Network Performance Analysis of Packet Scheduling Algorithms

by

Yashar Ghiassi-Farrokhfal

A thesis submitted in conformity with the requirements
for the degree of Doctor of Philosophy
Graduate Department of Electrical and Computer Engineering
University of Toronto

Abstract

Network Performance Analysis of Packet Scheduling Algorithms

Yashar Ghiassi-Farrokhfal

Doctor of Philosophy

Graduate Department of Electrical and Computer Engineering

University of Toronto

2012

Some of the applications in modern data networks are delay sensitive (e.g., video and voice). An end-to-end delay analysis is needed to estimate the required network resources of delay sensitive applications. The *schedulers* used in the network can impact the resulting delays to the applications. When multiple applications are multiplexed in a switch, a scheduler is used to determine the precedence of the arrivals from different applications.

Computing the end-to-end delay and queue sizes in a network of schedulers is difficult and the existing solutions are limited to some special cases (e.g., specific type of traffic). The theory of *Network Calculus* employs the *min-plus algebra* to obtain performance bounds. Given an upper bound on the traffic arrival in any time interval and a lower bound on the available service (called the *service curve*) at a network element, upper bounds on the delay and queue size of the traffic in that network element can be obtained. An equivalent end-to-end service curve of a tandem of queues is the min-plus convolution of the service curves of all nodes along the path. A probabilistic end-to-end delay bound using network service curve scales with $O(H \log H)$ in the path length H. This improves the results of the conventional method of adding per-node delay bounds scaling with $O(H^3)$.

We have used and advanced Network Calculus for end-to-end delay analysis in a network of schedulers. We formulate a service curve description for a large class of schedulers which we call Δ-schedulers. We show that with this service curve, tight single node delay and backlog bounds can be achieved. In an end-to-end scenario, we formulate a new convolution theo-

rem which considerably improves the end-to-end probabilistic delay bounds. We specify our probabilistic end-to-end delay and backlog bounds for exponentially bounded burstniess (EBB) traffic arrivals. We show that the end-to-end delay varies considerably by the type of schedulers along the path. Using these bounds, we also show that a if the number of flows increases, the queues inside a network can be analyzed in isolation and regardless of the network effect.

Contents

Chapter 1

Introduction

Conventional telephony network was based on *circuit switched networks* in which a communication channel is assigned between each sender and receiver before they start to communicate (see [12]). In a telephony network, a fixed bandwidth is allocated to each user regardless of the traffic it is injecting in the network. This allocated bandwidth is available for the whole duration of communication. If the rate of traffic generation of a user is smaller than its dedicated bandwidth, the leftover bandwidth cannot be used by other users and will be wasted. To obviate the problem of inefficient use of bandwidth, *data networks* have emerged, where the information is sent in small batches called *packets*. Data networks are also known as *packet switched networks*. Sharing a link by packets from multiple flows is called *statistical multiplexing*. If at a certain time, a packet arrives at a link, and the link is busy serving another packet, then the newly arrived packet will be stored in a *buffer* to be served later. The aggregate of all packets in the buffer at any time is called *backlog*. There is no preassigned dedicated bandwidth to a user in packet switched networks. All buffered packets from all users share the total bandwidth. Statistical multiplexing avoids bandwidth allocation to inactive users, and this increases the bandwidth utilization compared to circuit switched networks considerably. This gain is known as the *statistical multiplexing gain*.

Packet switched networks are employed to serve packets from different applications in modern data network. The applications are either delay sensitive (e.g., video and voice) or not delay sensitive (e.g., email). When packets from different applications are multiplexed in a bottleneck, managing the available capacity among applications is crucial. A scheduling algorithm decides which packet from which application to serve next, among all packets in the buffer. A scheduler is called *work-conserving* if it cannot be idle when the buffer is not empty. See [8] for a review of the gain obtained by the scheduling algorithms in terms of throughput and managing the available resources. In the sequel, we review some of the most widely used schedulers.

- First-In-First-Out (FIFO): In this scheduler, the packet with the earliest arrival time among all stored packets in the buffer will be served first. Although FIFO is an attractive scheduler because of the simple implementation, it cannot differentiate services between flows. For instance, FIFO is not a proper scheduler if traffic flows have largely different delay constraints.

- Static Priority (SP): All flows are categorized into a set of classes. A priority level is assigned to each class. SP can integrate applications with different service requirements by placing them in different priority classes. Packets from a certain class will be served only if there is no buffered packet from higher priority classes. Arriving packets from the same class will be served in a FIFO order. A SP scheduler is simple to implement and offers service differentiation for different applications. However, a problem with SP schedulers is that the lower priority classes may experience starvation (unfair usage of bandwidth by higher priority classes). Treating a traffic flow as if it belongs to the lowest priority class in a SP scheduler is also known as *blind multiplexing* (BMux) in the literature and has a special property that receives the lowest amount of service among all work-conserving schedulers.

- Earliest Deadline First (EDF): An a priori delay bound is assigned to each flow in this type of scheduler. Suppose d_i^* is the a priori delay bound for flow i. Then, if there is an arrival from flow i at time t, the deadline assigned to that arrival is $t + d_i^*$. Packets waiting to be transmitted in the buffer will be served in the order of their deadlines. EDF is relatively difficult to implement, since upon arrival of any packet, a search is required to find the position of that newly arrived packet among all previously stored packets. It is proved in [75] that EDF is the optimal scheduler in an underloaded regime to serve a set of delay sensitive flows in an isolated link. In an overloaded regime, however, Locke's [77] experiments discover a shortcoming of EDF schedulers known as *domino effect*. If one packet misses its deadline, then, this might cause the subsequent packets to miss their deadlines as well, and the deadline violation propagates.

- Rate-schedulers: This class of schedulers seeks to provide rate fairness to the flows. Rate fairness is ideally obtained in a fluid flow model, where the cumulative traffic arrival is a continuous function. The fluid flow model of the ideal scheduler with rate fairness is named Generalized Processor Sharing (GPS) in [81]. GPS is a generalization of the uniform processor sharing proposed in [57]. In GPS, a weight is assigned to each flow and the total link capacity is shared according to those weights. Suppose that $\mathcal{N}$ is a set of traffic flows arriving to a GPS link with capacity C. Represent the set of all backlogged flows at time t by $\mathcal{B}(t)$. If $\phi_i > 0$ is the weight of flow $i \in \mathcal{N}$, then, the bandwidth assigned to flow i with instantaneous arrival rate $r_i(t)$ at time t is

$$r_0(t) = \begin{cases} r_i(t) & \text{if } i \notin \mathcal{B}(t) \\ \frac{\phi_i C}{\sum_{j \in \mathcal{B}(t)} \phi_j} & \text{if } i \in \mathcal{B}(t) \ . \end{cases}$$

  Consequently, the guaranteed service rate to flow i for such a system is $\frac{\phi_i C}{\sum_{j \in \mathcal{N}} \phi_j}$.

  When a fluid flow assumption is not valid (e.g., packet sizes are not very small with respect to the link capacity), then GPS cannot be implemented accurately. There are packet schedulers in the literature which approximate the mechanism of processor sharing algorithms, e.g., Weighted Fair Queuing (WFQ) [33], Weighted Round Robin (WRR) [52], and Virtual Clock (VC) [105].

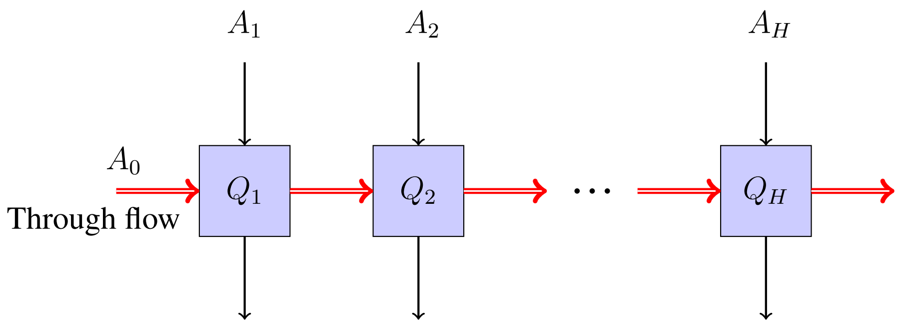


Figure 1.1: A tandem network of packet switches

1.1 Problem Definition

Suppose that a through flow A_0 passes through a tandem network consisting of H packet switches as depicted in Fig. 1.1. Each packet switch uses a scheduling algorithm to serve the packet arrivals. We use the term *node* to refer to a packet switch frequently throughout the thesis. At each node h there are cross flows with the aggregate process represented by A_h. Cross flows and through flow are not necessarily independent of each other. The problems which we consider in this thesis are the following: What is the end-to-end delay and backlog that through flow packets experience? How does this delay differ if scheduling algorithms of some or all nodes in the path change? How are the end-to-end delay and backlog affected by the choice of schedulers if the path length H is large?

The above problems may depend on the type of traffic arrivals and consequently on the type of applications in the network. Given the traffic statistics or type of applications, the answers to the above questions are needed to estimate the required bandwidth to support multiple delay sensitive applications in a network of schedulers (e.g., for Internet service providers to know the required bandwidth to support a certain number of customers). Moreover, the end-to-end backlog analysis is needed for switch designs and estimating the required buffer size at each switch in a network. Indeed, end-to-end scheduling analysis is difficult and has been mostly studied with simplifying assumptions or considering special cases.

The delay and backlog of a traffic flow in a scheduler is well-studied in single node scenarios. One of the insightful items which can be used to compare schedulers is the *schedulable region*. Consider a set $\mathcal{N}$ of applications which generate traffic flows to a scheduler. Each application has a delay constraint. n_i represents the number of flows from application $i \leq |\mathcal{N}|$. The deterministic schedulable region is a $|\mathcal{N}|$-dimensional area which includes any point $(n_1, \ldots, n_{|\mathcal{N}|})$ corresponding to the flow numbers for which all delay constraints of all applications are satisfied. The larger the schedulable region, the better the scheduler to serve delay sensitive applications. It is shown in [75] that EDF leads to the largest deterministic schedulable region among all schedulers. If rare violations of delay constraints are allowed, a probabilistic schedulable region is obtained which is much larger than the deterministic one. Statistical multiplexing gain can be captured in a probabilistic schedulable region. The comparison of the probabilistic schedulable regions of EDF with GPS in [1], [67], with SP in [9], [67], and with FIFO (via simulation) in [88] show that EDF outperforms other schedulers in the probabilistic settings as well. However, the improvement achieved by the choice of scheduler is not considerable compared to the statistical multiplexing gain [9], [67]. Under a many sources asymptotic regime, where the number of flows converges to infinity while the per-flow capacity and per-flow buffer size remains fixed, it is shown in [38], [102] that scheduling is not important and all schedulers perform the same as a SP scheduler which assigns the highest priority to the through flow.

Although the effect of schedulers on the per-flow delay is well-studied in single queue scenarios, in end-to-end scenarios, there are only observations in special cases. Via simulation, Grossglauser and Keshav [47] show that if constant bit rate (CBR) through flow traffic is multiplexed with a small (up to 30) number of CBR cross flows, FIFO and WRR perform similarly in terms of end-to-end delay percentiles. Replacing CBR cross flows with Poisson ones, FIFO yields much smaller end-to-end delays compared to WRR, especially at a high utilization. The analytical end-to-end delay bounds of EDF schedulers are compared with those of GPS [1], [91], and FIFO [87], [90]. However, the assumptions and system models are different from what we consider in this thesis. In particular, in the above works, through flow and cross flow are independent. The bounds are obtained by assuming a shaper before each scheduler [91], [87], [90], or by modifying the schedulers of all queues $h = 2, \ldots, H$ such that the traffic distortion is controlled by the scheduler and dropping those packets which exceed a certain threshold [1].

Although the existing observations on end-to-end delay and backlog in a network of schedulers are insightful, they do not provide conclusive answers to the raised questions.

1.2 Scheduling Analysis and Traffic Characterizations

Scheduling analysis in data networks was started by the evolution of *Queuing Theory* [58] in the early 20th century. The joint steady state distribution of the backlog status of all queues in a network is shown to be the product of the state probabilities of all queues in Jackson networks [57], BCMP networks [4], or Kelly networks [53] which all share the assumptions that external arrivals are Poisson and that service time distributions are independent. Although Queueing Theory has been influential in understanding the behavior of queuing systems, its traffic model assumption is not adequate for modern data network [82]. In particular, by the emergence of high-speed data network, applications such as voice and video are transmitted via links which are more bursty than Poisson traffic.

There are many alternative traffic models proposed in the literature to capture the burstiness of voice and video traffic. An ON-OFF model is used for voice traffic in [11]. Assuming a fluid flow model, Markov-Modulated Fluid (MMF) models are used for voice [31] and video traffic [78]. If packet sizes are not small compared to the link capacities, fluid flow assumption is not practical. Markov-Modulated Poisson Process (MMPP) is a packet model for the analysis of voice and video [86]. Burstiness of traffic is captured in MMF and MMPP models by using different transmission rates for different states of the underlying Morkov chains.

Via measurement studies, Internet aggregate traffic was shown to exhibit long-range dependency and self similarity [5], [24]. Long range dependent arrival processes (also referred to as processes with large memories) are illustrated by the coupling between arrivals in different time intervals which decreases slowly as the distance between time intervals increases. Long range dependency of an arrival process can be examined by computing the autocorrelation of that process. Markov-modulated processes are short range dependent (short memory), and Poisson traffic has zero memory. Self similar processes are referred to those which have identical distributions in any time scale. More precisely, X is a self similar process if for any $a > 0$, $X(t) \sim_{dist} a^{-H}X(at)$, [1] where $H \in (0, 1)$ is known as the Hurst parameter. Several traffic models have been proposed to capture the self similarity and long range dependency of the Internet traffic. For instance, using the properties of α-stable processes, self similar processes are characterized by envelopes in [42], [51]. In another work, Laskin *et al.* [65] introduce Fractional Levy Motion (FLM), which is a non-Gaussian self-similar stochastic process, to model Internet aggregate traffic. Exploiting the properties of FLM, an asymptotic lower bound for the probability of buffer overflow is computed and shown to decrease hyperbolically as a function of the buffer size.

To analyze non-Poisson traffic models, some traffic characterizations have been introduced. One of these traffic characterizations is the *effective bandwidth* [55] which was influential in the network performance analysis. One can also name *Exponentially Bounded Burstiness* [98] and *effective envelope* [9] as two important traffic characterizations in the literature.

The idea of the effective bandwidth is to compute a probabilistic upper bound on the per-flow required bandwidth similar to the concept of the dedicated per-user bandwidth in circuit switched networks. The effective bandwidth of a flow is a scalar between the average rate and peak rate of that flow and determines a minimum decay rate of exponentially decreasing delay or backlog bound violation probability. Effective bandwidth is defined in various forms in the literature. We review two of the most widely used ones in the following. For an arrival process

[1] $X \sim_{dist} Y$ for random variables X and Y if $P\{X \leq x\} = P\{Y \leq x\}$ for any x.

A and some $\alpha \geq 0$, the *effective bandwidth* Eb_A associated with A is defined by

$$Eb_A(\alpha) = \lim_{t\to\infty} \frac{1}{\alpha t} \sup_{u\geq 0} \log\Big(E[e^{\alpha A(u,u+t)}]\Big) , \tag{1.1}$$

where $A(s,t)$ is the total arrival from A in $[s,t)$ for any non-negative $s,t \geq 0$. Multiple flows can be served in a link if the sum of their effective bandwidths does not exceed the total capacity [54].

Backlog bounds were shown to decrease exponentially fast in the steady state for arrivals with *bounded*, *differentiable* effective bandwidths. More precisely, if A is an arrival process to a link with capacity C, and Eb_A is the corresponding effective bandwidth, then from [56], the steady state backlog B satisfies the following asymptotically as $b \to \infty$

$$P(B \geq b) \sim Ke^{-\alpha^* b} , \tag{1.2}$$

where $Eb_A(\alpha^*) = C$, $f(x) \sim g(x)$ means $\lim_{x\to\infty} \frac{f(x)}{g(x)} = 1$ for any functions f and g, and K is a constant. There is some work to improve the above bounds based on the type of traffic by evaluating K and including more complex functions of b in Eq. (1.2). For instance, Elwalid *et al.* [34] improve bounds for MMF models, and Guerin *et al.* [48] improve the bounds for ON-OFF sources. Skelly *et al.* [92] use histograms of the backlog distributions to improve the bounds for the case of video traffic.

The effective bandwidth of an aggregate of some flows is the sum of the effective bandwidths of all individual flows. However, it is shown in [18] that the effective bandwidth from Eq. (1.1) cannot capture the multiplexing gain fully if the traffic is more bursty than Poisson traffic. By incorporating the notion of time in *effective bandwidth functions* proposed by Kelly [55] the statistical multiplexing is captured. The effective bandwidth function of process A at any time $t \geq 0$ and some $\alpha > 0$ is defined as

$$Eb_A(\alpha, t) = \frac{1}{\alpha t} \sup_{u\geq 0} \log\Big(E[e^{\alpha A(u,u+t)}]\Big) . \tag{1.3}$$

There is extensive work to compute effective bandwidths (functions) for different traffic sources. For example, the effective bandwidth is formulated for CBR traffic, memoryless sources, and Markov modulated sources in [56]. The effective bandwidth functions for regulated traffic, periodic sources, two-state Markov chain, and FBM sources are derived in [55].

Another traffic description is Exponentially Bounded Burstiness (EBB) traffic [98], [99]. This class of arrivals includes sources such as Markov modulated processes. Each EBB traffic is characterized by a rate and the likelihood of exceeding that rate. The probability that the total traffic in any time interval exceeds the rate in that interval is exponentially decreasing. More precisely, a traffic arrival A is called an EBB traffic with parameters $M, \rho, \alpha > 0$ and is represented by $A \sim (M, \rho, \alpha)$ if it satisfies the following for any $\sigma \geq 0$ and $s \leq t$:

$$P\{A(s,t) > \rho(t-s) + \sigma\} \leq Me^{-\alpha\sigma} \ . \tag{1.4}$$

The exponential tail bound of EBB traffic can be used to construct a probabilistic delay and backlog. For instance, a backlog bound is obtained by applying Reich's equation which formulates the total backlog in a link with capacity C as follows

$$B(t) = \sup_{0 \leq s \leq t} \{A(s,t) - C(t-s)\} \ . \tag{1.5}$$

Using Reich's equation and Boole's inequality, if $A \sim (M, \rho, \alpha)$, then for any $b \geq 0$

$$P\{B(t) > b\} \leq Ke^{-\alpha b} \ ,$$

where $K = \frac{M}{1-e^{-\alpha(C-\rho)}}$.

Effective envelope [9] is another useful traffic description. A non-decreasing non-negative function G is a local effective envelope for arrival process A with bounding function ε, if it satisfies the following for some $\sigma \geq 0$ and any $s, t \geq 0$

$$P\{A(s,t) > G(t-s;\sigma)\} \leq \varepsilon(\sigma) \ , \tag{1.6}$$

where both G and ε can be functions of $\sigma \geq 0$. In fact, this definition is more general than EBB and effective bandwidth. Li *et al.* [67] formulate the relationship between the effective bandwidth functions and the effective envelopes.

The backlog behavior of this type of traffic characterization is determined by how fast ε in Eq. (1.6) decays in σ. Having effective envelopes for the traffic arrivals in a link, probabilistic delay and backlog bound can be computed using different techniques. For instance by defining ε as a special decreasing integrable function of $t - s$ in Eq. (1.6) or using an upper bound on the busy period [67].

In a single queue scenario, combining the useful features of one of the above traffic characterizations and the properties of each specific scheduler facilitates transient queue analysis for a large class of traffic models such as those for voice and video and even heavy-tailed self-similar (HTSS) traffic [70], and for a variety of schedulers, e.g., SP [9], [60], FIFO [17], [37], EDF [88], [90], and GPS [79], [84].

1.3 Challenges of End-to-end Scheduling Analyses

As the traffic traverses the first node in a network, the traffic statistics will be distorted. This distortion makes the scheduling analysis of the nodes inside the network (e.g., nodes $h = 2, \ldots, H$ in Fig. 1.1) more complicated since the scheduler operation is based on the traffic arrival times. In some scenarios, the distortion is bounded. The following are two of these cases.

1. **Using traffic shapers and controllers:**

   In [43], [103] the concept of *traffic shaper* is introduced which enforces an envelope to a traffic flow. Placing a traffic shaper before each node in a network, the original statistics of the traffic arrival to the network can be restored. This enables a per-node delay and backlog analysis regardless of other nodes in the network. Traffic shapers are not work-conserving, i.e., there might be packets in the buffer waiting to be served, but the shaper does not pass it to the scheduler. This induces a delay of size d_{sh} to the end-to-end delay [43]. Suppose that d_h is a delay bound at node h assuming that the arrival traffic matches the envelope that the shaper enforces. An end-to-end delay bound d_{net} for the through flow is

$$d_{net} = d_{sh} + \sum_{h=1}^{H} d_h \ . \tag{1.7}$$

End-to-end scheduling analysis in the presence of shapers is extensively studied in the literature, e.g., [59], [87], [104].

2. **Coordinated scheduling:**

   This type of schedulers modifies the scheduling algorithms of the nodes inside the network (not the nodes on the edges of the network) such that the original traffic distortion can be controlled. Consider a tandem network of schedulers as in Fig. 1.1. In plain schedulers, the packets at any node are scheduled based on their local arrival times disregarding their deadlines at previous nodes. A Coordinated Multi-hop Schedulers (CMS) [68], however, determines the level of precedence of the packets at node $h = 2, \dots, H$ by considering their arrival times at the first node $h = 1$. Assume that flow i passes through some schedulers. Suppose we have a tagged virtual arrival from flow i that arrives to the first scheduler at time t^i. CMS assigns the following deadline to the tagged arrival at its h'th node

$$D_h^i = \begin{cases} t^i + d_1^* & h = 1 \\ D_{h-1}^i + d_h^* & h > 1 \ , \end{cases} \tag{1.8}$$

   where d_h^* is called the local deadline at node h. The arrivals at each node are served in the order of earliest deadline. The packet arrival times at any node $h = 2, \dots, H$ inside the network is bounded by a constant shifted versions of the arrival times to the first node $h = 1$. Coordinated versions of some important schedulers are proposed in the literature, e.g., coordinated EDF (CEDF) [1], [2], FIFO (known as FIFO$^+$) [22], and rate schedulers [94].

In this thesis, we do not assume the use of shapers or coordinated schedulers since they do not always appear in all network scenarios. We aim to compute a statistical end-to-end delay and backlog bound for a through flow when it passes through a cascade of schedulers shared by cross flows. The conventional method to obtain an end-to-end delay bound is to compute single node delays at each node along the end-to-end path, and add them up [64], [93], [98]. To do so, an output characterization at each node is required. In a tandem network, the through flow arrival to node h is the through flow departure from node $h-1$. This method, however, leads to loose bounds by overestimating the traffic burstiness. Indeed, it is assumed that the increase of traffic burstiness can happen simultaneously at all nodes. This event does not occur. Thus, accounting for this event to compute bounds makes the resulting end-to-end delay bound obtained by this method looser as the path length increases.

The problem of the looseness of adding per-node delay bounds has been addressed and resolved in *Network Calculus*. There are books [10], [16], [50], [63], and surveys [40], [80] which explore the concepts of Network Calculus. Network Calculus was originally proposed as a theory for deterministic worst case analysis in [25], [26]. In Network Calculus, the arrival traffic is represented by upper bounds and the available service is lower bounded by so-called *service curves*. Deterministic Network Calculus employs these bounds to compute upper bounds on the worst-case delay, backlog, and the departure traffic. The strength of the Network Calculus lies in the fact that it turns an end-to-end performance analysis to a single node one by computing an end-to-end equivalent *network service curve*.

Deterministic Network Calculus always considers the worst-case scenarios and does not capture the statistical multiplexing gain. Thus, Deterministic Network Calculus was extended soon to probabilistic settings in many subsequent papers such as [64] and named as the Stochastic Network Calculus. This theory is very general in terms of the traffic characterizations it assumes. Network Calculus needs upper bounds on the traffic arrivals and such bounds are available for a wide range of traffic models including any traffic with valid effective bandwidth functions [67], EBB descriptions [20], or effective envelopes [20], [67].

Ciucu *et al.* [20] provide a statistical network service curve formulation and compute an end-to-end analysis in a tandem of SP which assigns the lowest priority to the through flow. All traffic flows are assumed to be EBB arrivals. The resulting end-to-end delay bound is shown to scale by $O(H \log H)$, compared to adding per queue delay bounds which scales by $O(H^3)$. By computing lower bounds on the end-to-end delay, it is shown in [14] that the statistical end-to-end delay scales by $\Theta(H \log H)$. The lower bound is obtained by removing the cross flow in a FIFO tandem network with packetized through flow. Combining the properties of moment generating functions and the concepts of the Network Calculus, Fidler [39] shows that if the assumption of independence of through flow and cross flow is also added to the model, then similar to the deterministic worst case scenarios, the delay bound scales linearly with path length, i.e., $d_{net} = O(H)$.

1.4 Thesis Statement and Contribution

In spite of extensive improvements in the network analysis using Network Calculus, the state-of-the-art on the end-to-end delay and backlog analysis are not conclusive yet. The statistical end-to-end analysis in [20] assumes that all queues are SP with the lowest priority assigned to the through flow (BMux).

In this thesis, we advance the Stochastic Network Calculus for end-to-end delay and backlog analysis in the presence of packet scheduling. The results of this thesis facilitate the scheduling analysis and gain new insights on the impact of schedulers on the end-to-end delay and backlog.

Thesis Statement: The Stochastic Network Calculus can be used to formulate end-to-end delay and backlog bounds (tight in some regimes) in a tandem network of schedulers.

The contributions of this thesis are in two directions: theory and studying the impact of scheduling on EBB traffic sources.

Theory: We use the Stochastic Network Calculus to advance scheduler analysis. Our results are applied to a large class of schedulers which we define and call Δ-*schedulers*, which includes some schedulers such as FIFO, SP, and EDF.

In a single node, we formulate a tight *service curve* formulation for all Δ-schedulers, in the sense that it leads to a necessary and sufficient condition for deterministic (worst-case) delay bounds.

In a tandem network, we provide a new *statistical network service curve* which leads to considerably tighter end-to-end scheduling analyses. We do not assume traffic shapers, or scheduling algorithm modifications as in CMSs. Moreover, our analyses are applicable to the cases where through flows and cross flows are correlated. For such scenarios, we provide a methodology to compute statistical bounds for end-to-end delay, backlog, and departure processes.

Studying the impact of scheduling on EBB traffic sources: We study the effect of schedulers on the end-to-end delay for EBB traffic arrivals. Employing our new statistical network service curve formulation, we show that the end-to-end delay bounds for different schedulers can be extremely different on long paths. We also observe that the effect of passing through one SP which assigns the lowest priority (BMux) in a tandem of FIFO schedulers still exists even on long paths.

We apply our performance bounds to compute an end-to-end backlog bound in a tandem of schedulers. We show that the end-to-end backlog might be negligible for some schedulers if there are a few hundred flows. This implies that the input and output traffic processes to a link are similar and traffic is not distorted considerably as it passes through a network of schedulers. This allows us to decompose a network and analyze queues in isolation. Network decomposition is a well-studied research area in a network of FIFO or blind multiplexors and in an asymptotic regime when the number of flows converges to infinity. This thesis considers the network decomposition in a non-asymptotic regime and in a network of Δ-schedulers.

Finally, we show that long-term average rate of the departure process of a traffic in a FIFO scheduler converges to that of a tandem of blind multiplexors.

1.5 Thesis Structure

The remainder of the thesis is structured as follows.

In Chapter 2, we review definitions and the concepts of the Deterministic and Stochastic Network Calculus which will be used in this thesis. We start by providing the definitions of arrival envelopes and service curves, and computations of single node performance bounds. Then, we discuss network service curves, end-to-end delay bound computations, and pay burst only once phenomenon. In the stochastic regime, we review arrival envelope and service curve definitions, statistical network service curve, and probabilistic performance bound formulation.

In Chapter 3, we review methods for computing performance bounds in single node scenarios. Then, as a deterministic analysis, we introduce our class of schedulers. We provide a tight service curve description for our schedulers. The tightness is proved by showing that applying this service curve to the Network Calculus provides a necessary and sufficient condition for any delay bound. As an example of our analysis, we consider leaky bucket arrivals in the deterministic regime and Exponential Bounded Burstiness in the stochastic regime. We compute the probabilistic scheduling regions of different scheduling algorithms and different envelopes with numerical examples.

In Chapter 4, we first introduce existing approaches for statistical end-to-end analysis. Then, we provide a departure characterization in a network of FIFO schedulers. The characterization is in the context of long-term average rate, and in both overloaded and underloaded regimes. We also use this result for an end-to-end departure characterization in a large tandem network. We show that the long-term average rate in a network of FIFO schedulers converges to that of a network of blind multiplexors as the path length increases. In the second part of the chapter, we extend the single queue performance bound computations for Δ-schedulers to an end-to-end scenario. This is achieved by providing a new statistical network service curve, with which we can obtain end-to-end performance bounds. To examine our bounds, we consider leaky-bucket arrivals in the deterministic setting. We compute a lower bound for the end-to-end delay to show that our analysis leads to good end-to-end delay bounds and tight bounds in some cases. In the probabilistic setting, we consider EBB arrivals. We examine our results with those obtained from the previous network service curve formulation. We also study the effect of schedulers on the end-to-end delay and backlog bounds with numerical examples.

In Chapter 5, we review the literature on network decomposition in asymptotic regimes. Then, we use the end-to-end backlog and output bounds from Chapter 4 to analytically examine the viability of network decomposition in a network of schedulers as a function of scheduling mechanisms. The numerical results indicate that network decomposition might be valid for medium number (few hundreds) of sources for some schedulers.

Finally, in Chapter 6, we conclude the thesis and discuss future work.

Chapter 2

Network Calculus Review

In this chapter, we introduce the Network Calculus for end-to-end network analysis. We start by exploring the analogy between Network Calculus and systems theory. Then, we describe our model and our notations. We explore the concepts of the Deterministic Network Calculus, the bounds, and the advantage of using Network Calculus in terms of end-to-end delay bounds. Finally, we review the Stochastic Network Calculus, and how it can be used to compute probabilistic performance bounds.

2.1 Systems Theory vs. Network Calculus

Systems theory is a multi-disciplinary approach that considers a *system* as a set of independent and interacting parts. This theory provides an elegant method for end-to-end departure characterization in a tandem of *linear time invariant* (LTI) systems. Suppose that the system responses to arbitrary inputs $X_i(t)$ and $Y_i(t)$ are, respectively, $X_0(t)$ and $Y_0(t)$. The system is called linear if the system response to $aX_i(t) + bY_i(t)$ is $aX_0(t) + bY_0(t)$ for any real-value a and b. Moreover, the system is called time-invariant if the system response to $X_i(t + \tau)$ is $X_0(t+\tau)$ for any real-value τ. In fact, for a LTI system, it is possible to describe the operation of the system by so-called *transfer functions* which are independent of the input signal and which characterize the output signal. The equivalent transfer function of a cascade of systems

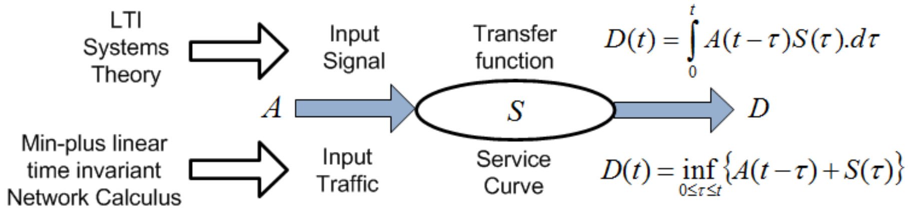


Figure 2.1: Systems theory vs. Network Calculus.

is simply, the convolution of all systems in the cascade. For instance, systems theory is used for circuit design by formulating the transfer functions of the components of a complex circuit and convolving those transfer functions to obtain the end-to-end functionality of the whole circuit.

The transfer function of a LTI system is defined as the output of the system to the Dirac impulse function δ, where

$$\delta(t) = \begin{cases} \text{undefined} & \text{if } t = 0 \\ 0 & \text{if } t \neq 0\,, \end{cases}$$

and $\int_{-\infty}^{\infty} \delta(\tau)d\tau = 1$. The output signal D can be characterized by convolving the transfer function, S and the input signal A, that is $D(t) = \int_{-\infty}^{+\infty} A(t-\tau)S(\tau)d\tau$. A system is called *causal* if $S(t) = 0$ for any $t < 0$. For a causal system the convolution is reduced to $D(t) = \int_0^t A(t-\tau)S(\tau)d\tau$.

Network Calculus shares similar characteristics as systems theory and can be viewed as a systems theory for analyzing networks. The mapping between systems theory and Network Calculus is illustrated in Fig. 2.1. The input signal in the systems theory is replaced by input traffic. There exists a comparable concept to the transfer functions in the Network Calculus called *service curves* which describe the available service at each network element independent of the input. Network Calculus is based on *min-plus algebra*, which substitutes the basic operands in the conventional algebra (+, $\times$) with (min, +). Some of the operations in the conventional algebra can be directly translated into min-plus algebra using the direct substitution of $(+, \times)$ with $(\min, +)$. For example, min-plus convolution of two positive non-decreasing functions f and g can be obtained by the direct substitution from the convolution definition in the conventional algebra. That is, the min-plus convolution of two arbitrary non-negative functions f and g is

$$f * g(t) \triangleq \inf_{0 \leq s \leq t} \Big\{ f(t-s) + g(s) \Big\} . \tag{2.1}$$

Min-plus convolution is commutative and associative. The min-plus deconvolution of f and g is defined by

$$f \oslash g(t) \triangleq \sup_{s \geq 0} \Big\{ f(t+s) - g(s) \Big\} . \tag{2.2}$$

In a *min-plus linear system*, the output traffic is the convolution of the input traffic and the service curve. Suppose that the system responses to arbitrary input traffic flows $X_i(t)$ and $Y_i(t)$ are, respectively, $X_0(t)$ and $Y_0(t)$. The system is called min-plus linear if the system response to $\min\{a + X_i(t), b + Y_i(t)\}$ is $\min\{a + X_0(t), b + Y_0(t)\}$ for any real-value a and b. Fidler and Recker [41] show that Legendre transform serves as the counterpart of Fourier transform in the system theory which provides a dual computational environment and sometimes is easier to work with. In the following, we will elaborate on this analogy and the basics of the Network Calculus.

2.2 Network Model

Suppose that a traffic flow enters a node. The node can be a link with fixed capacity, a packet switch, a router, a modem, etc. We assume a left-continuous arrival process and denote the total arrival from process A in time interval $[0, t)$ by $A(t)$. For the sake of simplicity of notation we define

$$A(s,t) \triangleq A(t) - A(s),$$

for any $0 \leq s \leq t$.

The *backlog* from flow A at a node at time t is defined as the total flow A which has arrived to the node at that time and has not departed the node yet. If D represents the corresponding

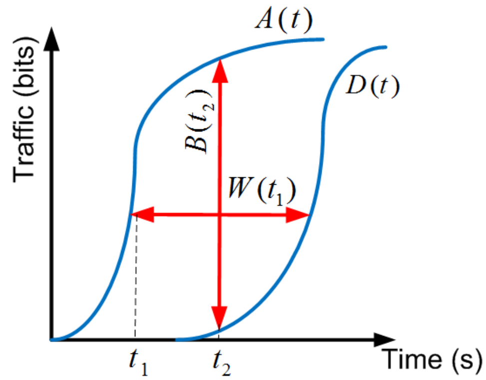


Figure 2.2: Computing delay and backlog visually.

departure process of flow A in that node, then the backlog is represented by

$$B(t) = A(t) - D(t) \ , \tag{2.3}$$

for any $t \geq 0$. The backlog at time t is the vertical difference between arrival A and departure D processes at that time as shown in Fig. 2.2. In addition, the delay at time t is defined by

$$W(t) = \inf\{s \geq 0 \mid A(t) \leq D(t+s)\} \ . \tag{2.4}$$

The delay is the horizontal difference between the arrival A and departure D processes at that time as shown in Fig. 2.2. Eqs. (2.3) and (2.4) are used as the definitions of backlog and delay in this thesis. Network Calculus computes upper bounds for delay and backlog. The bounds are tight for some scenarios.

2.3 Deterministic Network Calculus

The Network Calculus was originally developed as a theory for a worst-case network performance analysis. The Deterministic Network Calculus formulates end-to-end deterministic delay and backlog bounds. These bounds can never be violated in any scenario including the worst-case scenario which leads to the maximum end-to-end delay or backlog.

2.3.1 Deterministic arrival envelopes

The Deterministic Network Calculus characterizes each arrival by an envelope. These envelopes are defined as follows.

Definition 2.1 (Deterministic arrival envelope)**.** *A deterministic arrival envelope E for an arrival process A, must satisfy the following for any $t \geq 0$*

$$\forall \tau \leq t : \quad A(\tau, t) \leq E(t - \tau) \ . \tag{2.5}$$

One of the most frequently used deterministic arrival envelopes is the multi-level leaky bucket envelope. An arrival process is said to be bounded by a K-level leaky bucket with parameters (σ_j, ρ_j) for any $j = 1, \ldots, K$ if

$$E(t) = \min_{1 \leq j \leq K} \{\sigma_j + \rho_j t\} I_{t>0}$$

is a deterministic envelope for the arrival process satisfying Eq. (2.5), and I is the indicator function defined as

$$I_{expr} = \begin{cases} 1 & \text{if } expr \text{ is true} \\ 0 & \text{if } expr \text{ is false} \ , \end{cases} \tag{2.6}$$

where $expr$ is a logical expression, e.g., an inequality. A special case of this envelope is the single leaky bucket envelope for which $K = 1$. A process A is a leaky bucket arrival with parameters (ρ, σ). We write $A \sim (\rho, \sigma)$, if $E(t) = (\sigma + \rho t) I_{t>0}$ is a deterministic arrival envelope for A. So called *shapers* are sometimes used to enforce a deterministic arrival envelope to a traffic flow and such a flow is referred to as *regulated* traffic in the literature.

2.3.2 Deterministic service curves

For the purpose of network performance analysis, it is desirable to describe the operation of the network elements independent of the arrival and characterize the output process as a result. Service curves [85] are analogous concepts to the transfer functions in the systems theory and are defined as follows.

Definition 2.2 (Deterministic service curve)**.** *Assume that A is the arrival process to a node and D is the corresponding departure process. A non-decreasing non-negative function S is a deterministic service curve at that node if*

$$\forall t \geq 0: \qquad A * S(t) \leq D(t) \,, \tag{2.7}$$

where '$*$' in the above equation is the min-plus convolution. Moreover, S is an **exact service curve** *if Eq. (2.7) holds with equality.*

Similar to the transfer functions in the systems theory, the exact service curves in the Network Calculus can be considered as the departure process of the δ_0 function, where

$$\delta_a(t) = \begin{cases} 0 & \text{if } t \leq a \\ \infty & \text{if } t > a \,, \end{cases} \tag{2.8}$$

for any real-value a. This function δ_a is known as the shift function in the Network Calculus since for any non-negative function f and any $t \geq 0$

$$f * \delta_a(t) = f(t-a) \,. \tag{2.9}$$

Another useful property of the δ function is that

$$\delta_b * \delta_a(t) = \delta_{a+b}(t) \,, \tag{2.10}$$

for any value of a and b. Now, let us review some of the most widely used service curves in the literature.

• **Aggregate flow in a general work-conserving scheduler:** An example of an exact service curve is the service curve for a *work-conserving* scheduler with total capacity C. If A is the aggregate of the arrival process to that scheduler, then for any fixed time t, there exists a time $s_0 \leq t$ which is the last time before t that the backlog in the system is zero. That is

$$s_0 = \sup\{s \leq t \mid A(s) = D(s)\} \,, \tag{2.11}$$

Thus, $A(s_0) = D(s_0)$ and since the link is work conserving it is serving arrivals with rate C in $[s_0, t)$

$$D(t) = A(s_0) + C(t - s_0)\ .$$

The above equation proves that $S(t) = Ct$ is a service curve. However, to prove that this is also an exact service curve, we need to show that

$$\inf_{0 \le s \le t} \{A(s) + C(t - s)\} = A(s_0) + C(t - s_0)\ .$$

Contradiction can be employed to prove the correctness of the above equality. Suppose that the above infimum happens at $s_1 \neq s_0$. Then,

$$A(s_1) + C(t - s_1) < A(s_0) + C(t - s_0)\ ,$$

or equivalently

$$\begin{cases} A(s_0, s_1) < C(s_1 - s_0) & \text{if } s_1 > s_0 \\ A(s_1, s_0) > C(s_0 - s_1) & \text{if } s_1 < s_0\ . \end{cases} \tag{2.12}$$

Both inequalities in the above equation contradict the definition of s_0 in Eq. (2.11), and that $s_1 \neq s_0$. The inequality in the first line implies that if backlog is zero at s_0 it is also zero at s_1. The second line inequality implies that the system backlog cannot be zero at time s_0 if the system is causal. Thus, the proof is complete and by Def. 2.2, $S(t) = Ct$ is an exact service curve for a work conserving link with total capacity C.

