Asymptotic Performance of Queue Length Based Network Control Policies

by

Krishna Prasanna Jagannathan

B.Tech., Indian Institute of Technology Madras (2004)
S.M., Massachusetts Institute of Technology (2006)

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2010

Author ....
Department of Electrical Engineering and Computer Science
August 26, 2010

Certified by ....
Eytan H. Modiano
Associate Professor
Thesis Supervisor

Accepted by ....
Terry P. Orlando
Chair, Department Committee on Graduate Students

Asymptotic Performance of Queue Length Based Network Control Policies

by

Krishna Prasanna Jagannathan

Submitted to the Department of Electrical Engineering and Computer Science
on August 26, 2010, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Electrical Engineering and Computer Science

Abstract

In a communication network, asymptotic quality of service metrics specify the probability that the delay or buffer occupancy becomes large. An understanding of these metrics is essential for providing worst-case delay guarantees, provisioning buffer sizes in networks, and to estimate the frequency of packet-drops due to buffer overflow. Second, many network control tasks utilize queue length information to perform effectively, which inevitably adds to the control overheads in a network. Therefore, it is important to understand the role played by queue length information in network control, and its impact on various performance metrics. In this thesis, we study the *interplay between the asymptotic behavior of buffer occupancy, queue length information, and traffic statistics* in the context of scheduling, flow control, and resource allocation.

First, we consider a single-server queue and deal with the question of how often control messages need to be sent in order to effectively control congestion in the queue. Our results show that arbitrarily infrequent queue length information is sufficient to ensure optimal asymptotic decay for the congestion probability, as long as the control information is accurately received. However, if the control messages are subject to errors, the congestion probability can increase drastically, even if the control messages are transmitted often.

Next, we consider a system of parallel queues sharing a server, and fed by a statistically homogeneous traffic pattern. We obtain the large deviation exponent of the buffer overflow probability under the well known max-weight scheduling policy. We also show that the queue length based max-weight scheduling outperforms some well known queue-blind policies in terms of the buffer overflow probability.

Finally, we study the asymptotic behavior of the queue length distributions when a mix of heavy-tailed and light-tailed traffic flows feeds a system of parallel queues. We obtain an exact asymptotic queue length characterization under generalized max-weight scheduling. In contrast to the statistically homogeneous traffic scenario, we show that max-weight scheduling leads to poor asymptotic behavior for the light-tailed traffic, whereas a queue-blind priority policy gives good asymptotic behavior.

Thesis Supervisor: Eytan H. Modiano
Title: Associate Professor

This dissertation is a humble offering at the Lord's feet.

यत्करोषि यदश्नासि यज्जुहोषि ददासि यत्।
यत्तपस्यसि कौन्तेय तत्कुरुष्व मदर्पणम्॥

"Whatever you do, whatever you eat, whatever you sacrifice, whatever you give away, and whatever austerities you practise – do it as an offering to Me." – *Bhagavad Gita 9-27.*

Acknowledgments

This section of the dissertation is unquestionably the pleasantest to write. Ironically, it is also the *hardest* to write. The pleasantness, of course, is due to the fact that the author finally gets an opportunity to express his gratitude towards everyone who made this document possible. The *difficulty* lies in finding the words that adequately reflect this sense of gratitude and indebtedness. I humbly embark on this highly satisfying task, fully realizing that my words are likely to fall short.

The sense of inadequacy that I feel in writing this section is at least partly due to the first recipient of my gratitude – my advisor, Professor Eytan Modiano. A student's thesis advisor is often the biggest influence in his or her professional making, and this is certainly true in my case. I suppose that the job of an advisor is relatively easy when things are progressing steadily and smoothly in a student's research. It is during the weeks and months of laborious plodding, crumbling confidence, and growing frustration that I most appreciate Eytan's patience and support. Always encouraging and never judgemental, his ability and willingness to see a student's frustrations as partly *his own* makes him an exemplary advisor. Overall, I feel lucky, privileged, and proud to have been at the receiving end of his mentorship.

It's been a pleasure, Eytan!

I am also deeply grateful to the members of my thesis committee, Professors John Tsitsiklis and Lizhong Zheng. Far from just being readers of my thesis, John and Lizhong acted more like co-advisors during the various stages of my thesis research, actively helping me with their valuable suggestions and perspectives.

While Eytan was away on a sabbatical at the University of Minnesota during 2007, Lizhong voluntarily acted as my de facto advisor at MIT. Much of the work in Chapter 2 was carried out during this time, with valuable inputs from him. I thank Lizhong for all his support and help during my stay at MIT. Oh, the two classes I took with him (6.441: Transmission of Information, and 6.452: Wireless Communications) were lots of fun too!

During the later stages of my doctoral research, and in particular during the past year, I have tremendously benefitted from my interactions with Prof. John Tsitsiklis. Although my interactions with him were very sporadic and brief, I obtained some very crucial insights during those brief meetings. As a matter of fact, the two main theorems in Chapter 5 materialized as a result of a 30 minute meeting with John in June this year. I am also fortunate to have been John's teaching assistant for 6.262 (Discrete Stochastic Processes) in Spring 2008, which was a greatly edifying and enjoyable experience.

The work reported in Appendix A was carried out in collaboration with Prof. Shie Mannor (Technion) and Dr. Ishai Menache. Although I had formulated the problem of scheduling over uncertain wireless links more than a year ago, I was hopelessly stuck on the problem for a few months, before our collaboration started paying off. Many thanks to Shie and Ishai for their valuable help and co-operation.

I am also grateful to all the students and post-docs in CNRG, with whom I have had the pleasure of interacting over the last six years.

I offer my profound gratitude to *all my teachers.* I refer not only to the many great lecturers at MIT who taught my classes at graduate school. I would be amiss to forget the teachers who taught circuit theory and communications to a reticent loner in IIT Madras, or the teacher who taught $\sin^2\theta + \cos^2\theta = 1$ to a barefooted, street cricket-crazy boy in Chennai.

I could place here a clichéd sentence thanking my parents for all they have done and been for me. Instead, I will make it less sheepish by placing a mere assertion: Any virtues that exist in me, including those that I exude in my work, are largely a result of the values instilled in me as a part of my upbringing.

As for my wife Prabha, I realize that jetting back and forth across this vast continent while having her own thesis to deal with should have been stressful at times, because it was for me. However, I will not thank my wife – one hand of a competent pianist does not thank the other for playing well in unison. Instead, I merely look forward to the tunes that lie ahead of us.

The late Prof. Dilip Veeraraghavan, my friend, teacher and exemplar in several ways, would have been boundlessly happy to see this day. I have learned a lot from his life; I have learned even more from his death.

Finally, my stay at MIT would have been much duller, if it was not for my circle of close friends. In no specific order, thanks to Arvind, Vivek, Lavanya, Mythili, and Aditi for all the good food, music, and conversations. Thanks in particular to Jaykumar for the above three, plus the occasional dose of queueing theory.

Contents

Chapter 1

Introduction

In today's increasingly connected world, data communication networks play a vital role in practically every sphere of our daily lives. Well established infrastructures such as satellite communications, trans-continental and inter-continental fiber-optic links, cellular phones, and wireless internet are taken for granted. An internet connection is often considered an essential utility in urban households, along with water supply and electricity. More recently, there has been a tremendous growth in voice-over-IP (VoIP) applications, and hand held devices that access both voice and data over cellular networks.

Whatever may be the specific application or the medium, it is essential to design effective data transport mechanisms in a network, so as to utilize the resources available in an efficient manner. *Network control* is the generic and collective term used to refer to such transport mechanisms in a network.

From a theoretical perspective, the problem of optimally transporting data over a general network is a staggeringly complex one. Optimal data transmission over a single communication link has been well understood for over six decades now, since the inception of Shannon theory [57]. However, the extension of Shannon theoretic capacity results to the simplest of multi-user network models, such as the broadcast [17] and relay channels [16], has proved notoriously hard. This being the case, we can safely assert that a complete theoretical understanding of optimal data transmission over a general network is well beyond our grasp.

Due to the lack of a unified underlying theory [19], the study and design of communication networks has been dominated by 'divide and conquer' approaches, in which the various functionalities of a network are treated as separate modules that come together. This modularity of network design is achieved through *layering* in network architecture; see [5]. Layering was envisioned as a means to promote clean and modularized network design, in which the different layers in a network can treat the other layers as black boxes. Although layering is somewhat artificial in reality in the sense that the different layers in a network are invariably coupled, it effectively discourages 'spaghetti designs' that are less adaptable to future changes. Apart from promoting modularized and scalable design, layering also facilitates the analysis and theoretical understanding of communication networks. For example, once the communication links in a network are abstracted away as edges in a graph, standard results from combinatorics and graph theory can be brought to bear upon routing and scheduling problems. Queueing theoretic results can be used to characterize the delay experienced by data packets at various nodes in a stochastic network.

In the context of a layered architecture, various network control tasks are also thought of as operating within different layers. The most important examples of network control tasks include *scheduling* and *resource allocation* at the link layer, *routing* at the network layer, and *flow control* at the transport layer.

We next review some important results from the literature on network control.

1.1 Network Control and the Max-Weight Framework

The theoretical foundations of network control, including the notions of stability, rate region, and throughput optimal control of a constrained queueing system were laid out by Tassiulas and Ephremides in [63]. More importantly from a practical perspective, the authors also propose a joint scheduling and routing policy which they show can stably support the largest possible set of rates in any given multi-hop network. Their policy takes into account the instantaneous channel qualities and queue backlogs to perform *maximum-weight scheduling* of links and *back-pressure routing* of data. This algorithm is quintessentially cross-layer in nature, combining physical layer channel qualities and link layer queue lengths to perform link layer scheduling and network layer routing. An important and desirable feature of this approach is that no *a priori* information on arrival or link statistics is necessary to implement it. However, a centralized network controller is necessary to perform the network control tasks, and the computational complexity could be exponential for some networks.

The mathematical framework used in [63] for proving the stability of a constrained queueing network is known as *Lyapunov stability*. The idea is to define a suitable *Lyapunov function* of the queue backlogs in the system, and to prove that the expected drift in the Lyapunov function is negative, when the queue backlogs are large. Lyapunov stability theory for queueing networks has been more thoroughly developed since its novel application in [63], see [42] for example.

As it sometimes happens with papers of fundamental theoretical value and significant practical implications, it took a few years for the networking community to fully realize the potential impact of the above work. However, during the last decade or so, the max-weight framework has been extended along several conceptual as well as practical lines. For example, in order to circumvent the potentially exponential computational complexity of the original algorithm in [63], randomized algorithms with linear complexity were proposed in [62] and [27]. Polynomial-time approximation methods were proposed in [58]. Distributed and greedy scheduling approaches were proposed in [14] and [73].

The max-weight framework was adopted more thoroughly into a wireless setting by including power allocation in [46]. In [45], it was used to develop an optimal power control and server allocation scheme in a multi-beam satellite network. Maximum weight matching has been used to achieve full throughput in input queued switches [41]. Algorithms developed in [12] for the dynamic reconfiguring and routing of light-paths in an optical network are also based on back-pressure. Joint congestion control and scheduling for multi-hop networks was studied in [37, 44], and [13], thus bringing the transport layer task of congestion control also into a cross-layer max-weight framework.

In [20], queue length based scheduling is generalized to encompass a wide class of functions of the queue backlogs, and the relationship between the arrival statistics and Lyapunov functions is explored in detail. This work is directly useful to us in Chapters 4 and 5, where we use the traffic statistics to design the appropriate queue length function to be employed within a generalized max-weight framework.

1.2 Performance Metrics for Network Control

There are several performance metrics that gauge the merits of scheduling, routing and flow control policies. Examples of such performance metrics include throughput, delay, and fairness. *Throughput*, which measures the long term average rate at which data can be transported under a given control policy, is perhaps the most important performance metric. Throughput is often referred to as a 'first order metric,' due to the fact that it only depends on the expected values of the arrival and service processes in a stochastic network. *Throughput optimality* is the ability of a control policy to support the largest possible set of rates in a network.

A more discerning performance metric than throughput is *delay*. In a stochastic network, the delay experienced by a packet at a node is a random variable, which is closely related to the buffer occupancy or the queue size at the node. The total end-to-end delay experienced by a packet is therefore a function of the buffer occupancies at each node traversed by a packet. Unlike throughput, the delay and buffer occupancy distributions depend on the complete statistical properties of the arrival and service processes, and not just on their means.

The requirements imposed on the delay experienced in a network are commonly referred to as Quality of Service (QoS) requirements. The recent burgeoning of VoIP applications, and hand-held devices that carry delay-sensitive voice along with delay-insensitive data on the same network, has made these QoS constraints more important and challenging than ever before. These QoS requirements can take the form of bounds on the average delay, or impose constrains on the behavior of the distribution function of the delay. For example, a worst case delay assurance such as 'the delay cannot exceed a certain value with at least 99% probability' necessitates the tail distribution of the delay to fall off sufficiently fast.

This brings us to the concept of *asymptotic* QoS metrics, which constitutes a recurring theme in this thesis. Loosely speaking, an asymptotic QoS metric captures the behavior of the probability that the delay or queue occupancy exceeds a certain large threshold. The manner in which the above tail probability behaves as a function of the large threshold sheds light on how 'likely' it is for large deviations from typical behavior to occur. Apart from being useful in the context of providing worst case delay assurances, asymptotic QoS metrics have other important applications. In particular, an understanding of the tail of the queue size distribution is essential for buffer provisioning in a network, and to estimate the frequency of packet drops due to buffer overflow. For example, if the tail of the queue occupancy distribution decays exponentially fast, it is clear that the buffer size needed to ensure a given probability of overflow, is usually much *smaller* than if the tail distribution were to decay according to a power-law.

Although the stability region and throughput optimality properties of the max-weight framework are well studied, there is relatively little literature on the QoS metrics under the framework. Delay bounds are derived in [34] for max-weight scheduling in spatially homogeneous wireless ad hoc networks. In [31], a scheduling algorithm that simultaneously provides delay as well as throughput guarantees is proposed. In a parallel queue setting, max-weight scheduling is shown in [43] to attain order optimal delay for arrival rates within a scaled stability region. Delay analysis for switches under max-weight matching and related scheduling policies is studied in [54, 55], and [26]. Large deviation analysis of buffer overflow probabilities under max-weight scheduling is studied in [6, 56, 60, 65–68] and [76].

1.3 The Role of Control Information

Network control policies generally base their control decisions on the instantaneous network state, such as channel quality of the various links, and the queue backlogs at the nodes. For example, congestion control policies regulate the rate of traffic entering a network based on prevailing buffer occupancies. The max-weight framework, developed in [63] and extended in [46], utilizes instantaneous channel states as well as queue length information to perform joint scheduling and routing. We use the phrase 'control information' to denote the information about the network state that is necessary to operate a control policy.

Since this control information usually shares the same communication medium as the payload data, the exchange of control information inevitably adds to the signalling overheads in a network. Therefore, it is important to better understand the role played by the control information in network control. In particular, we are interested in understanding how various performance metrics are impacted if the control information is subject to delays, losses or infrequent updates.

Perhaps the earliest, and certainly the most well known investigation about control information in data networks was carried out by Gallager in [23]. He derives information theoretic lower bounds on the amount of protocol information needed for network nodes to keep track of source and destination addresses, as well as message starting and stopping times. A paper that is closely related to our work in Chapter 2, and specifically deals with the role of queue length information in the congestion control of a single- server queue, is [49]. In that paper, the authors consider the problem of maximizing throughput in a single-server queue subject to an overflow probability constraint, and show that a threshold policy achieves this objective.

Since the general max-weight framework utilizes both instantaneous channel states and queue length information, a question arises as to how the policy would perform if either the channel state or queue length information is delayed or conveyed imperfectly. From the perspective of throughput, it turns out that the role of channel state information (CSI) is more important than that of queue length information (QLI).

There are several studies in the literature [28, 47, 74, 75] that show that under imperfect, delayed, or incomplete CSI, the throughput achievable in a network is negatively impacted. In contrast, when QLI is arbitrarily delayed or infrequently available, it is possible to ensure that there is no loss in throughput. However more discerning QoS metrics such as buffer occupancy and delay could suffer under imperfect or delayed QLI.

In this thesis, we investigate the role played by queue length information in the operation of scheduling, congestion control, and resource allocation policies. In particular, we focus on the asymptotic QoS metrics under queue length based scheduling and congestion control policies in simple queueing networks. In Chapter 2, we consider a single-server queue with congestion based flow control, and study the relationship between the tail of the queue occupancy distribution, and the rate of queue length information available to the flow controller. In Chapters 3, 4, and 5, we consider a system of parallel queues sharing a server, and study the asymptotic behavior of buffer overflow probability under various queue-aware and queue-blind scheduling policies, and under different qualitative assumptions on the traffic statistics.

In Appendix A, we study the problem of scheduling over wireless links, when there is no explicit CSI available to the scheduler. This work is reported as an appendix rather than as a chapter, because the contents therein are not directly related to the rest of the chapters.

An outline of this thesis, including a summary of our main results and contributions, is given in the following section.

1.4 Thesis Outline and Contributions

The recurring theme in this thesis is the role of control information in network control, and its impact on the asymptotic behavior of queue occupancy distributions. We consider simple queueing models such as single server queues and parallel queues sharing a server, and study the interplay between network control policies, control information, traffic statistics, and the asymptotic behavior of buffer occupancy. However, we believe that the conceptual understanding and guidelines that are obtained from our study are more widely applicable.

1.4.1 Queue length information and congestion control

In Chapter 2, we consider a single server queue and deal with the basic question of how often control messages need to be sent in order to effectively control congestion in the queue. We separately consider the flow control and resource allocation problems, and characterize the rate of queue length information necessary to achieve a certain congestion control performance in the queue.

For the flow control problem, we consider a single server queue with congestion based flow control. The queue is served at a constant rate, and is fed by traffic that is regulated by a flow controller. The arrival rate at a given instant is chosen by a flow control policy, based on the queue length information obtained from a queue observer. We identify a simple 'two-threshold' flow control policy and derive the corresponding tradeoff between the rate of control and congestion probability in closed form. We show that the two threshold policy achieves the lowest possible asymptotic overflow probability for arbitrarily low rates of control.

Next, we consider a model where losses may occur in the control channel, possibly due to wireless transmission. We characterize the impact of control-channel errors on the congestion control performance of the two threshold policy. We assume a probabilistic model for the errors on the control channel, and show the existence of a critical error probability, beyond which the errors in receiving the control packets lead to a drastic increase in the congestion probability. However, for error probabilities below the critical value, the congestion probability is of the same exponential order as in a system with an error free control channel. Moreover, we determine the optimal apportioning of bandwidth between the control signals and the server in order to achieve the best congestion control performance.

Finally, we study the server allocation problem in a single server queue. In particular, we consider a queue with a constant input rate. The service rate at any instant is chosen depending on the congestion level in the queue. This framework turns out to be mathematically similar to the flow control problem, so that most of our results for the flow control case also carry over to the server allocation problem.

Our results in Chapter 2 indicate that arbitrarily infrequent queue length information is sufficient to ensure optimal asymptotic decay for the buffer overflow probability, as long as the control information is accurately received. However, if the control messages are subject to errors, the congestion probability can increase drastically, even if the control messages are transmitted often.

In the remaining chapters of the thesis, we study a system of parallel queues, served by a single server. In this setting, we characterize the asymptotic behavior of buffer overflow events under various scheduling policies and traffic statistics. Specifically, we compare well known queue-aware and queue-blind scheduling policies in terms of buffer overflow performance, and the amount of queue length information required to operate them, if any. We incorporate two different statistical paradigms for the arrival traffic, namely light-tailed and heavy-tailed, and derive widely different asymptotic behaviors under some well known scheduling policies.

1.4.2 Queue-aware vs. queue-blind scheduling under symmetric traffic

Chapter 3 characterizes and compares the large deviation exponents of buffer overflow probabilities under queue-aware and queue-blind scheduling policies. We consider a system consisting of N parallel queues served by a single server, and study the impact of queue length information on the buffer overflow probability. Under statistically identical arrivals to each queue, we explicitly characterize the large deviation exponent of buffer overflow under max-weight scheduling, which in our setting, amounts to serving the Longest Queue First (LQF).

Although any non-idling scheduling policy would achieve the same throughput region and total system occupancy distribution in our setting, the LQF policy outperforms queue blind policies such as processor sharing (PS) in terms of the buffer overflow probability. This implies that the buffer requirements are lower under LQF scheduling than under queue blind scheduling, if we want to achieve a given overflow probability. For example, our study indicates that under Bernoulli and Poisson traffic, the buffer size required under LQF scheduling is only about 55% of that required under random scheduling, when the traffic is relatively heavy.

On the other hand, with LQF scheduling, the scheduler needs queue length information in *every* time slot, which leads to a significant amount of control signalling. Motivated by this, we identify a 'hybrid' scheduling policy, which achieves the same buffer overflow exponent as the LQF policy, with arbitrarily infrequent queue length information. This result, as well as the ones in Chapter 2, suggests that the large deviation behavior of buffer overflow can be preserved under arbitrarily infrequent queue length updates. This is a stronger assertion than the well known result that the stability region of a queueing system is preserved under arbitrarily infrequent queue length information.

1.4.3 Queue-aware vs. queue-blind scheduling in the presence of heavy-tailed traffic

In the next two chapters, 4 and 5, we study the asymptotic behavior of the queue size distributions, when a mix of heavy-tailed and light-tailed traffic flows feeds queueing network. Modeling traffic using heavy-tailed random processes has become common during the last decade or so, due to empirical evidence that internet traffic is much more bursty and correlated than can be captured by any light-tailed random process [35]. We consider a system consisting of two parallel queues, served by a single server according to some scheduling policy. One of the queues is fed by a heavy-tailed arrival process, while the other is fed by light-tailed traffic. We refer to these queues as the 'heavy' and 'light' queues, respectively.

In Chapter 4, we consider the wireline case, where the queues are reliably connected to the server. We analyze the asymptotic performance of max-weight-α scheduling, which is a generalized version of max-weight scheduling. Under this throughput optimal policy, we derive an exact asymptotic characterization of the queue occupancy distributions. Our characterization shows that the light queue occupancy is heavier than a power-law curve under max-weight-α scheduling, for all values of the scheduling parameters. A surprising outcome of our asymptotic characterization is that the 'plain' max-weight scheduling policy induces the worst possible asymptotic behavior on the light queue tail.

On the other hand, we show that under the queue-blind priority scheduling for the light queue, the tail distributions of *both* queues are asymptotically as good as they can possibly be under any policy. However, priority scheduling suffers from the drawback that it may not be throughput optimal in general, and can lead to instability effects in the heavy queue. To remedy this situation, we propose a log-max-weight (LMW) scheduling policy, which gives significantly more importance to the light queue, compared to max-weight-α scheduling. However, the LMW policy does not ignore the heavy queue when it gets overwhelmingly large, and can be shown to be throughput optimal.

We analyze the asymptotic behavior of the LMW policy and show that the light queue occupancy distribution decays exponentially fast. We also obtain the exact large deviation exponent of the light queue tail under a regularity assumption on the heavy-tailed input. Thus, the LMW policy has both desirable attributes – it is throughput optimal in general, and ensures an exponentially decaying tail for the light queue distribution.

In Chapter 5, we extend the above results to a wireless setting, where the queues are connected to the server through randomly time-varying links. In this scenario, we show that the priority policy fails to stabilize the queues for some traffic rates inside the rate region of the system. However, the max-weight-α and LMW policies are throughput optimal. Next, from the point of view of queue length asymptotics, we show that the tail behavior of the steady-state queue lengths under a given policy depends strongly on the arrival rates. This is an effect which is not observed when the queues are always connected to the server. For example, we show that under max-weight-α scheduling, the light queue distribution has an exponentially decaying tail if the arrival rate is below a critical value, and a power-law like behavior if the arrival rate is above the critical value. On the other hand, we show that LMW scheduling guarantees much faster decay of the light queue tail, in addition to being throughput optimal.

Our results in Chapters 4 and 5 suggest that a blind application of max-weight scheduling to a network with heavy-tailed traffic can lead to a very poor asymptotic QoS profile for the light-tailed traffic. This is because max-weight scheduling forces the light-tailed flow to compete for service with the highly bursty heavy-tailed flow. On the other hand, the queue length-blind priority scheduling for the light queue ensures good asymptotic QoS for both queues, whenever it can stabilize the system. This is in stark contrast to our results in Chapter 3, wherein the max-weight policy outperforms the queue length-blind policies in terms of the large deviation exponent of the buffer overflow probability, when the traffic is completely homogeneous.

We also believe that the LMW policy represents a unique 'sweet spot' in the context of scheduling light-tailed flows in the presence of heavy-tailed traffic. This is because the LMW policy affords a very favorable treatment to the light-tailed traffic, without completely ignoring large build-up of the heavy-tailed flow.

1.4.4 Throughput maximization over uncertain wireless channels

In Appendix A, we consider the problem of scheduling users in a wireless down-link or up-link, when no explicit CSI is made available to the scheduler. However, the scheduler can indirectly estimate the current channels states using the acknowledgement history from past transmissions. We characterize the capacity region of such a system using tools from Markov Decision Processes (MDP) theory. Specifically, we prove that the capacity region boundary is the uniform limit of a sequence of Linear Programming (LP) solutions. Next, we combine the LP solution with a queue length based scheduling mechanism that operates over long 'frames,' to obtain a throughput optimal policy for the system. By incorporating results from MDP theory within the Lyapunov-stability framework, we show that our frame-based policy stabilizes the system for all arrival rates that lie in the interior of the capacity region.

1.4.5 Reading suggestions

Chapters 2 and 3 can be read independently. Although Chapter 4 can also be read independently, it is recommended that Chapters 4 and 5 be read together, in that order. In any case, Chapter 5 should be read *after* Chapter 4. Finally, Appendix A is independent of the rest of the chapters.

Chapter 2

On the Role of Queue Length Information in Congestion Control and Resource Allocation

2.1 Introduction

In this chapter, we study the role played by queue length information in flow control and resource allocation policies. Flow control and resource allocation play an important role in keeping congestion levels in the network within acceptable limits. Flow control involves regulating the rate of the incoming exogenous traffic to a network, depending on its congestion level. Resource allocation, on the other hand, involves assigning larger service rates to the queues that are congested, and vice-versa. Most systems use a combination of the two methods to avoid congestion in the network, and to achieve various performance objectives [38, 44].

The knowledge of queue length information is often useful, sometimes even necessary, in order to perform these control tasks effectively. Almost all practical flow control mechanisms base their control actions on the level of congestion present at a given time. For example, in networks employing TCP, packet drops occur when buffers are about to overflow, and this in turn, leads to a reduction in the window size and packet arrival rate. Active queue management schemes such as Random Early Detection (RED) are designed to pro-actively prevent congestion by randomly dropping some packets *before* the buffers reach the overflow limit [21]. On the other hand, resource allocation policies can either be queue-blind, such as round-robin, first come first served (FCFS), and generalized processor sharing (GPS), or queue-aware, such as maximum weight scheduling. Queue length based scheduling techniques are known to have superior throughput, delay and queue overflow performance than queue-blind algorithms such as round-robin and processor sharing [6, 43, 64].

Since the queue lengths can vary widely over time in a dynamic network, queue occupancy based flow control and resource allocation algorithms typically require the exchange of control information between agents that can observe the various queue lengths in the system, and the controllers which adapt their actions to the varying queues. This control information can be thought of as being a part of the inevitable protocol and control overheads in a network. Gallager's seminal paper [23] on basic limits on protocol information was the first to address this topic. He derives information theoretic lower bounds on the amount of protocol information needed for network nodes to keep track of source and destination addresses, as well as message starting and stopping times.

This chapter deals with the basic question of how often control messages need to be sent in order to effectively control congestion in a single server queue. We separately consider the flow control and resource allocation problems, and characterize the rate of control necessary to achieve a certain congestion control performance in the queue. In particular, we argue that there is an inherent tradeoff between the rate of control information, and the corresponding congestion level in the queue. That is, if the controller has accurate information about the congestion level in the system, congestion control can be performed very effectively by adapting the input/service rates appropriately. However, furnishing the controller with accurate queue length information requires significant amount of control. Further, frequent congestion notifications may also lead to undesirable retransmissions in packet drop based systems such as TCP. Therefore, it is of interest to characterize how frequently congestion notifications need to be employed, in order to achieve a certain congestion control objective. We do not explicitly model the packet drops, but instead associate a cost with each congestion notification. This cost is incurred either because of the ensuing packet drops that may occur in practice, or might simply reflect the resources needed to communicate the control signals.

We consider a single server queue with congestion based flow control. Specifically, the queue is served at a constant rate, and is fed by packets arriving at one of two possible arrival rates. In spite of being very simple, such a system gives us enough insights into the key issues involved in the flow control problem. The two input rates may correspond to different quality of service offerings of an internet service provider, who allocates better service when the network is lightly loaded but throttles back on the input rate as congestion builds up; or alternatively to two different video streaming qualities where a better quality is offered when the network is lightly loaded.

The arrival rate at a given instant is chosen by a flow control policy, based on the queue length information obtained from a queue observer. We identify a simple 'two-threshold' flow control policy and derive the corresponding tradeoff between the rate of control and congestion probability in closed form. We show that the two-threshold policy achieves the best possible decay exponent (in the buffer size) of the congestion probability for arbitrarily low rates of control. Although we mostly focus on the two-threshold policy owing to its simplicity, we also point out that the two-threshold policy can be easily generalized to resemble the RED queue management scheme.

Next, we consider a model where losses may occur in the control channel, possibly due to wireless transmission. We characterize the impact of control channel losses on the congestion control performance of the two-threshold policy. We assume a probabilistic model for the losses on the control channel, and show the existence of a critical loss probability, beyond which the losses in receiving the control packets lead to an exponential worsening of the congestion probability. However, for loss probabilities below the critical value, the congestion probability is of the same exponential order as in a system with an loss-free control channel. Moreover, we determine the optimal apportioning of bandwidth between the control signals and the server in order to achieve the best congestion control performance.

Finally, we study the server allocation problem in a single server queue. In particular, we consider a queue with a constant input rate. The service rate at any instant is chosen from two possible values depending on the congestion level in the queue. This framework turns out to be mathematically similar to the flow control problem, so that most of our results for the flow control case also carry over to the server allocation problem. Parts of the contents of this chapter were previously published in [32, 33].

The rest of the chapter is organized as follows. Section 2.2 introduces the system model, and the key parameters of interest in the design of a flow control policy. In Section 2.3, we introduce and analyze the two-threshold policy. In Section 2.4, we investigate the effect of control channel losses on the congestion control performance of the two-threshold policy. Section 2.5 deals with the problem of optimal bandwidth allocation for control signals in a loss-prone system. The server allocation problem is presented in Section 2.6, and Section 2.7 concludes the chapter.

2.2 Preliminaries

2.2.1 System description

Let us first describe a simple model of a queue with congestion based flow control. Figure 2-1 depicts a single server queue with a constant service rate μ. We assume that the packet sizes are exponentially distributed with mean 1. Exogenous arrivals are fed to the queue in a regulated fashion by a flow controller. An observer watches the queue evolution and sends control information to the flow controller, which changes the input rate $A(t)$ based on the control information it receives. The purpose of the observer-flow controller subsystem is to change the input rate so as to the control congestion level in the queue.

We assume that the input rate at any instant is chosen to be one of two distinct possible values, $A(t) \in \{\lambda_1, \lambda_2\}$, where $\lambda_2 < \lambda_1$ and $\lambda_2 < \mu$. Physically, this model

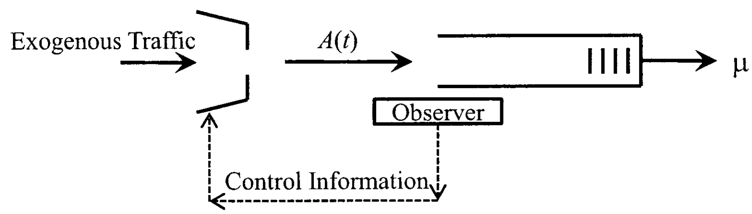


Figure 2-1: A single server queue with input rate control.

may be motivated by a DSL-like system, wherein a minimum rate λ_2 is guaranteed, but higher transmission rates might be intermittently possible, as long as the system is not congested.

Our assumption about the flow control policy choosing between just two arrival rates is partly motivated by a theoretical result in [49]. In that paper, it is shown that in a single server system with flow control, where the input rate is allowed to vary continuously in the interval $[\lambda_2, \lambda_1]$, a 'bang-bang' solution is optimal. That is, a queue length based threshold policy that only uses the two extreme values of the possible input rates, is optimal in the sense of maximizing throughput for a given congestion probability constraint. Since we are interested in the rate of control in addition to throughput and congestion, a solution with two input rates may not be optimal in our setting. However, we will see that our assumption does not entail any sub-optimality in the asymptotic regime.

The congestion notifications are sent by the observer in the form of information-less control packets. Upon receiving a control packet, the flow controller switches the input rate from one to the other. We focus on Markovian control policies, in which the input rate chosen after an arrival or departure event is only a function of the previous input rate and queue length. Note that due to the memoryless arrival and service time distributions, there is nothing to be gained by using non-Markovian policies.

2.2.2 Markovian control policies

We begin by defining notions of Markovian control policies, and its associated congestion probability.

Let $t > 0$ denote continuous time. Let $Q(t)$ and $A(t)$ respectively denote the queue length and input rate (λ_1 or λ_2) at time t. Define $Y(t) = (Q(t), A(t))$ to be the state of the system at time t. We assign discrete time indices $n \in \{0, 1, 2, \dots\}$ to each arrival and departure event in the queue ("queue event"). Let Q_n and A_n respectively denote the queue length and input rate just after the n^{th} queue event. Define $Y_n = (Q_n, A_n)$. A flow control policy assigns service rates A_n after every queue event.

Definition 2.1 *A control policy is said to be Markovian if it assigns input rates A_n such that*

$$\mathbb{P}\{A_{n+1}|Q_{n+1}, Y_n, \dots, Y_0\} = \mathbb{P}\{A_{n+1}|Q_{n+1}, Y_n\}, \tag{2.1}$$

$\forall n = 0, 1, 2 \dots.$

For a Markovian control policy operating on a queue with memoryless arrival and packet size distributions, it is easy to see that $Y(t)$ is a continuous time Markov process with a countable state space, and that Y_n is the imbedded Markov chain for the process $Y(t)$. For a control policy under which the Markov process $Y(t)$ is positive recurrent, the steady-state queue occupancy exists. Let us denote by Q the steady-state queue occupancy under a generic policy, when it exists.

Definition 2.2 *The congestion probability is defined as $\mathbb{P}\{Q \geq M\}$, where M is some congestion limit.*

2.2.3 Throughput, congestion, and rate of control

We will focus on three important parameters of a flow control policy, namely, throughput, congestion probability, and rate of control. There is usually an inevitable tradeoff between throughput and congestion probability in a flow control policy. In fact, a good flow control policy should ensure a high enough throughput, in addition to effectively controlling congestion. We assume that a minimum throughput guarantee γ should be met. Observe that a minimum throughput of λ_2 is guaranteed, whereas any throughput less than $\min(\lambda_1, \mu)$ can be supported, by using the higher input rate λ_1 judiciously. Loosely speaking, a higher throughput is achieved by maintaining the higher input rate λ_1 for a longer fraction of time, with a corresponding tradeoff in the congestion probability.

In the *single-threshold* policy, the higher input rate λ_1 is used whenever the queue occupancy is less than or equal to some threshold l, and the lower rate is used for queue lengths larger than l. It can be shown that a larger value of l leads to a larger throughput, and vice-versa. Thus, given the throughput requirement γ, we can determine the corresponding threshold l to meet the requirement. Once the threshold l has been fixed, it can be easily shown that the single-threshold policy minimizes the probability of congestion. However, it suffers from the drawback that it requires frequent transmission of control packets, since the system may often toggle between states l and $l+1$. It turns out that a simple extension of the single threshold policy gives rise to a family of control policies, which provide more flexibility with the rate of control, while still achieving the throughput guarantee and ensuring good congestion control performance.

2.3 The Two-Threshold Flow Control Policy

As suggested by the name, the input rates in the two-threshold policy are switched at two distinct thresholds l and m, where $m \geq l+1$, and l is the threshold determined by the throughput guarantee. As we shall see, the position of the second threshold gives us another degree of freedom, using which the rate of control can be fixed at a desired value. The two-threshold policy operates as follows.

Suppose we start with an empty queue. The higher input rate λ_1 is used as long as the queue length does not exceed m. When the queue length grows past m, the input rate switches to the lower value λ_2. Once the lower input rate is employed,

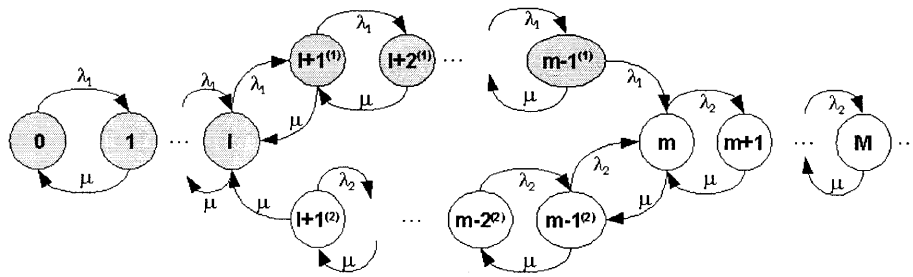


Figure 2-2: The Markov process $Y(t)$ corresponding to the two-threshold flow control policy.

it is maintained until the queue length falls back to l, at which time the input rate switches back to λ_1. We will soon see that this 'hysteresis' in the thresholds helps us tradeoff the control rate with the congestion probability. The two-threshold policy is easily seen to be Markovian, and the state space and transition rates for the process are shown in Figure 2-2.

Define $k = m - l$ to be the difference between the two queue length thresholds. We use the short hand notation $l+i^{(1)}$ and $l+i^{(2)}$ in the figure, to denote respectively, the states $(Q(t) = l + i, A(t) = \lambda_1)$ and $(Q(t) = l + i, A(t) = \lambda_2)$, $i = 1, \ldots, k - 1$. For queue lengths l or smaller and m or larger, we drop the subscripts because the input rate for these queue lengths can only be λ_1 and λ_2 respectively. Note that the case $k = 1$ corresponds to the single threshold policy. It can be shown that the throughput of the two-threshold policy for $k > 1$ cannot be smaller than that of the single threshold policy. Thus, given a throughput guarantee, we can solve for the threshold l using the single threshold policy, and the throughput guarantee will also be met for $k > 1$. We now explain how the parameter k can be used to tradeoff the rate of control and the congestion probability.

2.3.1 Congestion probability vs. rate of control tradeoff

Intuitively, as the gap between the two-thresholds $k = m - l$ increases for a fixed l, the rate of control packets sent by the observer should decrease, while the probability of congestion should increase. It turns out that we can characterize the rate-congestion tradeoff for the two-threshold policy in closed form. We do this by solving for the steady state probabilities in Figure 2-2. Define $\rho_2 = \frac{\lambda_2}{\mu}, \rho_1 = \frac{\lambda_1}{\mu}$, and $\eta_1 = 1/\rho_1$. Note that by assumption, we have $\rho_2 < 1$ and $\rho_1 > \rho_2$.

Let us denote the steady state probabilities of the non-superscripted states in Figure 2-2 by p_j, where $j \leq l$, or $j \geq m$. Next, denote by $p_{l+i}^{(1)}$ ($p_{l+i}^{(2)}$) the steady state probability of the state $l + i^{(1)}$ ($l + i^{(2)}$), for $i = 1, 2, \ldots, k-1$. By solving for the steady state probabilities of various states in terms of p_l, we obtain:

$$p_i = p_l \eta_1^{l-i}, \quad 0 \leq i \leq l,$$

$$p_{m-j}^{(1)} = \begin{cases} \frac{1-\eta_1^j}{1-\eta_1^k} p_l, & \eta_1 \neq 1 \\ \frac{j}{k} p_l, & \eta_1 = 1 \end{cases}, j = 1, 2, \ldots, k-1,$$

$$p_{l+j}^{(2)} = \rho_1 \frac{1-\rho_2^j}{1-\rho_2} p_{m-1}^{(1)}, j = 1, 2, \ldots, k-1,$$

and

$$p_j = \rho_2^{j-m} \rho_1 \frac{1-\rho_2^k}{1-\rho_2} p_{m-1}^{(1)}, j \geq m.$$

The value of p_l, which is the only remaining unknown in the system can be determined by normalizing the probabilities to 1:

$$p_l = \begin{cases} \left[\frac{k(1-\rho_2\eta_1)}{\eta_1(1-\eta_1^k)(1-\rho_2)} - \frac{\eta_1^{l+1}}{1-\eta_1} \right]^{-1}, & \eta_1 \neq 1 \\ \left[l + \frac{k+1}{2} + \frac{1}{1-\rho_2} \right]^{-1}, & \eta_1 = 1 \end{cases}. \tag{2.2}$$

Using the steady-state probabilities derived above, we can compute the probability of congestion as

$$\mathbb{P}\{Q \geq M\} = \sum_{j \geq M} p_j = \rho_2^{M-m} \rho_1 \frac{1-\rho_2^k}{(1-\rho_2)^2} p_{m-1}^{(1)}. \tag{2.3}$$

We define the control rate simply as the average number of control packets transmitted by the queue observer per unit time. Since there is one packet transmitted by the observer every time the state changes from $m-1^{(1)}$ to m or from $l+1^{(2)}$ to l, the rate (in control packets per second) is given by

$$R = \lambda_1 p_{m-1}^{(1)} + \mu p_{l+1}^{(2)}.$$

Next, observe that for a positive recurrent chain, $\lambda_1 p_{m-1}^{(1)} = \mu p_{l+1}^{(2)}$. Thus,

$$R = 2\lambda_1 p_{m-1}^{(1)} = \begin{cases} \frac{2\lambda_1(1-\eta_1)}{1-\eta_1^k} p_l, & \eta_1 \neq 1 \\ \frac{2\lambda_1 p_l}{k}, & \eta_1 = 1 \end{cases}, \tag{2.4}$$

where p_l was found in terms of the system parameters in (2.2).

It is clear from (2.3) and (2.4) that k determines the tradeoff between the congestion probability and rate of control. Specifically, a larger k implies a smaller rate of control, but a larger probability of congestion, and vice versa. Thus, we conclude that for the two-threshold policy, the parameter l dictates the minimum throughput guarantee, while k trades off the congestion probability with rate of control packets. Next, we show that the increase in the congestion probability with k can be made slower than exponential in the buffer size.

2.3.2 Large deviation exponents

In many queueing systems, the congestion probability decays exponentially in the buffer size M. Furthermore, when the buffer size gets large, the exponential term dominates all other sub-exponential terms in determining the decay probability. It is therefore useful to focus only on the exponential rate of decay, while ignoring all other sub-exponential dependencies of the congestion probability on the buffer size M. Such a characterization is obtained by using the large deviation exponent (LDE). For a given control policy, we define the LDE corresponding to the decay rate of the congestion probability as

$$E = \lim_{M\to\infty} -\frac{1}{M} \log \mathbb{P}\{Q \geq M\},$$

when the limit exists. Next we compute the LDE for the two-threshold policy.

Proposition 2.1 *Assume that k scales with M sub-linearly, so that $\lim_{M\to\infty} \frac{k(M)}{M} = 0$. The LDE of the two-threshold policy is then given by*

$$E = \log \frac{1}{\rho_2}. \tag{2.5}$$

The above result follows from the congestion probability expression (2.3) since the only term that is exponential in M is $\rho_2^{M-m} = \rho_2^{M-l-k}$. Note that l is determined based on the throughput requirement, and does not scale with M. We pause to make the following observations:

- If k scales linearly with M as $k(M) = \beta M$ for some constant $\beta > 0$, the LDE becomes
$$E = (1-\beta) \log \frac{1}{\rho_2}.$$

- The control rate (2.4) can be made arbitrarily small, if $k(M)$ tends to infinity. This implies that as long as $k(M)$ grows to infinity sub-linearly in M, we can achieve an LDE that is constant (equal to $-\log \rho_2$) for all rates of control.

- As k becomes large, the congestion probability will increase. However, the increase is only sub-exponential in the buffer size, so that the LDE remains constant.

In what follows, we will be interested only in the LDE corresponding to the congestion probability, rather than its actual value. The following theorem establishes the optimality of the LDE for the two-threshold policy.

Theorem 2.1 *The two-threshold policy has the best possible LDE corresponding to the congestion probability among all flow control policies, for any rate of control.*

This result is a simple consequence of the fact that the two-threshold policy has the same LDE as an M/M/1 queue with the lower input rate λ_2, and the latter clearly cannot be surpassed by any flow control policy.

2.3.3 More general Markovian policies and relationship to RED

In the previous section, we analyzed the two-threshold policy and concluded that it has the optimal congestion probability exponent for any rate of control. This is essentially because the input rate switches to the lower value deterministically, well before the congestion limit M is reached. In this subsection, we show that the two-threshold policy can be easily modified to a more general Markovian policy, which closely resembles the well known RED active queue management scheme [21]. Furthermore, this modification can be done while maintaining the optimal exponent behavior for the congestion probability.

Recall that RED preemptively avoids congestion by starting to drop packets randomly even before the buffer is about to overflow. Specifically, consider two queue thresholds, say l and m, where $m > l$. If the queue occupancy is no more than l, no packets are dropped, no matter what the input rate is. On the other hand, if the queue length reaches or exceeds m, packets are always dropped, which then leads to a reduction in the input rate (assuming that the host responds to dropped packets). If the queue length is between l and m, packets are randomly dropped with some probability q.[1]

Consider the following flow control policy, which closely resembles the RED scheme described above:

For queue lengths less than or equal to l, the higher input rate is always used. If the queue length increases to m while the input rate is λ_1, a congestion notification is sent, and the input rate is reduced to λ_2. If the current input rate is λ_1 and the queue length is between l and m, a congestion notification occurs with probability[2] q upon the arrival of a packet, and the input rate is reduced to λ_2. With probability $1 - q$, the input continues at the higher rate. The Markov process corresponding to this policy is depicted in Figure 2-3.

[1]Often, the dropping probability is dependent on the queue length.

[2]We can also let this probability to depend on the current queue length, as often done in RED, but this makes the analysis more difficult

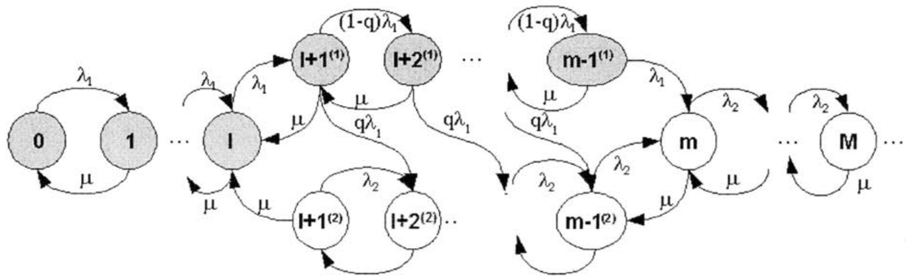


Figure 2-3: The Markov process corresponding to the control policy described in subsection 2.3.3 that approximates RED.

We can derive the tradeoff between the congestion probability and the rate of congestion notifications for this policy by analyzing the Markov chain in Figure 2-3. Once the lower threshold l has been determined from the throughput guarantee, the control rate vs. congestion probability tradeoff is determined by both q and m. Further, since the input rate switches to the lower value λ_2 when the queue length is larger than m, this flow control policy also achieves the optimal LDE for the congestion probability, equal to $\log \frac{1}{\rho_2}$. We skip the derivations for this policy, since it is more cumbersome to analyze than the two-threshold policy, without yielding further qualitative insights. We focus on the two-threshold policy in the remainder of the chapter, but point out that our methodology can also model more practical queue management policies like RED.

2.4 The Effect of Control Errors on Congestion

In this section, we investigate the impact of control errors on the congestion probability of the two-threshold policy. We use a simple probabilistic model for the losses on the control channel. In particular, we assume that any control packet sent by the observer can be lost with some probability δ, independently of other packets. Using the decay exponent tools described earlier, we show the existence of a critical value of the loss probability, say δ^*, beyond which the losses in receiving the control packets lead to an exponential degradation of the congestion probability.

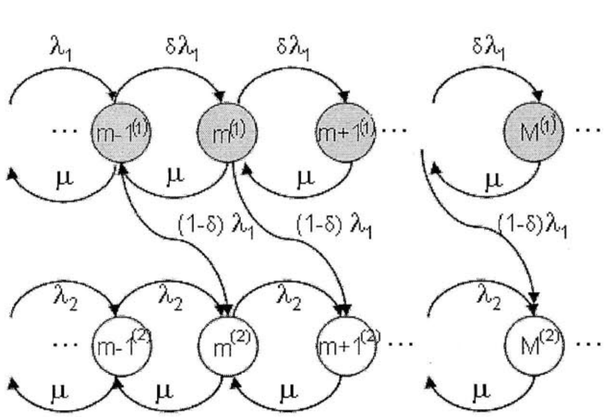


Figure 2-4: The Markov process $Y(t)$ corresponding to the loss-prone two-threshold policy. Only a part of the state space (with $Q(t) \geq M$) is shown.

2.4.1 The two-threshold policy over a loss-prone control channel

As described earlier, in the two-threshold policy, the observer sends a control packet when the queue length reaches $m = l + k$. This packet may be received by the flow controller with probability $1 - \delta$, in which case the input rate switches to λ_2. The packet may be lost with probability δ, in which case the input continues at the higher rate λ_1. We assume that if a control packet is lost, the observer immediately knows about it[3], and sends another control packet the next time an arrival occurs to a system with at least $m - 1$ packets.

