma - (\lambda+n_2\mu_2))(v_j(n) - v_j(m))$
5			$(\gamma - (\lambda+\mu_2))(v_j(n) - v_j(m))$
6		$(n_2-1)\mu_2(v_j(m) - v_j(m-e_2))$	$(\gamma - (\lambda+n_2\mu_2))(v_j(n) - v_j(m))$
7	$n_1\mu_1(v_j(n-e_1) - v_j(m-e_1))$		$(\gamma - (\lambda+n_1\mu_1+\mu_2))(v_j(n) - v_j(m))$
8	$n_1\mu_1(v_j(n-e_1) - v_j(m-e_1))$		$(\gamma - (\lambda+n_1\mu_1+\mu_2))(v_j(n) - v_j(m))$
9	$n_1\mu_1(v_j(n-e_1) - v_j(m-e_1))$	$(n_2-1)\mu_2(v_j(m) - v_j(m-e_2))$	$(\gamma - (\lambda+n_1\mu_1+n_2\mu_2))(v_j(n) - v_j(m))$
10	$n_1\mu_1(v_j(n-e_1) - v_j(m-e_1))$	$(n_2-1)\mu_2(v_j(m) - v_j(m-e_2))$	$(\gamma - (\lambda+n_1\mu_1+n_2\mu_2))(v_j(n) - v_j(m))$
11	$n_1\mu_1(v_j(n-e_1) - v_j(m-e_1))$		$(\gamma - (\lambda+n_1\mu_1+\mu_2))(v_j(n) - v_j(m))$
12	$n_1\mu_1(v_j(n-e_1) - v_j(m-e_1))$	$(n_2-1)\mu_2(v_j(m) - v_j(m-e_2))$	$(\gamma - (\lambda+n_1\mu_1+n_2\mu_2))(v_j(n) - v_j(m))$

which may be rearranged as

$$\begin{aligned}\Delta(n,i) - (r(n) - r(m)) \quad \geq \quad & \rho(-e_2)\mu_2 - \rho(e_2)\lambda_2 - (\rho(e_2) + f_p)\lambda_1 \\ & + n_1\mu_1(v_j(n-e_1) - v_j(m-e_1)) \\ & + (n_2-1)\mu_2(v_j(m) - v_j(m-e_2)) \\ & + (\gamma - (\lambda + n_1\mu_1 + n_2\mu_2))(v_j(n) - v_j(m)) \qquad \text{(B.17)}\end{aligned}$$

The four terms in the four rows in (B.17) are the four column entries in Table B.2 for row #10.

We now prove $\Delta(n,i) \geq -\gamma\rho(e_2)$ for all $n \in \mathcal{N}$ with $n_2 > 0$ and all $i \geq 0$ by induction in the time index i. Fix the state n.

- When $f_p \geq -\rho(e_2)$, the smallest value in the first column in the top half of Table B.2 (corresponding to (B.4)) is for cases $\#2, \#4, \#8, \#10$. Requiring this value to exceed $-\gamma\rho(e_2)$ ensures all other entries in the first column in the top half of the table will also exceed that

value. This requirement

$$r(n) - r(m) + \rho(-e_2)\mu_2 - \rho(e_2)\lambda_2 - (\rho(e_2) + f_p)\lambda_1 \geq -\gamma\rho(e_2) \tag{B.18}$$

is equivalent to

$$f_p \leq \frac{1}{\lambda_1}\left(r(n) - r(m) + \rho(-e_2)\mu_2 + (\gamma - \lambda_1 - \lambda_2)\rho(e_2)\right). \tag{B.19}$$

- When $f_p < -\rho(e_2)$, the smallest value in the first column in the top half of Table B.1 (corresponding to (B.4)) is for cases $\#1, \#3, \#7, \#9$. Requiring this value to exceed $-\gamma\rho(e_2)$ ensures all other entries in the first column in the top half of the table will also exceed that value. This requirement

$$r(n) - r(m) + \rho(-e_2)\mu_2 - \rho(e_2)\lambda_2 \geq -\gamma\rho(e_2) \tag{B.20}$$

is always true since $r(n) - r(m) > 0, \rho(-e_2) \geq 0$ and $(\gamma - \lambda_2)\rho(e_2) \geq 0$.

The lemma follows by the same induction argument used at the end of the proof of Lemma 1.

B.4 Proof of Corollary 3

The condition (3.44) is sufficient for (3.52), which by the policy improvement theorem, ensures $g_{\bar{\pi}} \geq g_\pi$, as described in Prop. 3. We now seek an analogous condition to (3.44) that would ensure the opposite of (3.52), namely,

$$v_{\bar{\pi}}(n + e_2, i) - v_{\bar{\pi}}(n, i) \leq -\rho(e_2),\ n \in \mathcal{N}, i > 0. \tag{B.21}$$

If (B.21) holds, the policy improvement theorem ensures the desired conclusion, $g_\pi \geq g_{\bar{\pi}}$. Consequently, the objective is to use the proof technique in Prop. 3 to find the sufficient condition for (B.21). Table B.2 gives the values for $\Delta(n, i) - (r(n) - r(m))$. The difference is that for Prop. 3 we wanted the smallest value in the first column, now we want the largest value in the first column. There are two cases:

- $f_p \geq -\frac{\lambda_1+\lambda_2}{\lambda_1}\rho(e_2)$. In this case the largest value in the first column in the top half of Table B.2 (corresponding to (B.4)) is for cases $\#5, 6, 11, 12$. This requirement

$$r(n) - r(m) + \rho(-e_2)\mu_2 \leq -\gamma\rho(e_2) \tag{B.22}$$

  is impossible since $r(n) - r(m) \geq 0$, $\rho(-e_2) \geq 0$ and $-\gamma\rho(e_2) \leq 0$.

- $f_p < -\frac{\lambda_1+\lambda_2}{\lambda_1}\rho(e_2)$. In this case the largest value in the first column in the top half of Table B.2 (corresponding to (B.4)) is for cases $\#2, 4, 8, 10$. Requiring this value to be less than $-\gamma\rho(e_2)$ ensures all other entries in the first column in the top half of the table will also exceed that value. This requirement

$$r(n) - r(m) + \rho(-e_2)\mu_2 - \rho(e_2)\lambda_2 - (\rho(e_2) + f_p)\lambda_1 \leq -\gamma\rho(e_2) \tag{B.23}$$

  is impossible since $r(n) - r(m) \geq 0$, $\rho(-e_2) \geq 0$ and $-\rho(e_2)\lambda_2 - (\rho(e_2) + f_p)\lambda_1 > 0$.

We conclude that we cannot obtain a sufficient condition for (B.21) using this proof technique.

Vita

Zhen Zhao received the B.S. degree in Computer Science from South China University of Technology (SCUT) at Guangzhou, Guangdong, China in July 1998. His B.S. studies included half year industry experience as a part-time programmer at China-U.S. Networking Inc. and half year as an intern at IBM Research and Development center at SCUT. During 1998-2004, he worked in South China Financial IT Co. (Previous IT Dept. at the People's Bank of China (Central Bank)) as a software developer. He received the M.S. degree in Communication and Information System (from Dept. of Electrical Engineering) from Zhongshan University (National Sun Yat-sen University) at Guangzhou, Guangdong, China. In 2004, he enrolled in the doctoral program in the department of Electrical and Computer Engineering at Drexel University. He was the recipient of George Hill, Jr. Fellowship and other fellowships. He is also a member of IEEE and American Association for the Advancement of Science (AAAS). In 2010, he received his Master degree in Computer Engineering. He expects to receive his Ph.D. degree in Electrical Engineering from Drexel University in June, 2011.