• **Rate-latency service curves [30]:** Another widely used model for service curves is *rate-latency* service curve. This service curve corresponds to the combination of a work-conserving link with a fixed capacity and a delay element. Assume that C is the capacity of the link and d is the value of the delay. Then, the rate-latency service curve at time t for such a system in the sense of Eq. (2.7) is

$$S(t) = C[t - d]_+,$$

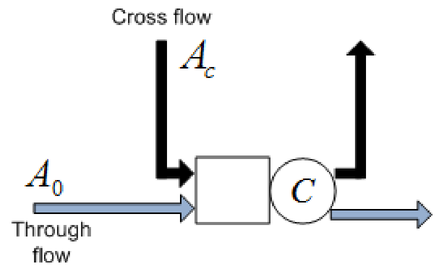


Figure 2.3: A single node with two traffic arrivals.

where for any real-valued x

$$[x]_+ \triangleq max\{0, x\} \ .$$

• **Leftover service curves:** Another service curve we review here is the service curve allocated to a through flow in a link with fixed capacity and shared with cross flows as in Fig. 2.3. It is assumed that all cross flows have higher precedence than the through flow. Thus, the through flow arrival are served only if there is a fraction of the total capacity unused by cross flows. The corresponding service curve is called leftover service curve. The following theorem formulates a leftover service curve.

Theorem 2.1 (Leftover service curve [10])**.** *Suppose that an arrival process A_0 enters a work-conserving link with total capacity C. There is another flow A_c using that node as in Fig. 2.3. If E_c is the deterministic arrival envelope for A_c in the sense of Eq. (2.5), then, a service curve available to A_0 is*

$$S(t) = [Ct - E_c(t)]_+ \ . \tag{2.13}$$

Here we sketch the proof. Assume that D_0 and D_c, respectively represent the corresponding departures of A_0 and A_c. Fix $t \geq 0$ and define s_0 to be the first time before or at time t such that the system backlog is zero (Eq. (2.11)). This implies that

$$D_0(t) \geq A_0(s_0) \ . \tag{2.14}$$

Then, since the link is work conserving, we have

$$D_0(s_0, t) + D_c(s_0, t) = C(t - s_0) \ .$$

Thus,

$$\begin{aligned} D_0(t) &= D_0(s_0) + C(t - s_0) - D_c(s_0, t) \\ &= A(s_0) + C(t - s_0) - D_c(s_0, t) \\ &\geq A(s_0) + [C(t - s_0) - E_c(t - s_0)] \ , \end{aligned} \tag{2.15}$$

where the definition of s_0 from Eq. (2.11) is used in the second line. The last line uses the assumption that E_c is a deterministic arrival envelope for the cross flow. Combining Eqs. (2.14) and (2.15) completes the proof.

2.3.3 Worst-case performance bounds

The Deterministic Network Calculus provides upper bounds on the worst-case backlog and delay of an arrival process to a node if an arrival envelope and a service curve at that node is known. The bounds are presented in the following theorem.

Theorem 2.2 (Deterministic performance bounds [10])**.** *Assume that an arrival process A with a deterministic arrival envelope E satisfying Eq. (2.5) enters a node with a deterministic service curve S, satisfying Eq. (2.7). Suppose that D is the departure process of A. Define the backlog and delay respectively by Eqs. (2.3) and (2.4), then the following upper bounds exists:*

1. **Output burstiness bound:** *The departure at any time interval $[s, t)$ is lower bounded by*

$$D(s, t) \geq E \oslash S(t - s) \ . \tag{2.16}$$

2. **Backlog bound:** *The backlog at any time t is upper bounded by*

$$B(t) \leq E \oslash S(0) \ . \tag{2.17}$$

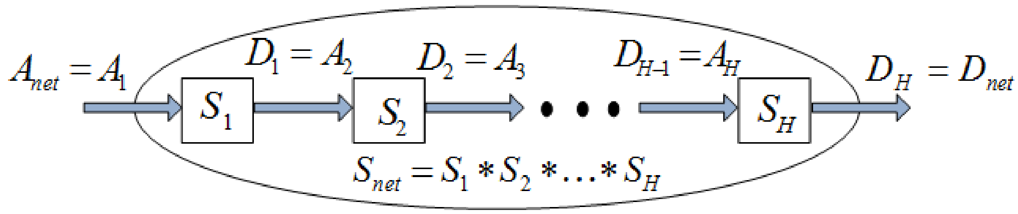


Figure 2.4: Network service curve definition.

3. **Delay bound:** *The delay at any time t is bounded by*

$$W(t) \le \inf\{s \mid \forall t \ge 0 : \quad E(t) \le S(t+s)\} \ . \tag{2.18}$$

These bounds are proved to be tight if E is a sub-additive function, and $E(0) = S(0) = 0$ (see Le Boudec and Thiran [10], pp. 27).

The proof of the bounds in Theorem 2.2 are similar. We sketch the proof of the output burstiness as an example. The total departure in $[s, t)$ can be expressed as

$$\begin{aligned} D(s,t) = D(t) - D(s) &\le A(t) - A * S(s) \\ &= \sup_{0 \le u \le s} \{A(t) - A(u) - S(s-u)\} \\ &\le \sup_{0 \le u \le s} \{E(t-u) - S(s-u)\} \\ &\le \sup_{w \ge 0} \{E(t-s+w) - S(w)\} \\ &= E \oslash S(t-s) \ , \end{aligned}$$

where the first lines uses the definition of the deterministic service curve in Eq. (2.7) and that $A(t) \ge D(t)$ for any $t \ge 0$. The second line applies the definition of min-plus convolution. The third line uses the assumption that E is a deterministic arrival envelope for A and applies the definition from Eq. (2.5). In the next line, a new variable has been introduced $w = s - u$. Finally, the last line uses the definition of the min-plus deconvolution from Eq. (2.2).

2.3.4 Convolution theorem

The elegance of the Network Calculus analysis lies in extending a single node analysis to a multi-node analysis. This is an essential analogy between systems theory and Network Calculus. In systems theory, an equivalent transfer function of multiple systems in tandem is the convolution of the transfer functions of all systems along the path. Similarly, in the Network Calculus, an equivalent end-to-end service curve of a cascade of nodes is the min-plus convolution of the service curves of all nodes along the path. This is expressed in the following theorem.

Theorem 2.3 (Deterministic network service curve)**.** *Consider the scenario depicted in Fig. 2.4. A through flow traverses a cascade of H nodes. Represent the arrival and departure at node h, respectively, by A_h and D_h. Denote the arrival traffic to the first node by A_{net} and the departures from node H by D_{net}. The traffic arrival at node h is the departures from the previous node, i.e., $A_h = D_{h-1}$. At each node h, there exists a service curve function S_h satisfying Eq. (2.7). Then, the convolution of the service curves of all nodes along the path is a network service curve, that is*

$$D_{net}(t) \geq A_{net} * S_{net}(t) ,$$

where

$$S_{net}(t) = S_1 * S_2 * \ldots * S_H(t) ,$$

and $*$ is the min-plus convolution defined in Eq. (2.1).

The proof of the above theorem iterates the definition of the deterministic service curve

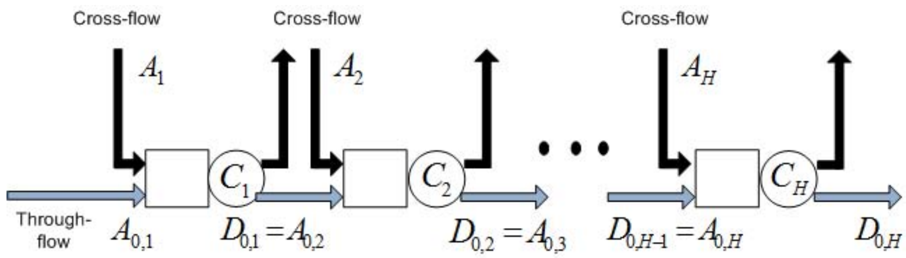


Figure 2.5: A tandem network of schedulers with cross flow.

from Eq. (2.7) as follows:

$$\begin{aligned}
D_{net}(t) &\geq A_H * S_H(t) \\
&\geq (A_{H-1} * S_{H-1}) * S_H(t) \\
&= A_{H-1} * (S_{H-1} * S_H)(t) \\
&\vdots \\
&\geq A_{net} * (S_1 * S_2 * \ldots * S_H)(t) \ ,
\end{aligned}$$

where the first line uses the definition of the service curve at node H. The second line uses $A_H = D_{H-1}$ and replaces D_{H-1} with the lower bound obtained by applying the service curve definition at node $H-1$. The third link uses the associativity of the min-plus convolution. Iterating this method leads to the last line and proves the theorem.

2.3.5 Pay bursts only once

Using the concepts we have reviewed so far, an end-to-end delay bound for the through flow in the scenario illustrated in Fig. 2.5 can be obtained in two different fashions. One way is to obtain an arrival envelope by Theorem 2.2 at any node, and then apply that in Theorem 2.2 once more to attain per-node delay bounds. Adding per-node delay bounds along the path yields an end-to-end delay bound. The other method is to compute a network service curve and then apply the network service curve along with a network arrival envelope to Theorem 2.2. The latter method is the Network Calculus approach and is proved to be a tighter computation. Indeed, the delay induced to a traffic arrival at a certain node consists of two parts: The delay caused by the burstiness of the traffic itself and the delay added as a consequence of non-availability of service at the node. Adding per-node delay bounds overestimates the traffic burstiness delay by considering it at all nodes, while the network service curve counts it only one. This phenomenon is known as *pay burst only once* and makes Network Calculus a promising method. In the following we discuss this phenomenon in detail.

Consider the scenario depicted in Fig. 2.5, where a through flow traverses a path of H nodes. Assume that $E_0(t) = r_0 t + \sigma_0$ is a deterministic arrival envelope for a through flow for some r_0 and σ_0. The total capacity at each node h is C_h, and there exists a cross flow A_h at that node as well.

To simplify notation, consider a homogeneous network, where all nodes are identical. Then, we can replace the subscript h with c for all $h = 1, \ldots, H$. Suppose that the cross flows at each node have the same deterministic arrival envelope $E_c(t) = r_c t + \sigma_c$. Replacing this deterministic envelope in the leftover service curve in Eq. (2.13), a service curve for the through flow at any node $1 \leq h \leq H$ is given by

$$S(t) = [(C - r_c)t - \sigma_c]_+ \ . \tag{2.19}$$

The following are two methods to compute an end-to-end delay bound for the through flow in the above scenario.

• **Using network service curve:** In this method, per-node service curves from Eq. (2.19) are applied to Theorem 2.3 to compute a network service curve. First, the per-node service curve in Eq. (2.19) is reformulated by incorporating the shift functions from Eq. (2.8)

$$\begin{aligned} S_h(t) &= [(C - r_c)(t - \frac{\sigma_c}{C - r_c})]_+ I_{t > \frac{\sigma_c}{C - r_c}} \\ &= [(C - r_c)t] I_{t>0} * \delta_{\frac{\sigma_c}{C - r_c}}(t) \ . \end{aligned} \tag{2.20}$$

Then,

$$\begin{aligned} S_{net}(t) &= S_1 * S_2 * \ldots * S_H(t) \\ &= \bigotimes_{h=1}^{H} \Big[[(C - r_c)t] I_{t>0} * \delta_{\frac{\sigma_c}{C-r_c}} \Big](t) \\ &= [(C - r_c)t] I_{t>0} * \delta_{\frac{H\sigma_c}{C-r_c}}(t) \\ &= [(C - r_c)t - H\sigma_c] I_{t>\frac{H\sigma_c}{C-r_c}} \\ &= [(C - r_c)t - H\sigma_c]_+ \, . \end{aligned} \tag{2.21}$$

where $\bigotimes_{h=1}^{H} f_h = f_1 * f_2 * \ldots * f_H$ for any non-negative functions f_h. Eq. (2.20) is used to obtain the second line. The third line uses the fact that the convolution of rate functions is the minimum rate, and one of the properties of the δ function from Eq. (2.10). The next line rearranges terms using the definition of the δ function and the indicator function.

An end-to-end delay bound can be obtained by applying Eq. (2.21) into Theorem 2.2. That is, $d \geq 0$ is a deterministic delay bound for the scenario depicted in Fig. 2.5 if it is computed as

$$d = \inf\{\tau \mid \forall t \geq 0 : S_{net}(t + \tau) \geq r_0 t + \sigma_0\} \, .$$

By replacing the value of $S_{net}(t)$ from Eq. (2.21), and if the system is stable, i.e., $r_0 + r_c \leq C$, we have

$$d = \frac{\sigma_0 + H\sigma_c}{C - r_c} \, . \tag{2.22}$$

This shows that the delay bound obtained by using the network service curve scales linearly (with $O(H)$) with the number of nodes H on the end-to-end path.

• **Adding per-node delay bounds:** Another method to compute the end-to-end delay bound in the above scenario is by computing per-node delay bounds along the path and sum them up. This can be achieved by computing an upper bound for the departure process at node h which is the arrival process to node $h + 1$. Starting from the first node, an output envelope computed by Theorem 2.2 is as follows:

$$
\begin{aligned}
D_1(s,t) &\leq E_0 \oslash S_1(t-s) \\
&= \sup_{u \geq 0} \Big\{ r_0(t-s+u) + \sigma_0 - [(C-r_c)u - \sigma_c]_+ \Big\} \\
&= r_0(t-s) + \sigma_0 + r_0 \frac{\sigma_c}{C-r_c} \, ,
\end{aligned}
$$

where the last line uses the stability assumption $r_0 + r_c \leq C$. Iterating the above computations, the departures at node h is upper bounded by

$$
D_h(s,t) \leq r_0(t-s) + \sigma_0 + h r_0 \frac{\sigma_c}{C-r_c} \, . \tag{2.23}
$$

The output envelope at node h can be considered as the input envelope to node $h+1$ in a tandem network. Thus, applying D_{h-1} from Eq. (2.23) as an arrival envelope at node h in Theorem 2.2, a delay bound for the through flow at node h (d_h) is computed as

$$
\begin{aligned}
d_h &= \inf \Big\{ \tau \mid \forall t > 0 : S_h(t+\tau) \geq r_0 t + \sigma_0 + (h-1) r_0 \frac{\sigma_c}{C-r_c} \Big\} && (2.24) \\
&= \inf \Big\{ \tau \geq \frac{\sigma_c}{C-r_c} \mid \forall t > 0 : (C-r_c)(t+\tau - \frac{\sigma_c}{C-r_c}) \\
&\qquad\qquad \geq r_0 t + \sigma_0 + (h-1) r_0 \frac{\sigma_c}{C-r_c} \Big\} && (2.25) \\
&= \frac{\sigma_0 + \sigma_c}{C-r_c} + \frac{(h-1) r_0 \sigma_c}{(C-r_c)^2} \, , && (2.26)
\end{aligned}
$$

where in the second line, we inserted the service curve from Eq. (2.20). The last line enforces the inequality inside the bracket in Eq. (2.24) at $t = 0$. Enforcing the inequality at $t = 0$ together with the stability condition $r_0 + r_c \leq C$ provides a sufficient condition for Eq. (2.24). Thus, an end-to-end delay bound would be

$$
\sum_{h=1}^{H} d_h = H \frac{\sigma_0 + \sigma_c}{C-r_c} + \frac{H(H-1) r_0 \sigma_c}{2(C-r_c)^2} \, . \tag{2.27}
$$

The above equation shows that adding per-node delay bound scales quadratically with the number of nodes in the path $O(H^2)$, and thus is not as tight as the delay bound computed by using a network service curve.

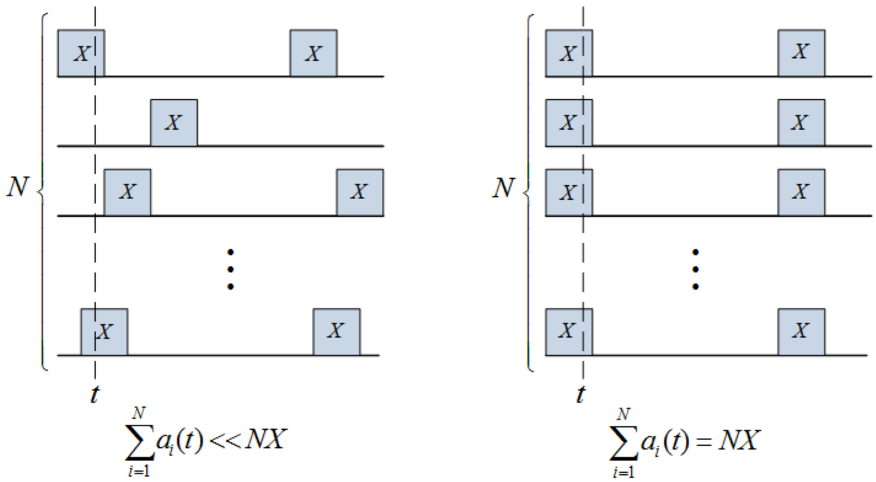


Figure 2.6: Multiplexing N periodic ON-OFF sources, worst case (right hand side) vs. a general case (left hand side).

2.4 Stochastic Network Calculus

The Deterministic Network Calculus is an appropriate theory if worst case scenario matters. However, worst-case scenarios may occur very rarely. As an example, consider an aggregate of N periodic ON-OFF sources as illustrated in Fig. 2.6. Each flow transmits with rate X in On period and is idle in the off period. Let us represent the instantaneous rate of flow i, $1 \leq i \leq N$, at time t by $a_i(t)$. The rate of the aggregate of N flows is NX if all flows are synchronous (which is the worst-case) and it is much smaller if they have independent shifts in time. More precisely, as also illustrated in Fig. 2.6, the maximum aggregate rate in the worst-case scenario is NX while it can be much smaller in practice specially if N is a large number. Accounting for this difference leads to a larger utilization and the corresponding gain is referred to as *statistical multiplexing gain*. The Stochastic Network Calculus is an extension of the Network Calculus which accounts for the statistical multiplexing gain by replacing deterministic bounds with statistical ones.

Suppose that A^N is an aggregate of N independent identically distributed flows enter a link with total capacity C. By Reich's equation, the backlog tail bound can be formulated as the following

$$P\{B(t) > \sigma\} = P\Big\{\sup_{0\le s\le t}\Big(A^N(s,t) - C(t-s)\Big) > \sigma\Big\}\,. \tag{2.28}$$

There are two challenges in evaluating the expression on the right hand side. First, this is the place where the multiplexing gain can be captured by exploiting the independence of different flows. This has been accomplished using different methods in the literature such as Central Limit Theorem [9], [60], [61], Chernoff bound [1], [9], [67] and Hoeffding bound [95]. In this thesis, we apply the Chernoff bound which states that for a random variable X and a constant k, we have

$$P\{X > k\} \le \frac{E[e^{\alpha X}]}{e^{\alpha k}}\,, \tag{2.29}$$

for any $\alpha \ge 0$. We choose Chernoff bound because it provides an upper bound rather than an approximation or asymptotic bounds. Moreover, it uses moment generating functions of the sources and there are many traffic sources for which the exact or upper bounds on the moment generating function are known (even includes heavy-tailed traffic [70]).

The second challenge for evaluating the right hand side of Eq. (2.28) is to compare the probability of a supremum of a set. There are various methods to resolve this difficulty in the literature [62].

1- Using an approximation: For the case of Gaussian arrivals, Knightly and Shroff [62] prove that

$$P\Big\{\sup_{0\le s\le t}\Big(A^N(s,t) - C(t-s)\Big) > \sigma\Big\} \approx \sup_{0\le s\le t} P\Big\{\Big(A^N(s,t) - C(t-s)\Big) > \sigma\Big\}\,. \tag{2.30}$$

The right hand side of the above inequality holds as a lower bound in general. Since this lower bound simplifies the analysis, it is used in the literature for computing queue tail bounds, e.g., [9], [59], [60], [61].

2- Using Martingale bounds: In this method, super martingales and Doobe's inequality are used to compute an upper bound on Eq. (2.28) ([16], pp. 339), [19], [21]. Construct the process

$$Z(s) = A^N(t-s,t) - Cs\,. \tag{2.31}$$

For the filtration of σ-algebra

$$\mathcal{F}_s = \sigma\{A^N(t-s,t)\} \tag{2.32}$$

and independent increment traffic sources, $e^{\alpha Z(s)}$ is a supermartingale for any $\alpha > 0$ [21]. Applying Doobe's inequality to Eq. (2.30) yields

$$\begin{aligned} &P\Big\{\sup_{0\le s\le t}\Big(A^N(t-s,t) - Cs\Big) > \sigma\Big\} \\ &= P\Big\{\sup_{0\le s\le t} e^{\alpha Z(s)} > e^{\alpha\sigma}\Big\} \\ &\le E[e^{\alpha Z(0)}]e^{-\alpha\sigma} \\ &= e^{-\alpha\sigma}\ . \end{aligned}$$

3- Using the union bound: This method provides a valid upper bound for the queue tail probability by applying the union bound as the following:

$$P\Big\{\max_{0\le s\le t}\Big(A^N(s,t) - C(t-s)\Big) > \sigma\Big\} \le \sum_{s=0}^{t} P\Big\{\Big(A^N(s,t) - C(t-s)\Big) > \sigma\Big\}\ .$$

This method has been used extensively in the literature, e.g., [9], [98], [99].

In the rest of this chapter, we focus on the state-of-the-art in the Stochastic Network Calculus.

2.4.1 Statistical arrival envelopes

The deterministic arrival envelope defined in Eq. (2.5) cannot be violated at any s and t, where $0 \le s \le t$. The arrival envelope extensions in the probabilistic regime is relaxed such that they allow a small violation probability. Statistical arrival envelopes have many variations in the literature (see [80]). Bounds are defined either as a sequence of random variables [64], [97], [104], or by deterministic functions [9], [20]. The former bounds are stochastic processes which are stochastically larger than the corresponding traffic processes at any time interval.

Statistical arrival envelopes as deterministic functions have various definitions in the literature [9], [50]. They can be divided into two groups. Some envelopes bound the total arrivals in any time interval of fixed size (which we refer to as *local effective envelopes* [9]). The other group allow the interval to vary such that the past history or arrivals are also taken care of (which we refer to as *statistical sample path envelopes*). These bounds are introduced in the literature under different names. There are effective envelope formulations for different traffic sources in the literature; regulated traffic [9], MPEG video traces [69], Markov-Modulated ON-OFF sources and FBM in [67]. For N independent regulated traffic, the following local effective envelope is formulated in [9] using an upper bound on the moment generating functions and central limit theorem

$$G(t;\sigma) = N\rho t + zt\sqrt{N}\sqrt{RV(t)}\ , \tag{2.33}$$

where $z = \sqrt{|\log(2\pi\varepsilon(\sigma))|}$, $\rho = \lim_{t\to\infty}\frac{A(t)}{t}$, and $RV(t) = Var\left(\frac{A(t)}{t}\right)$. This bound has also been obtained in [61] using the concept of *rate variance envelopes* introduced in [60].

In another work, Li *et al.* [67] extracted a local effective envelope from the effective bandwidth function. If A is an arrival process with effective bandwidth function $Eb_A(\alpha, t)$ in the sense of Eq. (1.3), then the following is a local effective envelope for A

$$G(t;\sigma) = \inf_{\alpha>0}\left\{t\alpha_A(\alpha,t) - \frac{\log\varepsilon(\sigma)}{\alpha}\right\}\ . \tag{2.34}$$

Another important category of envelopes which falls into the definition of the local effective envelopes are Stochastically Bounded Burstiness (SBB) traffic arrivals [93]. SBB traffic model is a more general class of EBB traffic (Eq. (1.4)). A traffic arrival A is SBB if it satisfies the following for any $\sigma \geq 0$ and $s \leq t$

$$P\{A(s,t) > \rho(t-s) + \sigma\} \leq \varepsilon(\sigma)\ , \tag{2.35}$$

where ε is an n-fold integrable function, meaning that

$$\underbrace{\int\int}_{n}\ldots\varepsilon(w)dw < \infty\ .$$

If $\varepsilon(\sigma)$ in Eq. (2.35) is equal to $Me^{-\alpha\sigma}$ for some M and α, then we will have an EBB traffic with parameters (M, ρ, α). SBB models a large class of traffic sources including Markov-Modulated ON-OFF sources and even Gaussian Fractional Brownian Motion (FBM) but it excludes some, e.g., heavy tailed traffic. EBB is a subset of SBB and excludes some SBB traffic sources such as FBM. However, it is yet a large group encompasses important arrivals such as Markov-Modulated ON-OFF sources. In the following we review some examples of EBB sources.

2.4.2 Examples of EBB sources

In the following we introduce two types of EBB sources.

• **Traffic sources with a special effective bandwidth:**

There exists an EBB characterization for a traffic flow A, if its effective bandwidth from Eq. (1.1) satisfies the following:

$$Eb_A(\alpha) = \sup_{t\geq 0} \frac{1}{\alpha t} \sup_{u\geq 0} \log(E[e^{\alpha A(u,u+t)}]) \ . \tag{2.36}$$

For such a case, there exists an EBB characterization for that flow for any $\alpha \geq 0$. This can be verified by employing the Chernoff Bound as follows:

$$\begin{aligned} P\{A(s,t) > Eb_A(\alpha)(t-s)+\sigma\} &\leq \frac{E[e^{\alpha A(s,t)}]}{e^{\alpha(Eb_A(\alpha)(t-s)+\sigma)}} \\ &\leq \frac{e^{\alpha(Eb_A(\alpha)(t-s))}}{e^{\alpha(Eb_A(\alpha)(t-s)+\sigma)}} \\ &= e^{-\alpha\sigma} \ , \end{aligned} \tag{2.37}$$

where in the second line, we use Eq. (2.36). Comparing Eq. (2.37) with the definition of an EBB arrival in Eq. (1.4), shows that A is an EBB arrival with parameters $(1, Eb(\alpha), \alpha)$. There are important traffic sources which fit into this group, e.g., Poisson process and many Markov-Modulated processes.

• Regulated traffic:

As explained in Sec. 2.3.1, any traffic which passes through a shaper is called regulated traffic. One of the most widely used regulated traffic flows are peak-rate constraint leaky bucket regulated traffic flows which have the following deterministic arrival envelope for some P, ρ, and $\sigma \geq 0$,

$$E(t) = \min\{Pt, \rho t + \sigma\} \; . \tag{2.38}$$

Assume that there is an aggregate of N independent peak-rate constraint leaky bucket flows. A local effective envelope for the aggregate flow is computed in [9] as follows. For each flow an upper bound for the moment generating function is computed. These bounds are applied to the Chernoff Bound to yield the following for any $\alpha \geq 0$

$$P\{A(s,t) \geq Nx\} \leq e^{-\alpha N x} \left(1 + \frac{\rho(t-s)}{E(t-s)} \left(e^{\alpha E(t-s)} - 1\right)\right)^N \; . \tag{2.39}$$

From Eq. (1.6), the above equation implies a local effective envelope of the form

$$G(t;x) = N \min\{x, E(t)\}; \qquad \varepsilon(x) = \inf_{\alpha \geq 0} \left\{ e^{-\alpha N x} \left(1 + \frac{\rho t}{E(t)} \left(e^{\alpha E(t)} - 1\right)\right)^N \right\} . \tag{2.40}$$

If the violation probability is fixed to ε^*, then x is the minimum value for which $\varepsilon(x) \leq \varepsilon^*$. For a fixed x, the optimum value of α in the above equation is computed in [9], and it is chosen so that

$$e^{\alpha E(t)} = \frac{x}{\rho t} \frac{E(t) - \rho t}{E(t) - x} \; .$$

We can formulate another local effective envelope for regulated traffic by computing the corresponding EBB characteristics. Choose x in Eq. (2.39) to be $x = \rho'(t-s) + \frac{\sigma}{N}$, for any $\sigma \geq 0$, and fixed ρ', α such that $\rho < \rho' < P$ and $\alpha \geq 0$. Then, we have

$$P(A(s,t) > N\rho'(t-s) + \sigma) \leq e^{-\alpha\sigma} \left[e^{-\alpha\rho'(t-s)} \Big(1 + \frac{\rho(t-s)}{E(t-s)} (e^{\alpha E(t-s)} - 1)\Big) \right]^N \; . \tag{2.41}$$

Compare the above equation with the EBB definition in Eq. (1.4). If there exists a constant M such that

$$\forall t \geq 0 : \qquad \left[e^{-\alpha\rho' t} \Big(1 + \frac{\rho t}{E(t)} (e^{\alpha E(t)} - 1)\Big) \right]^N \leq M \; , \tag{2.42}$$

then from Eqs. (2.41) and (2.42), $(M, N\rho', \alpha)$ is an EBB characterization for the aggregate flow. Note that since $\lim_{t\to\infty} \frac{E(t)}{t} = \rho < \rho'$, M in the above inequality is bounded in t. Hence, we choose M to be

$$M = \sup_{t\geq 0} \left[e^{-\alpha\rho' t}\Big(1 + \frac{\rho t}{E(t)}(e^{\alpha E(t)} - 1)\Big)\right]^N . \tag{2.43}$$

If we fix ρ' and α such that $\rho < \rho' < P$ and $\alpha \geq 0$, and compute M from Eq. (2.43), then we can construct another local effective envelope for a peak rate constraint leaky bucket for any $\sigma \geq 0$ as follows

$$G(t;\sigma) = N\rho' t + \sigma; \qquad \varepsilon(\sigma) = Me^{-\alpha\sigma} . \tag{2.44}$$

The above envelope is valid for any choice of ρ' and α satisfying their constraints. Thus, we can obtain a new local effective envelope by varying ρ' and α. If $G_1, G_2, \ldots, G_k$ are local effective envelopes for the same traffic process with bounding functions $\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_k$, then, at any time t, $\min_{1\leq i\leq k} G_i(t)$ is also a local effective envelope for that process with bounding function $\varepsilon = \max_{1\leq i\leq k} \varepsilon_i$. Using this fact, we can improve the local effective bound in Eq. (2.44) as follows:

$$\mathcal{G}(t;\sigma) = \min_{\rho',\alpha}\{N\rho' t + \sigma\}; \quad \varepsilon(\sigma) = \max_{\rho',\alpha}\{Me^{-\alpha\sigma}\} . \tag{2.45}$$

In the rest of this section, we study some technical issues on computing the above envelopes. First, we decrease the time interval we need to consider to compute M from Eq. (2.43). To find the time index in which the supremum in Eq. (2.43) is happening at, we can disregard the power N. We aim to find t_{max} which maximizes the expression inside the bracket in Eq. (2.43). Using the above simplification, t_{max} can also be obtained from the following:

$$t_{max} = \arg\sup_{t\geq 0}\Big\{ \Big(1 - \frac{\rho t}{E(t)}\Big)e^{-\alpha\rho' t} + \frac{\rho t}{E(t)} e^{\alpha(E(t)-\rho' t)}\Big\} . \tag{2.46}$$

We search for the supremum index separately for two subintervals.

Case 1 ($t \leq \frac{\sigma}{P-\rho}$): In this case, $E(t) = Pt$. Thus, the expression inside the bracket in Eq. (2.46) is simplified to

$$X := \sup_{t\geq 0}\Big\{ \Big(1 - \frac{\rho}{P}\Big)e^{-\alpha\rho' t} + \frac{\rho}{P} e^{\alpha(P-\rho')t}\Big\} . \tag{2.47}$$

Taking the second derivative of X from the above equation, we have

$$\frac{d^2X}{dt^2} = \alpha^2\rho'^2(1-\frac{\rho}{P})e^{-\alpha\rho' t} + \alpha^2(P-\rho')^2 e^{\alpha(P-\rho')t} ,$$

which is always positive (since $\rho', \rho < P$), showing that X is a convex function. Therefore, the supremum value in $[0, \frac{\sigma}{P-\rho}]$ can occur either at $t = 0$, or $t = \frac{\sigma}{P-\rho}$.

Case 2 ($t \geq \frac{\sigma}{P-\rho}$): In this range, $E(t) = \sigma + \rho t$. Thus, the expression in the bracket in Eq. (2.46) is simplified to

$$Y := \sup_{t\geq 0}\Big\{\Big(\frac{\sigma}{\rho t+\sigma}\Big)e^{-\alpha\rho' t} + \Big(\frac{\rho t}{\rho t+\sigma}\Big)e^{\alpha((\rho-\rho')t+\sigma)}\Big\} . \tag{2.48}$$

Taking the first derivative of the above expression, we have

$$\begin{aligned}\frac{dY}{dt} = \frac{1}{(\rho t+\sigma)^2}\Big[&\big(-\sigma\rho - \alpha\rho'\sigma(\rho t+\sigma)\big)e^{-\alpha\rho' t} \\ &+ \big(\sigma\rho - \alpha\rho t(\rho'-\rho)(\rho t+\sigma)\big)e^{\alpha((\rho-\rho')t+\sigma)}\Big] .\end{aligned} \tag{2.49}$$

The first term in the derivative is always negative. The second term is decreasing in t. Define t_{max} to be the time when the second term turns negative, which is

$$\begin{aligned} t_{ub} &= \inf\{u \mid \sigma\rho - \alpha\rho u(\rho'-\rho)(\rho u+\sigma) < 0\} \\ &= \frac{\alpha\sigma(\rho'-\rho) + \sqrt{(\alpha\sigma(\rho'-\rho))^2 + 4\alpha\rho\sigma(\rho'-\rho)}}{2\alpha\rho(\rho'-\rho)} ,\end{aligned}$$

where in the second line we took the maximum root of the inequality in the first line. Then, it is guaranteed that for any $t > t_{ub}$ the derivative $\frac{dY}{dt}$ is negative. Combining the above results in Case 1 and Case 2, with Eq. (2.46), we have that $t_{max} = 0$ (for which $M = 1$), or $t_{max} \in [\frac{\sigma}{P-\rho}, t_{ub}]$. Hence,

$$M = \max\Big\{1, \sup_{\frac{\sigma}{P-\rho}\leq t\leq t_{ub}}\Big\{e^{-N\alpha\rho' t}\Big(1 + \frac{\rho t}{E(t)}\big(e^{\alpha E(t)} - 1\big)\Big)^N\Big\}\Big\} . \tag{2.50}$$

The second technical issue on the numerical computations is how to improve the local effective envelope we can get for regulated traffic by EBB characterization with a fixed violation probability ε^*. We use the following algorithm to achieve this goal.

Algorithm 1 Computing local effective envelopes for regulate traffic flows using EBB characteristics

Input N

$\varepsilon^* \leftarrow 10^{-6}$

for $t = 1$ to 200 **do**

  for $\alpha = 10^{-10} : 0.01 : 10$ **do**

    for all ρ' such that $\rho < \rho' < P$ **do**

      Compute $M(\rho', \alpha)$ from Eq. (2.50)

      Compute σ by setting $\varepsilon(\sigma) = \varepsilon^*$ from Eq. (2.44)

      Compute $G(t; \alpha, \rho') = \rho' t + \frac{\sigma}{N}$.

    end for

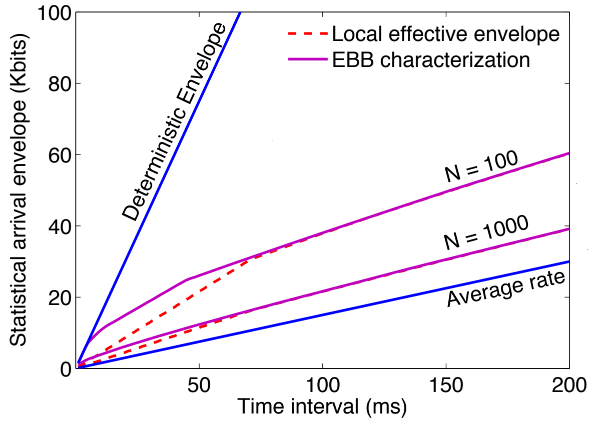
  end for

  return $G(t) = \min_{\alpha,\gamma} G(t; \alpha, \rho')$.

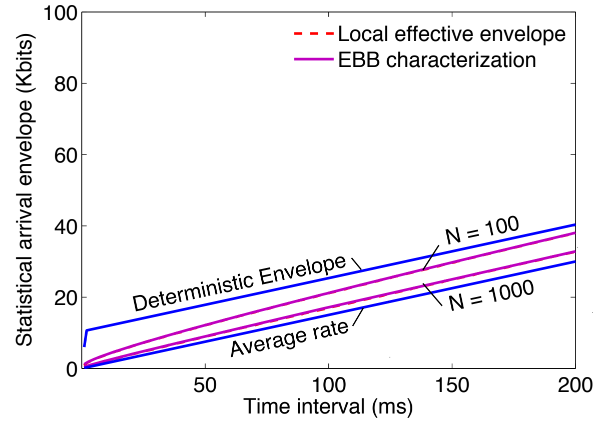
end for

Table 2.1: Regulated traffic parameters

	P (Mbps)	ρ (Mbps)	σ (Kb)
Type 1	1.5	0.15	95.4
Type 2	6	0.15	10.345



(a) Type 1 flows



(b) Type 2 flows

Figure 2.7: Two different arrival envelopes for $N = 100$ regulated flows with parameters as shown in Table 2.1, and $\varepsilon^* = 10^{-6}$.

It is not clear whether characterizing an arrival process by two parameters (burstiness and long-term average rate) as EBB is an accurate characterization of that arrival. For instance, although regulated traffic is categorized as an EBB process as shown above, there is some inaccuracy induced by characterizing a regulated traffic with EBB parameters. We examine it with numerical examples.

Suppose there are two types of peak-rate constraint leaky bucket traffic flows with parameters as in Table 2.1 which are also used in [9]. We want to compute a local effective envelope for an aggregate of N independent flows with violation probability $\varepsilon^* = 10^{-6}$. For each type of traffic, we compare the local effective envelope from Eq. (2.40) with the corresponding envelope obtained by EBB characterization described above. We plot the normalized per-flow envelopes, $\frac{G(t;\varepsilon^*)}{N}$,which makes the multiplexing gain analysis easier. However, Figs. 2.7a and 2.7b show that this is not the case at least for peak-rate constraint leaky buckets, and the resulting envelope from EBB is very close to that of Eq. (2.40), especially if the leaky-bucket burstiness is not large (as in Type 2 flow). The local effective envelopes are plotted for $N = 100, 1000$. As shown in the plots, the envelopes capture the multiplexing gain by comparing the local effective envelope, and EBB envelopes with the deterministic envelope (which cannot capture the multiplexing gain and serves as an upper bound-bench mark), and the average rate (which assumes a constant bit rate arrival with the long-term average rate.

2.4.3 Sample path arrival envelopes

To formulate probabilistic performance bounds a more stringent envelope which bounds the whole past history of arrivals is required. Such an envelope is called statistical sample path envelope and defined in the following.

Definition 2.3 (Statistical sample path arrival envelope)**.** *For an arrival process A, a non-decreasing non-negative function $\mathcal{G}$ is a statistical arrival envelope for A if for any $t > 0$ it satisfies*

$$P\Big\{\sup_{s\le t}\Big(A(s,t) - \mathcal{G}(t-s;\sigma)\Big) > 0\Big\} \le \varepsilon(\sigma)\ , \tag{2.51}$$

for any $\sigma \ge 0$.

A special case of this definition is gSBB [100] which extends the concept of SBB to statistical sample path envelops. A process A is gSBB with parameters ρ and bounding function ε if it satisfies the following for any $\sigma \ge 0$ and any time $t \ge 0$

$$P\{\sup_{0\le s\le t}(A(s,t) - \rho(t-s)) > \sigma\} < \varepsilon(\sigma)\ , \tag{2.52}$$

where ε is a non-increasing function on $[0,\infty)$, such that $1-\varepsilon$ is a distribution function. This restricts the bounding function to be always smaller than 1. gSBB includes some traffic sources which are not SBB including non-Gaussian self-similar processes such as α-stable sources.

While gSBB is inherently a statistical sample path envelope, statistical sample path envelopes can be extracted from local effective envelopes using different techniques. We review the methods in the following

1. *EBB assumption [20]:* If A is EBB with parameters (M,ρ,α) then, for any $\gamma > 0$, there is a statistical sample path envelope satisfying Eq. (2.51) given by

$$\mathcal{G}(t;\sigma) = (\rho+\gamma)t + \sigma, \qquad \varepsilon(\sigma) = Me\Big(1+\frac{\rho}{\gamma}\Big)e^{-\alpha\sigma}\ . \tag{2.53}$$

This is proved by discretizing time and applying Boole's inequality for the violation probabilities at each time unit. Here we sketch the proof. Let us assume that the time unit is τ_0. Then, for any $0 \le s \le t$, there exists a unique $i \ge 1$ satisfying

$$(i-1)\tau_0 \le t - s < i\tau_0 \; . \tag{2.54}$$

Since A and $\mathcal{G}$ are non-decreasing functions of time, the following holds

$$A(s,t) - \mathcal{G}(t-s;\sigma) \le A(t - i\tau_0, t) - \mathcal{G}((i-1)\tau_0;\sigma) \; . \tag{2.55}$$

The proof also uses an inequality which is based on the Riemann integral and holds for any $a, b \ge 0$, and non-increasing function ε

$$\sum_{j=1}^{\infty} \varepsilon(a + bj) \le \frac{1}{b} \int_a^{\infty} \varepsilon(u) du \; . \tag{2.56}$$

Starting from sample path arrival envelope definition, we have

$$\begin{aligned}
& P\Big\{ \sup_{0 \le s \le t} \{A(s,t) - \mathcal{G}(t-s;\sigma)\} > 0 \Big\} && (2.57)\\
& \quad \le P\Big\{ \max_{1 \le j \le \lceil \frac{t-s}{\tau_0} \rceil} \{A(t - j\tau_0, t) - \mathcal{G}((j-1)\tau_0;\sigma)\} > 0 \Big\} && (2.58)\\
& \quad \le P\{ \max_{1 \le j \le \lceil \frac{t-s}{\tau_0} \rceil} \{A(t - j\tau_0, t) - (\rho+\gamma)(j-1)\tau_0 - \sigma\} > 0 \} && (2.59)\\
& \quad \le \sum_{1 \le j \le \lceil \frac{t-s}{\tau_0} \rceil} P\{A(t - j\tau_0, t) > (\rho+\gamma)(j-1)\tau_0 + \sigma\} && (2.60)\\
& \quad \le \sum_{j=1}^{\infty} \varepsilon(j\gamma\tau_0 + \sigma - (\rho+\gamma)\tau_0) && (2.61)\\
& \quad \le \frac{M}{\alpha\gamma\tau_0} e^{-\alpha(\sigma - (\rho+\gamma)\tau_0)} \; , && (2.62)
\end{aligned}$$

where the second line uses Eq. (2.55). The third line replaces the assumed value of the statistical arrival envelope from Eq. (2.53). Eq. (2.60) applies Boole's inequality, and Eq. (2.61) extends the boundaries of the summation to infinity. ε in that equation is the bounding function of an EBB traffic from Eq. (1.4). Finally, the last line uses Eq. (2.56). Minimizing the last line for τ_0 completes the proof.