The process $Y(t) = (Q(t), A(t))$ is a Markov process even for this loss-prone two-threshold policy. Figure 2-4 shows a part of the state space for the process $Y(t)$, for queue lengths larger than $m - 1$. Note that due to control losses, the input rate does not necessarily switch to λ_2 for queue lengths greater than $m-1$. Indeed, it is possible to have not switched to the lower input rate even for arbitrarily large queue lengths. This means that the congestion limit can be exceeded under both arrival rates, as shown in Figure 2-4. The following theorem establishes the LDE of the loss-prone two-threshold policy, as a function of the loss probability δ.

[3]This is an idealized assumption; in practice, delayed feedback can be obtained using ACKS.

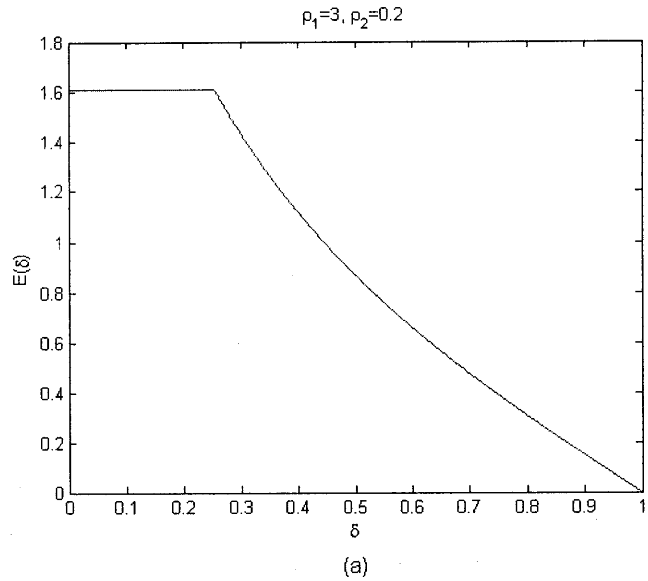


Figure 2-5: LDE as a function of δ for $\rho_1 > 1$.

Theorem 2.2 *Consider a two-threshold policy in which k grows sub-linearly in M. Assume that the control packets sent by the observer can be lost with probability δ. Then, the LDE corresponding to the congestion probability is given by*

$$E(\delta) = \begin{cases} \log \frac{1}{\rho_2}, & \delta \leq \delta^*, \\ \\ \log \frac{2}{1+\rho_1 - \sqrt{(\rho_1+1)^2 - 4\delta\rho_1}}, & \delta > \delta^*, \end{cases} \tag{2.6}$$

where δ^* is the critical loss probability *given by (2.10).*

Before we give a proof of this result, we pause to discuss its implications. The theorem shows that the two-threshold policy over a loss-prone channel has two regimes of operation. In particular, for 'small enough' loss probability ($\delta < \delta^*$), the exponential rate of decay of the congestion probability is the same as in a loss-free system. However, for $\delta > \delta^*$, the decay exponent begins to take a hit, and therefore, the congestion probability suffers an exponential increase. For this reason, we refer to δ^* as the critical loss probability. Figure 2-5 shows a plot of the decay exponent as a function of the loss probability δ, for $\rho_1 > 1$. The 'knee point' in the plot corresponds to δ^* for the stated values of ρ_1 and ρ_2.

Proof: The balance equations for the top set of states in Figure 2-4 can be written as

$$p_{i+1}^{(1)}\mu - (\lambda_1 + \mu)p_i^{(1)} + \delta\lambda_1 p_{i-1}^{(1)}, \ i = m, m+1, \ldots \tag{2.7}$$

Solving the second order recurrence relation above, we find that the top set of states in Figure 2-4 (which correspond to arrival rate λ_1) have steady state probabilities that satisfy

$$p_{m-1+i}^{(1)} = s(\delta)^i p_{m-1}^{(1)}, i = 1, 2, \ldots,$$

where

$$s(\delta) = \frac{1 + \rho_1 - \sqrt{(1+\rho_1)^2 - 4\rho_1\delta}}{2}. \tag{2.8}$$

Similarly, the balance equations for the bottom set of states are

$$p_{i+1}^{(2)}\mu - (\lambda_2 + \mu)p_i^{(1)} + \lambda_2 p_{i-1}^{(2)} + (1-\delta)\lambda_1 p_{i-1}^{(1)}, i = m, m+1, \ldots$$

from which we can deduce that the steady state probabilities have the form

$$p_{m-1+i}^{(2)} = A\rho_2^i + Bs(\delta)^i, i = 1, 2, \ldots,$$

where A, B are constants that depend on the system parameters ρ_1, ρ_2, and δ. Using the two expressions above, we can deduce that the congestion probability has *two* terms that decay exponentially in the buffer size:

$$\mathbb{P}\{Q \geq M\} = Cs(\delta)^{M-l-k} + D\rho_2^{M-l-k}, \tag{2.9}$$

where C, D are constants.

In order to compute the LDE, we need to determine which of the two exponential terms in (2.9) decays slower. It is seen by direct computation that $s(\delta) \leq \rho_2$ for $\delta \leq \delta^*$, where

$$\delta^* = \frac{\rho_2}{\rho_1}(1 + \rho_1 - \rho_2). \tag{2.10}$$

Thus, for loss probabilities less than δ^*, ρ_2 dominates the rate of decay of the congestion probability. Similarly, for $\delta > \delta^*$, we have $s(\delta) > \rho_2$, and the LDE is determined by $s(\delta)$. This proves the theorem. □

Remark 2.1 *Large deviation theory has been widely applied to study congestion and overflow behaviors in queueing systems. Tools such as the Kingman bound [24] can be used to characterize the LDE of any G/G/1 queue. Large deviation framework also exists for more complicated queuing systems, with correlated inputs, several sources, finite buffers etc., see for instance [25]. However, for* controlled *queues, where the input or service rates can vary based on queue length history, simple large deviation formulas do not exist. It is remarkable that for a single server queue with Markovian control, we are able to obtain rather intricate LDE characterizations such as in Figure 2-5, just by applying 'brute force' steady state probability computations.*

2.4.2 Repetition of control packets

Suppose we are given a control channel with a loss probability δ that is greater than the critical loss probability in (2.10). This means that a two-threshold policy operating on this control channel has an LDE in the decaying portion of the curve in Figure 2-5. In this situation, adding error protection to the control packets will reduce the effective probability of loss, thereby improving the LDE. To start with, we consider the simplest form of adding redundancy to control packets, namely repetition.

Suppose that each control packet is transmitted n times by the observer, and that all n packets are communicated without delay. Assume that each of the n packets has a probability δ of being lost, independently of other packets. The flow controller fails to switch to the lower input rate only if all n control packets are lost, making the effective probability of loss δ^n. In order to obtain the best possible LDE, the operating point must be in the flat portion of the LDE curve, which implies that the effective probability of loss should be no more than δ^*. Thus, $\delta^n \leq \delta^*$, so that the number of transmissions n should satisfy

$$n \geq \frac{\log \delta^*}{\log \delta} \tag{2.11}$$

in order to obtain the best possible LDE of $\log \frac{1}{\rho_2}$. If the value of δ is close to 1, the number of repeats is large, and vice-versa.

2.5 Optimal Bandwidth Allocation for Control Signals

As discussed in the previous subsection, the LDE operating point of the two-threshold policy for any given $\delta < 1$, can always be 'shifted' to the flat portion of the curve by repeating the control packets sufficiently many times (2.11). This ignores the bandwidth consumed by the additional control packets.

While the control overheads constitute an insignificant part of the total communication resources in optical networks, they might consume a sizeable fraction of bandwidth in some wireless or satellite applications. In such a case, we cannot add an arbitrarily large amount of control redundancy without sacrificing some service bandwidth. Typically, allocating more resources to the control signals makes them more robust to losses, but it also reduces the bandwidth available to serve data. To better understand this tradeoff, we explicitly model the service rate to be a function of the redundancy used for control signals. We then determine the optimal fraction of bandwidth to allocate to the control packets, so as to achieve the best possible decay exponent for the congestion probability.

2.5.1 Bandwidth sharing model

Consider, for the time being, the simple repetition scheme for control packets outlined in the previous section. We assume that the queue service rate is linearly decreasing function of the number of repeats $n-1$:

$$\mu(n) = \mu \left[1 - \frac{n-1}{\Phi}\right]. \tag{2.12}$$

The above model is a result of the following assumptions about the bandwidth consumed by the control signals:

- μ corresponds to the service rate when no redundancy is used for the control packets ($n = 1$).
- The amount of bandwidth consumed by the redundancy in the control signals is proportional to the number of repeats $n - 1$.
- The fraction of total bandwidth consumed by each repetition of a control packet is equal to $1/\Phi$, where $\Phi > 0$ is a constant that represents how 'expensive' it is in terms of bandwidth to repeat control packets.

Thus, with $n - 1$ repetitions, the fraction of bandwidth consumed by the control information is $\frac{n-1}{\Phi}$, and the fraction available for serving data is $1 - \frac{n-1}{\Phi}$.

Let us denote by f the fraction of bandwidth consumed by the redundancy in the control information, so that $f = \frac{n-1}{\Phi}$, or $n = \Phi f + 1$. From (2.12), the service rate corresponding to the fraction f can be written as

$$\mu(f) = \mu[1 - f].$$

In what follows, we do not restrict ourselves to repetition of control packets, so that we are not constrained to integer values of n. Instead, we allow the fraction f to take continuous values, while still maintaining that the loss probability corresponding to f is $\delta^{\Phi f+1}$. We refer to f as the 'fraction of bandwidth used for control', although it is really the fraction of bandwidth utilized by the *redundancy* in the control. For example, $f = 0$ does not mean no control is used; instead, it corresponds to each control packet being transmitted just once.

2.5.2 Optimal fraction of bandwidth to use for control

The problem of determining the optimal fraction of bandwidth to be used for control can be posed as follows:

Given the system parameters ρ_1, ρ_2 and Φ, and a control channel with some probability of loss $\delta \in [0, 1)$, find the optimal fraction of bandwidth $f^*(\delta)$ to be used for control, so as to maximize the LDE of the congestion probability.

Let us define

$$\rho_i(f) = \frac{\lambda_i}{\mu(f)} = \frac{\rho_i}{1-f}, i = 1, 2, \tag{2.13}$$

as the effective server utilization corresponding to the reduced service rate $\mu(f)$. Accordingly, we can also define the effective knee point as

$$\delta^*(f) = \frac{\rho_2}{\rho_1}\left(1 + \frac{\rho_1 - \rho_2}{1-f}\right), \tag{2.14}$$

which is analogous to (2.10), with $\rho_i(f)$ replacing $\rho_i, i = 1, 2$.

First, observe that for the queueing system to be stable, we need the effective service rate to be greater than the lower input rate λ_2. Thus, we see that $\lambda_2 < \mu[1-f]$, or $f < 1 - \rho_2$. Next, we compute the LDE corresponding to a given probability of loss δ, and fraction f of bandwidth used for control.

Proposition 2.2 *For any $\delta \in [0, 1)$ and $f \in [0, 1 - \rho_2)$, the corresponding LDE is given by*

$$E(\delta, f) = \begin{cases} \log \frac{1}{\rho_2(f)}, & \delta^{\Phi f+1} \leq \delta^*(f), \\ \\ \log \frac{1}{s(f,\delta)}, & \delta^{\Phi f+1} > \delta^*(f), \end{cases} \tag{2.15}$$

where

$$s(f, \delta) = \frac{1 + \rho_1(f) - \sqrt{(\rho_1(f) + 1)^2 - 4\delta^{\Phi f+1}\rho_1(f)}}{2}.$$

The derivation and expression for $E(\delta, f)$ are analogous to (2.6), except that ρ_i is replaced with $\rho_i(f), i = 1, 2$, and δ is replaced with the effective probability of loss $\delta^{\Phi f+1}$.

Definition 2.3 *For any given $\delta \in [0, 1)$, the optimal fraction $f^*(\delta)$ is the value of f that maximizes $E(\delta, f)$ in (2.15). Thus,*

$$f^*(\delta) = \text{argmax}_{f\in[0,1-\rho_2)} E(\delta, f). \tag{2.16}$$

Recall that the value of $1/\Phi$ represents how much bandwidth is consumed by each added repetition of a control packet. We will soon see that Φ plays a key role in determining the optimal fraction of bandwidth to use for control. Indeed, we show that there are three different regimes for Φ such that the optimal fraction $f^*(\delta)$ exhibits qualitatively different behavior in each regime as a function of δ. The three ranges of Φ are: (i) $\Phi \leq \underline{\Phi}$, (ii) $\Phi \geq \overline{\Phi}$, and (iii) $\underline{\Phi} < \Phi < \overline{\Phi}$, where

$$\begin{aligned} \underline{\Phi} &= \frac{\rho_2 - \delta^*}{\log(\delta^*)(1+\rho_1-\rho_2)}, \\ \overline{\Phi} &= \begin{cases} \frac{1}{\rho_1 - 1} & \rho_1 > 1 \\ \infty & \rho_1 \leq 1 \end{cases}. \end{aligned} \tag{2.17}$$

It can be shown that $\underline{\Phi} < \overline{\Phi}$ for $\rho_2 < 1$.

We shall refer to case (i) as the 'small Φ regime', case (ii) as the 'large Φ regime', and case (iii) as the 'intermediate regime'. We remark that whether a value of Φ is considered 'small' or 'large' is decided entirely by ρ_1 and ρ_2. Note that the large Φ regime is non-existent if $\rho_1 \leq 1$, so that even if Φ is arbitrarily large, we would still be in the intermediate regime.

The following theorem, which is our main result for this section, specifies the optimal fraction of bandwidth $f^*(\delta)$, for each of the three regimes for Φ.

Theorem 2.3 *For a given ρ_1 and ρ_2, the optimal fraction of bandwidth $f^*(\delta)$ to be used for control, has one of the following forms, depending on the value of Φ:*

(i) Small Φ regime ($\Phi \leq \underline{\Phi}$): $f^*(\delta) = 0, \ \forall \ \delta \in (0,1)$.

(ii) Large Φ regime ($\Phi \geq \overline{\Phi}$):

$$f^*(\delta) = \begin{cases} 0, & \delta \in [0, \delta^*] \\ \hat{f}(\delta), & \delta \in (\delta^*, 1) \end{cases},$$

where $\hat{f}(\delta)$ is the unique solution to the transcendental equation

$$\delta^{\Phi \hat{f}+1} = \frac{\rho_2}{\rho_1}\left(1 + \frac{\rho_1 - \rho_2}{1-\hat{f}}\right). \tag{2.18}$$

(iii) Intermediate regime ($\underline{\Phi} < \Phi < \overline{\Phi}$): there exist δ' and δ'' such that $\delta^* < \delta' <$

$\delta'' < 1$, and the optimal fraction is given by

$$f^*(\delta) = \begin{cases} 0, & \delta \in [0, \delta^*] \\ \hat{f}(\delta), & \delta \in (\delta^*, \delta') \\ \tilde{f}(\delta), & \delta \in (\delta', \delta'') \\ 0, & \delta \in (\delta'', 1) \end{cases},$$

where $\hat{f}(\delta)$ is given by (2.18) and $\tilde{f}(\delta)$ is the unique solution in $(0, 1 - \rho_2)$ to the transcendental equation

$$\delta^{\Phi \tilde{f}+1}[\Phi(1-\tilde{f}) \log \delta^* + 1] = s(\tilde{f}, \delta). \tag{2.19}$$

The proof of the above theorem is not particularly interesting, and is postponed to Chapter Appendix 2.A. Instead, we provide some intuition about the optimal solution.

2.5.3 Discussion of the optimal solution

Loss probability less than δ^*

In all three regimes, we find that $f^*(\delta) = 0$ for $\delta \in [0, \delta^*]$. This is because, as shown in Figure 2-5, the LDE has the highest possible value of $-\log \rho_2$ for δ in this range, and there is nothing to be gained from adding any control redundancy.

Small Φ regime

In case (i) of the theorem, it is optimal to not apply any control redundancy at all. That is, the best possible LDE for the congestion probability is achieved by using a single control packet every time the observer intends to switch the input rate. In this regime, the amount of service bandwidth lost by adding any control redundancy at all, hurts us more than the gain obtained from the improved loss probability. The plot of the optimal LDE as a function of δ for this regime is identical to Figure 2-5, since no redundancy is applied.

Large Φ regime

Case (ii) of the theorem deals with the large Φ regime. For $\delta > \delta^*$, the optimal $f^*(\delta)$ in this regime is chosen as the fraction f for which the knee point $\delta^*(f)$ equals the effective loss probability $\delta^{\Phi f+1}$. This fraction is indeed $\hat{f}$, defined by (2.18). Figure 2-6(a) shows a plot of the optimal fraction (solid line) as a function of δ. In this example, $\rho_1 = 1.2,\ \rho_2 = 0.3$, and $\Phi = 10$. The resulting optimal LDE is equal to $\log \frac{1-\hat{f}(\delta)}{\rho_2}$ for $\delta > \delta^*$. The optimal LDE is shown in Figure 2-6(b) with a solid line.

Comparison with naïve repetition

It is interesting to compare the optimal solution in the large Φ regime to the 'naïve' redundancy allocation policy mentioned in Equation (2.11). Recall that the naïve policy simply repeats the control packets to make the effective loss probability equal to the critical probability δ^*, without taking into account any service bandwidth penalty that this might entail. Let us see how the naïve strategy compares to the optimal solution if the former is applied to a system with a finite Φ. This corresponds to a network with limited communication resources in which the control mechanisms are employed without taking into account the bandwidth that they consume.

The fraction of bandwidth occupied by the repeated control packets can be found using (2.11) to be

$$f = \frac{1}{\Phi}\left(\frac{\log \delta^*}{\log \delta} - 1\right),$$

where we have ignored integrality constraints on the number of repeats. A plot of this fraction is shown in Figure 2-6(a), and the corresponding LDE in Figure 2-6(b), both using dashed lines. As seen in the figure, the naive strategy is more aggressive in adding redundancy than the optimal strategy, since it does not take into account the loss in service rate ensuing from the finiteness of Φ. The LDE of the naïve strategy is strictly worse for $\delta > \delta^*$. In fact the naïve strategy causes instability effects for some values of δ close to 1 by over-aggressive redundancy addition, which throttles the service rate $\mu(f)$ to values below the lower arrival rate λ_2. This happens at the point where the LDE reaches zero in Figure 2-6(b). The naïve strategy has even worse

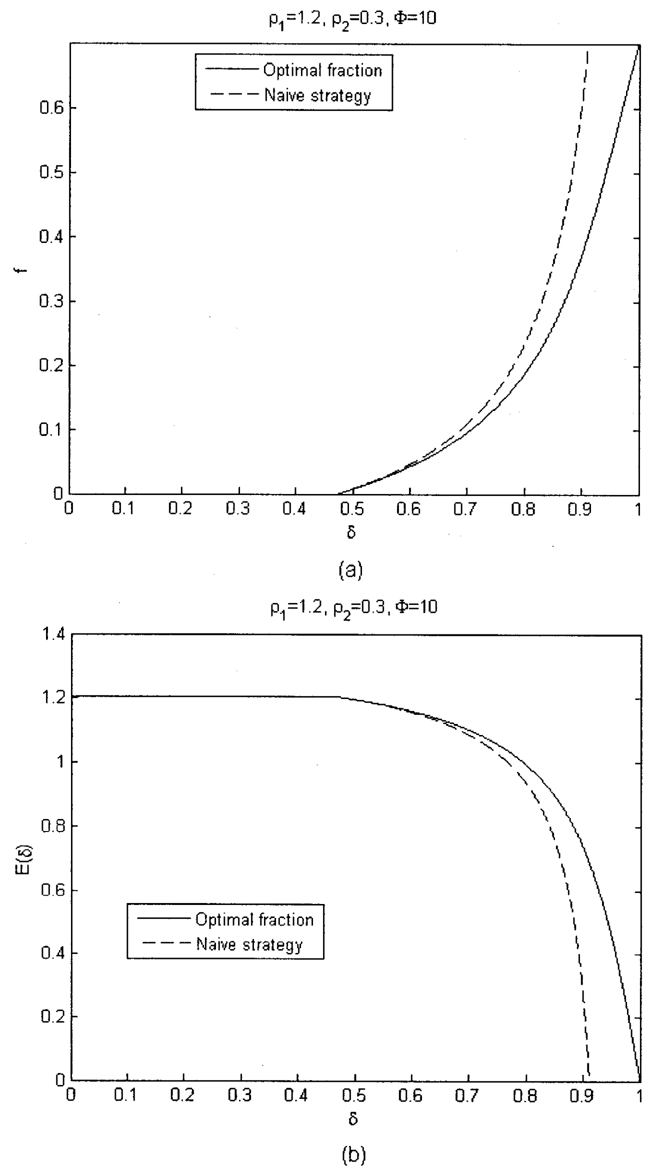


Figure 2-6: (a) Optimal fraction $f^*(\delta)$ for the large Φ regime, and the fraction for the naïve strategy (b)The corresponding LDE curves

consequences in the other two regimes. However, we point out that the repetition strategy approaches the optimal solution as Φ becomes very large.

Intermediate regime

Case (iii) in the theorem deals with the intermediate regime. For $\delta > \delta^*$, the optimal fraction begins to increase along the curve $\hat{f}(\delta)$ exactly like in the large Φ regime (see Figure 2-7). That is, the effective loss probability is made equal to the knee point. However, at a particular value of loss probability, say δ', the optimal fraction begins to decrease sharply from the $\hat{f}(\delta)$ curve, and reaches zero at some value δ''. Equation (2.19) characterizes the optimal fraction for values of δ in (δ', δ''). No redundancy is applied for $\delta \in (\delta'', 1)$. For this range of loss probability, the intermediate regime behaves more like the small Φ regime (case(i)). Thus, the intermediate Φ regime resembles the large Φ regime for small enough loss probabilities $\delta < \delta'$, and the small Φ regime for large loss probability $\delta > \delta''$. There is also a non empty 'transition interval' in between the two, namely (δ', δ'').

2.6 Queue Length Information and Server Allocation

In this section, we discuss the role of queue length information on server allocation policies in a single server queue. We mentioned earlier that queue aware resource allocation policies tend to allocate a higher service rate to longer queues, and vice-versa. Intuitively, if the controller is frequently updated with accurate queue length information, the service rate can be adapted to closely reflect the changing queue length. However, if the queue length information is infrequently conveyed to the controller, we can expect a larger queue length variance, and hence a higher probability of congestion. We study this tradeoff between the probability of congestion and the rate of queue length information in a single server queue.

Figure 2-8 depicts a single server queue with Poisson inputs of rate λ. An observer

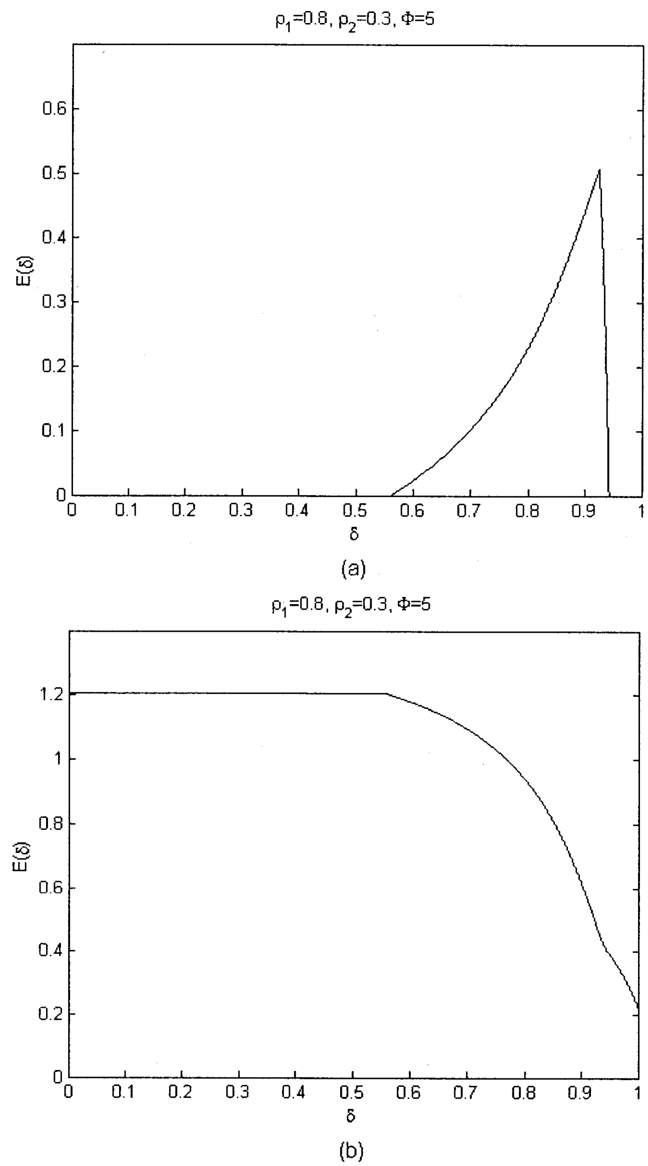


Figure 2-7: (a) Optimal control fraction $f^*(\delta)$ for the intermediate regime (b) The corresponding LDE

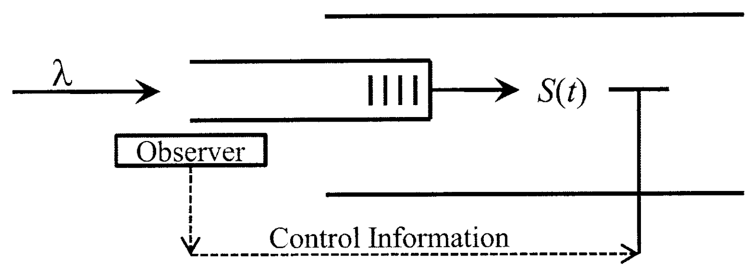


Figure 2-8: A single server queue with service rate control.

watches the queue evolution and sends control information to the to the service rate controller, which changes the service rate $S(t)$ based on the control information it receives. The purpose of the observer-controller subsystem is to assign service rates at each instant so as to control congestion in the queue.

For analytical simplicity, we assume that the service rate at any instant is chosen to be one of two distinct values: $S(t) \in \{\mu_1, \mu_2\}$, where $\mu_2 > \mu_1$ and $\mu_2 > \lambda$. The control decisions are sent by the observer in the form of information-less packets. Upon receiving a control packet, the rate controller switches the service rate from one to the other. As before, we only focus on Markovian control policies, which are defined analogously to (2.1).

Note that if there is no restriction imposed on using the higher service rate μ_2, it is optimal to use it all the time, since the congestion probability can be minimized without using any control information. However, in a typical queueing system with limited resources, it may not be possible to use higher service rate at all times. There could be a cost per unit time associated with using the faster server, which restricts its use when the queue occupancy is high. Alternately, one could explicitly restrict the use of the faster server by allowing its use only when the queue occupancy is over a certain threshold value. In this section, we impose the latter constrain, i.e., when the queue length is no more than some threshold l, we are forced to use the lower service rate μ_1. If the queue length exceeds l, we are allowed to use the higher rate μ_2 without any additional cost until the queue length falls back to l.[4]

It turns out that this model is, in a certain sense, dual to the flow control problem

[4]Recall that in the flow control problem, the threshold l was derived from a throughput constraint.

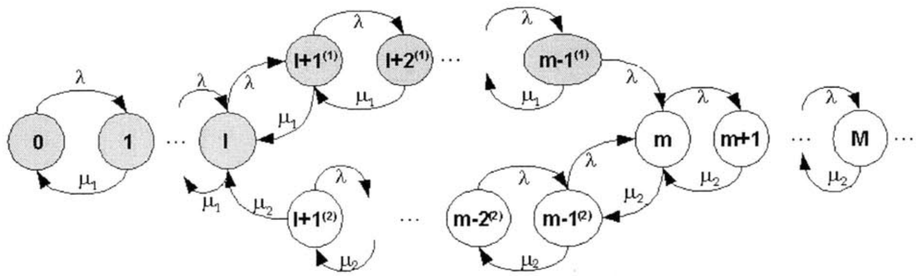


Figure 2-9: The Markov process corresponding to the two-threshold server allocation policy.

considered earlier in this chapter. In fact, for every Markovian flow control policy operating on the queue in Figure 2-1, it is possible to identify a corresponding server allocation policy which has identical properties. For example, we can define a two-threshold server allocation policy analogously to the flow control policy as follows:

The service rates are switched at two distinct queue length thresholds l and m. Specifically, when the queue length grows past m, the service rate switches to μ_2. Once the higher service rate is employed, it is maintained until the queue length falls back to l, at which time the service rate switches back to μ_1. The Markov process corresponding to the two-threshold server allocation policy is depicted in Figure 2-9.

Evidently, the Markov chain in Figure 2-9 has the same structure as the chain in Figure 2-2, and therefore can be analyzed in the same fashion. In particular, we can derive the control rate vs. congestion probability tradeoff, along the same lines as Equations (2.3) and (2.4). The following result regarding the two-threshold server allocation policy, can be derived along the lines of Proposition 2.1 and Theorem 2.1.

Theorem 2.4 *Suppose that k goes to infinity sub-linearly in the buffer size M, in a two-threshold server allocation policy. Then, the LDE can be maintained constant at*

$$E = \log \frac{\mu_2}{\lambda},$$

while the control rate can be made arbitrarily small. Further, the two-threshold policy has the largest possible congestion probability LDE among all server allocation policies, for any rate of control.

The above result shown the optimality of the two-threshold policy with respect to the congestion probability exponent. Next, if the control signals that lead to switching the service rate are subject to losses (as detailed in Section 2.4), we can show that the LDE behaves exactly as in Theorem 2.2, with the critical loss probability given by (2.10).

In essence, we conclude that both flow control and resource allocation problems in a single server queue lead to the same mathematical framework, and can thus be treated in a unified fashion.

2.7 Conclusions

The goal of this chapter was to study the role played by queue length information in flow control and resource allocation policies. Specifically, we deal with the question of how often queue length information needs to be conveyed in order to effectively control congestion. To our knowledge, this is the first attempt to analytically study this particular tradeoff. Since this tradeoff is difficult to analyze in general networks, we consider a simple model of a single server queue in which the control decisions are based on the queue occupancy. We learned that in the absence of control channel losses, the control rate needed to ensure the optimal decay exponent for the congestion probability can be made arbitrarily small. However, if control channel losses occur probabilistically, we showed the existence of a critical loss probability threshold beyond which the congestion probability undergoes a drastic increase due to the frequent loss of control packets. Finally, we determine the optimal amount of error protection to apply to the control signals by using a simple bandwidth sharing model. For loss probabilities larger than the critical value, a significant fraction of the system resources may be consumed by the control signals, unlike in the loss free scenario. We also pointed out that allocating control resources without considering the bandwidth they consume, might have adverse effects on congestion. Finally, we observed that the sever allocation problem and the flow control problem can be treated in a mathematically unified manner.

2.A Proof of Theorem 2.3

Given Φ and δ we want to find the fraction f that satisfies (2.16). As shown in figure 5, the LDE curve for $f = 0$ is flat and has the highest possible value of $-\log \rho_2$ for $\delta \in [0, \delta^*]$. Indeed, for δ in the above range, using any strictly positive fraction f would reduce the LDE to $\log \frac{1-f}{\rho_2}$. This implies that the optimal fraction

$$f^*(\delta) = 0, \;\; \delta \in [0, \delta^*].$$

Thus, the problem of finding the optimal $f^*(\delta)$ is non-trivial only for $\delta \in (\delta^*, 1)$. We begin our exposition regarding the optimal fraction with two simple propositions.

Proposition 2.3 *For any given $\delta \in (\delta^*, 1)$, the optimal fraction $f^*(\delta)$ is such that the effective loss probability $\delta^{\Phi f^*+1}$ cannot be strictly lesser than the knee point of the curve $E(\delta, f^*)$. That is, $\delta^*(f^*) \leq \delta^{\Phi f^*+1}$, $\delta \in (\delta^*, 1)$.*

Proof: Suppose the contrary, i.e, for some $\delta \in (\delta^*, 1)$, the optimal f^* is such that $\delta^*(f^*) > \delta^{\Phi f^*+1}$. The optimal LDE would then be $E(\delta, f^*) = -\log \rho_2(f^*) = \log \frac{1-f^*}{\rho_2}$. Continuity properties imply that $\exists\ \xi > 0$ for which $\delta^*(f^* - \xi) > \delta^{\Phi(f^*-\xi)+1}$. Thus, if we use the smaller fraction $f^* - \xi$ for control, the LDE would be $E(\delta, f^* - \xi) = \log \frac{1-f^*+\xi}{\rho_2}$. Since this value is greater than the "optimal value" $E(\delta, f^*)$, we arrive at a contradiction. $\square$

Proposition 2.4 *For a given $\delta \in (\delta^*, 1)$, there exists a unique fraction $\hat{f}(\delta) \in (0, 1-\rho_2)$ such that the knee point $\delta^*(\hat{f})$ equals the effective loss probability $\delta^{\Phi \hat{f}+1}$. Furthermore, the optimal fraction $f^*(\delta)$ lies in the interval $[0, \hat{f}(\delta)]$.*

Proof: The knee point corresponding to any fraction f is given by (2.14). Therefore, if there exists a fraction $\hat{f}$ for which $\delta^*(\hat{f}) = \delta^{\Phi \hat{f}+1}$, then $\hat{f}$ satisfies

$$\delta^{\Phi \hat{f}+1} = \frac{\rho_2}{\rho_1}\left(1 + \frac{\rho_1 - \rho_2}{1 - \hat{f}}\right). \tag{2.20}$$

The transcendental equation in (2.20) has a solution in $(0, 1-\rho_2)$ for any given $\delta \in (\delta^*, 1)$. This can be shown by applying the intermediate value theorem to the difference of the right and left hand side functions in the equation. The uniqueness of $\hat{f}$ follows from the monotonicity properties of the right and left hand side functions. $\square$

To prove the second statement, suppose that $f^* > \hat{f}$. Since the knee point (2.14) is monotonically strictly increasing in f, we have $\delta^*(f^*) > \delta^*(\hat{f}) = \delta^{\Phi\hat{f}+1} > \delta^{\Phi f^*+1}$. This contradicts Proposition 2.3. $\square$

The above proposition shows that the optimal fraction lies in the interval $[0, \hat{f}(\delta)]$. Thus, for a given $\delta > \delta^*$, we seek $f^*(\delta) \in [0, \hat{f}(\delta)]$ for which $s(f, \delta)$ (defined in Proposition 2.2) is minimized. In particular, if $s(f, \delta)$ is monotonically decreasing in $[0, \hat{f}(\delta)]$ for some δ, then clearly, $f^*(\delta) = \hat{f}(\delta)$. The following proposition asserts the condition under which $s(f, \delta)$ is monotonically decreasing.

Proposition 2.5 *For some $\delta > \delta^*$, suppose the following inequality holds*

$$\delta^{\Phi\hat{f}+1}[\Phi(1-\hat{f})\log\delta + 1] \leq \frac{\rho_2}{1-\hat{f}}. \tag{2.21}$$

Then, $f^*(\delta) = \hat{f}(\delta)$.

Proof: Fix $\delta > \delta^*$. By direct computation, we find that

$$s'(f,\delta) \leq 0 \iff \delta^{\Phi f+1}[\Phi(1-f)\log\delta + 1] \leq s(f,\delta).$$

However, since the left side of the inequality above is strictly increasing in f, we find that $s(f, \delta)$ is monotonically decreasing whenever

$$\delta^{\Phi\hat{f}+1}[\Phi(1-\hat{f})\log\delta + 1] \leq s(\hat{f}(\delta), \delta) = \frac{\rho_2}{1-\hat{f}},$$

where the last equality follows from the definition of $\hat{f}$. Thus, if (2.21) is satisfied for a particular δ, $s(f, \delta)$ is decreasing in f, and hence the optimal fraction is given by $f^*(\delta) = \hat{f}(\delta)$. $\square$

Now, suppose that $\Phi > \underline{\Phi}$. Upon rearrangement, this implies that $\delta^*[\Phi\log\delta^* + 1] < \rho_2$. In other words, (2.21) is satisfied with strict inequality, at $\delta = \delta^*$. By continuity, we can argue that there exist a range $\delta \in (\delta^*, \delta')$ for which (2.21) is satisfied. By Proposition 2.5, we have $f^*(\delta) = \hat{f}(\delta)$, $\delta \in (\delta^*, \delta')$, which partially proves part (iii) of the theorem. Note that δ' is the smallest value of the loss probability, if any, for which the strict monotonicity of $s(f, \delta)$ (as a function of f) is compromised. As argued in Proposition 2.5, this implies (2.21) holds with equality. A simple rearrangement yields a transcendental equation for δ'

$$\frac{\rho_1}{\rho_1 - \rho_2 + 1 - \hat{f}(\delta')} = 1 + \Phi(1 - \hat{f}(\delta')) \log \delta' \tag{2.22}$$

Using basic calculus, it is possible to show that there always exists a solution $\delta' < 1$ to (2.22) if $\rho_1 < 1$. However, if $\rho_1 > 1$, there exists a solution iff $\Phi < \frac{1}{\rho_1 - 1} = \overline{\Phi}$. In particular, if $\Phi > \overline{\Phi}$, we have that $s(f, \delta)$ is monotonically decreasing in f for all δ, so that $f^*(\delta) = \hat{f}(\delta)$ for all $\delta \in (\delta^*, 1]$. This proves part (ii) of the theorem.

On the other hand suppose that $\underline{\Phi} < \Phi < \overline{\Phi}$. (It is straightforward to show that $\underline{\Phi} < \overline{\Phi}$.) Then a solution $\delta' < 1$ to (2.22) exists, and for $\delta > \delta'$, the optimal fraction is no longer equal to $\hat{f}(\delta)$. It can also be shown that the existence of $\delta' < 1$ guarantees the existence of another $\delta'' > \delta'$ such that $s(f, \delta)$ is *increasing* in f for each $\delta \in (\delta'', 1)$. In such a case, $f^*(\delta)$ would be equal to zero.

Proposition 2.6 *For some $\delta > \delta^*$, suppose the following inequality holds*

$$\delta[\Phi \log \delta + 1] \geq \frac{1 + \rho_1 - \sqrt{(1 + \rho_1)^2 - 4\delta\rho_1}}{2}. \tag{2.23}$$

Then, $f^*(\delta) = 0$.

The proof is similar to Proposition 2.5. The value of δ'' is obtained as a solution to the transcendental equation

$$\delta''[\Phi \log \delta'' + 1] = \frac{1 + \rho_1 - \sqrt{(1 + \rho_1)^2 - 4\delta''\rho_1}}{2} \tag{2.24}$$

Again using basic calculus, we can show there exits a solution $\delta'' < 1$, to (2.24) if $\Phi < \overline{\Phi}$, which is the same condition as for the existence of δ'. Thus, in the intermediate regime, there exists (i) $\delta' \in (\delta^*, 1)$ such that $f^*(\delta) = \hat{f}(\delta)$ for $\delta^* < \delta < \delta'$, and (ii) $\delta'' > \delta'$ such that $f * (\delta) = 0$ for $\delta \geq \delta''$. For $\delta \in (\delta', \delta'')$, the function $s(f, \delta)$ has a minimum in $(0, \hat{f})$, so that the optimum fraction is obtained by setting $s'(f, \delta)$ to zero. This condition is the same as (2.19), and part (iii) of the theorem is thus proved.

We finally show that it is optimal to not use any redundancy in the small Φ regime.

Proposition 2.7 *For $\Phi \leq \underline{\Phi}$, $s(f, \delta)$ is monotone increasing in $f \in [0, \hat{f}(\delta)]$ for each $\delta > \delta^*$.*

Proof: Simple rearrangement shows that the condition $\Phi \leq \underline{\Phi}$ is equivalent to saying that $\delta'' \leq \delta^*$. Since no redundancy is used for values of loss probability greater than δ'', it follows that no redundancy is used in the small Φ regime. □

Chapter 3

The Impact of Queue Length Information on Buffer Overflow in Parallel Queues

3.1 Introduction

Scheduling is an essential component of any queueing system where the server resources need to be shared between many queues. The most basic requirement of a scheduling algorithm is to ensure the stability of all queues in the system, whenever feasible. Much research work has been reported on throughput optimal scheduling algorithms that achieve stability over the entire capacity region of a network [45, 64]. While stability is an important and necessary first-order metric, most practical queueing systems have more stringent Quality of Service (QoS) requirements. For example, voice and video streams are delay sensitive. Further, due to the finiteness of the buffers in practical systems, maintaining a low buffer overflow probability is an important objective.

In this chapter, we consider a system consisting of N parallel queues and a single server. A scheduling policy decides which of the queues gets service in each time slot. Our aim is to better understand the relationship between the amount of queue length information required to operate a scheduling policy, and the corresponding buffer overflow probability. The scheduling decisions may take into account the current queue lengths in the system, in which case we will call the policy 'queue-aware.' If the scheduling decisions do not depend on the current queue lengths, except to the extent of knowing whether or not a queue is empty, we will call it a 'queue-blind' policy.

We analyze the buffer overflow probability of the widely studied max-weight scheduling policy in the large buffer regime. In our simple setting, max-weight scheduling amounts to serving the longest queue at every instant, and we refer to it as the Longest Queue First (LQF) policy. We assume that the queues are fed by statistically identical arrival processes. However, the input statistics could otherwise be quite general. Under such a symmetric traffic pattern, we show that the large deviation exponent of the buffer overflow probability under LQF scheduling is expressible purely in terms of the *total system occupancy* exponent of an m queue system, where $m \leq N$ is determined by the input statistics. We also characterize the likeliest overflow trajectories, and show that there are at most N possible overflow modes that dominate.

Although any non-idling policy (such as LQF, processor sharing (PS) or random scheduling) will achieve the same throughput region and total system occupancy distribution, the LQF policy outperforms queue-blind policies in terms of the buffer overflow probability. Equivalently, this implies that the buffer requirements are lower under LQF scheduling than under queue-blind scheduling, if we want to achieve a given overflow probability. For example, our study indicates that under Bernoulli and Poisson traffic, the buffer size required under LQF scheduling is only about 55% of that required under random scheduling, when the traffic is relatively heavy. On the other hand, with LQF scheduling, the scheduler needs queue length information in *every* time slot, which leads to a significant amount of control signalling. Motivated by this, we identify a "hybrid" scheduling policy, which achieves the same buffer overflow exponent as the LQF policy, with arbitrarily infrequent queue length information.

3.1.1 Related work

To our knowledge, Bertsimas, Paschalidis and Tsitsiklis [6] were among the first to analyze the large deviations behavior of parallel queues. They consider the case of *two* parallel queues, and characterize the buffer overflow exponents under two important service disciplines, namely Generalized Processor Sharing (GPS) and Generalized Longest Queue First (GLQF). We also refer to the related papers [56, 67, 76] where the authors analyze a system of parallel queues, with deterministic arrivals and time-varying connectivity. In [60], the authors study large deviations for the largest weighted delay first policy, and [61] deals with large deviations of max-weight scheduling for general convex rate regions. In each case, the optimal exponent and the likeliest overflow trajectory are obtainable by solving a variational control problem. Often, the optimal solution to the variational problem can be found by solving a finite dimensional optimal control problem [6, 60]. In [65, 66], an interesting link is established between large deviation optimality, and Lyapunov drift minimizing scheduling policies. Large deviations of total end-to-end buffer overflow probability in a multi-hop network with fixed routes is studied in [68].

The remainder of this chapter is organized as follows. In Section 3.2, we present the system description, and some preliminaries on large deviations. Our main result on the large deviation behavior of LQF scheduling is presented in Section 3.3. Section 3.4 compares LQF scheduling to queue-blind scheduling in terms of the overflow probability and buffer scaling. In Section 3.5, we study scheduling with infrequent queue length information.

3.2 System Description and Preliminaries

Figure 3-1 depicts a system consisting of N parallel queues, served by one server. We assume that time is slotted, and the server is capable of serving one packet per slot. Within each queue, packets are served on a first come, first served (FCFS) basis. Arrivals occur according to a random process $A_i[t]$, $i = 1, \ldots, N$, which denotes the number of packets that arrive at queue i during slot t. The arrival processes to the

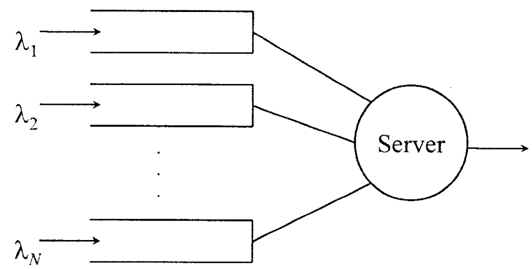


Figure 3-1: N parallel queues served by one server

different queues are independent. We assume a symmetric traffic pattern, i.e., the arrival processes to each queue are statistically identical to each other. For ease of exposition, let us assume that the arrivals are independent across time slots, although our results hold under more general assumptions[1]. The average arrival rate to a queue is $\mathbb{E}[A_i[t]] = \lambda$ packets/slot for each i. For stability, we assume that the condition $\lambda < \frac{1}{N}$ is satisfied. Let us also define

$$S_i[t_1, t_2] = \sum_{\tau=t_1}^{t_2} A_i[\tau], \; t_1 \leq t_2$$

as the number of arrivals to queue i between time slots t_1 to t_2.

The log-moment generating function of the input process to each queue, defined by[2]

$$\Lambda(\theta) = \log \mathbb{E}\left[\exp(\theta A_i[t])\right].$$

is assumed to exist for some $\theta > 0$. The *Fenchel-Legendre transform* or the *convex dual* of $\Lambda(\theta)$ is defined by

$$\Lambda^*(x) = \sup_{\theta}[\theta x - \Lambda(\theta)]. \tag{3.1}$$

$\Lambda^*(x)$ is referred to as the *rate function* of the large deviation principle (LDP) satisfied by each input process.

[1]We only need the input processes to satisfy a sample path large deviation principle (LDP), as detailed in [6].

[2]This definition applies when the inputs are independent across time. If the inputs are correlated across time slots, we define $\Lambda(\theta) = \lim_{n\to\infty} \frac{1}{n} \log \mathbb{E}\left[\exp(\theta \sum_{t=1}^{n} A_i[t])\right]$.

We are interested in the steady state probability of a buffer overflow, under a given scheduling policy Π. The queues are assumed to be initialized such that $Q_i[-MT] = 0, 1 \leq i \leq N$ for some $T > 0$. As $M \to \infty$ with T fixed, the distribution of $Q_i[0]$ will approach the steady-state distribution. More specifically, we are interested in the exponent of the above probability under the large-buffer scaling, which is defined as

$$E_N^{\Pi} = \lim_{M \to \infty} -\frac{1}{M} \log \mathbb{P} \left\{ \max_{i=1,\ldots,N} Q_i[0] \geq M \right\}, \tag{3.2}$$

when the limit exists. We emphasize that this exponent depends on the scheduling policy Π, as well as the system size N and the input statistics. We also define the exponent corresponding to the *total system occupancy* exceeding a certain limit:

$$\Theta_N = \lim_{q \to \infty} -\frac{1}{q} \log \mathbb{P} \left\{ \sum_{i=1}^{N} Q_i[0] \geq q \right\}. \tag{3.3}$$

The system occupancy exponent in (3.3), which can be shown to exist, plays an important role in our analysis of the buffer overflow exponent (3.2) under the LQF policy. The following well known lemma asserts that Θ_N is the same for all *non-idling* scheduling policies.

Lemma 3.1 *All non-idling policies achieve the same steady-state system occupancy distribution (and hence the same system exponent Θ_N).*

In fact, the above result holds at a sample-path level, since one packet would leave the system every time slot if the system is not empty, under any non-idling policy.

We mainly analyze the Longest Queue First (LQF) scheduling policy, which, as the name implies, serves the longest queue in each slot, with an arbitrary tie-breaking rule. We also consider two other non-idling policies: *random scheduling* (RS), which serves a random occupied queue in each slot (each with equal probability), and *processor sharing* (PS), which divides the server capacity equally between all occupied queues. Note that LQF scheduling is queue-aware, while RS and PS are queue-blind.

3.3 Large Deviation Analysis of LQF Scheduling

In this section, we present our main results regarding the buffer overflow exponents and trajectories under LQF scheduling. We begin by characterizing the system occupancy exponent Θ_N for a non-idling policy.

Proposition 3.1 *Under any non-idling policy, the system occupancy exponent is given by*

$$\Theta_N = \inf_{a>0} \frac{1}{a} \Lambda^* \left(a + \frac{1}{N} \right) \tag{3.4}$$

Proof: The result is a consequence of the fact that the total system occupancy distribution is the same as the queue length distribution of a single queue, served by the same server, but fed by the sum process $\sum_i A_i[t]$. Since the input processes to the different queues are independent and identically distributed (i.i.d), the log-moment generating function of the sum process is $N\Lambda(\theta)$. Next, from the definition of the convex dual, the rate function of the sum process can be expressed as $N\Lambda^*(x/N)$. Once the rate function of the input process is known, the overflow exponent of a single server queue can be easily computed; see [25]. □

Let us denote by a_N^* the optimizing value of a in (3.4).