2. *Special Violation Probability [67]:* If the violation probability of the local effective envelope is defined as a special function of time then the following sample path envelope exists in the sense of Eq. (2.67). This function is defined in discrete time setting. Thus, we need to first discretize time as in Eq. (2.54). Then, define the local effective envelope violation probability at time t by

$$\varepsilon_G(t) = \frac{2\varepsilon}{\pi(1 + \lceil \frac{t}{\tau_0} \rceil^2)} , \tag{2.63}$$

for some $0 < \varepsilon < 1$ and $\tau_0 > 0$. Then the following is a sample path arrival envelope

$$\mathcal{G}(t;\sigma) = G(t + \tau_0;\sigma) , \qquad \varepsilon_{\mathcal{G}}(\sigma) = \varepsilon . \tag{2.64}$$

It can be proved as follows.

$$\begin{aligned}
&P\Big\{ \sup_{0 \le s \le t} \{A(s,t) - \mathcal{G}(t-s;\sigma)\} > 0 \Big\} \\
&\quad \le P\Big\{ \max_{1 \le j \le \lceil \frac{t-s}{\tau_0} \rceil} \{A(t - j\tau_0, t) - \mathcal{G}((j-1)\tau_0;\sigma)\} > 0 \Big\} \\
&\quad \le \sum_{j=1}^{\infty} P\{A(t - j\tau_0, t) > G(j\tau_0;\sigma)\} \\
&\quad \le \sum_{j=1}^{\infty} \frac{2\varepsilon}{\pi(1+j^2)} \\
&\quad \le \varepsilon ,
\end{aligned}$$

where the second line uses Eq. (2.55). The third line replaces $\mathcal{G}$ from Eq. (2.64). The last line uses the inequality $\sum_{j=1}^{\infty} \frac{1}{1+j^2} \le \frac{\pi}{2}$.

3. *Time Scale Limit [67]:* A time scale limit is an upper bound over the maximum time that the events in the network are correlated. For example, an upper bound over the maximum busy period length is a time scale limit. A probabilistic upper bound on the busy period length is computed in [67] as follows. Assume that multiple traffic flows with aggregate statistical sample path envelope $\mathcal{G}_{tot}$ with bounding function ε_b (from Eq. (2.64)) enter a link with capacity C. The maximum busy period in the system is statistically bounded

by

$$T^{\varepsilon_b} = \sup\{t \geq 0 \mid \mathcal{G}_{tot}(t) > Ct\} \ . \tag{2.65}$$

In the sense that at any time $t \geq 0$ if $\hat{x}^t$ is the start of the busy period containing t, then

$$P\{t - \hat{x}^t > T^{\varepsilon_b}\} < \varepsilon_b \ . \tag{2.66}$$

The concept of time scale limit can be used to construct other sample path envelopes. Since it limits the interval that the sample path must be valid. If G is a local effective envelope for A with bounding function ε_G, then a sample path envelope (in the sense of Eq. (2.51)) can be obtained as follows

$$\mathcal{G}(t;\sigma) = G(t+\tau_0;\sigma) \ , \qquad \varepsilon_{\mathcal{G}}(\sigma) = \left\lceil \frac{T^{\varepsilon_b}}{\tau_0} \right\rceil \varepsilon_G(\sigma) + \varepsilon_b \ , \tag{2.67}$$

for any $\tau_0 > 0$. To verify that, let τ_0 be the time unit and define i as introduced in Eq. (2.54). Then

$$\begin{aligned}
&P\Big\{ \sup_{0\leq s\leq t} \{A(s,t) - \mathcal{G}(t-s;\sigma)\} > 0 \Big\} \\
&\quad \leq P\Big\{ \Big(\sup_{0\leq s\leq t} \{A(s,t) - \mathcal{G}(t-s;\sigma)\} > 0 \Big) \cap (t - \hat{x}^t < T^{\varepsilon_b}) \Big\} \\
&\qquad\qquad + P\Big\{ \Big(\sup_{0\leq s\leq t} \{A(s,t) - \mathcal{G}(t-s;\sigma)\} > 0 \Big) \cap (t - \hat{x}^t \geq T^{\varepsilon_b}) \Big\} \\
&\quad \leq P\Big\{ \sup_{t-T^{\varepsilon_b}\leq s\leq t} \{A(s,t) - \mathcal{G}(t-s;\sigma)\} > 0 \Big\} + \varepsilon_b \\
&\quad \leq P\Big\{ \max_{1\leq j\leq \lceil \frac{T^{\varepsilon_b}}{\tau_0} \rceil} \{A(t-j\tau_0,t) - G((j-1)\tau_0 + \tau_0;\sigma)\} > 0 \Big\} + \varepsilon_b \\
&\quad \leq \sum_{j=1}^{\lceil \frac{T^{\varepsilon_b}}{\tau_0} \rceil} P\left\{ A(t-j\tau_0,t) > G(j\tau_0;\sigma) \right\} + \varepsilon_b \\
&\quad \leq \left\lceil \frac{T^{\varepsilon_b}}{\tau_0} \right\rceil \varepsilon_G(\sigma) + \varepsilon_b \ ,
\end{aligned}$$

where in the second line we use the total probability theorem. In the next line we use the assumption that the busy period is bounded by T^{ε_b}, and the fact that $P(X \cap Y) \leq P(X)$ for any X and Y. We also use the fact that the statistical sample path envelope does not need to take care of intervals larger than the time scale limit. The fifth line uses Eq. (2.55) and Eq. (2.67). By applying Boole's inequality in the next line and Eq. (2.67) in the last line the proof is complete.

4. *Global Effective Envelope [9], [74]:* A global effective envelope for an arrival process A, is a non-decreasing function of time $\mathcal{G}$ defined in any time interval of length ℓ ($[u, u+\ell]$), if it satisfies the following

$$P\Big\{\sup_{u\leq s\leq t\leq u+\ell} (\, A(t-s,t) - \mathcal{G}(s;\ell)) > 0\Big\} \leq \varepsilon_{\mathcal{G}}(\ell), \tag{2.68}$$

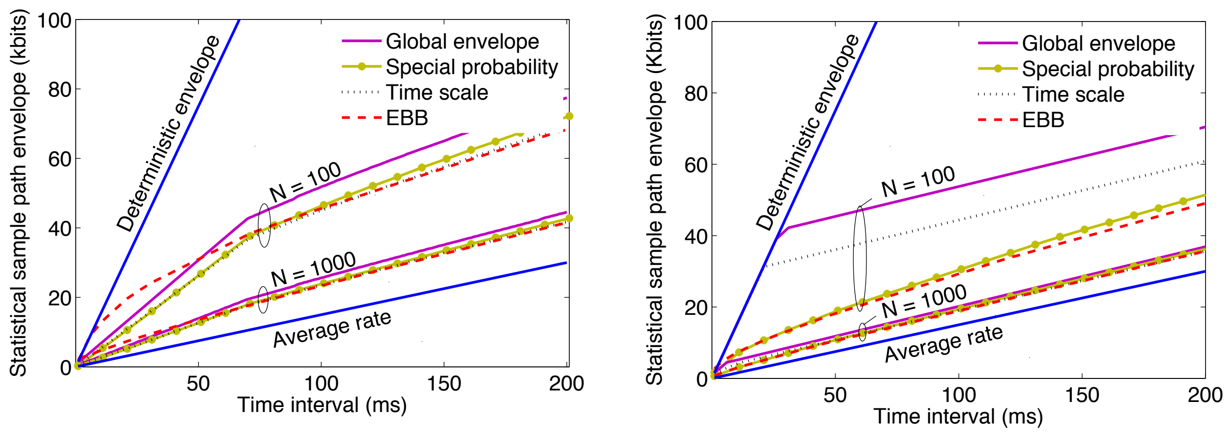
where $0 < \varepsilon < 1$, and ℓ is the size of time interval that this envelope is valid in. Comparing Eq. (2.51) with Eq. (2.68), it can be verified that the global effective envelope is a more stringent definition than the sample path statistical arrival envelope. As derived in [74], the global effective envelope can be expressed using a local effective envelope. If G and ε_G are respectively the local effective envelope and its bounding function A, then the following describe a statistical sample path envelope for that process:

$$\mathcal{G}(s;\ell) = G(\gamma s + a)\ , \qquad \varepsilon_{\mathcal{G}}(\ell) = \varepsilon_G \cdot \frac{\ell}{a}\frac{\sqrt{\gamma+1}}{\sqrt{\gamma}-1}\ ,$$

where $\gamma > 1$, and $a \approx \sqrt{\gamma}(\gamma-1)s$.

We compare the above statistical sample path envelope construction methods for two types of traffic sources in numerical examples. Before starting the numerical results, we point out some remarks on numerical computations.

First, to obtain each sample path envelope, there might be some free parameters. For instance, EBB sample path envelopes for regulated traffic as described in this section have three free parameters: γ, α, ρ'. We can take the point-wise minimum of all sample paths by varying the free parameters. The bounding function of the resulting sample path is the sum of the bounding functions of all considered sample path envelopes, or $\sum_{\alpha,\gamma,\rho'} \varepsilon(\sigma)$. By increasing the set we choose our sample path from, we increase the chance to get a smaller point-wise



(a) Regulated traffic

(b) Discrete-time MMP traffic

Figure 2.8: Sample path envelope comparisons for $N = 100, 1000$ regulated or discrete time MMP flows with $\varepsilon^* = 10^{-6}$.

minimum but at the same time we are increasing the equivalent violation probability. Thus, we have to choose the range of α, γ, ρ' just large enough to include critical choices.

Second, to compute γ for an EBB sample path characterization of regulated traffic if the sample path bounding function ε^* is fixed. From Eq. (2.53), we have

$$\sigma = -\frac{1}{\alpha}\left(\log \frac{\varepsilon^*}{Me} - \log\left(1 + \frac{\rho}{\gamma}\right)\right) . \tag{2.69}$$

Inserting this value in the EBB sample path bound formulation in Eq. (2.53) yields

$$\mathcal{G}(t;\sigma) = (\rho + \gamma)t - \frac{1}{\alpha}\left(\log \frac{\varepsilon^*}{Me} - \log\left(1 + \frac{\rho}{\gamma}\right)\right) . \tag{2.70}$$

Taking the first derivative of the above equation with respect to γ and setting it to zero gives us

$$\gamma_{opt} = \frac{-\alpha\rho t + \sqrt{(\alpha\rho t)^2 + 4\alpha\rho t}}{2\alpha t} . \tag{2.71}$$

Algorithm 2 explains the steps we follow to compute a statistical sample path envelope for regulated traffic using EBB characteristics. Note that in this algorithm, we compute a sample path in discrete time for the time interval $t = 1 \sim 200$. Thus, the resulting point-wise minimum sample path envelope can be at most from 200 sample paths regardless of how large the set of different choices for γ, α, ρ' are. This justifies $\varepsilon^* = \frac{10^{-6}}{200}$, in the first line of the algorithm if the sample path envelope violation probability is supposed to be 10^{-6}.

Algorithm 2 Computing statistical sample path envelope for regulated traffic flows using EBB characteristics

Input N

$\varepsilon^* \leftarrow \frac{10^{-6}}{200}$

for $t = 1$ to 200 **do**

  for $\alpha = 10^{-10} : 0.01 : 10$ **do**

    for all ρ' such that $\rho < \rho' < P$ **do**

      Compute $M(\rho', \alpha)$ from Eq. (2.50)

      Compute γ_{opt} from Eq. (2.69)

      Compute σ by setting $\varepsilon(\sigma) = \varepsilon^*$ from Eq. (2.69)

      Compute $\mathcal{G}(t; \alpha, \rho') = \rho' t + \frac{\gamma_{opt} t + \sigma}{N}$.

    end for

  end for

  return $\mathcal{G}(t) = \min_{\alpha,\gamma} \mathcal{G}(t; \alpha, \rho')$.

end for

With these remarks, we can now compare the sample path envelopes. We have N independent flows with the same characteristics. For each type of flow we compute the normalized sample path envelopes $\frac{\mathcal{G}(t,\varepsilon^*)}{N}$ using all methods we reviewed, where the violation probability is fixed to $\varepsilon^* = 10^{-6}$. For the methods which are based on a busy period bound (time scale limit sample path, and global effective envelope), we assume that the traffic enters a link in which it creates a link utilization of 90%, i.e., $C = \frac{10}{9} N \rho_{ave}$, where ρ_{ave} is the long-term average rate of the aggregate flows. This assumption is applied to Eq. (2.65) to obtain a time sale limit. In case there is no deterministic arrival envelope, we choose $\varepsilon_b = \varepsilon^*$.

We consider two traffic types for our numerical examples. The first traffic source is Type 1 peak-rate constraint leaky bucket flow from Table 2.1. In Fig. 2.8a we plotted the per-flow

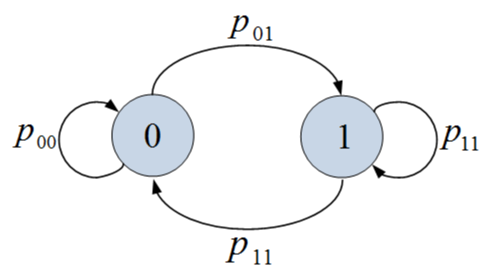


Figure 2.9: Discrete time Markov-Modulated Process (MMP).

statistical envelopes constructed by each method for the aggregate traffic. The plot shows that the sample path envelopes do not considerably differ from each other. The global effective envelope is slightly looser than the other two bounds. This stems from the fact that global effective envelope has a more stringent definition than other envelopes as discussed before. Another interesting point is the capability of capturing the multiplexing gain in these bounds which is noticeable by comparing the bounds for $N = 100$, with those of $N = 1000$.

Markov-Modulated Process (MMP) (See Fig. 2.9). At each time slot, the MMP source is either at state '0' which is the idle state, or at state '1' which generates traffic with rate P. The transition between states can happen at the start of each time slot. The transition probability of $0 \to 1$ and $1 \to 0$ are, respectively, $1 - p_{00}$ and $1 - p_{11}$, as shown in Fig. 2.9. A bound on the effective bandwidth for such a flow is [16]

$$Eb(\alpha) \leq \frac{1}{\alpha} \log(p_{00} + p_{11}e^{\alpha P} + \sqrt{(p_{00} + p_{11}e^{\alpha P})^2 - 4(p_{00} + p_{11} - 1)e^{\alpha P}}) \;, \tag{2.72}$$

for any $\alpha \geq 0$. Thus, there exists an EBB characterization for discrete time MMP traffic from Eq. (2.37). The comparison of sample path bounds in Fig. 2.8b shows that since there is no deterministic time scale bound in this case, the time scale limit sample path is not among the best envelopes anymore. In both plots we see that there is no considerable difference between different sample path descriptions.

2.4.4 Statistical service curves

There are different definitions for a statistical service curve in the literature. One of the important categorization might be representing the service curve in the stochastic regime by either a deterministic function such as [15], or a stochastic process as in [97]. In this thesis, we use a deterministic function service curve description which is simpler to manipulate. We use the following definition for the stochastic service curve.

Definition 2.4 (Statistical service curve)**.** *Given an arrival process A at a node and a departure process D. A non-decreasing non-negative function $\mathcal{S}$ is a statistical service curve for process A with a non-increasing bounding function ε, if it satisfies the following at any time $t \geq 0$*

$$P\{D(t) > A * \mathcal{S}(t;\sigma)\} < \varepsilon(\sigma) \ , \tag{2.73}$$

for any $\sigma \geq 0$.

As an example of a statistical envelope, consider the following.

Theorem 2.4. *Assume that a through flow enters a node with the total capacity C. There is also a cross flow at that node. If $\mathcal{G}_c$ is a statistical sample path envelope for the cross flow arrival in the sense of Eq. (2.51) with bounding function ε_c, then for any $\sigma \geq 0$, the following is a statistical service curve for the through flow at that node in the sense of Eq. (2.73)*

$$\mathcal{S}(t) = [Ct - \mathcal{G}_c(t;\sigma)]_+ \ , \tag{2.74}$$

with bounding function $\varepsilon_c(\sigma)$.

The proof is similar to the proof of Theorem 2.1 except that the arrival envelope in this setting is not deterministic and can be violated with some violation probability. This turns the deterministic service curve to a statistical one in the sense of Def. 2.4. The bounding function of the corresponding statistical service curve is equal to the bounding function of the cross flow statistical sample path envelope [20].

2.4.5 From fluid flow model to packet model

In this thesis, we use a fluid flow model. The extension of the analysis can be obtained in the expense of adding notation and by using the method introduced in [14]. In a packet model, packet transmission cannot be interrupted even if packets with higher precedences arrive. The following theorem establishes the bridge between fluid and packet model.

Theorem 2.5. *Assume that a through flow enters a node with statistical service curve $\mathcal{S}^f$ in the sense of Eq. (2.73) with bounding function ε_s. There are also some cross flows in that node. The packet size of the through flow and cross flows are, respectively, represented by P_0 and P_c with the following probabilistic bounds for any $\sigma \geq 0$*

$$P\{P_0 > \sigma\} \leq \varepsilon_0^p(\sigma) \tag{2.75}$$

$$P\{P_c > \sigma\} \leq \varepsilon_c^p(\sigma) \; . \tag{2.76}$$

For any σ^f, σ_0^p, $\sigma_0^p \geq 0$, define $\underline{\sigma} = (\sigma^f, \sigma_0^p, \sigma_c^p)$. Then, $\mathcal{S}^p$ from the following is a service curve for the through flow in the packet model

$$\mathcal{S}^p(t; \underline{\sigma}) = [\mathcal{S}^f(t; \sigma^f) - \sigma_0^p - \sigma_c^p]_+ \tag{2.77}$$

with bounding function

$$\varepsilon_s^p(\underline{\sigma}) = \varepsilon_s^f(\sigma_s) + \frac{1}{E[P_c]} \int_{\sigma_c^p}^{\infty} \varepsilon_c^p(u) du + \frac{1}{E[P_0]} \int_{\sigma_0^p}^{\infty} \varepsilon_0^p(u) du \; . \tag{2.78}$$

Proof. If D_0^f and D_0^p are, respectively, the through flow departure process in the fluid flow and packet model, then

$$D_0^f(t) = D_0^p(t) + \tilde{Z}_0(t) + \tilde{Z}_c(t) \; , \tag{2.79}$$

where $\tilde{Z}_0(t)$ is the portion of the through flow packet that is being served and has already been completed. $\tilde{Z}_c(t)$ is the portion of the cross flow packet which is being served and has already been completed and would not be served in a fluid model by time t. This cross flow packet has a lower precedence than an existing through flow backlogged packet, thus, will not be served in a fluid model. By conditioning on the through flow packet size and from [14]

$$P\{\tilde{Z}_0(t) > \sigma\} \leq \frac{1}{E[P_0]} \int_{\sigma}^{\infty} \varepsilon_0^p(u)du \ . \tag{2.80}$$

Similar bound exists for $\tilde{Z}_c$. Combining all of the above results,

$$P\{D_0^p(t) < A_0 * [\mathcal{S}(t;\sigma) - \sigma_0^p - \sigma_c^p]_+\} \tag{2.81}$$

$$\leq P\{D_0^f(t) < A_0 * \mathcal{S}(t;\sigma)\} + P\{\tilde{Z}_0(t) > \sigma_0^p\} + P\{\tilde{Z}_c(t) > \sigma_c^p\} \tag{2.82}$$

$$\leq \varepsilon_s^f(\sigma_s) + \frac{1}{E[P_c]} \int_{\sigma_c^p}^{\infty} \varepsilon_c^p(u)du + \frac{1}{E[P_0]} \int_{\sigma_0^p}^{\infty} \varepsilon_0^p(u)du \ . \tag{2.83}$$

□

2.4.6 Statistical performance bounds

Having a statistical arrival envelope and a statistical service curve, probabilistic upper bounds for the output burstiness, backlog, and delay have been computed in [20].

Theorem 2.6 ([20]). *Assume that an arrival process A with statistical sample path arrival envelope $\mathcal{G}$ satisfying Eq. (2.5) with bounding function ε_g enters a node with a statistical service curve $\mathcal{S}$ satisfying Eq. (2.7) with bounding function ε_s. Suppose D is the departure process from the node, and the backlog and delay at that node are as defined in Eqs. (2.3) and (2.4). Also define $\varepsilon(\sigma_g, \sigma_s)$ as*

$$\varepsilon(\sigma_g, \sigma_s) = \varepsilon_g(\sigma_g) + \varepsilon_s(\sigma_s) \ , \tag{2.84}$$

for any σ_g, $\sigma_s \geq 0$. Then, the following probabilistic bounds exists for any arbitrary σ_g, $\sigma_s \geq 0$

1. **Output burstiness bound:** *The departure at any time interval $[s, t)$ is bounded by*

$$P\{D(s,t) \geq \mathcal{G} \oslash \mathcal{S}(t-s;\sigma_g,\sigma_s)\} \leq \varepsilon(\sigma_g,\sigma_s) \ . \tag{2.85}$$

2. **Backlog bound:** *The backlog at any time t is bounded by*

$$P\{B(t) \leq \mathcal{G} \oslash \mathcal{S}(0;\sigma_g,\sigma_s)\} \leq \varepsilon(\sigma_g,\sigma_s) \ . \tag{2.86}$$

3. **Delay bound:** *The delay at any time t is bounded by*

$$P\Big\{W(t) \leq \inf\{s \mid \forall u \geq 0: \ \mathcal{S}(u+s;\sigma_s) \geq \mathcal{G}(u;\sigma_g)\}\Big\} \leq \varepsilon(\sigma_g,\sigma_s)\ . \tag{2.87}$$

The proofs of the above bounds are similar. Here we only sketch the proof of the output burstiness from [20]. Fix, s and t, such that $s \leq t$. Suppose that we want to compute a lower bound for $D(s,t)$. Assume that u_0 is the time that the infimum value of $A * \mathcal{S}(s)$ is happening at, that is

$$A(s-u_0) + \mathcal{S}(u_0) = A * \mathcal{S}(s)\ . \tag{2.88}$$

Then, we have

$$P\Big\{D(s,t) > \sup_{u\geq 0}\{\mathcal{G}(t-s+u;\sigma_g) - \mathcal{S}(u;\sigma_s)\}\Big\} \tag{2.89}$$

$$\leq P\{D(s,t) > \mathcal{G}(t-s+u_0;\sigma_g) - \mathcal{S}(u_0;\sigma_s)\} \tag{2.90}$$

$$\begin{aligned}&= P\Big\{\Big(D(s,t) > \mathcal{G}(t-s+u_0;\sigma_g) - \mathcal{S}(u_0;\sigma_s)\Big) \cap \Big(\sup_{u\leq t}(A(u,t) - \mathcal{G}(t-u;\sigma_g)) \leq 0\Big)\Big\}\\ &\quad + P\Big\{\Big(D(s,t) > \mathcal{G}(t-s+u_0;\sigma_g) - \mathcal{S}(u_0;\sigma_s)\Big) \cap \Big(\sup_{u\leq t}(A(u,t) - \mathcal{G}(t-u;\sigma_g)) > 0\Big)\Big\}\end{aligned} \tag{2.91}$$

$$\begin{aligned}&\leq P\{D(t) - D(s) > A(t) - A(s-u_0) - \mathcal{S}(u_0;\sigma_s)\}\\ &\qquad + P\{\Big(\sup_{u\leq t}(A(u,t) - \mathcal{G}(t-u;\sigma_g)) > 0\Big)\}\end{aligned} \tag{2.92}$$

$$\leq P\{D(s) < A(s-u_0) + \mathcal{S}(u_0;\sigma_s)\} + P\Big\{\Big(\sup_{u\leq t}(A(u,t) - \mathcal{G}(t-u;\sigma_g)) > 0\Big)\Big\} \tag{2.93}$$

$$\leq \varepsilon_s(\sigma_s) + \varepsilon_g(\sigma_g)\ , \tag{2.94}$$

where the second line uses the fact that the inequality inside the bracket of the first line must hold for any $u \geq 0$. The third line uses the total probability theorem. Eq. (2.92) is obtained by using the fact that $P(U \cap W) \leq P(W)$ for events U and W. Eq. (2.93) is obtained by using $D(t) \leq A(t)$ at any time $t \geq 0$. Finally, the last line uses the definition of the service curve in Eq. (2.73) and statistical arrival envelope in Eq. (2.51).

2.4.7 Statistical convolution theorem

In order to extend the probabilistic performance bound analysis to multiple-node scenarios, an end-to-end statistical service curve is required. Statistical network service curve is not the convolution of the per-node statistical service curves along the path as in the deterministic case. To clarify, the additional complexity with respect to the deterministic case, consider a tandem network consisting of node 1 and node 2. The arrival traffic to node 2 is the departure traffic from node 1. $\mathcal{S}_1$ and $\mathcal{S}_2$, are respectively, statistical service curves for node 1 and node 2 in the sense of Def. 2.4 with bounding functions ε_1 and ε_2. Then, the departure at the second node satisfies the following

$$D_2(t) \geq \inf_{0\leq s\leq t}\{A_2(s) + \mathcal{S}_2(t-s)\} \ . \tag{2.95}$$

To compute an end-to-end service curve, the relation between A_2 and $\mathcal{S}_1$ must be incorporated in the above equation which is

$$\forall 0 \leq s \leq t : \quad A_2(s) \geq \inf_{0\leq u\leq s}\{A_1(s) + \mathcal{S}_1(t-s)\} \ .$$

However, by the definition of the statistical service curve, ε_1 is only bounding the violation probability of the above inequality for a fixed time interval (i.e., fixed s in the above equation) and not for the entire past history of the departures. The following theorem resolves this problem and formulates a statistical network service curve.

Theorem 2.7 (Statistical network service curve [20]). *Assume that a traffic flow traverses a path consisting of H nodes. Denote the arrival and departure to node h, respectively, by A_h and D_h. The through flow departure at node h in the path is the traffic flow arrival to node $h+1$, that is, $D_h = A_{h+1}$. At each node h, there exists a function $\mathcal{S}_h$ that serves as a statistical service curve in the sense of Def. 2.4. This service curve function is in the form of $\mathcal{S}_h(t;a) = \mathcal{S}_h(t) - a$ for any $a > 0$ and with bounding function ε_h satisfying $\int_{-\infty}^{\infty} \varepsilon_h(w)dw < \infty$. Then, for any choice of $\gamma_h, \tau_h \geq 0$, $h = 1, \ldots, H-1$, the following is a statistical network service curve*

$$\mathcal{S}_{net}(t;\sigma) = \Big(\mathcal{S}_1 * \mathcal{S}_2^{\gamma_1} * \ldots * \mathcal{S}_H^{\sum_{h=1}^{H-1}\gamma_h}(t - \sum_{h=1}^{H-1}\tau_h)\Big) - \sum_{h=1}^{H-1}\gamma_h\tau_h - \sigma \ , \tag{2.96}$$

where $\mathcal{S}_h^{h\gamma}(t) = \mathcal{S}_h(t) - h\gamma t$*, in the sense that*

$$P\{D_H(t) > A_1 * \mathcal{S}_{net}(t;\sigma)\} \leq \varepsilon_{net}(\sigma) ,$$

and

$$\varepsilon_{net}(\sigma) = \inf_{\sum_{h=1}^{H}\sigma_h=\sigma} \Big\{ \varepsilon_H(\sigma_H) + \sum_{h=1}^{H-1} \frac{1}{\gamma_h \tau_h} \int_{\sigma_h}^{\infty} \varepsilon_h(w) dw \Big\} . \tag{2.97}$$

Here we sketch the proof from [20]. Fix time t, then for any fixed $\tau_h > 0$ and $u \leq t$ there exists an $i \geq 1$ such that

$$(i-1)\tau_h \leq t - u < i\tau_h .$$

By the monotonicity of A_h, D_h, and $\mathcal{S}_h$, we have

$$D_h(u) \geq D_h(t - i\tau_h); \quad A_h * \mathcal{S}_h(u - \tau_h) \leq A_h * \mathcal{S}_h(t - i\tau_h) . \tag{2.98}$$

Using the above results, we can obtain an upper bound for the bounding function of the following expression which will be needed for the proof.

$$\begin{aligned}
&P\Big\{ \inf_{0\leq u\leq t} \{D_h(u) - A_h * \mathcal{S}_h(u-\tau_h) + \sigma_h + \gamma_h(t-u+\tau_h)\} < 0 \Big\} \\
&\qquad \leq P\Big\{ \min_{1\leq j\leq \lceil \frac{t}{\tau_h} \rceil} \{D_h(t-j\tau_h) - A_h * \mathcal{S}_h(t-j\tau_h) + \sigma_h + j\gamma_h\tau_h\} < 0 \Big\} \\
&\qquad \leq \sum_{j=1}^{\infty} \varepsilon_h(\sigma_h + j\gamma_h\tau_h) \\
&\qquad \leq \frac{1}{\gamma_h\tau_h} \int_{\sigma_h}^{\infty} \varepsilon_h(w) dw ,
\end{aligned} \tag{2.99}$$

where the second line uses Eq. (2.98) and the monotonicity of the consisting functions. The third line uses the definition of the statistical service curve and the union bound. The last line uses Riemann sums.

To simplify notation, define

$$\sigma_h' = \sigma_h + \gamma_h\tau_h . \tag{2.100}$$

Also

$$X_h = \inf_{0 \le u \le t} \{D_h(u) - A_h * \mathcal{S}_h(u - \tau_h; \sigma_h) + \gamma(t - u + \tau_h)\} < 0 \, .$$

Note that we used $\mathcal{S}(t;\sigma)$ to mean $\mathcal{S}(t) - \sigma$ in the above formulation (and for the rest of the proof as well) to simplify notation. Denote the compliment of the event X_h by X_h^c. Using these notations, the departure process at node H is characterized by

$$\begin{aligned}
&P\{D_H(t) \ge \inf_{0 \le u \le t}\{A_H(u) + \mathcal{S}_H(t-u;\sigma_H)\}\} \\
&\quad = P\Big\{\Big(D_H(t) \ge \inf_{0 \le u \le t}\{A_H(u) + \mathcal{S}_H(t-u;\sigma_H)\}\Big) \cap X_h^c\Big\} \\
&\qquad\qquad + P\Big\{\Big(D_H(t) \ge \inf_{0 \le u \le t}\{A_H(u) + \mathcal{S}_H(t-u;\sigma_H)\}\Big) \cap X_h\Big\} \\
&\quad \le P\Big\{\Big(D_H(t) \ge \inf_{0 \le u \le t}\{A_H(u) + \mathcal{S}_H(t-u;\sigma_H)\}\Big) \cap X_h^c\Big\} \\
&\qquad\qquad + P\{X_h\} + P\{D_H(t) \ge \inf_{0 \le u \le t}\{A_H(u) + \mathcal{S}_H(t-u;\sigma_H)\}\} \\
&\quad \le P\{D_H(t) \ge \inf_{0 \le s \le u \le t}\{A_{H-1}(s) + \mathcal{S}_{H-1}(u-s;\sigma'_{H-1}) \\
&\qquad\qquad - \gamma(t-u) + \mathcal{S}_H(t-u;\sigma_H)\} + \frac{1}{\gamma_{H-1}\tau_{H-1}}\int_{\sigma'_{H-1}}^{\infty} \varepsilon_{H-1}(w)dw + \varepsilon_H(\sigma_H) \\
&\quad = P\{D_H(t) \ge \inf_{0 \le s \le u \le t}\{A_{H-1}(s) + \mathcal{S}_{H-1}(u-s-\tau_{H-1};\sigma'_{H-1}) + \mathcal{S}_H^{-\gamma_{H-1}}(t-u;\sigma_H)\} \\
&\qquad\qquad + \frac{1}{\gamma_{H-1}\tau_{H-1}}\int_{\sigma_{H-1}}^{\infty} \varepsilon_{H-1}(w)dw + \varepsilon_H(\sigma_H) \\
&\quad = P\{D_H(t) \ge A_{H-1} * (\mathcal{S}_{H-1} * \mathcal{S}_H^{-\gamma_{H-1}})(t;\sigma'_{H-1},\sigma_H))\} \\
&\qquad\qquad + \frac{1}{\gamma_{H-1}\tau_{H-1}}\int_{\sigma_{H-1}}^{\infty} \varepsilon_{H-1}(w)dw + \varepsilon_H(\sigma_H) \, ,
\end{aligned}$$

where the second line uses the total probability theorem. The next line uses the fact that for two events W and U, $P(W \cap U) \le P(W) + P(U)$. The next line combines the events in the first probability term, uses notational simplification from Eq. (2.100), and also uses the statistical service curve definition at node H, along with Eq. (2.99). The last two lines rearrange terms and use the convolution definition.

Iterating the above method backwards to the first node proves the theorem. Note that for each node $1 \le h \le H-1$, Eq. (2.99) must hold. Adding up all corresponding bounding functions from Eq. (2.99) proves the validity of Eq. (2.102).

Remark: A network service curve can be implied in the discrete time model from Theorem 2.7. By setting τ_h, $\gamma_h = 0$ for any $h = 1, \ldots, H-1$,

$$\mathcal{S}_{net}(t;\sigma) = \Big(\mathcal{S}_1 * \mathcal{S}_2^{\gamma_1} * \ldots * \mathcal{S}_H^{\sum_{h=1}^{H-1}\gamma_h}(t)\Big) - \sigma \ , \tag{2.101}$$

and

$$\varepsilon_{net}(\sigma) = \inf_{\sum_{h=1}^{H}\sigma_h=\sigma} \Big\{\varepsilon_H(\sigma_H) + \sum_{h=1}^{H-1}\sum_{k=0}^{\infty}\varepsilon_h(\sigma_h + k\gamma)\Big\} \ . \tag{2.102}$$

To compute the infimum value in Eq. (2.97), a lemma is provided in [20] which can be used when the bounding function decays exponentially fast as for EBB traffic sources.

Lemma 2.1 ([20])**.** *For any positive α_h, M_h, and any σ_h ($h = 1, \ldots H$)*

$$\inf_{\sum_{h=1}^{H}\sigma_h=\sigma} \sum_{h=1}^{H} M_h e^{-\alpha_h\sigma_h} = \prod_{h=1}^{H}(M_h\alpha_h\nu)^{\frac{1}{\alpha_h\nu}} e^{-\frac{\sigma}{\nu}} \ ,$$

where $\nu = \sum_{h=1}^{H}\frac{1}{\alpha_h}$.

As an example of a probabilistic network service curve, consider the homogeneous network in Fig. 2.5. We replace the subscript h with c for any $1 \le h \le H$. Assume that cross flow at each node is an EBB traffic with parameters (M_c, ρ_c, α_c). Thus, from Theorem 2.4, a per-node statistical service curve for the cross flow for any $\sigma_c \ge 0$ is

$$\mathcal{S}_c(t;\sigma_c) = [(C - \rho_c - \gamma_c)t - \sigma_c]_+ \ ,$$

for any $\gamma_c \ge 0$ with bounding function $\varepsilon_c(\sigma_c) = M_c e(1 + \frac{\rho_c}{\gamma_c})e^{-\alpha_c\sigma_c}$. Applying this per-node service curve to Theorem 2.7, an end-to-end probabilistic network service curve is

$$\mathcal{S}_{net}(t;\sigma) = (C - \rho_c - \gamma_c)t - (H-1)\gamma_c\tau_c - \sigma \ , \tag{2.103}$$

with bounding function

$$\varepsilon_{net}(\sigma) = \inf_{\sum_{h=1}^{H}\sigma^h=\sigma} \Big\{M_c e(1 + \frac{\rho_c}{\gamma_c})e^{-\alpha_c\sigma_c} + \frac{(H-1)}{\gamma_c\tau_c\alpha_c} M_c e(1 + \frac{\rho_c}{\gamma_c})e^{-\alpha_c\sigma_c}\Big\} \tag{2.104}$$

$$= M_c e(1 + \frac{\rho_c}{\gamma_c})\left(\frac{H}{\gamma_c\alpha_c\tau_c}\right)^{\frac{H-1}{H}} e^{-\frac{\alpha_c}{H}\sigma} \ , \tag{2.105}$$

where Lemma 2.1 is used to obtain the last line.

Chapter 3

Single Node Analysis

In this chapter we compute single node delay bounds for a large class of schedulers (which we call Δ-schedulers) using Network Calculus. The main difficulty in attaining tight delay bounds is to formulate a tight service curve description for the schedulers.

We begin this chapter by reviewing the main analytical approaches in queueing analysis to date. Then, we introduce the class of schedulers we consider in this thesis. We formulate a service curve applicable to any scheduler in that class. Using this service curve we compute delay bounds and show that this bound is necessary and sufficient. As two examples, we specify our results for leaky bucket arrivals in the deterministic regime and EBB arrivals in the probabilistic one. The probabilistic bounds are examined by computing the schedulable region for two types of EBB sources (regulated traffic, and discrete time MMP traffic). The results of this chapter were developed in joint works with Liebeherr and Burchard (see [72], [73]).

3.1 Literature Review

The first packet scheduling analyses were in the context of Queueing Theory [46], [57]. The best results in Queueing Theory correspond to $M/M/1$ queues. Relaxing the assumption of Poisson arrivals, only few results remain. The average delay expression in a $M/G/1$ queue and in the presence of some schedulers including FIFO and SP is formulated and named after its creators Pollaczek-Khinchine [57]. Finally, the most important result in $G/G/1$ queues is the average delay computation in the heavy traffic asymptotic regime [46].

3.1.1 Envelope-based scheduling analysis

We call all analytical results which are based on upper bounds on the traffic arrivals, or computing upper delay and backlog bounds rather than computing the exact distributions as *envelope based* methods. This type of analysis is much broader than Queuing Theory and extends the queue analysis to a broad range of traffic sources (e.g., regulated traffic [9], EBB [98], SBB [93], gSBB [50], [100] heavy tailed self similar processes [70]). The literature on this method is extensive and includes many schedulers, e.g., SP [9], [50], [60], FIFO [9], [17], [21], [36], [50], EDF [1], [9], [87], [90], and GPS [50], [79]. We discuss them in detail in the following.

Elwalid *et al.* [37] assume a fixed buffer size and compute the loss probability (buffer overflow probability) for two classes of peak rate constraint leaky bucket arrivals in a FIFO link. (For each class j of arrivals, the per-flow required bandwidth e_0^j to exhaust the link capacity C and buffer size at the same time is formulated.) Multiplexing a number of N_j flows from class j, a loss probability is defined as the probability that the total instantaneous demand bandwidth exceeds the total capacity, or

$$P\{B > b\} = P\Big\{\sum_{j=1}^{2}\sum_{k=1}^{N_j} \xi_k^j e_0^j(b) > C\Big\} ,$$

where ξ_k^j is an indicator function which is set to one if flow k from class j is backlogged in the steady state. The above probability is then computed using a large deviation method derived in [83]. This method was extended to a two-class priority scheduler in [35] by characterizing the highest priority (class 1) output process and noting that the lowest priority class (class 2) can only use the leftover capacity of the link. The highest priority output *lossless effective bandwidth* R_1 is formulated using Markov state probabilities for Markov modulated ON-OFF sources. Then, the loss probability of the lower priority class is

$$P\{B_2 > b\} = P\Big\{R_1 + \sum_{k=1}^{N_2} \xi_k^2 e_0^2(b) > C\Big\} .$$

In another work, Elwalid and Mitra [36] compute per-flow loss probability for a two-class peak rate constraint leaky buckets in a GPS scheduler. Given a delay bound d_j^* for each class j, the per-flow lossless effective bandwidth required bandwidth e_0^j for class j to satisfy its delay bound is computed. Then, the loss probability for each flow from class 1 is the following

$$P(B_1 > b) = P\left\{ e_0^1 > \frac{\phi_1 C}{\phi_1(1 + \sum_{k=1}^{N_1-1} \xi_k^1) + \phi_2 \sum_{k=1}^{N_2} \xi_k^2} \right\} ,$$

where ϕ_j is the GPS weight for class j. Large deviation results in [83] are used to compute the above probability.