We now define scaled processes for the arrivals and queue lengths, which are often used to study sample path large deviations in the large buffer regime. For every sample path that leads to a buffer overflow at time slot 0, there exists a time $-n \leq 0$ for which both queues are empty. Since we are interested in large M asymptotics, we let $T = -\frac{n}{M}$, and define the sequence of scaled queue length processes

$$q_i^{(M)}(t) = \frac{Q_i[Mt]}{M}, i = 1, \ldots, N,$$

if $t \in \{\frac{-n}{M}, \ldots, \frac{-1}{M}, 0\}$, and by linear interpolation otherwise. Similarly, we define a scaled version of the cumulative arrival processes

$$S_i^{(M)}(t) = \frac{S_i[-MT, Mt]}{M},$$

if $t \in \{\frac{-n}{M}, \ldots, \frac{-1}{M}, 0\}$, and by linear interpolation otherwise. The initial condition implies that $q_i(-T) = 0, i \leq N$. Under the above scaling, $q_i(0) \geq 1$ corresponds to the overflow of queue i at time 0. The functions $S_i^{(M)}(t)$ are Lipschitz continuous maps from $[-T, 0]$ to $\mathbb{R}^N$, and therefore posses derivatives almost everywhere. We define the sequence of empirical rates of the input process on queue i as $x_i^{(M)}(t) = \dot{S}_i^{(M)}(t)$ whenever the derivative exists. Mogulskii's theorem [18] asserts that the sequence of maps $S_i^{(M)}(\cdot)$ satisfies a sample path LDP in the space of maps from $[-T, 0]$ to $\mathbb{R}^N$ equipped with a supremum norm. Further, the rate function of the sample path LDP is given by[3]

$$I_T(S(\cdot)) = \int_{-T}^{0} \Lambda^*(x_i(t))\mathrm{d}t. \tag{3.5}$$

We remark that Mogulskii's theorem applies directly only to arrivals processes which are independent across time. If the arrivals are correlated in time, the results in this chapter will still apply, if we take Equation (3.5) as a starting point. That is, we need to *assume* that the arrival process to each queue satisfies a sample path LDP with rate function given by (3.5).

Next, the sequence of scaled queue lengths $q_i^{(M)}(t)$, $t \in [-T, 0]$ is uniformly bounded (by unity), and is Lipschitz continuous. Thus, there exists a subsequence that converges uniformly over the compact set $[-T, 0]$. (This is a consequence of the Arzelá-Ascoli theorem, see [53]). Let us denote this uniform limit by $q_i(t)$, $i = 1, \ldots, N$, and refer to these functions as the fluid limit of the queue lengths.

We now specify the evolution of the fluid limit $q_i(t)$, $i = 1, \ldots, N$, under LQF scheduling. First, note that for a given sample path and a fixed t, the N-tuple of the queue length fluid limits $(q_1(t), \ldots, q_N(t))$ is a point in $[0, 1]^N$. Next, let $\mathcal{I}$ be any non-empty subset of $\{1, 2, \ldots, N\}$. We define $\mathcal{R}_{\mathcal{I}}$ as the subset of $[0, 1]^N$, such that $(q_1(t), \ldots, q_N(t)) \in \mathcal{R}_{\mathcal{I}}$ iff $q_i(t) = q_j(t)\ \forall i, j \in \mathcal{I}$, and for any $k \notin \mathcal{I}$, $j \in \mathcal{I}$, $q_k(t) < q_j(t)$. Intuitively, in region $\mathcal{R}_{\mathcal{I}}$, the queues in the index set $\mathcal{I}$ 'grow together',

[3] The rate function expression holds only for absolutely continuous maps $S(\cdot)$, and is infinite otherwise. However, $S_i^{(M)}(t)$ are Lipschitz continuous, and Lipschitz continuity implies absolute continuity.

and all other queues are smaller. It is clear that the regions $\mathcal{R}_{\mathcal{I}}$ are convex, and constitute a partition of the set $[0,1]^N$ as $\mathcal{I}$ ranges over all non-empty index sets. The queue evolution equation in region $\mathcal{R}_{\mathcal{I}}$ is given by

$$\begin{aligned} \sum_{i\in\mathcal{I}} \dot{q}_i(t) &= \sum_{i\in\mathcal{I}} x_i(t) - 1; \\ \dot{q}_k(t) &= x_k(t),\ \forall k \notin \mathcal{I} \end{aligned} \tag{3.6}$$

We now state the main result regarding the large deviations behavior of LQF scheduling.

Theorem 3.1 *Under statistically independent and identical arrival processes to each queue, the large deviation behavior of buffer overflow under LQF scheduling is given as follows:*

(i) The exponent is given by

$$E_N^{LQF} = \min_{k=1,\ldots,N} k\Theta_k, \tag{3.7}$$

where Θ_k is the system occupancy exponent for k parallel queues, given by (3.4).

(ii) For a given λ, suppose that a unique $j \le N$ minimizes (3.7), i.e.,

$$j = \arg\min_{k=1,\ldots,N} k\Theta_k.$$

Then, for that λ, the likeliest overflow trajectory consists of j queues reaching overflow. More specifically, the likeliest overflow trajectory[4] (in the $(q_1(t),\ldots,q_N(t))$ space) is the line segment joining the origin to the point $(q_1(0)=1,\ldots,q_j(0)=1, q_{j+1}(0)=\frac{\lambda}{a_j^*},\ldots,q_N(0)=\frac{\lambda}{a_j^*})$, where $\frac{\lambda}{a_j^*}<1$.

The proof of the theorem follows a rather elaborate sample path large deviations argument that involves solving a variational problem. We relegate the proof to Chapter Appendix 3.A.1, and discuss the theorem intuitively.

[4]The symmetry allows us to only present the case where the first j queues overflow.

The first part of the theorem states that the buffer overflow exponent under LQF scheduling is only a function of the system occupancy exponent Θ_k of a system with k parallel queues, where $k \leq N$. The second part of the theorem asserts that if E_N^{LQF} equals $j\Theta_j$ for a unique $j \leq N$, then the likeliest overflow scenario consists of j queues overflowing, and the other $N-j$ queues grow approximately to $M\frac{\lambda}{a_j^*}$, which is less than M. In particular, the queues that do not overflow are never the longest, and hence get no service at all. The service is shared equally among the j queues that overflow, and a_j^* denotes the likeliest *rate* at which the j queues overflow in spite of getting all the service. On the other hand, the queues that do not overflow get to keep all their arrivals, which occur at the average rate λ. The exponent for this case is given by $j\Theta_j$, which corresponds to all the queues in a j-queue system overflowing together. This is because the other $N-j$ queues which do not get service, get arrivals at the average rate, and hence do not contribute to the exponent.

3.3.1 Illustrative examples with Bernoulli traffic

In this section, we obtain the LQF exponents explicitly for a system with symmetric Bernoulli inputs to each queue. We deal with $N=2$ and $N=3$, since these cases are easily visualized, and elucidate the nature of the solution particularly well. We begin by making the following elementary observation regarding LQF scheduling and Bernoulli arrivals.

Proposition 3.2 *Under Bernoulli arrivals and LQF scheduling, the system evolves such that the two longest queues never differ by more than two packets.*

Next, we state a well known result regarding the rate function $\Lambda^*(\cdot)$ for a Bernoulli process.

Proposition 3.3 *For a Bernoulli process of rate λ, the rate function is given by*

$$\Lambda^*(x) = D(x||\lambda) := x \log \frac{x}{\lambda} + (1-x) \log \frac{1-x}{1-\lambda},$$

where $D(x||\lambda)$ is the Kullback-Liebler (KL) divergence (or the relative entropy) between x and λ.

The result is a consequence of Sanov's theorem for finite alphabet [25].

Let us now consider a two queue system with Bernoulli arrivals. For this simple system, it turns out that the exponent can be computed from first principles, without resorting to sample path large deviations. First, the system exponent Θ_2 under Bernoulli arrivals can be computed either from Equation (3.4), or directly from the system occupancy Markov chain, yielding

$$\Theta_2 = 2\log\frac{1-\lambda}{\lambda}.$$

The overflow behavior under LQF scheduling is derived from first principles in the following proposition.

Proposition 3.4 *Under LQF scheduling and Bernoulli arrivals, the following statements hold for the case $N = 2$:*

(i) The likeliest overflow trajectory is along the diagonal, $(q_1 = q_2)$

(ii) $E_2^{LQF} = 2\Theta_2 = 4\log\frac{1-\lambda}{\lambda}$.

Proof: Part (i) of the result is a simple consequence of Proposition 3.2. Specifically, suppose that one of the queues (say Q_1) overflows, so that $Q_1 \geq M$. From Proposition 3.2, it follows that $Q_2 \geq M - 2$. Thus, when an overflow occurs in one queue, the other queue is also about to overflow, so that the only possible (and thus the likeliest) overflow trajectory is along the diagonal.

In order to show part (ii), we first argue that $E_2^{LQF} \geq 2\Theta_2$. Indeed, when a buffer overflow occurs, the total system occupancy is at least $2M - 2$. Thus, the buffer overflow probability is upper-bounded by the probability of the total system occupancy being at least $2M - 2$:

$$\mathbb{P}\{Q_1 \geq M\} \leq \mathbb{P}\{Q_1 + Q_2 \geq 2M - 2\}.$$

We thus have,

$$E_2^{LQF} = \lim_{M\to\infty} -\frac{1}{M}\log \mathbb{P}\{Q_1 \geq M\}$$

$$\geq \lim_{M\to\infty} -\frac{1}{M}\log \mathbb{P}\{Q_1 + Q_2 \geq 2M-2\} = 2\Theta_2,$$

where the last equality follows from the definition of Θ_2.

To show a matching upper bound, note that when the system occupancy is $2M$ or greater, at least one of the queues will necessarily overflow. Thus,

$$\mathbb{P}\{Q_1 + Q_2 \geq 2M\} \leq \mathbb{P}\{\max(Q_1, Q_2) \geq M\}.$$

We can then argue as above that $E_2^{LQF} \leq 2\Theta_2$. □

Let us now analyze a system with three queues, fed by symmetric Bernoulli traffic. In this case, although the longest two queues grow together, it is not immediately clear how the third queue behaves during overflow. As before, the system occupancy exponent Θ_3 can be obtained from (3.4) or directly from the Markov chain to yield

$$\Theta_3 = \log \frac{2A}{(B-2C) + \sqrt{(B-2C)^2 + 4AC}}$$

where $A = (1-\lambda)^3$, $B = 3\lambda^2$, and $C = \lambda^3$.

We can now invoke Theorem 3.1, and conclude that the desired overflow exponent is given by $\min(2\Theta_2, 3\Theta_3)$. (Note that Θ_1 is infinite in this case, since a single queue fed by Bernoulli input cannot overflow). Figure 3-2 shows a plot of $2\Theta_2$ and $3\Theta_3$ as functions of the input rate λ on each queue. It is clear from the figure that for small values of λ, the exponent $2\Theta_2$ dominates the overflow behavior. In this regime, the likeliest manner of overflow involves two queues reaching overflow, while the third queue grows to approximately $M\frac{\lambda}{1/2-\lambda}$. For larger values of λ (> 0.07), the exponent is $3\Theta_3$, and all three queues overflow together.

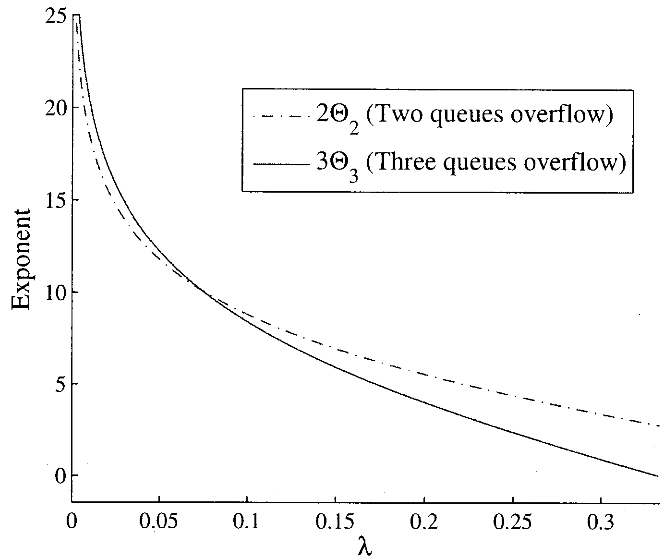


Figure 3-2: Exponent behavior for $N = 3$ under Bernoulli traffic.

3.4 LQF vs. Queue-Blind Policies

In this section, we compare the performance of LQF scheduling with that of queue-blind policies. We only consider a two queue system, since the large deviation behavior of PS and RS is difficult to characterize for $N > 2$. The following result for processor sharing follows from [6].

Proposition 3.5 *The buffer overflow exponent for a two queue system under PS is given by*

$$E_2^{PS} = \inf_{a>0} \frac{1}{a} \left[\Lambda^*(a + \frac{1}{2}) + \Lambda^*(\frac{1}{2}) \right]. \tag{3.8}$$

The likeliest manner of overflow under processor sharing is as follows. Suppose it is the first queue that overflows. The second queue receives traffic at rate $1/2$, which is also its service rate. Thus, the second queue does not overflow, and grows to at most $o(M)$. The first queue receives service at rate $1/2$ and input traffic at rate $a_{ps}^* + 1/2$, where a_{ps}^* optimizes (3.8). Thus, a_{ps}^* is the likeliest rate of overflow of the first queue.

Next, we present the exponent for random scheduling.

Proposition 3.6 *The buffer overflow exponent for a two queue system under RS is given by*

$$E_2^{RS} = \inf_{a>0} \frac{1}{a} \inf_{\phi \in (0,1)} \left[\Lambda^*(a+1-\phi) + \Lambda^*(\phi) + D(\phi || \frac{1}{2}) \right]. \tag{3.9}$$

The proof is outlined in Chapter Appendix 3.A.2. We now describe the most likely overflow event. Suppose queue 1 overflows. The parameter ϕ that appears in the inner infimization in (3.9) denotes the empirical fraction of service received by queue 2. In other words, the 'fair' coin tosses that decide which queue to serve when both queues are nonempty, 'misbehave' statistically. The exponent corresponding to this event is given by $D(\phi||\frac{1}{2})$. If ϕ^* is the optimal value of ϕ in (3.9), the second queue receives traffic at rate ϕ^*, and therefore grows to an $o(M)$ level. The first queue receives traffic at rate $a^*_{rs} + 1 - \phi^*$, where a^*_{rs} is the optimizing value of a in (3.9).

The following result establishes the order among the overflow exponents for the three policies considered in this chapter.

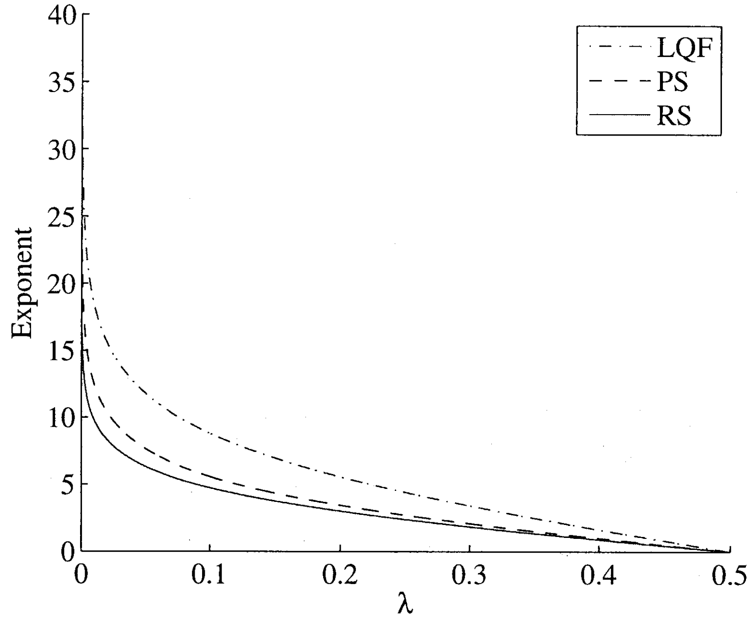
Proposition 3.7 *It holds that $E_2^{RS} \leq E_2^{PS} \leq E_2^{LQF}$.*

Proof: To see the first inequality $E_2^{RS} \leq E_2^{PS}$, note that substituting $\phi = 1/2$ in the RS exponent (3.9) yields the PS exponent. To prove the second inequality, it suffices to show that $E_2^{PS} \leq \Theta_1$ and $E_2^{PS} \leq 2\Theta_2$. First note that for all $a \geq 0$, we have $\Lambda^*(a+1/2) \geq \Lambda^*(1/2)$ since the input rate λ is less than 1/2. Thus, for all $a \geq 0$,

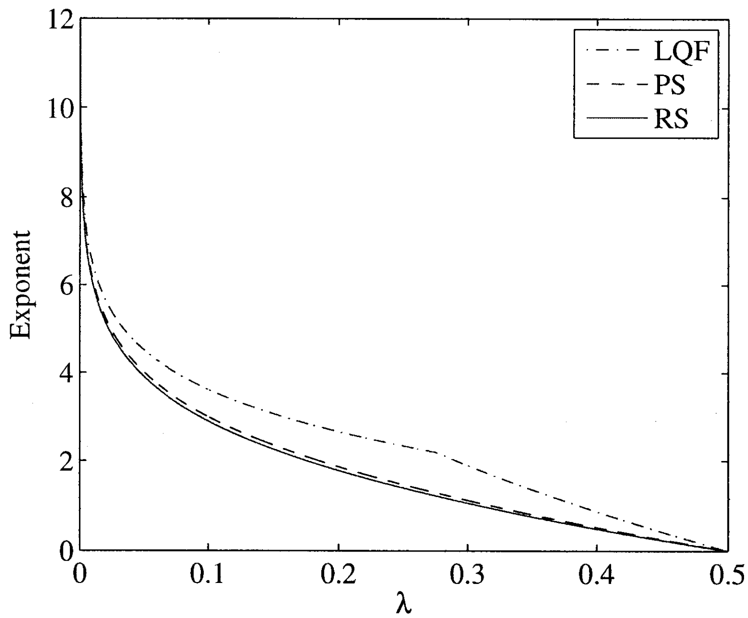
$$\frac{2}{a}\Lambda^*(a+1/2) \geq \frac{1}{a}[\Lambda^*(a+1/2) + \Lambda^*(1/2)].$$

Taking inf on both sides, we have $E_2^{PS} \leq 2\Theta_2$. Similarly, for all $a > 0$, it can be shown that $\Lambda^*(a+1) \geq \Lambda^*(a+1/2) + \Lambda^*(1/2)$, using the fact that $\Lambda^*()$ is an increasing convex function, for arguments greater than λ. Dividing the preceding inequality by a and taking infimum, it follows that $E_2^{PS} \leq \Theta_1$. $\square$

In Figure 3-3, we plot the exponents corresponding to LQF, PS and random scheduling for a two queue system, as a function of the arrival rate λ. Figure 3-3(a) corresponds to having Bernoulli arrivals in each time slot, while in Figure 3-3(b), the



(a)



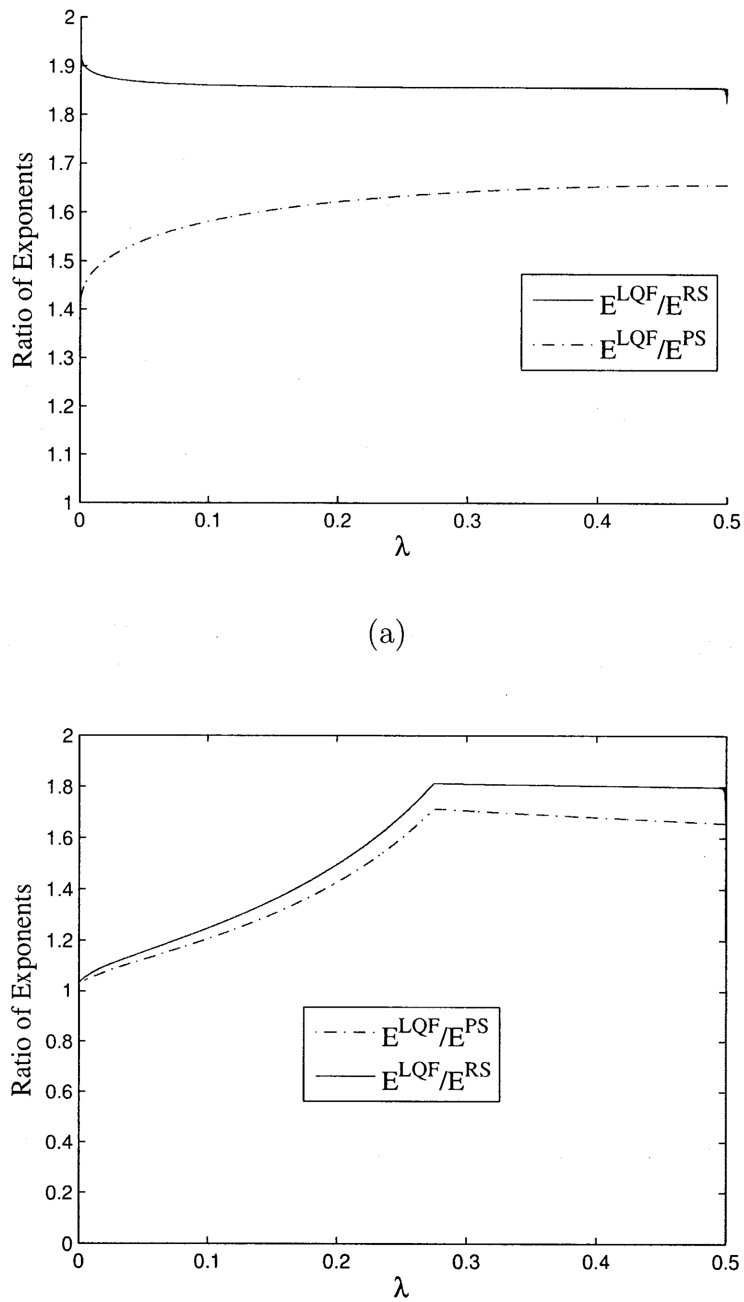
(b)

Figure 3-3: Comparison of LQF, PS and RS exponents for a two queue system, under (a) Bernoulli arrivals (b) Poisson arrivals number of arrivals in each slot is a Poisson random variable. The first observation we make from Figure 3-3 is that, for a given arrival rate, the exponent values for a given policy are generally larger under Bernoulli traffic. This is because Poisson arrivals have a larger potential for being more bursty, and hence the overflow probability is larger (and the exponent smaller) for a given average rate. Next, notice that the LQF exponent under Poisson traffic (Figure 3-3(b)) exhibits a cusp at $\lambda \approx 0.27$. This is because under Poisson traffic, we have two competing exponents Θ_1 and $2\Theta_2$, corresponding respectively to one queue and both queues overflowing. For λ below the cusp, Θ_1 dominates, and vice-versa. On the other hand, under Bernoulli traffic, Θ_1 is infinite. Thus, the LQF exponent is given by $2\Theta_2$, which is a smooth curve as shown in Figure 3-3(a).

3.4.1 Buffer scaling comparison

It is well known that large deviation exponents have direct implications on the buffer size required in order to achieve a certain low probability of overflow. We now compare LQF scheduling with the two queue-blind policies in terms of the buffer scaling required to guarantee a given overflow probability.

In Figure 3-4, we plot the ratio of the LQF exponent to the PS and RS exponents. This ratio is directly related to the savings in the buffer size that results from using LQF scheduling, as opposed to using one of the queue-blind policies. For example, consider the ratio of the LQF exponent to the RS exponent, when the traffic is relatively heavy (say $\lambda > 0.3$). This is the regime where overflows are most likely to occur. We see that under both Bernoulli and Poisson traffic, the LQF exponent is roughly 1.8 times the RS exponent. This implies that in order to achieve a certain overflow probability, the LQF policy requires only 55% of the buffer size required under random scheduling in heavy traffic. A similar comparison can also be made between the LQF and PS exponents.



(a)

(b)

Figure 3-4: Ratio of LQF exponent to PS and RS exponents for (a) Bernoulli arrivals (b) Poisson arrivals.

3.5 Scheduling with Infrequent Queue Length Information

We have seen that the LQF policy has a superior buffer overflow performance compared to queue-blind policies. This is because the queue-blind policies cannot discern and mitigate large build-up on one of the queues, whereas the LQF policy tries to achieve a more balanced set of queues by serving the longest queue in each slot. On the other hand, the scheduler needs to know queue length information in every slot in order to perform LQF scheduling. In this section, we will show that the buffer overflow performance of LQF scheduling can be maintained even if we allow for arbitrarily infrequent queue length information to be conveyed to the scheduler.

The basic idea is that it is sufficient to serve the longest queue only when the queues are large. When the queue lengths are all small, we can save on the queue length information by adopting a non-idling, but queue-blind scheduling strategy. To achieve this, we suggest the following scheduling policy which is a 'hybrid' version of the queue-blind RS, and the LQF policy.

Hybrid Scheduling: Let $K < M$ be a given queue length threshold. In each slot, if all queues are smaller than K, then serve any occupied queue at random. If at least one queue exceeds K, serve the longest queue in that slot.

The following theorem asserts that the hybrid policy achieves the same buffer overflow exponent as LQF scheduling, while requiring queue length information in a vanishingly small fraction of slots.

Theorem 3.2 *For the hybrid scheduling policy proposed above, the following statements hold.*

(i) The fraction of slots in which queue length information is required can be made arbitrarily small.

(ii) The buffer overflow exponent of hybrid scheduling is equal to E_N^{LQF}, as long as $K = o(M)$.

Observe that queue length information is required only in time slots when the longest queue in the system is longer than K. Since RS is a stabilizing policy, the steady state probability that the longest queue exceeds K approaches zero as K becomes large. (In fact, this probability goes to zero *exponentially* in K.) Therefore, the fraction of slots in which queue length information is required can be made arbitrarily small. On the other hand, the overflow exponent remains the same as in the LQF case. This is because the hybrid policy differs from LQF scheduling only in a 'small' neighborhood around the origin. We relegate the proof to Chapter Appendix 3.A.3.

3.6 Conclusions

In this chapter, we studied the buffer overflow probabilities in a system of parallel queues, under some well known scheduling policies. We showed that under max-weight (or LQF) scheduling and symmetric traffic on all the queues, the large deviation exponent of buffer overflow probability is purely a function of the total system occupancy exponent. We also showed that queue length-blind policies such as PS have a smaller overflow exponent (and hence larger buffer size requirements) than max-weight scheduling. Finally, we showed that the superior buffer overflow performance of LQF policy can be preserved even under arbitrarily infrequent queue length information.

3.A Proofs

3.A.1 Proof of Theorem 3.1

The proof can be divided into two parts. The first part involves showing that the queue length process under LQF scheduling satisfies an LDP, whose rate function is given by the solution to a variational problem. The second step involves solving the variational problem in the case of symmetric arrivals, and proving that the optimal solution to the variational problem takes a simple form, as given by the theorem.

The existence of an LDP for the queue length was shown in [60] for longest weighted waiting time as well as longest weighted queue length scheduling. Assuming without loss of generality that the first queue overflows, the exponent is given by the following variational problem

$$\min \int_{-T}^{0} \left[\sum_{i=1}^{N} \Lambda^*(x_i(t)) \right] \mathrm{d}t \tag{3.10}$$

subject to

$$q_i(-T) = 0, \forall i$$

$$q_1(0) = 1,$$

$$T : \text{ free},$$

$$q_j(0) : \text{ free for } j > 1,$$

and the queue length trajectories $q_i(t)$ evolve according to (3.6).

Our emphasis is on solving the above variational problem under the symmetric traffic scenario. In (3.10), the empirical rates $x_i(t)$ are the control variables, and the cost function is the exponent corresponding to the control variables, as given by Mogulskii's theorem. In words, the variational problem is to find the set of empirical rates which leads to the smallest exponent, and results in the overflow of at least one queue. Note that the above is a free time problem, i.e., the time T over which overflow occurs is not constrained. Also, it is possible for queues other than the first queue to reach overflow.

An important property which helps us solve (3.10) is given by the following lemma, which states that when the queue lengths are within one of the regions $\mathcal{R}_{\mathcal{I}}$, the empirical rates $x_i(t)$ can be taken as constants, without loss of optimality.

Lemma 3.2 *Fix a time interval* $[-T_1, -T_2]$ *and consider a control trajectory* $x_i(t)$, $i = 1, \ldots, N$, $t \in [-T_1, -T_2]$, *such that the fluid limit of the queue lengths* $q_i(t)$, $i = 1, \ldots, N$, $t \in [-T_1, -T_2]$ *stay within a particular region* $\mathcal{R}_{\mathcal{I}}$. *Define the average con-*

trol trajectory $\bar{x}_i$ in the interval $[-T_1, -T_2]$ as

$$\bar{x}_i(\tau) = \frac{1}{T_1 - T_2} \int_{-T_1}^{-T_2} x_i(t)\mathrm{d}t$$

for $i = 1, \ldots, N$ and $\tau \in [-T_1, -T_2]$. Then, the queue lengths under the average control trajectory $\bar{x}_i(t)$ lie entirely within $\mathcal{R}_{\mathcal{I}}$, and satisfy the same initial and final conditions at $t = -T_1$ and $t = -T_2$ respectively. Furthermore, the cost achieved under the (constant) control trajectory $\bar{x}_i(t)$ is not larger than the cost achieved under $x_i(t)$.

Proof: The proof is akin to the two dimensional case treated in [6]. That the queue length trajectories under the average control $\bar{x}_i$ satisfy the initial and final conditions is easy to verify. Further, the trajectory moves along a straight line, and therefore stays entirely with $\mathcal{R}_{\mathcal{I}}$, due to the convexity of the region. Finally, due to the convexity of $\Lambda^*(\cdot)$, we have

$$\begin{aligned} \int_{-T_1}^{-T_2} \left[\sum_{i=1}^{N} \Lambda^*(x_i(t)) \right] dt &\geq (T_1 - T_2) \left[\sum_{i=1}^{N} \Lambda^* \left(\frac{1}{T_1 - T_2} \int_{-T_1}^{-T_2} x_i(t)\mathrm{d}t \right) \right] \\ &= (T_1 - T_2) \left[\sum_{i=1}^{N} \Lambda^* \left(\bar{x}_i \right) \right]. \end{aligned}$$

This implies that the average control trajectory is not more costly than the original control trajectory. □

Using Lemma 3.2, we next compute the exponents corresponding to overflow trajectories that stay entirely within a particular region $\mathcal{R}_{\mathcal{I}}$. Later, we will show that overflow trajectories that traverse more than one region cannot have a strictly smaller exponent than trajectories that stay within exactly one of the regions. This will give us the result we want.

Consider an overflow trajectory that lies entirely within $\mathcal{R}_{\mathcal{I}_j}$, where $|\mathcal{I}_j| = j$ for some $1 \leq j < N$. In this case, the j queues in the index set $\mathcal{I}_j$ reach overflow, while the other $N - j$ queues are strictly smaller, and hence receive no service. Due to the symmetry of arrivals, we can compute the exponent assuming that $\mathcal{I}_j = \{1, \ldots, j\}$, i.e., the first j queues overflow. Lemma 3.2 implies that the optimal empirical rates can be restricted to constant values[5] $x_i,\ i = 1, \ldots, N$ for this particular overflow event. Let $a = 1/T$ denote the rate at which the first j queues overflow. Since each queue $k \in \{1, \ldots, j\}$ overflows at rate a, the empirical input rate x_k must be of the form $x_k = a + \phi_k$, where $\phi_k \geq 0$ can be thought of as the rate at which queue k receives service in the overflow interval. Since the first j queues receive all the service, we have $\sum_{k=1}^{j} \phi_k = 1$. Next, for $l > j$, we need $x_l \leq a$, since these queues are never the longest, and hence get no service.

The optimization in (3.10) takes the following form when the first j queues reach overflow.

$$\inf_{a>0} \frac{1}{a} \inf_{\substack{\phi_k \geq 0; \sum_{k=1}^{j} \phi_k = 1 \\ x_l \leq a,\ \forall\ l > j}} \sum_{k=1}^{j} \Lambda^*(a + \phi_k) + \sum_{l=j+1}^{N} \Lambda^*(x_k) \tag{3.11}$$

Let us now perform the inner minimization in (3.11). It is obvious that the minimization over $\phi_k,\ k \leq j$ and $x_l,\ l > j$ can be performed independently. Due to convexity of the rate function, we have

$$\frac{1}{j} \sum_{k=1}^{j} \Lambda^*(a + \phi_k) \geq \Lambda^*(\frac{1}{j} \sum_{k=1}^{j} (a + \phi_k)) = \Lambda^*(a + \frac{1}{j}).$$

Therefore, the optimal value of the ϕ_ks is given by $\phi_k = 1/j,\ k \leq j$. Next, consider optimizing over x_l for $l > j$. We distinguish two cases:

(i) $a > \lambda$: In this case, it is optimal to choose $x_l = \lambda$ for each $l > j$, since $\Lambda^*(\lambda) = 0$.

(ii) $a \leq \lambda$: In this case, the constraint $x_l \leq a$ has to be active, since for $x < \lambda$, $\Lambda^*(x)$ is decreasing in x. Thus, we have $x_l = a$.

Putting the two cases together, we get from (3.11) the exponent E_j corresponding to exactly j queues overflowing, while the trajectory stays inside $\mathcal{R}_{\mathcal{I}_j}$.

$$E_j = \min(\chi_j, \xi_j) \tag{3.12}$$

[5]For simplicity of notation, we henceforth use x_i in place of $\bar{x}_i$.

with

$$\chi_j = \inf_{0<a\le\lambda} \frac{1}{a}\left[j\Lambda^*(a+\frac{1}{j}) + (N-j)\Lambda^*(a)\right], \text{ and}$$

$$\xi_j = \inf_{a>\lambda} \frac{j}{a}\Lambda^*(a+\frac{1}{j}). \tag{3.13}$$

The above expression holds for $1 \le j < N$. The exponent for all the N queues overflowing is simpler to obtain; it is given by

$$E_N = \inf_{a>0} \frac{N}{a}\Lambda^*(a+\frac{1}{N}) = N\Theta_N, \tag{3.14}$$

where the last equality follows by recalling (3.4). The optimal exponent considering the set of all overflow trajectories that stay inside *any one* of the regions $\mathcal{R}_{\mathcal{I}}$, $\mathcal{I} \subset \{1,\ldots,N\}$ is obtained by minimizing E_j over $j = 1,\ldots,N$.

At this point, we are two steps away from obtaining the result. The first step involves showing that there is nothing further to be gained by considering paths that traverse more than one of the partitioning regions. This would imply that the optimal exponent is given by $\min_{1\le j\le N} E_j$. The second step involves showing that $\min_{1\le j\le N} E_j = \min_{1\le j\le N} j\Theta_j$, where Θ_j is the system occupancy exponent of j parallel queues, defined in (3.4). The following two lemmata establish what is needed.

Lemma 3.3 *For every queue overflow trajectory that traverses more than one of the regions $\mathcal{R}_{\mathcal{I}}$, $\mathcal{I} \subset \{1,\ldots,N\}$, there exists an overflow trajectory that lies entirely within one of the regions, while achieving an exponent that is no larger.*

Proof: We only rule out overflow trajectories that traverse two regions; similar arguments can be used for trajectories that visit more than two regions. Consider a queue trajectory that starts out in a region $\mathcal{R}_{\mathcal{I}}$ but reaches overflow in region $\mathcal{R}_{\mathcal{J}}$, while staying in one of the two regions at every instant in between. Note that the region $\mathcal{R}_{\mathcal{I}}$ is a convex set of dimension $N - |\mathcal{I}| + 1$. That is, regions that involve a larger number of queues growing together, have a smaller dimension and vice-versa.

We will consider two cases, $\mathcal{I} \supset \mathcal{J}$ and $\mathcal{I} \subset \mathcal{J}$. Brief reflection should make it clear that if one of the above two containments is not satisfied, the trajectory has to necessarily traverse more than two regions. The arguments that follow are easier to understand if visualized in two dimensions.

Suppose $\mathcal{I} \subset \mathcal{J}$. Consider a trajectory that starts out at the origin at $t = -T$, and stays inside $\mathcal{R}_{\mathcal{I}}$ until time $t = -T_1$, when it enters $\mathcal{R}_{\mathcal{J}}$. The trajectory stays in $\mathcal{R}_{\mathcal{J}}$ until overflow at $t = 0$. Intuitively, the queues $q_i,\ i \in \mathcal{I}$ start out growing together. At time $-T_1$, the queues $q_i,\ i \in \mathcal{J} - \mathcal{I}$ 'catch up', and overflow occurs in all the queues in the index set $\mathcal{J}$. Since constant empirical input rates are optimal inside each partition region (Lemma 3.2), the arbitrary trajectory in $\mathcal{R}_{\mathcal{I}}$ can be replaced at no further cost by a straight segment that has the same initial and final values ($q_i(-T) = 0$, and $q_i(-T_1) \in \mathcal{R}_{\mathcal{J}}$ for each i). This segment lies entirely in $\mathcal{R}_{\mathcal{I}}$, but is arbitrarily close to the region $\mathcal{R}_{\mathcal{J}}$. (Note that $\mathcal{R}_{\mathcal{J}}$ forms one of the 'boundaries' of $\mathcal{R}_{\mathcal{I}}$). However, the cost of this replaced segment is clearly not lower than the optimal trajectory in $\mathcal{R}_{\mathcal{J}}$ with the same initial and final conditions. The part of the trajectory from $t = -T_1$ until over flow at $t = 0$, can again be replaced by the optimal trajectory in $\mathcal{R}_{\mathcal{J}}$ with the corresponding end points. Thus, overall, the cost of the original trajectory is greater than or equal to that of the optimal trajectory in $\mathcal{R}_{\mathcal{J}}$.

Now consider the case $\mathcal{I} \supset \mathcal{J}$. Intuitively, this case corresponds to the queues $q_i,\ i \in \mathcal{I}$ starting to grow together. At some time instant, the queues $q_i,\ i \in \mathcal{I} - \mathcal{J}$ start 'losing out', and overflow occurs within $\mathcal{R}_{\mathcal{J}}$. The arbitrary trajectories in each of the regions can be replaced with an optimal segment in each of the regions, with the same boundary conditions at no added cost. The cost of this replaced trajectory, is a convex combination of the optimal overflow trajectories in regions $\mathcal{R}_{\mathcal{J}}$ and $\mathcal{R}_{\mathcal{I}}$, and hence cannot be smaller than the smaller of the two costs. Thus, a strictly smaller cost cannot be obtained by a trajectory that traverses two regions. $\square$

Lemma 3.4 $\min_{1 \le j \le N} E_j = \min_{1 \le j \le N} j\Theta_j$.

Proof: We first prove that $\chi_j \ge E_N$ for all $j < N$. First, using convexity, we can write

$$\begin{aligned} \frac{j}{N}\Lambda^*(a + \frac{1}{j}) + \frac{N-j}{N}\Lambda^*(a) &\geq \Lambda^*\left(\frac{j}{N}(a + \frac{1}{j}) + \frac{N-j}{N}a\right) \\ &= \Lambda^*(a + \frac{1}{N}). \end{aligned} \tag{3.15}$$

We now have

$$
\begin{aligned}
\chi_j &= \inf_{0<a\le\lambda} \frac{1}{a}\left[j\Lambda^*(a+\frac{1}{j}) + (N-j)\Lambda^*(a) \right] \\
&\ge \inf_{a>0} \frac{1}{a}\left[j\Lambda^*(a+\frac{1}{j}) + (N-j)\Lambda^*(a) \right] \\
&\overset{(a)}{\ge} \inf_{a>0} \frac{N}{a}\Lambda^*(a+\frac{1}{N}) = E_N.
\end{aligned}
$$

The inequality (a) follows from (3.15). It is now clear that the χ_js are irrelevant, as they are always dominated by $E_N = N\Theta_N$. We next write the following series of equalities that imply the lemma.

$$
\begin{aligned}
\min_{1\le j\le N} E_j &= \min(\xi_1,\ldots,\xi_{N-1}, N\Theta_N) \\
&= \min_{1\le j<N} \min(\xi_j, N\Theta_N) \\
&\overset{(b)}{=} \min_{1\le j<N} \min(j\Theta_j, N\Theta_N) \\
&= \min_{1\le j\le N} j\Theta_j
\end{aligned}
$$

In the above, equality (b) is shown as follows. Consider $\min(\xi_j, N\Theta_N)$. The definition of ξ_j (3.13) involves the infimum of a convex function of a over $a > \lambda$. If the convex function attains its *global* minimum for $0 < a < \lambda$, then the infimum in (3.13) will be obtained at $a = \lambda$. In this case, it is easy to show that $N\Theta_N \le \xi_j$. Thus, if ξ_j has to be smaller than $N\Theta_N$, the infimum in (3.13) must be obtained at the global minimum, which lies at $a > \lambda$.[6] Thus, whenever $\min(\xi_j, N\Theta_N) = \xi_j$, we necessarily have

$$
\xi_j = \inf_{a>\lambda} \frac{j}{a}\Lambda^*(a+\frac{1}{j}) = \inf_{a>0} \frac{j}{a}\Lambda^*(a+\frac{1}{j}) = j\Theta_j,
$$

so that equality (b) follows, and we are done. □

[6]It follows that $\lambda/a_j^* < 1$, which proves the claim made in part (ii) of Theorem 3.1.

3.A.2 Proof outline of Proposition 3.6

Let $B_i[t] \in \{0, 1\}$ denote the i.i.d fair 'coin tosses' that decide which queue to serve when both queues are occupied. If $B_i[t] = 1$, then the second queue is served if occupied in slot t; if $B_i[t] = 0$, the first queue is served if occupied. If one of the queues is not occupied in slot t, the occupied queue is served, and $B_i[t]$ becomes irrelevant. Let $\phi(t)$ be the empirical fraction of coin tosses in favor of the second queue, defined analogously to the empirical input rates in Section 3.3. The dynamics of the fluid queue length processes under RS is given by

$$\begin{aligned} \dot{q}_1(t) &= x_1(t) - (1 - \phi(t)) \\ \dot{q}_2(t) &= x_2(t) - \phi(t), \end{aligned}$$

whenever $q_1(t)$ and $q_2(t)$ are non-zero. If either $q_1(t) = 0$ or $q_2(t) = 0$, then

$$\dot{q}_1(t) + \dot{q}_2(t) = x_1(t) + x_2(t) - 1.$$

Here, $x_1(t)$ and $x_2(t)$ are the empirical rates of the input processes.

Using a result analogous to Lemma 3.2, we can prove that constant empirical rates for the inputs as well as the coin tosses is optimal, within each of the regions (i) $q_1(t) > 0,\ q_2(t) > 0$ (ii) $q_1(t) > 0,\ q_2(t) = 0$, and (iii) $q_1(t) = 0,\ q_2(t) > 0$. The problem can now be mapped to an instance of generalized processor sharing with variable service rate, as treated in [6]. The result follows by applying the GPS exponent results to our symmetric case, and noting that the rate function corresponding to the fair coin tosses is given by $D(\cdot||1/2)$.

3.A.3 Proof of Theorem 3.2

Let us first prove part (i), which is quite straightforward. Given $\delta > 0$, suppose that we wish to make the fraction of slots in which queue length information is required less than δ. Since the hybrid policy is non-idling for every K, the steady state probability of the largest queue exceeding K approaches zero as K becomes large. In other words, we can choose a K_δ such that for any $K > K_\delta$, the probability of the longest queue exceeding K is less than δ. It is now clear that a hybrid policy with $K > K_\delta$ will achieve what we want, since the hybrid policy requires queue length information only in slots when the longest queue exceeds K.

We now proceed to show part (ii) of the theorem. For any fixed parameter K of the hybrid policy, we will show that the the overflow exponent remains the same as that of the LQF policy. We first prove an elementary lemma, which asserts that given two systems with different initial queue occupancies, the LQF policy does not allow the queue evolution trajectories to 'diverge', when the two systems are fed by the same input process.

Definition 3.1 *For any two vectors $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{Z}^N$, define $d(\boldsymbol{x}, \boldsymbol{y}) = \max_{i=1,\ldots,N} |x_i - y_i|$.*

Lemma 3.5 *Consider two fictitious systems U and V, in which the initial queue lengths (at time zero) are given by $Q_i^{(U)}[0],\ i = 1, \ldots, N$ and $Q_i^{(V)}[0],\ i = 1, \ldots, N$ respectively. Let $\Delta = d(\boldsymbol{Q}^{(U)}[0], \boldsymbol{Q}^{(V)}[0])$. Suppose that*

(a) The same input sample path $\boldsymbol{A}[t],\ t = 1, \ldots, T_0$ feeds both systems for T_0 time slots, and that

(b) LQF scheduling is performed (with the same tie breaking rule) on both systems for $t = 1, \ldots, T_0$.

Then, for any input sample path and T_0, we have $d(\boldsymbol{Q}^{(U)}[T_0], \boldsymbol{Q}^{(V)}[T_0]) \leq \Delta$.

Proof: The queue lengths in the two systems after the arrivals during the first time slot are given by $B_i^{(U)}[1] = Q_i^{(U)}[0] + A_i[1], i = 1, \ldots, N$, and $B_i^{(V)}[1] = Q_i^{(V)}[0] + A_i[1], i = 1, \ldots, N$. At the end of the first time slot, LQF scheduling is performed based on these queue lengths. We will consider the following exhaustive possibilities, and show that $d(\mathbf{Q}^{(U)}[1], \mathbf{Q}^{(V)}[1]) \leq \Delta$.

(i) If the LQF policy chooses the same queue to serve in both systems, it is clear that $d(\mathbf{Q}^{(U)}[T_0], \mathbf{Q}^{(V)}[T_0]) = \Delta$.

(ii) Suppose the LQF policy chooses queue u in system U and queue v in system V, with $u \neq v$. This implies that $B_u^{(U)}[1] \geq B_v^{(U)}[1]$ and $B_u^{(V)}[1] \leq B_v^{(V)}[1]$. The following sub-cases arise

(iia) $B_u^{(U)}[1] > B_u^{(V)}[1]$ and $B_v^{(V)}[1] > B_v^{(U)}[1]$. In this case, after the LQF policy finishes service, $|Q_i^{(U)}[1] - Q_i(V)[1]| = |Q_i^{(U)}[0] - Q_i(V)[0]| - 1$ for $i = u, v$, and $|Q_i^{(U)}[1] - Q_i(V)[1]| = |Q_i^{(U)}[0] - Q_i(V)[0]|$ $i \neq u, v$. Thus, in this case, $d(\mathbf{Q}^{(U)}[1], \mathbf{Q}^{(V)}[1]) \leq \Delta$.

(iib) One of the inequalities in (iia) fail to hold. Suppose $B_u^{(U)}[1] \leq B_u^{(V)}[1]$. This implies that $B_v^{(U)}[1] \leq_a B_u^{(U)}[1] \leq_b B_u^{(V)}[1] \leq_c B_v^{(V)}[1]$. If inequalities a and c are both met with equalities, there is a tie in both systems between queues u and v. In such a case, the (common) tie breaking rule would select the same queue for both systems[7], and this case is covered under (i). We can therefore assume that at least one of the inequalities a and c is strict. Now, it is clear that $|Q_v^{(U)}[1] - Q_v(V)[1]| = |Q_v^{(U)}[0] - Q_v(V)[0]| - 1$. However, the difference *increases* in the u co-ordinate: $|Q_u^{(U)}[1] - Q_u(V)[1]| = |Q_u^{(U)}[0] - Q_u(V)[0]| + 1$. This increase does not contribute to a strict increase in $d(\mathbf{Q}^{(U)}[1], \mathbf{Q}^{(V)}[1])$, since $|Q_u^{(U)}[1] - Q_u(V)[1]| = |Q_u^{(U)}[0] - Q_u(V)[0]| + 1 \leq_d |Q_v^{(U)}[0] - Q_v(V)[0]| \leq \Delta$. Inequality d holds since one of the inequalities a and c is strict. Thus, $d(\mathbf{Q}^{(U)}[1], \mathbf{Q}^{(V)}[1]) \leq \Delta$ in this case too.

Iterating over time slots, it can be shown that $d(\mathbf{Q}^{(U)}[t], \mathbf{Q}^{(V)}[t]) \leq \Delta$, for all $t \geq 1$. $\square$

Let us now show that the overflow exponent under hybrid scheduling is greater than or equal to E_N^{LQF}. We prove this by showing that for every input sample path that leads to a buffer overflow under hybrid scheduling, the LQF policy also gets close to overflow. Specifically, let $\mathbf{A}[t]$, $t = 1, \ldots, T$ be an input sample path which leads to a buffer overflow at time T under hybrid scheduling. Thus, $Q_i^{(H)}[T] \geq M$, for some $i \leq N$. (We use the superscript H to denote queue lengths under hybrid scheduling,

[7] If the tie breaking rule is randomized, we need to assume that the same sample path for the tie breakers apply to both systems.

and L for LQF scheduling). Let $\tau \leq T$ denote the last time that all the queues were less than or equal to K. Thus, $Q_j^{(H)}[\tau] \leq K, j = 1, \ldots, N$. Now, since both hybrid scheduling and LQF scheduling are non-idling, the total number of packets in the system is conserved. Thus, if the same input sample path were to feed a system with LQF scheduling in each slot, we would have $\sum_j Q_j^{(L)}[\tau] = \sum_j Q_j^{(H)}[\tau] < NK$, from which it is immediate that

$$d(\mathbf{Q}^{(H)}[\tau], \mathbf{Q}^{(L)}[\tau]) < NK. \tag{3.16}$$

Observe that by the definition of τ, the hybrid policy actually performs LQF scheduling during the time slots $\tau + 1, \ldots, T$. Thus, we have two systems which start with different initial queue lengths $\mathbf{Q}^{(H)}[\tau]$ and $\mathbf{Q}^{(L)}[\tau]$. However, they are both fed by the same input sample path, and are served according to the LQF policy for $t > \tau$. Lemma 3.5 now applies, and we can conclude that $d(\mathbf{Q}^{(H)}[T], \mathbf{Q}^{(L)}[T]) \leq d(\mathbf{Q}^{(H)}[\tau], \mathbf{Q}^{(L)}[\tau])$. When combined with (3.16), this yields $d(\mathbf{Q}^{(H)}[T], \mathbf{Q}^{(L)}[T]) < NK$. Thus, $Q_i^{(L)}[T] \geq M - NK$, whenever $Q_i^{(H)}[T] \geq M$. This shows that for every input sample path that leads to an overflow under Hybrid scheduling, the LQF policy is also close to overflow.