The deterministic schedulable region in a multi-class arrivals in FIFO, SP, or EDF schedulers is computed in [75]. Assume that there is a tagged arrival from flow i at time t. Denote $A_j^{i,t}(t_1, t_2)$ the arrivals from flow j in $[t_1, t_2)$ which must be served before the tagged arrival. Define $\hat{x}_i^t$ to be the last time before or at time t, when there is no traffic in the queue that must be served before the tagged arrival. It can be expressed by the following

$$\hat{x}_i^t = \sup\left\{ s \le t \mid \forall j \in \mathcal{N} : A_j^{i,t}(s) \le D_j(s) \right\} ,$$

where $\mathcal{N}$ is the set of all arrivals. The above definition implies that $A_j(\hat{x}_i^t) = D_j(\hat{x}_i^t)$ for any $j \in \mathcal{N}$. A sufficient condition that the tagged arrival leaves the system before its delay bound has expired (i.e., before $t + d_i^*$) is that the workload ahead of the tagged arrival at time $t + d_i^*$ is negative. That is,

$$\sum_{j \in \mathcal{N}} A_j(\hat{x}_i^t, t + \tau_{i,j}) - C(t + d_i^* - \hat{x}_i^t) \le 0 , \tag{3.1}$$

where $\tau_{i,j}$ depends on the type of scheduler; $\tau_{i,j} = 0$ for FIFO, $\tau_{i,j} = \max\{\hat{x}_i^t - t, d_i^* - d_j^*\}$ for EDF, and $\tau_{i,j} = d_i^*$ in SP if j has higher priority than i, and $\tau_{i,j} = \hat{x}_i^t - t$, otherwise. Thus, a sufficient condition that flow i delay bound is never violated at any time is obtained by employing the deterministic arrival envelopes in Eq. (3.1), which yields

$$\sup_{t\geq 0}\Big(\sum_{j\in\mathcal{N}} E_j(t+\tau_{i,j}) - C(t+d_i^*)\Big) \leq 0 \ .$$

In case E_j for any $j \in \mathcal{N}$ is a concave function, the above condition is also shown to be a necessary condition meaning that if it does not hold, there exists a scenario in which, the achievable delay violates the delay constraints [75]. The extension of this work to the statistical regime is considered in [9]. If each flow j has a local effective envelope G_j with bounding function ε_j^G, Eq. (3.1) yields a statistical delay constraint for any $\sigma_j \geq 0$ as follows:

$$\sup_{t\geq 0}\Big(\sum_{j\in\mathcal{N}} G_j(t+\tau_{i,j};\sigma_j) - C(t+d_i^*)\Big) \leq 0 \ . \tag{3.2}$$

Two methods are used in [9] to limit the violation probability of Eq. (3.2). The first method is to use the lower bound approximation in Eq. (2.30) which yields a bounding function of size $\sum_{j\in\mathcal{N}} \varepsilon_j^G(\sigma_j)$. The second method is to replace G_j with a global effective envelope (Eq.(2.68)) $\mathcal{G}_j$ with bounding function $\varepsilon_j^{\mathcal{G}}$ for any flow j in Eq. (3.2). Then, Eq. (3.2) provides a statistical schedulable region constraint with bounding function $\sum_{j\in\mathcal{N}} \varepsilon_j^{\mathcal{G}}(\sigma_j)$. The statistical multiplexing gain is captured in [9] by employing central limit theorem and Chernoff bound.

Sivaraman and Chiussi [89] consider a set $|\mathcal{N}|$ of traffic arrivals A_i with a priori delay bounds d_i^* for any $i \in \mathcal{N}$. The delay constraints are sorted as $d_1^* \leq \ldots \leq d_{|\mathcal{N}|}^*$. It is assumed that $d_{|\mathcal{N}|}^* - d_1^*$ is small compared to $d_{|\mathcal{N}|}^*$ and there exists a stationary backlog distribution B. With the above assumptions, a loss probability which is the probability that a delay bound is violated is formulated as follows:

$$P(loss) \approx P\{B + \sum_{i=1}^{|\mathcal{N}|} A(d_{|\mathcal{N}|}^* - d_i^*) > C d_{|\mathcal{N}|}^*\} \ .$$

The above equation is bounded by using the stationary backlog tail bound for ON-OFF Markov Modulated processes from Eq. (1.2) with $K = 1$.

Ciucu and Liebeherr [21] obtain backlog and output burstiness bounds when there is a through flow and a cross flow, respectively, represented by index 0 and c in a FIFO scheduler.

To do so, the following property of FIFO schedulers is used that for any $\sigma_j \geq 0$

$$B_0(t) > \sigma \Rightarrow \{\exists s \leq t : A_0(s,t) > \sigma \text{ and } B_{tot}(s) \geq C(t-s)\} \ , \tag{3.3}$$

where $B_{tot} = B_0 + B_s$. The first inequality in the bracket is bounded by sample path arrival envelopes. The second inequality is bounded by using Doobe's inequality.

Jiang and Liu (see [50] pp. 150-155) formulate per-flow statistical violation probabilities for gSBB traffic sources in a general work-conserving scheduler (with one flow), FIFO, SP, or GPS. Consider a discrete time model and a tagged arrival from process A at time t. The delay W that this virtual arrival experiences exceeds d if the backlog is positive in $[t, t+d-1]$. That is

$$P\{W(t) \geq d\} \leq P\{B^C(t+s) > 0; s = 0, 1, \ldots, d-1\} \ , \tag{3.4}$$

where $B^C(t)$ represents Reich's backlog equation with arrival process A in a link with capacity C, i.e.,

$$B^C(t) = \sup_{0 \leq s \leq t} \Big(A(s,t) - C(t-s) \Big) \ . \tag{3.5}$$

With this notation and assuming that A is a gSBB traffic in the sense of Eq. (2.52) with rate ρ and bounding function ε, then

$$P\{B^\rho(t) > \sigma\} \leq \varepsilon(\sigma) \ . \tag{3.6}$$

By showing that

$$B^\rho(t) \geq B^C(t) + (C - \rho) \tag{3.7}$$

and employing induction, it is proved that a sufficient condition to have the left hand side event in Eq. (3.4) is to have

$$B^\rho(t+d-1) \geq B^C(t+d-1) + (C-\rho)d \ . \tag{3.8}$$

Combining all above, yields

$$P\{W(t) \geq d\} \leq \varepsilon((C-\rho)d) . \tag{3.9}$$

For a single flow in a FIFO scheduler, a statistical delay bound is obtained by using a property of FIFO that $W(t) \geq d$ is equivalent to $B^C(t) > C(d-1)$, and consequently

$$\begin{aligned} P\{W(t) \geq d\} &= P\{B^C(t) > C(d-1)\} \\ &\leq P\{B^\rho(t) > Cd - \rho\} \\ &\leq \varepsilon(Cd-\rho) , \end{aligned} \tag{3.10}$$

where the second line uses Eq. (3.7).

For SP, it is assumed that there are P classes of arrivals. The arrival process A_p of any class $p \leq P$ is a gSBB traffic with rate ρ_p and bounding function ε_p. The highest priority class (class 1) can be viewed as a single flow in a FIFO link with the capacity C. Thus, the delay bound for the highest priority class is obtained from Eq. (3.10) to be

$$P\{W_1(t) \geq d\} \leq \varepsilon_1(Cd-\rho_1) . \tag{3.11}$$

For the other classes, $p > 1$, a general scheduler can be assumed with aggregate flow of arrivals from class 1 to p, i.e., $\sum_{1\leq k\leq p} A_k$. Thus, from Eq. (3.9)

$$P\{W_p(t) \geq d\} \leq \varepsilon_{1\ldots p}((C - \sum_{1\leq k\leq p} \rho_k)d) , \tag{3.12}$$

where $\varepsilon_{1\ldots p}(x) = \varepsilon_1 * \ldots * \varepsilon_p(x)$.

For GPS, there exists a guaranteed service $\frac{\phi_i C}{\sum_j \phi_j}$ for each flow i. Thus, each flow i can be considered as a single flow in a FIFO link with the guaranteed rate. Hence, from Eq. (3.10)

$$P\{W_i(t) \geq d\} \leq \varepsilon_i\left(\frac{\phi_i Cd}{\sum_j \phi_j} - \rho_i\right) . \tag{3.13}$$

By the evolution of Network Calculus, service curves are introduced as a tool for scheduling analysis. Computing delay and backlog bounds using service curves has an advantage over any other envelope based analyses. It can be extended to multi-node analysis, or more complicated per-flow scenarios much easier than others.

Network Calculus can be used for any system for which we can compute a service curve. Thus, scheduling analysis in the context of Network Calculus requires a per-flow service curve formulation.

3.1.2 Using service curves for scheduling analysis

This method is inspired by Cruz in [27]. Given a service curve and an arrival envelope for a traffic process in a link, Theorem 2.6 can be applied to compute performance bounds for that arrival. There are some service curve formulations for the schedulers in the literature which we review in the sequel.

Characterizing a per-flow service curve requires a per-flow output characterization (see Eq. (2.7)) as a function of the capacity, scheduler parameters and input processes of other flows. We review the departure characterization for the case of SP from [10] in the following. Define $\hat{x}_p^t$ to be the last time before time t that there is no backlogged arrival from classes 1 to p. That is

$$\hat{x}_p^t = \sup\left\{s \le t \mid \forall j \le p : A_j(s) \le D_j(s)\right\} \ . \tag{3.14}$$

Then,

$$\begin{aligned} D_p(t) &= A_p(\hat{x}_p^t) + \sum_{j=1}^{p}(D_j(t) - D_j(\hat{x}_p^t)) - \sum_{j=1}^{p-1}(D_j(t) - D_j(\hat{x}_p^t)) \\ &\ge A_p(\hat{x}_p^t) + [C(t-\hat{x}_p^t) - \sum_{j=1}^{p-1}(A_j(t) - A_j(\hat{x}_p^t))]_+ \\ &\ge A_p(\hat{x}_p^t) + [C(t-\hat{x}_p^t) - \sum_{j=1}^{p-1} E_j(t-\hat{x}_p^t)]_+ \ , \end{aligned} \tag{3.15}$$

where the first line uses the fact that $A_p(\hat{x}_p^t) = D_p(\hat{x}_p^t)$. The second line uses the fact that the link is busy in $[\hat{x}_p^t, t)$ and the fact that $D_p(t) \ge A_p(\hat{x}_p^t)$. The third line assumes that E_j

is a deterministic arrival envelope for class j. Comparing Eq. (3.15) with the service curve definition in Eq. (2.7), a deterministic service curve for class p arrivals is

$$S_p(t) = \Big[Ct - \sum_{1 \le j \le p-1} E_j(t)\Big]_+ . \tag{3.16}$$

For FIFO schedulers, Cruz [28] derived the following service curve

$$S_i(t) = \Big[Ct - \sum_{j \in \mathcal{N} \setminus \{i\}} E_j(t-\theta)\Big]_+ I_{t>\theta} , \tag{3.17}$$

where $\theta \ge 0$ is a free parameter. This service curve can be considered as a family of service curves which is valid for any non-negative θ. The special case of $\theta = 0$ yields the service curve for the lowest priority in a SP scheduler (S_P in Eq. (3.16)).

Finally, service curve description for EDF schedulers has also been explored in the literature, e.g., [67], [97]. The following is the per-flow service curve formulation for EDF schedulers in [67]

$$S_i(t) = \Big[Ct - \sum_{j \in \mathcal{N} \setminus \{i\}} E_j(t - [d_j^* - d_i^*]_+)\Big]_+ , \tag{3.18}$$

where d_j^* is the a priori delay bound for any flow j.

Service curve method is advantageous for end-to-end analysis, and also computing bounds for correlated through flow and cross flows. However, it is not clear (expect for FIFO schedulers in some special network scenarios as discussed above) how tight the delay bounds are. We reviewed a list of existing service curves for schedulers in the literature. Interested readers are referred to [40] for a survey on service curves.

3.2 Δ-schedulers

We define a class of schedulers which we call Δ-schedulers for which we are able to provide a service curve formulation.

Definition 3.1 (Δ-schedulers [72], [73])**.** *Suppose there is a set $\mathcal{N}$ of flows that enter a scheduler. The scheduler is called a Δ-scheduler if for any two flows $i, j \in \mathcal{N}$, there is a constant*

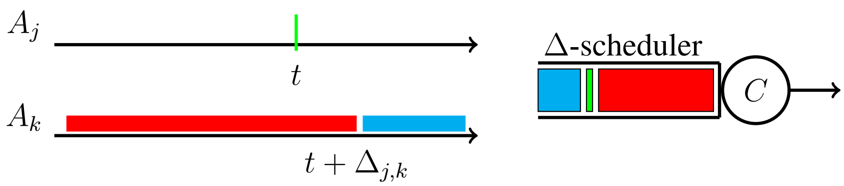


Figure 3.1: Δ-scheduler mechanism.

$\Delta_{i,j}$ which determines the precedence between the arrivals of those flows. More precisely, if there is a tagged arrival from flow i at time t, then the arrivals from flow j have higher precedence than the tagged arrival if and only if they arrive before $t + \Delta_{i,j}$ (Fig. 3.1).

• **Remark:** If we represent Δ values in a matrix form, i.e., $\Delta = \left(\Delta_{i,j}\right)_{i,j \in \mathcal{N}}$ then Δ is an anti-symmetric square matrix of size $|\mathcal{N}| \times |\mathcal{N}|$. That is $\Delta_{i,j} = -\Delta_{j,i}$ and $\Delta_{j,j} = 0 \; \forall j \in \mathcal{N}$ (i.e., arrivals from the same flow are scheduled in FIFO order).

By the above definition, the following relative arrival dependent schedulers can be categorized as Δ-schedulers:

- FIFO: If $\Delta_{i,j} = 0$ for any $i, j \in \mathcal{N}$.
- SP: If $\Delta_{i,j} = +\infty$ if i has a higher priority and $\Delta_{i,j} = -\infty$ if i has a lower priority for any $i, j \in \mathcal{N}$.
- EDF: If $\Delta_{i,j} = d_i^* - d_j^*$ for any $i, j \in \mathcal{N}$.

There are some important schedulers which are not Δ-schedulers. For instance, WRR is not a Δ-scheduler since there is no *constant* that can accurately characterize the precedence of two arrivals at any time. The backlog of all flows affects the precedence of two arrivals from two different flows. Clearly, backlog varies by time and is not a constant.

3.3 Service Curve Formulation for Δ-Schedulers

To compute tight delay and backlog bounds in Δ-schedulers using Network Calculus, tight service curve descriptions of the Δ-schedulers are needed. Before formulating a network service curve for Δ-schedulers, we express some definitions and notations.

Suppose $\mathcal{N}$ is the set of all arrival flows to a Δ-scheduler. For any flow $i \in \mathcal{N}$, define $\mathcal{N}_i$, and $\mathcal{N}_{-i}$ as follows

$$\mathcal{N}_i = \{k \in \mathcal{N} \mid \Delta_{i,k} \neq -\infty\}, \quad \mathcal{N}_{-i} = \mathcal{N}_i - \{i\} .$$

Let us assume that there is a tagged arrival from flow i at time t. Denote by $A_j^{i,t}(t_1, t_2)$ all traffic arrivals from flow j in $[t_1, t_2)$ which must be served before the tagged arrival. Define $\hat{x}_i^t$ as the last time before t when there is no traffic in the queue that must be served before the tagged arrival. It can be expressed by the following

$$\hat{x}_i^t = \sup\left\{s \leq t \mid \forall j \in \mathcal{N}_i : A_j^{i,t}(s) \leq D_j(s)\right\} . \tag{3.19}$$

The above definition implies $A_j(\hat{x}_i^t) = D_j(\hat{x}_i^t)$ for any $j \in \mathcal{N}_i$.

If the tagged arrival has not departed the system by time $t+y$, flow i will use all transmission capacity left unused by other flows in $[\hat{x}_i^t, t+y)$. Since the link is work conserving and busy in that time interval, the total service is $C(t+y-\hat{x}_i^t)$. Thus, the total departures from flow i in that interval is the total capacity in that interval reduced by the total served traffic from other flows, that is

$$\begin{aligned} D_i(\hat{x}_i^t, t+y) &= [C(t+y+\hat{x}_i^t) - \sum_{j \in \mathcal{N}_{-i}} D_j(\hat{x}_i^t, t+y)]_+ \\ &\geq [C(t+y+\hat{x}_i^t) - \sum_{j \in \mathcal{N}_{-i}} A_j^{i,t}(\hat{x}_i^t, t+y)]_+, \end{aligned} \tag{3.20}$$

where the second line uses $A_j(\hat{x}_i^t) = D_j(\hat{x}_i^t)$, and the fact that the system is causal meaning that the departure process cannot exceed the arrival process at any time. By Def. 3.1, we have

$$\begin{aligned} A_j^{i,t}(\hat{x}_i^t, t+y) &= A_j^{i,t}(\hat{x}_i^t, \min\{t+y, t+\Delta_{i,j}\}) \\ &= A_j^{i,t}(\hat{x}_i^t, t+\Delta_{i,j}(y)) \end{aligned}$$

where for any real value a

$$\Delta_{i,j}(a) = \min\{a, \Delta_{i,j}\} .$$

With the above preliminaries, we can proceed to the main theorem.

Theorem 3.1 (Δ-scheduler service curves [72], [73]). *Suppose that a set of traffic flows $\mathcal{N}$ arrives to a link with total capacity C. Flows are multiplexed by a Δ-scheduler in that link. Assume that $\mathcal{G}_j$ for each $j \in \mathcal{N}$ is a statistical sample path envelope in the sense of Def. 2.3 with bounding function ε_j. Let $\underline{\sigma} = (\sigma_j)_{j \in \mathcal{N}_{-i}}$ be a vector of positive constants. Then, for any flow $i \in \mathcal{N}$ and arbitrary $\theta \geq 0$, the function $\mathcal{S}_i$ given below is a service curve in the sense of Def. 2.4*

$$\mathcal{S}_i(t; \underline{\sigma}) = [Ct - \sum_{j \in \mathcal{N}_{-j}} \mathcal{G}_j(t - \theta + \Delta_{i,j}(\theta); \sigma_j)]_+ I_{t>\theta} \tag{3.21}$$

with bounding function

$$\varepsilon_s(\underline{\sigma}) = \sum_{j \in \mathcal{N}_{-i}} \varepsilon_j(\sigma_j) \ .$$

• **Remark 1:** The above theorem extends the Cruz service curve in Eq. (3.17) from FIFO schedulers to all Δ-schedulers. The free parameter θ in the service curve formulation helps to improve the delay and backlog bounds by optimizing over this parameter.

• **Remark 2:** Theorem 3.1 implies a deterministic per-flow service curve, which can be obtained by replacing the statistical sample path envelopes $\mathcal{G}_j$ with deterministic envelopes E_j in the sense of Eq. (2.5), and setting $\varepsilon_j = 0$ for any $j \in \mathcal{N}_{-i}$.

Proof. Assume that we have a tagged arrival from flow i at time t. Let $\hat{x}_i^t$ be the last time before t that there is no buffered traffic with a higher precedence level than the tagged arrival (as defined in Eq. (3.19)). Let us first compute a lower bound for the departures from flow i by time $t + \theta$ for any $\theta \geq 0$.

Two different cases may happen depending on the value of θ. If the tagged arrival does not depart the system by $t + \theta$, then the scheduler is busy in $[\hat{x}_i^t, t + \theta)$, and by Eq. (3.20), we have

$$D_i(\hat{x}_i^t, t + \theta) \geq [C(t + \theta - \hat{x}_i^t) - \sum_{j \in \mathcal{N}_{-i}} A_j(\hat{x}_i^t, t + \Delta_{i,j}(\theta))]_+. \tag{3.22}$$

The above equality can be rearranged as follows

$$\begin{aligned} D_i(t+\theta) &\geq D_i(\hat{x}_i^t) + [C(t+\theta-\hat{x}_i^t) - \sum_{j\in\mathcal{N}_{-i}} A_j(\hat{x}_i^t, t+\Delta_{i,j}(\theta))]_+ \\ &\geq A_i(\hat{x}_i^t) + [C(t+\theta-\hat{x}_i^t) - \sum_{j\in\mathcal{N}_{-i}} A_j(\hat{x}_i^t, t+\Delta_{i,j}(\theta))]_+ \\ &= A_i(\hat{x}_i^t) + [C(t+\theta-\hat{x}_i^t) - \sum_{j\in\mathcal{N}_{-i}} A_j(\hat{x}_i^t, t+\Delta_{i,j}(\theta))]_+ I_{(t+\theta-\hat{x}_i^t>\theta)}, \end{aligned} \tag{3.23}$$

where in the second line we use $A_i(\hat{x}_i^t) = D_i(\hat{x}_i^t)$ which is immediate by Eq. (3.19). The last line uses the fact that the indicator function added to the above expression is always evaluated to 1.

The other case occurs when the tagged arrival departs the system before '$t+\theta$'. Since we know that the scheduler is locally FIFO and the tagged arrival from flow i at time 't' has departed the system by time '$t+\theta$', we have

$$D_i(t+\theta) \geq A_i(t)\ . \tag{3.24}$$

Replacing t with $t-\theta$, and then replacing $\hat{x}_i^j$ with s in Eq. (3.23), yields

$$D_i(t) \geq A_i(s) + [C(t-s) - \sum_{j\in\mathcal{N}_{-i}} A_j(s, t-\theta+\Delta_{i,j}(\theta))]_+ I_{(t-s>\theta)}\ . \tag{3.25}$$

Besides, replacing t with $t-\theta$ in Eq. (3.24) yields

$$D_i(t) \geq A_i(t-\theta)\ .$$

Note that if we have the above inequalities, Eq. (3.25) follows by setting $s = t-\theta$ in that equation. Hence, both Eqs. (3.23) and (3.24) imply that the following inequality always hold

$$\begin{aligned} \forall t\geq 0\ \exists s\leq t:\quad D_i(t) \geq A_i(s) + \Big[&C(t-s) \\ &- \sum_{j\in\mathcal{N}_{-i}} A_j(s, t-\theta+\Delta_{i,j}(\theta))\Big]_+ I_{(t-s>\theta)}. \end{aligned} \tag{3.26}$$

Having the results obtained above, we can prove the theorem as follows

$$
\begin{aligned}
& P\Big\{D_i(t) \geq A_i * \mathcal{S}_i(t;\underline{\sigma})\Big\} \\
& = P\Big\{D_i(t) \geq \inf_{0\leq s\leq t}\big(A_i(s) + \mathcal{S}_i(t-s;\underline{\sigma})\big)\Big\} \\
& = P\Big\{\exists s \leq t: \;\; D_i(t) \geq A_i(s) \\
& \qquad + \Big[C(t-s) - \sum_{j\in\mathcal{N}_{-i}} \mathcal{G}_j(t-s-\theta+\Delta_{i,j}(\theta);\sigma_j)\Big]_+ I_{(t-s>\theta)}\Big\} \\
& \geq P\big\{\forall j \in \mathcal{N}_{-i}, \forall s \leq t: \;\; A_j(t-s,t) \leq \mathcal{G}_j(s;\sigma_j)\big\} \\
& \geq 1 - \sum_{j\in\mathcal{N}_{-i}} P\Big\{\sup_{s\leq t}\left(A_j(t-s,t) - \mathcal{G}_j(s;\sigma_j) > 0\right)\Big\} \\
& \geq 1 - \sum_{j\in\mathcal{N}_{-i}} \varepsilon_j(\sigma_j) \; ,
\end{aligned}
$$

where in the second line the convolution definition is applied. The value of the service curve from the theorem is inserted in the third line. Comparing with the deterministic inequality in Eq. (3.26) with the fourth line, proves that the event in the fifth line is a subset of the event in the fourth line. Boole's inequality and the bounding function of the statistical sample path arrival envelope yields the next two lines and the theorem is proved. □

3.4 Sufficient Condition

We claim that the service curve formulation in Theorem 3.1 provides an accurate description of the operation of a Δ-scheduling algorithm. The accuracy is in terms of the resulting delay bounds which we will prove is a necessary and sufficient constraint. In this section, we prove the sufficiency and the proof of necessity is discussed in the next section.

Applying the per-flow service curve from Theorem 3.1 to the delay bound formulation in Theorem 2.6, yields that $d_i > 0$ is a probabilistic backlog bound for flow i if there exists a vector of positive elements $\underline{\sigma_s} = (\sigma_j)_{j\in\mathcal{N}_{-i}}$ and $\sigma_i \geq 0$ such that

$$
\sup_{x\geq 0}\{\mathcal{G}_i(x;\sigma_i) - \mathcal{S}_i(x+d_i;\underline{\sigma_s})\} \leq 0 \; , \tag{3.27}
$$

in the sense that

$$P\{W_i(t) > d_i\} \leq \varepsilon_i(\sigma_i) + \varepsilon_s(\underline{\sigma_s})$$
$$= \sum_{j \in \mathcal{N}_i} \varepsilon_j(\sigma_j) \ . \tag{3.28}$$

Substituting the service curve from Theorem 3.1 into Eq. (3.27) yields the following

$$\forall x > 0 : \Big[C(x + d_i) - \sum_{j \in \mathcal{N}_{-i}} \mathcal{G}_j(x + d_i - \theta + \Delta_{i,j}(\theta); \sigma_j)\Big]_+ I_{(x+d_i>\theta)} \geq \mathcal{G}_i(x; \sigma_i) \ , \tag{3.29}$$

is a sufficient condition for flow i to meet delay bound d_i with bounding function as in Eq. (3.28). As a special choice, if we set $\theta = d_i$ and noting that $\Delta_{i,i} = 0$, the above delay bound constraint will be reduced to

$$\forall x > 0 : \ C(x + d_i) - \sum_{j \in \mathcal{N}_i} \mathcal{G}_j(x + \Delta_{i,j}(d_i); \sigma_j) \geq 0 \ . \tag{3.30}$$

3.5 Necessary Condition

In this section we show the accuracy of the Δ-scheduler service curve by showing that the delay bound constraint obtained by the Δ-scheduler service curve in Eq. (3.30) is a necessary condition meaning that if this constraint does not hold, there will be a scenario in which the achievable delay violates the delay bounds. The necessity proof is in the context of the deterministic regime and by showing that the delay bounds are achievable. The deterministic delay bound constraints can be obtained by replacing statistical sample path envelopes $\mathcal{G}_j$ with deterministic arrival envelopes E_j in (3.30), which yields the following:

$$\forall x > 0 : \ \Big[\sum_{j \in \mathcal{N}_i} E_j(x + \Delta_{i,j}(d_i)) - Cx\Big] \leq Cd_i \ . \tag{3.31}$$

Theorem 3.2 (Necessity of the delay constraints [72], [73])**.** *Suppose that $\mathcal{N}$ is the set of flow arrivals to a link with total capacity C. A Δ-scheduler is used in that link to share the capacity between flows. Given that E_j is a* concave *deterministic arrival envelope for any $j \in \mathcal{N}_i$, Eq. (3.31) is a necessary delay bound condition for flow i. That is, if $d_i > 0$ is a delay bound for flow i it must satisfy Eq. (3.31).*

Proof. Consider the scenario where the arrival of any flow at any time is equal to its deterministic envelope at that time, that is $A_j(t) = E_j(t)$ for any $t \geq 0$, and $j \in \mathcal{N}_i$. Also assume that we have a tagged arrival from flow i at time t_2. Denote by $B_i^{t_2}(t_1)$ the total backlog at time t_1 of arrivals with higher precedence than the tagged arrival described above. That is

$$\begin{aligned} B_i^{t_2}(t_1) &= \sum_{j \in \mathcal{N}_i} A_j(t_2 + \Delta_{i,j}(t_1 - t_2)) - D_j(t_1) \\ &\geq \underbrace{\sum_{j \in \mathcal{N}_i} E_j(t_2 + \Delta_{i,j}(t_1 - t_2)) - Ct_1}_{:= X(t_1)} , \end{aligned} \tag{3.32}$$

where in the second line we use the assumption that arrivals match their envelopes and the fact that the departure rate cannot exceed the link capacity. Note that X in the above equation is a concave function of t_1. Using this expression, we can prove the theorem as follows.

We use contradiction to prove the theorem. Assume that d_i is a deterministic delay bound for flow i and Eq. (3.31) is violated at some time denoted by t^*. By the contradiction assumption we have $X(t_1)$ is zero at $t_1 = 0$ and positive at $t_1 = t^* + d_i$. Thus, $X(t_1)$ is always positive for any $t_1 \in [0, t^* + d_i]$. Since $B_i^{t^*}(t_1)$ is lower bounded by Eq. (3.32), $B_i^{t^*}(t_1) > 0$ for any $t_1 \in [0, t^* + d_i]$ which means that d_i cannot be a delay bound for flow i and this contradicts the assumption and the theorem is proved. □

The above theorem infers that the service curve formulation for Δ-scheduler is tight in single node analysis.

3.6 The Case of Two Leaky Bucket Flows

In this section, we specify our performance bound formulation for leaky bucket constrained arrivals. Consider an isolated node with total capacity C which uses a Δ-scheduler. There are only two leaky bucket arrivals (see Sec. 2.3.1) in that link: the through flow $A_0 \sim (\sigma_0, \rho_0)$ and the cross flow $A_c \sim (\sigma_c, \rho_c)$. The deterministic service curve for the through flow for such a scenario from Theorem 3.1 is

$$S_0(t) = \Big[Ct - [\rho_c(t - \theta + \Delta(\theta)) + \sigma_c] I_{t>\theta-\Delta(\theta)}\Big]_+ I_{t>\theta} \; , \tag{3.33}$$

where we used Δ to mean $\Delta_{0,c}$. Applying this service curve and leaky bucket arrival envelopes to Theorem 2.2, the following bounds hold for the through flow:

1. **Output burstiness Bound:** If D_0 is the through flow departure process, then for any $s \leq t$ and $t \geq 0$

$$D_0(s,t) \leq \sigma_0 + \rho_0(t - s + \theta_{opt}) \; , \tag{3.34}$$

where

$$\theta_{opt} = \begin{cases} \min\{\frac{\sigma_c}{C-\rho_c}, \frac{\sigma_c+\rho_c\Delta}{C}\} & \text{if } \Delta \geq 0, \\ \frac{[\sigma_c+(C-\rho_0)\Delta]_+}{C} & \text{if } \Delta < 0 \; . \end{cases} \tag{3.35}$$

To show this, we apply Theorem 2.2 as follows

$$\begin{aligned} D_0(s,t) &\leq \sup_{w\geq 0}\{E_0(t-s+w) - S_0(w)\} \\ &= \sup_{w\geq 0}\{(\rho_0(t-s+w) + \sigma_0) \\ &\qquad - [Cw - [\rho_c(w-\theta+\Delta(\theta)) + \sigma_c] I_{w>\theta-\Delta(\theta)}]_+ I_{w>\theta}\} \; . \end{aligned} \tag{3.36}$$

We can optimize over θ to obtain tighter departure envelope. We carry this out separately for $\Delta \geq 0$ and $\Delta < 0$ both in case of stable queue, i.e., $C \geq \rho_0 + \rho_c$.

$\underline{\Delta \geq 0}$: In this case, if $w > \theta$, then $w > \theta - \Delta(\theta)$ and thus we can disregard the term $I_{w>\theta-\Delta(\theta)}$ in Eq. (3.36). By varying w, the expression in the bracket increases until the second part becomes positive and it decreases afterwards. Denote the value of w that makes the second term in the bracket of Eq. (3.36) to zero by w^*. That is,

$$w^* = \frac{\sigma_c - \rho_c(\theta - \Delta(\theta))}{C - \rho_c} \; . \tag{3.37}$$

If $w \in [0, \max(\theta, w^*))$, Eq. (3.36) is increasing in w and it is decreasing for any $w > \max(\theta, w^*)$. Thus, the supremum in Eq. (3.36) happens at $w = \max(\theta, w^*)$. Since w^* is a function of θ, the minimum departure envelope occurs if we choose $\theta = w^*(\theta)$, or the optimum θ is

$$\theta_{opt} = \begin{cases} \frac{\sigma_c}{C-\rho_c} & \text{if } \frac{\sigma_c}{C-\rho_c} \leq \Delta \\ \frac{\sigma_c+\rho_c\Delta}{C} & \text{if } \frac{\sigma_c}{C-\rho_c} > \Delta \ . \end{cases}$$

Inserting this value of θ in Eq. (3.36) and taking the supremum, we have

$$\theta_{opt} = \min\Big\{\frac{\sigma_c}{C-\rho_c}, \frac{\sigma_c+\rho_c\Delta}{C}\Big\} \ , \tag{3.38}$$

Inserting this value of θ in Eq. (3.36) and taking the supremum, we have

$$D_0(s,t) \leq \sigma_0 + \rho_0(t - s + \theta_{opt}) \ .$$

Thus, we proved that Eqs. (3.34) and (3.35) are valid for $\Delta \geq 0$.

$\underline{\Delta < 0}$: In this case $\Delta(\theta) = \Delta$. Inserting this in Eq. (3.36) shows that the expression in the supremum operation is increasing if $w \in [0, \theta)$. It is decreasing in $w \in [\theta, \theta - \Delta)$, increasing in $w \in [\theta - \Delta, \max(\theta - \Delta, w^*))$, and then decreasing afterwards. Thus, the supremum will happen at $w = \theta$ or $w = \max(\theta - \Delta, w^*)$. We assume that $\max(\theta - \Delta, w^*) = \theta - \Delta$. Then, we find θ_{opt} such that Eq. (3.36) gives the same value for $w = \theta_{opt}$ and $w = \theta_{opt} - \Delta$, which leads to

$$\theta_{opt} = \frac{[\sigma_c + (C - \rho_0)\Delta]_+}{C} \ . \tag{3.39}$$

Replacing θ_{opt} from the above equation, to Eq. (3.37) shows that the earlier assumption $\max(\theta - \Delta, w^*) = \theta - \Delta$ is valid. Replacing $\theta = \theta_{opt}$ and $w = \theta_{opt}$ in Eq. (3.36) yields

$$D_0(s,t) \leq \sigma_0 + \rho_0(t - s + \theta_{opt}) \ .$$

Combining Eqs. (3.38) and (3.39), yields Eq. (3.35).

2. **Backlog Bound:** An upper bound for the through flow backlog can be obtained similarly to the output burstiness bound for any $t \geq 0$

$$B_0(t) \leq \sigma_0 + \rho_0\theta_{opt} \ , \tag{3.40}$$

where θ_{opt} is as computed in Eq. (3.35).

3. **Delay Bound:** A delay bound for the through flow is d_0 ($W(t) \leq d_0$), if

$$d_0 = \begin{cases} \min\{\frac{\sigma_0+\sigma_c}{C-\rho_c}, \frac{\sigma_0+\sigma_c+\rho_c\Delta}{C}\} & \text{if } \Delta \geq 0, \\ \frac{\sigma_0+[\sigma_c+(C-\rho_0)\Delta]_+}{C} & \text{if } \Delta < 0 \ . \end{cases} \tag{3.41}$$

We can show this as follows. For the delay bound, since $\theta = d_0$ turns Eq. (3.36) to a necessary and sufficient condition, we have $\theta_{opt} = d_0$. Thus,

$$W_0(t) \leq \inf\{x \geq 0 \mid \forall t \geq 0: \ E_0(t) \leq S_0(t + d_0)\} \ .$$

Inserting service curve from Eq. (3.33) with $\theta = d_0$ and leaky bucket arrival envelope, we have the following

$$\forall t \geq 0: \ \rho_0 t + \sigma_0 \leq [C(t + d_0) - [\rho_c(t + \Delta(d_0)) + \sigma_c]I_{t>-\Delta(d_0)}]_+ \ . \tag{3.42}$$

A delay bound for the through flow is the minimum d_0 that satisfies the above inequality. We compute d_0, separately, for $\Delta \geq 0$ and $\Delta < 0$.

$\underline{\Delta \geq 0}$: Two cases must be considered; $\Delta \geq d_0$ and $0 \leq \Delta < d_0$. d_0 is the minimum value that satisfy the above inequality for both cases. For $\Delta \geq d_0$, Eq. (3.42) is reduced to

$$\forall t \geq 0: \ \rho_0 t + \sigma_0 \leq [(C - \rho_c)(t + d_0) - \sigma_c]_+ \ . \tag{3.43}$$

Since $\rho_c + \rho_0 \leq C$, and the right hand side of the above inequality is concave, it holds for all $t \geq 0$ if it holds for $t = 0$. Thus,

$$d_0 \geq \frac{\sigma_0 + \sigma_c}{C - \rho_c} \ . \tag{3.44}$$

If $0 < \Delta < d_0$, Eq. (3.42) will be reduced to

$$\forall t \geq 0: \ \rho_0 t + \sigma_0 \leq [C(t + d_0) - \rho_c(t + \Delta) - \sigma_c]_+ \ . \tag{3.45}$$

The right hand side of the above inequality is concave. Thus, if $\rho_0 + \rho_c \leq C$, the above equation is valid if it holds at $t = 0$, or

$$d_0 \geq \frac{\sigma_0 + \sigma_c + \rho_c\Delta}{C} \ . \tag{3.46}$$

$\underline{\Delta < 0}$: In this case, Eq. (3.42) will be reduced to

$$\forall t \geq 0 : \ \rho_0 t + \sigma_0 \leq [C(t + d_0) - [\rho_c(t + \Delta) + \sigma_c] I_{t > -\Delta}]_+ \, .$$

There are two critical points $t = 0$ and $t = -\Delta$, which, respectively, leads to $d_0 \geq \frac{\sigma_0}{C}$ and $d_0 \geq \frac{(C-\rho_0)\Delta + \sigma_0 + \sigma_c}{C}$. Considering both cases leads to

$$d_0 \geq \frac{\sigma_0 + [\sigma_c + (C - \rho_0)\Delta]_+}{C} \, . \tag{3.47}$$

Combining this result with Eqs. (3.44), (3.46), and (3.47) yields Eq. (3.41).

- **Concavification:**

To facilitate the extension of the framework to an end-to-end analysis we use a smaller service curve than what is formulated in Eq. (3.33) as follows:

$$S_0(t) = [Ct - [\rho_c(t - \theta + \Delta(\theta)) + \sigma_c]_+] I_{t > \theta} \, . \tag{3.48}$$

The above service curve is concave once it is positive. For this reason we call this simplification *concavification*. Here we introduce concavification for leaky bucket arrivals and compute the error it causes. Comparing Eqs. (3.33) and (3.48) we realize that we do not lose anything for $\Delta \geq 0$ by concavification. If we apply the above service curve for performance bound analysis, for $\Delta < 0$, the corresponding backlog and delay bound would be

$$b_0^{conc} = \rho_0 \frac{[\sigma_c + \rho_c \Delta]_+}{C} + \sigma_0 \, , \tag{3.49}$$

and

$$d_0^{conc} = \frac{\sigma_0 + [\sigma_c + \rho_c \Delta]_+}{C} \, . \tag{3.50}$$

Comparing Eqs. (3.40) and (3.41), with Eqs. (3.49) and (3.50) we have

$$b_0^{conc} - b_0 \leq \rho_0 (1 - u)[\Delta]_- \tag{3.51}$$

$$d_0^{conc} - d_0 \leq (1 - u)[\Delta]_- \, , \tag{3.52}$$

where $u = \frac{\rho_0 + \rho_c}{C}$ is the link utilization. This shows that concavification error exists only for $\Delta < 0$, and decreases as the link utilization increases.

3.7 The Case of Two EBB Flows

Suppose that an EBB through flow $A_0 \sim (M_0, \rho_0, \alpha_0)$ enters a Δ-scheduler with total capacity C. There is also an EBB cross flow in that link with parameters $A_c \sim (M_c, \rho_c, \alpha_c)$. From Eq. (2.53), a statistical sample path envelope for the through $\mathcal{G}_0$, and cross flow $\mathcal{G}_c$ is the following for any choice of $\gamma_0, \gamma_c \geq 0$, and any $\sigma_0, \sigma_c \geq 0$

$$\mathcal{G}_0(t;\sigma_0) = (\rho_0 + \gamma_0)t + \sigma_0; \;\; \varepsilon_0(\sigma_0) = e\Big(1 + \frac{\rho_0}{\gamma_0}\Big)e^{-\alpha_0\sigma_0} \tag{3.53}$$

$$\mathcal{G}_c(t;\sigma_c) = (\rho_c + \gamma_c)t + \sigma_c; \;\; \varepsilon_c(\sigma_c) = e\Big(1 + \frac{\rho_c}{\gamma_c}\Big)e^{-\alpha_c\sigma_c} \; . \tag{3.54}$$

Inserting the above envelope for cross flow in Theorem 3.1, a service curve for the through flow is attained as follows

$$\mathcal{S}(t;\sigma_c) = \Big[Ct - [(\rho_c + \gamma_c)(t - \theta + \Delta(\theta)) + \sigma_c]_+ I_{t>\theta-\Delta(\theta)}\Big]_+ I_{t>\theta} \tag{3.55}$$

for any $\sigma_c \geq 0$ with bounding function

$$\varepsilon_s(\sigma_c) = e\Big(1 + \frac{\rho_c}{\gamma_c}\Big)e^{-\alpha_c\sigma_c} \; . \tag{3.56}$$

Applying the sample path arrival envelope for the through flow from Eq. (3.53), and the corresponding statistical service curve from Eqs. (3.55) and (3.56) in Theorem 2.6. We proceed with the computations for probabilistic delay bound. The argument for the output and backlog bounds follow similar steps. d_0 is a probabilistic delay bound for the through flow in the sense that $P(W_0(t) > d_0) \leq \varepsilon_s(\sigma_c) + \varepsilon_0(\sigma_0)$, the following holds:

$$\forall t > 0 : \qquad \mathcal{S}(t + d_0; \sigma_c) \geq (\rho_0 + \gamma_0)t + \sigma_0 \; .$$

Replacing the service curve from Eq. (3.55) and setting $\theta = d_0$, the above equation yields

$$\forall t > 0 : \qquad \Big[C(t + d_0) - [(\rho_c + \gamma_c)(t + \Delta(d_0)) + \sigma_c] I_{(t>-\Delta(d_0))}\Big]_+ \geq (\rho_0 + \gamma_0)t + \sigma_0 \; .$$

Comparing the above equation with Eq. (3.42) verifies that statistical delay bounds for EBB arrivals can be extracted from the deterministic delay bounds for leaky bucket arrivals by replacing ρ_0 and ρ_c, respectively, by $\rho_0 + \gamma_0$ and $\rho_c + \gamma_c$. Using these replacements and from Sec. 3.6 for any σ_0 and $\sigma_c \geq 0$ the following bounds exist.

1. **Output burstiness bound:**

$$P\{D(s,t) > (\sigma_0 + (\rho_0 + \gamma)(t - s + \theta_{opt})\}$$
$$\leq e\Big(1 + \frac{\rho_0}{\gamma_0}\Big)e^{-\alpha_0\sigma_0} + e\Big(1 + \frac{\rho_c}{\gamma_c}\Big)e^{-\alpha_c\sigma_c} .$$

where

$$\theta_{opt} = \begin{cases} \min\{\frac{\sigma_c}{C-\rho_c-\gamma}, \frac{\sigma_c+(\rho_c+\gamma_c)\Delta}{C}\} & \text{if } \Delta \geq 0, \\ \frac{[\sigma_c+(C-\rho_0-\gamma_c)\Delta]_+}{C} & \text{if } \Delta < 0 . \end{cases} \tag{3.57}$$

2. **Backlog bound:** The through flow backlog is probabilistically bounded as follows

$$P\{B(t) > \sigma_0 + (\rho_0 + \gamma_0)\theta_{opt}\} \leq e\Big(1 + \frac{\rho_0}{\gamma_0}\Big)e^{-\alpha_0\sigma_0} + e\Big(1 + \frac{\rho_c}{\gamma_c}\Big)e^{-\alpha_c\sigma_c} ,$$

where θ_{opt} is as computed in Eq. (3.35).

3. **Delay bound:** For the delay bound, we have

$$P\{W_0(t) > d\} \leq e\Big(1 + \frac{\rho_0}{\gamma_0}\Big)e^{-\alpha_0\sigma_0} + e\Big(1 + \frac{\rho_c}{\gamma_c}\Big)e^{-\alpha_c\sigma_c} ,$$

where

$$d = \begin{cases} \min\{\frac{\sigma_0+\sigma_c}{C-\rho_c-\gamma_c}, \frac{\sigma_0+\sigma_c+(\rho_c+\gamma_c)\Delta}{C}\} & \text{if } \Delta \geq 0, \\ \frac{\sigma_0+[\sigma_c+(C-\rho_0-\gamma_0)\Delta]_+}{C} & \text{if } \Delta < 0 . \end{cases} \tag{3.58}$$

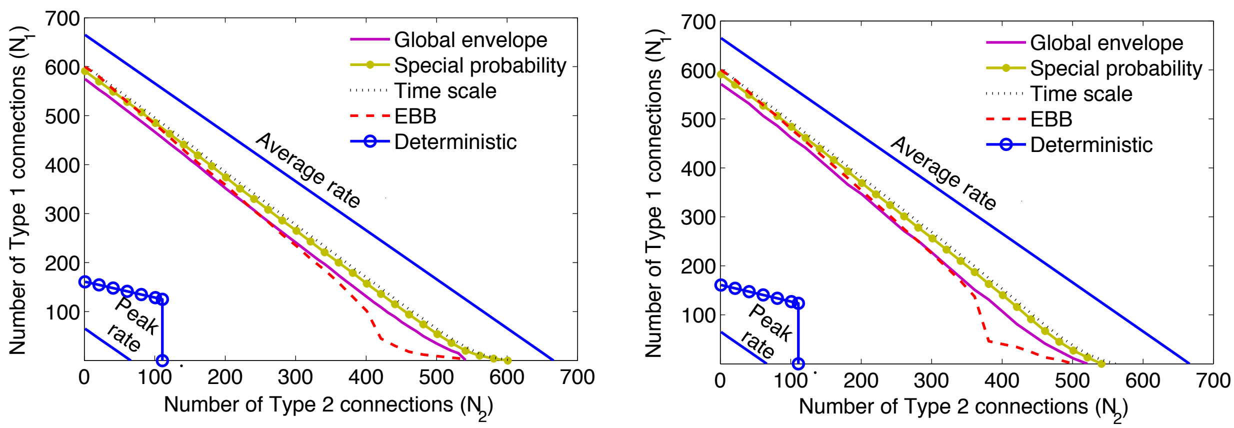
The above bounds assume EBB arrivals in a Δ-scheduler.

3.8 Schedulable Region

Suppose $\mathcal{N}$ is a set of arrivals to a work conserving link with some scheduler. There is an *a priori* delay constraint for each flow. The *schedulable region* is referred to the set of all feasible number of connections from each flow without delay constraint violation. In the following, we compute the schedulable region for a simple scenario of two flow arrivals using the EBB formulation in this chapter. We compare the resulting schedulable region with those in the

Table 3.1: Discrete MMP traffic parameters

	P (Mbps)	p_{00}	p_{11}
Type 1	1.5	0.9	0.1
Type 2	1.5	0.989	0.9

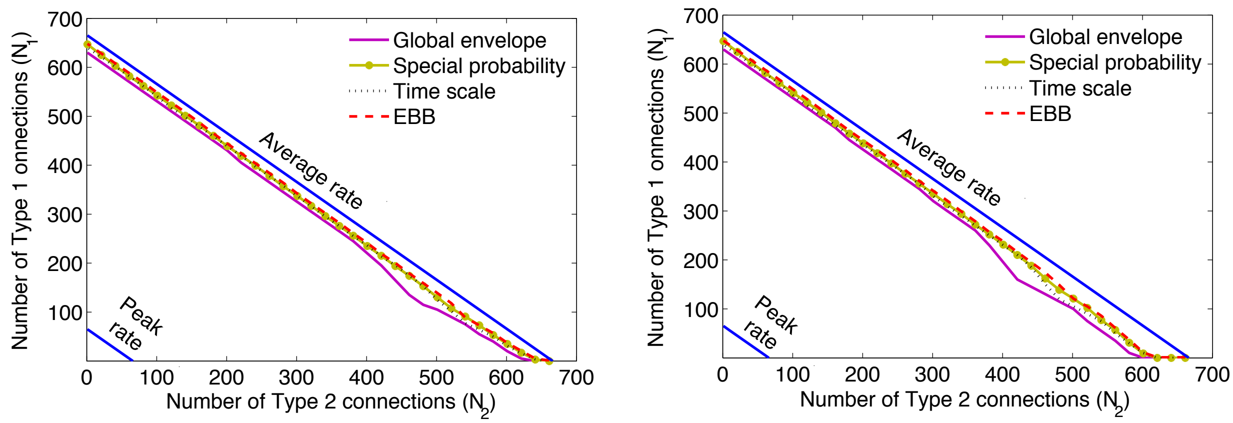


(a) EDF (b) SP

Figure 3.2: Schedulable region for regulated traffic for Type 1 and Type 2 flows from Table 2.1 with $C = 100$ Mbps, $\varepsilon^* = 10^{-6}$, $d_1^* = 100$ ms, and d_2^* = 10 ms.

literature which use different sample path envelopes; global effective envelope [9], time-scale limit [67], and special violation probability [67]. If there are free parameters for a sample path envelope and there is a family of sample path envelopes, we say the set of flows are schedulable if there exists a member of the family which satisfies the delay bound constraints for all time. We also compare the impact of choosing different schedulers. We choose two types of traffic sources for our numerical results: regulated traffic, and discrete time MMP traffic.

For each traffic source, we pick two different parameters which we name as Type 1 and Type 2, with parameters values taken from Table 2.1 and Table 3.1 for the regulated traffic and discrete time MMP traffic, respectively. These parameters are the same as in [9], [20], [67]. There are N_1 connections from Type 1 and N_2 connections from Type 2 multiplexed at a link



(a) EDF

(b) SP

Figure 3.3: Schedulable region for discrete time MMP traffic for Type 1 and Type 2 flows from Table 3.1 with $C = 100$ Mbps, $\varepsilon^* = 10^{-6}$, $d_1^* = 100$ ms, and d_2^* = 10 ms.

with total capacity $C = 100$ Mbps. Assume that $d_1^* = 100$ ms and $d_2^* = 10$ ms, are respectively the *a priori* delay bounds for Type 1 and Type 2 connections. We vary N_2 and compute the maximum N_1 for which delay bound violation is not more than $\varepsilon^* = 10^{-6}$.

Fig. 3.2a illustrates the schedulable region for regulated traffic for different statistical sample path envelopes reviewed in Sec. 2.4.3 when EDF scheduler is used. There are three other curves in that figure, which are used as benchmarks. The curves entitled as 'Average rate' and 'Peak rate' are, respectively, the curves which show the schedulable region of CBR traffic with average and peak rate, which serve as upper and lower bounds on the schedulable region. Finally, the deterministic curve considers that the schedulable region with violation probability zero by considering deterministic arrival envelopes instead of statistical sample path envelopes. The schedulable region for different sample path envelopes are not very different and they all gain considerably compared to the deterministic curve. In Fig. 3.2b, we repeat the experiment for SP. Comparing Figs. 3.2a and 3.2b and, as also acknowledged in the literature such as [9], [67], the choice of scheduler is not as important as the statistical multiplexing gain in terms of schedulable region.

We repeat the same experiments for discrete time MMP traffic sources (with parameters from Table 3.1) in Fig. 3.3a and 3.3b to study the effect of different statistical sample path envelopes, and schedulers on the resulting schedulable region for a different traffic source. Eq. (2.72) is used to Compute the sample path statistical arrival envelopes as described in Sec. 2.4.3. This traffic source also corroborates the results from the regulated traffic. The choice of scheduler, or sample path envelopes does not affect the schedulable region considerably.

Chapter 4

End-to-End Analysis

In this chapter, we develop an end-to-end delay and backlog analysis in the presence of schedulers. Our multi-node analysis consists of two parts. First, we characterize the departure long-term average rate in a tandem of FIFO schedulers and compare it with that of a blind multiplexing tandem. The second part is devoted to the extension of our single node scheduling analysis to end-to-end multi-node scenarios. We evaluate our bounds with numerical examples. The results of this chapter were developed in joint works with Liebeherr and Burchard (See [45], [73]).

4.1 Literature on End-to-End Scheduler Analysis

As mentioned in Chapter 1, end-to-end performance bound computations are simpler in the presence of traffic shapers before each scheduler or use of coordinated schedulers. However, we consider a tandem network scenario as in Fig. 1.1 which excludes any of the above assumptions.

4.1.1 Product-form queuing networks

The joint state probability of a network of queues can be represented in a product form in a *Jackson network*, which can be described as follows. The aggregate of all arrivals from outside the network to node i is a Poisson process with mean rate γ_i. Denote p_{ji} the probability that a departure from node j is routed to node i. If λ_i represents the mean arrival rate to node i, then

$$\lambda_i = \gamma_i + \sum_{j=1}^{H} p_{ji}\lambda_j ,$$

where H is the total number of nodes in the network. In a Jackson network, the service time at each node i is independent and exponentially distributed with parameter μ_i. If $u_i = \frac{\lambda_i}{\mu_i}$ and N_i, respectively, represents node i utilization and the number of packets buffered at node i in the steady state, then, the joint state distribution is characterized by

$$P\{N_1 = n_1, N_2 = n_2, \ldots, N_H = n_H\} = \prod_{i=1}^{H}(1-u_i)u_i^{n_i} .$$

The concise product form of the state probability in Jackson networks is valid for independent external Poisson arrivals, independent exponential service times, and independent routing of packets.

Baskett *et al.* [4] (BCMP network) and Kelly [53] (Kelly network) extend the work by Jackson to multiple classes of arrival flows and to a larger set of schedulers including SP and processor sharing. The joint state distribution of the queues in BCMP and Kelly networks are still in product form. The penalty for this generalization is a more complex formulation. The results of [4], [53] relax the assumption on the service distribution, and routing in Jackson networks, but they still keep the assumptions of Poisson external arrival traffic and independence of the service times.

4.1.2 Network performance analyses using departure characterization

Given a tandem of H nodes as in Fig. 1.1. For each node $h = 1, \ldots, H$, there exists a statistical service curve $\mathcal{S}_h$ with bounding function ε_h. Suppose that a through flow A_0 with a statistical sample path arrival envelope $\mathcal{G}_0$ and bounding function ε_0 passes through all these H nodes. Applying $\mathcal{G}_0$ and $\mathcal{S}_1$ with their corresponding bounding functions in Theorem 2.6, yields a local effective envelope for the through flow departure from the first node $D_{0,1}$. Applying this local effective envelope to one of the methods described in Sec. 2.4.3, a statistical sample path envelope is obtained for $D_{0,1}$. The through flow departure from the first node is the through flow arrival to the second node, i.e., $D_{0,1} = A_{0,2}$. Hence, we have a statistical sample path envelope for the through flow at node $h = 2$. Iterating these steps yields a statistical sample path envelope for the through flow arrival at each node. Applying the through flow statistical sample path envelope and the corresponding statistical service curve $\mathcal{S}_h$ at any node to Theorem 2.6, a per-node probabilistic delay bound can be obtained. Adding up all delay bounds along the path, yields a probabilistic end-to-end delay bound.

In the deterministic regime, pay bursts only once phenomenon expresses that adding per-node delay bounds is outperformed by using a network service curve as described in Sec. 2.3.5. Similar phenomenon exists in the stochastic regime. This is shown in [20] that employing network service curve in Theorem 2.7 leads to a delay bound which scales by $d = O(H \log H)$, compared to the adding per-node delay bounds which scales by $d = O(H^3)$.

4.1.3 Network performance analyses using network service curves

The Network Calculus approach for end-to-end performance bound computations is by formulating a network service curve. A network service curve in the Deterministic Network Calculus is the min-plus convolution of the service curves of all nodes along the path. In a stochastic regime, a service curve formulation is more complicated. Several approaches are proposed in the literature to obtain a statistical network service curve.

1- Using busy period bounds [15], [67]: This method assumes that there is a busy period bound at any node along the end-to-end path. Suppose $\mathcal{S}_h$ is a statistical service curve with bounding function ε_h at node h. If T_h is a busy period bound at that node, then

$$P\Big\{D_h(t) < \inf_{0 \le s \le t} \{A_h(s) + \mathcal{S}_h(t-s)\}\Big\} = P\Big\{D_h(t) < \inf_{0 \le s \le T_h} \{A_h(s) + \mathcal{S}_h(t-s)\}\Big\} \ .$$

This helps to construct a sample path service curve and formulate a network service curve. In fact, using a union bound, a sample path service curve can be achieved at node h with bounding function $T_h \varepsilon_h$. This leads to the following network service curve formulation [15], [67]

$$\mathcal{S}_{net}(t; \sigma_{net}) = \mathcal{S}_1 * \mathcal{S}_2 * \ldots * \mathcal{S}_H(t; \sigma_{net}) \ ,$$

with bounding function

$$\varepsilon_{net}(\sigma_{net}) = \varepsilon_H(\sigma_H) + \sum_{h=1}^{H-1} \Big(\varepsilon_h(\sigma_h) + \sum_{k=h+1}^{H} T_k \varepsilon_k(\sigma_k) \Big) \ .$$

2- Using adaptive service curves [15]: By using a modified definition of service curves, referred to as *adaptive service curves*, a network service curve is formulated in [15]. Suppose A and D are, respectively, the arrival and departure processes to a node. Then, S is an adaptive service curve at that node if for any $\ell > 0$ and $t \ge 0$

$$D(t, t+\ell) \ge A *_t S(\ell) \ , \tag{4.1}$$

where

$$A *_t S(\ell) = \min\Big\{ S(\ell), B(t) + \inf_{x \le \ell} \{A(t, t+\ell-x) + S(x)\} \Big\} \ ,$$

and $B(t)$ is the total backlog at time t. For $t = 0$, Eq.(4.1) reduces to a statistical service curve in Eq. (2.73).

In a probabilistic setting, Eq. (4.1) is replaced by

$$P\{D(t, t+\ell) < A *_t \mathcal{S}(\ell)\} \le \varepsilon^{\ell} \ , \tag{4.2}$$

where ε^{ℓ} is the bounding function. Assuming that in a tandem network, $\mathcal{S}_h$ is a probabilistic adaptive service curve at node h with bounding function ε_h^{ℓ} an adaptive network service curve for any $a > 0$ is

$$\mathcal{S}_{net}^{\ell} = \mathcal{S}_1 * \mathcal{S}_2 * \ldots * \mathcal{S}_H * \delta_{(H-1)a} \ ,$$

with bounding function

$$\varepsilon_{net}^{\ell} = \varepsilon_H^{\ell} + \frac{\sum_{h=1}^{H-1} \varepsilon_h^{\ell}}{a} \ .$$

This method is limited to the class of nodes which satisfies adaptive service curve definitions. Some examples include shapers, schedulers with delay guarantees, and rate-controlled schedulers such as GPS.

3- Using rate relaxation [20]: This convolution theorem was reviewed in Theorem 2.7. This network service curve formulation assumes that the bounding function of the per-node service curves are integrable.

Among the above convolution theorems, we choose the third one which is the easiest method to be used for numerical evaluation and is the only one which does not depend on a priori limits on delay, backlog, or busy periods and is not restrictive in per-node service curve definitions. While the analysis in [20] assumes BMux, we aim to use the statistical network service curve to extend our single node results from the previous chapter to end-to-end results.

4.2 Rate Characterization in FIFO Schedulers

The departure characterization of CBR traffic in FIFO schedulers is used in the literature for the purpose of bandwidth estimation, e.g., [71]. If A_0 and A_c are two CBR traffic arrivals with respective rates r_0 and r_c to a FIFO scheduler with capacity C, then it was shown in [44] that the departure process D_0 corresponding to A_0 is a CBR traffic and can be characterized by

$$D_0(t) = \begin{cases} r_0 t & \text{if } C \geq r_0 + r_c \\ \frac{r_0 C}{r_0 + r_c} t & \text{if } C < r_0 + r_c \ . \end{cases} \tag{4.3}$$

Extending the concept to a more general type of traffic sources, recently, the departure long-term rate of EBB or heavy-tailed self-similar arrivals is shown to converge to Eq. (4.3), almost surely [29].

Assuming EBB or heavy-tailed self-similar (HTSS) traffic arrivals, Ciucu *et al.* [29] employ Cruz's FIFO service curve formulation in (Eq. (3.17)) to prove the following bound for through flow departure

$$P\Big\{D_0(t) < \Big(\frac{r_0}{r_0 + r_c} - k\Big)Ct - \sigma\Big\} \leq \varepsilon(\sigma) \; , \tag{4.4}$$

where $k > 0$ is a correction factor, and ε is a decreasing function of σ and k. ε must satisfy $\lim_{\sigma\to\infty} \varepsilon(\sigma) = 0$. Note that for $k \to 0$, $\varepsilon(\sigma)$ may become a trivial bound (greater than one).

Indeed, k is a constant which depends on both lower and upper bounds on the traffic rates. In particular, for the case of EBB arrivals, r_0^u is called an MGF *upper envelope rate* for through flow, if for any fixed $\theta > 0$, and all $s \leq t$

$$E[e^{\theta A(s,t)}] \leq e^{\theta r_0^u (t-s)} \; , \tag{4.5}$$

and r_0^l is called an MGF *lower envelope rate* for through flow, if for any fixed $\theta > 0$, and all $s \leq t$

$$E[e^{-\theta A(s,t)}] \leq e^{-\theta r_0^l (t-s)} \; . \tag{4.6}$$

Then, combining $\log\Big(E[e^X]\Big) \geq E[X]$ for any random variable X from Jensen's inequality, and Eqs. (4.5) and (4.6) yields

$$r_0^l \leq r_0 \leq r_0^u \; .$$

The overload condition is defined in [29] to be

$$r_0^l + r_c^l > C \; , \tag{4.7}$$

which happens at larger utilization than where the real overload occurs in Eq. (4.3). In this section, we first prove that the conjecture in Eq. (4.3) is valid analytically not only for CBR traffic but also for the long-term average rate of a general traffic source. Then, we will show that on long paths, the departure long-term average rate from a network of FIFO schedulers merges to that of blind multiplexing. While the results of [29] are interesting in the sense that they determine how large the time interval must be to expect the average behavior as in Eq. (4.3), our single node results adds to [29] from the following aspects:

- The overload condition in [29] (See Eq. (4.7)) is a subset of overload condition in Eq. (4.3). However, we assume the same definition of overload condition as in Eq. (4.3).
- The traffic sources are assumed to be EBB, and HTSS in [29], whereas our results applies to general traffic arrival.

Since the analysis in this section is in the context of asymptotic long-term average rate, we recap the definition of the *limit*, below.

Definition 4.1 (Limit in the infinity)**.** *Suppose that x is an independent variable, and $\mathcal{F}$ is a function of x. $\mathcal{F}$ is said to converge to f as x goes to infinity, if and only if for any $\varepsilon > 0$ there exists a constant $K^{\varepsilon} > 0$ such that, for any $x > K^{\varepsilon}$, $|\mathcal{F}(x) - f| < \varepsilon$. This can also be expressed as the following*

$$\lim_{x\to\infty} \mathcal{F}(x) = f \Leftrightarrow \forall\varepsilon > 0 \ \exists K^{\varepsilon} > 0 : \sup_{x > K^{\varepsilon}} |\mathcal{F}(x) - f| < \varepsilon \ .$$

We assume that there is a long-term average rate for through flow and cross flow in the sense that for any $s, t \geq 0$

$$\lim_{t\to\infty} \frac{A_0(s, s+t)}{t} = r_0, \quad \lim_{t\to\infty} \frac{A_c(s, s+t)}{t} = r_c, \tag{4.8}$$

where A_0 and A_c are the arrivals from through and cross flow, respectively. The following theorem captures the asymptotic rates of the departures in FIFO schedulers.

Theorem 4.1 (Departure rate characterization in FIFO schedulers)**.** *Suppose that A_0 and A_c are two arrival processes with respective long-term average rates of r_0 and r_c. These two arrivals enter a FIFO scheduler with total capacity C. Represented by D_0 and D_c are the* *corresponding departure processes from through and cross flow. Then, the long-term average rate of the through flow departure can be characterized as follows*

$$\lim_{t\to\infty} \frac{D_0(t)}{t} = \begin{cases} r_0 & \textit{if } C \geq r_0 + r_c \\ \frac{r_0 C}{r_0 + r_c} & \textit{if } C < r_0 + r_c \ . \end{cases} \tag{4.9}$$

• **Remark**: Although there is no time functionality in Theorem 4.1, it is indeed a more general statement of Eq. (4.3). In a fluid flow model, if all arrivals to a fixed capacity link are CBR and the buffer size is unlimited, then the output traffic is also CBR. Thus, a long-term average rate of the departures in that case is equivalent to the departure CBR rate.

For the proof of the theorem we need the following lemma.

Lemma 4.1. *If u is a function of t satisfying $\lim_{t\to\infty} \frac{u}{t} = 0$ and A is an arrival process with long-term average rate r, i.e., for any $s, t \geq 0$*

$$\lim_{t\to\infty} \frac{A(s, s+t)}{t} = r \ , \tag{4.10}$$

then

$$\lim_{t\to\infty} \frac{A(t-u, t)}{t} = 0 \ .$$

Proof. Note that $\lim_{t\to\infty} \frac{u}{t} = 0$ implies $\lim_{t\to\infty} \frac{t-u}{t} = 1$. The total arrivals in $[t-u, t)$ are given by

$$A(t-u, t) = A(0, t) - A(0, t-u) \ .$$

Dividing both sides by t, and taking the limit yields

$$\begin{aligned} \lim_{t\to\infty} \frac{A(t-u,t)}{t} &= \lim_{t\to\infty} \frac{A(0,t)}{t} - \lim_{t\to\infty} \frac{A(0,t-u)}{t} \\ &= r - \lim_{t\to\infty} \frac{A(0,t-u)}{t-u} \lim_{t\to\infty} \frac{t-u}{t} \\ &= r - r \\ &= 0 \ , \end{aligned}$$

where in the second line, we use Eq. (4.10) and the chain rule. In the third line, we use $\lim_{t\to\infty} \frac{t-u}{t} = 1$. $\square$

Proof. We will prove this theorem separately, for the two cases as follows:

Case 1 $(r_0 + r_c \leq C)$**:**

Fix $t > 0$, and define $t - \hat{x}^t \leq t$ as the start time of the busy period containing time t. In other words, $t - \hat{x}^t$ is the last time before or at time t that the buffer was empty. If we denote the total arrivals, and departure processes to the scheduler respectively by A_{tot} and D_{tot}, then

$$t - \hat{x}^t = \sup\{v \leq t \mid A_{tot}(v) = D_{tot}(v)\}. \tag{4.11}$$

We will first prove that in Case 1, $\hat{x}^t$ is bounded by a constant as t grows. To do so, note that Eq. (4.11) implies that the total arrivals in $[t - \hat{x}^t, t)$ are larger than the total capacity in that interval, or

$$A_{tot}(t - \hat{x}^t, t) > C\hat{x}^t \ .$$

Define $\varepsilon_1 = \frac{A_{tot}(t-\hat{x}^t,t) - C\hat{x}^t}{2\hat{x}^t} > 0$. Combining this with the above equation, we have

$$\frac{A_{tot}(t - \hat{x}^t, t)}{\hat{x}^t} > C + \varepsilon_1 \ . \tag{4.12}$$

On the other hand, from Eq (4.8), and Def. 4.1, there exists a constant K^{ε_1} such that

$$\forall x > K^{\varepsilon_1}: \qquad \frac{A_{tot}(t - x, t)}{x} < r_0 + r_c + \varepsilon_1 \ . \tag{4.13}$$

Comparing Eqs. (4.12) and (4.13) and the fact that $r_0 + r_c \leq C$ shows that $\hat{x}^t \leq K^{\varepsilon_1}$, or $\frac{\hat{x}^t}{t}$ decays to zero as t grows. Since the total departure at time t is lower bounded by the total arrival up to time $t - \hat{x}^t$, we have the following:

$$\begin{aligned}
\lim_{t\to\infty} \frac{D_{tot}(t)}{t} &\geq \lim_{t\to\infty} \frac{A_{tot}(t) - A_{tot}(t - \hat{x}^t, t)}{t} \\
&\geq r_0 + r_c - \lim_{t\to\infty} \frac{A_{tot}(t - \hat{x}^t, t)}{t} \\
&= r_0 + r_c \ ,
\end{aligned} \tag{4.14}$$

where we use Lemma 4.1 and the fact that $\lim_{t\to\infty} \frac{\hat{x}^t}{t} = 0$ to obtain the second line.

In addition, $D_{tot}(t) \leq A_{tot}(t)$ and thus, $\lim_{t\to\infty} \frac{D_{tot}(t)}{t} \leq \lim_{t\to\infty} \frac{A_{tot}(t)}{t} = r_0 + r_c$. Combining this result with Eq. (4.14) completes the proof for the first case.

Case 2 ($r_0 + r_c > C$)**:**

We first need to show that the last idle time before t, i.e., $t - \hat{x}_t$ is bounded by a constant as t grows. By the definition of $t - \hat{x}^t$ in Eq. (4.11), we have

$$A_{tot}(t - \hat{x}^t) \le C(t - \hat{x}^t) \; . \tag{4.15}$$

On the other hand, Def. 4.1 together with Eq. (4.8) convey that for any arbitrary $\varepsilon_2 > 0$ there exists a constant K^{ε_2} such that

$$\forall x > K^{\varepsilon_2} : \qquad \frac{A_{tot}(x)}{x} > r_0 + r_c - \varepsilon_2 \; . \tag{4.16}$$

Since $r_0 + r_c > C$, if we set $\varepsilon_2 = \frac{r_0 + r_c - C}{2} > 0$ in the above equation, then there exists a constant K^{ε_2} such that

$$\forall x > K^{\varepsilon_2} : \qquad \frac{A_{tot}(x)}{x} > \frac{r_0 + r_c + C}{2} > C \; . \tag{4.17}$$

Comparing Eqs. (4.15) and (4.17), we have $t - \hat{x}^t \le K^{\varepsilon_2}$ which means that $t - \hat{x}^t$ is always bounded by a constant regardless of t. This also implies that $\lim_{t\to\infty} \frac{\hat{x}^t}{t} = 1$. Using this result, we have

$$\begin{aligned}
\lim_{t\to\infty} \frac{D_{tot}(t - \hat{x}^t, t)}{t} &= \lim_{t\to\infty} \frac{D_{tot}(t) - D_{tot}(t - \hat{x}^t)}{t} \\
&= \lim_{t\to\infty} \frac{D_{tot}(t) - A_{tot}(t - \hat{x}^t)}{t} \qquad (4.18) \\
&= \lim_{t\to\infty} \frac{D_{tot}(t)}{t} \; , \qquad (4.19)
\end{aligned}$$

where in the second line we use the definition of $t - \hat{x}^t$ from Eq. (4.11). In the last line, we use Lemma 4.1 and the fact that $\lim_{t\to\infty} \frac{t - \hat{x}^t}{t} = 0$ if $r_0 + r_c > C$. Hence,

$$\begin{aligned}
\lim_{t\to\infty} \frac{D_{tot}(t)}{t} &= \lim_{t\to\infty} \frac{D_{tot}(t - \hat{x}^t, t)}{t} \\
&= \lim_{t\to\infty} \frac{C\hat{x}^t}{t} \\
&= C \; , \qquad (4.20)
\end{aligned}$$

where in the second line, we use the definition of $t - \hat{x}^t$ and the last line uses $\lim_{t\to\infty} \frac{\hat{x}^t}{t} = 1$.

Denote $\hat{u}_t$ the latest arrival time of any departed traffic up to time t. That is

$$D_{tot}(t) = A_{tot}(\hat{u}_t). \tag{4.21}$$

Since the scheduler is busy in $[t - \hat{x}, t)$ and it cannot serve more than $D_{tot}(t)$ in that time interval, we have

$$C\hat{x}^t \leq A_{tot}(\hat{u}_t) \ .$$

Dividing both sides by $\hat{x}^t$, and taking the limit, we have

$$\begin{aligned} C &\leq \lim_{t\to\infty} \frac{A_{tot}(\hat{u}_t)}{\hat{x}^t} \\ &= \lim_{t\to\infty} \frac{A_{tot}(\hat{u}_t)}{t} \lim_{t\to\infty} \frac{t}{\hat{x}^t} \\ &= \lim_{t\to\infty} \frac{A_{tot}(\hat{u}_t)}{t} \ , \end{aligned} \tag{4.22}$$

where chain rule is used in the second line, and $\lim_{t\to\infty} \frac{\hat{x}^t}{t} = 1$ is employed to obtain the last line. With this result, we have $\lim_{t\to\infty} \hat{u}_t = \infty$, otherwise Eq. (4.22) along with Lemma 4.1 infer that $C \leq 0$. Hence, we have

$$\begin{aligned} \lim_{t\to\infty} \frac{D_{tot}(t)}{t} &= \lim_{t\to\infty} \frac{A_{tot}(\hat{u}_t)}{\hat{u}_t} \lim_{t\to\infty} \frac{\hat{u}_t}{t} \\ &= (r_0 + r_c) \lim_{t\to\infty} \frac{\hat{u}_t}{t} \ , \end{aligned} \tag{4.23}$$

where in the first line we use Eq. (4.21) and that $\lim_{t\to\infty} \hat{u}_t = \infty$ to obtain the second line. Combining Eqs. (4.20) and (4.23), yields

$$\lim_{t\to\infty} \frac{\hat{u}_t}{t} = \frac{C}{r_0 + r_c} \ . \tag{4.24}$$

We can exploit the above equation to prove the theorem as follows

$$\begin{aligned} \lim_{t\to\infty} \frac{D_0(t)}{t} &= \lim_{t\to\infty} \frac{A_0(\hat{u}_t)}{\hat{u}_t} \frac{\hat{u}_t}{t} \\ &= \lim_{t\to\infty} \frac{A_0(\hat{u}_t)}{\hat{u}_t} \lim_{t\to\infty} \frac{\hat{u}_t}{t} \\ &= \frac{r_0 C}{r_0 + r_c} \ , \end{aligned}$$

where in the first line we use the definition of $\hat{u}_t$ and the chain rule. In the last line, we inserted the values from Eqs. (4.8) and (4.24). $\square$

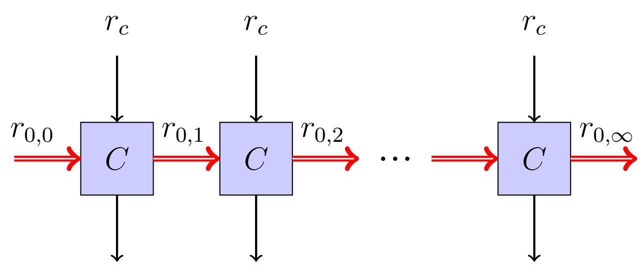


Figure 4.1: The multi-node scenario used in Theorem 4.2.

With iterating the above theorem, we can characterize the long-term average rate of the departure of a large tandem network. This is considered in the following theorem.

Theorem 4.2 ([44]). *Consider the scenario depicted in Fig. 4.1. Suppose that the long-term average rate of the cross flow at all nodes are identical and equal to r_c. Represented by $r_{0,h}$ is the long-term through flow departure rate from node h. The long-term through flow arrival rate to the network is denoted by $r_{0,0}$ and equals to $r_{0,0} = r_0$. If the number of nodes in the path tends to infinity, then the long-term average rate departure of the network can be characterized by the following*

$$r_{0,\infty}(t) = \begin{cases} r_0 & \text{if } C \geq r_0 + r_c \\ [C - r_c]_+ & \text{if } C < r_0 + r_c. \end{cases} \tag{4.25}$$

Proof. If $r_0 + r_c \leq C$, then by Theorem 4.1, the long-term rate of the through flow arrival and departures from the first node are equal to r_0. Repeating the same argument for the next node, we realize that in the case of $r_0 + r_c \leq C$, we have $r_{0,\infty} = r_0$.

If $r_0 + r_c > C$, we will show by induction that the long-term average rate of the through flow departure from any node h along the path can be obtained by the following recursion formula

$$r_{0,h} = \frac{r_{0,h-1} C}{r_{0,h-1} + r_c}, \tag{4.26}$$

with initial condition $r_{0,0} = r_0$. Moreover, the rate obtained from the above equation satisfies the inequalities

$$[C - r_c]_+ < r_{0,h} < r_{0,h-1}. \tag{4.27}$$

For $h = 1$, Eqs. (4.26) and (4.27) are immediately obtained by using Theorem 4.1.

Now assume that the through flow long-term average departure rate from the k'th node is $r_{0,k} = \frac{r_{0,k-1}C}{r_{0,k-1}+r_c}$, for some $r_{0,k-1}$ and that $[C - r_c]_+ < r_{0,k} < r_{0,k-1}$. Since $C - r_c < r_{0,k}$ by the induction assumption, we will have $r_{0,k+1} = \frac{r_{0,k}C}{r_{0,k}+r_c}$ from Theorem 4.1. Combining $r_{0,k+1} = \frac{r_{0,k}C}{r_{0,k}+r_c}$, and $[C - r_c]_+ < r_{0,k}$ leads to $[C - r_c]_+ < r_{0,k+1} < r_k$. Thus, Eqs. (4.26), (4.27) are proved by induction.

Eq. (4.27) shows that the long-term average rate of the through flow departure strictly decreases as through flow traverses each node along the path. On the other hand the same equation shows that the departure long-term average rate is always lower bounded by $[C - r_c]_+$. Thus, if the number of nodes in the path tends to infinity, we will have $r_{0,\infty} = [C - r_c]_+$.

□

To capture the effect of FIFO schedulers on long paths, for CBR arrivals, we consider the case where all FIFO schedulers are replaced by SP with through flow having the lowest priority, and compute the departure process for that scenario in the following theorem.

Theorem 4.3 (Departure rate characterization in BMux)**.** *Replacing all FIFO schedulers in the scenario described in Theorem 4.2 by SP with through flow having the lowest priority, the long-term average rate from the network matches the formulation in Eq. (4.25) for a path of arbitrary length.*

Proof. Denote by $A_{0,h}$, $D_{0,h}$, and A_c, respectively, the through flow arrival and departure at node h, and the cross flow arrival at any node. We prove the theorem initially for the special case of an isolated node. The departure at time t from the through flow is lower bounded by the following

$$D_{0,1}(t) \geq A_{0,1}(\hat{x}^t) + [C(t - \hat{x}^t) - A_c(\hat{x}^t, t)]_+ \, , \tag{4.28}$$

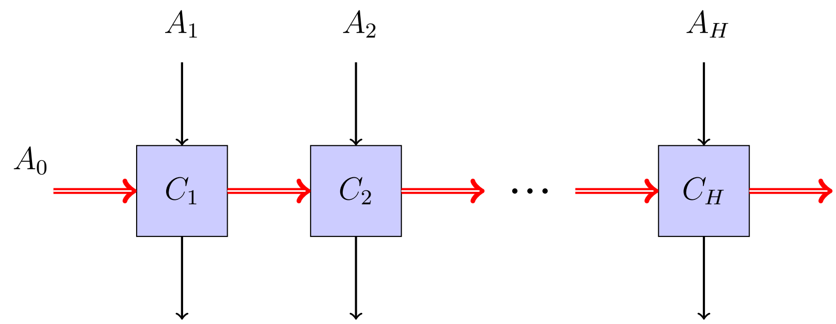


Figure 4.2: Tandem network scenario

where $\hat{x}^t$ is as defined in Eq. (4.11). Dividing both sides by t and using the fact that $t - \hat{x}^t$ is bounded by a constant if $C \geq r_0 + r_c$, and $\hat{x}^t$ is bounded by a constant if $C < r_0 + r_c$ proves that the claim for single node scenario. Iterating the above argument proves the theorem for any network sizes. In fact, we have shown that the departure rate in case of BMux is lower bounded by that of FIFO. Given that BMux is the worst-case scheduling and cannot be worse than FIFO completes the proof. $\square$

The above theorem implies that on long paths, FIFO converges to the worst-case scheduling (BMux) in terms of long-term average rate for any type of traffic sources.

4.3 Statistical End-to-End Performance Bounds

In this section, we return to the performance analysis of Δ-schedulers. We consider the tandem network scenario depicted in Fig. 4.2. Each node is a Δ-scheduler with parameter $\Delta_{0,c}^h$ for the through flow with respect to the cross flow and total capacity C_h. To simplify notation, we use Δ^h to mean $\Delta_{0,c}^h$. Computing statistical performance bounds using Network Calculus requires a statistical network service curve formulation. As reviewed in Theorem 2.7, a statistical convolution theorem is provided in [20] and can be used to compute a statistical network service curve if per-node service curves have integrable bounding functions. To formulate a network service curve, it is assumed that through flow traverses a tandem network in which all nodes are SP with the lowest priority assigned to the through flow at each node. In the following, we use the convolution theorem from [20] for end-to-end delay bound analysis in a tandem of Δ-schedulers.

Suppose that the tandem network in Fig. 4.2 is a homogeneous network. Each node $h = 1, \ldots, H$ in that scenario is a Δ-scheduler with parameter $\Delta_{0,c}^h = \Delta$, and capacity C. The through flow and cross flow are leaky bucket sources with respective parameters $A_0 \sim (\sigma_0, \rho_0)$ and $A_h \sim (\sigma_c, \rho_c)$. The statistical service curve formulation we use for this type of traffic is a lower bound version of the service curve from Eq. (3.48). That is, for fixed $\theta_c \geq 0$, and any $\sigma_c > 0$

$$S_h(t) = \Big[Ct - [\rho_c(t - \theta_c + \Delta(\theta_c)) + \sigma_c]_+\Big]_+ I_{t>\theta_c} \tag{4.29}$$

$$= \Big[C(t + \theta_c) - [\rho_c(t + \Delta(\theta_c)) + \sigma_c]_+\Big]_+ I_{t>0} * \delta_{\theta_c}(t) \ , \tag{4.30}$$

where in the second line, we use a shift function δ_{θ_c}.

In order to examine the accuracy of the convolution theorem in [20] for scheduling analysis, we can compare the resulting network service curve from Theorem 2.7 in the deterministic special case with that of deterministic convolution theorem in Theorem 2.3. Theorem 2.7 enforces per-node service curves to be of the form of $S_h(t) - \sigma_c$. For this reason, σ_c must be taken out from the bracket in Eq. (4.30), which leads to a lower bound on the service curve which we represent it by $\widetilde{\mathcal{S}}_h$, that is

$$\widetilde{\mathcal{S}}_h(t; \sigma_c) = \Big[C(t + \theta_c) - [\rho_c(t + \Delta(\theta_c))]_+\Big]_+ I_{t>0} * \delta_{\theta_c}(t) - \sigma_c \ . \tag{4.31}$$

Applying the per-node service curve in Eq. (4.31) to Theorem 2.7 and setting $\gamma_h, \tau_h = 0$ for any $h = 1, \ldots, H-1$ to be in the special case of deterministic regime, we have

$$\begin{aligned} S_{net}(t) &= \widetilde{\mathcal{S}}_1 * \ldots * \widetilde{\mathcal{S}}_H(t) - H\sigma_c \\ &= \bigotimes_{h=1}^{H}\Big(\Big[C(t+\theta_c) - [\rho_c(t+\Delta(\theta_c))]_+\Big]_+ I_{t>0}\Big) * \delta_{H\theta_c} - H\sigma_c \\ &= \Big[C(t+\theta_c) - [\rho_c(t+\Delta(\theta_c))]_+\Big]_+ I_{t>0} * \delta_{H\theta_c} - H\sigma_c \\ &= \Big[C(t-(H-1)\theta_c) - [\rho_c(t-H\theta_c+\Delta(\theta_c))]_+\Big]_+ I_{(t>H\theta_c)} - H\sigma_c \ , \end{aligned} \tag{4.32}$$

where in the second line, we use the associativity and commutativity of the min-plus convolution operator. In the third line, we use the fact that Eq. (4.31) is concave for any $t > 0$. Let us compare the resulting network service curve from the above methods for two cases of BMux and FIFO.