Since this is true for every overflow sample path, we have the steady state relation

$$\mathbb{P}\left\{\max_i Q_i^{(H)} \geq M\right\} \leq \mathbb{P}\left\{\max_i Q_i^{(L)} \geq M - NK\right\},$$

from which it follows that $E_N^H \geq E_N^{LQF}$. Next, in order to show that $E_N^H \leq E_N^{LQF}$, we can argue as above that every input sample path that leads to overflow under LQF scheduling, also leads 'close' to an overflow under hybrid scheduling. We have shown that for a fixed K, the hybrid scheduling policy has overflow exponent equal to E_N^{LQF}. It is not difficult to see that if K increases sub-linearly in M, i.e., $K = o(M)$, the exponent would still remain the same. This implies that by scaling K sub-linearly in the buffer size M, the rate of queue length information can be sent to zero, while still achieving the exponent corresponding to LQF scheduling. $\square$

Chapter 4

Asymptotic Analysis of Generalized Max-Weight Scheduling in the presence of Heavy-Tailed Traffic

4.1 Introduction

Traditionally, traffic in telecommunication networks has been modeled using Poisson and Markov-modulated processes. These simple traffic models exhibit 'local randomness', in the sense that much of the variability occurs in short time scales, and only an average behavior is perceived at longer time scales. With the spectacular growth of packet-switched networks such as the internet during the last couple of decades, these traditional traffic models have been shown to be inadequate. This is because the traffic in packetized data networks is intrinsically more 'bursty', and exhibits correlations over longer time scales than can be modeled by any Markovian point process. Empirical evidence, such as the famous Bellcore study on self-similarity and long-range dependence in ethernet traffic [35] lead to increased interest in traffic models with high variability.

Heavy-tailed distributions, which have long been used to model high variability and risk in finance and insurance, were considered as viable candidates to model traffic in data networks. Further, theoretical work such as [30], linking heavy-tails to long-range dependence (LRD) lent weight to the belief that extreme variability in the internet file sizes is ultimately responsible for the LRD traffic patterns reported in [35] and elsewhere.

Many of the early queueing theoretic results for heavy-tailed traffic were obtained for the single server queue; see [48] for a detailed survey of these results. It turns out that the service discipline plays an important role in the latency experienced in a queue, when the traffic is heavy-tailed. For example, it was shown in [3] that any non-preemptive service discipline leads to infinite expected delay, when the traffic is sufficiently heavy-tailed. Further, the asymptotic behavior of latency under various service disciplines such as first-come-first-served (FCFS) and processor sharing (PS), is markedly different under light-tailed and heavy-tailed scenarios. This is important in the context of scheduling jobs in server farms, and has been widely studied [10, 29, 72].

In the context of communication networks, a subset of the traffic flows may be well modeled as heavy-tailed, and the rest better modeled as light-tailed. In such a scenario, there are relatively few studies on the problem of scheduling *between* the different flows, and the ensuing nature of interaction between the heavy-tailed and light-tailed traffic. Perhaps the earliest, and one of the most important studies in this category is [11], where the interaction between light and heavy-tailed traffic flows under generalized processor sharing (GPS) is studied. In that paper, the authors derive the asymptotic workload behavior of the light-tailed flow, when its GPS weight is greater than its traffic intensity.

One of the key considerations in the design of a scheduling policy for a queueing network is *throughput optimality*, which is the ability to support the largest set of traffic rates that is supportable by a given queueing network. Queue length based scheduling policies, such as max-weight scheduling [64] and its many variants, are known to be throughput optimal in a general queueing network. For this reason, the max-weight family of scheduling policies has received much attention in various networking contexts, including switches [41], satellites [45], wireless [46,63], and optical networks [12].

In spite of a large and varied body of literature related to max-weight scheduling, it is somewhat surprising that the policy has not been adequately studied in the context of heavy-tailed traffic. Specifically, a question arises as to what behavior we can expect due to the interaction of heavy and light-tailed flows, when a throughput optimal max-weight-like scheduling policy is employed. Our present work is aimed at addressing this basic question.

In a recent paper [40], a special case of the problem considered here is studied. Specifically, it was shown that when the heavy-tailed traffic has an infinite variance, the light-tailed traffic experiences an infinite expected delay under max-weight scheduling. Further, it was shown that the max-weight policy can be tweaked to favor the light-tailed traffic, so as to make the expected delay of the light-tailed traffic finite. In the present chapter, we considerably generalize these results by providing a precise asymptotic characterization of the occupancy distributions under the max-weight scheduling family, for a large class of heavy-tailed traffic distributions.

We study a system consisting of two parallel queues, served by a single server. One of the queues is fed by a heavy-tailed arrival process, while the other is fed by light-tailed traffic. We refer to these queues as the 'heavy' and 'light' queues, respectively. In this setting, we analyze the asymptotic performance of max-weight-α scheduling, which is a generalized version of max-weight scheduling. Specifically, while max-weight scheduling makes scheduling decisions by comparing the queue lengths in the system, the max-weight-α policy uses different powers of the queue lengths to make scheduling decisions. Under this policy, *we derive an exact asymptotic characterization of the light queue occupancy distribution*, and *specify all the bounded moments of the queue lengths.*

A surprising outcome of our asymptotic characterization is that the 'plain' max-weight scheduling policy induces the worst possible asymptotic behavior on the light queue tail. We also show that by a choice of parameters in the max-weight-α policy that increases the preference afforded to the light queue, the tail behavior of the light queue can be improved. Ultimately however, the tail of the light queue distribution is lower bounded by a power-law-like curve, for *any* scheduling parameters used in the max-weight-α scheduling policy. Intuitively, the reason max-weight-α scheduling induces a power-law-like decay on the light queue distribution is that the light queue has to compete with a typically large heavy queue for service.

The simplest way to guarantee a good asymptotic behavior for the light queue distribution is to give the light queue complete priority over the heavy queue, so that it does not have to compete with the heavy queue for service. We show that under priority for the light queue, the tail distributions of *both* queues are asymptotically as good as they can possibly be under any policy. Be that as it may, giving priority to the light queue has an important shortcoming – it is not throughput optimal for a general constrained queueing system.

We therefore find ourselves in a situation where on the one hand, the throughput optimal max-weight-α scheduling leads to poor asymptotic performance for the light queue. On the other hand, giving priority to the light queue leads to good asymptotic behavior for both queues, but is not throughput optimal in general. To remedy this situation, we propose a throughput optimal log-max-weight (LMW) scheduling policy, which gives significantly more importance to the light queue compared to max-weight-α scheduling. We analyze the asymptotic behavior of the LMW policy and show that the *light queue occupancy distribution decays exponentially.* We also obtain the exact large deviation exponent of the light queue tail under a regularity assumption on the heavy-tailed input. Thus, the LMW policy has both desirable attributes – it is throughput optimal, and ensures an exponentially decaying tail for the light queue distribution.

The remainder of this chapter is organized as follows. In Section 4.2, we describe the system model. In Section 4.3, we present the relevant definitions and mathematical preliminaries. Section 4.4 deals with the queue length behavior under priority scheduling. Sections 4.5 and 4.7 respectively contain our asymptotic results for max-weight α scheduling, and the LMW policy. We conclude the chapter in Section 4.8.

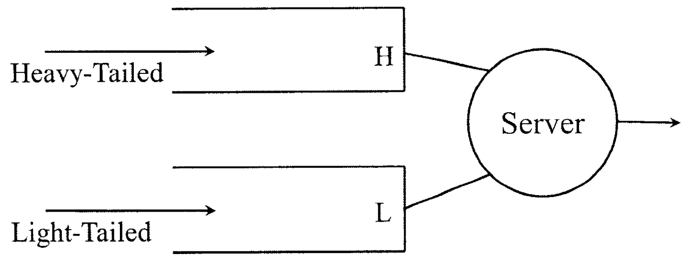


Figure 4-1: A system of two parallel queues, with one of them fed with heavy-tailed traffic.

4.2 System Model

Our system consists of two parallel queues, H and L, served by a single server, as depicted in Figure 4-1. Time is slotted, and stochastic arrivals of packet bursts occur to each queue in each slot. The server is capable of serving one packet per time slot from only one of the queues according to a scheduling policy. Let $H(t)$ and $L(t)$ denote the number of packets that arrive during slot t to H and L respectively. Although we postpone the precise assumptions on the traffic to Section 4.3.2, let us loosely say that the input $L(t)$ is light-tailed, and $H(t)$ is heavy-tailed. We will refer to the queues H and L as the heavy and light queues, respectively. The queues are assumed to be always connected to the server. Let $q_H(t)$ and $q_L(t)$, respectively, denote the number of packets in H and L during slot t, and let q_H and q_L denote the steady-state queue lengths, when they exist. Our aim is to characterize the behavior of $\mathbb{P}\{q_L > b\}$ and $\mathbb{P}\{q_H > b\}$ as b becomes large, under various scheduling policies.

4.3 Definitions and Mathematical Preliminaries

4.3.1 Heavy-tailed distributions

We begin by defining some properties of tail distributions of non-negative random variables.

Definition 4.1 *A random variable X is said to be* light-tailed *if there exists $\theta > 0$ for which $\mathbb{E}[\exp(\theta X)] < \infty$. A random variable is* heavy-tailed *if it is not light-tailed.*

In other words, a light-tailed random variable is one that has a well defined moment generating function in a neighborhood of the origin. The complementary distribution function of a light-tailed random variable decays at least exponentially fast. Heavy-tailed random variables are those which have complementary distribution functions that decay slower than any exponential. This class is often too general to study, so sub-classes of heavy-tailed distributions, such as sub-exponentials have been defined and studied in the past [59]. We now review some definitions and existing results on some relevant classes of heavy-tailed distributions. In the remainder of this section, X will denote a non-negative random variable, with complementary distribution function $\overline{F}(x) = \mathbb{P}\{X > x\}$. For the most part, we adhere to the terminology in [9, 15].

Notation: If $f(b)$ and $g(b)$ are positive functions, we write $f(b) \sim g(b)$ to mean

$$\lim_{b\to\infty} \frac{f(b)}{g(b)} = 1.$$

Similarly, $f(b) \gtrsim g(b)$ means

$$\liminf_{b\to\infty} \frac{f(b)}{g(b)} \geq 1. \tag{4.1}$$

Definition 4.2 *1. $\overline{F}(x)$ is said to have a* regularly varying *tail of index ν, notation $\overline{F} \in \mathcal{R}(\nu)$, if*

$$\lim_{x\to\infty} \frac{\overline{F}(kx)}{\overline{F}(x)} = k^{-\nu}, \ \forall\ k > 0.$$

2. $\overline{F}(x)$ is said to be extended-regularly varying*, notation $\overline{F} \in \mathcal{ER}$, if for some real $c, d > 0$, and $\Gamma > 1$,*

$$k^{-d} \leq \liminf_{x\to\infty} \frac{\overline{F}(kx)}{\overline{F}(x)} \leq \limsup_{x\to\infty} \frac{\overline{F}(kx)}{\overline{F}(x)} \leq k^{-c}, \ \forall k \in [1, \Gamma].$$

3. $\overline{F}(x)$ is said to be intermediate-regularly varying*, notation $\overline{F} \in \mathcal{IR}$, if*

$$\lim_{k\downarrow 1} \liminf_{x\to\infty} \frac{\overline{F}(kx)}{\overline{F}(x)} = \lim_{k\downarrow 1} \limsup_{x\to\infty} \frac{\overline{F}(kx)}{\overline{F}(x)} = 1.$$

4. $\overline{F}(x)$ *is said to be* order-regularly varying, *notation* $\overline{F} \in \mathcal{OR}$, *if for some* $\Gamma > 1$,

$$0 < \liminf_{x\to\infty} \frac{\overline{F}(kx)}{\overline{F}(x)} \leq \limsup_{x\to\infty} \frac{\overline{F}(kx)}{\overline{F}(x)} < \infty, \ \forall k \in [1, \Gamma].$$

It is easy to see from the definitions that $\mathcal{R} \subset \mathcal{ER} \subset \mathcal{IR} \subset \mathcal{OR}$. In fact, the containments are proper, as shown in [15]. Intuitively, $\mathcal{R}$ is the class of distributions with tails that decay according to a power-law with parameter ν. Indeed, it can be shown that

$$\overline{F} \in \mathcal{R}(\nu) \iff \overline{F}(x) = U(x)x^{-\nu},$$

where $U(x)$ is a *slowly varying* function, i.e, a function that satisfies $U(kx) \sim U(x)$, $\forall k > 0$. The other three classes are increasingly more general, but as we shall see, they all correspond to distributions that are asymptotically heavier than some power-law curve. In what follows, a statement such as $X \in \mathcal{IR}$ should be construed to mean $\mathbb{P}\{X > x\} \in \mathcal{IR}$.

Next, we define the lower and upper orders of a distribution.

Definition 4.3 1. *The* lower order *of* $\overline{F}(x)$ *is defined by*

$$\xi(\overline{F}) = \liminf_{x\to\infty} -\frac{\log \overline{F}(x)}{\log x}.$$

2. *The* upper order *of* $\overline{F}(x)$ *is defined by*

$$\rho(\overline{F}) = \limsup_{x\to\infty} -\frac{\log \overline{F}(x)}{\log x}.$$

It can be shown that for regularly varying distributions, the upper and lower orders coincide with the index ν. It also turns out that both orders are finite for the class $\mathcal{OR}$, as asserted below.

Proposition 4.1 $\rho(\overline{F}) < \infty$ *for every* $\overline{F} \in \mathcal{OR}$.

Proof: Follows from Theorem 2.1.7 & Proposition 2.2.5 in [9]. □

The following result, which is a consequence of Proposition 4.1, shows that every $\overline{F} \in \mathcal{OR}$ is asymptotically heavier than a power-law curve.

Proposition 4.2 *Let $\overline{F} \in \mathcal{OR}$. Then, for each $\rho > \rho(\overline{F})$, we have $x^{-\rho} = o(\overline{F}(x))$ as $x \to \infty$.*

Proof: See Equation (2.4) in [51].

Definitions 4.2 and 4.3 deal with asymptotic tail probabilities of a random variable. Next, we introduce the notion of tail coefficient, which is a *moment* property.

Definition 4.4 *The* tail coefficient *of a random variable X is defined by*

$$C_X = \sup\{c \geq 0 \mid \mathbb{E}[X^c] < \infty\}.$$

In other words, the tail coefficient is the threshold where the power moment of a random variable starts to blow up. Note that the tail coefficient of a light-tailed random variable is infinite. On the other hand, the tail coefficient of a heavy-tailed random variable may be infinite (e.g., log-normal) or finite (e.g., Pareto). The next result shows that the tail coefficient and order are, in fact, closely related parameters.

Proposition 4.3 [1] *The tail coefficient of X is equal to the lower order of $\overline{F}(x)$.*

Proof: Suppose first that the lower order is infinite, so that for any $s > 0$, we can find an x large enough such that

$$-\frac{\log \mathbb{P}\{X > x\}}{\log x} > s.$$

Thus, for large enough x, we have

$$\mathbb{P}\{X > x\} < x^{-s},\ \forall\, s > 0.$$

This implies $\mathbb{E}[X^c] < \infty$ for all $c > 0$. Therefore, the tail coefficient of X is also infinite.

Next suppose that $\xi(\overline{F}) \in (0, \infty)$. We will show that (i) $\mathbb{E}[X^c] < \infty$ for all $c < \xi(\overline{F})$, and (ii)$\mathbb{E}[X^c] = \infty$ for all $c > \xi(\overline{F})$. To show (i), we argue as above that

[1]The author is grateful to Jayakrishnan Nair (Caltech) for suggesting a proof of Proposition 4.3 via a personal communication.

for large enough x, we have $\mathbb{P}\{X > x\} < x^{-s}$, when $s < \xi(\overline{F})$. Thus, $\mathbb{E}[X^c] < \infty$ for all $c < \xi(\overline{F})$. To show (ii), let us consider some s such that $c > s > \xi(\overline{F})$. By the definition of $\xi(\overline{F})$ there exists a sequence $\{x_i\}$ that increases to infinity as $i \to \infty$, such that

$$-\frac{\log \mathbb{P}\{X > x_i\}}{\log x_i} \leq s, \ \forall\, i \iff \mathbb{P}\{X > x_i\} \geq x^{-s}, \ \forall\, i.$$

Therefore,

$$\mathbb{E}[X^c] = \int_0^\infty x^c \mathrm{d}F_X(x) \geq \int_{x_i}^\infty x^c \mathrm{d}F_X(x) \geq x_i^c \mathbb{P}\{X > x_i\} \geq x_i^c x_i^{-s}, \ \forall\, i,$$

from which it follows that $\mathbb{E}[X^c] = \infty$. Therefore, the tail coefficient of X is equal to $\xi(\overline{F})$. □

We emphasize that Proposition 4.3 holds for *any* random variable, regardless of its regularity properties. Finally, we show that any distribution in the class $\mathcal{OR}$ necessarily has a finite tail coefficient.

Proposition 4.4 *If $X \in \mathcal{OR}$, then X has a finite tail coefficient.*

Proof: From Proposition 4.1, the upper order is finite: $\rho(\overline{F}) < \infty$. Thus, the lower order $\xi(\overline{F})$ is also finite. By Proposition 4.3, the lower order is equal to the tail coefficient, so that the result is immediate. □

4.3.2 Assumptions on the arrival processes

We are now ready to state the precise assumptions on the arrival processes.

1. The arrival processes $H(t)$ and $L(t)$ are independent of each other, and independent of the current state of the system.

2. $H(t)$ is independent and identically distributed (i.i.d.) from slot-to-slot.

3. $L(t)$ is i.i.d. from slot-to-slot.

4. $L(\cdot)$ is light-tailed with $\mathbb{E}[L(t)] = \lambda_L$.

5. $H(\cdot) \in \mathcal{OR}$ with tail coefficient $C_H > 1$, and $\mathbb{E}[H(t)] = \lambda_H$.

We also assume that $\lambda_L + \lambda_H < 1$, so that the input rate does not overwhelm the service rate. Then, it can be shown that the system is stable[2] under any non-idling policy, and that the steady-state queue lengths q_H and q_L exist.

4.3.3 Residual and age distributions

Here, we define the residual and age distributions for the heavy-tailed input process, which will be useful later. First, we note that $H(\cdot)$ necessarily has a non zero probability mass at zero, since $\lambda_H < 1$. Define H_+ as the strictly positive part of $H(\cdot)$. Specifically,

$$\mathbb{P}\{H_+ = m\} = \frac{\mathbb{P}\{H(\cdot) = m\}}{1 - \mathbb{P}\{H(\cdot) = 0\}}, \quad m = 1, 2, \ldots.$$

Note that H_+ has tail coefficient equal to C_H, and inherits any regularity property of $H(\cdot)$.

Now consider a discrete-time renewal process with inter-renewal times distributed as H_+. Let $H_R \in \{1, 2, \ldots\}$ denote the residual random variable, and $H_A \in \{0, 1, \ldots\}$ the age of the renewal process [24].[3] The joint distribution of the residual and the age can be derived using basic renewal theory:

$$\mathbb{P}\{H_R = k, H_A = l\} = \frac{\mathbb{P}\{H_+ = k + l\}}{\mathbb{E}[H_+]}, \quad k \in \{1, 2 \ldots\}, \ l \in \{0, 1, \ldots\}. \tag{4.2}$$

The marginals of H_R and H_A can be derived from (4.2):

$$\mathbb{P}\{H_R = k\} = \frac{\mathbb{P}\{H_+ \geq k\}}{\mathbb{E}[H_+]}, \quad k \in \{1, 2, \ldots\}. \tag{4.3}$$

[2]The notion of stability used here is the positive recurrence of the system occupancy Markov chain.

[3]We have defined the residual time and age such that if a renewal occurs at a particular time slot, the age at that time slot is zero, and the residual time is equal to the length of the upcoming renewal interval.

$$\mathbb{P}\{H_A = k\} = \frac{\mathbb{P}\{H_+ > k\}}{\mathbb{E}[H_+]}, \quad k \in \{0, 1, \dots\}. \tag{4.4}$$

Next, let us invoke a useful result from the literature.

Lemma 4.1 *If $H(\cdot) \in \mathcal{OR}$, then $H_R \in \mathcal{ER}$, and*

$$\sup_n \frac{n\mathbb{P}\{H_+ > n\}}{\mathbb{P}\{H_R > n\}} < \infty. \tag{4.5}$$

A corresponding result also holds for the age H_A.

Proof: See [15. Lemma 4.2(i)]. □

Using the above, we prove the important result that the residual distribution is *one order heavier* than the original distribution.

Proposition 4.5 *If $H(\cdot) \in \mathcal{OR}$ has tail coefficient equal to C_H, then H_R and H_A have tail coefficient equal to $C_H - 1$.*

Proof: According to (4.5), we have, for all a and some real χ,

$$-\log \mathbb{P}\{H_R > a\} \leq -\log a - \log \mathbb{P}\{H_+ > a\} + \chi.$$

Let us now consider the lower order of H_R :

$$\liminf_{a\to\infty} -\frac{\log \mathbb{P}\{H_R > a\}}{\log a} \leq \liminf_{a\to\infty} \frac{-\log a - \log \mathbb{P}\{H_+ > a\} + \chi}{\log a} = C_H - 1.$$

In the last step above, we have used the tail coefficient of H_+. Since the lower order of H_R equals its tail coefficient (Lemma 4.3), the above relation shows that the tail coefficient of H_R is *at most* $C_H - 1$.

Next, to show the opposite inequality, let us consider the *duration* random variable, defined as

$$H_D = H_R + H_A.$$

Using the joint distribution (4.2), we can obtain the marginal of H_D as

$$\mathbb{P}\{H_D = k\} = \frac{k\mathbb{P}\{H_+ = k\}}{\mathbb{E}[H_+]}, \ k \in \{1, 2, \dots\}.$$

Thus, for any $\epsilon > 0$, the $C_H - 1 - \epsilon$ moment of H_D is finite:

$$\mathbb{E}\left[H_D^{C_H-1-\epsilon}\right] = \sum_{k\geq 1} \frac{k^{C_H-1-\epsilon} k\mathbb{P}\{H_+ = k\}}{\mathbb{E}[H_+]} = \frac{\mathbb{E}\left[H_+^{C_H-\epsilon}\right]}{\mathbb{E}[H_+]} < \infty.$$

Since H_R is stochastically dominated by H_D, it is immediate that $\mathbb{E}\left[H_R^{C_H-1-\epsilon}\right] < \infty$. Therefore, the tail coefficient of H_R is *at least* $C_H - 1$, and the result is proved. □

4.4 The Priority Policies

In this section, we study the two 'extreme' scheduling policies, namely priority for L and priority for H. Our analysis helps us arrive at the important conclusion that the tail of the heavy queue is asymptotically insensitive to the scheduling policy. In other words, there is not much we can do to improve or hurt the tail distribution of H by the choice of a scheduling policy. Further, we show that giving priority to the light queue ensures the best possible asymptotic decay for *both* queue length distributions.

4.4.1 Priority for the Heavy-Tailed Traffic

In this policy, H receives service whenever it is non-empty, and L receives service only when H is empty. It should be intuitively clear at the outset that this policy is bound to have undesirable impact on the light queue. The reason we analyze this policy is that it gives us a best case scenario for the heavy queue.

Our first result shows that the steady-state heavy queue occupancy is one order heavier than its input distribution.

Theorem 4.1 *Under priority scheduling for H, the steady-state queue occupancy distribution of the heavy queue satisfies the following bounds.*

1. *For every* $\epsilon > 0$, *there exists a constant* $\kappa_H(\epsilon)$ *such that*

$$\mathbb{P}\left\{q_H > b\right\} < \kappa_H(\epsilon) b^{-(C_H - 1 - \epsilon)}, \ \forall\ b. \tag{4.6}$$

2.

$$\mathbb{P}\left\{q_H > b\right\} \geq \lambda_H \mathbb{P}\left\{H_R > b\right\}, \ \forall\ b. \tag{4.7}$$

Further, q_H *is a heavy-tailed random variable with tail coefficient equal to* $C_H - 1$. *That is, for each* $\epsilon > 0$, *we have*

$$\mathbb{E}\left[q_H^{C_H - 1 - \epsilon}\right] < \infty, \tag{4.8}$$

and

$$\mathbb{E}\left[q_H^{C_H - 1 + \epsilon}\right] = \infty. \tag{4.9}$$

Proof: Equation (4.8) can be shown using a straightforward Lyapunov argument, along the lines of [40, Proposition 6]. Equation (4.6) follows from (4.8) and the Markov inequality.

Next, to show (4.7), we consider a time instant t at steady-state, and write

$$\mathbb{P}\left\{q_H(t) > b\right\} = \mathbb{P}\left\{q_H(t) > b | q_H(t) > 0\right\} \mathbb{P}\left\{q_H(t) > 0\right\} = \lambda_H \mathbb{P}\left\{q_H(t) > b | q_H(t) > 0\right\}.$$

We have used Little's law at steady-state to write $\mathbb{P}\left\{q_H(t) > 0\right\} = \lambda_H$. Let us now lower bound the term $\mathbb{P}\left\{q_H(t) > b | q_H(t) > 0\right\}$. Conditioned on H being non-empty, denote by $\tilde{B}(t)$ the number of packets that belong to the burst in service that still remain in the queue at time t. Then, clearly, $q_H(t) \geq \tilde{B}(t)$, from which $\mathbb{P}\left\{q_H(t) > b | q_H(t) > 0\right\} \geq \mathbb{P}\left\{\tilde{B}(t) > b\right\}$. Now, since the H queue receives service whenever it is non-empty, it is clear that the time spent at the head-of-line (HoL) by a burst is *equal* to its size. It can therefore be shown that in steady-state, $\tilde{B}(t)$ is distributed according to the residual variable H_R. Thus, $\mathbb{P}\left\{q_H(t) > b | q_H(t) > 0\right\} \geq \mathbb{P}\left\{H_R > b\right\}$, and (4.7) follows. Finally, (4.9) follows from (4.7) and Proposition 4.5.

□

When the distribution of $H(\cdot)$ is regularly varying, the lower bound (4.7) takes on a power-law form that agrees with the upper bound (4.6).

Corollary 4.1 *If $H(\cdot) \in \mathcal{R}(C_H)$, then*

$$\mathbb{P}\{q_H > b\} > U(b)b^{-(C_H-1)}, \ \forall \ b,$$

where $U(\cdot)$ is some slowly varying function.

Since priority for H affords the most favorable treatment to the heavy queue, it follows that the asymptotic behavior of H can be no better than the above under *any* policy.

Proposition 4.6 *Under any scheduling policy, q_H is heavy-tailed with tail coefficient at most $C_H - 1$. That is, Equation (4.9) holds for all scheduling policies.*

Proof: The tail probability $\mathbb{P}\{q_H > b\}$ under any other policy stochastically dominates the tail under priority for H. Therefore, the lower bounds (4.7) and (4.9) would hold for all policies. □

Interestingly, under priority for H, the steady-state light queue occupancy q_L is also heavy-tailed with the *same* tail coefficient as q_H. This should not be surprising, since the light queue has to wait for the entire heavy queue to clear, before it receives any service.

Theorem 4.2 *Under priority for H, q_L is heavy-tailed with tail coefficient $C_H - 1$. Furthermore, the tail distribution $\mathbb{P}\{q_L > b\}$ satisfies the following asymptotic bounds.*

1. *For every $\epsilon > 0$, there exists a constant $\kappa_L(\epsilon)$ such that*

$$\mathbb{P}\{q_L > b\} < \kappa_L(\epsilon)b^{-(C_H-1-\epsilon)}. \tag{4.10}$$

2. *If $H(\cdot) \in \mathcal{OR}$, then*

$$\mathbb{P}\{q_L > b\} \gtrsim \lambda_H \mathbb{P}\left\{H_A > \frac{b}{\lambda_L}\right\} \tag{4.11}$$

Proof: The upper bound (4.10) is a special case of Theorem 4.4 given in the next section. Let us show (4.11). Notice first that the lower bound (4.11) is asymptotic in the sense of (4.1), unlike (4.7) which is exact. As before, let us consider a time t at steady-state, and write using Little's law

$$\mathbb{P}\{q_L(t) > b\} \geq \mathbb{P}\{q_L(t) > b | q_H(t) > 0\} \mathbb{P}\{q_H(t) > 0\} = \lambda_H \mathbb{P}\{q_L(t) > b | q_H(t) > 0\}.$$

Let us denote by $\tilde{A}(t)$ the number of slots that the current head-of-line burst has been in service. Clearly then, L has not received any service in the interval $[t - \tilde{A}(t), t]$, and has kept all the arrivals that occurred during the interval. Thus, conditioned on H being non-empty, $q_L(t) \geq \sum_{\sigma=t-\tilde{A}(t)}^{t} L(\sigma)$. Next, it can be seen that in steady-state, $\tilde{A}(t)$ is distributed as the age variable H_A. Putting it all together, we can write

$$\mathbb{P}\{q_L > b\} \geq \lambda_H \mathbb{P}\{q_L(t) > b | q_H(t) > 0\} \geq \lambda_H \mathbb{P}\left\{\sum_{i=1}^{H_A} L(i) > b\right\}. \tag{4.12}$$

Next, since $H(\cdot) \in \mathcal{OR}$, Lemma 4.1 implies that $H_A \in \mathcal{ER} \subset \mathcal{IR}$. We can therefore invoke Lemma 4.4 in the chapter appendix to write

$$\mathbb{P}\left\{\sum_{i=1}^{H_A} L(i) > b\right\} \sim \mathbb{P}\left\{H_A > \frac{b}{\lambda_L}\right\}. \tag{4.13}$$

Finally, (4.11) follows from (4.12) and (4.13). □

We note that if $H(\cdot)$ is regularly varying, the lower bound (4.11) takes on a power-law form that matches the upper bound (4.10).

4.4.2 Priority for the Light-Tailed Traffic

We now study the policy that serves L whenever it is non-empty, and serves H only if L is empty. This policy affords the best possible treatment to L and the worst possible treatment to H, among all non-idling policies. Under this policy, L is completely oblivious to the presence of H, in the sense that it receives service whenever it has a packet to be served. Therefore, L behaves like a discrete time G/D/1 queue, with light-tailed inputs. Classical large deviation bounds can be derived for such a queue; see [25] for example.

Recall that since $L(\cdot)$ is light-tailed, the log moment generating function

$$\Lambda_L(\theta) = \log \mathbb{E}\left[e^{\theta L(\cdot)}\right]$$

exists for some $\theta > 0$. Define

$$E_L = \sup\{\theta | \Lambda_L(\theta) - \theta < 0\}. \tag{4.14}$$

Proposition 4.7 *Under priority for L, q_L satisfies the large deviation principle (LDP)*

$$\lim_{b \to \infty} -\frac{1}{b} \log \mathbb{P}\{q_L > b\} = E_L. \tag{4.15}$$

In words, the above proposition asserts that the tail distribution of q_L decays exponentially, with rate function E_L. We will refer to E_L as the *intrinsic exponent* of the light queue. An alternate expression for the intrinsic exponent that is often used in the literature is

$$E_L = \inf_{a>0} \frac{1}{a} \Lambda_L^*(1+a), \tag{4.16}$$

where $\Lambda_L^*(\cdot)$ is the Fenchel-Legendre transform (see Equation (3.1)) of $\Lambda_L(\theta)$.

It is clear that the priority policy for L gives the best possible asymptotic behavior for the light queue, and the worst possible treatment for the heavy queue. Surprisingly however, it turns out that the heavy queue tail under priority for L is asymptotically as good as it is under priority for H.

Proposition 4.8 *Under priority for L, q_H is heavy-tailed with tail coefficient $C_H - 1$.*

Proof: It is clear from Proposition 4.6 that the tail coefficient is no more than $C_H - 1$. We need to show that the tail coefficient is no less than $C_H - 1$. This is a special case of Theorem 4.4, given in the next section. □

The above result also implies that the tail coefficient of H cannot be worse than $C_H - 1$ under any other scheduling policy.

Proposition 4.9 *Under any non-idling scheduling policy, q_H has a tail coefficient of at least $C_H - 1$. That is, Equation (4.8) holds for all non-idling scheduling policies.*

Proof: The tail probability $\mathbb{P}\{q_H > b\}$ under any other policy is stochastically dominated by the tail probability under priority for L. □

Propositions 4.6 and 4.9 together imply the insensitivity of the heavy queue's tail distribution to the scheduling policy. We state this important result in the following theorem.

Theorem 4.3 *Under* any *non-idling scheduling policy, q_H is heavy-tailed with tail coefficient equal to $C_H - 1$. Further, $\mathbb{P}\{q_H > b\}$ satisfies bounds of the form (4.6) and (4.7) under all non-idling policies.*

Therefore, it is not possible to either improve or hurt the heavy queue's asymptotic behavior, by the choice of a scheduling policy.

It is evident that the light queue has the best possible asymptotic behavior under priority for L. Additionally, Theorem 4.3 asserts that H is asymptotically the same under any non-idling policy, including priority for L. In other words, this policy ensures the best asymptotic behavior for both queues. Furthermore, it is non-idling, and therefore throughput-optimal in this simple setting.

However, we are ultimately interested in studying more sophisticated network models, where priority for L may not be throughput optimal. In fact, when we study queues with intermittent connectivity in the next chapter, the priority for L policy fails to achieve a significant portion of the rate region of the system. This motivates us to study the asymptotic behavior of general throughput optimal policies belonging to the max-weight family.

4.5 Asymptotic Analysis of Max-Weight-α Scheduling

In this section, we analyze the asymptotic tail behavior of the light queue distribution under max-weight-α scheduling. For fixed parameters $\alpha_H > 0$ and $\alpha_L > 0$, the max-weight-α policy operates as follows: During each time slot t, perform the comparison

$$q_L(t)^{\alpha_L} \gtreqless q_H(t)^{\alpha_H},$$

and serve one packet from the queue that wins the comparison. Ties can be broken arbitrarily, but we break them in favor of the light queue for the sake of definiteness. Note that $\alpha_L = \alpha_H$ corresponds to the usual max-weight policy, which serves the longest queue in each slot. $\alpha_L/\alpha_H > 1$ corresponds to emphasizing the light queue over the heavy queue, and vice-versa.

We derive an asymptotic characterization of the light queue occupancy distribution under max-weight-α scheduling, by deriving matching upper and lower bounds. Our characterization shows that the light queue occupancy is heavy-tailed under max-weight-α scheduling for all values of the parameters α_H and α_L. Since we obtain distributional bounds on the light queue occupancy, our results also shed further light on the moment results derived in [40] for max-weight-α scheduling.

4.5.1 Upper bound

In this section, we derive two different upper bounds on the overflow probability $\mathbb{P}\{q_L > b\}$, that both hold under max-weight-α scheduling. However, depending on the values of α_H and α_L, one of them would be tighter. The first upper bound holds for all non-idling policies, including max weight-α scheduling.

Theorem 4.4 *Under* any *non-idling policy, and for every $\epsilon > 0$, there exists a constant $\kappa_1(\epsilon) > 0$, such that*

$$\mathbb{E}\left[q_L^{C_H-1-\epsilon}\right] < \infty \tag{4.17}$$

and

$$\mathbb{P}\{q_L > b\} < \kappa_1(\epsilon) b^{-(C_H-1-\epsilon)}. \tag{4.18}$$

Proof: Let us combine the two queues into one, and consider the sum input process $H(t) + L(t)$ feeding the composite queue. The server serves one packet from the composite queue in each slot. Under any non-idling policy in the original system, the occupancy of the composite queue is given by $q = q_H + q_L$. Let us first show that the combined input has tail coefficient equal to C_H.

Lemma 4.2 *The tail coefficient of $H(\cdot) + L(\cdot)$ is C_H.*

Proof: Clearly, $\mathbb{E}\left[(H+L)^{C_H+\delta}\right] \geq \mathbb{E}\left[H^{C_H+\delta}\right] = \infty$, for every $\delta > 0$. We next need to show that $\mathbb{E}\left[(H+L)^{C_H-\delta}\right] < \infty$, for every $\delta > 0$. For a random variable X and event E, let us introduce the notation $\mathbb{E}\left[X; E\right] = \mathbb{E}\left[X 1_E\right]$, where 1_E is the indicator of E. (Thus, for example, $\mathbb{E}\left[X\right] = \mathbb{E}\left[X; E\right] + \mathbb{E}\left[X; E^c\right]$.) Now,

$$\begin{aligned}
\mathbb{E}\left[(H+L)^{C_H-\delta}\right] &= \mathbb{E}\left[(H+L)^{C_H-\delta}; H > L\right] + \mathbb{E}\left[(H+L)^{C_H-\delta}; H \leq L\right] \\
&\leq \mathbb{E}\left[(2H)^{C_H-\delta}; H > L\right] + \mathbb{E}\left[(2L)^{C_H-\delta}; H \leq L\right] \\
&< 2^{C_H-\delta}\left\{\mathbb{E}\left[H^{C_H-\delta}\right] + \mathbb{E}\left[L^{C_H-\delta}\right]\right\} < \infty
\end{aligned}$$

where the last inequality follows from the tail coefficient of $H(\cdot)$, and the light-tailed nature of $L(\cdot)$. □

The composite queue is therefore a G/D/1 queue with input tail coefficient C_H. For such a queue, it can be shown that

$$\mathbb{E}\left[q^{C_H-1-\epsilon}\right] < \infty. \tag{4.19}$$

This is, in fact, a direct consequence of Theorem 4.1.

Thus, in terms of the queue lengths in the original system, we have

$$\mathbb{E}\left[(q_H + q_L)^{C_H-1-\epsilon}\right] < \infty,$$

from which it is immediate that $\mathbb{E}\left[q_L^{C_H-1-\epsilon}\right] < \infty$. This proves (4.17). To show (4.18), we use the Markov inequality to write

$$\mathbb{P}\left\{q_L > b\right\} = \mathbb{P}\left\{q_L^{C_H-1-\epsilon} > b^{C_H-1-\epsilon}\right\} < \frac{\mathbb{E}\left[q_L^{C_H-1-\epsilon}\right]}{b^{C_H-1-\epsilon}} < \kappa_1(\epsilon) b^{-(C_H-1-\epsilon)}.$$

□

The above result asserts that the tail coefficient of q_L is at least $C_H - 1$ under any non-idling policy, and that $\mathbb{P}\{q_L > b\}$ is uniformly upper bounded by a power-law curve. Our second upper bound is specific to max-weight-α scheduling. It hinges on a simple observation regarding the scaling of the α parameters, in addition to a theorem in [40]. We first state the following elementary observation due to its usefulness.

Observation: (Scaling of α parameters) Let α_H and α_L be given parameters of a max-weight-α policy, and let $\beta > 0$ be arbitrary. Then, the max-weight-α policy that uses the parameters $\beta\alpha_H$ and $\beta\alpha_L$ for the queues H and L respectively, is *identical* to the original policy. That is, in each time slot, the two policies make the same scheduling decision.

Next, let us invoke an important result from [40].

Theorem 4.5 *If max-weight-α scheduling is performed with $0 < \alpha_H < C_H - 1$, then, for* any *$\alpha_L > 0$, we have $\mathbb{E}\left[q_L^{\alpha_L}\right] < \infty$.*

Thus, by choosing a large enough α_L, any moment of the light queue length can be made finite, as long as $\alpha_H < C_H - 1$. Our second upper bound, which we state next, holds regardless of how the α parameters are chosen.

Theorem 4.6 *Define*

$$\gamma = \frac{\alpha_L}{\alpha_H}(C_H - 1).$$

Under max weight-α scheduling, and for every $\epsilon > 0$, there exists a constant $\kappa_2(\epsilon) > 0$, such that

$$\mathbb{E}\left[q_L^{\gamma-\epsilon}\right] < \infty \tag{4.20}$$

and

$$\mathbb{P}\{q_L > b\} < \kappa_2(\epsilon) b^{-(\gamma-\epsilon)}. \tag{4.21}$$

Proof: Given $\epsilon > 0$, let us choose $\beta = (C_H - 1)/\alpha_H - \epsilon/\alpha_L$, and perform max-weight-α scheduling with parameters $\beta\alpha_H$ and $\beta\alpha_L$. According to the above observation, this policy is identical to the original max-weight-α policy. Next, since $\beta\alpha_H < C_H - 1$, Theorem 4.5 applies, and we have $\mathbb{E}\left[q_L^{\beta\alpha_L}\right] = \mathbb{E}\left[q_L^{\gamma-\epsilon}\right] < \infty$, which proves (4.20). Finally, (4.21) can be proved using (4.20) and the Markov inequality. □

The above theorem asserts that the tail coefficient of q_L is at least γ under the max weight-α policy. We remark that Theorem 4.4 and Theorem 4.6 both hold for max-weight-α scheduling with any parameters. However, one of them yields a stronger bound than the other, depending on the α parameters. Specifically, we have the following two cases:

(i) $\frac{\alpha_L}{\alpha_H} \leq 1$: This is the regime where the light queue is given lesser priority, when compared to the heavy queue. In this case, Theorem 4.4 yields a stronger bound.

(ii) $\frac{\alpha_L}{\alpha_H} > 1$: This is the regime where the light queue is given more priority compared to the heavy queue. In this case, Theorem 4.6 gives the stronger bound.

Remark 4.1 *The upper bounds in this section hold whenever $H(\cdot)$ is heavy-tailed with tail coefficient C_H. We need the assumption $H(\cdot) \in \mathcal{OR}$ only to derive the lower bounds in the next subsection.*

4.5.2 Lower bound

In this section, we state our main lower bound result, which asymptotically lower bounds the tail of the light queue distribution in terms of the tail of the residual variable H_R.

Theorem 4.7 *Let $H(\cdot) \in \mathcal{OR}$. Then, under max-weight-α scheduling with parameters α_H and α_L, the distribution of the light queue occupancy satisfies the following asymptotic lower bounds:*

1. *If $\frac{\alpha_L}{\alpha_H} < 1$,*

$$\mathbb{P}\{q_L \geq b\} \gtrsim \lambda_H \mathbb{P}\left\{H_R \geq \frac{b}{\lambda_L}\right\} \tag{4.22}$$

2. *If $\frac{\alpha_L}{\alpha_H} = 1$,*

$$\mathbb{P}\{q_L \geq b\} \gtrsim \lambda_H \mathbb{P}\left\{H_R \geq b\left(1 + \frac{1}{\lambda_L}\right)\right\} \tag{4.23}$$

3. If $\frac{\alpha_L}{\alpha_H} > 1$,

$$\mathbb{P}\{q_L \geq b\} \gtrsim \lambda_H \mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H}\right\}. \tag{4.24}$$

As a special case of the above theorem, when $H(\cdot)$ is regularly varying with index C_H, the lower bounds take on a more pleasing power-law form that matches the upper bounds (4.18) and (4.21).

Corollary 4.2 *Suppose $H(\cdot) \in \mathcal{R}(C_H)$. Then, under max-weight-α scheduling with parameters α_H and α_L, the distribution of the light queue satisfies the following asymptotic lower bounds:*

1. If $\frac{\alpha_L}{\alpha_H} \leq 1$,

$$\mathbb{P}\{q_L \geq b\} \gtrsim U(b) b^{-(C_H-1)} \tag{4.25}$$

2. If $\frac{\alpha_L}{\alpha_H} > 1$,

$$\mathbb{P}\{q_L \geq b\} \gtrsim U(b) b^{-\gamma}, \tag{4.26}$$

where $U(\cdot)$ is some slowly varying function.

It takes several steps to prove Theorem 4.7; we start by defining and studying a related fictitious queueing system.

4.5.3 Fictitious system

The fictitious system consists of two queues, fed by the *same input processes* that feed the original system. In the fictitious system, let us call the queues fed by heavy and light traffic $\tilde{H}$ and $\tilde{L}$ respectively. The fictitious system operates under the following service discipline.

Service for the fictitious system: The queue $\tilde{H}$ receives service in every time slot. The queue $\tilde{L}$ receives service at time t if and only if $q_{\tilde{L}}(t)^{\alpha_L} \geq q_{\tilde{H}}(t)^{\alpha_H}$.

Note that if $\tilde{L}$ receives service and $\tilde{H}$ is non-empty, *two* packets are served from the fictitious system. Also, $\tilde{H}$ is just a discrete time $G/D/1$ queue, since it receives service at every time slot. We now show a simple result which asserts that the light queue in the original system is 'longer' than in the fictitious system.

Proposition 4.10 *Suppose a given input sample path feeds the queues in both the original and the fictitious systems. Then, for all t, it holds that $q_{\tilde{L}}(t) \leq q_L(t)$. In particular, for each $b > 0$, we have*

$$\mathbb{P}\{q_L > b\} \geq \mathbb{P}\{q_{\tilde{L}} > b\}.$$

Proof: We will assume the contrary and arrive at a contradiction. Suppose $q_{\tilde{L}}(0) = q_L(0)$, and that for some time $t > 0$, $q_{\tilde{L}}(t) > q_L(t)$. Let $\tau > 0$ be the first time when $q_{\tilde{L}}(\tau) > q_L(\tau)$. It is then necessary that $q_{\tilde{L}}(\tau - 1) = q_L(\tau - 1)$, since no more than one packet is served from a queue in each slot. Next, $q_{\tilde{L}}(\tau - 1) = q_L(\tau - 1)$, and $q_{\tilde{L}}(\tau) > q_L(\tau)$ together imply that L received service at time $\tau - 1$, but $\tilde{L}$ did not. This is possible only if $q_H(\tau - 1) < q_{\tilde{H}}(\tau - 1)$, which is a contradiction, since $\tilde{H}$ receives service in each slot. □

Next, we show that the distribution of $q_{\tilde{L}}$ satisfies the lower bounds in Equations (4.22)-(4.24). Theorem 4.7 then follows, in light of Proposition 4.10.

Theorem 4.8 *In the fictitious system, the distribution of $q_{\tilde{L}}$ is asymptotically lower bounded as follows.*

1. *If $\frac{\alpha_L}{\alpha_H} < 1$,*
$$\mathbb{P}\{q_{\tilde{L}} > b\} \gtrsim \lambda_H \mathbb{P}\left\{H_R > \frac{b}{\lambda_L}\right\} \tag{4.27}$$

2. *If $\frac{\alpha_L}{\alpha_H} = 1$,*
$$\mathbb{P}\{q_{\tilde{L}} > b\} \gtrsim \lambda_H \mathbb{P}\left\{H_R > b\left(1 + \frac{1}{\lambda_L}\right)\right\} \tag{4.28}$$

3. *If $\frac{\alpha_L}{\alpha_H} > 1$,*
$$\mathbb{P}\{q_{\tilde{L}} > b\} \gtrsim \lambda_H \mathbb{P}\left\{H_R > b^{\alpha_L/\alpha_H}\right\} \tag{4.29}$$

Proof: Let us consider an instant t when the fictitious system is in steady-state. Since the heavy queue in the fictitious system receives service in each slot, the steady-state probability $\mathbb{P}\{q_{\tilde{H}} > 0\} = \lambda_H$ by Little's law. Therefore, we have the lower bound

$$\mathbb{P}\{q_{\tilde{L}} > b\} \geq \lambda_H \mathbb{P}\{q_{\tilde{L}} > b | q_{\tilde{H}} > 0\}.$$

In the rest of the proof, we will lower bound the above conditional probability.

Indeed, conditioned on $q_{\tilde{H}} > 0$, denote as before by $\tilde{B}(t)$, the number of packets that belong to the head-of-line burst that still remain in $\tilde{H}$ at time t. Similarly, denote by $\tilde{A}(t)$ the number of packets from the head-of-line burst that have already been served by time t. Since $\tilde{H}$ is served in every time slot, $\tilde{A}(t)$ also denotes the number of time slots that the HoL burst has been in service at $\tilde{H}$.

The reminder of our proof shows that $q_{\tilde{L}}(t)$ stochastically dominates a particular heavy-tailed random variable. Indeed, at the instant t, there are two possibilities:

(a) $q_{\tilde{L}}(t)^{\alpha_L} \geq \tilde{B}(t)^{\alpha_H}$, or

(b) $q_{\tilde{L}}(t)^{\alpha_L} < \tilde{B}(t)^{\alpha_H}$,

Let us take a closer look at case (b) in the following proposition.

Proposition 4.11 *Suppose that*

$$q_{\tilde{L}}(t)^{\alpha_L} < \tilde{B}(t)^{\alpha_H}.$$

Let $\sigma \leq t$ be the instant before t that $\tilde{L}$ last received service. Then, the current head-of-line burst arrived at $\tilde{H}$ after *the instant σ.*

Proof: We have

$$q_{\tilde{H}}(\sigma)^{\alpha_H} \leq q_{\tilde{L}}(\sigma)^{\alpha_L} \leq q_{\tilde{L}}(t)^{\alpha_L} < \tilde{B}(t)^{\alpha_H}.$$

The first inequality holds because $\tilde{L}$ received service at σ, the second inequality is true since $\tilde{L}$ does not receive service between σ and t, and the final inequality is from the hypothesis.

We have shown that $q_{\tilde{H}}(\sigma) < \tilde{B}(t)$, and hence the HoL burst could not have arrived by the time slot σ. $\square$

The above proposition implies that if case (b) holds, $\tilde{L}$ has not received service ever since the HoL burst arrived at $\tilde{H}$. In particular, $\tilde{L}$ has not received service for $\tilde{A}(t)$ time slots, and it accumulates all arrivals that occur during the interval $[t - \tilde{A}(t), t]$.