- BMux ($\Delta = \infty$): From Eq. (4.29),

$$S_h(t) = [(C-\rho_c)t - \sigma_c]_+ I_{(t>H\theta_c)} \ .$$

S_h from the above equation is always decreasing in θ_c. Thus, the optimum value of θ_c is zero. By setting $\theta_c = 0$ and using shift functions, the above per-node service curve can be reformulated as

$$\begin{aligned} S_h(t) &= [(C-\rho_c)t - \sigma_c]_+ I_{(t>0)} \\ &= (C-\rho_c)tI_{(t>0)} * \delta_{\frac{\sigma_c}{C-\rho_c}} \ . \end{aligned}$$

Applying this per-node service curve in Theorem 2.3, yields

$$\begin{aligned} S_{net}(t) &= S_1 * \ldots * S_H(t) \\ &= (C-\rho_c)tI_{(t>0)} * \delta_{\frac{H\sigma_c}{C-\rho_c}} \\ &= [(C-\rho_c)t - H\sigma_c]_+ \ , \end{aligned} \tag{4.33}$$

where in the second line, we use the fact that $(C-\rho_c)t$ is a concave function. Replacing $\Delta = \infty$ and $\theta_c = 0$ in Eq. (4.32) and noting that $S_{net}(t) \geq 0$ lead to the same network service curve as in Eq. (4.33). Consequently, the existing statistical convolution theorem is as good as the deterministic convolution in Theorem 2.3 in a tandem of blind multiplexors.

- FIFO ($\Delta = 0$): Applying the per-node service curve from Eq. (4.30) to Theorem 2.3, yields

$$\begin{aligned} S_{net}(t) &= S_1 * \ldots * S_H(t) \\ &= \bigotimes_{h=1}^{H} \Big([(C-\rho_c)t + C\theta_c - \sigma_c]_+ I_{t>0} \Big) * \delta_{H\theta_c} \ , \end{aligned} \tag{4.34}$$

where we use the associativity and commutativity of the convolution theorem in the second line. In this case, for any $\theta_c > \frac{\sigma_c}{C}$, the term in the bracket in Eq. (4.29) is concave and increasing for any $t > 0$. Hence, the resulting network service curve from Eq. (4.29) would be

$$S_{net}(t) = [C(t-(H-1)\theta_c) - \rho_c(t-H\theta_c) - \sigma_c]_+ I_{t>H\theta_c} \ . \tag{4.35}$$

On the other hand, applying $\Delta = 0$ in Eq. (4.32), yields

$$S_{net}(t) = [C(t-(H-1)\theta_c) - \rho_c(t-H\theta_c)]_+ I_{t>H\theta_c} - H\sigma_c \ . \tag{4.36}$$

Comparing Eqs. (4.35) and (4.36) shows that cross flow burstiness is overestimated in the existing convolution theorem for FIFO schedulers. Hence, Theorem 2.7 is more pessimistic than Theorem 2.3 for FIFO schedulers. This was the motivation for our work in [45] to strengthen the convolution theorem for scheduling analysis.

4.3.1 New convolution theorem

As shown in the previous section, although the existing convolution theorem formulation works properly for BMux, applying that convolution theorem to per-node service curve for schedulers leads to loose network service curve descriptions. More precisely, excluding the service bursts σ_h's from the convolution leads to a pessimistic network service curve for Δ-schedulers. To resolve this problem, we propose a new convolution which incorporates service bursts in the convolution and provides a tighter network service curve formulation.

Theorem 4.4 (New statistical network service curve [45]). *Suppose that a through flow traverses a cascade of H nodes. There is a statistical service curve in the sense of Def. 2.4 $\mathcal{S}_h$ at each node for the through flow at that node, with bounding function ε_h. Assume that ε_h for each h is integrable, i.e.,*

$$\int_0^\infty \varepsilon_h(y)dy < \infty \ , \tag{4.37}$$

and $\mathcal{S}_h$ satisfies the following for any σ_h, $x, t \geq 0$

$$\mathcal{S}_h(t;\sigma_h + x) \geq [\mathcal{S}_h(t;\sigma_h) - x]_+ \ . \tag{4.38}$$

For any $h = 1, \dots, H-1$, define γ_h, $\tau_h \geq 0$, and any vector of non-negative elements $\sigma_{net} = (\sigma_1, \dots, \sigma_H)$. The following is a statistical end-to-end network service curve for the through flow

$$\mathcal{S}_{net}(t;\sigma_{net}) = [\mathcal{S}_1 * \dots * \mathcal{S}_H * \delta_{\sum_{h<H}\tau_h}(t;\sigma_{net}) - \sum_{h<H}\gamma_h t]_+ \ , \tag{4.39}$$

with bounding function

$$\varepsilon_{net}(\sigma_{net}) = \varepsilon_H(\sigma_H) + \sum_{h<H}\frac{1}{\gamma_h\tau_h}\int_{\sigma_h}^\infty \varepsilon_h(y)dy \ . \tag{4.40}$$

Proof. We first prove that the following is a statistical network service curve with the same bounding function as stated in Eq. (4.40),

$$\mathcal{S}_{net}(t;\sigma_{net}) = \inf_{x_1,\dots,x_H}\Big\{\mathcal{S}_H(x_H;\sigma_H) + \sum_{h<H}\mathcal{S}_h(x_h;\sigma_h + \gamma_h \sum_{k=h+1}^{H}(\tau_{k-1} + x_k))\Big\} \ , \tag{4.41}$$

where x_k must belong to the following set

$$\Big\{x_1, \dots, x_H \mid \sum_{k=1}^{H} x_k + \sum_{k=1}^{H-1}\tau_k \leq t\Big\} \ . \tag{4.42}$$

Then, using the assumptions in Eqs. (4.38) and (4.42), we infer that Eq. (4.41) implies the theorem statement in Eq. (4.39) and the proof would be completed.

We prove Eq. (4.41) by induction. For $H = 1$ the problem reduces to a single node case. Hence, we start the proof by considering a network of two nodes, i.e., $H = 2$. Denote the network service curve in this case by $\mathcal{S}_{1,2}$, and the corresponding bounding function by $\varepsilon_{1,2}$. Represent through flow arrival and departure at node h respectively by A_h, and D_h. With these notations we need to show that $\mathcal{S}_{1,2}$ from the following

$$\mathcal{S}_{1,2}(t;(\sigma_1,\sigma_2)) = \inf_{x_1+x_2\le t-\tau_1} \{\mathcal{S}_2(x_2;\sigma_2) + \mathcal{S}_1(x_1;\sigma_1+\gamma_1(\tau_1+x_2))\}$$

must satisfy

$$P\{D_2(t) < A_1 * \mathcal{S}_{1,2}(t;(\sigma_1,\sigma_2))\} \le \varepsilon_2(\sigma_2) + \frac{1}{\gamma_1\tau_1}\int_{\sigma_1}^{\infty} \varepsilon_1(y)dy =: \varepsilon_{1,2}(\sigma_1,\sigma_2)\ . \quad (4.43)$$

Since $\mathcal{S}_2$ is a statistical service curve at the second node, the following inequality

$$\forall t \ge 0\ \exists x_2 \le t: \quad D_2(t) \ge A_2(t-x_2) + \mathcal{S}_2(x_2)\ , \quad (4.44)$$

will not be violated with violation probability more than $\varepsilon_2(\sigma_2)$. Now, suppose the following inequality also holds

$$\forall x \le t\ \exists x_1 \le t-x-\tau_1: \quad D_1(t-x_1) \ge A_1(t-x-\tau_1-x_1) + \mathcal{S}_1(x_1;\sigma_1+\gamma_1(x+\tau_1))\ . \quad (4.45)$$

Then, inserting the right hand side of the above inequality as a lower bound for A_2 in Eq. (4.44) shows that

$$D_2(t) \ge A_1 * \mathcal{S}_{1,2}(t;(\sigma_1,\sigma_2))\ . \quad (4.46)$$

Now we need to show that the bounding function of the above statistical service curve matches Eq. (4.43). By the union bound, the bounding function of Eq. (4.46) is upper bounded by the sum of the violation probabilities of Eqs. (4.44) and (4.45). The bounding function of Eq. (4.44) is $\varepsilon_2(\sigma_2)$ from the theorem statement. To compute the bounding function of Eq. (4.45) we discretize time with time unit τ_1. There exists an integer $k \ge 1$ such that $(k-1)\tau_1 \le x < k\tau_1$. Since A_1, $\mathcal{S}_1$, and D_1 are non-decreasing functions, a necessary condition that Eq. (4.45) fails is that

$$D_1(t-k\tau_1) < A_1(t-x_1-k\tau_1) + \mathcal{S}_1(x_1;\sigma_1+k\gamma_1\tau_1)\ . \quad (4.47)$$

Thus,

$$\sum_{k=1}^{\infty} P\Big\{D_1(t-k\tau_1) < \inf_{x_1 \le t-k\tau_1} \{A_1(t-k\tau_1-x_1) + \mathcal{S}_1(x_1;\sigma_1+k\gamma_1\tau_1)\}\Big\}$$
$$\le \sum_{k=1}^{\infty} \varepsilon_1(\sigma_1 + k\gamma_1\tau_1)$$
$$\le \frac{1}{\gamma_1\tau_1}\int_{\sigma_1}^{\infty} \varepsilon_1(y)dy \ ,$$

where in the second line we use that $\mathcal{S}_1$ is a statistical service curve with bounding function ε_1, and the last line follows with the Riemann integral. Adding the violation probabilities of Eqs. (4.44) and (4.45) completes the proof for $H = 2$.

Now suppose that the theory is correct for any $H-1$ tandem of nodes and we want to prove the correctness for H nodes, with $H > 2$. Consider a cascade of $H > 2$ nodes. For the last $H-1$ nodes we have

$$\mathcal{S}_{2,\ldots,H}(t;(\sigma_2,\ldots,\sigma_H)) = \inf_{x_2,\ldots,x_H} \{\mathcal{S}_H(x_H;\sigma_H) + \sum_{1<h<H} \mathcal{S}_h(x_h;\sigma_h + \gamma_h \sum_{k=h+1}^{H} (\tau_{k-1} + x_k))\} \ ,$$

where

$$\sum_{h=1}^{H} x_h + \sum_{h=1}^{H-1} \tau_h \le t \ ,$$

and the bounding function is

$$\varepsilon_{2,\ldots,H}(\sigma_2,\ldots,\sigma_H) = \varepsilon_H(\sigma_H) + \sum_{h=2}^{H-1} \frac{1}{\gamma_h\tau_h}\int_{\sigma_h}^{\infty} \varepsilon_h(y)dy \ .$$

Then, replacing $\mathcal{S}_2$ with $\mathcal{S}_{2,\ldots,H}$ and ε_2 with $\varepsilon_{2,\ldots,H}$ in the argument for $H = 2$ proves the theorem for $H > 2$ case, and we proved the theorem by induction. □

For the rest of this chapter we take a simpler form of the network service curve by choosing $\gamma_h = \gamma$ for any $h = 1,\ldots,H-1$. Thus, Eqs. (4.39) and (4.40) are simplified, respectively, to

$$\mathcal{S}_{net}(t;\sigma_{net}) = [\mathcal{S}_1 * \ldots * \mathcal{S}_H * \delta_{\tau_{net}}(t;\sigma_{net}) - (H-1)\gamma t]_+ \ , \tag{4.48}$$

and

$$\varepsilon_{net}(\sigma_{net}) = \varepsilon_H(\sigma_H) + \sum_{h<H} \frac{1}{\gamma\tau_h}\int_{\sigma_h}^{\infty} \varepsilon_h(y)dy \ , \tag{4.49}$$

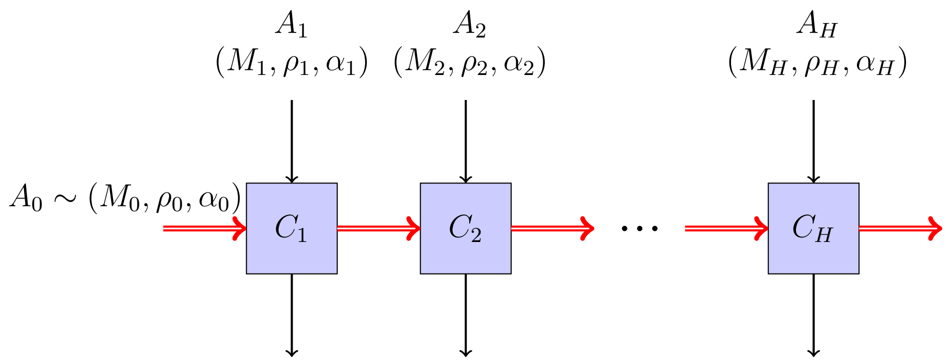


Figure 4.3: The multi-node EBB arrivals scenario

where we also simplified the notation in Eq. (4.48) by using $\tau_{net} = \sum_{h<H} \tau_h$.

Theorem 3.1 formulates a per-node service curve for Δ-schedulers. Applying Theorem 4.4 to that service curve gives us an end-to-end statistical service curve in a tandem of Δ-schedulers. If we insert this network service curve and a statistical through flow sample path envelope to Theorem 2.6, then end-to-end probabilistic performance bounds can be computed. In the following, we compute the corresponding bounds for EBB traffic arrivals.

4.3.2 Statistical performance bounds for EBB traffic

Suppose we have a tandem network scenario as depicted in Fig. 4.3. The through flow and cross flow (at node h) are EBB sources, respectively, with EBB parameters $A_0 \sim (M_0, \rho_0, \alpha_0)$ and $A_h \sim (M_h, \rho_h, \alpha_h)$. Each node $h = 1, \ldots, H$ in that scenario is a Δ-scheduler with parameter $\Delta_{0,c}^h = \Delta^h$. A network service curve for the through flow in this scenario for any $\gamma_c \geq 0$ with bounding function

$$\varepsilon_{net}(\sigma_{net}) = \varepsilon_H(\sigma_H) + \sum_{h<H} \frac{1}{\gamma_c \tau_h} \int_{\sigma_h}^{\infty} \varepsilon_h(y) dy \,, \tag{4.50}$$

can be obtained from Eq. (4.48),

$$\begin{aligned}\mathcal{S}_{net}(t;\sigma_{net}) &= [\mathcal{S}_1 * \ldots * \mathcal{S}_H * \delta_{\tau_{net}}(t;\sigma_{net}) - (H-1)\gamma_c t]_+ \\ &\geq [\mathcal{S}_1 * \ldots * \mathcal{S}_H(t) - (H-1)\gamma_c t]_+ * \delta_{\tau_{net}}(t;\sigma_{net}) - (H-1)\gamma_c \tau_{net} \\ &= [\mathcal{S}_1(t) - (H-1)\gamma_c t]_+ * \ldots * [\mathcal{S}_H(t) - (H-1)\gamma_c t]_+ \\ &\qquad\qquad * \delta_{\tau_{net}}(t;\sigma_{net}) - (H-1)\gamma_c \tau_{net}, \qquad (4.51)\end{aligned}$$

where the second line holds since $[a+b]_+ \geq [a]_+ - [b]_+$ for any $b \geq 0$ and any a. The third line follows from the second line since for any functions f and g, and constant k, $f * g(t) - kt = (f(t) - kt) * (g(t) - kt)$. To compute a network service curve from Eq. (4.51), we need to characterize the elements in the bracket in that equation. Replacing the corresponding cross flow arrival envelopes in Theorem 3.1, a service curve and the corresponding bounding function for the through flow at node h for any $\sigma_h, \gamma_c \geq 0$, and fixed $\theta_h \geq 0$ is

$$\mathcal{S}_h(t;\sigma_h) = \Big[C_h t - [(\rho_h+\gamma_c)(t-\theta_h+\Delta^h(\theta_h))+\sigma_h]_+ I_{(t>\theta_h-\Delta(\theta_h))}\Big]_+ I_{t>\theta_h} \qquad (4.52)$$

$$\varepsilon_h(\sigma_h) = M_h e\Big(1+\frac{\rho_h}{\gamma_h}\Big)e^{-\alpha_h\sigma_h}\ . \qquad (4.53)$$

To simplify the convolution computations, we apply concavification as introduced in Sec. 3.6. We showed in Eqs. (3.51) and (3.52) that the loss of accuracy is not significant by concavification in high utilizations. Thus, we replace the service curve formulation in Eq. (4.52) by the following lower bound

$$\begin{aligned}\mathcal{S}_h(t;\sigma_h) &= \Big[C_h t - [(\rho_h+\gamma_c)(t-\theta_h+\Delta^h(\theta_h))+\sigma_h]_+\Big]_+ I_{t>\theta_h} \\ &= \Big[C_h(t+\theta_h) - [(\rho_h+\gamma_c)(t+\Delta^h(\theta_h))+\sigma_h]_+\Big]_+ I_{t>\theta_h} * \delta_{\theta_h}\ . \qquad (4.54)\end{aligned}$$

Define

$$U_h := (C_h - (H-1)\gamma_c)\theta_h - [(\rho_h+\gamma_c)\Delta^h(\theta_h)+\sigma_h]_+\ . \qquad (4.55)$$

Suppose that γ_c is chosen such that $C_h - \rho_h - H\gamma_c > 0$, then, U_h from the above equation is increasing in θ_h. We choose θ_h large enough such that $U_h \geq 0$. That is, $\theta_h \geq \theta_h^*$, where

$$\theta_h^* = \min\Big\{\frac{\sigma_h}{C_h-\rho_h-H\gamma_c}, \frac{[\sigma_h+(\rho_h+\gamma_c)\Delta^h]_+}{C_h-(H-1)\gamma_c}\Big\}\ . \qquad (4.56)$$

Then, from Eq. (4.54), we have

$$[\mathcal{S}_h(t;\sigma_h) - (H-1)\gamma_c t]_+ = \widetilde{\mathcal{S}}_h * \delta_{\theta_h}(t) \ , \tag{4.57}$$

where

$$\widetilde{\mathcal{S}}_h(t) = \min\{(C_h - (H-1)\gamma_c)(t+\theta_h), (C_h - \rho_h - H\gamma_c)t + U_h\} I_{t>0} \ . \tag{4.58}$$

If $U_h \geq 0$, then $\widetilde{S}_h(t)$ is a concave non-decreasing function for any $t > 0$. Combining all above, the network service curve in Eq. (4.51) can be rewritten as

$$\begin{aligned}
\mathcal{S}_{net}(t;\sigma_{net}) &= (\widetilde{\mathcal{S}}_1 * \delta_{\theta_1}) * \ldots * (\widetilde{\mathcal{S}}_H * \delta_{\theta_H}) * \delta_{\tau_{net}}(t;\sigma_{net}) - (H-1)\gamma_c \tau_{net} \\
&= \min_h (\widetilde{S}_h) * \delta_{\sum_{k=1}^{H} \theta_k + \tau_{net}}(t;\sigma_{net}) - (H-1)\gamma_c \tau_{net} \\
&= \min_h \Big\{ \min\{(C_h - (H-1)\gamma_c)(t+\theta_h), \\
&\qquad (C_h - \rho_h - H\gamma_c)t + U_h\} I_{t>0} \Big\} * \delta_{\sum_{k=1}^{H} \theta_k + \tau_{net}} - (H-1)\gamma_c \tau_{net},
\end{aligned} \tag{4.59}$$

where in the second line we use commutativity and associativity of the min-plus convolution operator, and the concavity of $\widetilde{S}_h$. In the last line, we insert the value of $\widetilde{S}_h$ from Eq. (4.58). The bounding function of this service curve from Eq. (4.50) is

$$\varepsilon_{net}(\sigma_{net}) = M_H e \Big(1 + \frac{\rho_H}{\gamma_c}\Big) e^{-\alpha_H \sigma_H} + \sum_{h=1}^{H-1} \frac{M_h C^{min}}{\gamma_c} e \Big(1 + \frac{\rho_h}{\gamma_c}\Big) e^{-\alpha_h \sigma_h}, \tag{4.60}$$

where $C^{min} = \min_k C_k$ and we choose τ_h in Eq. (4.50) as follows

$$\tau_h = \frac{1}{\alpha_h C^{min}} \tag{4.61}$$

and correspondingly

$$\tau_{net} = \sum_{h=1}^{H-1} \frac{1}{\alpha_h C^{min}} \ . \tag{4.62}$$

From Theorem 2.2, and since the through flow is EBB with parameter (M_0, ρ_0, α_0), d_{net} from the following is an end-to-end delay bound for the through flow arrivals

$$d_{net}(\sigma_{net}, \sigma_0) = \inf\{d \geq 0 \mid \forall t \geq 0 : \ \mathcal{S}_{net}(t+d;\sigma_{net}) \geq (\rho_0 + \gamma_0)t + \sigma_0\} \ , \tag{4.63}$$

in the sense that

$$P\{W_{net}(t) > d_{net}(\sigma_{net}, \sigma_0)\} \leq M_0 e\Big(1 + \frac{\rho_0}{\gamma_0}\Big) e^{-\alpha_0 \sigma_0} + M_H e\Big(1 + \frac{\rho_H}{\gamma_c}\Big) e^{-\alpha_H \sigma_H} + \sum_{h=1}^{H-1} \frac{M_h C^{min}}{\gamma_c} e\Big(1 + \frac{\rho_h}{\gamma_c}\Big) e^{-\alpha_h \sigma_h} . \quad (4.64)$$

Inserting $\mathcal{S}_{net}$ from Eq. (4.59) to the delay bound condition in Eq. (4.63), yields

$$\forall t > 0 : \ \min_h \Big\{ \min\{(C_h - (H-1)\gamma_c)(t + d + \theta_h - \sum_{k=1}^{H} \theta_k - \tau_{net}), (C_h - \rho_h - H\gamma_c)(t + d - \sum_{k=1}^{H} \theta_k - \tau_{net}) + U_h\} \Big\} I_{(t+d > \sum_{k=1}^{H} \theta_k + \tau_{net})} \geq (\rho_0 + \gamma_0)t + \sigma_0 + (H-1)\gamma_c \tau_{net}. \quad (4.65)$$

Since the minimum of concave functions is a concave function, the left-hand side of the above inequality is concave. Thus, to enforce Eq. (4.65) for any $t > 0$, we need to show that it holds for $t = 0^+$, and the long-term average rate of the concave function in the left hand side at $t \to \infty$ is not smaller than $\rho_0 + \gamma_0$. That is, for $t \to \infty$ we need to satisfy $\min_h\{C_h - \rho_h - H\gamma_c\} \geq \rho_0 + \gamma_0$. We need to choose $d \geq 0$ to be the minimum value that satisfies Eq. (4.65) at $t = 0^+$ with all the constraints mentioned above. From the indicator function in Eq. (4.65), a delay bound d must satisfy $d > \sum_{k=1}^{H} \theta_k + \tau_{net}$. With a change of variable $X := d - \sum_{h=1}^{H} \theta_h - \tau_{net}$, the constraint $d > \sum_{h=1}^{H} \theta_h + \tau_{net}$, will be translated into $X \geq 0$, and Eq. (4.65) results in the following optimization problem:

$$\min_{X, \theta_1, \theta_2, \ldots, \theta_H} d = X + \sum_{h=1}^{H} \theta_h + \tau_{net} \quad (4.66)$$

$$\text{s.t. } \forall h : \ U_h \geq 0 \quad (4.67)$$

$$\forall h : \ (C_h - (H-1)\gamma_c)(X + \theta_h) \geq \sigma_0 + (H-1)\gamma_c \tau_{net} \quad (4.68)$$

$$\forall h : \ (C_h - \rho_h - H\gamma_c)X + U_h \geq \sigma_0 + (H-1)\gamma_c \tau_{net} \quad (4.69)$$

$$X, \theta_1, \theta_2, \ldots, \theta_H \geq 0 . \quad (4.70)$$

Note that a feasible θ_h from the above constraints satisfies $\theta_h \geq \theta_h^*$ (where θ_h^* is formulated in Eq. (4.56)) since it must satisfy $U_h \geq 0$. The above optimization problem is convex if $\Delta^h < 0$

for all $1 \leq h \leq H$. However, the convexity does not hold if there exists one h with $\Delta^h \geq 0$, because of the inequality in Eq. (4.69). Although the problem is non-convex, we can still find the optimal solution. This is explained in Sec. 4.3.4.

To avoid going into the complexities of optimization problem formulated above, we make a proper choice to obtain a closed-form end-to-end result. We choose $\theta_h = \theta_h^*$ and compute closed-form performance bounds correspondingly in the following theorem.

Theorem 4.5 (Closed-form performance bounds [45])**.** *Consider the scenario depicted in Fig. 4.3, where a through flow passes through a tandem of H nodes. Each node h is a Δ-scheduler with Δ^h and capacity C_h. Through flow and cross flow (at node h) are EBB traffic, respectively, with parameters (M_0, ρ_0, α_0) and (M_h, ρ_h, α_h) satisfying Eq. (1.4). Define a vector of non-negative elements $\sigma_{net} = (\sigma_0, \sigma_1, \ldots, \sigma_H)$. For any γ_0 and γ_c satisfying $0 \leq \gamma_0 + H\gamma_c \leq \min_h\{C_h - \rho_h\}$, define the following:*

$$b_{net}(\sigma_{net}, \sigma_0) = (\rho_0 + \gamma_0 + (H-1)\gamma_c)\tau_{net} + \sigma_0 + (\rho_0 + \gamma_0)\sum_{h=1}^{H}\theta_h^* \,, \tag{4.71}$$

$$d_{net}(\sigma_{net}, \sigma_0) = \sum_{h=1}^{H}\theta_h^* + \tau_{net} + \max_{h=1,\ldots,H}\Big\{\max\Big\{\frac{\sigma_0 + (H-1)\gamma_c\tau_{net}}{C_h - (H-1)\gamma_c}, \frac{\sigma_0 + (H-1)\gamma_c\tau_{net} - [\sigma_h + (\rho_h + \gamma_h)\Delta^h]_-}{C_h - \rho_h - H\gamma_c}\Big\}\Big\}, \tag{4.72}$$

$$\varepsilon(\sigma_{net}, \sigma_0) = M_0 e\Big(1 + \frac{\rho_0}{\gamma_0}\Big)e^{-\alpha_0\sigma_0} + M_H e\Big(1 + \frac{\rho_H}{\gamma_c}\Big)e^{-\alpha_H\sigma_H} + \sum_{h=1}^{H-1}\frac{M_h C^{min}}{\gamma_c}e\Big(1 + \frac{\rho_h}{\gamma_c}\Big)e^{-\alpha_h\sigma_h} \,, \tag{4.73}$$

where $C^{min} = \min_h C_h$ and $\tau_{net} = \sum_{h=1}^{H-1}\frac{1}{\alpha_h C^{min}}$. Using the above notation, the end-to-end through flow output and backlog can be bounded as follows:

1- Output burstiness bound: *The network through flow departure D_{net} can be bounded statistically by the following for any $s \leq t$*

$$P\{D_{net}(s,t) > (\rho_0 + \gamma_0)(t-s) + b_{net}(\sigma_{net}, \sigma_0)\} \leq \varepsilon(\sigma_{net}, \sigma_0) \,. \tag{4.74}$$

2- Backlog bound: *The end-to-end through flow backlog* B_{net} *is statistically bounded by*

$$P\{B_{net}(t) > b_{net}(\sigma_{net}, \sigma_0)\} \leq \varepsilon(\sigma_{net}, \sigma_0) \ . \tag{4.75}$$

3- Delay bound: *The end-to-end through flow delay* W_{net} *is statistically bounded by*

$$P\{W_{net}(t) > d_{net}(\sigma_{net}, \sigma_0)\} \leq \varepsilon(\sigma_{net}, \sigma_0) \ . \tag{4.76}$$

Proof. Here we prove the correctness of backlog and delay bounds. The proof of the output burstiness bound follows similar steps as that of the backlog bound. With the EBB parameters for the through flow and from Eq. (2.53), $\mathcal{G}_0(t;\sigma_0) = (\rho_0+\gamma_0)t+\sigma_0$ is a statistical sample path envelope for any $\gamma_0 \geq 0$, with bounding function $\varepsilon_0(\sigma_0) = M_0\Big(1+\frac{\rho_0}{\gamma_0}\Big)e^{-\alpha_0\sigma_0}$. In addition, a statistical network service curve $\mathcal{S}_{net}(t;\sigma_{net})$ for such a scenario is as reported in Eq. (4.59) with bounding function $\varepsilon_{net}(\sigma_{net})$ as in Eq. (4.50). From Theorem 2.6, output burstiness, backlog, and delay bounds can be obtained with bounding function $\varepsilon_0(\sigma_0) + \varepsilon_{net}(\sigma_{net})$ which is equal to $\varepsilon(\sigma_{net}, \sigma_0)$ in Eq. (4.73).

The backlog bound in Theorem 2.6 is as follows:

$$b_{net}(\sigma_{net}, \sigma_0) = \sup_{t\geq 0}\{\mathcal{G}_0(t;\sigma_0) - \mathcal{S}_{net}(t;\sigma_{net})\} \ . \tag{4.77}$$

Insert $\mathcal{S}_{net}$ formulation in Eq. (4.59) with $\theta_h = \theta_h^*$ (Eq. (4.56)). Since $\mathcal{S}_{net}(t;\sigma_{net})$ is a concave function for any $t > \sum_{k=1}^{H}\theta_k^* + \tau_{net}$, and zero for $t > \sum_{k=1}^{H}\theta_k^* + \tau_{net}$. Moreover, $\mathcal{G}_0(t;\sigma_0)$ is a concave function for any $t > 0$, the supremum in Eq. (4.77) happens at $t = \sum_{k=1}^{H}\theta_k^* + \tau_{net}$ replacing the corresponding values in Eq. (4.77) proves the backlog bound in Eq. (4.75).

For the delay bound, we prove that d_{net} from Eq. (4.72) is a feasible solution to the optimization problem formulated in Eq. (4.66). Note that any delay bound must satisfy the constraints in Eqs. (4.67)-(4.70). Inserting this value of $\theta_h = \theta_h^*$ in Eq. (4.55), the corresponding value of U_h is

$$U_h^* = [\sigma_h + (\rho_h + \gamma_c)\Delta^h]_- \ . \tag{4.78}$$

The choice of θ_h^* confirms the validity of Eq. (4.67) and Eq. (4.70) except for $X \geq 0$. Thus, X must be chosen as the minimum value that satisfies all the remaining constraints in Eqs. (4.68-4.70). That is

$$\begin{aligned} X^* &= \max_h \Big\{ \max \Big\{ \frac{\sigma_0 + (H-1)\gamma_c \tau_{net}}{C_h - (H-1)\gamma_c} - \theta_h^*, \frac{[\sigma_0 + (H-1)\gamma_c \tau_{net} - U_h^*]_+}{C_h - \rho_h - H\gamma_c} \Big\} \Big\} \\ &= \max_h \Big\{ \max \Big\{ \frac{\sigma_0 + (H-1)\gamma_c \tau_{net}}{C_h - (H-1)\gamma_c} - \theta_h^*, \frac{[\sigma_0 + (H-1)\gamma_c \tau_{net} - [\sigma_h + (\rho_h + \gamma_c)\Delta^h]_-]_+}{C_h - \rho_h - H\gamma_c} \Big\} \Big\} \\ &= \max_h \Big\{ \max \Big\{ \frac{\sigma_0 + (H-1)\gamma_c \tau_{net}}{C_h - (H-1)\gamma_c}, \frac{[\sigma_0 + (H-1)\gamma_c \tau_{net} - [\sigma_h + (\rho_h + \gamma_c)\Delta^h]_-]_+}{C_h - \rho_h - H\gamma_c} \Big\} \Big\}, \end{aligned} \tag{4.79}$$

where in the last line, various cases might happen for θ_h^* and $[\sigma_0 + (H-1)\gamma_c\tau_{net} - U_h^*]_+$ depending on weather $\Delta \geq 0$ or $\Delta \geq 0$. Considering all cases, the third line can be obtained. The corresponding closed-form delay bound from Eq. (4.79) is

$$d_{net}(\sigma_{net}, \sigma_0) = \sum_{h=1}^{H} \theta_h^* + \tau_{net} + X^* \ .$$

Replacing the value of θ_h^* and X^*, respectively, from Eqs. (4.56) and (4.79) proves the theorem.

□

Discrete time analysis: Note that the discrete time version of the above bounds obtained by setting $\tau_{net} = 0$ for all h, and

$$\varepsilon(\sigma_{net}, \sigma_0) = \frac{M_0}{1 - e^{-\alpha_0 \gamma_0}} e^{-\alpha_0 \sigma_0} + \frac{M_H}{1 - e^{-\alpha_H \gamma_c}} e^{-\alpha_H \sigma_H} + \sum_{h=1}^{H-1} \frac{M_h}{(1 - e^{-\alpha_h \gamma_c})^2} e^{-\alpha_h \sigma_h} \ . \tag{4.80}$$

4.3.3 Leaky bucket arrivals in Δ-schedulers

The statistical performance bounds in previous section can be used as deterministic worst-case bounds by setting $\gamma_c, \gamma_0, \tau_{net} = 0$. We investigate the deterministic special case of our stochastic results from the previous sections by replacing EBB traffic sources with leaky bucket arrivals. Suppose that a traffic flow satisfying a deterministic arrival envelope $E_0(t) = \sigma_0 + \rho_0 t$ in the sense of Eq. (2.5) traverses a path of H links with capacities $C_1, \ldots, C_H$ as depicted in Fig. 4.4. Each link is shared by a cross flow shaped by a deterministic arrival envelope

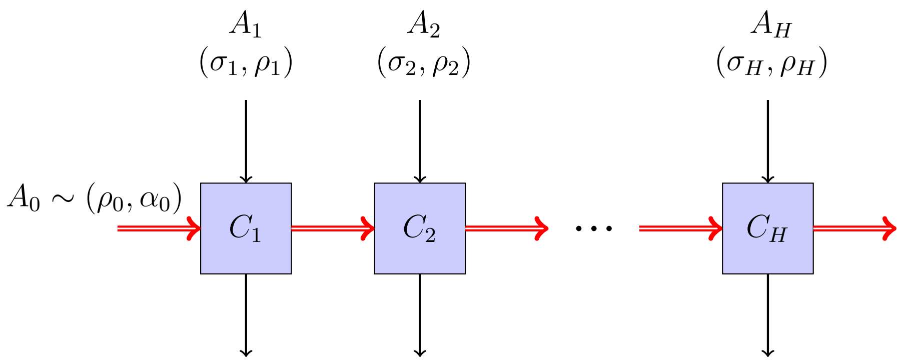


Figure 4.4: The multi-node leaky bucket arrivals scenario

$E_h(t) = \sigma_h + \rho_h t$. A Δ-scheduler with parameter Δ^h multiplexes the arrivals from the two flows.

4.3.4 Optimal solution

If the stability condition holds, that is if $\rho_0 \leq \min_h\{C_h - \rho_h\}$, then from Eq. (4.66), an end-to-end deterministic delay bound for through flow can be computed by solving the following optimization problem:

$$\min_{X,\theta_1,\theta_2,\ldots,\theta_H} d = X + \sum_{h=1}^{H} \theta_h \tag{4.81}$$

$$\text{s.t. } \forall h: \; U_h \geq 0 \tag{4.82}$$

$$\forall h: \; C_h(X + \theta_h) \geq \sigma_0 \tag{4.83}$$

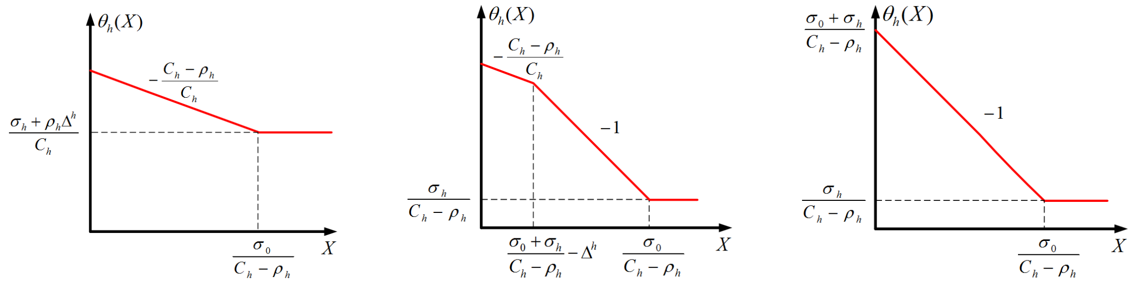
$$\forall h: \; (C_h - \rho_h)X + U_h \geq \sigma_0 \tag{4.84}$$

$$X, \theta_1, \theta_2, \ldots, \theta_H \geq 0, \tag{4.85}$$

where

$$U_h = C_h\theta_h - [\rho_h\Delta^h(\theta_h) + \sigma_h]_+ \,. \tag{4.86}$$

The variables in the optimization problem in Eq. (4.81) are X and the θ_h's. Fix $X \geq 0$, and set the $\theta_h^{op}(X)$ to be the minimum value of θ_h which satisfies all constraints in Eqs. (4.82)-(4.85). Inserting the value of $\theta_h^{op}(X)$ into Eq. (4.81), the objective function is represented as a



(a) $0 \leq \Delta^h < \frac{\sigma_h}{C_h-\rho_h}$ (b) $\frac{\sigma_h}{C_h-\rho_h} \leq \Delta^h < \frac{\sigma_0+\sigma_h}{C_h-\rho_h}$ (c) $\frac{\sigma_0+\sigma_h}{C_h-\rho_h} \leq \Delta^h$

Figure 4.5: $\theta_h^{op}(X)$ for $\Delta^h \geq 0$

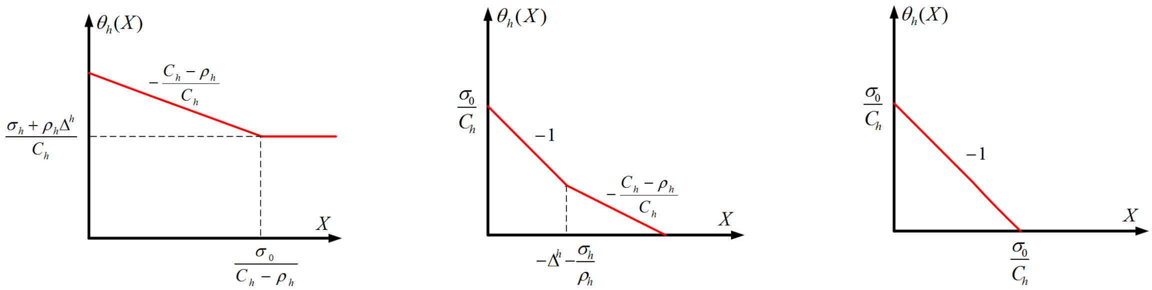
function of only one variable X. Then, we optimize the objective function over X. We obtain the value of $\theta_h^{op}(X)$ separately for $\Delta^h \geq 0$ and $\Delta^h < 0$.

• $\Delta^h \geq 0$: Since $\Delta^h \geq 0$, Eq. (4.84) always implies Eq. (4.83), and consequently we can eliminate Eq. (4.83). Thus, $\theta_h^{op}(X)$ for any $1 \leq h \leq H$ is the minimum value which satisfies inequalities (4.82), (4.84), and (4.85) simultaneously. $\theta_h^{op}(X)$ computed above is as plotted in Fig. 4.5. As illustrated in the plots, $\theta_h^{op}(X)$ varies with X and Δ^h. However, combining all cases, $\theta_h^{op}(X)$ can be also represented as follows

$$\theta_h^{op}(X) = \begin{cases} \frac{\sigma_0+\sigma_h+\rho_h\Delta^h}{C_h} - \frac{C_h-\rho_h}{C_h}X, & \text{if } X < \frac{\sigma_0}{C_h-\rho_h} + \min\{0, \frac{\sigma_h}{C_h-\rho_h} - \Delta^h\} \\ \frac{\sigma_0+\sigma_h}{C_h-\rho_h} - X, & \text{if } \frac{\sigma_0}{C_h-\rho_h} + \min\{0, \frac{\sigma_h}{C_h-\rho_h} - \Delta^h\} \leq X < \frac{\sigma_0}{C_h-\rho_h} \\ \min\{\frac{\sigma_h}{C_h-\rho_h}, \frac{\sigma_h+\rho_h\Delta^h}{C_h}\}, & \text{if } \frac{\sigma_0}{C_h-\rho_h} \leq X. \end{cases} \tag{4.87}$$

Eq. (4.87) is equivalent to the values of θ_h^{op} from Fig. 4.5. This is verified b comparing θ_h^{op} for fixed Δ^h and X from Fig. 4.5, and compare it with that from Eq. (4.87). The above equation shows that $\theta_h^*(X)$ for any $1 \leq h \leq H$ is a piece-wise linear function of X with at most two break points.