Let us denote the number of arrivals to $\tilde{L}$ during this interval as

$$S_{\tilde{A}} = \sum_{i=t-\tilde{A}(t)}^{t} L(i).$$

In this notation, our argument above implies that if case (b) holds, then $q_{\tilde{L}}(t) \geq S_{\tilde{A}}$. Putting this together with case (a), we can conclude that

$$q_{\tilde{L}}(t) \geq \min(\tilde{B}(t)^{\alpha_H/\alpha_L},\ S_{\tilde{A}}). \tag{4.30}$$

Therefore,

$$\mathbb{P}\left\{q_{\tilde{L}}(t) > b\right\} \geq \lambda_H \mathbb{P}\left\{\tilde{B}(t)^{\alpha_H/\alpha_L} > b,\ S_{\tilde{A}} > b\right\}. \tag{4.31}$$

Recall now that in steady-state, $\tilde{B}(t)$ is distributed as H_R, and $\tilde{A}(t)$ is distributed as H_A. Therefore, the above bound can be written as

$$\mathbb{P}\left\{q_{\tilde{L}} > b\right\} \geq \lambda_H \mathbb{P}\left\{H_R^{\alpha_H/\alpha_L} > b,\ \sum_{i=1}^{H_A} L(i) > b\right\}. \tag{4.32}$$

Lemma 4.5 shows that

$$\mathbb{P}\left\{H_R^{\alpha_H/\alpha_L} > b,\ \sum_{i=1}^{H_A} L(i) > b\right\} \sim \begin{cases} \mathbb{P}\left\{H_R \geq \frac{b}{\lambda_L}\right\} & \frac{\alpha_L}{\alpha_H} < 1, \\ \mathbb{P}\left\{H_R \geq b + \frac{b}{\lambda_L}\right\} & \frac{\alpha_L}{\alpha_H} = 1, \\ \mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H}\right\} & \frac{\alpha_L}{\alpha_H} > 1. \end{cases}$$

Notice that the assumption $H(\cdot) \in \mathcal{OR}$ is used in the proof of Lemma 4.5.

Theorem 4.8 now follows from the above asymptotic relation and (4.32). □

Proof of Theorem 4.7: The result follows from Theorem 4.8 and Proposition 4.10. □

4.6 Tail Coefficient of q_L

In this section, we characterize the exact tail coefficient of the light queue distribution under max-weight-α scheduling. In particular, we show that the upper bound (4.17) is tight for $\frac{\alpha_L}{\alpha_H} \leq 1$, and (4.20) is tight for $\frac{\alpha_L}{\alpha_H} > 1$.

Theorem 4.9 *The tail coefficient of the steady-state queue length q_L of the light queue is given by*

(i) $C_H - 1$ *for* $\frac{\alpha_L}{\alpha_H} \leq 1$, *and*

(ii) $\gamma = \frac{\alpha_L}{\alpha_H}(C_H - 1)$ *for* $\frac{\alpha_L}{\alpha_H} > 1$.

Proof: Consider first the case $\frac{\alpha_L}{\alpha_H} \leq 1$. The lower order (Definition 4.3) of q_L can be upper bounded using (4.22) or (4.23) as follows

$$\begin{aligned}\liminf_{b\to\infty} -\frac{\log \mathbb{P}\{q_L > b\}}{\log b} &\leq \liminf_{b\to\infty} -\frac{\log \lambda_H + \log \mathbb{P}\left\{H_R \geq \frac{b}{\lambda_L}\right\}}{\log b} \\ &= \liminf_{a\to\infty} -\frac{\log \mathbb{P}\{H_R \geq a\}}{\log a} = C_H - 1.\end{aligned}$$

The last step is from Proposition 4.5. The above equation shows that the tail coefficient of q_L is at most $C_H - 1$. However, it is evident from (4.17) that the tail coefficient of q_L is *at least* $C_H - 1$. Therefore, the tail coefficient of q_L equals $C_H - 1$ for $\frac{\alpha_L}{\alpha_H} \leq 1$. This proves case (i) of the theorem.

Next, consider $\frac{\alpha_L}{\alpha_H} > 1$. Using (4.24), we can upper bound the lower order of q_L as

$$\begin{aligned}\liminf_{b\to\infty} -\frac{\log \mathbb{P}\{q_L > b\}}{\log b} &\leq \liminf_{b\to\infty} -\frac{\log \mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H}\right\}}{\log b} \\ &= \frac{\alpha_L}{\alpha_H} \liminf_{a\to\infty} \frac{-\log \mathbb{P}\{H_R \geq a\}}{\log a} = \frac{\alpha_L}{\alpha_H}(C_H - 1)\end{aligned} \quad (4.33)$$

Equation (5.15) shows that the tail coefficient of q_L is at most γ. However, it is evident from (4.20) that the tail coefficient of q_L is *at least* γ. Therefore, the tail coefficient of q_L equals $\gamma = \frac{\alpha_L}{\alpha_H}(C_H - 1)$ for $\frac{\alpha_L}{\alpha_H} > 1$. This proves case (ii) of the theorem. □

In Figure 4-2, we show the tail coefficient of q_L as a function of the ratio α_L/α_H. We see that the tail coefficient is constant at the value $C_H - 1$ as α_L/α_H varies from 0 to 1. Recall that $\alpha_L/\alpha_H = 1$ corresponds to max-weight scheduling, while $\alpha_L/\alpha_H \downarrow 0$ corresponds to priority for H. Thus, the tail coefficient of q_L under max-weight scheduling is the same as the tail coefficient under priority for H, implying that the max-weight policy leads to the *worst possible* asymptotic behavior for the light queue

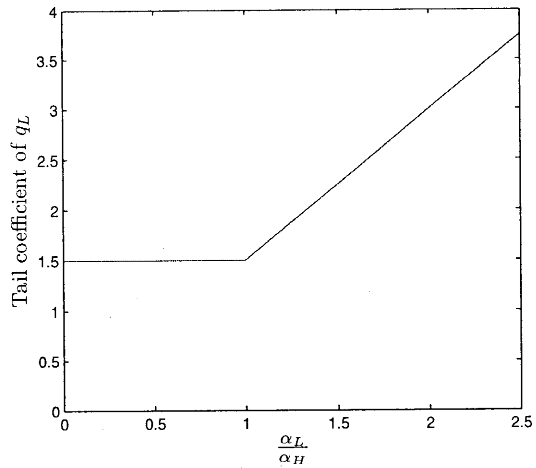


Figure 4-2: The tail coefficient of q_L under max-weight α scheduling, as a function of α_L/α_H, for $C_H = 2.5$.

among all non-idling policies. However, the tail coefficient of q_L begins to improve in proportion to the ratio α_L/α_H in the regime where the light queue is given more importance.

Remark 4.2 *If the heavy-tailed input has an infinite variance ($C_H < 2$), then it follows from Theorem 4.9 that the expected delay in the light queue is infinite under max-weight scheduling. Thus, [40. Proposition 5] is a special case of the above theorem.*

4.7 Log-Max-Weight Scheduling

We showed in Theorem 4.9 that the light queue occupancy distribution is necessarily heavy-tailed with a finite tail coefficient, under max-weight-α scheduling. On the other hand, the priority for L policy which ensures the best possible asymptotic behavior for both queues, suffers from possible instability effects in more general queueing networks.

In this section, we propose and analyze the log-max-weight (LMW) policy. We show that the *light queue distribution is light-tailed* under LMW scheduling, i.e., that

$\mathbb{P}\{q_L > b\}$ decays exponentially fast in b. However, unlike the priority for L policy, LMW scheduling is throughput optimal even in more general settings. For our simple system model, we define the LMW policy as follows:

In each time slot t, the log-max-weight policy compares

$$q_L(t) \gtreqless \log(1 + q_H(t)),$$

and serves one packet from the queue that wins the comparison. Ties are broken in favor of the light queue.

The main idea in the LMW policy is to give preference to the light queue to a far greater extent than any max-weight-α policy. Specifically, for $\alpha_L/\alpha_H > 1$, the max-weight-α policy compares q_L to a power of q_H that is smaller than 1. On the other hand, LMW scheduling compares q_L to a logarithmic function of q_H, leading to a significant preference for the light queue. It turns out that this significant de-emphasis of the heavy queue with respect to the light queue is sufficient to ensure an exponential decay for the distribution of q_L in our setting.

Furthermore, the LMW policy has another useful property when the heavy queue gets overwhelmingly large. Although the LMW policy significantly de-emphasizes the heavy queue, it does not *ignore* it, unlike the priority for L policy. That is, if the H queue occupancy gets overwhelmingly large compared to L, the LMW policy will serve the H queue. In contrast, the priority for L policy will ignore any build-up in H, as long as L is non-empty. This property turns out to be crucial in more complex queueing models, where throughput optimality is non-trivial to obtain. For example, when the queues have time-varying connectivity, the LMW policy will stabilize both queues for all rates within the rate region, whereas priority for L leads to instability effects in H.

Our main result in this section shows that under the LMW policy, $\mathbb{P}\{q_L > b\}$ decays exponentially fast in b, unlike under max-weight-α scheduling.

Theorem 4.10 *Under log-max-weight scheduling, q_L is light-tailed. Specifically, it* *holds that*

$$\liminf_{b\to\infty} -\frac{1}{b}\log \mathbb{P}\{q_L \geq b\} \geq \min(E_L,\ C_H - 1), \tag{4.34}$$

where E_L is the intrinsic exponent, given by (4.14), (4.16).

Proof: Fix a small $\delta > 0$. We first write the equality

$$\begin{aligned}
\mathbb{P}\{q_L \geq b\} &= \underbrace{\mathbb{P}\{q_L \geq b;\ \log(1+q_H) < \delta b\}}_{(i)} \\
&+ \underbrace{\mathbb{P}\{q_L \geq b;\ (1-\delta)b \geq \log(1+q_H) \geq \delta b\}}_{(ii)} \\
&+ \underbrace{\mathbb{P}\{q_L \geq b;\ \log(1+q_H) > (1-\delta)b\}}_{(iii)}
\end{aligned} \tag{4.35}$$

We will next upper bound each of the above three terms on the right.

(i) $\mathbb{P}\{q_L \geq b;\ \log(1+q_H) < \delta b\}$: Intuitively, this event corresponds to an overflow of the light queue, when the light queue is not 'exponentially large' in b, i.e., $q_H < \exp(\delta b) - 1$. Suppose without loss of generality that this event happens at time 0. Denote by $-\tau \leq 0$ the last instant when the heavy queue received service. Since H has not received service since $-\tau$, it is clear that $\log(1+q_H(-\tau)) < \delta b$. Thus, $q_L(-\tau) < \delta b$.

In the time interval $[-\tau+1, 0]$ the light queue receives service in each slot. In spite of receiving all the service, it grows from less than δb to overflow at time 0. This implies that every time the event in (i) occurs, there necessarily exists $-u \leq 0$ satisfying

$$\sum_{i=-u+1}^{0} (L(i) - 1) > (1-\delta)b.$$

Therefore,

$$\mathbb{P}\{q_L \geq b;\ \log(1+q_H) < \delta b\} \leq \mathbb{P}\left\{\exists u \geq 0 \,\middle|\, \sum_{i=-u+1}^{0} (L(i)-1) > (1-\delta)b\right\}.$$

Letting $S_u = \sum_{i=-u+1}^{0} L(i)$, the above inequality can be written as

$$\mathbb{P}\left\{q_L \geq b;\ \log(1+q_H) < \delta b\right\} \leq \mathbb{P}\left\{\sup_{u\geq 0}(S_u - u) > (1-\delta)b\right\}. \tag{4.36}$$

The right hand side of (4.36) is precisely the probability of a single server queue fed by the process $L(\cdot)$ reaching the level $(1-\delta)b$. Standard large deviation bounds are known for such an event. Specifically, from [25 . Lemma 1.5], we get

$$\begin{aligned}\liminf_{b\to\infty} -\frac{1}{b}\log\mathbb{P}\left\{\sup_{u\geq 0} S_u - u > (1-\delta)b\right\} &\geq \inf_{u>0} u\Lambda_L^*\left(1+\frac{1-\delta}{u}\right)\\ &= \inf_{a>0}\frac{1-\delta}{a}\Lambda_L^*(1+a) = (1-\delta)E_L. \end{aligned}\tag{4.37}$$

From (4.36) and (4.37), we see that for every $\epsilon > 0$ and for large enough b,

$$\mathbb{P}\left\{q_L \geq b;\ \log(1+q_H) < \delta b\right\} < \kappa_1 e^{-b(1-\delta)(E_L-\epsilon)}. \tag{4.38}$$

(iii) Let us deal with the term (iii) before (ii). This is the regime where the overflow of L occurs, along with H becoming exponentially large in b. We have

$$\begin{aligned}\mathbb{P}\left\{q_L \geq b;\ \log(1+q_H) > (1-\delta)b\right\} &= \mathbb{P}\left\{q_L \geq b;\ q_H > e^{(1-\delta)b} - 1\right\}\\ &\leq \mathbb{P}\left\{q_L + q_H > e^{(1-\delta)b}\right\}\end{aligned}$$

We have shown earlier in the proof of Theorem 4.4 that for any non-idling policy,

$$\mathbb{P}\left\{q_L + q_H > M\right\} < \kappa_2 M^{-(C_H-1-\epsilon)}$$

for every $\epsilon > 0$ and some $\kappa_2 > 0$. Therefore,

$$\mathbb{P}\left\{q_L \geq b;\ \log(1+q_H) > (1-\delta)b\right\} < \kappa_2 \exp\left(-(1-\delta)b(C_H-1-\epsilon)\right),\ \forall\ \epsilon > 0. \tag{4.39}$$

(ii) Let us now deal with the second term, $\mathbb{P}\left\{q_L \geq b;\ (1-\delta)b \geq \log(1+q_H) \geq \delta b\right\}$.

Let us call this event $\mathcal{E}_2$. Suppose this event occurs at time 0. Denote by $-\tau \leq 0$ the last time during the current busy period that H received service, and define

$$\eta = \log(1 + q_H(-\tau)).$$

If H never received service during the current busy period, we take τ to be equal to the last instant that the system was empty, and $\eta = 0$. We can deduce that $\eta \leq (1-\delta)b$, because H receives no service in $[-\tau+1, 0]$. It is also clear that $q_L(-\tau) < \eta$. Therefore, L grows from less than η to more than b, in spite of receiving all the service in $[-\tau+1, 0]$. Using u and ξ as 'dummy' variables that represent the possible values taken by τ and η respectively, we can write

$$\begin{aligned}
\mathbb{P}\{\mathcal{E}_2\} &\leq \mathbb{P}\left\{\exists\, \xi \leq (1-\delta)b, u \geq 0 \,\middle|\, S_u - u > b - \xi;\ q_H(-u) + q_L(-u) \geq e^{\xi}\right\} \\
&\leq \sum_{\xi=0}^{(1-\delta)b} \mathbb{P}\left\{\exists\, u \geq 0 \,\middle|\, S_u - u > b - \xi;\ q_H(-u) + q_L(-u) \geq e^{\xi}\right\} \\
&\leq \sum_{\xi=0}^{(1-\delta)b} \sum_{u \geq 0} \mathbb{P}\left\{S_u - u > b - \xi;\ q_H(-u) + q_L(-u) \geq e^{\xi}\right\}
\end{aligned}$$

where the last two steps are by the union bound. Notice now that for every $u \geq 0$, the event $S_u - u > b - \xi$ is independent of the value of $q_H(-u) + q_L(-u)$, since these are determined by arrivals in disjoint intervals. Therefore, continuing from above

$$\begin{aligned}
&= \sum_{\xi=0}^{(1-\delta)b} \sum_{u \geq 0} \mathbb{P}\{S_u - u > b - \xi\}\, \mathbb{P}\left\{q_H(-u) + q_L(-u) \geq e^{\xi}\right\} \\
&\leq \sum_{\xi=0}^{(1-\delta)b} \sum_{u \geq 0} \mathbb{P}\{S_u - u > b - \xi\}\, \kappa_2 e^{-(C_H - 1 - \epsilon)\xi}, \ \forall\, \epsilon > 0 \qquad (4.40) \\
&\leq \sum_{\xi=0}^{(1-\delta)b} \kappa_1 e^{-(E_L - \epsilon)(b-\xi)} \kappa_2 e^{-(C_H - 1 - \epsilon)\xi}, \ \forall\, \epsilon > 0. \qquad (4.41)
\end{aligned}$$

Equation (4.40) follows from (4.18), and (4.41) is a classical large deviation bound that follows, for example, from [25, Lemma 1.5]. Thus, for every $\epsilon > 0$,

$$\mathbb{P}\{\mathcal{E}_2\} \leq \sum_{\xi=0}^{(1-\delta)b} \kappa_1 \kappa_2 e^{-[(C_H-1-\epsilon)\xi+(E_L-\epsilon)(b-\xi)]}. \tag{4.42}$$

Let us now distinguish two cases:

– $C_H - 1 > E_L$: In this case, we can bound the above probability as

$$\mathbb{P}\{\mathcal{E}_2\} \leq \kappa e^{-b(E_L-\epsilon)}, \ \forall \epsilon > 0, \tag{4.43}$$

where $\kappa > 0$ is some constant.

– $C_H - 1 \leq E_L$: In this case,

$$\mathbb{P}\{\mathcal{E}_2\} \leq \kappa e^{-b(C_H-1-\epsilon)(1-\delta)}, \ \forall \epsilon > 0. \tag{4.44}$$

Let us now put together the bounds on terms (i), (ii) and (iii) into Equation (4.35).

1. If $C_H - 1 > E_L$, we get from (4.38), (4.39), and (4.43),

$$\mathbb{P}\{q_L \geq b\} < e^{-b(1-\delta)(E_L-\epsilon)} \left[\kappa_1 + \kappa_2 e^{-((1-\delta)b(C_H-1-E_L))} + \kappa\right], \tag{4.45}$$

from which it is immediate that

$$\liminf_{b\to\infty} -\frac{1}{b} \log \mathbb{P}\{q_L \geq b\} \geq (1-\delta)(E_L - \epsilon).$$

Since the above is true for each ϵ and δ, we get

$$\liminf_{b\to\infty} -\frac{1}{b} \log \mathbb{P}\{q_L \geq b\} \geq E_L. \tag{4.46}$$

2. If $C_H - 1 \leq E_L$, we get from (4.38), (4.39), and (4.44),

$$\mathbb{P}\{q_L \geq b\} < e^{-b(1-\delta)(C_H-1-\epsilon)} \left[\kappa_1 e^{-((1-\delta)b(E_L-C_H+1))} + \kappa_2 + \kappa\right], \tag{4.47}$$

from which it is immediate that

$$\liminf_{b\to\infty} -\frac{1}{b}\log \mathbb{P}\{q_L \geq b\} \geq (1-\delta)(C_H - 1 - \epsilon).$$

Since the above is true for each ϵ and δ, we get

$$\liminf_{b\to\infty} -\frac{1}{b}\log \mathbb{P}\{q_L \geq b\} \geq C_H - 1. \tag{4.48}$$

Theorem 4.10 now follows from (4.46) and (4.48). □

Thus, the light queue tail is upper bounded by an exponential term, whose rate of decay is given by the smaller of the intrinsic exponent E_L, and $C_H - 1$. We remark that Theorem 4.10 utilizes only the light-tailed nature of $L(\cdot)$, and the tail coefficient of $H(\cdot)$. Specifically, we do not need to assume any regularity property such as $H(\cdot) \in \mathcal{OR}$ for the result to hold. However, if we assume that the tail of $H(\cdot)$ is regularly varying, we can obtain a matching lower bound to the upper bound in Theorem 4.10.

Theorem 4.11 *Suppose that $H(\cdot) \in \mathcal{R}(C_H)$. Then, under LMW scheduling, the tail distribution of q_L satisfies an LDP with rate function given by*

$$\lim_{b\to\infty} -\frac{1}{b}\log \mathbb{P}\{q_L \geq b\} = \min(E_L,\ C_H - 1).$$

Proof: In light of Theorem 4.10, it is enough to prove that

$$\limsup_{b\to\infty} -\frac{1}{b}\log \mathbb{P}\{q_L \geq b\} \leq \min(E_L,\ C_H - 1).$$

Let us denote by $q_L^{(p)}$ the queue length of the light queue, when it is given complete priority over H. Note that $\mathbb{P}\left\{q_L^{(p)} > b\right\}$ is a lower bound on the overflow probability under *any* policy, including LMW. Therefore, for all $b > 0$, $\mathbb{P}\{q_L \geq b\} \geq \mathbb{P}\left\{q_L^{(p)} > b\right\}$. This implies

$$\limsup_{b\to\infty} -\frac{1}{b}\log \mathbb{P}\{q_L \geq b\} \ \leq\ \limsup_{b\to\infty} -\frac{1}{b}\log \mathbb{P}\left\{q_L^{(p)} > b\right\} = E_L, \tag{4.49}$$

where the last step is from (4.15).

Next, we can show, following the arguments in Proposition 4.10 and Theorem 4.8 that

$$\mathbb{P}\{q_L \geq b\} \geq \lambda_H \mathbb{P}\left\{H_R \geq e^b - 1;\ \sum_{i=1}^{H_A} L(i) \geq b\right\}.$$

But arguing similarly to Lemma 4.5, we can show that

$$\mathbb{P}\left\{H_R \geq e^b - 1;\ \sum_{i=1}^{H_A} L(i) \geq b\right\} \sim \mathbb{P}\left\{H_R \geq e^b - 1\right\}.$$

Thus,

$$\mathbb{P}\{q_L \geq b\} \gtrsim \mathbb{P}\left\{H_R \geq e^b - 1\right\}.$$

Next, since $H(\cdot)$ is regularly varying with tail coefficient C_H, H_R is also regularly varying with tail coefficient C_H-1, so that $\mathbb{P}\left\{H_R \geq e^b - 1\right\} = U(e^b)e^{-b(C_H-1)}$. Finally we can write

$$\limsup_{b\to\infty} -\frac{1}{b}\log \mathbb{P}\{q_L \geq b\} \leq \limsup_{b\to\infty} -\frac{1}{b}\log \mathbb{P}\left\{H_R \geq e^b - 1\right\} = C_H - 1 - \limsup_{b\to\infty} \frac{\log U(e^b)}{b}.$$

The final limit supremum is shown to be zero in Lemma 4.6, using a representation theorem for slowly varying functions. Thus,

$$\limsup_{b\to\infty} -\frac{1}{b}\log \mathbb{P}\{q_L \geq b\} \leq C_H - 1. \tag{4.50}$$

Equations (4.49) and (4.50) imply the theorem. □

Figure 4-3 shows the large deviation exponent given by Theorem 4.11 as a function of λ_L, for $C_H = 2.5$, and Poisson inputs feeding the light queue. There are two distinct regimes in the plot, corresponding to two fundamentally different modes of overflow. For relatively large values of λ_L, the exponent for the LMW policy equals E_L, the intrinsic exponent. In this regime, the light queue overflows entirely due to atypical behavior in the input process $L(\cdot)$. In other words, q_L would have grown close to the level b even if the heavy queue was absent. This mode of overflow is more likely for larger values of λ_L, which explains the diminishing exponent in this regime.

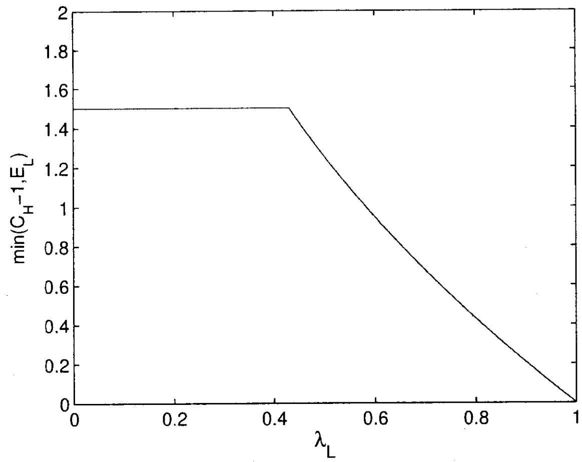


Figure 4-3: The large deviation exponent for q_L under LMW scheduling, as a function of λ_L. The light queue is fed by Poisson bursts, and $C_H = 2.5$.

The flat portion of the curve in Figure 4-3 corresponds to a second overflow mode. In this regime, the overflow of the light queue occurs due to extreme misbehavior on the part of the heavy-tailed input. Specifically, the heavy queue becomes larger than e^b after receiving a very large burst. After this instant, the heavy queue hogs all the service, and the light queue gets starved until it gradually builds up to the level b. In this regime, the light queue input behaves typically, and plays no role in the overflow of L. That is, the exponent is independent of λ_L, being equal to a constant $C_H - 1$. The exponent is decided entirely by the 'burstiness' of the heavy-tailed traffic, which is reflected in the tail coefficient.

4.8 Concluding Remarks

We considered a system of parallel queues fed by a mix of heavy-tailed and light-tailed traffic, and served by a single server. We studied the asymptotic behavior of the queue size distributions under various scheduling policies. We showed that the occupancy distribution of the heavy queue is asymptotically insensitive to the scheduling policy used, and inevitably heavy-tailed. In contrast, the light queue occupancy distribution can be heavy-tailed or light-tailed depending on the scheduling policy.

Our major contribution is in the derivation of an exact asymptotic characterization of the light queue occupancy distribution, under max-weight-α scheduling. We showed that the light queue distribution is heavy-tailed with a finite tail coefficient under max-weight-α scheduling, for any values of the scheduling parameters. However, the tail coefficient can be improved by choosing the scheduling parameters to favor the light queue. We also observed that 'plain' max-weight scheduling leads to the worst possible asymptotic behavior of the light queue distribution, among all non-idling policies.

Another important contribution is the log-max-weight policy, and the corresponding asymptotic analysis. We showed that the light queue occupancy distribution is light-tailed under LMW scheduling, and explicitly derived an exponentially decaying upper bound on the tail of the light queue distribution. Additionally, the LMW policy also has the desirable property of being throughput optimal in a general queueing network.

Although we study a very simple queueing network in this chapter, we believe that the insights obtained from this study are valuable in much more general settings. For instance, in a general queueing network with a mix of light-tailed and heavy-tailed traffic flows, we expect that the celebrated max-weight policy has the tendency to 'infect' competing light-tailed flows with heavy-tailed asymptotics. We also believe that the LMW policy occupies a unique 'sweet spot' in the context of scheduling light-tailed traffic in the presence of heavy-tailed traffic. This is because the LMW policy de-emphasizes the heavy-tailed flow sufficiently to maintain good light queue asymptotics, while also ensuring network-wide stability.

4.A Technical Lemmata

Lemma 4.3 $\mathbb{P}\{H_R \geq m,\ H_A \geq n\} = \mathbb{P}\{H_R \geq m + n\}$

Proof: Using (4.2) and (4.3),

$$\begin{aligned}
\mathbb{P}\{H_R \geq m,\ H_A \geq n\} &= \sum_{k\geq m}\sum_{l\geq n} \frac{\mathbb{P}\{H_+ = k+l\}}{\mathbb{E}[H_+]} \\
&= \sum_{k\geq m}\sum_{p=k+n}^{\infty} \frac{\mathbb{P}\{H_+ = p\}}{\mathbb{E}[H_+]} \\
&= \sum_{k\geq m} \mathbb{P}\{H_R = k+n\} \\
&= \mathbb{P}\{H_R \geq m+n\}.
\end{aligned}$$

□

Lemma 4.4 *Let $N \in \mathcal{IR}$ be a non-negative integer valued random variable. Let $X_i, i \geq 1$ be i.i.d. non-negative light-tailed random variables, with mean μ, independent of N. Define*

$$S_N = \sum_{i=1}^{N} X_i.$$

Then,

$$\mathbb{P}\{S_N > b\} \sim \mathbb{P}\{N > b/\mu\}.$$

Proof: For notational ease, we will prove the result for $\mu = 1$, although the result and proof technique are applicable for any $\mu > 0$. First, for a fixed $\delta > 0$, we have

$$\begin{aligned}
\mathbb{P}\{S_N > b\} &= \mathbb{P}\{S_N > b;\ N \leq b(1-\delta)\} + \mathbb{P}\{S_N > b;\ N > b(1-\delta)\} \\
&< \mathbb{P}\left\{S_{\lfloor b(1-\delta)\rfloor} > b\right\} + \mathbb{P}\{N > b(1-\delta)\}. \qquad (4.51)
\end{aligned}$$

Next, we write a lower bound:

$$\begin{aligned}
\mathbb{P}\{S_N > b\} &\geq \mathbb{P}\{S_N > b;\ N > b(1+\delta)\} \\
&= \mathbb{P}\{N > b(1+\delta)\} - \mathbb{P}\{S_N \leq b;\ N > b(1+\delta)\} \\
&\geq \mathbb{P}\{N > b(1+\delta)\} - \mathbb{P}\left\{S_{\lceil b(1+\delta)\rceil} \leq b\right\}. \qquad (4.52)
\end{aligned}$$

Since the X_i have a well defined moment generating function, their sample average satisfies an exponential concentration inequality around the mean. Specifically, we can show using the Chernoff bound that there exist constants κ, η such that

$$\mathbb{P}\left\{S_{\lfloor b(1-\delta)\rfloor} > b\right\} < \kappa e^{-b\eta}.$$

Thus, it follows that

$$\mathbb{P}\left\{S_{\lfloor b(1-\delta)\rfloor} > b\right\} = o(\mathbb{P}\{N > b\}) \tag{4.53}$$

as $b \to \infty$. Similarly,

$$\mathbb{P}\left\{S_{\lfloor b(1+\delta)\rfloor} \leq b\right\} = o(\mathbb{P}\{N > b\}). \tag{4.54}$$

Next, getting back to (4.51),

$$\limsup_{b\to\infty} \frac{\mathbb{P}\{S_N > b\}}{\mathbb{P}\{N > b\}} \leq \limsup_{b\to\infty} \frac{\mathbb{P}\left\{S_{\lfloor b(1-\delta)\rfloor} > b\right\}}{\mathbb{P}\{N > b\}} + \limsup_{b\to\infty} \frac{\mathbb{P}\{N > b(1-\delta)\}}{\mathbb{P}\{N > b\}}.$$

The first term on the right hand side is zero in view of (4.53), so that for all δ, we have

$$\limsup_{b\to\infty} \frac{\mathbb{P}\{S_N > b\}}{\mathbb{P}\{N > b\}} \leq \limsup_{b\to\infty} \frac{\mathbb{P}\{N > b(1-\delta)\}}{\mathbb{P}\{N > b\}}.$$

Taking the limit as $\delta \downarrow 0$,

$$\limsup_{b\to\infty} \frac{\mathbb{P}\{S_N > b\}}{\mathbb{P}\{N > b\}} \leq \lim_{\delta\downarrow 0} \limsup_{b\to\infty} \frac{\mathbb{P}\{N > b(1-\delta)\}}{\mathbb{P}\{N > b\}} = 1 \tag{4.55}$$

The final limit is unity, by the definition of the class $\mathcal{IR}$. Similarly, we can show using (4.52), (4.54) and the intermediate-regular variation of the tail of N that

$$\liminf_{b\to\infty} \frac{\mathbb{P}\{S_N > b\}}{\mathbb{P}\{N > b\}} \geq 1. \tag{4.56}$$

Equations (4.55) and (4.56) imply the result. □

The above lemma can be proved under more general assumptions than stated here, see [51].

Lemma 4.5 *If $H(\cdot) \in \mathcal{OR}$, we have*

$$\mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H},\ \sum_{i=1}^{H_A} L(i) \geq b\right\} \sim \begin{cases} \mathbb{P}\left\{H_R \geq \frac{b}{\lambda_L}\right\} & \frac{\alpha_L}{\alpha_H} < 1, \\ \mathbb{P}\left\{H_R \geq b + \frac{b}{\lambda_L}\right\} & \frac{\alpha_L}{\alpha_H} = 1, \\ \mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H}\right\} & \frac{\alpha_L}{\alpha_H} > 1. \end{cases} \tag{4.57}$$

Proof: In this proof, let us take $\lambda_L = 1$ for notational simplicity, although the same proof technique works without this assumption. Denote $S_n = \sum_{i=1}^{n} L(i)$. We first get an upper bound. For every $\delta > 0$, we have

$$\begin{aligned}
\mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H};\ S_{H_A} \geq b\right\} &= \\
\mathbb{P}\{\cdot\ ;\ H_A < b(1-\delta)\} &+ \mathbb{P}\{\cdot\ ;\ H_A \geq b(1-\delta)\} \\
\leq \mathbb{P}\{S_{H_A} \geq b;\ H_A < b(1-\delta)\} &+ \mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H};\ H_A \geq b(1-\delta)\right\} \\
\leq \mathbb{P}\{S_{H_A} \geq b;\ H_A < b(1-\delta)\} &+ \mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H} + b(1-\delta)\right\} \qquad (4.58) \\
< \mathbb{P}\left\{S_{\lfloor b(1-\delta)\rfloor} > b\right\} &+ \mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H} + b(1-\delta)\right\}. \qquad (4.59)
\end{aligned}$$

In (4.58) we have utilized Lemma 4.3, and in Equation (4.59), we have used the independence of H_A and $L(\cdot)$. Next, let us derive a lower bound.

$$\begin{aligned}
\mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H};\ S_{H_A} \geq b\right\} &\geq \mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H};\ S_{H_A} \geq b;\ H_A > b(1+\delta)\right\} = \\
\mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H};\ H_A > b(1+\delta)\right\} &- \mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H};\ S_{H_A} < b;\ H_A > b(1+\delta)\right\} \geq \\
\mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H};\ H_A > b(1+\delta)\right\} &- \mathbb{P}\{S_{H_A} < b;\ H_A > b(1+\delta)\} \geq \\
\mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H} + b(1+\delta)\right\} &- \mathbb{P}\left\{S_{\lceil b(1+\delta)\rceil} \leq b\right\}. \qquad (4.60)
\end{aligned}$$

Equation (4.60) uses Lemma 4.3. Now, observe that the terms $\mathbb{P}\left\{S_{\lfloor b(1-\delta)\rfloor} > b\right\}$ in (4.59) and $\mathbb{P}\left\{S_{\lceil b(1+\delta)\rceil} \leq b\right\}$ in (4.60) decay exponentially fast as $b \to \infty$, for any $\delta > 0$. This is because $L(\cdot)$ is light-tailed, and their sample average satisfies an exponential concentration inequality around the mean (unity). More precisely, a Chernoff bound can be used to show that

$$\mathbb{P}\left\{S_{\lfloor b(1-\delta)\rfloor} > b\right\} = o\left(\mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H} + b\right\}\right), \tag{4.61}$$

and

$$\mathbb{P}\left\{S_{\lceil b(1+\delta)\rceil} \leq b\right\} = o\left(\mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H} + b\right\}\right). \tag{4.62}$$

Case (i): $\frac{\alpha_L}{\alpha_H} < 1$. Using (4.59), we write

$$\limsup_{b\to\infty} \frac{\mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H};\ S_{H_A} \geq b\right\}}{\mathbb{P}\{H_R \geq b\}} \leq$$

$$\limsup_{b\to\infty} \frac{\mathbb{P}\left\{S_{\lfloor b(1-\delta)\rfloor} > b\right\}}{\mathbb{P}\{H_R \geq b\}} + \limsup_{b\to\infty} \frac{\mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H} + b(1-\delta)\right\}}{\mathbb{P}\{H_R \geq b\}}.$$

The first limit supremum on the right is zero in view of (4.61). Since $\frac{\alpha_L}{\alpha_H} < 1$, we can write

$$\limsup_{b\to\infty} \frac{\mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H};\ S_{H_A} \geq b\right\}}{\mathbb{P}\{H_R \geq b\}} \leq \limsup_{b\to\infty} \frac{\mathbb{P}\{H_R \geq b(1-\delta)\}}{\mathbb{P}\{H_R \geq b\}},\ \forall\delta > 0.$$

Thus,

$$\limsup_{b\to\infty} \frac{\mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H};\ S_{H_A} \geq b\right\}}{\mathbb{P}\{H_R \geq b\}} \leq \lim_{\delta\downarrow 0} \limsup_{b\to\infty} \frac{\mathbb{P}\{H_R \geq b(1-\delta)\}}{\mathbb{P}\{H_R \geq b\}} = 1. \tag{4.63}$$

The final limit is unity, because according to Lemma 4.1, $H(\cdot) \in \mathcal{OR}$ implies $H_R \in \mathcal{ER}$. Since $\mathcal{ER} \subset \mathcal{IR}$, the final limit in (4.63) is unity, by the definition of intermediate-regular variation (Definition 4.2).

Along similar lines, we can use (4.60), (4.62), and the fact that $H_R \in \mathcal{IR}$ to show that

$$\liminf_{b\to\infty} \frac{\mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H};\ S_{H_A} \geq b\right\}}{\mathbb{P}\{H_R \geq b\}} \geq \lim_{\delta\downarrow 0} \liminf_{b\to\infty} \frac{\mathbb{P}\{H_R \geq b(1+2\delta)\}}{\mathbb{P}\{H_R \geq b\}} = 1. \tag{4.64}$$

Equations (4.63) and (4.64) imply that

$$\mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H};\ S_{H_A} \geq b\right\} \sim \mathbb{P}\left\{H_R \geq b\right\},$$

which implies Lemma 4.5 for $\frac{\alpha_L}{\alpha_H} < 1$, and $\lambda_L = 1$.

Case (ii): $\frac{\alpha_L}{\alpha_H} = 1$. Similar to the previous case. Here, we get

$$\mathbb{P}\left\{H_R \geq b;\ S_{H_A} \geq b\right\} \sim \mathbb{P}\left\{H_R \geq 2b\right\}.$$

Case (iii): $\frac{\alpha_L}{\alpha_H} > 1$.

For the upper bound, we have from (4.59) and (4.53),

$$\limsup_{b\to\infty} \frac{\mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H};\ S_{H_A} \geq b\right\}}{\mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H}\right\}} \leq \limsup_{b\to\infty} \frac{\mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H} + b(1-\delta)\right\}}{\mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H}\right\}} \leq 1.$$

Similarly, for the lower bound, we have from (4.60) and (4.54),

$$\liminf_{b\to\infty} \frac{\mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H};\ S_{H_A} \geq b\right\}}{\mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H}\right\}} \geq \liminf_{b\to\infty} \frac{\mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H} + b(1+\delta)\right\}}{\mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H}\right\}},$$

$$\geq \liminf_{b\to\infty} \frac{\mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H}(1+\delta)\right\}}{\mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H}\right\}}, \ \forall\delta > 0.$$

Thus,

$$\liminf_{b\to\infty} \frac{\mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H};\ S_{H_A} \geq b\right\}}{\mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H}\right\}} \geq \lim_{\delta\downarrow 0} \liminf_{b\to\infty} \frac{\mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H}(1+\delta)\right\}}{\mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H}\right\}} = 1,$$

where the last limit is unity due to the intermediate-regular variation of H_R. Therefore, we can conclude for $\frac{\alpha_L}{\alpha_H} > 1$ that

$$\mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H};\ S_{H_A} \geq b\right\} \sim \mathbb{P}\left\{H_R \geq b^{\alpha_L/\alpha_H}\right\}.$$

Lemma 4.5 is now proved. $\square$

Lemma 4.6 *For any slowly varying function $U(\cdot)$,*

$$\lim_{a\to\infty} \frac{\log U(a)}{\log a} = 0.$$

Proof: We use the representation theorem for slowly varying functions derived in [22]. For every slowly varying function $U(\cdot)$, there exists a $B > 0$ such that for all $x \geq B$, the function can be written as

$$U(x) = \exp\left(v(x) + \int_B^x \frac{\zeta(y)}{y} \mathrm{d}y\right),$$

where $v(x)$ converges to a finite constant, and $\zeta(x) \to 0$ as $x \to \infty$. Therefore,

$$\lim_{a\to\infty} \frac{\log U(a)}{\log a} = \lim_{a\to\infty} \frac{v(a) + \int_B^a \frac{\zeta(y)}{y} \mathrm{d}y}{\log a} = \lim_{a\to\infty} \frac{\int_B^a \frac{\zeta(y)}{y} \mathrm{d}y}{\log a},$$

where the last step is because $v(a)$ converges to a constant. Next, given any $\epsilon > 0$, choose $C(\epsilon)$ such that $|\zeta(a)| < \epsilon$, $\forall\ a > C(\epsilon)$. Then, we have

$$\lim_{a\to\infty} \frac{\left|\int_B^a \frac{\zeta(y)}{y} \mathrm{d}y\right|}{\log a} \leq \lim_{a\to\infty} \frac{\int_B^{C(\epsilon)} \frac{|\zeta(y)|}{y} \mathrm{d}y + \int_{C(\epsilon)}^a \frac{|\zeta(y)|}{y} \mathrm{d}y}{\log a} \leq \lim_{a\to\infty} \frac{\epsilon \log \frac{a}{C(\epsilon)}}{\log a} = \epsilon.$$

Since the above is true for every ϵ, the result follows. □

Chapter 5

Throughput Optimal Scheduling in the presence of Heavy-Tailed Traffic

In this chapter, we extend the results obtained in the previous chapter to the setting where the two queues are connected to the server through randomly time-varying links. This channel model can be used to represent fading wireless links in a two-user up-link or down-link system. We study this model for two main reasons. First, in this setting, throughput optimality turns out to be a non-trivial objective. That is, while any non-idling policy stabilizes the system considered in the previous chapter, queue length blind scheduling policies (such as priority) generally fail to stabilize the system when the channels are time-varying. Second, it turns out that the asymptotic behavior of the queue length distributions under a given policy strongly depends on the arrival rates. In fact, under a given scheduling policy, we derive vastly different queue length behaviors in different parts of the rate region – an effect not observed in the previous chapter.

We now state the specific assumptions about the system model.

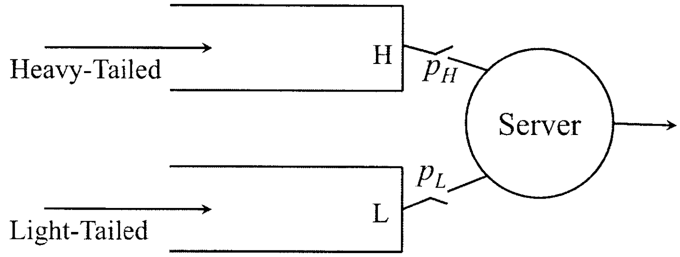


Figure 5-1: A system of two parallel queues, with one of them fed with heavy-tailed traffic. The channels connecting the queues to the server are unreliable ON/OFF links.

5.1 System Model

The system shown in Figure 5-1 consists of two parallel queues H and L, with heavy-tailed traffic feeding H, and light-tailed traffic feeding L. The only main difference from the model in the previous chapter is that the channels connecting the queues to the server are time-varying links. Let $S_H(t) \in \{0, 1\}$ and $S_L(t) \in \{0, 1\}$ respectively denote the states of the channels connecting the H and L queues to the server. When a channel is in state 0, it is OFF, and no packets can be served from the corresponding queue in that slot. When a channel is in state 1, it is ON, and a packet can be served from the corresponding queue if the server is assigned to that queue. We assume that the scheduler can observe the channel states $S_L(t)$ and $S_H(t)$ before making a scheduling decision during time slot t.

The processes $S_H(t)$ and $S_L(t)$ are independent of each other, and independent of the arrival processes and the current queue occupancies. We assume that $S_H(t)$ and $S_L(t)$ are i.i.d. from slot to slot, distributed according to Bernoulli processes with positive means p_H and p_L respectively. That is, $\mathbb{P}\{S_i(\cdot) = 1\} = p_i,\ i \in \{H, L\}$. We say that a particular time slot t is *exclusive* to H, if $S_H(t) = 1$ and $S_L(t) = 0$, and similarly for L.

The assumptions on the arrival processes are almost the same as before, except that we do not assume any regularity property for the heavy-tailed input distribution. More precisely, our assumptions on the input distributions are as follows.

1. The arrival processes $H(t)$ and $L(t)$ are independent of each other, and inde-

pendent of the processes $S_H(t)$ and $S_L(t)$.

2. $H(t)$ is i.i.d. from slot-to-slot.

3. $L(t)$ is i.i.d. from slot-to-slot.

4. $L(\cdot)$ is light-tailed[1], with $\mathbb{E}[L(t)] = \lambda_L$.

5. $H(\cdot)$ is heavy-tailed, with tail coefficient[2] C_H, $(1 < C_H < \infty)$, and $\mathbb{E}[H(\cdot)] = \lambda_H$.

The conditions for a rate pair (λ_H, λ_L) to be stably supportable in this system are known in the literature. Specifically, it follows from the results in [64] that the rate region of the system is given by

$$\{(\lambda_H, \lambda_L) \mid 0 \leq \lambda_L < p_L,\ 0 \leq \lambda_H < p_H,\ \lambda_H + \lambda_L < p_H + p_L - p_H p_L\}. \tag{5.1}$$

Thus, the rate region is pentagonal, as illustrated by the solid line in Figure 5-2. Since we only derive *moment bounds* on the steady-state queue occupancy in this chapter, and not *distributional bounds* as we did in the previous chapter, we are able to do away with the order-regularity assumption on $H(\cdot)$.

We now proceed to analyze the behavior of the queue lengths in this system under three scheduling policies, namely, priority for L, max-weight-α, and LMW.

5.2 Priority for the Light-Tailed Traffic

Under priority for L, the light queue is served whenever its channel is ON, and L is non-empty. The heavy queue is served during the exclusive slots of H, and in the slots when both channels are ON, but L is empty.

Recall from Section 4.4.2 that when the queues are always connected to the server, priority for L leads to the best possible asymptotic behavior for both queues. In that simple setting, priority for L was also enough to stabilize all arrival rates in the rate

[1]See Definition 4.1.
[2]See Definitions 4.1 and 4.4.

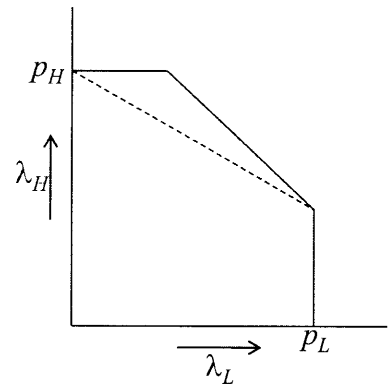


Figure 5-2: The rate region of the system is shown in solid line, and the set of stabilizable rates under priority for L is the region under the dashed line .

region of the system. However, in the present setting with time-varying channels, priority for L fails to stabilize the heavy queue for some arrival rates within the rate region in (5.1). The following theorem characterizes the behavior of both queues under priority for L.

Theorem 5.1 *The following statements hold under priority scheduling for L.*

(i) If $\lambda_H > p_H(1-\lambda_L)$, the heavy queue is unstable, *and no steady-state exists.*

(ii) If $\lambda_H < p_H(1-\lambda_L)$, the heavy queue is stable, and its steady-state occupancy q_H is heavy-tailed with tail coefficient $C_H - 1$.

(iii) q_L is light-tailed and satisfies the LDP

$$\lim_{b\to\infty} -\frac{1}{b}\log \mathbb{P}\left\{q_L > b\right\} = I_L,$$

where I_L is the intrinsic exponent of the light queue[3] *given by*

$$I_L = \sup\left\{\theta \,\middle|\, \Lambda_L(\theta) - \log\left(1 - p_L + p_L e^{\theta}\right) < 0\right\}. \tag{5.2}$$

In Figure 5-2, the line $\lambda_H = p_H(1-\lambda_L)$ is shown using a dashed segment. The above theorem asserts that H is stable under priority for L only in the trapezoidal

[3]Note that I_L equals E_L defined in (4.14) when $p_L = 1$.

region under the dashed line, while the rate region of the system is clearly larger. Therefore, priority for L is *not* throughput optimal in this setting. To summarize, priority for L can lead to instability of the heavy queue, but for all arrival rates that it *can* stabilize, the asymptotic behavior of both queues is as good as it can possibly be. Let us now prove the above theorem.

Proof: First, we note that the light queue behaves like a discrete time G/M/1 queue under priority, since the service time for each packet is geometrically distributed with mean $1/p_L$. Thus, q_L is light-tailed, and satisfies the same LDP as a G/M/1 queue. Statement (iii) therefore follows from classical large deviation results.

Let us now prove statement (i) of the theorem. Under priority for L, denote by $\hat{D}_H(t) \in \{0, 1\}$ the indicator of service opportunity afforded to the heavy queue in slot t. Thus, $\hat{D}_H(t) = 1$ if H is ON and the server is assigned to H during slot t, and zero otherwise. Note that $\hat{D}_H(t) = 1$ does not necessarily imply a departure from the heavy queue in that slot, since H could be empty. We will compute the long term average rate of service opportunities given to H under priority for L, defined as

$$\lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \hat{D}_H(t).$$

Since the light queue behaves as a G/M/1 queue, the intervals between successive commencements of busy periods of L are renewal intervals. Let us denote by X_L a random variable representing the length of a renewal interval. Also denote by $\overline{B}$ and $\overline{I}$ respectively the average length of a busy and idle period of L. The average length of a renewal interval is therefore $\mathbb{E}[X_L] = \overline{B} + \overline{I}$. Consider now the total number of service opportunities $\hat{d}_H(i)$ given to H during the ith renewal interval. $\hat{d}_H(i)$ equals the number of exclusive slots of H during the renewal interval, plus the number of slots when both channels are ON and L is empty. Therefore, it is clear that $\hat{d}_H(i)$ depends only on the events in the ith renewal interval, and is a valid reward process.