• $\Delta^h < 0$: If Δ^h is negative, then $\Delta^h(\theta_h) = \Delta^h$ and the optimization problem will be convex. Similar to the case of $\Delta^h \geq 0$, we must choose the minimum value of $\theta_h^{op}(X)$ such



(a) $0 \leq -\Delta^h < \frac{\sigma_h}{\rho_h}$ (b) $\frac{\sigma_h}{\rho_h} \leq -\Delta^h < \frac{\sigma_0}{C_h} + \frac{\sigma_h}{\rho_h}$ (c) $\frac{\sigma_0}{C_h} + \frac{\sigma_h}{\rho_h} \leq -\Delta^h$

Figure 4.6: $\theta_h^{op}(X)$ for $\Delta^h < 0$

that all inequalities (4.82)-(4.85) are satisfied which is

$$\theta_h^{op}(X) = \max\Big\{\frac{\sigma_h + \rho_h\Delta^h}{C_h}, \frac{\sigma_0}{C_h} - X, \frac{\sigma_0 + \sigma_h + \rho_h\Delta^h}{C_h} - \frac{(C_h - \rho_h)}{C_h}X, 0\Big\}. \tag{4.88}$$

$\theta_h^{op}(X)$ from the above equation will be a piece-wise linear function of X depending on the value of Δ^h as depicted in Fig. 4.6.

Figs. 4.5 and 4.6 show that $\theta_h^{op}(X)$ is a piece-wise linear function of X and can have at most two break points. Thus, the objective function, $X + \sum_{h=1}^{H} \theta_h^{op}(X)$, is a piece-wise linear function of X varies in $[0, \infty)$ with at most $2H$ break points. Consequently, it will take its minimum at $X = 0$ and ∞, or one of the break points. We construct a set $\mathcal{L}^*$ which is the set that includes the values of X's for which $X + \sum_{h=1}^{H} \theta_h^{op}(X)$ might be a local minimum. One of the elements of $\mathcal{L}^*$ is '0'. Note that since θ_h in all plots of Fig. 4.5 is a constant when $X \to \infty$, the objective function ($X + \sum_{h=1}^{H} \theta_h^{op}(X)$) cannot take its minimum exclusively at $X = \infty$. The local minimums of the objective function occur at the break points where the slope of at least one θ_h increases. As a result, we can eliminate the first break point in Fig. 4.5b for any h from $\mathcal{L}^*$. Considering all other break points in Figs. 4.5 and 4.6, for any h, there are at most two candidates to be an element in $\mathcal{L}^*$. We denote the corresponding set of candidates by X_h, which is

$$X_h = \begin{cases} \frac{\sigma_0}{C_h-\rho_h}, & \text{if } -\frac{\sigma_h}{\rho_h} < \Delta^h \\ -\Delta^h - \frac{\sigma_h}{C_h}, \frac{\sigma_0+\sigma_h+\rho_h\Delta^h}{C_h-\rho_h} & \text{if } -\frac{\sigma_h}{\rho_h} - \frac{\sigma_0}{C_h} < \Delta^h \le -\frac{\sigma_h}{\rho_h} \\ \frac{\sigma_0}{C_h} & \text{if } \Delta^h \le -\frac{\sigma_h}{\rho_h} - \frac{\sigma_0}{C_h}. \end{cases} \tag{4.89}$$

Thus,

$$\mathcal{L}^* = \bigcup_{h=1}^{H} \{X_h\} \cup \{0\}, \tag{4.90}$$

where each X_h from Eq. (4.89) can represent two elements. We can still eliminate some elements from $\mathcal{L}^*$. $\theta_h^{op}(X)$ can be either decreasing or constant with respect to X. Clearly, a fixed X cannot be a local optimal point if there exists a $\theta_h^{op}(X)$ which is proportional to X. Because in that case, by increasing X, the objective function $X + \sum_{h=1}^{H} \theta_h^{op}(X)$ does not increase. For this reason, if for any element in $\Big\{0, X_1, \ldots, , X_H\Big\}$, there exists a h such that $\theta_h^{op}(X)$ is proportional to $-X$, that element will be eliminated from $\mathcal{L}^*$. As a results, by Figs. 4.5 and 4.6, non-zero values of $\theta_h^{op}(X)$ belong to one of the following sets

$$\begin{aligned} \mathcal{O}(X) = \Big\{1 \le h \le H \mid \Big[-\Delta^h - \frac{\sigma_h}{\rho_h}\Big]_+ \le X \\ < \frac{\sigma_0 - [\sigma_h + \rho_h\Delta^h]_- - [\sigma_h - (C_h - \rho_h)\Delta^h]_-}{C_h - \rho_h}\Big\}, \end{aligned} \tag{4.91}$$

$$\mathcal{P}(X) = \Big\{1 \le h \le H \mid X \ge \frac{\sigma_0}{C_h - \rho_h}\Big\}, \tag{4.92}$$

where $[a]_- = \max\{0, -a\}$ for any real value of a. Correspondingly, the solution to the optimization problem in Eq. (4.81)-(4.85) is

$$\begin{aligned} d^* = \min_{X \in \mathcal{L}^*} \Big\{X + \sum_{h \in \mathcal{O}(X)} \Big(\frac{\sigma_0 + \sigma_h + \rho_h\Delta^h}{C_h} - \frac{C_h - \rho_h}{C_h} X\Big) \\ + \sum_{h \in \mathcal{P}(X)} \min\Big\{\frac{\sigma_h}{C_h - \rho_h}, \frac{[\sigma_h + \rho_h\Delta^h]_+}{C_h}\Big\}\Big\}. \end{aligned} \tag{4.93}$$

4.3.5 Closed-form solution

As shown in the previous section, a delay bound can be obtained by solving the optimization problem in Eqs. (4.81)-(4.85). That delay bound is obtained by taking the minimum value of at most $2H+1$ local optimum points. As the path length H increases, the set of local optimal points grows and makes the computation of the optimal solution more difficult. The following theorem is a special case of Theorem 4.5 and provides closed-form deterministic performance bounds by setting $\gamma_c, \gamma_0, \tau_{net} = 0$.

Theorem 4.6 ([45]). *Keep all the assumptions and notation for the leaky bucket arrivals and tandem network mentioned above. If the stability condition $\rho_0 \le \min_h\{C_h - \rho_h\}$ holds, then the following bounds exist for the through flow*

1- Output burstiness bound: *The through flow departure process satisfies the following for any $s \le t$*

$$D_{net}(s,t) \le \Big(\sigma_0 + \rho_0 \sum_{h=1}^{H} \theta_h^{cf}\Big) + \rho_0(t-s), \tag{4.94}$$

2- Backlog bound: *The through flow end-to-end backlog satisfies the following for any $t \ge 0$*

$$B_{net}(t) \le \sigma_0 + \rho_0 \sum_{h=1}^{H} \theta_h^{cf}, \tag{4.95}$$

3- Delay bound: *End-to-end through flow delay satisfies the following for any $t \ge 0$*

$$W_{net}(t) \le d^{cf}, \tag{4.96}$$

where

$$d^{cf} = \max_{h=1,\dots,H}\Big\{\max\Big\{\frac{\sigma_0}{C_h}, \frac{\sigma_0 - [\sigma_h + \rho_h \Delta^h]_-}{C_h - \rho_h}\Big\}\Big\} + \sum_{h=1}^{H} \theta_h^{cf}\ , \tag{4.97}$$

and

$$\theta_h^{cf} := \min\Big\{\frac{\sigma_h}{C_h - \rho_h}, \frac{[\sigma_h + \rho_h \Delta^h]_+}{C_h}\Big\}. \tag{4.98}$$

In the following we point out some remarks from [13] on the above theorem.

• **Remark 1**: d^{cf} is increasing in Δ^h.

• **Remark 2**: The end-to-end delay bound in Theorem 4.6 can be simplified for homogeneous networks where all scheduling algorithms are identical (i.e. $\Delta^h = \Delta$). Consider the three following special cases.

$$d^{cf} = \begin{cases} \frac{\sigma_0}{\min_h\{C_h-\rho_h\}} + \sum_{h=1}^{H} \frac{\sigma_h}{C_h-\rho_h}, & \Delta = +\infty, \ \text{(lowest priority)} \\ \frac{\sigma_0}{\min_h\{C_h-\rho_h\}} + \sum_{h=1}^{H} \frac{\sigma_h}{C_h}, & \Delta = 0, \quad \text{(FIFO)} \\ \frac{\sigma_0}{\min_h\{C_h\}}, & \Delta = -\infty, \ \text{(highest priority)}. \end{cases} \tag{4.99}$$

• **Remark 3**: For any $\Delta^h < 0$ and ρ_h sufficiently large, the delay bound reduces to $d^{cf} = d^{cf}_{fifo} + \sum_{h=1}^{H} \frac{\rho_h}{C_h}\Delta^h$, where d^{cf}_{fifo} is the second line in Eq. (4.99).

• **Remark 4**: If $\Delta^h \geq 0$ for any h, then the through flow burstiness σ_0 is served by the bottleneck link, i.e., $\min_h\{C_h - \rho_h\}$. If $\Delta^h \leq -(\frac{\sigma_0}{C_h} + \frac{\sigma_h}{\rho_h})$ for any h, then the through flow burstiness will be served by the minimum capacity, i.e., $\min_h C_h$.

In the following section we investigate the tightness of the closed-form bounds by computing some lower bounds achievable delays.

4.3.6 Tightness of bounds

We have computed two deterministic end-to-end delay bounds (d^* in Eq. (4.93) and d^{cf} in Eq. (4.97)) for a leaky bucket through flow arrival in a tandem of Δ-schedulers with leaky bucket cross flows. d^* and d^{cf} hold as upper bounds for any arrival pattern that satisfies the leaky bucket constraints. The maximum achievable end-to-end delay among all scenarios is represented by d^{max} and the corresponding arrival pattern is known as the worst-case scenario. By the above notation and since d^* is minimized over parameter θ_h while d^{cf} is obtained by a specific choice of θ_h, we have

$$d^{max} \leq d^* \leq d^{cf}. \tag{4.100}$$

The delay bound d^* or d^{cf} are said to be tight if $d^{max} = d^*$ or $d^{max} = d^{cf}$.

To evaluate the tightness of the bounds we consider an arrival pattern (not necessarily the worst-case scenario) and compute the corresponding maximum delay in that scenario to use it as a lower bound on d^{max}. In the following, we describe our chosen traffic pattern which conform to leaky-bucket arrivals, with parameters (σ_0, ρ_0) for the through flow and (σ_h, ρ_h) for the cross flow at node h.

Arrival Patterns: Assume that the through flow starts at $t = -\infty$ with rate ρ_0, added by a burst of size σ_0 at time $t = 0$. The cross flow at node h starts at $t = -\infty$ with rate ρ_h added by a burst of size σ_h at time $t = \omega_h$, where

$$\omega_h = \begin{cases} T_h^{ar} - \nu & \text{if } \Delta^h \geq 0 \\ T_h^{ar} + \Delta^h - \nu & \text{if } \Delta^h < 0 \ , \end{cases} \tag{4.101}$$

T_h^{ar} is the arrival time of the first bit of through flow burst at node h, $T_1^{ar} = 0$, and $\nu > 0$ is smaller than any positive number, i.e., $\nu \to 0^+$.

Theorem 4.7 (Lower performance bounds)**.** *Keep all the assumptions and notation above. For the arrival pattern described above, the maximum end-to-end delay and backlog that the through flow will experience is lower bounded by the following*

1- Backlog lower bound:

$$b^{max} \geq \sigma_0 + \rho_0 \sum_{h=1}^{H} L_h, \tag{4.102}$$

2- Delay lower bound:

$$d^{max} \geq \min\Big\{\frac{\sigma_0}{\min_h C_h}, \min_h O_h\Big\} + \frac{\big[\sigma_0 - \max_h O_h . \min_h C_h\big]_+}{R_{H+1}} + \sum_{h=1}^{H} L_h, \tag{4.103}$$

where

$$L_h = \min\Big\{\frac{\sigma_h}{C_h - \rho_h}, \frac{[\sigma_h + \rho_h[\Delta^h]_+ - (C_h - \rho_0)[\Delta^h]_-]_+}{C_h}\Big\}, \tag{4.104}$$

$$O_h = \min\Big\{[\Delta^h]_-, \frac{[(C_h - \rho_0)[\Delta^h]_- - \sigma_h]_+}{\rho_h}\Big\}, \tag{4.105}$$

$$R_{h+1} = \frac{R_h C_h}{R_h + \rho_h}, \tag{4.106}$$

and $R_2 = C_1$.

Remark 1: The lower bound for the maximum end-to-end delay in Eq. (4.103) consists of three terms. The first two terms are lower bounds on the time needed to serve the through flow burst, respectively, by the bottleneck capacity and capacity sharing according to CBR arrivals in FIFO schedulers. The last term in Eq. (4.103) is a lower bound on the delay induced by cross flow exclusively occupying the whole capacity after T_h^{ar}.

Proof. We compute the end-to-end delay and backlog that the last bit of through flow burst will experience. We start by proving Eq. (4.102). After the first bit of through flow burst arrives to node h, it must wait for a period of L_h until it is transmitted. In fact, both terms in L_h (Eq. (4.104)) correspond to the cases that the capacity is fully devoted to the cross flow. This is shown in the following by considering the case of $\Delta^h \geq 0$ and $\Delta^h < 0$, separately.

If $\Delta^h \geq 0$, then from Eq. (4.104), $L_h = \min\{\frac{\sigma_h}{C_h - \rho_h}, \frac{\sigma_h + \rho_h \Delta^h}{C_h}\}$. The first term in the minimum is the maximum time interval to deplete cross flow backlog, and the second term is the time needed to serve all cross flow arrivals with higher precedence than the through flow burst. If $\Delta^h < 0$, then from Eq. (4.104), $L_h = \frac{[\sigma_h + (C_h - \rho_0)\Delta^h]_+}{C_h}$. In this case, the cross flow burst at node h occurs at $T_h^{ar} + \Delta - \nu$, which is $[\Delta]_-$ time unit before the arrival of the first bit of through flow burst. Any through flow arrival in $[T_h^{ar} + \Delta, T_h^{ar})$ has higher precedence than the cross flow arrivals in the same interval. Since through flow in that interval is CBR with rate ρ_0, it takes $\frac{\sigma_h}{C_h - \rho_0}$ to serve the cross flow burst at node h. The remaining cross flow burst $[\sigma_h + (C_h - \rho_0)\Delta^h]_+$ creates a delay of size L_h to the first bit of through flow burst at node h. Hence, the first bit of through flow burst is delayed by $\sum_{h=1}^{H} L_h$ which creates a through flow backlog of size $\rho_0 \sum_{h=1}^{H} L_h$. Adding this built-up backlog to the through flow burst σ_0, yields a backlog of size $\sigma_0 + \rho_0 \sum_{h=1}^{H} L_h$ which verifies Eq. (4.102).

The lower delay bound in Eq. (4.103) consists of three terms. The last term is the delay due to the exclusive allocation of the whole capacity at each node to the cross flow upon the arrival of the first bit of through flow burst which is $\sum_{h=1}^{H} L_h$ and discussed above. The first two terms correspond to the time needed to serve the through flow burst in the network. We compute a lower bound on the first two terms, separately, for $\Delta^h \geq 0$ and $\Delta^h < 0$.

• $\Delta^h \geq 0$ **:** If $\Delta^h \geq 0$ for any h, the end-to-end delay in a tandem of Δ-scheduler is lower bounded by that of a tandem of FIFO schedulers. In a FIFO system, the through flow burst departs the first node with rate C_1 after the cross flow burst is served. At the second node, through flow burst waits for L_2 seconds. Then, through flow must compete with a CBR cross flow with rate ρ_2 to be served with a constant rate C_2. Thus, the through flow arrival rate to the third node is $R_3 = \frac{R_2 C_2}{R_2+\rho_2}$, where $R_2 = C_1$. Continuing this argument, we have $R_{h+1} = \frac{R_h C_h}{R_h+\rho_h}$. Since from Eq. (4.105), $O_h = 0$ for $\Delta^h \geq 0$, we have proved the theorem for $\Delta^h \geq 0$.

• $\Delta^h < 0$ **:** Cross flow arrivals before T_h^{ar} have lower precedence than the first bit of through flow burst at node h. Some of that traffic is served before the through flow burst arrival with rate $(C_h - \rho_0)$. This is happening only if the whole cross flow burst is served before the arrival of the first bit of through flow, i.e., if $\frac{\sigma_h}{C_h-\rho_0} < [\Delta^h]_-$. The total CBR cross flow in that interval is $\rho_h[\Delta^h]_-$. Thus, the amount of lower precedence cross flow arrivals which has been served before the arrival of the first bit of through flow burst is $\min\{\rho_h[\Delta^h]_-, [(C_h - \rho_0)[\Delta^h]_- - \sigma_h]_+\}$. Consequently, the whole capacity is allocated to the through flow at most for a duration of $O_h = \min\Big\{[\Delta^h]_-, \frac{[(C_h-\rho_0)[\Delta^h]_- - \sigma_h]_+}{\rho_h}\Big\}$ after the arrival of the first through flow bit. In an end-to-end view, the time needed to serve σ_0 by the bottle-neck capacity is $\frac{\sigma_0}{\min_h C_h}$. If this time exceeds the time needed to serve through flow burst with the bottleneck capacity, (i.e., if $\frac{\sigma_0}{\min_h C_h} > \min_h O_h$), the remaining through flow burst which is lower bounded by $[\sigma_0 - \max_h O_h . \min_h C_h]_+$ is served similar to CBR arrivals in a FIFO scheduler, i.e., by sharing the capacity according to the rates. Collecting all of the above results proves the validity of the end-to-end lower delay bound in Eq. (4.103).

□

Using the above theorem as a lower delay bound benchmark for the delay, and comparing it with the closed-form delay bound from Eq. (4.97), we are able to evaluate the tightness of our upper-bounds. The following remarks discuss that.

Remark 1 : In general,

$$d^{cf} - d^{max} \leq \sigma_0\Big(\frac{1}{\min_h(C_h - \rho_h)} - \frac{1}{\min_h C_h}\Big) + \sum_{h=1}^{H} C_h[\Delta^h]_-(1 - u_h), \tag{4.107}$$

where $u_h = \frac{\rho_0+\rho_h}{C_h}$ is the total utilization at node h.

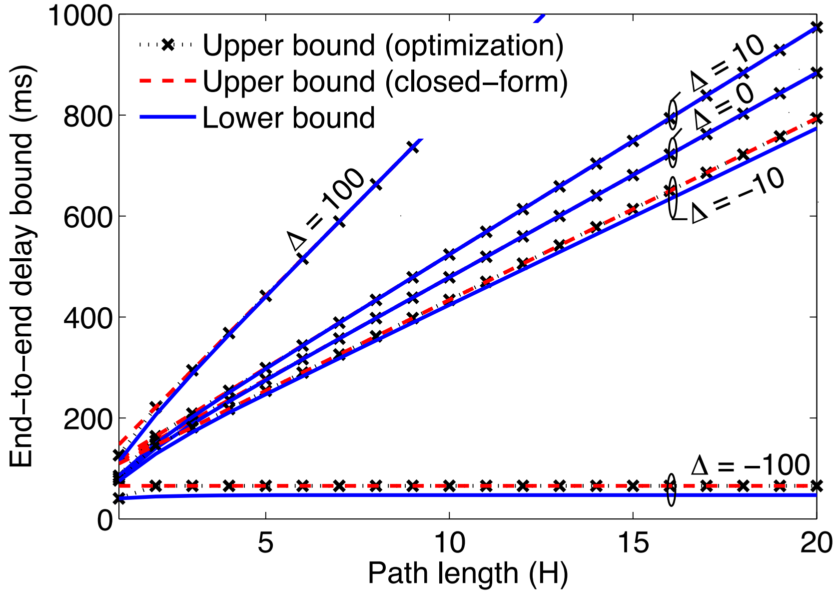


Figure 4.7: Comparing upper and lower delay bounds for leaky bucket arrivals with $\rho_h = 0.15$ Mbps and $\sigma_h = 13.5$ Kbits for any $h = 0, \ldots, H$, $C = 100$ Mbps, and $U = 90\%$.

Remark 2: Eq. (4.107) shows that d^{cf} is tighter in higher utilization.

Remark 3: In case of homogeneous networks and $\Delta \geq 0$, the closed-form bounds are tight if $H \to \infty$. This can be noticed from Eq. (4.103), and noting that $\lim_{H\to\infty} R_H = C - \rho_c$.

Remark 4: If $\Delta^h \geq 0$ or $\Delta^h \leq \frac{\sigma_h}{C_h - \rho_h}$ for any h, the difference between the closed-form bound and tight bound is upper bounded by a constant and does not scale with H.

Remark 5: In case of homogeneous networks, where $\rho_h = \rho_c$ and $C_h = C$,

$$d^{cf} - d^* \leq \frac{\sigma_0}{C - \rho_c}\Big[1 - \frac{H(C - \rho_c)}{C}\Big]_+ . \tag{4.108}$$

One of the immediate consequences is that $d^{cf} = d^*$ for any value of Δ if $H > \frac{C}{C - \rho_c}$.

Finally, with a numerical example we evaluate the tightness of the closed-form and the solution to the optimization problem. Consider the network scenario in Fig 4.4. There are $N_0 = 300$ through flows, and $N_c = 300$ cross flows. Each through and cross flow is a leaky bucket arrival with $\sigma_0 = \sigma_c = 13.5$ Kbits and $\rho_0 = \rho_c = 0.15$ Mbps. The capacity of any node h is $C_h = 100$ Mbps. We have plotted the lower bound computed for d^{max} from Eq. (4.103), d^*

from Eq. (4.93), and d^{cf} from Eq. (4.97) for different values of Δ as a function of path length H in Fig. 4.7.

As illustrated in the plot, the lower bounds converge to the upper bounds for any $\Delta \geq 0$ as the path length increases. This implies that even the closed-form bounds might be tight for larger network sizes.

4.3.7 Numerical results

We consider the multi-node scenario depicted in Fig. 4.3. We choose ON-OFF Markov-Modulated Poisson Processes (MMPP) sources for both through and cross flows. An MMPP flow can be modeled by a simple two-state Markov chain: ON, OFF. Traffic is generated with rate P in ON state and no traffic is generated in OFF state. The sojourn time between ON to OFF and OFF to ON are exponentially distributed with respective rates λ and μ. The average cycle time to return to the same state is $T^* = \frac{\lambda+\mu}{\lambda\mu}$. The effective bandwidth for this arrival is formulated in [55]. For any $\alpha \geq 0$

$$Eb(\alpha) = \frac{1}{2\alpha}(P\alpha + \lambda + \mu + \sqrt{(P\alpha + \lambda - \mu)^2 + 4\mu\lambda}) \ .$$

The discrete time MMP traffic introduced in Fig. 2.9 can be considered as a discrete time approximation of MMPP traffic, where $p_{11} = (1 - \lambda t_u)$, $p_{00} = (1 - \mu t_u)$ and t_u is the time unit in the discrete time model. In our numerical examples, $\lambda = 1ms^{-1}$ and $\mu = 0.11ms^{-1}$ which corresponds to the average rate of $\rho_{av} = 0.15$ Mbps and $T^* = 10$ ms. These parameters are also used for the numerical results in [20]. Unless otherwise stated, we consider a homogeneous network where all parameters at all nodes are identical. Since cross flows are also identical at all nodes $h = 1, \ldots, H$, we drop the subscript h and use subscript c to refer to the cross flow parameters. The capacities of all nodes are assumed to be $C = 100$ Mbps. There are $N_0 = 10$ through flows and $N_c = 590$ cross flows. The above parameters yield a utilization of $U = (N_0 + N_c)\frac{0.15}{100} = 90\%$ at all nodes. The violation probability in all examples is assumed to be $\varepsilon^* = 10^{-9}$. The numerical results use the closed-form bounds from Theorem 4.5 with the

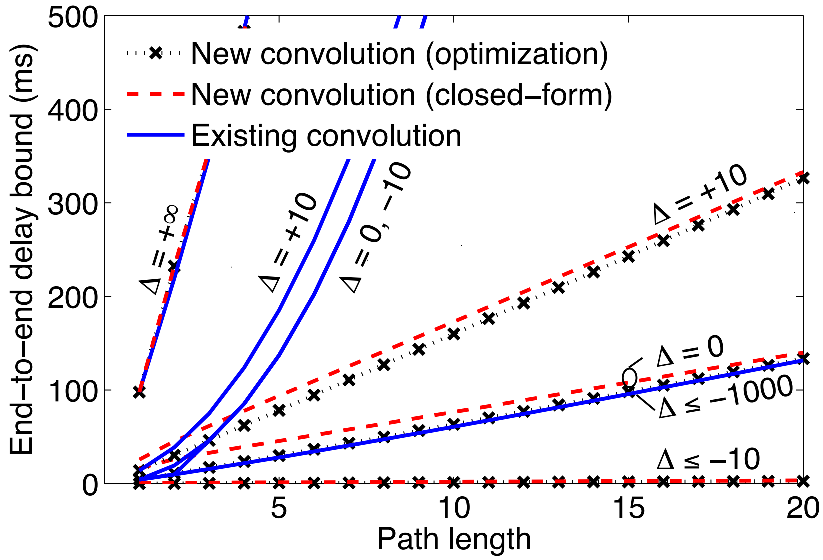


Figure 4.8: Statistical end-to-end delay bounds as a function of path length for MMPP traffic arrivals with $P = 1.5$ Mbps, $T^* = 10$ ms, $\varepsilon^* = 10^{-9}$, $N_c = 590$, $N_0 = 10$, and $U = 90\%$.

following choices of parameters. We choose α_0 and α_h such that the stability constraint is not violated, i.e.,

$$\rho_0(\alpha_0) < \min_{h=1,\ldots,H} \{C_h - \rho_h(\alpha_h)\} , \tag{4.109}$$

Set $\gamma_c, \gamma_0 = \gamma$, where

$$0 < (H+1)\gamma < \min_{h=1,\ldots,H} \{C_h - \rho_h - \rho_0\} . \tag{4.110}$$

Moreover, we compute σ_h for any h from the following equations

$$\alpha_{net} = \left(\sum_{h=0}^{H} \alpha_h^{-1}\right)^{-1}, \ \sigma_h = \frac{\alpha_{net}}{\alpha_h}\sigma, \qquad h = 0, \ldots, H ,$$

where σ can be obtained by fixing the violation probability from Eq. (4.73).

• **The improvement by the new convolution:**

In this experiment, we compute the end-to-end delay bounds for the through flow using Theorem 2.7 and compare it with the bounds computed by Theorem 4.4. We include the delay bounds obtained both from the solution to the optimization problem and the closed-form bound

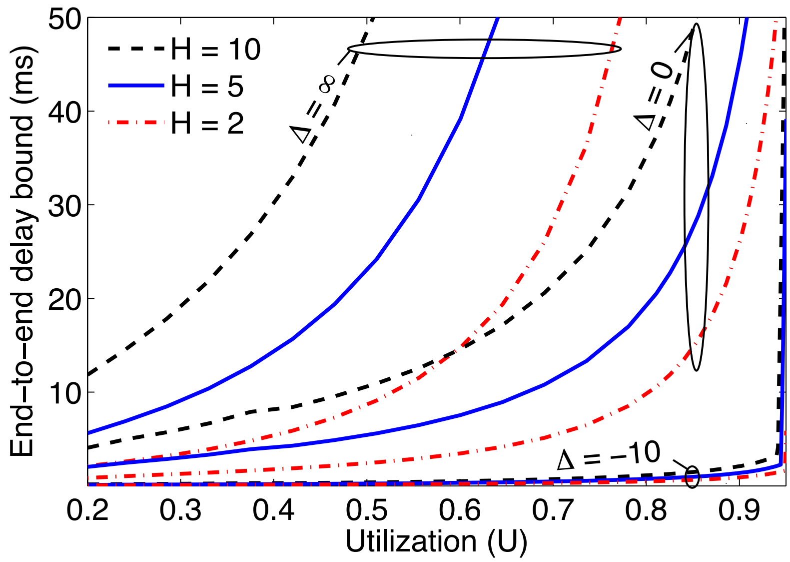


Figure 4.9: Statistical end-to-end delay bounds as a function of total utilization U for MMPP traffic arrivals with $P = 1.5$ Mbps, $T^* = 10$ ms, $\varepsilon^* = 10^{-9}$, $N_c = 590$, $N_0 = 10$, $H = 2, 5, 10$, $U_0 = 15\%$, and $\Delta = -10, 0, \infty$.

on a path of $H = 1, \ldots, 20$ identical Δ-schedulers. We compute the example for different values of Δ.

Fig. 4.8 shows that for $\Delta = \infty$ (BMux), the new convolution has no gain, or, in other words, the existing convolution theorem is suitable for BMux analysis. However, considerable improvement is gained for $\Delta < \infty$. For example, for FIFO networks ($\Delta = 0$), the curve for the new convolution theorem is much smaller than that of the existing one. For small values of H, the end-to-end delay bound corresponding to the existing convolution theorem is slightly smaller than that of the new convolution theorem. This can be justified by the concavification of the service curve that we applied to simplify the end-to-end analysis with the new convolution theorem.

The difference between the closed-form bounds and the solution to the optimization problem is not significant. Fig. 4.8 also illustrates that the importance of the schedulers does not deteriorate as the path length increases. This figure indicates that the impact of the scheduling

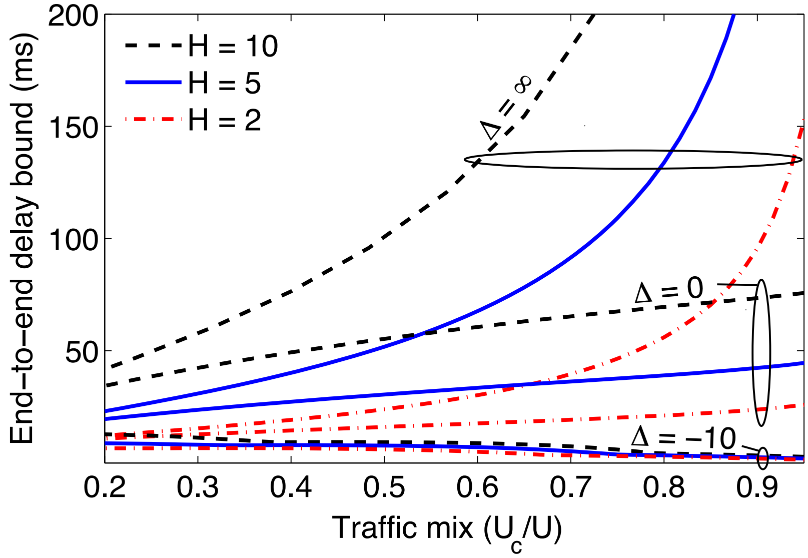


Figure 4.10: Statistical end-to-end delay bounds as a function of traffic mix $\left(\frac{U_c}{U}\right)$ for MMPP traffic arrivals with $P = 1.5$ Mbps, $T^* = 10$ ms, $\varepsilon^* = 10^{-9}$, $N_c = 590$, $N_0 = 10$, $U = 90\%$, $H = 2, 5, 10$, and $\Delta = -10, 0, \infty$.

scales with the network size in the stable regime.

- **The effect of scheduling as a function of utilization:**

The role of scheduling is more pronounced if through flow competes with more cross flows, or in other words, if through flow is a smaller fraction of the total flows in a link. We examine this with our delay bounds in Fig. 4.9 by fixing the through flow utilization at 15% and increasing the total utilization from 20% to 95%. We repeat the experiment for three schedulers: BMux, FIFO, and $\Delta = -10$ ms, and for three different path lengths: $H = 2, 5, 10$. The case of $\Delta = -10$ is equivalent to an EDF scheduler which assigns a priori delay bounds d_0^* and d_c^*, respectively, to the through flow and cross flow such that $d_0^* - d_0^* = 10$ ms. The most apparent result of this plot is that the difference between schedulers increases for higher utilization and path lengths. There are some additional interesting observations which can be extracted from this figure.

This experiment shows that the end-to-end delay bound in a cascade of Δ-schedulers with

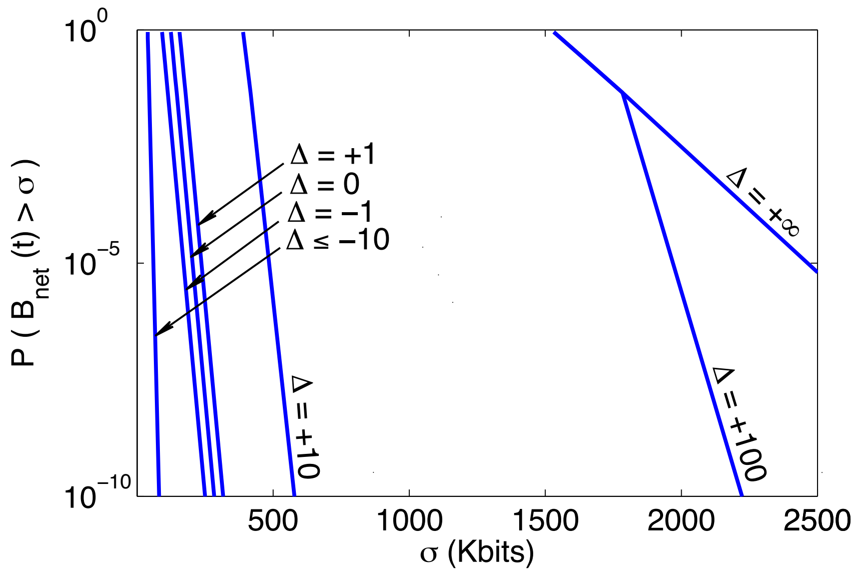


Figure 4.11: Tail bound of backlog distribution (Kbits) for MMPP traffic arrivals with $P = 1.5$ Mbps, $T^* = 10$ ms, $\varepsilon^* = 10^{-9}$, $N_c = 590$, $N_0 = 10$, $U = 90\%$, and $H = 10$.

$\Delta = -10$ is almost insensitive to the path length unless the utilization is very large. This occurs since with $\Delta = -10$, the scheduler behaves similar to a priority scheduler which assigns the highest priority to the through flow. Thus, the through flow can occupy the whole link capacity when needed and does not compete with cross flows. As a result, the original statistical properties of the through flow does not change as it passes through the scheduler.

The delay bound increases fast in utilization for BMux, because, as cross flow utilization increases, the likelihood that a through flow arrival encounters backlog with higher precedence increases. In addition, while a backlogged through flow is waiting to be served, any cross flow arrival will also be scheduled ahead of through flow to be served. Through flow end-to-end delay bound in a tandem of FIFO schedulers also increases with total utilization, but not as fast as that of BMux. This is happening because, by increasing the cross flow utilization, a tagged through flow arrival will experience a larger cross flow backlog as it arrives to the scheduler. However, while the tagged arrival is waiting to be served in the buffer, cross flow new arrivals will not receive higher precedence than the tagged arrival.

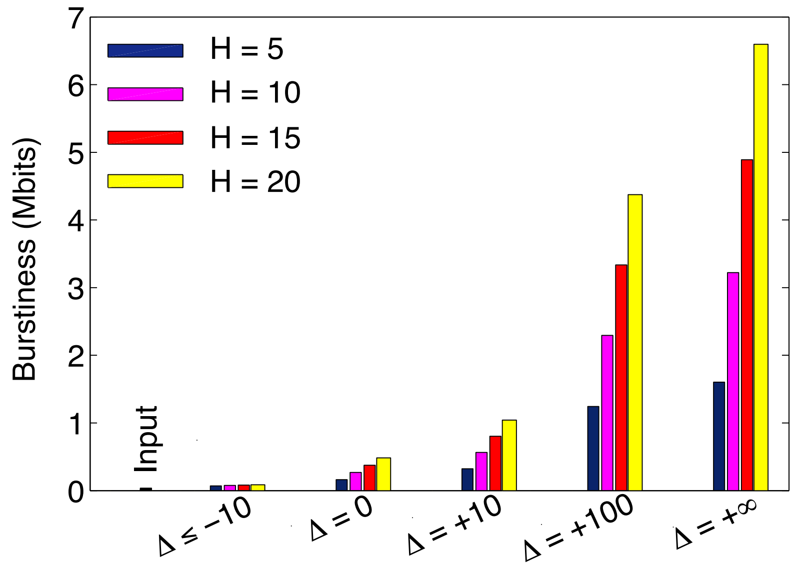


Figure 4.12: Burst size of the output traffic for MMPP traffic arrivals with $P = 1.5$ Mbps, $T^* = 10$ ms, $\varepsilon^* = 10^{-9}$, $N_c = 590$, $N_0 = 10$, and $U = 90\%$.

• The effect of scheduling as a function of traffic mix:

In this experiment, we fix the total utilization to 90% and increase cross flow utilization. The closed-form end-to-end delay bound for through flow is plotted by varying $\frac{U_c}{U}$ from 20% to 95%. This experiment is repeated for three schedulers: BMux, FIFO, and $\Delta = -10$, and for three different path lengths: $H = 2, 5, 10$. One of the interesting observations in Fig. 4.10 is that through flow delay bound increases for BMux, it is almost insensible for FIFO, while decreases for $\Delta = -10$ as $\frac{U_c}{U}$ increases. This is happening because increasing cross flow and decreasing through flow increases the likelihood that through flow encounters larger backlogged traffic with higher precedence if $\Delta > 0$. For FIFO, the through flow and cross flows have the same precedence level if they arrive at the same time. Any arrival has a lower precedence than any backlogged traffic regardless of which flow they belong to. For $\Delta < 0$, through flow has higher precedence than cross flow if they arrive at the same time. As a result, decreasing through flow and increasing cross flow increases the chance that an arriving through flow encounters smaller backlogged traffic with higher precedence.

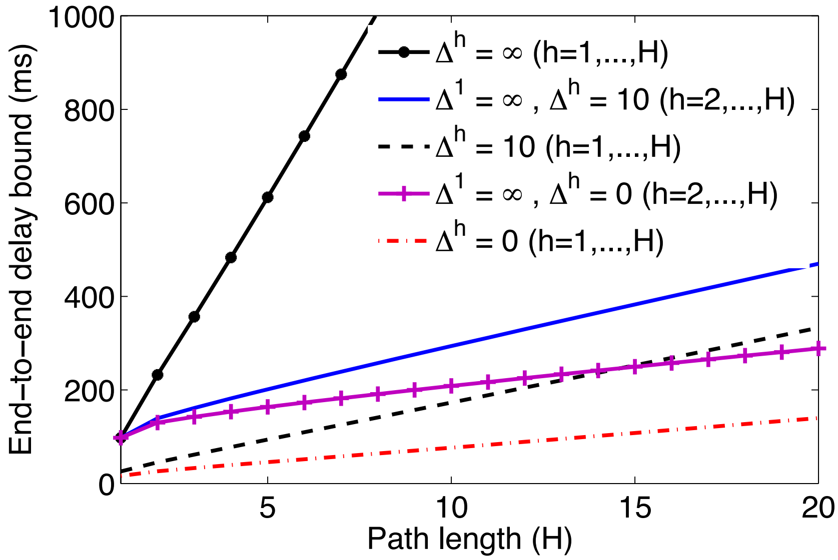


Figure 4.13: Statistical end-to-end delay bounds in a cascade of different schedulers as a function of path length for MMPP traffic arrivals with $P = 1.5$ Mbps, $T^* = 10$ ms, $\varepsilon^* = 10^{-9}$, $N_c = 590$, $N_0 = 10$, and $U = 90\%$.

• **Backlog tail bound:**

For a path length of $H = 10$ nodes, we compute the violation probability that the backlog exceeds a threshold σ. We use Eq. (4.75) to compute the bounds. The tail bound of the end-to-end through flow backlog from Eq. (4.75) is depicted in Fig. 4.11. The plot shows that for any $\Delta < 0$ and small $\Delta \geq 0$ ($\Delta \leq 1$), the tail bound behaves very closely to that of SP with the highest priority assigned to the through flow ($\Delta = -\infty$). Thus, a large class of Δ-scheduler have small end-to-end backlog and they behave similarly. However, for larger values of Δ, increasing Δ affects the end-to-end backlog bound considerably, e.g., compare the corresponding curves for $\Delta = 10$ and $\Delta = 100$ in Fig. 4.11.

• **Burstiness increase:**

In this part, we exemplify the burstiness increment as the through flow proceeds along the path. Fixing the violation probability to $\varepsilon^* = 10^{-9}$, we compare the burstiness bound of the input traffic with that of the output. For various path lengths $H = 5, 10, 15, 20$, and various Δ

parameters the experiment is repeated. Fig. 4.12 shows that the burstiness does not increase noticeably for negative or small values of Δ even on long paths. This implies that the original statistics of the input traffic are preserved as traffic traverses some schedulers including FIFO. This observation (if valid) is the key which enables per-node analysis in a network. This is considered in details in the next chapter.

For larger values of Δ, the output burstiness is much larger than that of the input. Thus, for these schedulers, the original traffic properties is distorted and the traffic burstiness increases. Per-node delay and backlog analysis for this class of schedulers cannot be conducted regardless of the effect of network.