We can now invoke the renewal reward theorem [24] and write (almost surely)

$$\lim_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} \hat{D}_H(t) = \frac{\mathbb{E}\left[\hat{d}_H(i)\right]}{\overline{B}+\overline{I}}.$$

Let us now compute $\mathbb{E}\left[\hat{d}_H(i)\right]$. First, the average number of exclusive slots of H during a renewal interval is given by $p_H(1-p_L)(\overline{B}+\overline{I})$. Second, the average number of slots when both channels are ON, and L is empty is given by $\overline{I}p_Hp_L$. Therefore, $\mathbb{E}\left[\hat{d}_H(i)\right] = p_H(1-p_L)(\overline{B}+\overline{I})+\overline{I}p_Hp_L$. Substituting this in the reward theorem, we get

$$\lim_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} \hat{D}_H(t) = p_H(1-p_L) + p_Hp_L\frac{\overline{I}}{\overline{B}+\overline{I}}.$$

Next, note that according to Little's law,

$$\frac{\overline{I}}{\overline{B}+\overline{I}} = 1-\frac{\lambda_L}{p_L}.$$

Therefore,

$$\lim_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} \hat{D}_H(t) = p_H(1-p_L) + p_Hp_L\left(1-\frac{\lambda_L}{p_L}\right) = p_H(1-\lambda_L). \tag{5.3}$$

Thus, the average service rate of H almost surely equals $p_H(1-\lambda_L)$. If $\lambda_H > p_H(1-\lambda_L)$, then the average service rate given to the heavy queue is dominated by the average arrival rate, leading to the instability of H. This proves statement (i).

Before we prove statement (ii) of the theorem, we state a useful result regarding random sums of a light-tailed number of heavy-tailed terms.

Proposition 5.1 *Let N be a non-negative, integer valued, light-tailed random variable that is not identically equal to zero. Let Y_i, $i = 1, 2, \ldots$ be i.i.d. heavy-tailed random variables, independent of N, with tail coefficient C_Y. Define*

$$S_N = \sum_{i=1}^{N} Y_i.$$

Then, S_N is heavy-tailed with tail coefficient C_Y.

Proof: Since N is not identically zero, it suffices to show that

$$\mathbb{E}\left[S_N^{C_Y-\delta}\right] < \infty, \ \forall\, \delta > 0.$$

Indeed, for a fixed n, note that $S_n^{C_Y-\delta} < n^{C_Y-\delta} \cdot \left(\sum_{i=1}^n Y_i^{C_Y-\delta}\right)$. Therefore, for a fixed n, we have

$$\mathbb{E}\left[S_n^{C_Y-\delta}\right] < n^{C_Y-\delta}\mathbb{E}\left[\sum_{i=1}^n Y_i^{C_Y-\delta}\right] = n^{C_Y+1-\delta}\mathbb{E}\left[Y_1^{C_Y-\delta}\right] = Mn^{C_Y+1-\delta},$$

for some finite M. Finally, we use iterated expectations, and the assumption that N is light-tailed to write

$$\mathbb{E}\left[S_N^{C_Y-\delta}\right] = \mathbb{E}_N\left[\mathbb{E}\left[S_N^{C_Y-\delta}|N\right]\right] < E_N\left[MN^{C_Y+1-\delta}\right] < \infty.$$

□

We are now ready to prove statement (ii) of the theorem. We need to show that when $\lambda_H < p_H(1-\lambda_L)$, the system is stable and the steady state queue length of H has tail coefficient equal to $C_H - 1$.

Let us first show that the heavy queue is stable under priority for L, and that the steady-state value q_H exists, when $\lambda_H < p_H(1-\lambda_L)$. We do this by considering a linear Lyapunov function, and showing a negative drift over long time frames. Specifically, choose T_0 large enough, and consider the expected drift

$$\frac{1}{T_0}\mathbb{E}\left[q_H(t+T_0) - q_H(t)\,|q_H(t)\right] = \frac{1}{T_0}\mathbb{E}\left[\sum_{k=1}^{T_0} H(t+k) - \sum_{k=1}^{T_0} D_H(t+k)\Big|q_H(t)\right].$$

$$\frac{1}{T_0}\mathbb{E}\left[\sum_{k=1}^{T_0} H(t+k) - \sum_{k=1}^{T_0} \hat{D}_H(t+k)\Big|q_H(t)\right] + \frac{1}{T_0}\mathbb{E}\left[\sum_{k=1}^{T_0} \hat{D}_H(t+k) - \sum_{k=1}^{T_0} D_H(t+k)\Big|q_H(t)\right].$$

The second expectation above is the number of lost departures due to an empty H

queue. In particular, the second expectation is zero if $q_H(t) > T_0$. Thus,

$$\frac{1}{T_0}\mathbb{E}\left[q_H(t+T_0) - q_H(t)\Big|q_H(t)\right] \leq \frac{1}{T_0}\mathbb{E}\left[\sum_{k=1}^{T_0} H(t+k) - \sum_{k=1}^{T_0} \hat{D}_H(t+k)\Big|q_H(t)\right] + 1_{q_H(t)<T_0}$$
$$= \lambda_H - \mathbb{E}\left[\frac{1}{T_0}\sum_{k=1}^{T_0} \hat{D}_H(t+k)\Big|q_H(t)\right] + 1_{q_H(t)<T_0}. \quad (5.4)$$

Next, let $\delta > 0$ be such that $\lambda_H + 2\delta < p_H(1-\lambda_L)$. In view of (5.3), we have for all t,

$$\lim_{T\to\infty} \frac{1}{T}\sum_{k=1}^{T} \hat{D}_H(t+k) = p_H(1-\lambda_L),$$

almost surely. By the dominated convergence theorem [8 . Theorem 16.4], it follows that

$$\lim_{T\to\infty} \mathbb{E}\left[\frac{1}{T}\sum_{k=1}^{T} \hat{D}_H(t+k)\right] = p_H(1-\lambda_L).$$

Since the above holds for all t, a T_0 can be chosen large enough such that

$$\mathbb{E}\left[\frac{1}{T_0}\sum_{k=1}^{T_0} \hat{D}_H(t+k)\Big|q_H(t)\right] > p_H(1-\lambda_L) - \delta,$$

regardless of the value of $q_H(t)$. Therefore, returning to the drift expression (5.4), we have

$$\frac{1}{T_0}\mathbb{E}\left[q_H(t+T_0) - q_H(t)\Big|q_H(t)\right] < \lambda_H - (p_H(1-\lambda_L) - \delta) + 1_{q_H(t)<T_0} < -\delta + 1_{q_H(t)<T_0}.$$

The above term is negative for $q_H > T_0$, so that by the Foster criterion [4], H is stable, and the steady-state queue length q_H exists.

To complete the proof of part (ii) of the theorem, we finally show that q_H has tail coefficient equal to $C_H - 1$. It is enough to show that $\mathbb{E}[q_H^\alpha] < \infty$ for all $\alpha < C_H - 1$. This is because we already know from Theorem 4.1 that the tail coefficient of q_H is no larger than $C_H - 1$, even if the heavy queue is always ON and always served.

Let us consider the renewal intervals corresponding to successive commencement of busy periods of L as above. Denote by T_i the time slot at which the ith renewal period commences. The heavy queue occupancy sampled at the renewal epochs can be shown to form a Markov chain. That is, $Q_H(i) := q_H(T_i), i \in \mathbb{Z}$, forms a Markov chain with the non-negative integers as its state space. We will show that when $\lambda_H < p_H(1-\lambda_L)$, this Markov chain is positive recurrent, and that its steady-state Q_H satisfies $\mathbb{E}\left[Q_H^{\alpha}\right] < \infty$ for all $\alpha < C_H - 1$.

Consider the Lyapunov function

$$V(x) = \frac{x^{1+\alpha}}{\alpha+1},\ 0 < \alpha < C_H - 1,$$

defined on the state-space of the Markov chain. We will compute the expected drift in the Lyapunov function, defined as

$$\mathbb{E}\left[V(Q_H(i+1)) - V(Q_H(i)) | Q_H(i)\right].$$

The queue evolution equation is given by

$$Q_H(i+1) = Q_H(i) + a_H(i) - d_H(i),$$

where we have used $a_H(i)$ and $d_H(i)$ to respectively denote the total number of arrivals to H and departures from H, during the ith renewal interval. We will now consider two sub-cases, (a) $C_H < 2$, $\alpha < 1$, and (b) $C_H > 2$, $\alpha > 1$.

(a) $\alpha < C_H - 1 < 1$. In this case, we have by Taylor's theorem

$$V(Q_H(i+1)) = \frac{Q_H(i+1)^{1+\alpha}}{1+\alpha} = \frac{(Q_H(i) + a_H(i) - d_H(i))^{1+\alpha}}{1+\alpha} = V(Q_H(i)) + \Delta_H(i)\xi^{\alpha},$$

where $\Delta_H(i) = a_H(i) - d_H(i)$, and $\xi \in (Q_H(i) - d_H(i), Q_H(i) + a_H(i))$. Therefore,

$$\mathbb{E}\left[V(Q_H(i+1)) - V(Q_H(i)) | Q_H(i)\right] = \mathbb{E}\left[\Delta_H(i)\xi^{\alpha} | Q_H(i)\right]$$

$$= \mathbb{E}\left[\Delta_H(i)\xi^{\alpha};\ \Delta_H(i) < 0 | Q_H(i)\right] + \mathbb{E}\left[\Delta_H(i)\xi^{\alpha};\ \Delta_H(i) \geq 0 | Q_H(i)\right]$$

$$\leq \mathbb{E}\left[\Delta_H(i)(Q_H(i)-d_H(i))^\alpha;\ \Delta_H(i)<0|Q_H(i)\right]+$$

$$\mathbb{E}\left[\Delta_H(i)(Q_H(i)+a_H(i))^\alpha;\ \Delta_H(i)\geq 0|Q_H(i)\right].$$

Since $\alpha < 1$ in this case, we have $(Q_H(i) - d_H(i))^\alpha \geq Q_H(i)^\alpha - d_H(i)^\alpha$, and $(Q_H(i) + a_H(i))^\alpha \leq Q_H(i)^\alpha + a_H(i)^\alpha$. Continuing the upper bound on the Lyapunov drift, we write

$$\mathbb{E}\left[V(Q_H(i+1)) - V(Q_H(i))|Q_H(i)\right] \leq$$

$$\mathbb{E}\left[\Delta_H(i)(Q_H(i)^\alpha - d_H(i)^\alpha);\ \Delta_H(i)<0|Q_H(i)\right]+$$

$$\mathbb{E}\left[\Delta_H(i)(Q_H(i)^\alpha + a_H(i)^\alpha);\ \Delta_H(i)\geq 0|Q_H(i)\right]$$

$$= \mathbb{E}\left[\Delta_H(i)Q_H(i)^\alpha|Q_H(i)\right] + \mathbb{E}\left[\Delta_H(i)(-d_H(i)^\alpha);\ \Delta_H(i)<0|Q_H(i)\right]$$

$$+\mathbb{E}\left[\Delta_H(i)(a_H(i)^\alpha);\ \Delta_H(i)\geq 0|Q_H(i)\right]$$

$$\leq \mathbb{E}\left[\Delta_H(i)Q_H(i)^\alpha|Q_H(i)\right] + \mathbb{E}\left[a_H(i)^{1+\alpha} + d_H(i)^{1+\alpha}|Q_H(i)\right]$$

The second expectation in the right-hand side above can be shown to be finite. To see this, first note that $d_H(i)$ is upper bounded by the renewal interval, which is light-tailed. Second, $a_H(i)$ is a sum of a light-tailed number of heavy-tailed terms, which according to Proposition 5.1 has tail coefficient equal to C_H. Therefore, the Lyapunov drift is upper bounded as

$$\mathbb{E}\left[V(Q_H(i+1)) - V(Q_H(i))|Q_H(i)\right] \leq W + \mathbb{E}\left[\Delta_H(i)Q_H(i)^\alpha|Q_H(i)\right]$$

$$= W + Q_H(i)^\alpha \mathbb{E}\left[a_H(i) - d_H(i)|Q_H(i)\right],$$

for some finite constant W. Recall now that $\hat{d}_H(i)$ denotes the number of service opportunities given to H during the ith renewal period, and that $\hat{d}_H(i) \geq d_H(i)$.

We can now write the above drift bound as

$$\mathbb{E}\left[V(Q_H(i+1)) - V(Q_H(i))|Q_H(i)\right] \le \\ W + Q_H(i)^\alpha \left(\mathbb{E}\left[a_H(i) - \hat{d}_H(i)|Q_H(i)\right] + \mathbb{E}\left[\hat{d}_H(i) - d_H(i)|Q_H(i)\right]\right).$$

Next, $a_H(i)$ and $\hat{d}_H(i)$ are independent of $Q_H(i)$, and $\mathbb{E}[a_H(i)] = \lambda_H \mathbb{E}[X_L]$, where X_L is the renewal interval. It was shown earlier in the proof that $\mathbb{E}\left[\hat{d}_H(i)\right] = \mathbb{E}[X_L] p_H(1 - \lambda_L)$. Thus,

$$\mathbb{E}\left[V(Q_H(i+1)) - V(Q_H(i))|Q_H(i)\right] \le W + \mathbb{E}[X_L](\lambda_H - p_H(1-\lambda_L))Q_H(i)^\alpha \\ + Q_H(i)^\alpha \mathbb{E}\left[\hat{d}_H(i) - d_H(i)|Q_H(i)\right].$$

Since $\lambda_H - p_H(1 - \lambda_L) < 0$ by assumption, there exists a $\delta > 0$ such that $\lambda_H - p_H(1 - \lambda_L) < -\delta$. Therefore,

$$\mathbb{E}\left[V(Q_H(i+1)) - V(Q_H(i))|Q_H(i)\right] \le \\ W + Q_H(i)^\alpha \mathbb{E}\left[\hat{d}_H(i) - d_H(i)|Q_H(i)\right] - \delta Q_H(i)^\alpha.$$

Next, we will upper bound the remaining expectation term on the right side. Notice that $\hat{d}_H(i) - d_H(i)$ is simply the total number of lost departures from H due to the queue being empty. In particular, $\hat{d}_H(i) - d_H(i) = 0$ if $Q_H(i) \ge X_L(i)$, where $X_L(i)$ is the ith renewal interval. In general, it holds that $\hat{d}_H(i) - d_H(i) \le X_L(i) 1_{X_L(i) > Q_H(i)}$. Therefore,

$$\mathbb{E}\left[\hat{d}_H(i) - d_H(i)|Q_H(i)\right] \le \mathbb{E}\left[X_L(i);\ X_L(i) > Q_H(i)\right],$$

and

$$\mathbb{E}\left[V(Q_H(i+1)) - V(Q_H(i))|Q_H(i)\right] \le \\ W + Q_H(i)^\alpha \mathbb{E}\left[X_L(i);\ X_L(i) > Q_H(i)\right] - \delta Q_H(i)^\alpha.$$

We finally show that the term $Q_H(i)^\alpha \mathbb{E}\left[X_L(i);\ X_L(i) > Q_H(i)\right]$ is bounded for all values of $Q_L(i)$. It is enough to consider large values of $Q_H(i)$, since the term clearly stays bounded for bounded values of $Q_H(i)$. Indeed, since X_L is light-tailed (being the renewal period of the light queue), $\mathbb{E}\left[X_L(i);\ X_L(i) > b\right]$ decays exponentially fast in b. Therefore, for large enough values of $Q_H(i)$, the term $Q_H(i)^\alpha \mathbb{E}\left[X_L(i);\ X_L(i) > Q_H(i)\right]$ can be made arbitrarily small, implying boundedness. Finally then, we can bound the Lyapunov drift as

$$\mathbb{E}\left[V(Q_H(i+1)) - V(Q_H(i))|Q_H(i)\right] \leq W' - \delta Q_H(i)^\alpha,$$

for some constant W'. Due to Foster's criterion, the Markov chain $Q_H(i),\ i \in \mathbb{Z}$, is positive recurrent, and at steady-state $\mathbb{E}\left[Q_H^\alpha\right] < \infty$ for all $\alpha < C_H - 1 < 1$.

(b) $1 < \alpha < C_H - 1$. In this case, we have by Taylor's theorem

$$V(Q_H(i+1)) = \frac{Q_H(i+1)^{1+\alpha}}{1+\alpha} = \frac{(Q_H(i) + a_H(i) - d_H(i))^{1+\alpha}}{1+\alpha}$$

$$= V(Q_H(i)) + Q_H(i)^\alpha \Delta_H(i) + \alpha \frac{\Delta_H(i)^2}{2} \xi^{\alpha-1},$$

where $\xi \in (Q_H(i) - d_H(i), Q_H(i) + a_H(i))$. Thus,

$$\mathbb{E}\left[V(Q_H(i+1)) - V(Q_H(i))|Q_H(i)\right] \leq$$

$$Q_H(i)^\alpha \mathbb{E}\left[\Delta_H(i)|Q_H(i)\right] + \mathbb{E}\left[\alpha \frac{\Delta_H(i)^2}{2} \xi^{\alpha-1} |Q_H(i)\right]$$

$$\leq Q_H(i)^\alpha \mathbb{E}\left[\Delta_H(i)|Q_H(i)\right] + \alpha \mathbb{E}\left[\frac{\Delta_H(i)^2}{2} (Q_H(i) + a_H(i))^{\alpha-1} |Q_H(i)\right].$$

Since $(Q_H(i) + a_H(i))^{\alpha-1} < 2^{\alpha-1}(Q_H(i)^{\alpha-1} + a_H(i)^{\alpha-1})$, we can continue thus

$$\leq Q_H(i)^\alpha \mathbb{E}\left[\Delta_H(i)|Q_H(i)\right] + \alpha \mathbb{E}\left[\Delta_H(i)^2 2^{\alpha-2}(Q_H(i)^{\alpha-1} + a_H(i)^{\alpha-1})|Q_H(i)\right],$$

$$\leq Q_H(i)^\alpha \mathbb{E}\left[\Delta_H(i)|Q_H(i)\right] + 2^{\alpha-2} \alpha Q_H(i)^{\alpha-1} \mathbb{E}\left[(a_H(i) + X_L(i))^2|Q_H(i)\right] +$$

$$2^{\alpha-2}\alpha\mathbb{E}\left[(a_H(i)+X_L(i))^2 a_H(i)^{\alpha-1}|Q_H(i)\right].$$

The last two expectation terms above are finite, since according to Proposition 5.1, the tail coefficient of $a_H(i)$ is C_H. Thus, for every $\delta > 0$, there exists a $W(\delta)$ such that

$$\mathbb{E}\left[V(Q_H(i+1)) - V(Q_H(i))|Q_H(i)\right] \leq Q_H(i)^\alpha \mathbb{E}\left[\Delta_H(i)|Q_H(i)\right] + W(\delta) + \delta Q_H(i)^\alpha.$$

Finally, the term $\mathbb{E}\left[\Delta_H(i)|Q_H(i)\right]$ can be bounded as in case (a) to finally yield

$$\mathbb{E}\left[V(Q_H(i+1)) - V(Q_H(i))|Q_H(i)\right] \leq W'(\delta) - \delta Q_H(i)^\alpha,$$

for some bounded $W'(\delta)$. Again, Foster's criterion implies positive recurrence of $Q_H(i),\ i \in \mathbb{Z}$, and at steady-state, $\mathbb{E}\left[Q_H^\alpha\right] < \infty$ for all $1 < \alpha < C_H - 1$.

Cases (a) and (b) above imply the positive recurrence of $Q_H(i),\ i \in \mathbb{Z}$ for all values of C_H, and that $\mathbb{E}\left[Q_H^{C_H-1-\epsilon}\right] < \infty$ for all $\epsilon > 0$. We are not quite done yet, as we need to show that $\mathbb{E}\left[q_H^{C_H-1-\epsilon}\right] < \infty$ for the steady-state queue length q_H.

Indeed, during an instant at steady-state, say $t = 0$, consider the most recent commencement of a light queue busy period, and denote it by $-\tau \leq 0$. Then, $q_H(-\tau)$, which is the heavy queue occupancy sampled at that renewal instant, is distributed as Q_H, the steady-state distribution of the Markov chain $Q_H(i),\ i \in \mathbb{Z}$. We then have $q_H(0) \leq q_H(-\tau) + \sum_{i=-\tau}^{0} H(i)$, so that

$$\mathbb{E}\left[q_H(0)^{C_H-1-\epsilon}\right] \leq \mathbb{E}\left[\left(q_H(-\tau) + \sum_{i=-\tau}^{0} H(i)\right)^{C_H-1-\epsilon}\right]$$

$$< 2^{C_H-1-\epsilon}\left(\mathbb{E}\left[q_H(-\tau)^{C_H-1-\epsilon}\right] + \mathbb{E}\left[\left(\sum_{i=-\tau}^{0} H(i)\right)^{C_H-1-\epsilon}\right]\right).$$

The first expectation above is finite since $q_H(-\tau)$ is distributed like Q_H in steady-state. To see that the second expectation is also finite, note that τ is distributed according to the age of the busy period in progress at time 0. Since the light queue busy periods are light-tailed, the age τ is also light-tailed. Thus, $\sum_{i=-\tau}^{0} H(i)$ is a sum of a light-tailed number of heavy-tailed terms, which has tail coefficient equal to C_H by Proposition 5.1. □

5.3 Max-Weight-α Scheduling

In this setting, max-weight-α scheduling works as follows. During each slot t, compare

$$q_L(t)^{\alpha_L} S_L(t) \gtreqless q_H(t)^{\alpha_H} S_H(t),$$

and serve one packet from the queue that wins the comparison. It can be shown using standard Lyapunov arguments that max-weight-α scheduling is throughput optimal for all $\alpha_H > 0$ and $\alpha_L > 0$. That is, it can stably support all arrival rates within the rate region (5.1). This throughput optimality result follows, for example, from [20, Theorem 1].

We recall from the previous chapter that when the queues are always connected, the light queue occupancy distribution is heavy-tailed with a finite tail coefficient under max-weight-α scheduling. In the present setting with time-varying channels, the behavior is more interesting. In fact, q_L turns out to be *light-tailed* for values of λ_L that are smaller than a certain threshold, and heavy-tailed with a finite tail coefficient for larger values of λ_L.

The following result shows that the light queue distribution is light-tailed under any 'reasonable' policy, as long as the rate λ_L is smaller than a threshold value.

Proposition 5.2 *Suppose that $\lambda_L < p_L(1-p_H)$. Then q_L is light-tailed under* any *policy that serves L during its exclusive slots.*

Proof: The proof is straightforward once we note that the exclusive slots of L occur independently during each slot with probability $p_L(1-p_H)$. Indeed, consider the L queue under a policy that serves L *only* during its exclusive slots. Under this policy, the L queue behaves like a G/M/1 queue with light-tailed inputs at rate λ_L, and service rate $p_L(1-p_H)$. It can be shown using standard large deviation arguments

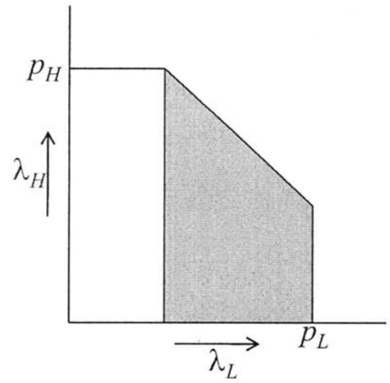


Figure 5-3: Under max-weight-α scheduling, q_L is light-tailed for arrival rates in the unshaded region, and heavy-tailed in the shaded region.

that q_L is light-tailed under the policy that serves L only during its exclusive slots. Therefore, q_L is light-tailed under any policy that serves L during its exclusive slots, and some other slots. □

The above proposition implies that for $\lambda_L < p_L(1 - p_H)$, the light queue distribution is light-tailed under max-weight-α scheduling. The region $\lambda_L < p_L(1 - p_H)$ is shown unshaded in Figure 5-3. Thus, q_L is light-tailed under max-weight-α scheduling for arrival rates in the unshaded region.

In the remainder of this section, we investigate the tail behavior of the light queue under max-weight-α scheduling when the arrival rate is above the threshold, i.e., for $\lambda_L > p_L(1 - p_H)$. In this case, the light queue receives traffic at a higher rate than can be supported by the exclusive slots of L alone. Therefore, the light queue has to compete for service with the heavy queue during the slots that both channels are ON. Since the heavy queue is very large with positive probability, it seems intuitively reasonable that the light queue will suffer from this competition, and also take on a heavy-tailed behavior. This intuition is indeed correct, although proving the result is a non-trivial task, and requires taking an entirely different approach from the methods used in the previous chapter.

We prove that the light queue distribution is heavy-tailed when $\lambda_L > p_L(1 - p_H)$ for all values of the scheduling parameters α_L and α_H. We also obtain the exact tail coefficient of the light queue distribution for 'plain' max-weight scheduling ($\alpha_L/\alpha_H =$

1), and for the regime where the light queue is given more importance ($\alpha_L/\alpha_H > 1$). Whereas we derived *distributional* bounds for the light queue tail in the previous chapter, doing the same for the present case appears difficult. We only derive a moment characterization of the steady-state light queue occupancy, by obtaining the exact tail coefficient.

5.3.1 Max-weight scheduling

Let us first characterize the tail coefficient of the steady-state light queue occupancy under the max-weight policy, which serves the longest connected queue in each slot. Since q_L is light-tailed for $\lambda_L < p_L(1-p_H)$ according to Proposition 5.2, we will focus on the case $\lambda_L > p_L(1-p_H)$. We directly state our main result.

Theorem 5.2 *Suppose that $\lambda_L > p_L(1-p_H)$. Then, under max-weight scheduling, q_L is heavy-tailed with tail coefficient $C_H - 1$.*

In terms of Figure 5-3, the theorem asserts that q_L is heavy-tailed with tail coefficient $C_H - 1$ for all arrival rates in the shaded region. Proving the above result involves showing (i) an upper bound of the form $\mathbb{E}\left[q_H^{C_H-1-\epsilon}\right] < \infty$, and (ii) a lower bound of the form $\mathbb{E}\left[q_H^{C_H-1+\epsilon}\right] = \infty$, for all $\epsilon > 0$. We deal with each of them below.

Upper Bound for max-weight scheduling

Proposition 5.3 *Under max-weight scheduling, we have*

$$\mathbb{E}\left[q_L^{C_H-1-\epsilon}\right] < \infty, \ \forall\, \epsilon > 0.$$

Proof: The result is a straightforward consequence of a theorem in [20]. Indeed, given any $\epsilon > 0$, max-weight scheduling in our context is equivalent to comparing $q_L(t)^{C_H-1-\epsilon}S_L(t)$ versus $q_H(t)^{C_H-1-\epsilon}S_H(t)$, and scheduling the winning queue in each slot. These functions of the queue lengths meet the conditions imposed in [20, Theorem 1], so that the steady-state queue lengths satisfy

$$\mathbb{E}\left[q_L^{C_H-1-\epsilon}\right] < \infty,$$

and

$$\mathbb{E}\left[q_H^{C_H-1-\epsilon}\right] < \infty. \tag{5.5}$$

□

Remark 5.1 *Although we are concerned primarily with the light queue tail behavior, it is interesting that Equation (5.5) gives us the tail coefficient of the heavy queue 'for free'. Indeed, Equation (5.5) asserts that the tail coefficient of the heavy queue under max-weight scheduling is at least $C_H - 1$. However, we know as a consequence of Proposition 4.6 that the tail coefficient of q_H cannot be any larger either. Therefore, q_H is heavy-tailed with tail coefficient equal to $C_H - 1$, under max-weight scheduling.*

Lower Bound for max-weight scheduling

Proposition 5.4 *Suppose that $\lambda_L > p_L(1-p_H)$. Then, under max-weight scheduling, we have*

$$\mathbb{E}\left[q_L^{C_H-1+\epsilon}\right] = \infty, \ \forall\, \epsilon > 0.$$

The proof of this result is quite involved, so we informally describe the idea behind its construction, before proceeding with the formal proof. In our intuitive argument, we will 'show' that

$$\lim_{t\to\infty} \mathbb{E}\left[q_L(t)^{C_H-1+\epsilon}\right] = \infty. \tag{5.6}$$

The above is the limit of the expectation of a sequence of random variables, whereas what we really want in Proposition 5.4 is the expectation of the limiting random variable q_L. Although it is by no means obvious that the limit and the expectation can be interchanged here, we will ignore this as a technical point for the time being.

The main idea behind the proof is to consider the renewal intervals that commence at the beginning of each busy period of the system. Let us define the renewal reward process $R(t) = q_L(t)^{C_H-1+\epsilon}$. By the key renewal theorem [24],

$$\lim_{t\to\infty} \mathbb{E}\left[R(t)\right] = \frac{\mathbb{E}\left[R\right]}{\mathbb{E}\left[T\right]},$$

where $\mathbb{E}[R]$ denotes the expected reward accumulated over a renewal interval, and $\mathbb{E}[T] < \infty$ is the mean renewal interval. It is therefore enough to show that[4]

$$\mathbb{E}\left[\sum_{i=0}^{T} q_L(i)^{C_H-1+\epsilon}\right] = \infty.$$

To see intuitively why the above expectation is infinite, let us condition on the busy period commencing at time 0 with a burst of size b to the heavy queue[5]. After this instant, the heavy queue drains at rate p_H, assuming for the sake of a lower bound that there are no further bursts arriving at H. In the mean time, the light queue receives traffic at rate λ_L, and gets served only during the exclusive slots of L, which occur at rate $p_L(1-p_H)$. With high probability therefore, the light queue will steadily build up at rate $\lambda_L - p_L(1-p_H)$, until it eventually catches up with the draining heavy queue. It can be shown that the light queue will build up to an $O(b)$ level before it catches up with the heavy queue. Further, the light queue occupancy stays at $O(b)$ for a time interval of length $O(b)$. Therefore, with high probability, the reward is at least $O(b^{C_H-1+\epsilon})$ for $O(b)$ time slots. Thus, for some constant K,

$$\mathbb{E}\left[\sum_{i=0}^{T} q_L(i)^{C_H-1+\epsilon}\right] \geq \mathbb{E}\left[Kb \cdot b^{C_H-1+\epsilon}\right] = \mathbb{E}\left[Kb^{C_H+\epsilon}\right] = \infty,$$

where the last expectation is infinite because the initial burst size has tail coefficient equal to C_H.

In words, the light queue not only grows to a level proportionate to the initial burst size, but also stays large for a period of time that is proportional to the burst size. This leads to a light queue distribution that is one order heavier than the burst size distribution. We now present the formal proof of Proposition 5.4 using the above reward theory approach.

Proof: We will first show (5.6) and then use a truncation argument to interchange the limit and the expectation. Consider the renewal process defined by the commence-

[4]Without loss of generality, we have considered a busy period that commences at time 0.
[5]It is easy to show that this event has positive probability.

ment of each busy period of the system. Let T denote a typical renewal interval. We have $\mathbb{E}[T] < \infty$ since the system is stable. Define the reward function

$$R(t) = q_L(t)^{C_H-1+\epsilon}.$$

It is easy to see that $R(t)$ is a legitimate reward function, since $q_L(t)$ is only a function of arrivals and departures during the current renewal interval. As argued above using the key renewal theorem, it is enough to show that the expected reward accumulated over a renewal interval is infinite. Without loss of generality, let us consider a busy period that commences at time 0. We need to show that

$$\mathbb{E}\left[\sum_{i=0}^{T} q_L(i)^{C_H-1+\epsilon}\right] = \infty.$$

The busy period that commences at time 0 can be of three different types. It can commence with (i) a burst arriving to L alone, or (ii) a burst arriving to H alone, or (iii) bursts arriving to both H and L simultaneously. It can be shown that all the three events have positive probabilities[6]. The event that is of interest to us is (ii), i.e., the busy period commencing with a burst at the heavy queue only, so that $q_H(0) > 0$ and $q_L(0) = 0$. Let us denote this event by $\mathcal{E}_H = \{q_H(0) > 0, q_L(0) = 0\}$. We now have the following lower bound

$$\mathbb{E}\left[\sum_{i=0}^{T} q_L(i)^{C_H-1+\epsilon}\right] \geq \mathbb{E}\left[\sum_{i=0}^{T} q_L(i)^{C_H-1+\epsilon};\ \mathcal{E}_H\right] = \mathbb{E}_b\left[\mathbb{E}\left[\sum_{i=0}^{T} q_L(i)^{C_H-1+\epsilon};\ \mathcal{E}_H \middle| q_H(0) = b\right]\right].$$

In the last step above, we have iterated the expectation over the initial burst size b. The inner expectation above is a function of b; let us denote it by

$$g_\epsilon(b) := \mathbb{E}\left[\sum_{i=0}^{T} q_L(i)^{C_H-1+\epsilon};\ \mathcal{E}_H \middle| q_H(0) = b\right].$$

[6]In fact, we can explicitly compute the probability of each of the three events in terms of the probability mass at 0 for $H(\cdot)$ and $L(\cdot)$, but the actual probabilities are not important.

Thus,

$$\mathbb{E}\left[\sum_{i=0}^{T} q_L(i)^{C_H-1+\epsilon}\right] \geq \mathbb{E}_b\left[g_\epsilon(b)\right] \geq \mathbb{E}_b\left[g_\epsilon(b);\ b > b_0\right],\ \forall b_0 \geq 1. \tag{5.7}$$

Since the above bound is true for any b_0, we can make b_0 as large as we want. In particular, we will make the initial burst size large enough to be able to assert that the arrival process to L as well as the channel processes behave 'typically' for time scales of order b.

To be more precise, choose $\delta > 0$ such that $\lambda_L - p_L(1 - p_H) - 3\delta = \eta > 0$, and choose any small $\kappa > 0$. Define

$$\tau_b = \frac{b}{2(p_H + \lambda_L)}.$$

For large enough b_0, and $b > b_0$, it is clear from the (weak) law of large numbers (LLN) that

$$\mathbb{P}\left\{\left|\frac{1}{\tau_b}\sum_{i=0}^{\tau_b} S_H(i) - p_H\right| > \delta\right\} < \kappa.$$

In words, the channel process of H is overwhelmingly likely to behave according to its mean p_H. Now for all $t \leq \tau_b$, the occupancy of H can be lower bounded as

$$q_H(t) \geq b - \sum_{i=0}^{\tau_b} S_H(i) \geq b - (p_H + \delta)\tau_b = b\left(\frac{p_H + 2\lambda_L - \delta}{2(p_H + \lambda_L)}\right), \tag{5.8}$$

with probability greater than $1 - \kappa$. Similarly, the input process to the light queue is also likely to behave according to its mean. That is, for large enough b_0 and $b > b_0$,

$$\mathbb{P}\left\{\left|\frac{1}{\tau_b}\sum_{i=0}^{\tau_b} L(i) - \lambda_L\right| > \delta\right\} < \kappa.$$

Therefore, for all $t \leq \tau_b$, the occupancy of L can be upper bounded as

$$q_L(t) \leq \sum_{i=0}^{\tau_b} L(i) \leq b\left(\frac{\lambda_L + \delta}{2(p_H + \lambda_L)}\right), \tag{5.9}$$

with probability greater than $1-\kappa$. From (5.8), (5.9), and the independence of the processes $L(\cdot)$ and $S_H(\cdot)$, we can conclude that $q_H(t) > q_L(t)$ for all $t \leq \tau_b$, with probability greater than $1-2\kappa$.

Since the light queue remains smaller that the heavy queue for $t \leq \tau_b$ with high probability, it follows that the light queue receives service only during its exclusive slots. More precisely, the departure process from the light queue can be bounded as

$$\sum_{i=1}^{\tau_b} D_L(i) \leq \sum_{i=1}^{\tau_b} S_L(i)(1-S_H(i)),$$

with probability at least $1-2\kappa$. However, the exclusive slots of L are also overwhelmingly likely to behave according to the mean:

$$\mathbb{P}\left\{\left|\frac{1}{\tau_b}\sum_{i=0}^{\tau_b} S_L(i)(1-S_H(i)) - p_L(1-p_H)\right| > 2\delta\right\} < \kappa.$$

Thus,

$$\sum_{i=1}^{\tau_b} D_L(i) \leq \tau_b(p_L(1-p_H)+2\delta),$$

with probability at least $1-3\kappa$. Using the above bound on the departures from L, along with the fact that arrivals to L are also typical, we can lower bound $q_L(\tau_b)$ with high probability. Indeed,

$$q_L(\tau_b) = \sum_{i=1}^{\tau_b} L(i) - \sum_{i=1}^{\tau_b} D_L(i) \geq \tau_b(\lambda_L - \delta) - \tau_b(p_L(1-p_H)+2\delta) = b\left(\frac{\eta}{2(p_H+\lambda_L)}\right),$$

with probability at least $1-3\kappa$. We have thus shown that $q_L(\tau_b)$ is $O(b)$ with as high a probability as we want.

It is now inevitable that $q_L(t)$ stays above $q_L(\tau_b)/2$ for at least another $O(b)$ slots, since at most one packet can be served in a slot. In particular,

$$q_L(t) \geq b\left(\frac{\eta}{4(p_H+\lambda_L)}\right), \ \forall\ \tau_b \leq t \leq \tau_b + \tau_b\left(\frac{\eta}{2}\right),$$

with probability at least $1-3\kappa$.

We can thus lower bound $g_\epsilon(b)$ for large enough b_0 and $b > b_0$ as

$$g_\epsilon(b)1_{\{b>b_0\}} = \mathbb{E}\left[\sum_{i=0}^{T} q_L(i)^{C_H-1+\epsilon};\ \mathcal{E}_H \middle| q_H(0) = b\right] 1_{\{b>b_0\}}$$

$$\geq \left[(1-3\kappa)\sum_{i=\tau_b}^{\tau_b+\tau_b(\frac{\eta}{2})} \left(\frac{\eta b}{4(p_H+\lambda_L)}\right)^{C_H-1+\epsilon}\right] 1_{\{b>b_0\}}$$

$$\geq (1-3\kappa)\frac{\eta\tau_b}{2}\left(\frac{\eta\tau_b}{2}\right)^{C_H-1+\epsilon} 1_{\{b>b_0\}} = Kb^{C_H+\epsilon}1_{\{b>b_0\}}, \tag{5.10}$$

for some constant $K > 0$. Thus, going back to (5.7),

$$\mathbb{E}\left[\sum_{i=0}^{T} q_L(i)^{C_H-1+\epsilon}\right] \geq \mathbb{E}_b\left[g_\epsilon(b);\ b > b_0\right] \geq \mathbb{E}_b\left[Kb^{C_H+\epsilon};\ b > b_0\right] = \infty.$$

The last step is because the initial burst size b has tail coefficient C_H, so that $\mathbb{E}_b\left[b^{C_H+\epsilon}\right] = \mathbb{E}_b\left[b^{C_H+\epsilon};\ b > b_0\right] = \infty$ for all b_0. Therefore, we are done proving (5.6).

Finally, we use a truncation argument to prove that $\mathbb{E}\left[q_L^{C_H-1+\epsilon}\right] = \infty$, where q_L is the steady-state limit of $q_L(t)$.

Truncation argument: Our intention is to show that the limit and the expectation in (5.6) can be interchanged, so that we get the desired moment result for the limiting random variable q_L. Our truncation argument utilizes one of the most fundamental results in integration theory, the Monotone Convergence Theorem (MCT) [8 . Theorem 16.2], as well as a result that affirms the convergence of moments when there is convergence in distribution [8 . Theorem 25.12].

The main idea here is to define a truncated reward function

$$R_M(t) = (M \wedge q_L(t))^{C_H-1+\epsilon},$$

where M is a large integer, and $M \wedge q_L(t) := \min(M, q_L(t))$. There are three steps in our truncation argument.

(i) Tracing all the steps leading up to (5.10) in the proof above, and using the key renewal theorem for the truncated reward function, we can show that

$$w_M := \lim_{t\to\infty} \mathbb{E}\left[R_M(t)\right] \geq \frac{1-3\kappa}{\mathbb{E}\left[T\right]}\mathbb{E}_b\left[\frac{\eta\tau_b}{2}\left(M \wedge \left(\frac{\eta\tau_b}{2}\right)\right)^{C_H-1+\epsilon} 1_{\{b>b_0\}}\right], \quad (5.11)$$

for all M and large enough b_0. The left hand side in the above equation is a function of M, which we have denoted by w_M. The expression inside the expectation on the right is a function of b and M, which we denote by

$$u_M(b) = \frac{\eta\tau_b}{2}\left(M \wedge \left(\frac{\eta\tau_b}{2}\right)\right)^{C_H-1+\epsilon} 1_{\{b>b_0\}}.$$

When viewed as a sequence of functions indexed by M, it is easy to see that $\{u_M(b),\ M > 1\}$ is a monotonically non-decreasing sequence of functions. Furthermore,

$$\lim_{M\to\infty} u_M(b) = Kb^{C_H+\epsilon}1_{\{b>b_0\}},\ \forall\ b, b_0$$

where K is the positive constant in Equation (5.10). Invoking the MCT for the sequence $u_M(b)$, we have

$$\lim_{M\to\infty} \mathbb{E}_b\left[u_M(b)\right] = \mathbb{E}_b\left[\lim_{M\to\infty} u_M(b)\right] = \mathbb{E}_b\left[Kb^{C_H+\epsilon};\ b > b_0\right] = \infty.$$

Next, going back to (5.11) and taking M to infinity, we have

$$\lim_{M\to\infty} w_M = \lim_{M\to\infty}\left(\lim_{t\to\infty} \mathbb{E}\left[R_M(t)\right]\right) \geq \frac{1-3\kappa}{\mathbb{E}\left[T\right]} \lim_{M\to\infty} \mathbb{E}_b\left[u_M(b)\right] = \infty. \quad (5.12)$$

(ii) Recall that the steady-state queue length q_L is defined as the distributional limit of $q_L(t)$, as t becomes large. In other words, viewing $q_L(t)$ as a sequence of random variables indexed by t, we have $q_L(t) \Rightarrow q_L$, where "$\Rightarrow$" denotes convergence in distribution. Next, let us fix M, and view $R_M(t)$ as a sequence of random variables indexed by t. We have

$$R_M(t) \Rightarrow (M \wedge q_L)^{C_H-1+\epsilon}.$$

Theorem 25.12 in [8] asserts that when a sequence of random variables converges in distribution, the corresponding sequence of means also converges to the mean of the limiting random variable, as long as a technical condition called *uniform integrability* is satisfied. Since $R_M(t)$ is bounded above by $M^{C_H-1+\epsilon}$ for all t, uniform integrability is trivially satisfied, and we have

$$\mathbb{E}\left[R_M(t)\right] \to \mathbb{E}\left[(M \wedge q_L)^{C_H-1+\epsilon}\right],$$

for each M as $t \to \infty$. Thus,

$$w_M = \mathbb{E}\left[(M \wedge q_L)^{C_H-1+\epsilon}\right]. \tag{5.13}$$

(iii) Consider finally the term inside the expectation on the right hand side of Equation (5.13). When viewed as a sequence of random variables indexed by M, the term $(M \wedge q_L)^{C_H-1+\epsilon}$ represents a monotonically non-decreasing sequence of random variables. Furthermore,

$$\lim_{M\to\infty} (M \wedge q_L)^{C_H-1+\epsilon} = q_L^{C_H-1+\epsilon}.$$

Thus, another application of the MCT gives

$$\lim_{M\to\infty} \mathbb{E}\left[(M \wedge q_L)^{C_H-1+\epsilon}\right] = \mathbb{E}\left[q_L^{C_H-1+\epsilon}\right]. \tag{5.14}$$

Finally, combining (5.14), (5.13), and (5.12), we get

$$\mathbb{E}\left[q_L^{C_H-1+\epsilon}\right] = \lim_{M\to\infty} \mathbb{E}\left[(M \wedge q_L)^{C_H-1+\epsilon}\right] = \lim_{M\to\infty} w_M = \infty.$$

Proposition 5.4 is now proved. □

5.3.2 Max-weight-α scheduling with $\alpha_L > \alpha_H$

In this subsection, we characterize the exact tail coefficient of the light queue distribution under max-weight-α scheduling, with $\alpha_L > \alpha_H$. We only treat the case $\lambda_L > p_L(1-p_H)$, since q_L is known to be light-tailed otherwise. Our main result for this regime is the following.

Theorem 5.3 *Suppose that $\lambda_L > p_L(1-p_H)$. Then, under max-weight-α scheduling with $\alpha_L > \alpha_H$, q_L is heavy-tailed with tail coefficient*

$$\gamma = \frac{\alpha_L}{\alpha_H}(C_H - 1). \tag{5.15}$$

In terms of Figure 5-3, the above theorem asserts that q_L is heavy-tailed with tail coefficient γ for all arrival rates in the shaded region. As usual, proving this result involves showing (i) an upper bound of the form $\mathbb{E}\left[q_H^{\gamma-\epsilon}\right] < \infty$, and (ii) a lower bound of the form $\mathbb{E}\left[q_H^{\gamma+\epsilon}\right] = \infty$, for all $\epsilon > 0$. We deal with each of them separately.

Upper Bound for max-weight-α scheduling

Proposition 5.5 *Under max-weight-α scheduling, we have*

$$\mathbb{E}\left[q_L^{\gamma-\epsilon}\right] < \infty, \ \forall\ \epsilon > 0.$$

Proof: The result is again a consequence of a theorem in [20]. Indeed, max-weight-α scheduling in our context is equivalent to comparing $q_L(t)^{\beta\alpha_L} S_L(t)$ versus $q_H(t)^{\beta\alpha_H} S_H(t)$, where $\beta > 0$ is arbitrary, and scheduling the winning queue in each slot. In particular, if we choose $\beta = (C_H - 1)/\alpha_H - \epsilon/\alpha_L$, the conditions imposed in [20 . Theorem 1] are satisfied for any $\epsilon > 0$, so that the steady-state queue lengths satisfy

$$\mathbb{E}\left[q_L^{\gamma-\epsilon}\right] < \infty,$$

and

$$\mathbb{E}\left[q_H^{C_H - 1 - \frac{\alpha_H}{\alpha_L}\epsilon}\right] < \infty. \tag{5.16}$$

□

Remark 5.2 *(i) Proposition 5.5 is valid for any parameters α_L and α_H, and not just for $\alpha_L > \alpha_H$.*

(ii) Equation (5.16) and Proposition 4.6 together imply that the tail coefficient of q_H is equal to $C_H - 1$ under max-weight-α scheduling, for any parameters α_L and α_H.

Lower Bound for max-weight-α scheduling with $\alpha_L > \alpha_H$

Proposition 5.6 *Suppose that $\lambda_L > p_L(1-p_H)$. Then, under max-weight-α scheduling with $\alpha_L > \alpha_H$, we have*

$$\mathbb{E}\left[q_L^{\gamma+\epsilon}\right] = \infty, \ \forall\ \epsilon > 0.$$

To prove the above result, we take an approach that is conceptually similar to the proof of Proposition 5.4. We consider the renewal process that commences at the beginning of each busy period of the system, and define the reward process $R_\gamma(t) = q_L(t)^{\gamma+\epsilon}$. We will show that the expected reward accumulated over a renewal interval is infinite. The key renewal theorem would then imply that $\lim_{t\to\infty} \mathbb{E}\left[q_L(t)^{\gamma+\epsilon}\right] = \infty$. Finally, the result we want can be obtained by invoking a truncation argument to interchange the limit and the expectation.

To intuitively see why the expected reward over a renewal interval is finite, let us condition on the busy period commencing with a burst of size b at the heavy queue. Starting at this instant, the light queue will build up at the rate $\lambda_L - p_L(1-p_H)$ with high probability. However, unlike in the case of max-weight scheduling, the light queue only builds up to an $O(b^{\alpha_H/\alpha_L})$ level before it 'catches up' with the heavy queue and wins back the service preference. It can also be shown that the light queue catches up in a time interval of length $O(b^{\alpha_H/\alpha_L})$. It might therefore be tempting to argue that the light queue stays above $O(b^{\alpha_H/\alpha_L})$ for an interval of duration $O(b^{\alpha_H/\alpha_L})$. Although this argument is not incorrect as such, it fails to capture what typically happens in the system. Let us briefly follow through with this argument, and conclude that it does not give us the lower bound we want.

Indeed, following the above argument, the reward is at least $O(b^{(\gamma+\epsilon)\alpha_H/\alpha_L}) = O(b^{C_H-1+\epsilon\alpha_H/\alpha_L})$ for $O(b^{\alpha_H/\alpha_L})$ time slots, so that the expected reward over the renewal interval is lower bounded by

$$\mathbb{E}_b\left[O(b^{\alpha_H/\alpha_L})O(b^{C_H-1+\epsilon\alpha_H/\alpha_L})\right] = \mathbb{E}_b\left[O(b^{C_H-1+\alpha_H/\alpha_L+\epsilon\alpha_H/\alpha_L})\right].$$

However, the right-hand side above turns out to be *finite* for $\alpha_L/\alpha_H > 1$. Therefore, the above simple bound fails to give the result we are after.

The problem with the above argument is that it looks at the time scale at which the light queue *catches up*, whereas the event that decides the tail coefficient happens *after* the light queue catches up. In particular, the light queue catches up relatively quickly, in a time scale of $O(b^{\alpha_H/\alpha_L})$. However, after the light queue catches up with the heavy queue, the two queues drain together, with most of the slots being used to serve the heavy queue. In fact, as we show, before the light queue occupancy can drain by a constant factor after catch-up, the heavy queue drains by $O(b)$. As such, the light queue remains at an $O(b^{\alpha_H/\alpha_L})$ level for $O(b)$ time slots. Therefore, the expected reward can be lower bounded by

$$\mathbb{E}_b\left[O(b)O(b^{C_H-1+\epsilon\alpha_H/\alpha_L})\right] = \mathbb{E}_b\left[O(b^{C_H+\epsilon\alpha_H/\alpha_L})\right] = \infty,$$

which is what we want. In sum, the light queue builds up relatively quickly until catch-up, but takes a long time to drain out after catch-up. We now proceed with the formal proof.

Proof: For the renewal process considered above, $R_\gamma(t) = q_L(t)^{\gamma+\epsilon}$ is easily seen to be a legitimate reward function. Our aim is to show that the expected reward over the renewal interval is infinite, or

$$\mathbb{E}\left[\sum_{i=0}^{T} q_L(i)^{\gamma+\epsilon}\right] = \infty.$$

The key renewal theorem would then imply that $\lim_{t\to\infty} q_L(t)^{\gamma+\epsilon} = \infty$. We can finally appeal to a truncation argument to interchange the limit and the expectation, and obtain the desired result.

Defining $\mathcal{E}_H = \{q_H(0) > 0, q_L(0) = 0\}$, and proceeding as in the proof of Proposition 5.4,

$$\mathbb{E}\left[\sum_{i=0}^{T} q_L(i)^{\gamma+\epsilon}\right] \geq \mathbb{E}\left[\sum_{i=0}^{T} q_L(i)^{\gamma+\epsilon};\ \mathcal{E}_H\right] = \mathbb{E}_b\left[\mathbb{E}\left[\sum_{i=0}^{T} q_L(i)^{\gamma+\epsilon};\ \mathcal{E}_H \middle| q_H(0) = b\right]\right].$$

In the last step above, we have iterated the expectation over the initial burst size b. The inner expectation above is a function of b; let us denote it by

$$g_\gamma(b) := \mathbb{E}\left[\sum_{i=0}^{T} q_L(i)^{\gamma+\epsilon};\ \mathcal{E}_H \middle| q_H(0) = b\right].$$

Thus,

$$\mathbb{E}\left[\sum_{i=0}^{T} q_L(i)^{\gamma+\epsilon}\right] \geq \mathbb{E}_b\left[g_\gamma(b)\right] \geq \mathbb{E}_b\left[g_\gamma(b);\ b > b_0\right],\ \forall b_0 \geq 1. \tag{5.17}$$

Since the above bound is true for any b_0, we can make b_0 as large as we want. We will make b_0 large enough for us to be able to invoke the law of large numbers several times in the rest of the proof.

At this point, we note that for the sake of a lower bound on the expected reward over the renewal interval, we can assume that the heavy queue receives no further arrivals after the initial burst. Under this assumption, we will next show that the light queue catches up with the heavy queue in $O(b^{\alpha_H/\alpha_L})$ time slots. We first need to define what exactly we mean by 'catch-up'.

The catch-up time τ_c is defined as

$$\tau_c = \min\left\{t > 0 \,\middle|\, q_L(t)^{\alpha_L/\alpha_H} \geq q_H(t) > 0\right\} \tag{5.18}$$

In words, the catch-up time is the first time after the arrival of the initial burst for which $q_L(\tau_c)^{\alpha_L/\alpha_H} \geq q_H(\tau_c)$. Note that the catch-up time need not always exist, even if $\mathcal{E}_H$ occurs[7]. However, we show that if the initial burst size is large, the catch-up time exists with high probability.

Indeed, let $b > b_0$ for large enough b_0, and suppose that a catch-up time does not exist. Let us consider the queue lengths after the first $b-1$ time slots, by which time the busy period could not have possibly ended. Since the light queue never catches up, the departure process from the light queue can be upper bounded by the number of exclusive slots. Thus, the light queue occupancy at time $b-1$ can be lower bounded as

$$q_L(b-1) \geq \sum_{i=0}^{b-1} L(i) - S_L(i)(1 - S_H(i)).$$

Since catch-up has not occurred until time $b-1$, it follows that $q_L(b-1)^{\alpha_L/\alpha_H} < q_H(b-1) < b$. Thus, assuming that a catch-up time does not exist implies

$$\left(\sum_{i=0}^{b-1} L(i) - S_L(i)(1 - S_H(i))\right)^{\alpha_L/\alpha_H} < b,$$

or equivalently,

$$\left(\frac{1}{b}\sum_{i=0}^{b-1} L(i) - S_L(i)(1 - S_H(i))\right)^{\alpha_L/\alpha_H} < \frac{b}{b^{\alpha_L/\alpha_H}}.$$

When b is large, the weak LLN implies that the above event has a small probability. This is because the term inside the parentheses on the left is a sample average of random variables with positive mean. Thus, assuming that catch-up does not happen necessitates the occurrence of an event with small probability. This implies that a catch-up time exists for large b with high probability[8].

[7]For example, the initial burst size might be small, and the system might empty again without the light queue ever receiving a single packet during the renewal interval.

[8]In this proof, when we state that an event occurs with high probability for large b, we mean the following: Given any $\kappa > 0$, there exists a large enough b_0 such that for all $b > b_0$, the event in question has probability greater than $1 - \kappa$. In a symmetric fashion, we can define a low probability event for large b as the complement of a high probability event.

Next, we show that τ_c is $O(b^{\alpha_H/\alpha_L})$ with high probability. First, to obtain a lower bound on τ_c, define $\tau_1(b)$ as the unique positive solution to the equation

$$(\lambda_L \tau_1(b))^{\alpha_L} = (b - p_H \tau_1(b))^{\alpha_H}.$$

It is easy to see that $\tau_1(b) = O(b^{\alpha_H/\alpha_L})$. Let us now bound the queue occupancies in the interval $0 \leq t \leq \lfloor \frac{\tau_1(b)}{2} \rfloor$. For the heavy queue,

$$q_H(t) \geq b - \sum_{i=0}^{\lfloor \frac{\tau_1(b)}{2} \rfloor} S_H(i) \geq b - (p_H + \delta) \lfloor \frac{\tau_1(b)}{2} \rfloor$$

with high probability for large b, where $\delta > 0$ can be chosen arbitrarily small. Similarly, for the light queue,

$$q_L(t) \leq \sum_{i=0}^{\lfloor \frac{\tau_1(b)}{2} \rfloor} L(i) \leq \lfloor \frac{\tau_1(b)}{2} \rfloor (\lambda_L + \delta)$$

with high probability for large b. Comparing the last two bounds, it is evident that

$$q_L(t)^{\alpha_L/\alpha_H} > q_H(t), \ 0 \leq t \leq \lfloor \frac{\tau_1(b)}{2} \rfloor,$$

for large b, with high probability. Thus, catch-up has not occurred by time $\lfloor \frac{\tau_1(b)}{2} \rfloor$, so that $\tau_c > \lfloor \frac{\tau_1(b)}{2} \rfloor$ with high probability for large b. Since $\tau_1(b) = O(b^{\alpha_H/\alpha_L})$, it follows that τ_c is at least $O(b^{\alpha_H/\alpha_L})$.

Second, to obtain an upper bound on the catch-up time, define

$$\tau_2(b) = \frac{(2b)^{\alpha_H/\alpha_L}}{\lambda_L - p_L(1 - p_H)}.$$

Suppose that catch-up has not occurred by time $\lceil \tau_2(b) \rceil$. Then, the departures from the light queue only occur during the exclusive slots of L. Thus,

$$q_L(\lceil \tau_2(b) \rceil) \geq \sum_{i=0}^{\lceil \tau_2(b) \rceil} L(i) - S_L(i)(1 - S_H(i)).$$

Since we assumed that catch-up has not occurred by time $\lceil \tau_2(b) \rceil$, we have $q_L(\lceil \tau_2(b) \rceil)^{\alpha_L/\alpha_H} < q_H(\lceil \tau_2(b) \rceil) \leq b$. Therefore,

$$\left(\sum_{i=0}^{\lceil \tau_2(b) \rceil} L(i) - S_L(i)(1 - S_H(i)) \right)^{\alpha_L/\alpha_H} < b,$$

or equivalently,

$$\frac{1}{\lceil \tau_2(b) \rceil} \sum_{i=0}^{\lceil \tau_2(b) \rceil} L(i) - S_L(i)(1 - S_H(i)) < \frac{b^{\alpha_H/\alpha_L}}{\lceil \tau_2(b) \rceil} < \frac{\lambda_L - p_L(1 - p_H)}{2^{\alpha_H/\alpha_L}}$$

By the weak LLN, the above event is of low probability when b is large. Therefore, we conclude that $\tau_c < \lceil \tau_2(b) \rceil$ with high probability when b is large.

We have so far shown that the light queue catches up with the heavy queue in a time scale of $O(b^{\alpha_H/\alpha_L})$ with high probability. Therefore, it easily follows that $q_L(\tau_c) = O(b^{\alpha_H/\alpha_L})$ and $q_H(\tau_c) = b - O(b^{\alpha_H/\alpha_L})$ with high probability. We have now reached the core of the proof where we show that after τ_c, the light queue stays at $O(b^{\alpha_H/\alpha_L})$ for $O(b)$ time slots.

To this end, define σ_c as the first time after τ_c that the light queue occupancy falls below $(q_H(\tau_c)/2)^{\alpha_H/\alpha_L}$. That is

$$\sigma_c = \min \left\{ t > \tau_c \,\middle|\, q_L(t) < \left(\frac{q_H(\tau_c)}{2} \right)^{\alpha_H/\alpha_L} \right\}.$$

It is clear that σ_c is well defined when τ_c exists, since the system eventually empties.

With the intention of necessitating a low probability event, let us assume that

$$q_H(t) \geq \frac{3q_H(\tau_c)}{4}, \text{ for all } t \in [\tau_c, \sigma_c]. \tag{5.19}$$

Next, define

$$\omega_c = \max \left\{ \tau_c \leq t < \sigma_c \,\middle|\, q_L(t) \geq \left(\frac{3q_H(\tau_c)}{4} \right)^{\alpha_H/\alpha_L} \right\}.$$

In words, ω_c is the last time before σ_c that the light queue occupancy exceeds $(3q_H(\tau_c)/4)^{\alpha_H/\alpha_L}$. Now, by the definition of ω_c and the assumption made in (5.19), it is clear that $q_L(t)^{\alpha_L/\alpha_H} < q_H(t)$ for $\omega_c < t \leq \sigma_c$. Thus, the departures that occur from the light queue during the interval $\omega_c < t \leq \sigma_c$ must necessarily occur during the exclusive slots of L. Therefore,

$$q_L(\sigma_c) = q_L(\omega_c) + \sum_{i=\omega_c+1}^{\sigma_c} L(i) - S_L(i)(1 - S_H(i)),$$

or equivalently,

$$\frac{1}{\sigma_c - \omega_c} \sum_{i=\omega_c+1}^{\sigma_c} L(i) - S_L(i)(1 - S_H(i)) = \frac{q_L(\sigma_c) - q_L(\omega_c)}{\sigma_c - \omega_c}.$$

This necessarily implies

$$\frac{1}{\sigma_c - \omega_c} \sum_{i=\omega_c+1}^{\sigma_c} L(i) - S_L(i)(1 - S_H(i)) < 0. \tag{5.20}$$

From the definition of σ_c and ω_c, it is clear that

$$\sigma_c - \omega_c > q_L(\omega_c) - q_L(\sigma_c) = \left(3^{\alpha_H/\alpha_L} - 2^{\alpha_H/\alpha_L}\right) \left(\frac{q_H(\tau_c)}{4}\right)^{\alpha_H/\alpha_L},$$

so that $\sigma_c - \omega_c$ is at least $O(b^{\alpha_H/\alpha_L})$. Therefore, by the weak LLN, the event in (5.20) is a low probability event for large b.

What we have shown now is that the assumption in (5.19) implies the occurrence of a low probability event for large b. Therefore, the assumption (5.19) should be false with high probability when b is large. In other words, with high probability, there exists $t \in [\tau_c, \sigma_c]$ for which $q_H(t) < \frac{3q_H(\tau_c)}{4}$. In particular, this implies that $\sigma_c - \tau_c > \frac{q_H(\tau_c)}{4}$, with high probability for large b.

Next, since $q_H(\tau_c) = b - O(b^{\alpha_H/\alpha_L})$ with high probability, we have $q_H(\tau_c) > b/2$ for large enough b. Thus, $\sigma_c - \tau_c > b/8$, with high probability, and for $\tau_c \leq t < \sigma_c$, the light queue occupancy is lower bounded by

$$q_L(t) \geq \left(\frac{q_H(\tau_c)}{2}\right)^{\alpha_H/\alpha_L} > \left(\frac{b}{4}\right)^{\alpha_H/\alpha_L},$$

also with high probability. We have thus shown that after catch-up, the light queue occupancy stays at $O(b^{\alpha_H/\alpha_L})$ for $O(b)$ slots, with high probability.

We can now return to (5.17) to finish the sequence of inequalities. In particular, let us choose b_0 large enough such that for $b > b_0$, the intersection of all the high probability events above has probability at least $1-\kappa$, for some $\kappa > 0$. Then,

$$\mathbb{E}\left[\sum_{i=0}^{T} q_L(i)^{\gamma+\epsilon}\right] \geq \mathbb{E}_b\left[g_\gamma(b);\ b > b_0\right] \geq (1-\kappa)\mathbb{E}_b\left[\frac{b}{8}\cdot\left(\frac{b}{4}\right)^{\alpha_H/\alpha_L(\gamma+\epsilon)};\ b > b_0\right]$$

$$= K_1\mathbb{E}_b\left[b\cdot b^{C_H-1+\epsilon\alpha_H/\alpha_L};\ b > b_0\right] = \infty,$$

since the burst size b has tail coefficient C_H. The key renewal theorem would then imply that

$$\lim_{t\to\infty}\mathbb{E}\left[q_L(t)^{\gamma+\epsilon}\right] = \infty,\ \forall\,\epsilon > 0.$$

We can finally invoke a truncation argument similar to the one in Proposition 5.4 to interchange the limit and the expectation. Thus, for the steady-state occupancy q_L, we have $\mathbb{E}\left[q_L^{\gamma+\epsilon}\right] = \infty,\ \forall\,\epsilon > 0$.

□

5.3.3 Max-weight-α scheduling with $\alpha_L < \alpha_H$

We finally consider the case $\alpha_L < \alpha_H$ under max-weight-α scheduling, and study the asymptotic behavior of q_L. Recall that max-weight-α scheduling with $\alpha_L < \alpha_H$ corresponds to giving the heavy queue preference over the light queue. In this regime, we show that q_L is heavy-tailed with a finite tail coefficient, for arrival rates in the shaded region of Figure 5-3. However, we are unable to determine the exact tail coefficient of q_L for some arrival rate pairs in this regime.

Our first result for this case is an upper bound on the tail coefficient of q_L. Intuitively, we would expect that the asymptotic behavior of q_L in this regime cannot be better than it is under max-weight scheduling. In other words, the tail coefficient of q_L in this regime cannot be larger than $C_H - 1$. This intuition is indeed correct.

Proposition 5.7 *Suppose that $\lambda_L > p_L(1-p_H)$. Then, under max-weight-α scheduling with $\alpha_L < \alpha_H$, the tail coefficient of q_L is at most $C_H - 1$.*

Proof: Follows similarly to the proof of Proposition 5.4. Specifically, conditioning on an initial burst of size b arriving to the heavy queue, it can be shown that with high probability, q_L will be $O(b)$ in size for at least $O(b)$ time slots. $\square$

Next, to obtain a lower bound on the tail coefficient of q_L, recall that Proposition 5.5 holds for the present regime as well. Thus, γ (defined in (5.15)) is a lower bound[9] on the tail coefficient of q_L. In sum, we have shown that for $\lambda_L > p_L(1-p_H)$, the light queue occupancy distribution is heavy-tailed, with a tail coefficient that lies in the interval $[\gamma, C_H - 1]$.

It turns out that we can obtain the exact tail coefficient of q_L for arrival rates in a *subset* of the shaded region in Figure 5-3. Specifically, consider the region represented by $p_L(1-p_H) < \lambda_L < p_L(1-\lambda_H)$. In Figure 5-4, this region is shown in gray. It can be shown that all arrival rates in the region shaded gray can be stabilized under priority for H. Furthermore, under priority for H, it can be shown that q_L is heavy-tailed with tail coefficient equal to $C_H - 1$, when $p_L(1-p_H) < \lambda_L < p_L(1-\lambda_H)$. This can be done using a proof strategy similar to the one used in Theorem 5.1(ii).

Since the tail of q_L under max-weight-α scheduling with any parameters is no worse than under priority for H, we can conclude that the tail coefficient of q_L is at least $C_H - 1$ when $p_L(1-p_H) < \lambda_L < p_L(1-\lambda_H)$. Combining this with Proposition 5.7, we conclude that the tail coefficient q_L is *equal* to $C_H - 1$, when the arrival rate pair lies in the gray region of Figure 5-4.

Proposition 5.8 *Suppose that $p_L(1-p_H) < \lambda_L < p_L(1-\lambda_H)$. Then, under max-weight-α scheduling with $\alpha_L < \alpha_H$, the tail coefficient of q_L is equal to $C_H - 1$.*

[9]Note that γ is *smaller* than $C_H - 1$ in this regime.

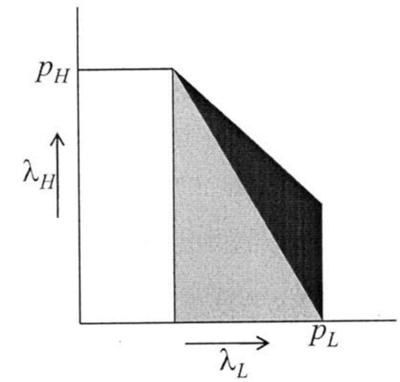


Figure 5-4: Under max-weight-α scheduling with $\alpha_L < \alpha_H$, q_L is light-tailed for arrival rates in the unshaded region, and heavy-tailed with tail coefficient equal to $C_H - 1$ in for arrival rates in the gray region. For arrival rates in the region colored black, the tail coefficient lies in $[\gamma, C_H - 1]$.

The region shaded black in Figure 5-4 ($\lambda_L > p_L(1-\lambda_H)$) corresponds to the arrival rates for which priority for H is *not* stabilizing[10]. Under max-weight-α scheduling with $\alpha_L < \alpha_H$, we are unable to determine the exact tail coefficient of q_L for arrival rates in the black region of Figure 5-4. However, we have shown earlier that the tail coefficient lies in the interval $[\gamma, C_H - 1]$. We *conjecture that the tail coefficient of* q_L *equals* γ, for arrival rates in the black region.

5.3.4 Section summary

We showed the following results in this section.

1. The light queue distribution is light-tailed under max-weight-α scheduling, when $\lambda_L < p_L(1 - p_H)$. This is true for all scheduling parameters.

2. When $\lambda_L > p_L(1 - p_H)$, the light queue distribution is inevitably heavy-tailed under max-weight-α scheduling. In particular, under max-weight scheduling ($\alpha_L = \alpha_H$), the tail coefficient of q_L is equal to $C_H - 1$. For $\alpha_L > \alpha_H$, the tail coefficient of q_L is $\gamma = (C_H - 1)\alpha_L/\alpha_H$. Finally, for $\alpha_L < \alpha_H$, the tail coefficient of q_L lies in $[\gamma, C_H - 1]$.

[10]This case is symmetric to the case in Theorem 5.1(i).

3. For all values of the scheduling parameters, q_H is heavy-tailed with tail coefficient equal to $C_H - 1$.

Remark 5.3 *The case $p_H = p_L = 1$ corresponds to a system where the queues are always connected to the server, i.e., the system considered in the previous chapter. Applying Theorem 5.2 and Proposition 5.8 to this special case implies that the tail coefficient of q_L is $C_H - 1$, for $\alpha_L \leq \alpha_H$. Similarly, applying Theorem 5.3 for the special case $p_H = p_L = 1$ implies that the tail coefficient of q_L equals γ, for $\alpha_L > \alpha_H$. What we have shown above is that Theorem 4.9 is a special case of the results in this section. Furthermore, since we have not made any regularity assumptions on the heavy-tailed input in this chapter, it follows that Theorem 4.9 holds even when $H(\cdot)$ is not order-regular. In other words, the assumption $H(\cdot) \in \mathcal{OR}$ is needed only to derive the distributional lower bounds in Theorem 4.7, and not for the tail coefficient result in Theorem 4.9.*

5.4 Log-Max-Weight Scheduling

In the previous section we studied the performance of the max-weight-α policy when the queues are connected to the server through time-varying channels. Although max-weight-α scheduling has the desirable property of throughput optimality, our analysis showed that the light queue occupancy distribution is heavy-tailed, except when λ_L is small enough to be supported by the exclusive slots of L.

In this section, we study the performance of log-max-weight scheduling policy. In the current setting with time-varying channels, the LMW policy works as follows. During each time slot t, the log-max-weight policy compares

$$q_L(t)S_L(t) \gtreqless \log(1 + q_H(t))S_H(t),$$

and serves one packet from the queue that wins the comparison.

We show that LMW scheduling has desirable performance on both fronts, namely throughput optimality, and the asymptotic behavior of the light queue occupancy.

The LMW policy can be shown to be throughput optimal, in the sense that it can stably support any arrival rate pair within the rate region specified in (5.1). This throughput optimality result can be shown directly using a Lyapunov argument, or viewed as a special case of [20 . Theorem 1].

In terms of the asymptotic performance, we show that the LMW policy guarantees that the light queue occupancy distribution is light-tailed, for all arrival rates that can be stabilized by priority for L. For arrival rates that are not stabilizable under priority for L, the LMW policy will still stabilize the system, although we are not able to guarantee that q_L is light-tailed for these arrival rates.

In particular, our analysis implies that the light queue occupancy distribution is light-tailed under LMW scheduling, for arrival rates in a larger region than under the max-weight-α policy. Let us state this precisely in the following theorem.

Theorem 5.4 *Under LMW scheduling, q_L is light-tailed if at least one of the following conditions hold:*

(i) $\lambda_L < p_L(1 - p_H)$, *or*

(ii) $\lambda_H < p_H(1 - \lambda_L)$.

Note that for $\lambda_L < p_L(1 - p_H)$, q_L is easily seen to be light-tailed under LMW scheduling, since the arrival rate is small enough to be supported by the exclusive slots of L. The second condition in Theorem 5.4 states that for all arrival rates that can be stabilized under priority for L (i.e., the trapezoidal region in Figure 5-2), q_L is light-tailed under LMW scheduling.

The union of the two regions in which q_L is light-tailed according to Theorem 5.4 is shown unshaded in Figure 5-5. As can be seen, the unshaded region occupies most of the rate region, except for the shaded triangle. For arrival rates in the shaded triangle, the LMW policy still stabilizes the system, unlike priority for L.

Proof: Statement (i) of the theorem is a direct consequence of Proposition 5.2. We now prove statement (ii) by explicitly deriving an exponentially decaying upper bound on $\mathbb{P}\{q_L > b\}$. Since the proof of (ii) is quite similar to the proof of Theorem 4.10, we skip some details.

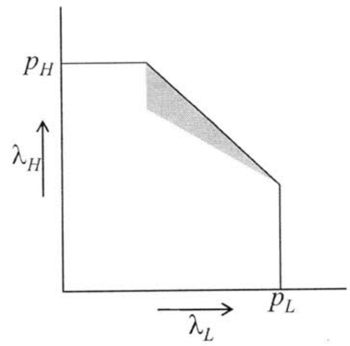


Figure 5-5: Under LMW scheduling, q_L is light-tailed in the unshaded region.

For $\lambda_H < p_H(1-\lambda_L)$, we show that

$$\liminf_{b\to\infty} -\frac{1}{b}\log \mathbb{P}\{q_L \geq b\} \geq \min(I_L,\ C_H - 1), \tag{5.21}$$

where I_L is defined in (5.2). Assume without loss of generality that the event $q_L > b$ occurs at time zero, with the system running since time $-\infty$. Define $-\tau \leq 0$, as the most recent time during the current busy period that the relation $\log(1+q_H(\cdot)) > q_L(\cdot)$ holds[11].

For any fixed $\delta > 0$, we have the following equality.

$$\begin{aligned}
\mathbb{P}\{q_L(0) \geq b\} &= \underbrace{\mathbb{P}\{q_L(0) \geq b;\ \log(1+q_H(-\tau)) < \delta b\}}_{(a)} \\
&+ \underbrace{\mathbb{P}\{q_L(0) \geq b;\ (1-\delta)b \geq \log(1+q_H(-\tau)) \geq \delta b\}}_{(b)} \\
&+ \underbrace{\mathbb{P}\{q_L(0) \geq b;\ \log(1+q_H(-\tau)) > (1-\delta)b\}}_{(c)}
\end{aligned} \tag{5.22}$$

We will next upper bound each of the above three terms on the right.

(a) Since $\log(1+q_H(-\tau)) < \delta b$, it follows from the definition of τ that $q_L(-\tau) < \delta b$. Next, during the interval $[-\tau+1, 0]$, we have $\log(1+q_H(\cdot)) \leq q_L(\cdot)$, so that the light queue receives service whenever $S_L(\cdot) = 1$. In other words, the light queue effectively has priority in the interval $[-\tau+1, 0]$, but still grows from less than

[11]If no such τ exists during the current busy period, take $q_H(-\tau) = 0$ for interpreting (5.22).

δb to overflow at time 0. A classical large deviation bound can be derived for this event. Indeed, it can be shown that for every $\epsilon > 0$ and for large enough b,

$$\mathbb{P}\left\{q_L(0) \geq b;\ \log(1 + q_H(-\tau)) < \delta b\right\} < \kappa_1(\epsilon) e^{-b(1-\delta)(I_L - \epsilon)}, \tag{5.23}$$

for some constant $\kappa_1(\epsilon)$.

(c) Let us deal with the term (c) before (b). For $\lambda_H < p_H(1 - \lambda_L)$, Theorem 5.1 asserts that q_H has tail coefficient equal to $C_H - 1$ under priority for L. Therefore, under LMW scheduling, it holds that $\mathbb{E}\left[q_H^{C_H - 1 - \epsilon}\right] < \infty$. Applying Markov inequality with the above expectation, we have under LMW scheduling for every $\epsilon > 0$,

$$\mathbb{P}\left\{q_H > M\right\} < \kappa_2 M^{-(C_H - 1 - \epsilon)},\ M > 0.$$

Therefore, for every $\epsilon > 0$,

$$\mathbb{P}\left\{q_L(0) \geq b;\ \log(1 + q_H(-\tau)) > (1 - \delta)b\right\} < \mathbb{P}\left\{\log(1 + q_H(-\tau)) > (1 - \delta)b\right\}$$

$$< \kappa_2(\epsilon) \exp\left(-(1 - \delta)b(C_H - 1 - \epsilon)\right). \tag{5.24}$$

(b) In this case, let $\eta = \log(1 + q_H(-\tau))$, so that $\delta b \leq \eta \leq (1 - \delta)b$. Proceeding similarly to the steps leading to Equation (4.42) in the proof of Theorem 4.10, we can show that for every $\epsilon > 0$ and some $\kappa(\epsilon)$,

$$\mathbb{P}\left\{q_L(0) \geq b;\ (1 - \delta)b \geq \log(1 + q_H(-\tau)) \geq \delta b\right\} < \sum_{\xi = \delta b}^{(1-\delta)b} \kappa(\epsilon) e^{-(\xi(C_H - 1 - \epsilon) + (I_L - \epsilon)(b - \xi))}, \tag{5.25}$$

where ξ is a 'dummy' variable that runs over all possible values of τ. Let us now distinguish two cases:

- $C_H - 1 > I_L$: In this case, we can bound the probability in (5.25) as

$$\mathbb{P}\left\{q_L(0) \geq b;\ (1 - \delta)b \geq \log(1 + q_H(-\tau)) \geq \delta b\right\} < \kappa_3 e^{-b[(1-\delta)(I_L - \epsilon)]},\ \forall \epsilon > 0, \tag{5.26}$$

where κ_3 is some constant.

– $C_H - 1 \leq I_L$: In this case,

$$\mathbb{P}\left\{q_L(0) \geq b;\ (1-\delta)b \geq \log(1+q_H(-\tau)) \geq \delta b\right\} < \kappa_4 e^{-b[(1-\delta)(C_H-1-\epsilon)]},\ \forall \epsilon > 0, \tag{5.27}$$

where κ_4 is some constant.

Finally, we put together the bounds on terms (a), (b) and (c) into Equation (5.22).

1. If $C_H - 1 > I_L$, we get from (5.23), (5.24), and (5.26),

$$\mathbb{P}\left\{q_L(0) \geq b\right\} < e^{-b(1-\delta)(I_L-\epsilon)}\left[\kappa_1 + \kappa_2 e^{-((1-\delta)b(C_H-1-I_L))} + \kappa_3\right], \tag{5.28}$$

from which it is immediate that

$$\liminf_{b\to\infty} -\frac{1}{b}\log \mathbb{P}\left\{q_L(0) \geq b\right\} \geq (1-\delta)(I_L - \epsilon).$$

Since the above is true for each ϵ and δ, we get

$$\liminf_{b\to\infty} -\frac{1}{b}\log \mathbb{P}\left\{q_L(0) \geq b\right\} \geq I_L. \tag{5.29}$$

2. If $C_H - 1 \leq I_L$, we get from (5.23), (5.24), and (5.27),

$$\mathbb{P}\left\{q_L(0) \geq b\right\} < e^{-b(1-\delta)(C_H-1-\epsilon)}\left[\kappa_1 e^{-((1-\delta)b(I_L-C_H+1))} + \kappa_2 + \kappa_4\right], \tag{5.30}$$

from which it is immediate that

$$\liminf_{b\to\infty} -\frac{1}{b}\log \mathbb{P}\left\{q_L(0) \geq b\right\} \geq (1-\delta)(C_H - 1 - \epsilon).$$

Since the above is true for each ϵ and δ, we get

$$\liminf_{b\to\infty} -\frac{1}{b}\log \mathbb{P}\left\{q_L(0) \geq b\right\} \geq C_H - 1. \tag{5.31}$$

Equation (5.21) now follows from (5.29) and (5.31), and we are done. □

We have shown that under LMW scheduling, the light queue occupancy distribution is necessarily light-tailed for arrival rates in the unshaded region of Figure 5-5. In other words, the LMW policy guarantees light-tailed asymptotics for the light queue distribution, whenever the arrival rate pair is stably supportable by priority for L. For arrival rates in the shaded triangle, we know that the LMW policy is stabilizing, but priority for L is not. However, we are unable to determine whether or not the light queue distribution is light-tailed, for arrival rates in the shaded triangle of Figure 5-5.

5.5 Chapter Summary

In this chapter, we extended the results obtained in the previous chapter to a setting with randomly time-varying links. We conclude by briefly summarizing the main results of this chapter.

1. When the light-tailed traffic is given full priority, the heavy queue can become unstable, even if the arrival rates are within the rate region of the system. However, *if* the system is stable under priority for the light queue, the asymptotic behavior of both queues is as good as it can possibly be.

2. Under max-weight-α scheduling, the light queue occupancy distribution is light-tailed when $\lambda_L < p_L(1-p_H)$, and heavy-tailed with a finite tail coefficient when $\lambda_L > p_L(1-p_H)$.

3. Under LMW scheduling, the light queue occupancy distribution is light-tailed for all arrival rates that are stably supportable under priority for the light queue. Additionally, the LMW policy is also throughput optimal, and can stabilize traffic rates that are not supportable under priority scheduling.

Chapter 6

Concluding Remarks

In this thesis, we studied the interplay between the asymptotic behavior of queue occupancy distributions, queue length information, and traffic statistics in the operation of network control policies. In addition to enhancing the conceptual understanding of the role of control information in network control, our study also has practical implications on buffer provisioning, estimating buffer overflow events, and providing worst-case delay guarantees. Furthermore, our results show that *queue length based scheduling policies should be designed based on the statistical properties of competing traffic flows*, in order to simultaneously ensure throughput optimality and good asymptotic behavior of the queue backlogs.

In Chapter 2, we studied the role of queue length information in the congestion control of a single-server queue. Our results indicate that arbitrarily infrequent queue length information is sufficient to ensure optimal asymptotic decay for the buffer overflow probability, as long as the control information is accurately received. However, if the control messages are subject to errors, the congestion probability can increase drastically, even if the control messages are transmitted often.

In Chapter 3, we studied the scheduling problem in a system of parallel queues sharing a server, when the system is fed by a statistically homogeneous traffic pattern. We showed that the queue length based max-weight scheduling outperforms some well known queue-blind policies in terms of the buffer overflow probability. This is because the max-weight policy tends to 'balance' the queue occupancies by serving the longest queue at each instant, while the queue-blind policies cannot discern large build-up in one of the queues. We also showed that the large deviation exponent of the overflow probability can be preserved under arbitrarily infrequent queue length updates. This result, as well as the one in Chapter 2, suggests that the large deviation exponent of buffer overflow is not susceptible to change under infrequent queue length updates, much like the stability of the queueing network.

In Chapters 4 and 5, we obtained an exact asymptotic characterization of the queue length distributions, when a mix of heavy-tailed and light-tailed traffic flows feeds a system of parallel queues. In stark contrast to the results in Chapter 3, we showed that max-weight scheduling leads to a poor asymptotic performance for the light-tailed traffic. This is because the max-weight policy forces the light-tailed traffic to compete for service with the highly bursty heavy-tailed traffic. In other words, the tendency of the max-weight policy to 'balance' the queue occupancies actually becomes a curse of sorts, when scheduling between traffic flows of greatly different burstiness levels.

We also analyzed a log-max-weight scheduling policy, which effectively smothers the impact of the heavy-tailed flow's burstiness, by giving significantly higher scheduling preference to the light-tailed flow. We showed that the log-max-weight policy leads to good asymptotic performance for the light-tailed traffic, while also preserving throughput optimality.

Overall, our study of queue length asymptotics under various scheduling policies indicates that the statistical nature of the traffic flows should be taken into account in the design of queue length based scheduling mechanisms.

Appendix A

Throughput Maximization over Uncertain Wireless Channels – A State Action Frequency Approach

A.1 Introduction

In this chapter, we consider the scheduling problem in a wireless uplink or downlink system, when there is no explicit instantaneous Channel State Information (CSI) available to the scheduler. The lack of CSI may arise in practice due to several reasons. For example, the control overheads, as well as the delay and energy costs associated with channel probing, might make instantaneous CSI too costly or impractical to obtain.

Our system consists of N wireless links, which are modeled as N parallel queues that are fed by stochastic traffic. Due to the shared wireless medium, only a single queue can be chosen at each time slot for transmitting its data. The channel quality (or state) of each wireless link is time-varying, evolving as an independent ON/OFF Markov chain. A given transmission is successful only if the underlying channel is currently in the ON state.

Our basic assumption in this chapter is that the scheduler cannot observe the current state of *any* of the wireless links. Nonetheless, when the scheduler serves one of the queues in a given time slot t, there is an ACK-feedback mechanism which acknowledges whether the transmission was successful or not, thereby revealing the channel state *a posteriori*. Since the channels are correlated across time by the Markovian assumption, this *a posteriori* CSI can be used for predicting the channel state of the chosen queue in future time slots. We emphasize that the ACK mechanism is the only means by which CSI is made available to the scheduler.

The *capacity region* (or the rate region) of the system described above, is the set of all arrival-rate vectors that are stably-supportable by *some* scheduling policy. Our aim is to characterize the capacity region of the system, and to design a throughput optimal scheduling policy.

The general problem of scheduling parallel queues with time-varying connectivity has been widely studied for almost two decades. The seminal paper of Tassiulas and Ephremides [64] considered the case where both channel states and queue lengths are fully available to the scheduler. It was shown in [64] that the *max-weight algorithm*, which serves the longest connected queue, is throughput optimal. Notably, the algorithm stabilizes all rates in the capacity region, without requiring any *a priori* knowledge on the arrival rates.

Following this paper, several variants of imperfect and delayed CSI scenarios have been considered in the literature, see, e.g., [28,47,74,75] and references therein. However, our scheduling problem fundamentally differs from the models considered in these references. Specifically, no explicit CSI is ever made available to the scheduler, and acquiring channel state information is a *part of the scheduling decision* made at each time instant. This adds significant difficulties to the scheduling problem.

Two recent papers consider the scheduling problem where the CSI is obtained through an acknowledgement process, as in our model. In [1], the authors consider the objective of maximizing the *sum-rate* of the system, under the assumption that the queues are *fully-backlogged* (i.e., there is always data to send in each queue). It is shown that a simple *myopic policy* is sum-rate optimal. The suggested policy keeps scheduling the channel that is being served as long as it remains ON, and switches to the least recently served channel when the current channel goes OFF.

In [36], the authors propose a randomized round-robin scheduling policy for the system, which is inspired by the myopic sensing results in [1]. Their policy is shown to stabilize arrivals that lie within an inner-bound to the rate region. However, their policy is *not* throughput optimal, and their method cannot be used to characterize the capacity region.

Here, we propose a throughput optimal scheduling policy for the system. In particular, the policy we propose can stabilize arrival rates that lie arbitrarily close to the capacity region boundary, with a corresponding tradeoff in the computational complexity. We also provide a characterization of the capacity region boundary, as the limit of a sequence of LP solutions.

The scheduling problem we consider is related to the celebrated restless bandits problem [71], which is known to be computationally difficult. In fact, every point on the boundary of the capacity region can be implicitly expressed as the optimal solution to a restless bandits problem. Such a solution involves solving an MDP with a countably infinite state-space. Since obtaining this solution may be computationally and analytically prohibitive, we approximate the original MDP by a finite-state MDP with a 'tunable' number of states. We then employ a linear programming approach to solve the resulting finite-state MDP [50].

We prove that the solution to the LP approximates the boundary of the capacity region arbitrarily closely, where the accuracy of the approximation improves with the number of states in the underlying finite MDP. Thus, there is a tradeoff between the accuracy of the approximation, and the dimensionality of the LP.

Next, we combine the LP solution with a queue length based scheduling mechanism that operates over long time-frames to obtain a dynamic scheduling policy for the system. Our main result establishes that this 'frame-based' policy is *throughput optimal*, i.e., can stably support all arrival rates in the interior of the capacity region. Our proof of throughput optimality combines tools from Markov decision theory within a Lyapunov stability framework.

The remainder of this chapter is organized as follows. The model is presented

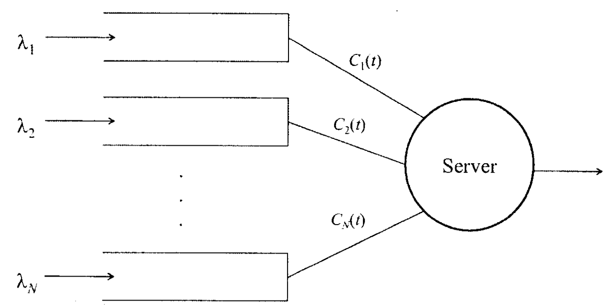


Figure A-1: A system of parallel queues served by a single server. The channels connecting the queues to the server are randomly time-varying.

in Section A.2. In Section A.3, we formulate a linear program which leads to the characterization of the capacity region. In Section A.4, we suggest the frame-based policy, which we prove to be throughput optimal. We conclude in Section A.5.

A.2 System Description

The network model. We model the wireless system as consisting of N parallel queues (see Figure A-1). Time is slotted ($t = 1, 2, \ldots$). Packets arrive to each queue $i \in \{1, 2, \ldots, N\}$ according to an independent stochastic process with rate λ_i. We assume that the arrival processes are independent of each other, and independent and identically distributed (i.i.d.) from slot-to-slot. We further assume that the number of arrivals in a slot to each of the queues has a finite variance.

Due to the shared wireless medium, only a single transmission is allowed at a given time. In our queuing model, this is equivalent to having the queues connected to a single server, which is capable of serving only a single packet per slot, belonging to one of the queues. Each queue is connected to the server by an ON/OFF channel, which models time-varying channel quality of the underlying wireless link. If a particular channel is OFF and the queue is chosen by the scheduler, the transmission of the packet would fail, and the packet has to be retransmitted. If it is ON and chosen by the scheduler, a single packet is properly transmitted, and an ACK is received by the scheduler.

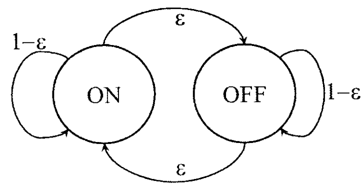


Figure A-2: The Markov chain governing the time evolution of each of the channels state $C_i(t)$.

We denote the channel state of the i-th link at time t by $C_i(t) \in \{ON, OFF\}$, $i = 1, \ldots, N$. We assume that the states of different channels are statistically independent of each other. The time evolution of each of the channels is given by a two state ON/OFF Markov chain (see Figure A-2). Although our methodology allows for different Markov chains for different channels, we shall assume for ease of exposition that the Markov chains are identically distributed across users as shown in Figure A-2. We further assume that $\epsilon < 0.5$, so that each channel is *positively correlated* in time.

Information structure. At each time t, we assume that the scheduler knows the current queue lengths $Q_i(t)$ prior to making the scheduling decision. Yet, *no* information about the current channel conditions is made available to the scheduler. Only *after* scheduling a particular queue, does the scheduler get to know whether the transmission succeeded or not, by virtue of the ACK-mechanism. The scheduler thus has access to the entire history of transmission successes and failures. However, due to the Markovian nature of the channels, it is sufficient to record *how long ago* each channel was served, and the *state of the channel* (ON/OFF) when it was last served. In addition to the above, the scheduler also knows precisely the statistical properties of each of the channels (i.e., the Markov chain of Figure A-2).

Scheduling objective. Given the above information structure, our objective is to design a scheduling policy that can support the largest possible set of input rates. More precisely, an arrival rate vector $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_N)$ is said to be *supportable*, if there exists some scheduling policy under which the queue lengths are finite (almost surely). The *capacity region* Γ of the system is the closure of all supportable rate vectors. A policy is said to be *throughput optimal* if it can support all arrival rates in the interior of Γ.

A.3 Optimal Policies for a Fully Backlogged System

In the interest of simplicity of notation and exposition, we restrict attention to the case of $N = 2$ queues in the rest of the chapter, although our methodology extends naturally to more queues. In this section, we assume that the queues are fully backlogged, i.e., the queues never empty.

Since the queues are assumed to be infinitely backlogged in this section, the state of the system is completely specified by the state of each channel the last time it was served, and how long ago each channel was served. In a system with two fully backlogged queues, the *information state* during slot t has the form $\mathbf{s}(t) = [k_1(t), b_1(t), k_2(t), b_2(t)]$, where $k_i(t)$ is the number of slots since the queue i was served, and $b_i(t) \in \{0, 1\}$ is the state of the channel the last time it was observed.[1] Since the channels are Markovian, $\mathbf{s}(t)$ is a sufficient statistic for the fully backlogged system. Note that $\min(k_1(t), k_2(t)) = 1,\ \forall\ t$, and $\max(k_1(t), k_2(t)) \geq 2\ \forall\ t$. Let $\mathcal{S}$ denote the (countably infinite) set of all possible states $\mathbf{s}(t)$.

Denote the l step transition probabilities of the channel Markov chain in Figure A-2 by $p_{11}^{(l)}$, $p_{01}^{(l)}$, $p_{10}^{(l)}$, and $p_{00}^{(l)}$. It can be shown by explicit computation that for $l \geq 1$,

$$p_{01}^{(l)} = p_{10}^{(l)} = \frac{1 - (1 - 2\epsilon)^l}{2}, \quad p_{11}^{(l)} = p_{00}^{(l)} = \frac{1 + (1 - 2\epsilon)^l}{2}.$$

Next, define the *belief* vector corresponding to state $\mathbf{s} \in \mathcal{S}$ as $[\omega_1(\mathbf{s}), \omega_2(\mathbf{s})]$, where $\omega_i(\mathbf{s}),\ i = 1, 2$ is the conditional probability that the channel i is ON. For example, if $\mathbf{s} = [1, ON, 3, OFF]$, the corresponding belief vector is $[1 - \epsilon, p_{01}^{(3)}]$. It can be shown that the belief vector has a one-to-one mapping to the information state, and is

[1] Throughout, 0 is used interchangeably to denote the channel state OFF, and 1 is used to denote ON.

therefore also a sufficient statistic for the fully backlogged problem.

In each slot, there are two possible actions, $a \in \{1, 2\}$, corresponding to serving one of the two queues. Given a state and an action at a particular time, the belief for the next slot is updated according to the following equation.

$$\omega_i(t+1) = \begin{cases} (1-\epsilon)\omega_i(t) + \epsilon(1-\omega_i(t)), \text{ if } a(t) \neq i, \\ 1-\epsilon, \text{ if } a(t) = i, \ C_{a(t)}(t) = 1, \\ \epsilon, \text{ if } a(t) = i, \ C_{a(t)}(t) = 0, \end{cases}$$

where we have abused notation to write $\omega_i(t) = \omega_i(\mathbf{s}(t))$.

A *policy* for the fully backlogged system is a rule that associates an action $a(t) \in \{1, 2\}$, to the state $\mathbf{s}(t)$ for each t. A *deterministic stationary* policy is a map from $\mathcal{S}$ to $\{1, 2\}$, whereas a *randomized stationary* policy picks an action given the state according to a fixed distribution $\mathbb{P}\{a|\mathbf{s}(\cdot)\}$.

Suppose that a unit reward is accrued from each of the two channels, every time a packet is successfully transmitted on that channel, i.e., when the server is assigned to a particular channel and the channel is ON. Given a state $\mathbf{s}(t)$ at a particular time, and an action $a(t)$, the probability that a unit reward is accrued in that time slot is simply equal to the belief of the channel that was chosen. We are interested in the long term time average rate achieved on each of the channels under a given policy. From the viewpoint of the reward defined above, the average rate translates to the infinite horizon time average reward obtained on each channel under a given policy.

We say that rate pair (λ_1, λ_2) is *achievable* in the fully backlogged system, if there exists *some* policy for which the infinite horizon time average reward vector equals (λ_1, λ_2). The closure of the set of all achievable rate pairs is called the *rate region* Λ of the fully backlogged system. It should be evident that a rate pair that is not achievable in the fully backlogged system, cannot be supportable in the dynamic system with finite queues. Thus, the capacity region Γ of the queueing system is contained in the rate region Λ of the fully backlogged system. In fact, we show in Section A.4 that the two rate regions have the same interior, by deriving a queue length based policy for the original system that can stabilize any arrival rate in the interior of Λ. We now proceed to obtain an implicit characterization of the rate region boundary.

A.3.1 MDP formulation and state action frequencies

Let us consider a Markov decision process (MDP) formulation on the belief space for characterizing the rate region boundary.

It is easy to show that the rate region Λ is convex. Indeed, given two points in the rate region, each attainable by some policy, we can obtain any convex combination of the rate points by time-sharing the policies over sufficiently long intervals. Further, the rate region is also closed by definition. Therefore, any point on its boundary maximizes a weighted sum- rate expression. That is, if (r_1^*, r_2^*) is a rate pair on the boundary of Λ, then

$$(r_1^*, r_2^*) = \arg\max_{(\lambda_1,\lambda_2)\in\Lambda} w_1\lambda_1 + w_2\lambda_2 \tag{A.1}$$

for some weight vector $\mathbf{w} = [w_1,\ w_2]$, with $w_1 + w_2 = 1$. The following proposition shows that if the rate pair (λ_1, λ_2) is in Λ, then there must necessarily exist *state action frequencies* that satisfy a set of balance equations.

Proposition A.1 *Let $(\lambda_1, \lambda_2) \in \Lambda$. Then, for each state $\boldsymbol{s} \in \mathcal{S}$ and action $a \in \{1, 2\}$, there exists state action frequencies $x(\boldsymbol{s}; a)$, that satisfy*

$$0 \leq x(\boldsymbol{s}; a) \leq 1, \tag{A.2}$$

the balance equations (A.3)-(A.6),

$$\begin{aligned} x([1, b_1, k, b_2]; 1) + x([1, b_1, k, b_2]; 2) = \\ x([1, b_1, k-1, b_2]; 1)(1-\epsilon) + x([1, 1-b_1, k-1, b_2]; 1)\epsilon, k > 2, \end{aligned} \tag{A.3}$$

$$x([1,b_1,2,b_2];1)+x([1,b_1,2,b_2];2) = \sum_{l\geq 2} x([l,b_1,1,b_2];1)p_{11}^{(l)} + x([l,1-b_1,1,b_2];1)p_{01}^{(l)}, \quad \text{(A.4)}$$

$$x([k,b_1,1,b_2];1)+x([k,b_1,1,b_2];2) = x([k-1,b_1,1,b_2];2)(1-\epsilon) + x([k-1,b_1,1,1-b_2];2)\epsilon, k>2, \quad \text{(A.5)}$$

$$x([2,b_1,1,b_2];1)+x([2,b_1,1,b_2];2) = \sum_{l\geq 2} x([1,b_1,l,b_2];2)p_{11}^{(l)} + x([1,b_1,l,1-b_2];2)p_{01}^{(l)}, \quad \text{(A.6)}$$

the normalization condition

$$\sum_{s\in\mathcal{S}} x(s;1)+x(s;2) = 1, \quad \text{(A.7)}$$

and the rate constraints

$$\lambda_i \leq \sum_{s\in\mathcal{S}} x(s;i)\omega_i(s), \ i=1,2. \quad \text{(A.8)}$$

Proof: Follows from the linear programming formulation of countable MDPs [2]. □

Intuitively, a set of state action frequencies corresponds to a stationary randomized policy such that $x(\mathrm{s};a)$ equals the steady-state probability that in a given time slot, the state is s and the action is a. Further, conditioned on being in state s, the action a is chosen with probability $\frac{x(\mathrm{s};a)}{\mathbb{P}\{\mathrm{s}\}}$, where $\mathbb{P}\{\mathrm{s}\} = x(\mathrm{s};1)+x(\mathrm{s};2)$. (If $\mathbb{P}\{\mathrm{s}\}=0$, the policy prescribes actions arbitrarily).

Let us now provide an intuitive explanation of the balance equations. Equations (A.3)-(A.6) simply equate the steady-state probability of being in a particular state, to the total probability of entering that state from all possible states. For example, the left side of (A.3) equals the steady-state probability of being in the state $[1,b_1,k,b_2]$, $k>2$, while the right side equals the total probability of getting to the above state from other states, and similarly for the other balance equations. Equation (A.7) equates the total steady-state probability to unity. Finally, in Equation (A.8), the term $x(\mathbf{s};i)\omega_i(\mathbf{s})$ equals the probability that the state is $\mathbf{s}$, the action i is chosen, *and* the transmission succeeds. Thus, the right-side side of (A.8) equals the total expected rate on channel i.

We now return to the characterization of the rate region boundary. In light of Proposition A.1, Equation (A.1) can be rewritten as follows.

Problem INFINITE($\mathbf{w}$):

$$(r_1^*, r_2^*) = \arg\max_{(\lambda_1,\lambda_2)} w_1\lambda_1 + w_2\lambda_2 \tag{A.9}$$

subject to (A.2)-(A.8).

Since the number of state-space of the MDP is countably infinite, the optimization in (A.9) involves an infinite number of variables. In order to make this problem tractable, we now introduce an LP approximation.

A.3.2 LP approximation using a finite MDP

In this section, we introduce an MDP with a finite state space, which as we show, approximates the original MDP arbitrarily closely. The state action frequencies corresponding to the finite MDP approximation can then be solved as a linear program.

First note that the belief of a channel that has not been observed for a long time increases monotonically toward the steady state value of 0.5 if it was OFF the last time it was scheduled. Similarly, the belief decreases monotonically to 0.5 if the channel was ON the last time it was scheduled. The key idea now is to construct a finite MDP whose states are the same as the original MDP, with the exception that the belief of a channel that remains unobserved for a long time is clamped to the steady state ON probability, 0.5. Specifically, when a channel has not been scheduled for τ or more time slots, its observation history is entirely forgotten, and the belief on it is assumed to be 0.5. The action space and the reward structure are exactly as before. We show that this truncated finite MDP approximates the original MDP better and better, as τ gets large.

Let us now specify the states and state action frequencies for this finite MDP. There are $4(\tau-2)$ states of the form $[1, b_1, k_2, b_2]$, $2 \leq k_2 \leq \tau-1$, $b_1, b_2 \in \{ON, OFF\}$ that correspond to the first channel being scheduled in the previous slot, and the second channel being scheduled less that τ time slots ago. In a symmetric fashion, there are $4(\tau-2)$ states of the form $[k_1, b_1, 1, b_2]$, $2 \leq k_1 \leq \tau-1$, $b_1, b_2 \in \{ON, OFF\}$ that correspond to the second channel being scheduled in the previous slot. Finally, there are 4 states $[1, b_1, \phi, \phi]$, $b_1 \in \{ON, OFF\}$ and $[\phi, \phi, 1, b_2]$, $b_2 \in \{ON, OFF\}$ in which one of the channels has not been seen for at least τ slots, and its belief reset to 0.5. Let us denote by $\hat{\mathcal{S}}$ the above set of states for the finite MDP, and let $\hat{x}(\mathbf{s}; a)$, $\mathbf{s} \in \hat{\mathcal{S}}$, $a \in \{1, 2\}$ denote the state action frequencies for the finite MDP. These state action frequencies satisfy

$$0 \leq \hat{x}(\mathbf{s}; a) \leq 1, \tag{A.10}$$

$$\sum_{s \in \hat{\mathcal{S}}} \hat{x}(\mathbf{s}; 1) + \hat{x}(\mathbf{s}; 2) = 1, \tag{A.11}$$

$$\hat{\lambda}_i \leq \sum_{\mathbf{s} \in \hat{\mathcal{S}}} \hat{x}(\mathbf{s}; i) \omega_i(\mathbf{s}), \ i = 1, 2, \tag{A.12}$$

and a set of balance equations analogous to (A.3)-(A.6).

For a fixed $\mathbf{w}$ and τ, let us now consider the following LP.

Problem FINITE(τ,$\mathbf{w}$):

$$(\hat{r}_1, \hat{r}_2) = \arg\max_{(\hat{\lambda}_1, \hat{\lambda}_2)} w_1 \hat{\lambda}_1 + w_2 \hat{\lambda}_2 \tag{A.13}$$

subject to (A.10)-(A.12) and balance equations.

The main result of this section shows that the *solution to this LP approximates* *the boundary point* specified by the problem INFINITE($\mathbf{w}$) for every $\mathbf{w}$, for large τ.

Proposition A.2 [2] *For a given $\boldsymbol{w}$ with $w_1+w_2=1$, and τ, let $\hat{\boldsymbol{\eta}}(\tau,\boldsymbol{w})$ denote the solution to the problem FINITE(τ,$\boldsymbol{w}$), and let $\boldsymbol{r}^*(\boldsymbol{w})$ denote the solution to INFINITE($\boldsymbol{w}$). Then, $\hat{\boldsymbol{\eta}}(\tau,\boldsymbol{w})$ converges* uniformly *to $\boldsymbol{r}^*(\boldsymbol{w})$, as $\tau \to \infty$. In other words, given any $\kappa > 0$ and any $\boldsymbol{w}$, there exists $\tau_0 > 0$ that depends on κ but not on $\boldsymbol{w}$, such that for all $\tau > \tau_0$, we have*

$$|\hat{\boldsymbol{\eta}}(\tau,\boldsymbol{w}) - \boldsymbol{r}^*(\boldsymbol{w})| < \kappa.$$

Proof: The convergence of $\hat{\boldsymbol{\eta}}(\tau,\mathbf{w})$ to $\mathbf{r}^*(\mathbf{w})$ for a fixed $\mathbf{w}$ follows from the classical work of Whitt [69, 70]. The difficulty is in proving that the convergence is uniform across all $\mathbf{w}$. Without loss of generality, we assume that $\mathbf{w} = (x, 1-x)$ for $x \in [0,1]$. The main observation here is that the function $f_\tau : [0,1] \to \mathbb{R}$ that takes an element x and returns $\hat{\boldsymbol{\eta}}(\tau,(x,1-x))$ is a *convex* function for every τ since it is the solution of a parametric linear program [7]. It also follows that $f_\tau(0)$ and $f_\tau(1)$ is the same for all τ (since these are the cases where only one of the channels matters). Let's denote the function $f_\infty(x) : [0,1] \to \mathbb{R}$ to be a function that takes x and returns $\mathbf{r}^*((x,1-x))$. Take a finite grid of points on $[0,1]$ denoted by G. We have convergence for every $g \in G$ of $f_\tau(g)$ to $f_\infty(g)$ [69, 70]. Since these are all convex functions, a uniform convergence for all values of x follows; see [52]. □

We next show a result that asserts that using the state action frequencies obtained from the finite MDP in the backlogged system entails only a negligible sub-optimality, when τ is large. The finite MDP solution is applied to the backlogged system as follows. If the state in the backlogged system is such that both channels were served no more than τ time slots ago, then we schedule according to the state action frequencies of that particular state in the finite MDP. On the other hand, if one of the channels was served more than τ time slots ago, the finite MDP would not *have* a corresponding state and state action frequencies. In such a case, we schedule according to the state action frequencies of one of the four states in the finite MDP in which the belief is clamped to the steady-state value. For example, if the system state is

[2]The proofs of Propositions A.2 and A.3 were supplied by Prof. Shie Mannor. They are included here for completeness.

$[1, b_1, k_2, b_2]$, with $k_2 > \tau$, we schedule according to the state action frequencies of the state $[1, b_1, \phi, \phi]$ in the finite MDP, and so on.

Proposition A.3 *Suppose that the optimal state action frequencies obtained by solving the problem FINITE($\tau, \boldsymbol{w}$) are used to perform scheduling in a fully backlogged system, as detailed above. Let $\hat{\boldsymbol{r}}(\tau, \boldsymbol{w})$ denote the average reward vector so obtained. Then for every $\boldsymbol{w}$ with $w_1 + w_2 = 1$, we have that $\hat{\boldsymbol{r}}(\tau, \boldsymbol{w})$ converges* uniformly *to the optimal reward $\boldsymbol{r}^*(\boldsymbol{w})$, as $\tau \to \infty$.*

Proof outline: Proposition A.2 asserts that $\hat{\boldsymbol{\eta}}(\tau, \mathbf{w})$ converges to $\mathbf{r}^*(\mathbf{w})$ uniformly. It therefore suffices to prove that $\hat{\mathbf{r}}(\tau, \mathbf{w})$ converges uniformly to $\hat{\boldsymbol{\eta}}(\tau, \mathbf{w})$. Indeed, we will prove a stronger result claiming that this holds for *any* stationary policy for the finite MDP, and not just for optimal policies under some $\mathbf{w}$.

Suppose that we are given a stationary policy π defined on the truncated MDP with a 'memory' of τ, and let π_∞ be the extension of π to the infinite state space as discussed above. To complete the proof, we imitate the methodology of [69, 70]. While the details are lengthy and technical, the main observation that is required to obtain uniform convergence is that the reward that is obtained in the finite MDP for π is obtained in the *same states* as it is obtained for π_∞ for the infinite MDP (and this is true for all $\mathbf{w}$). The difference between the finite and infinite MDPs in terms of transitions is only in the transitions out of the four additional states $[1, b_1, \phi, \phi]$, $b_1 \in \{ON, OFF\}$ and $[\phi, \phi, 1, b_2]$, $b_2 \in \{ON, OFF\}$ that have the same policy as the appropriate states where one of the queues was not visited for τ steps (by construction). As long as the transition is within these four states or within the other states that are identical for the truncated and infinite MDPs, the rewards are the same. Once there is a transition out of these states, the conditional transition probability becomes becomes close as τ increases (i.e., exiting each of the four states has a conditional probability that becomes closer to the conditional probability when exiting the matching states in the infinite MDP). The fact that the transitions are becoming closer makes the value of the policies similar uniformly over all policies. □

We pause momentarily to emphasize the subtle difference between Propositions

A.2 and A.3. Proposition A.2 asserts that optimal reward obtained from the finite MDP is close to the optimal reward of the infinite MDP. In this case, the optimal solution to the finite MDP is applied to the finite state-space. On the other hand, in Proposition A.3, the optimal policy obtained from the finite MDP is used on the original *infinite* state-space, and the ensuing reward is shown to be close to the optimal reward. From a practical perspective, Propositions A.2 is useful in obtaining a characterization of the rate region, while Proposition A.3 plays a key role in the throughput optimality proof of the frame-based policy.

A.3.3 An outer bound

We now derive an outer bound to the rate region Λ, using 'genie-aided' channel information. Although the bound is *not* used in deriving our optimal policy, it is of interest to compare the outer bound we obtain to existing bounds in the literature.

Consider a fictitious, fully backlogged system in which the channel processes follow the same sample paths as in the original system. However, after a channel is served in a particular time slot, a genie reveals the states of all the channels in the system. Therefore, at the beginning of a time slot in the fictitious system, the scheduler has access to *all* the channel states in the previous slot, and not just the channel that was served. Clearly, the rate region boundary for the genie-aided system is an outer bound to the rate region of the original system.

Let us explicitly compute the above outer bound for our two queue system. Indeed, there are only four possibilities for the channel states in the previous slots, $\{ON, ON\}$, $\{OFF, ON\}$, $\{ON, OFF\}$, and $\{OFF, OFF\}$. Furthermore, each of the states above occurs with probability $1/4$ in steady-state. Using these facts, we can explicitly obtain the rate region for the genie-aided fictitious system.

Indeed, let Λ_{00} be the convex hull of the vectors $(\epsilon, 0)$ and $(0, \epsilon)$. Intuitively, Λ_{00} is the set of all rate vectors that are achievable exclusively in the time slots with $\{OFF, OFF\}$ as the channel states in the previous slot. Similarly, let $\Lambda_{01} = \mathcal{C}\left\{(\epsilon, 0),\ (0, 1-\epsilon)\right\}$, $\Lambda_{10} = \mathcal{C}\left\{(1-\epsilon, 0),\ (0, \epsilon)\right\}$, and $\Lambda_{11} = \mathcal{C}\left\{(1-\epsilon, 0),\ (0, 1-\epsilon)\right\}$, where $\mathcal{C}$ stands for convex hull. Then, the rate region of the fictitious system is given by

$$\overline{\Lambda} = \left\{ \boldsymbol{\lambda} \geq 0 \,\middle|\, \boldsymbol{\lambda} = \frac{1}{4}\left(\boldsymbol{\lambda}_{00} + \boldsymbol{\lambda}_{01} + \boldsymbol{\lambda}_{10} + \boldsymbol{\lambda}_{11}\right) \right\},$$

where $\boldsymbol{\lambda}_{00} \in \Lambda_{00}$, etc. The boundary of the region $\overline{\Lambda}$ can be explicitly characterized in terms of ϵ:

$$\overline{\Lambda} = \left\{ (\lambda_1, \lambda_2) \,\middle|\, \begin{array}{c} \epsilon\lambda_1 + (1-\epsilon)\lambda_2 \leq (1-\epsilon)/2; \\ (1-\epsilon)\lambda_1 + \epsilon\lambda_2 \leq (1-\epsilon)/2; \\ \lambda_1 + \lambda_2 \leq 3/4 - \epsilon/2 \end{array} \right\}. \tag{A.14}$$

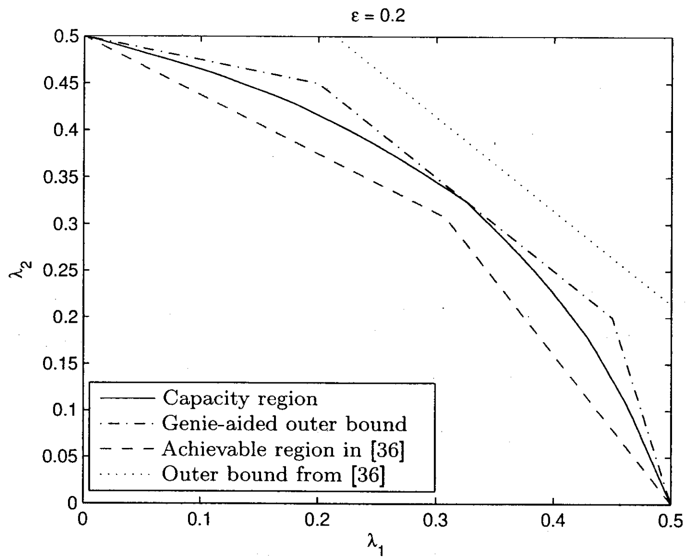
A desirable property of the genie-aided outer bound is that it is *tight at the symmetric rate point* for the two queue system with symmetric channels. In other words, the symmetric rate point on the outer-bound, namely $(3/8-\epsilon/4, 3/8-\epsilon/4)$, is achievable in the original backlogged system. To see this, consider the *myopic* policy which stays with a queue as long as its channel remains ON, and switches to the other queue when it goes OFF. The sum throughput of this policy can be shown to be $3/4-\epsilon/2$, by direct computation (see [1] for example). Since this sum throughput is equally shared between the two channels, it follows that the symmetric rate point on the outer bound is achievable.

Interestingly, the above argument also constitutes a simple optimality proof of myopic sensing for the case of two symmetric channels. This is a special case of the optimality result derived in [1] for any N.

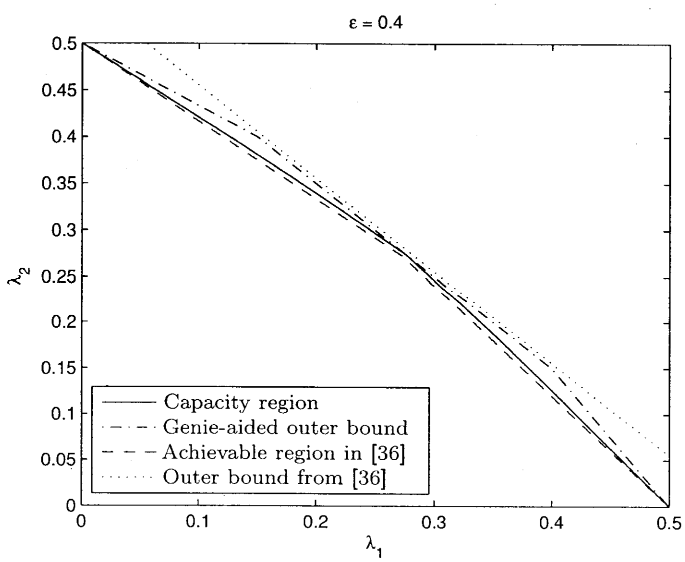
A.3.4 Numerical examples

In this section, we use the finite LP approximation obtained in Section A.3.2 to numerically compute and plot the capacity region for a two queue system. Specifically, we use the solution to the problem FINITE(τ,$\mathbf{w}$) with large enough τ, which, according to Proposition A.2, uniformly approximates the rate region boundary for all $\mathbf{w}$. We also plot the genie-aided outer bound obtained above, and compare our rate region and outer bound to the inner and outer bounds derived in [36].

Figure A-3 shows the numerically obtained rate region, the genie-aided outer bound, and the inner and outer bounds derived in [36] for our symmetric two queue



(a)



(b)

Figure A-3: The rate region, our outer bound, and the inner and outer bounds derived in [36], for (a) $\epsilon = 0.2$, and (b) $\epsilon = 0.4$.

system. Figure A-3(a) is for the case $\epsilon = 0.2$ (higher correlation in time), while Figure A-3(b) is for $\epsilon = 0.4$ (lower correlation in time). The rate region, shown with a dark solid line, was obtained by solving the LP approximation FINITE$(\tau, \mathbf{w})$ for all weight vectors, and large enough τ. We observed that $\tau \approx 30$ and $\tau \approx 10$ were sufficient for the cases $\epsilon = 0.2$, and $\epsilon = 0.4$, respectively. The dash-dot curve in the figure is the genie-aided outer bound, given by (A.14). The achievable region of the randomized round-robin policy proposed in [36], is shown by a dashed line. Finally, the outer most region in the figure is the outer bound derived in [36].

The tightness of the genie-aided outer bound at the symmetric rate point is evident from Figure A-3, since the rate region boundary touches the outer bound. We also observe that the genie-aided bound is uniformly better than the outer bound in [36].

A.4 A Throughput Optimal Frame-Based Policy

In this section, we return to the original problem, with finite queues and stochastic arrivals. We propose a throughput optimal queue length based policy that operates over long 'frames.'

In our frame-based policy, the time axis is divided into frames consisting of T slots each, and the queue lengths are updated at the beginning of each frame. Given the queue length vector $\mathbf{Q}(kT)$ at the beginning of each frame, the idea is to maximize a weighted sum rate quantity over the frame, where the *weight vector is the queue length vector* for that frame. The weighted rate maximization is, in turn, performed approximately by solving the finite MDP. Intuitively, the above procedure has the net effect of performing max-weight scheduling over each time-frame, where MDP techniques are employed to compute each of the 'optimal schedules.'

FRAME-BASED POLICY:

(i) At the beginning of time frame k, update the queue length vector $\mathbf{Q}(kT)$.

(ii) Compute the normalized queue length vector $\tilde{\mathbf{Q}}(kT)$, whose entries sum to 1.

(iii) Solve the problem FINITE($\tau, \tilde{\mathbf{Q}}(kT)$) and obtain the state action frequencies $\hat{x}(\mathbf{s}, a),\ \mathbf{s} \in \hat{\mathcal{S}}, a \in \{1, 2\}$.

(iv) Schedule according to the state action frequencies obtained in the previous step during each slot in the frame, even if it means scheduling an empty queue.

Our main result in the chapter is the throughput optimality of the frame-based policy, for large enough values of T and τ. Specifically, our frame-based policy can stabilize all arrival rates within a δ-stripped region of Λ, for any $\delta > 0$. As we shall see, a small δ could require large values of T and τ, which increases the dimensionality of the LP (depends on τ) as well as the average delay (depends on T). Thus our policy offers a tradeoff between computational complexity and delay on the one hand, and better throughput on the other. Our main theorem is stated below. Note also that our policy requires queue length information only at the beginning of each time frame.

Theorem A.1 *Given any $\delta > 0$, there exist large enough τ and T such that the frame-based policy stabilizes all arrival rates in the δ-stripped rate region $\Lambda - \delta\mathbf{1}$.*

Proof: Let us define the Lyapunov function

$$L(\mathbf{Q}(t)) = \sum_i Q_i^2(t)$$

and the corresponding drift over a frame

$$\Delta_T(kT) = \mathbb{E}\left[L(\mathbf{Q}((k+1)T)) - L(\mathbf{Q}(kT))\Big|\mathbf{Q}(kT)\right].$$

Using the queue evolution relation

$$Q_i(t+1) = Q_i(t) + A_i(t) - D_i(t),$$

where $A_i(\cdot)$ and $D_i(\cdot)$ respectively denote the arrival and departure processes from the ith queue, we can easily arrive at the following bound on the T step Lyapunov drift.

$$\Delta_T(kT)/T \leq B + \sum_i Q_i(kT)\lambda_i - \sum_i Q_i(kT)\mathbb{E}\left[\frac{1}{T}\sum_{\sigma=0}^{T-1} D_i(kT+\sigma)|\mathbf{Q}(kT)\right], \quad \text{(A.15)}$$

where B is a constant. The above expression holds for any policy. In order to particularize the bound, we need to bound the departure term in (A.15) for the frame-based policy.

We now pause to make some definitions. Define

$$W_T(kT) = \sum_i Q_i(kT)\left(\frac{1}{T}\sum_{\sigma=0}^{T-1} D_i(kT+\sigma)\right).$$

Next, define $\hat{D}_i(t)$ as the departure process from a fully backlogged system when our frame-based policy is used on it. That is, $\hat{D}_i(t)$ is the same as the departure process $D_i(t)$ in the original system, except there are no lost departures due to empty queues. Define

$$\hat{D}_T(kT) = \sum_i Q_i(kT)\left(\frac{1}{T}\sum_{\sigma=0}^{T-1} \hat{D}_i(kT+\sigma)\right).$$

Given any weight vector $\mathbf{w}$, let $\mathbf{r}^*(\mathbf{w})$ denote the rate vector on the boundary of the original capacity region Λ that maximizes the $\mathbf{w}$-weighted sum of rates. Define

$$R^*(kT) = \sum_i Q_i(kT)r_i^*(\tilde{\mathbf{Q}}(kT)).$$

Next, for the same weight vector $\mathbf{w}$, let $\overline{\mathbf{r}}(\mathbf{w},\tau)$ denote the weighted rate maximizing point in the truncated MDP. Define

$$\overline{R}(kT) = \sum_i Q_i(kT)\overline{r}_i(\tilde{\mathbf{Q}}(kT),\tau).$$

Observe that $\mathbf{r}^*(\cdot)$ and $\overline{\mathbf{r}}(\cdot,\tau)$ are deterministic vectors, once the weight vector and the truncation threshold τ are fixed. On the other hand, $\hat{D}_i(\cdot)$ is a random variable, which is determined by the channel outcomes and the outcomes of the randomized actions dictated by the state action frequencies.

Our next result states that the mixing of the finite MDP is exponential in time, so that the empirical average reward obtained over a long frame of length T is very close to the infinite horizon reward of the finite MDP.

Lemma A.1 *Regardless of the starting state, we have,* $\forall\ \kappa > 0$

$$\mathbb{P}\left\{\left\|\frac{1}{T}\sum_{\sigma=0}^{T-1}\hat{\boldsymbol{D}}(kT+\sigma)-\overline{\boldsymbol{r}}(\tilde{\boldsymbol{Q}}(kT),\tau)\right\|>\frac{\kappa}{2}\right\}<ce^{-\eta(\kappa)T}.$$

Proof: Follows from [39]. □

Let us now get back to the drift expression (A.15), and rewrite it as

$$\Delta_T(kT)/T \leq B+\sum_i Q_i(kT)\lambda_i-\mathbb{E}\left[W_T(kT)|\mathbf{Q}(kT)\right]$$

$$= B+\sum_i Q_i(kT)\left[\lambda_i-r_i^*(\tilde{\mathbf{Q}}(kT))\right]+\mathbb{E}\left[R^*(kT)-W_T(kT)|\mathbf{Q}(kT)\right]$$

$$\leq B+\sum_i Q_i(kT)\left[\lambda_i-r_i^*(\tilde{\mathbf{Q}}(kT))\right]+\mathbb{E}\left[|R^*(kT)-\overline{R}(kT)|\ \Big|\mathbf{Q}(kT)\right]$$

$$+\ \mathbb{E}\left[|\overline{R}(kT)-W_T(kT)|\ \Big|\mathbf{Q}(kT)\right]. \tag{A.16}$$

We now bound the two expectation terms in the RHS of (A.16). First, we have

$$\mathbb{E}\left[|R^*(kT)-\overline{R}(kT)|\ \Big|\mathbf{Q}(kT)\right]=\mathbb{E}\left[|<\mathbf{Q}(kT),\mathbf{r}^*(\tilde{\mathbf{Q}}(kT))-\overline{\mathbf{r}}(\tilde{\mathbf{Q}}(kT),\tau)>|\ \Big|\mathbf{Q}(kT)\right]$$

$$\leq\ \mathbb{E}\left[\|\mathbf{Q}(kT)\|\ \|\mathbf{r}^*(\tilde{\mathbf{Q}}(kT))-\overline{\mathbf{r}}(\tilde{\mathbf{Q}}(kT),\tau)\|\ \Big|\mathbf{Q}(kT)\right] \tag{A.17}$$

$$\leq \kappa\|\mathbf{Q}(kT)\|. \tag{A.18}$$

where (A.17) follows from Schwarz Inequality, and (A.18) is due to Proposition A.3 for large enough τ.

We next bound the second expectation term in (A.16) as shown in (A.19).

$$\mathbb{E}\left[|\overline{R}(kT)-W_T(kT)|\ \Big|\mathbf{Q}(kT)\right]\leq\mathbb{E}\left[|\overline{R}(kT)-W_T(kT)|\ \Big|\mathbf{Q}(kT),\ |\overline{R}(kT)-W_T(kT)|\leq\kappa\|\mathbf{Q}(kT)\|\right]$$

$$\cdot\mathbb{P}\left\{|\overline{R}(kT)-W_T(kT)|\leq\kappa\|\mathbf{Q}(kT)\|\ \Big|\mathbf{Q}(kT)\right\}+$$

$$\mathbb{E}\left[|\overline{R}(kT) - W_T(kT)| \,\Big|\mathbf{Q}(kT),\ |\overline{R}(kT) - W_T(kT)| > \kappa\|\mathbf{Q}(kT)\|\right]$$
$$\cdot\mathbb{P}\left\{|\overline{R}(kT) - W_T(kT)| > \kappa\|\mathbf{Q}(kT)\| \,\Big|\mathbf{Q}(kT)\right\}$$
$$\leq \kappa\|\mathbf{Q}(kT)\| + \left(\sum_i Q_i(kT)\right)\mathbb{P}\left\{|\overline{R}(kT) - W_T(kT)| > \kappa\|\mathbf{Q}(kT)\| \,\Big|\mathbf{Q}(kT)\right\}. \tag{A.19}$$

Let us now bound the final probability term in (A.19):

$$\mathbb{P}\left\{|\overline{R}(kT) - W_T(kT)| > \kappa\|\mathbf{Q}(kT)\| \,\Big|\mathbf{Q}(kT)\right\} \leq$$
$$\mathbb{P}\left\{|\overline{R}(kT) - \hat{D}_T(kT)| > \frac{\kappa}{2}\|\mathbf{Q}(kT)\| \,\Big|\mathbf{Q}(kT)\right\}$$
$$+\ \mathbb{P}\left\{|\hat{D}_T(kT) - W_T(kT)| > \frac{\kappa}{2}\|\mathbf{Q}(kT)\| \,\Big|\mathbf{Q}(kT)\right\}$$
$$\leq \mathbb{P}\left\{\left\|\overline{\mathbf{r}}(\tilde{\mathbf{Q}}(kT),\tau) - \frac{1}{T}\sum_{\sigma=0}^{T-1}\hat{\mathbf{D}}(kT+\sigma)\right\| > \frac{\kappa}{2}\,\Big|\mathbf{Q}(kT)\right\}$$
$$+\ \mathbb{P}\left\{\left\|\frac{1}{T}\sum_{\sigma=0}^{T-1}\hat{\mathbf{D}}(kT+\sigma) - \frac{1}{T}\sum_{\sigma=0}^{T-1}\mathbf{D}(kT+\sigma)\right\| > \frac{\kappa}{2}\,\Big|\mathbf{Q}(kT)\right\},$$

where the final bound is due to the Schwarz Inequality. In the RHS of the above inequality, the first probability term is bounded by Lemma A.1. To bound the second term, note that the difference $\sum_{\sigma=0}^{T-1}\hat{\mathbf{D}}(kT+\sigma) - \sum_{\sigma=0}^{T-1}\mathbf{D}(kT+\sigma)$ represents the number of lost rewards over a frame due to empty queues. In particular, if each queue in the system has at least T packets at the beginning of the frame, this difference is zero. Thus,

$$\mathbb{P}\left\{\left\|\frac{1}{T}\sum_{\sigma=0}^{T-1}\hat{\mathbf{D}}(kT+\sigma) - \frac{1}{T}\sum_{\sigma=0}^{T-1}\mathbf{D}(kT+\sigma)\right\| > \frac{\kappa}{2}\,\Big|\mathbf{Q}(kT)\right\} < 1_{\mathbf{Q}(kT)<T.\mathbf{1}}.$$

Let us now return to (A.19) and upper bound the RHS as

$$\leq \kappa\|\mathbf{Q}(kT)\| + \left(\sum_i Q_i(kT)\right)\left(ce^{-\eta(\kappa)T} + 1_{\mathbf{Q}(kT)<T.\mathbf{1}}\right)$$
$$\leq \left(\sum_i Q_i(kT)\right)\left(\kappa + ce^{-\eta(\kappa)T}\right) + NT,$$

where N is the number of queues. We can now use the above bound on the RHS of (A.19), along with (A.18) to upper bound the drift in (A.16).

$$\Delta_T(kT)/T \leq B+\sum_i Q_i(kT)\left[\lambda_i - r_i^*(\mathbf{Q}(kT))\right]+\left(\sum_i Q_i(kT)\right)\left(2\kappa + ce^{-\eta(\kappa)T}\right)+NT. \tag{A.20}$$

Let $\delta = 2\kappa + ce^{-\eta(\kappa)T}$. Assume now that the input rate vector λ lies in the interior of the δ-stripped region $\Lambda - \delta\mathbf{1}$. That is, there exists a $\xi > 0$ such that $\lambda + \xi\mathbf{1} = r - \delta\mathbf{1}$, for $r \in \Lambda$. Thus,

$$\Delta_T(kT)/T \leq B + NT + \sum_i Q_i(kT)\left[r_i - r_i^*(\mathbf{Q}(kT))\right] - \left(\sum_i Q_i(kT)\right)\xi.$$

Finally, noting that $\sum_i Q_i(kT)\left[r_i - r_i^*(\mathbf{Q}(kT))\right] \leq 0$ by the definition of $r_i^*(\mathbf{Q}(kT))$, we get

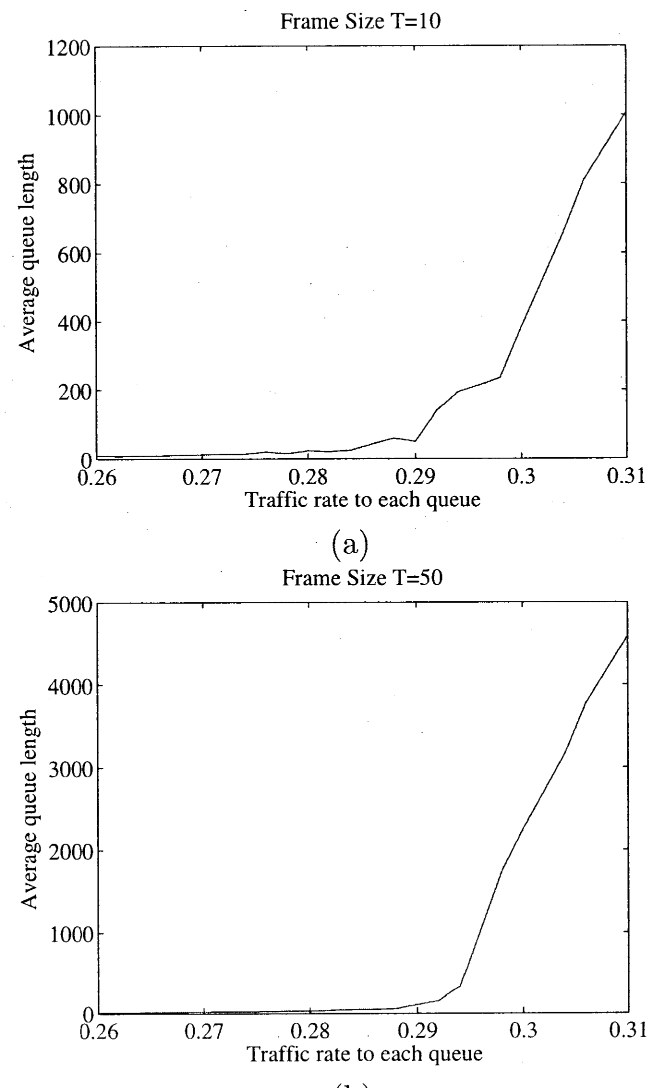
$$\Delta_T(kT)/T \leq B + NT - \left(\sum_i Q_i(kT)\right)\xi.$$

This shows that the system is stable under our frame-based policy for arrival rates inside the δ-stripped region $\Lambda - \delta\mathbf{1}$. Since δ can be made arbitrarily small by choosing sufficiently large values for T and τ, our policy can support rates arbitrarily close to the capacity region boundary, with a corresponding tradeoff in delay and computational complexity. □

A.4.1 Simulation results for the frame-based policy

We now provide some basic simulation results for the frame-based policy. In Figure A-4, we plot the average queue length of one of the queues, under the frame-based policy, as a function of the arrival rate. We take $\epsilon = 0.25$, and consider a symmetric rate scenario, where independent Poisson traffic of equal rates feeds the two queues. Each simulation run was carried out over ten thousand frames, with a frame sizes of $T = 10$ and $T = 50$ in Figure A-4(a) and Figure A-4(b), respectively.

Under this symmetric traffic scenario, the theoretical boundary of the capacity region lies at $(\lambda_1, \lambda_2) = (0.3125, 0.3125)$. The first observation we make from the



(a)

(b)

Figure A-4: The average queue length as a function of the symmetric arrival rate under the frame-based policy, for (a) $T = 10$ and (b) $T = 50$.

figure is that the frame-based policy easily stabilizes arrival rates up to 0.29 even for small frame sizes such as $T = 10$. There is considerable queue build-up at $(\lambda_1, \lambda_2) = (0.3, 0.3)$, and very large build up when the symmetric rate equals 0.31.

Another interesting point to note from the figure is that in heavy traffic, the average queue length when $T = 50$ is roughly a factor of five larger than when $T = 10$. This conforms to the theoretical prediction that the frame-based policy inherently suffers from an $O(T)$ average congestion level in the queues. This implies that although the frame-based policy is theoretically optimal for large T, it is possible that for a given traffic rate, a large frame size leads to considerable delay.

A.5 Conclusions

In this chapter, we studied the problem of scheduling over uncertain wireless channels, where channel state information can only be indirectly obtained, using past successes and failures of transmissions. We showed that the capacity region boundary for such a system can be approximated arbitrarily well by a sequence of LP solutions. We then incorporated the LP solution into a queue length based scheduling framework, to obtain a throughput optimal policy for the system.

Although we explicitly dealt with a two channel symmetric setting, our methodology extends naturally to more than two asymmetric channels. However, when the number of channels becomes asymptotically large, the dimensionality of the LP approximation increases exponentially in the number of channels. In such a case, it may be more practical to resort to the sub-optimal policy from [36]. On the other hand, for relatively small system sizes ($N \approx 10$), our method may entail solving an LP with a dimensionality of a few thousands, which is not prohibitive in today's setting.

For future work, it would be interesting to obtain *structural properties* of optimal policies for the backlogged system. For example, we believe that threshold policies should be sufficient to achieve the rate region boundary. If this is indeed the case, we can use a simple threshold policy over long frames to obtain a throughput optimal policy, instead of solving a large LP in every frame.

Bibliography

[1] S.H.A. Ahmad, M. Liu, T. Javidi, Q. Zhao, and B. Krishnamachari. Optimality of myopic sensing in multichannel opportunistic access. *IEEE Transactions on Information Theory*, 55(9):4040–4050, 2009.

[2] E. Altman. *Constrained Markov Decision Processes.* Chapman and Hall, 1999.

[3] V. Anantharam. Scheduling strategies and long-range dependence. *Queueing Systems*, 33(1):73–89, 1999.

[4] S. Asmussen. *Applied probability and queues.* Springer-Verlag, 2003.

[5] D.P. Bertsekas and R. Gallager. *Data networks.* Prentice Hall, 1987.

[6] D. Bertsimas, I.C. Paschalidis, and J.N. Tsitsiklis. Asymptotic buffer overflow probabilities in multiclass multiplexers: An optimal control approach. *IEEE Transactions on Automatic Control*, 43(3):315–335, 1998.

[7] D. Bertsimas and J. N. Tsitsiklis. *Introduction to Linear Optimization.* Athena Scientific, 1997.

[8] P. Billingsley. *Probability and Measure.* Wiley, New York, third edition, 1995.

[9] N.H. Bingham, C.M. Goldie, and J.L. Teugels. *Regular variation.* Cambridge Univ. Press, 1987.

[10] S.C. Borst, O.J. Boxma, R. Núñez-Queija, and A.P. Zwart. The impact of the service discipline on delay asymptotics. *Performance Evaluation*, 54(2):175–206, 2003.

[11] S.C. Borst, M. Mandjes, and M. van Uitert. Generalized processor sharing with light-tailed and heavy-tailed input. *IEEE/ACM Transactions on Networking*, 11(5):834, 2003.

[12] A. Brzezinski and E. Modiano. Dynamic reconfiguration and routing algorithms for IP-over-WDM networks with stochastic traffic. *Journal of Lightwave Technology*, 23(10):3188, 2005.

[13] L. Bui, A. Eryilmaz, R. Srikant, and X. Wu. Joint asynchronous congestion control and distributed scheduling for multi-hop wireless networks. In *IEEE INFOCOM*, page 1. IEEE, 2006.

[**14**] P. Chaporkar, K. Kar, X. Luo, and S. Sarkar. Throughput and fairness guarantees through maximal scheduling in wireless networks. *IEEE Transactions on Information Theory*, 54(2):572, 2008.

[**15**] D.B.H. Cline. Intermediate Regular and Π Variation. *Proceedings of the London Mathematical Society*, 3(3):594, 1994.

[**16**] T. Cover and A. El Gamal. Capacity theorems for the relay channel. *IEEE Transactions on Information Theory*, 25(5):572–584, 1979.

[**17**] T.M. Cover. Comments on broadcast channels. *IEEE Transactions on Information Theory*, 44(6):2524–2530, 1998.

[**18**] A. Dembo and O. Zeitouni. *Large deviations techniques and applications.* Springer-Verlag, 2009.

[**19**] A. Ephremides and B. Hajek. Information theory and communication networks: An unconsummated union. *IEEE Transactions on Information Theory*, 44(6):2416–2434, 1998.

[**20**] A. Eryilmaz, R. Srikant, and J.R. Perkins. Stable scheduling policies for fading wireless channels. *IEEE/ACM Transactions on Networking*, 13(2):411–424, 2005.

[**21**] S. Floyd and V. Jacobson. Random early detection gateways for congestion avoidance. *IEEE/ACM Transactions on networking*, 1(4):397–413, 1993.

[**22**] J. Galambos and E. Seneta. Regularly varying sequences. *Proceedings of the American Mathematical Society*, 41(1):110–116, 1973.

[**23**] R. Gallager. Basic limits on protocol information in data communication networks. *IEEE Transactions on Information Theory*, 22(4):385–398, 1976.

[**24**] R.G. Gallager. *Discrete stochastic processes.* Kluwer, 1996.

[**25**] A. Ganesh, N. O'Connell, and D. Wischik. *Big queues.* Springer Verlag, 2004.

[**26**] P. Giaccone, E. Leonardi, B. Prabhakar, and D. Shah. Delay bounds for combined input-output switches with low speedup. *Performance Evaluation*, 55(1-2):113–128, 2004.

[**27**] P. Giaccone, D. Shah, and B. Prabhakar. An efficient randomized algorithm for input-queued switch scheduling. In *Proceedings of the The Ninth Symposium on High Performance Interconnects*, page 3. IEEE Computer Society, 2001.

[**28**] A. Gopalan, C. Caramanis, and S. Shakkottai. On wireless scheduling with partial channel-state information. In *Proc. Ann. Allerton Conf. Communication, Control and Computing*, 2007.

[**29**] M. Harchol-Balter, A. Scheller-Wolf, and A.R. Young. Surprising results on task assignment in server farms with high-variability workloads. In *ACM SIGMETRICS 2009*, pages 287–298. ACM, 2009.

[**30**] D. Heath, S. Resnick, and G. Samorodnitsky. Heavy tails and long range dependence in on/off processes and associated fluid models. *Mathematics of Operations Research*, 23(1):145–165, 1998.

[**31**] S. Jagabathula and D. Shah. Optimal delay scheduling in networks with arbitrary constraints. In *SIGMETRICS*, pages 395–406. ACM, 2008.

[**32**] K. Jagannathan, E. Modiano, and L. Zheng. Effective resource allocation in a single-server queue: How much control is necessary. In *Forty-Sixth Annual Allerton Conference on Communication, Control, and Computing, Monticello IL*, 2008.

[**33**] K. Jagannathan, E. Modiano, and L. Zheng. On the trade-off between control rate and congestion in single server systems. In *IEEE INFOCOM*, pages 271–279, 2009.

[**34**] L.B. Le, K. Jagannathan, and E. Modiano. Delay analysis of maximum weight scheduling in wireless ad hoc networks. *Proc. of IEEE CISS*, pages 389–394, 2009.

[**35**] W.E. Leland, M.S. Taqqu, W. Willinger, and D.V. Wilson. On the self-similar nature of Ethernet traffic (extended version). *IEEE/ACM Transactions on networking*, 2(1):1–15, 1994.

[**36**] C. Li and M.J. Neely. On achievable network capacity and throughput-achieving policies over Markov ON/OFF channels. *Arxiv preprint arXiv:1003.2675*, 2010.

[**37**] X. Lin and N.B. Shroff. Joint rate control and scheduling in multihop wireless networks. In *IEEE Conference on Decision and Control*, 2004.

[**38**] X. Lin, N.B. Shroff, and R. Srikant. On the connection-level stability of congestion-controlled communication networks. *IEEE Transactions on Information Theory*, 54(5):2317–2338, 2008.

[**39**] S. Mannor and J.N. Tsitsiklis. On the empirical state-action frequencies in Markov decision processes under general policies. *Mathematics of Operations Research*, 30(3):545, 2005.

[**40**] M. Markakis, E. Modiano, and J.N. Tsitsiklis. Scheduling policies for single-hop networks with heavy-tailed traffic. In *Allerton Conference, Monticello IL*, 2009.

[**41**] N. McKeown, A. Mekkittikul, V. Anantharam, and J. Walrand. Achieving 100% throughput in an input-queued switch. *IEEE Transactions on Communications*, 47(8):1260–1267, 1999.

[**42**] S.P. Meyn, R.L. Tweedie, and J.L. Hibey. *Markov chains and stochastic stability*. Springer London et al., 1996.

[**43**] M.J. Neely. Order optimal delay for opportunistic scheduling in multi-user wireless uplinks and downlinks. *IEEE/ACM Transactions on Networking*, 16(5):1188–1199, 2008.

[**44**] M.J. Neely, E. Modiano, and C.P. Li. Fairness and optimal stochastic control for heterogeneous networks. *IEEE/ACM Transactions on Networking (TON)*, 16(2):396–409, 2008.

[**45**] M.J. Neely, E. Modiano, and C.E. Rohrs. Power and server allocation in a multi-beam satellite with time varying channels. In *IEEE INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings*, volume 3, 2002.

[**46**] M.J. Neely, E. Modiano, and C.E. Rohrs. Dynamic power allocation and routing for time varying wireless networks. In *IEEE INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 1, 2003.

[**47**] A. Pantelidou, A. Ephremides, and A.L. Tits. Joint scheduling and routing for ad-hoc networks under channel state uncertainty. *Wireless Networks*, 15(5), 2009.

[**48**] K. Park and W. Willinger. *Self-similar network traffic and performance evaluation*. John Wiley & Sons, Inc. New York, NY, USA, 2000.

[**49**] J.R. Perkins and R. Srikant. The role of queue length information in congestion control and resource pricing. In *IEEE Conference on Decision and Control*, volume 3, pages 2703–2708, 1999.

[**50**] M.L. Puterman. *Markov decision processes: Discrete stochastic dynamic programming*. John Wiley & Sons, Inc. New York, NY, USA, 1994.

[**51**] C.Y. Robert and J. Segers. Tails of random sums of a heavy-tailed number of light-tailed terms. *Insurance: Mathematics and Economics*, 43(1):85–92, 2008.

[**52**] R.T. Rockafellar. *Convex Analysis*. Princeton University Press, Princeton, N.J., 1970.

[**53**] W. Rudin. *Principles of mathematical analysis*, page 158. McGraw-Hill, 1976.

[**54**] D. Shah and M. Kopikare. Delay bounds for the approximate maximum weight matching algorithm for input queued switches. In *INFOCOM*, 2002.

[**55**] D. Shah and D. Wischik. Optimal scheduling algorithms for input-queued switches. In *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, pages 1–11, 2006.

[**56**] S. Shakkottai. Effective capacity and QoS for wireless scheduling. *IEEE Transactions on Automatic Control*, 53(3):749–761, 2008.

[**57**] C.E. Shannon. A Mathematical Theory of Communication. *Bell Sys. Tech. J.*, 27:379–423, 1948.

[**58**] G. Sharma, R. Mazumdar, and N. B. Shroff. On the complexity of scheduling in wireless networks. In *MobiCom '06: Proceedings of the 12th annual international conference on Mobile computing and networking*, pages 227–238, New York, NY, USA, 2006. ACM.

[**59**] K. Sigman. A primer on heavy-tailed distributions. *Queueing Systems*, 33(1-3):261–275, 1999.

[**60**] A.L. Stolyar and K. Ramanan. Largest weighted delay first scheduling: Large deviations and optimality. *Annals of Applied Probability*, 11(1):1–48, 2001.

[**61**] V.G. Subramanian. Large Deviations Of Max-Weight Scheduling Policies On Convex Rate Regions. *Proceedings of ITA*, 2008.

[**62**] L. Tassiulas. Linear complexity algorithms for maximum throughput in radionetworks and input queued switches. In *IEEE INFOCOM'98. Seventeenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings*, volume 2, 1998.

[**63**] L. Tassiulas and A. Ephremides. Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks. *IEEE Transactions on Automatic Control*, 37(12):1936–1948, 1992.

[**64**] L. Tassiulas and A. Ephremides. Dynamic server allocation to parallel queues with randomly varying connectivity. *IEEE Transactions on Information Theory*, 39(2):466–478, 1993.

[**65**] V.J. Venkataramanan and X. Lin. On the large-deviations optimality of scheduling policies minimizing the drift of a Lyapunov function. In *Forty-Seventh Annual Allerton Conference on Communication, Control, and Computing, Monticello IL*, 2009.

[**66**] V.J. Venkataramanan and X. Lin. On the queue-overflow probability of wireless systems: A new approach combining large deviations with Lyapunov functions. *submitted to IEEE Trans. on Information Theory*, 2009.

[**67**] V.J. Venkataramanan and X. Lin. On wireless scheduling algorithms for minimizing the queue-overflow probability. *IEEE/ACM Transactions on Networking*, 2009.

[**68**] V.J. Venkataramanan, X. Lin, S. Shakkottai, and L. Ying. On scheduling for minimizing end-to-end buffer usage over multihop wireless networks. In *IEEE INFOCOM*, 2010.

[**69**] W. Whitt. Approximation of dynamic programs – I. *Mathematics of Operations Research*, 3:231–243, 1978.

[**70**] W. Whitt. Approximation of dynamic programs – II. *Mathematics of Operations Research*, 4:179–185, 1979.

[**71**] P. Whittle. Restless bandits: Activity allocation in a changing world. *Journal of Applied Probability*, 25:287–298, 1988.

[**72**] A. Wierman and B. Zwart. Is tail-optimal scheduling possible? Under Submission.

[**73**] X. Wu, R. Srikant, and J.R. Perkins. Queue-length stability of maximal greedy schedules in wireless networks. In *Proceedings of Information Theory and Applications Inaugural Workshop*, pages 6–10, 2006.

[**74**] L. Ying and S. Shakkottai. On throughput optimal scheduling with delayed channel state feedback. In *Information Theory and Applications Workshop*, pages 339–344, 2008.

[**75**] L. Ying and S. Shakkottai. Scheduling in mobile ad hoc networks with topology and channel state uncertainty. *IEEE INFOCOM, Rio de Janeiro, Brazil*, 2009.

[**76**] L. Ying, R. Srikant, A. Eryilmaz, and G.E. Dullerud. A large deviations analysis of scheduling in wireless networks. *IEEE Transactions on Information Theory*, 52(11):5088–5098, 2006.