• **Heterogeneous network:**

In the previous examples, all schedulers along the paths were assumed to be identical. In this experiment, we want to see the effect of having one bottleneck scheduler on a long path. We have accommodated five curves in Fig. 4.13. Three of these curves correspond to homogeneous networks which serve as benchmarks: a path of all BMux '$\Delta^h = \infty\,(h = 1, \ldots, H)$', all FIFO schedulers '$\Delta^h = 0\,(h = 1, \ldots, H)$', and '$\Delta^h = 10\,(h = 1, \ldots, H)$'. We consider two additional curves which consider a heterogeneous network. For one of them, the first link in the path is a BMux, and the rest of the links are FIFO schedulers. The other one considers a BMux as the first link and the next links are Δ-schedulers with $\Delta^h = 10$. Comparing the curve labeled as '$\Delta^1 = \infty, \Delta^h = 0\,(h = 2, \ldots, H)$' with all FIFO curve, and '$\Delta^1 = \infty, \Delta^h = 10\,(h = 2, \ldots, H)$' with '$\Delta^h = 10\,(h = 1, \ldots, H)$' we realize the effect of one bottleneck scheduler still exists even on long paths, but the difference remains unchanged as the path length increases (comparable to the difference in a single node scenario).

Chapter 5

Network Decomposition

In this chapter, we consider network decomposition in a tandem of Δ-schedulers. This is achieved by enforcing constraints on the choice of traffic sources, and assuming that the number of traffic flows and the link capacity at each node are large. We start this chapter by exploring related work. Then, we introduce our system model and notation. The analyses of this chapter are based on our closed-form probabilistic output and backlog bounds from Theorem 4.6. Using that theorem, and the concept of probabilistic busy period bounds from [67], we can prove that sometimes we can eliminate a Δ-scheduler from the queuing analyses of other nodes in the network being affected. We show that if $\Delta \leq 0$, the constraint of large number of traffic flows might be unnecessary. The analytical results are examined in numerical examples.

5.1 Literature on Network Decomposition

Multiplexing traffic flows in a link alters their original statistical properties and this makes an end-to-end queuing analysis complicated. However, there are cases found in the literature in which the original traffic statistics are preserved inside the network. This facilitates the end-to-end delay, and backlog analyses since each node can be analyzed in isolation.

Product-form networks (See Sec. 4.1) are examples of these cases. Another example is Better Than Poisson (BTP) traffic which refers to the traffic sources creating a smaller workload than Poisson arrivals in a FIFO scheduler. It is shown via analysis and simulation in [7] that this property of BTP traffic is preserved as the traffic passes through FIFO schedulers. A third example, which is the focus of this section, is that traffic distortion in a node can be ignored under *large buffer size* or *many sources* asymptotic regimes. Large deviation methods are used to approximate the queue tail probabilities in these asymptotic regimes in the literature as follows.

Suppose that process A is an aggregate of N independent, identically distributed, traffic flows. These N flows are fed to a link with per-flow capacity c and per-flow buffer size b. Using a Chernoff bound, the total backlog in the steady state $B(\infty)$ satisfies the following (See [54])

$$\frac{1}{Nb}\log(P\{B > Nb\}) \leq -H(c) ,$$

where H is a positive-valued function of c and the traffic source. In large buffer asymptotic regime, where $b \to \infty$, the above inequality will turn to an equality. This asymptotic regime has been studied in many papers such as [56].

In a many sources asymptotic regime, the number of independent input flows to a link N goes to infinity, while per-flow buffer size b and per-flow capacity c remain fixed. For such a scenario, using a Chernoff bound, De Veciana *et al.* [32] show that

$$\lim_{N\to\infty}\frac{1}{N}\log(P\{B(\infty) > Nb\}) \leq -I(c,b) , \tag{5.1}$$

where

$$I(c,b) = \inf_{t>0}\sup_{\alpha\geq 0}\Big(\alpha(Nct+b) - \log(E[e^{\alpha A(t)}])\Big) . \tag{5.2}$$

Likhanov and Mazumdar [76] obtain a similar asymptotic formulation using the Gaussian approximation in the Bahadur-Rao Theorem [3]. Courcoubetis and Weber [23] show that $\lim_{b\to\infty}\frac{I(c,b)}{b} = H(c)$ suggesting that both asymptotic regimes are equivalent if b and N are large. $I(c,b)$ is a function of traffic sources, and has been estimated for Gaussian [23] and

Markovian sources [23], [49] and was improved and extended to ON/OFF sources with heavy-tailed distributed ON periods in [76], and regulated traffic in [102].

Both asymptotic regimes provide cases which allow decomposing a network into single nodes, and analyzing each node independently. This is achieved by the multiplexing gain from flow independence assumption. In any paper we review in the following, all flows at a node are assumed to be independent unless otherwise stated.

Assuming a large buffer asymptotic, De Veciana *et al.* [32] provide a constraint which guarantees that the effective bandwidth of the input and output traffic to a FIFO link match. The is achieved by introducing the concept of *decoupling bandwidth*. Decoupling bandwidth $Db_A(\alpha)$ is a function of traffic process A, and for any fixed $\alpha > 0$ satisfies $Eb_A(\alpha) < Db_A(\alpha) \leq 2Eb_A(\alpha) - Eb_A(0)$, where Eb_A is the effective bandwidth of A in the sense of Eq. (1.1). If $\mathcal{N}$ is the set of all flows at a link with capacity C, then the effective bandwidths of the input (Eb_i) and output (Eb_i^*) of flow $i \in \mathcal{N}$ are identical for some $\alpha \geq 0$ ($Eb_i(\alpha) = Eb_i^*(\alpha)$) if the following holds:

$$Db_i(\alpha) + \sum_{j \in \mathcal{N} \setminus \{i\}} Eb_j(0) < C \, .$$

To use this constraint, the decoupling bandwidth, and the effective bandwidth of the traffic sources in the link must be known. This is formulated for Markov Modulated On-Off sources, and Gaussian processes in [32].

Ying *et al.*, [101] obtain two probabilistic backlog bound approximations for a aggregate of independent peak-rate constraint leaky buckets flows in asymptotic regimes. The first approximation is obtained in a many sources asymptotic regime by identifying per-flow worst-case scenario, and multiplexing them using the Bahadur-Rao Theorem. The total backlog in a FIFO scheduler in a many sources asymptotic regime converges to zero, suggesting that this scheduler can be removed from the analyses of other nodes. The second approximation considers a *fixed load many sources* asymptotic regime, where the total capacity and the utilization are fixed, but the number of flows tends to infinity. Given this assumption, it is proved in [101] that regulated traffic is a BTP traffic. Thus, the workload of a multiplex of N independent leaky bucket with parameter (P, ρ, σ) can be approximated by the workload of a marked Poisson process with rate $\lambda = \frac{N\rho}{\sigma}$ arriving to an $M/G/1$ queue. In a non-asymptotic regime, the workload of the described $M/G/1$ serves as an upper bound for the aggregate of regulated flows. Using this approximation, a backlog bound is obtained by taking the inverse Laplace transform of the explicit formulation of an $M/G/1$ queue from [57].

Wischik [96] uses the backlog bound formulation in Eq. (5.1) to compare the statistical properties of the input and output traffic under a many sources asymptotic. He defines the *limiting moment generating function* of a process A as follows

$$\Lambda_t^A(\alpha) \triangleq \lim_{L\to\infty} \log(E(e^{\alpha A^L(t)})) ,$$

for any fixed $\alpha > 0$, where $A^L(t)$ represents the average of L independent sample paths in $[0, t)$. Then, it is shown that both input and output traffic have identical limiting moment generating functions in a many sources asymptotic regime. The simulation results suggest that this might still be the case even if the number of flows is handful.

Ying *et al.* [102] study the burstiness increase of leaky bucket arrivals in a cascade of FIFO schedulers. Suppose that a leaky bucket through flow with parameters (σ_0, ρ_0) passes through a tandem of H FIFO schedulers with capacities C_h for any $h = 1, \ldots, H$. At any node h there is a leaky bucket cross flow with parameters (σ_h, ρ_h). The *stochastic burstiness* of the through flow in time interval $[s, t)$, $\omega_0(s, t)$ is defined as the amount of traffic arrivals that exceed than a CBR traffic with rate ρ_0 in that interval. That is

$$\omega_0(s, t) \triangleq [A_0(s, t) - \rho_0(t - s)]_+ ,$$

where A_0 is the through flow arrival process. By applying the Markov inequality, it is proved that output stochastic burstiness of the through flow ω_0^* satisfies the following for any $x \geq 0$ and $s \leq t$:

$$P\{|\omega_0^*(s, t) - \sigma_0| > x\} \leq \frac{2\rho_0}{x} \sum_{h=1}^{H} \frac{E[B_h]}{C_h} ,$$

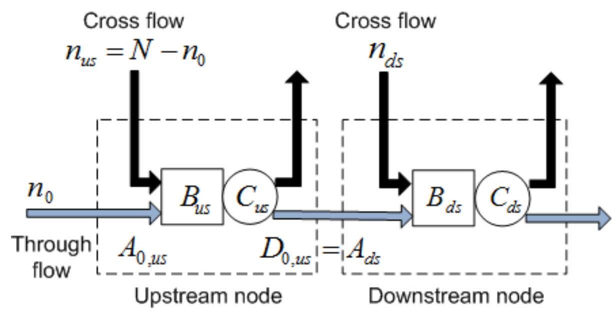


Figure 5.1: Network decomposition: disregarding upstream node for the queue analysis of the downstream node.

where $E[B_h]$ is the average total backlog at node h in the steady state, which is bounded if $\rho_0 + \rho_h < C_h$. Thus, the above equation implies that $\omega_0^*(s,t)$ converges to σ_0 in probability as $\frac{\rho_0}{C_h} \to 0$ for $h = 1, \ldots, H$. This last constraint is equivalent to a many sources asymptotic with adding a constraint on the traffic mix that through flows are not scaled by the total number of flows.

Ciucu and Liebeherr [21] assume a set of independent flows, each with a bounded peak-rate that satisfies

$$\sup_{s \geq 0} E\left[e^{\alpha A(s,t+s)}\right] \leq e^{\alpha \rho(\alpha) t} \ , \tag{5.3}$$

where $\rho(\alpha)$ must be independent of t, and is bounded between the long-term average rate and peak rate for any value of α. This property holds for many Markov-modulated processes including Markov-modulated ON-OFF sources. More precisely, for a Markov-modulated ON-OFF flow with parameters λ, μ, and P

$$\rho(\alpha) = \frac{1}{2\alpha}(P\alpha + \lambda + \mu + \sqrt{(P\alpha + \lambda - \mu)^2 + 4\mu\lambda}) \ .$$

If A is the aggregate of n independent flows all satisfying Eq. (5.3), then applying a Chernoff bound, yields the following for any $s \leq t$

$$P\{A(s,t) > n\rho(\alpha)(t-s) + \sigma\} \leq e^{-\alpha\sigma} \ , \tag{5.4}$$

which means that an aggregate of n independent flows of such a function is EBB in terms of Eq. (1.4) with $(1, n\rho(\alpha), \alpha)$. [1] For a fixed number of flows n, different EBB characterizations will be obtained by varying α. In fact, a larger α implies a larger decay rate and a larger EBB rate $\rho(\alpha)$. In other words, there are different probabilistic upper bounds for a stochastic arrival process. The looser the bounds are (larger ρ), the faster the decay rates of the violation probabilities are (larger α).

Consider the scenario depicted in Fig. 5.1, which is a two-node tandem network consisting of an upstream and a downstream node, respectively, represented by subindexes us and ds. Assume that there exist ρ_0 and ρ_c, respectively, for through and cross flows, satisfying Eq. (5.3). There are n_0 through flows and $N - n_0$ cross flows being multiplexed at a FIFO link with capacity Nc. Then, through flow output, and backlog bounds are formulated [21] by employing Doobe's inequality as follow

$$P\{D_{0,us}(s,t) > n_0\rho_0(\alpha)(t-s) + \sigma\} \leq K_d(\alpha)e^{-\alpha\sigma} \ ,$$

and

$$P\{B_{us}(t) > \sigma\} \leq K_b e^{-N\alpha\sigma}$$

for any $\sigma \geq 0$, where $K_d(\alpha)$ and K_b both converge to 1 as $N \to \infty$. This implies that the through flow backlog decays exponentially fast to zero in N, and the EBB parameters of the output converge to those of the input envelope exponentially fast in N. Suppose that B_{ds}^{I} denote the backlog at the downstream node in Fig. 5.1. In the same scenario, B_{ds}^{I} represents the backlog at the downstream node when the through flow enters the downstream node directly and without passing through the upstream node. By comparing [21] that B_{ds}^{I} via *a numerical example*, the authors conjecture that network decomposition might be valid for moderate values of N.

Eun and Shroff [38] compare B_{ds}^{I} and B_{ds}^{II} analytically. The upstream node can be ignored

[1] For the sake of simplicity of notation, we drop the dependency on ρ to α throughout this chapter, i.e., we will use ρ instead of $\rho(\alpha)$.

in a queue analysis of the downstream node if $B^{\mathrm{I}}_{ds} = B^{\mathrm{II}}_{ds}$. Obviously, this condition for network decomposition is more stringent than comparing the statistical properties of input and output in the upstream network. In particular, the point-wise convergence of statistical properties of the input and output induces the convergence of B^{I}_{ds} and B^{II}_{ds} only if the traffic flows are regulated.

Eqs. (5.1) and (5.2) are used in [38] to show that the total backlog in the upstream node in Fig. 5.1 in a many sources asymptotic decays to zero exponentially fast in the number of flows. Using this result, and the fact that busy periods are deterministically bounded for regulated traffic arrivals, it is shown that B^{I}_{ds} converges to B^{II}_{ds} almost surely in a many sources asymptotic regime, i.e.,

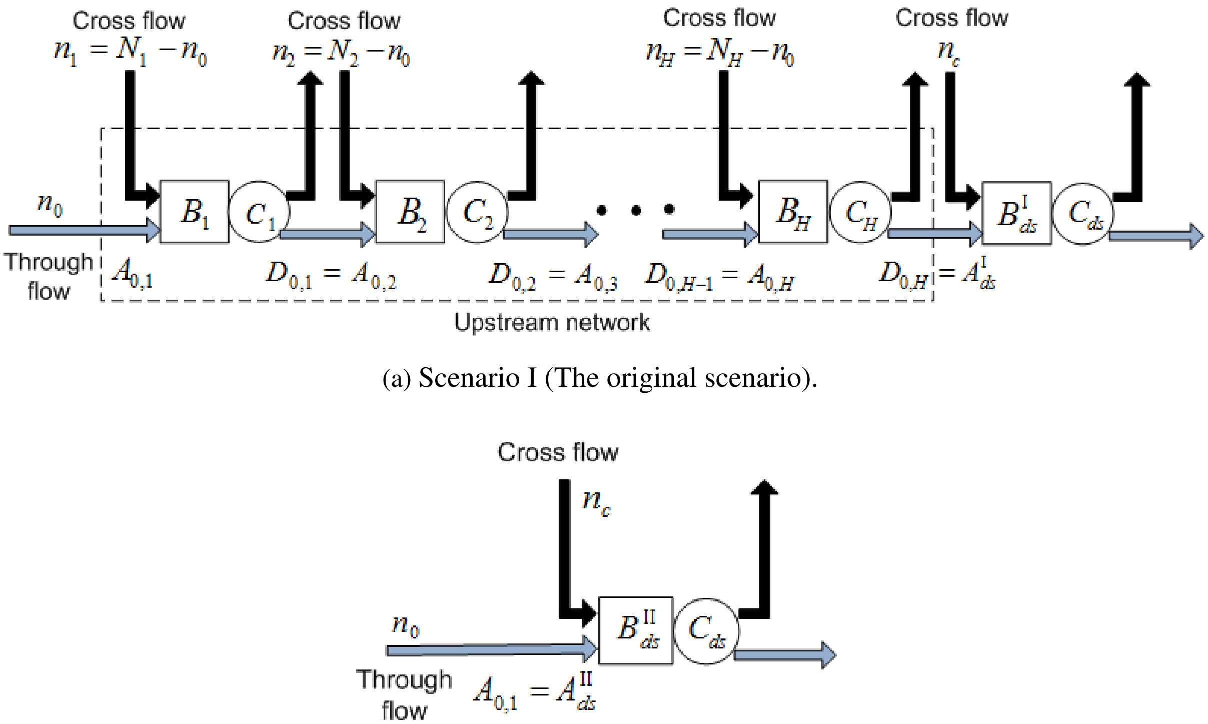
$$\lim_{N\to\infty} P\{|B^{\mathrm{I}}_{ds} - B^{\mathrm{II}}_{ds}| > 0\} = 0 \ .$$

If traffic regulation is reduced to only having a peak rate constraint, then it is shown that B^{I}_{ds} converges to B^{II}_{ds} in probability, i.e., for any $x \geq 0$

$$\lim_{N\to\infty} \Big| P\{B^{\mathrm{I}}_{ds} > x\} - P\{B^{\mathrm{II}}_{ds} > x\} \Big| = 0 \ .$$

Two improvements in network decomposition have been accomplished in [38]. First, the independence of through and cross flows is relaxed from the conditions of decomposition. Second, the convergence of the input and output traffic in the upstream network is mainly used in the literature to claim that the decomposition is valid. Whereas, decomposition is defined as the convergence of B^{I}_{ds} to B^{II}_{ds} in the downstream network in [38]. The fluid flow network decomposition arguments in [38] are based on the asymptotic convergence of the total backlog at the upstream node. However, the simulation results in that work suggest that the negligible upstream total backlog might be unnecessary for the validity of the network decomposition.

In all of the above, the scheduling used in the upstream network is assumed to be either FIFO or BMux. It is shown in [38], [102] that in a many sources asymptotic regime, scheduling at the upstream network does not change the backlog statistics in the downstream network. In this chapter, for the first time, we study the effect of scheduling in the upstream network in



(a) Scenario I (The original scenario).

(b) Scenario II (The simplified scenario).

Figure 5.2: Multi-node network decomposition.

network decomposition in a non-asymptotic regime as well as the asymptotic behavior. To allow a network decomposition, we consider the convergence of B^{I}_{ds} to B^{II}_{ds} as in [38].

5.2 System Model

Consider the two network scenarios depicted in Figs. 5.2. There are n_0 through flows traversing a cascade of H nodes in Fig. 5.2a to arrive to the downstream node. At each node h, in the upstream network, through flows are multiplexed with $n_h = N_h - n_0$ cross flows. Each node h in the upstream network is a Δ-scheduler with $\Delta^h_{0,c} = \Delta^h$, with link capacity $N_h c_h$. Define the scaling parameter to be $N = \min_h\{N_h\}$. If $N \to \infty$, and $N_h = \Theta(N)$ for any h, and $\lim_{N\to\infty} \frac{N_h c_h}{n_h \rho^{av}_h + n_0 \rho^{av}_0} < 1$, where ρ^{av}_i is the long-term average rate of flow i, then we are in a many sources asymptotic regime. In Fig. 5.2b, the upstream node is removed and the through flow enters the downstream node directly. The capacity of the downstream node is C_{ds} and there are n_{ds} cross flows in that link in addition to through flows.

Any traffic flow in the network satisfies Eq. (5.3), and consequently the aggregate of flows is EBB. In particular, the EBB parameters of the n_0 through flows are $(1, n_0\rho_0(\alpha_0), \alpha_0)$, for any $\alpha_0 > 0$. Similarly, the aggregate of n_c cross flows is EBB with parameters $(1, n_c\rho_c(\alpha_c), \alpha_c)$, for any $\alpha_c > 0$.

We compare the total backlog at the downstream node in Scenario I (B^{I}_{ds}), and Scenario II (B^{II}_{ds}) to see the effect of removing the upstream network on the downstream backlog statistics.

5.3 Network Decomposition in a Tandem of Δ-schedulers

Since the traffic sources we consider in this chapter are EBB arrivals, and the upstream network is a tandem of Δ-schedulers, we can employ Theorem 4.6 to obtain closed-form probabilistic end-to-end backlog and output bound in the upstream network. Before we use the theorem, we need to adapt the theorem to the notations of this chapter.

Define a vector of non-negative elements $\underline{\sigma} = (\sigma_0, \sigma_1, \ldots, \sigma_H)$. For any γ_0, γ_c, satisfying $0 \leq \gamma_0 + H\gamma_c \leq \min_h\{N_h c_h - n_h\rho_h - n_0\rho_0\}$, we have the following bounds for the through flow from Theorem 4.6.

1- Output burstiness bound: The through flow upstream network departure process $D_{0,H}$ can be bounded stochastically for any $s \leq t$ by

$$P\{D_{0,H}(s,t) > (n_0\rho_0 + \gamma_0)(t-s) + b_{net}(\underline{\sigma})\} \leq \varepsilon_{net}(\underline{\sigma})\ . \tag{5.5}$$

2- Backlog bound: The end-to-end through flow backlog B_{net} at any time t is statistically bounded by

$$P\{B_{net}(t) > b_{net}(\underline{\sigma})\} \leq \varepsilon_{net}(\underline{\sigma})\ , \tag{5.6}$$

where

$$b_{net}(\underline{\sigma}) = (n_0\rho_0 + \gamma_0 + (H-1)\gamma_c)\tau_{net} + \sigma_0 + (n_0\rho_0 + \gamma_0)\sum_{h=1}^{H}\theta_h^* \,, \tag{5.7}$$

$$\theta_h^* = \min\Big\{\frac{\sigma_h}{N_h c_h - n_h\rho_h - H\gamma_c}, \frac{[\sigma_h + (n_h\rho_h + \gamma_c)\Delta^h]_+}{N_h c_h - (H-1)\gamma_c}\Big\} \,, \tag{5.8}$$

$$\varepsilon_{net}(\underline{\sigma}) = M_0 e\Big(1 + \frac{n_0\rho_0}{\gamma_0}\Big)e^{-\alpha_0\sigma_0} + M_H e\Big(1 + \frac{n_H\rho_H}{\gamma_c}\Big)e^{-\alpha_H\sigma_H} + \sum_{h=1}^{H-1}\frac{M_h C^{min}}{\gamma_c}e\Big(1 + \frac{n_h\rho_h}{\gamma_c}\Big)e^{-\alpha_h\sigma_h} \,, \tag{5.9}$$

where $C^{min} = \min_h C_h$ and $\tau_{net} = \sum_{h=1}^{H-1}\frac{1}{\alpha_h C^{min}}$.

With these upper bounds, we can study the asymptotic impact of the upstream network on the queue statistics of the downstream node.

5.3.1 Asymptotic behavior of upstream backlog

In the sequel, we study the behavior of the through flow backlog at the upstream network B_{net} as a function of N. Define $\Delta_{max} = \max_h \Delta^h$. We consider B_{net}, separately, for $\Delta_{max} \geq 0$ and $\Delta_{max} < 0$.

• **Exponentially fast decay of backlog in N for $\Delta_{max} \geq 0$:**

Choose α_h small enough so that $c_h > \rho_h(\alpha_h)$. Define α_{net} to be

$$\alpha_{net} = \min_h\Big\{\Big(1 - \frac{1}{N}\Big)\frac{\alpha_h(N_h c_h - n_h\rho_h(\alpha_h) - H\gamma_c)}{HN(n_0 P_0 + \gamma_0)}\Big\} \,, \tag{5.10}$$

where P_0 is the peak rate of through flow. Note that $\alpha_{net} = O(1)$. Then, for any arbitrary $\sigma \geq 0$ set the parameters as follows

$$\begin{aligned}
&\alpha_0 = N^2\alpha_{net}; \qquad \sigma_0 = \frac{\sigma}{N}; \qquad \sigma_h = \frac{N\alpha_{net}\sigma}{\alpha_h};\\
&C^{min} = \min_h N_h c_h; \qquad \tau_{net} = \frac{\sum_{h=1}^{H-1} 1/\alpha_h}{C^{min}}\\
&M_{net} = M_0 e\Big(1 + \frac{n_0\rho_0}{\gamma_0}\Big) + M_H e\Big(1 + \frac{n_H\rho_H}{\gamma_c}\Big) + \frac{C^{min}}{\gamma_c}\sum_{h=1}^{H-1} M_h e\Big(1 + \frac{n_h\rho_h}{\gamma_c}\Big) \,.
\end{aligned} \tag{5.11}$$

These choices of parameters guarantee that the second term in Eq. (5.7) is $\frac{\sigma}{N}$, and the last terms is upper bounded by $\sigma - \frac{\sigma}{N}$. Replacing σ by $\sigma - [n_0\rho_0 + \gamma_0 + (H-1)\gamma_c]\tau_{net}$, and combining all above with Eq. (5.6), yields the following

$$P\{B_{net}(t) > \sigma\} \leq Ke^{-N\alpha_{net}\sigma} , \tag{5.12}$$

where

$$K = M_{net}e^{N\alpha_{net}[n_0\rho_0+\gamma_0+(H-1)\gamma_c]\tau_{net}} . \tag{5.13}$$

Note that from Eq. (5.11), $\tau_{net} = O(\frac{1}{N})$, and consequently $K = O(N)$. Combining all of the above, the upstream backlog decays to zero exponentially fast for any $\Delta_{max} \geq 0$.

• **Super exponentially fast decay of backlog in N for $\Delta_{max} < 0$:**

Choose α_h small enough so that $c_h > \rho_h(\alpha_h)$, for any $h = 1, \ldots, H$, and $\alpha_{net} = \min_h\{\alpha_h\}$. Then, for any arbitrary $\sigma \geq 0$, set the parameters as follows

$$\alpha_0 = N^2\alpha_{net}, \quad \sigma_0 = \frac{\sigma}{N^2}, \quad \sigma_h = \frac{\alpha_{net}\sigma}{\alpha_h} , \tag{5.14}$$

and the rest of the parameters as in Eq. (5.11). Then,

$$P\{B_{net}(t) > X(\sigma)\} \leq Ke^{-\alpha_{net}\sigma} ,$$

where K is as in Eq. (5.13). Since $\Delta^h < 0$, we also have

$$X(\sigma) = \frac{\sigma}{N^2} + (n_0\rho_0 + \gamma_0)\sum_{h=1}^{H} \frac{[\frac{\alpha_{net}}{\alpha_h}\sigma + (n_h\rho_h + \gamma_c)\Delta^h]_+}{N_hc_h - (H-1)\gamma_c} .$$

If N is large enough, the second term in $X(\sigma)$ evaluates to zero because $\Delta^h < 0$ and $n_h \to \infty$. Thus, there exists a constant N_0, such that $X(\sigma) = \frac{\sigma}{N^2}$ for any $N > N_0$. Replacing $\frac{\sigma}{N^2}$ with σ we have the following for any $N > N_0$

$$P\{B_{net}(t) > \sigma\} \leq Ke^{-N^2\alpha_{net}\sigma} , \tag{5.15}$$

and this proves that B_{net} decays super exponentially fast in N for $\Delta_{max} < 0$.

As shown in [38], the point-wise decay of B_{net} to zero is not sufficient to prove that B^{I}_{ds} converges to B^{II}_{ds} as N increases. Instead, a sample path backlog bound must decay to zero to guarantee that convergence. The following lemma shows how the violation probability scales when we construct a sample path from point-wise backlog bound. Combining this lemma with the concept of the probabilistic busy period bound, we are able to construct a sample path backlog bound.

Lemma 5.1. *Consider Scenario I depicted in Fig. 5.2a. The maximum peak-rate of the through flow aggregate is bounded by $n_0 P_0$. Suppose that the following holds for end-to-end upstream through flow backlog B_{net} for any $\sigma \geq 0$ and some non-increasing function ε_b:*

$$P\{B_{net}(t) > \sigma\} \leq \varepsilon_b(\sigma). \tag{5.16}$$

Then, for any arbitrary τ_s and $T \geq 0$,

$$P\Big\{\sup_{0\leq t\leq T} B_{net}(t) > \sigma\Big\} \leq \Big(\Big\lceil\frac{T}{\tau_s}\Big\rceil + 2\Big)\varepsilon_b(\sigma - n_0 P_0 \tau_s)\ . \tag{5.17}$$

Proof. We first need to discretize time. Set τ_s to be the time unit. Then, the maximum difference between $B_{net}(t)$ at any time instant t with the closest last discrete time cannot be larger than the total through flow arrivals in an interval of size τ_s. Thus, defining $T_Z = \big\{0, \tau_s, 2\tau_s, \ldots, \lceil\frac{T}{\tau_s}\rceil\tau_s\big\}$, we have

$$\begin{aligned}\sup_{0\leq t\leq T} B_{net}(t) &\leq \max_{t\in T_Z}\Big(B_{net}(t) + \max_{t\in T_Z} A_0(t, t+\tau_s)\Big)\\ &\leq \max_{t\in T_Z} B_{net}(t) + n_0 P_0 \tau_s\ ,\end{aligned} \tag{5.18}$$

where the peak rate arrival constraint of the through flow is used in the second line. With this result, we have

$$
\begin{aligned}
P\Big\{\sup_{0\le t\le T} B_{net}(t) > \sigma\Big\}
&\le P\Big\{\max_{t\in T_Z} B_{net}(t) > \sigma - n_0 P_0 \tau_s\Big\} \\
&\le \sum_{t\in T_Z} P\Big\{B_{net}(t) > \sigma - n_0 P_0\tau_s\} \\
&= \Big(\Big\lceil \frac{T}{\tau_s}\Big\rceil + 2\Big)\varepsilon_b(\sigma - n_0 P_0 \tau_s)\ ,
\end{aligned}
$$

where the third line uses Boole's inequality. The last line uses $|T_Z| \le \lceil \frac{T}{\tau_s} \rceil + 2$. The assumption in Eq. (5.16). □

The above bound can be minimized over τ_s. Note that $\lceil \frac{T}{\tau_s} \rceil$ is decreasing in τ_s, while $\varepsilon_b(\sigma - n_0 P_0 \tau_s)$ increases in τ_s.

5.3.2 Formulating a probabilistic busy period bound

A busy period is referred to as the time duration in which the backlog of a work conserving link is non-zero. If each busy period is bounded, sample path bounds can be obtained from point-wise bounds as discussed in Sec. 2.4.3. We compute probabilistic upper bounds on the busy periods at the downstream node for both scenarios in Fig. 5.2 in the sense of Eqs. (2.65) and (2.66).

Let us assume that through and cross flows at the downstream node are EBB with parameters $(1, n_0\rho_0^{\mathrm{II}}, \alpha_0^{\mathrm{II}})$ and $(1, n_c\rho_c^{\mathrm{II}}, \alpha_c^{\mathrm{II}})$, where we simplified notation by setting $\rho_0^{\mathrm{II}} = \rho_0(\alpha_0^{\mathrm{II}})$ and $\rho_c^{\mathrm{II}} = \rho_c(\alpha_c^{\mathrm{II}})$. Inserting the EBB sample path envelopes from Eq. (2.53) in Eq. (2.65), a probabilistic busy period bound can be obtained. If α_c^{II}, α_0^{II}, and γ^{II} are arbitrary constants which are chosen such that $C_{ds} > n_0\rho_0^{\mathrm{II}} + n_c\rho_c^{\mathrm{II}}$ and $\gamma^{\mathrm{II}} \le \frac{C_{ds} - n_c\rho_c^{\mathrm{II}} - n_0\rho_0^{\mathrm{II}}}{2}$, then, for any $\sigma_0, \sigma_c \ge 0$

$$T_{\mathrm{II}} = \frac{\sigma_c + \sigma_0}{C_{ds} - n_c\rho_c^{\mathrm{II}} - n_0\rho_0^{\mathrm{II}} - 2\gamma^{\mathrm{II}}} \tag{5.19}$$

$$\varepsilon^{T_{\mathrm{II}}}(\sigma_0, \sigma_c) = e\Big(1 + \frac{n_0\rho_0^{\mathrm{II}}}{\gamma^{\mathrm{II}}}\Big)e^{-\alpha_0^{\mathrm{II}}\sigma_0} + e\Big(1 + \frac{n_c\rho_c^{\mathrm{II}}}{\gamma^{\mathrm{II}}}\Big)e^{-\alpha_c^{\mathrm{II}}\sigma_c} \tag{5.20}$$

Choosing $\sigma = \sigma_0 + \sigma_c$ and $\alpha^{\mathrm{II}}_{ds} = (\frac{1}{\alpha^{\mathrm{II}}_c} + \frac{1}{\alpha^{\mathrm{II}}_0})^{-1}$ reduces the above equations to

$$T_{\mathrm{II}} = \frac{\sigma}{C_{ds} - n_c \rho^{\mathrm{II}}_c - n_0 \rho^{\mathrm{II}}_0 - 2\gamma^{\mathrm{II}}} \tag{5.21}$$

$$\varepsilon^{T_{\mathrm{II}}}(\sigma) = e\Big(2 + \frac{n_0 \rho^{\mathrm{II}}_0}{\gamma^{\mathrm{II}}} + \frac{n_c \rho^{\mathrm{II}}_c}{\gamma^{\mathrm{II}}}\Big) e^{-\alpha^{\mathrm{II}}_{ds}\sigma} \ , \tag{5.22}$$

which is a probabilistic busy period bound in the sense of Eqs. (2.65) and (2.66).

To compute an upper bound on the corresponding busy period bound in Scenario I, we need to have an upper bound on the through flow departures from the upstream network $D_{0,H}$. From Eqs. (5.5)–(5.9) it is clear that for a fixed violation probability, the output envelope increases as Δ increases. Thus, an output envelope in a BMux link serves as an output envelope for any scheduler. Choose α^{I}_h small enough so that $c_h > \rho_h(\alpha^{\mathrm{I}}_h)$, then for any arbitrary $\sigma \geq 0$ choose other parameters as follows

$$\alpha^{\mathrm{I}}_0 = \min_h \Big\{ \frac{\alpha_h (N_h c_h - n_h \rho^{\mathrm{I}}_h - H\gamma^{\mathrm{I}})}{NH(n_0 P_0 + \gamma^{\mathrm{I}})} \Big\} \tag{5.23}$$

$$\sigma_0 = \sigma\Big(1 - \frac{1}{N}\Big); \quad \forall 0 \leq h \leq H : \ \sigma_h = \frac{\alpha^{\mathrm{I}}_0}{\alpha_h}\sigma, \tag{5.24}$$

where γ^{I} is a free parameter satisfying $\gamma^{\mathrm{I}} < \min_h \frac{N_h c_h - n_h \rho^{\mathrm{I}}_h - n_0 \rho^{\mathrm{I}}_0}{H+1}$ (from stability condition at the upstream network). This guarantees that the through flow departure process is EBB with parameters $(K^{\mathrm{I}}_0, n_0\rho^{\mathrm{I}}_0 + \gamma^{\mathrm{I}}, \alpha^{\mathrm{I}}_0)$, where K^{I}_0 is as formulated in Eq. (5.13). Thus, a statistical sample path envelope for the through flow departure from the upstream network is (see Eq. (2.53))

$$\mathcal{G}^{\mathrm{I}}_0(t) = (n_0 \rho^{\mathrm{I}}_0 + 2\gamma^{\mathrm{I}})t + \sigma; \quad \varepsilon(\sigma) = eK^{\mathrm{I}}_0\Big(1 + \frac{n_0\rho^{\mathrm{I}}_0 + \gamma^{\mathrm{I}}}{\gamma^{\mathrm{I}}}\Big) e^{-\alpha^{\mathrm{I}}_0 \sigma} \ . \tag{5.25}$$

Multiplexing this EBB departure process with EBB cross flow at the downstream node with parameters $(1, n_c\rho^{\mathrm{I}}_c, \alpha^{\mathrm{I}}_c)$, a busy period bound for Scenario I at the upstream network can be obtained similar to that of T_{II}. Define $\alpha^{\mathrm{I}}_{ds} = (\frac{1}{\alpha^{\mathrm{I}}_c} + \frac{1}{\alpha^{\mathrm{I}}_0})^{-1}$ then, for any $\gamma^{\mathrm{I}} \leq \frac{C_{ds} - n_0\rho^{\mathrm{I}}_0 - n_c\rho^{\mathrm{I}}_c}{3}$ (Recap that γ^{I} must also satisfy $\gamma^{\mathrm{I}} < \min_h \frac{N_h c_h - n_h\rho^{\mathrm{I}}_h - n_0\rho^{\mathrm{I}}_0}{H+1}$ in the upstream network) the following is a busy period bound in the sense of Eqs. (2.65) and (2.66).

$$T_{\mathrm{I}} = \frac{\sigma}{C_{ds} - n_0\rho_0^{\mathrm{I}} - n_c\rho_c^{\mathrm{I}} - 3\gamma^{\mathrm{I}}} \tag{5.26}$$

$$\varepsilon^{T_{\mathrm{I}}}(\sigma) = e\Big(1 + K_0^{\mathrm{I}} + K_0^{\mathrm{I}}\frac{n_0\rho_0^{\mathrm{I}} + \gamma^{\mathrm{I}}}{\gamma^{\mathrm{I}}} + \frac{n_c\rho_c^{\mathrm{I}}}{\gamma^{\mathrm{I}}}\Big)e^{-\alpha_{ds}^{\mathrm{I}}\sigma} \ . \tag{5.27}$$

The above busy period bounds are used in the next section to compare B_{ds}^{I} and B_{ds}^{II}, analytically.

5.3.3 Almost sure network decomposition

Although we have shown that the through flow backlog at the upstream network decays exponentially fast for $\Delta_{max} \geq 0$, and super exponentially fast for $\Delta_{max} < 0$, the convergence of the queue size at the downstream node in the simplified and original scenarios has not been investigated yet. This is considered in the following theorem.

Theorem 5.1 (Almost sure convergence of B_{ds}^{I} to B_{ds}^{II})**.** *Consider the Scenarios in Fig. 5.2. Any through and cross flow source satisfies Eq. (5.3). There exists a constant $\alpha > 0$ and non-negative functions L and Q, such that for any $\sigma \geq 0$*

$$P\{|B_{ds}^{I}(t) - B_{ds}^{II}(t)| > \sigma\} \leq \begin{cases} L(\sigma)e^{-N\alpha\sigma} & \textit{if } \Delta_{max} \geq 0 \\ Q(\sigma)e^{-N^2\alpha\sigma} & \textit{if } \Delta_{max} < 0 \ , \end{cases} \tag{5.28}$$

where $L(\sigma) = O(N^2)$ and $Q(\sigma) = O(N^3)$.

Proof. Denote the traffic processes to the downstream node in Scenarios I and II, respectively, by A_{ds}^{I} and A_{ds}^{II}. Moreover, A_{ds}^{c} is the cross flow arrival at that node. Suppose that T_{I} and T_{II} are probabilistic busy period bounds as computed in Eqs. (5.26) and (5.21). In addition, define

$T^{max} = \max\{T_{\mathrm{I}}, T_{\mathrm{II}}\}$. Then, for any $\sigma, \sigma^{\mathrm{I}}$ and $\sigma^{\mathrm{II}} \geq 0$:

$$
\begin{aligned}
&P\{|B^{\mathrm{I}}_{ds}(t) - B^{\mathrm{II}}_{ds}(t)| > \sigma\} \\
&\quad \leq P\Big\{\Big|\sup_{0\leq u\leq T^{max}}[A^{\mathrm{I}}_{ds}(t-u,t) + A^{c}_{ds}(t-u,t) - C_{ds}u] \\
&\qquad\qquad - \sup_{0\leq u\leq T^{max}}[A^{\mathrm{II}}_{ds}(t-u,t) + A^{c}_{ds}(t-u,t) - C_{ds}u]\Big| > \sigma\Big\} \\
&\qquad\qquad + P\{t - \hat{x}^{\mathrm{I}}_t > T^{max}\} + P\{t - \hat{x}^{\mathrm{II}}_t > T^{max}\} && (5.29) \\
&\quad \leq P\Big\{\sup_{0\leq u\leq T^{max}}[A^{\mathrm{I}}_{ds}(t-u,t) - A^{\mathrm{II}}_{ds}(t-u,t)] > \sigma\Big\} \\
&\qquad\qquad + P\{t - \hat{x}^{\mathrm{I}}_t > T^{max}\} + P\{t - \hat{x}^{\mathrm{II}}_t > T^{max}\} && (5.30) \\
&\quad = P\Big\{\sup_{0\leq u\leq T^{max}}[B_{net}(t-u) - B_{net}(t)] > \sigma\Big\} + P\{t - \hat{x}^{\mathrm{I}}_t > T_{\mathrm{I}}\} + P\{t - \hat{x}^{\mathrm{II}}_t > T_{\mathrm{II}}\} \\
& && (5.31) \\
&\quad \leq P\Big\{\sup_{0\leq u\leq T^{max}}\{B_{net}(t-u)\} > \sigma\Big\} + P\{t - \hat{x}^{\mathrm{I}}_t > T_{\mathrm{I}}\} + P\{\hat{x}^{\mathrm{II}}_t > T_{\mathrm{II}}\} && (5.32) \\
&\quad \leq \left\lceil\frac{T^{max}}{\tau_s}\right\rceil \varepsilon_b(\sigma - n_0 P_0 \tau_s) + \varepsilon^{T_{\mathrm{I}}}(\sigma^{\mathrm{I}}) + \varepsilon^{T_{\mathrm{II}}}(\sigma^{\mathrm{II}}), && (5.33)
\end{aligned}
$$

where $\hat{x}^{\mathrm{I}}_t$ and $\hat{x}^{\mathrm{II}}_t$ are, respectively, the start of the busy period bound containing t at the downstream node in Scenarios I and II. In Eq. (5.29) we use the fact that $P(X) \leq P(X|Y) + P(Y')$ for any events X and Y, and $P(Y') = 1 - P(Y)$. Eq. (5.31) follows since $A^{\mathrm{I}}_{ds} = D_{0,H}$, and $A^{\mathrm{II}}_{ds} = A_{0,1}$. The last line is an application of Lemma 5.1 and the busy period conditions. ε_b is formulated in Eq. (5.12) for $\Delta_{max} \geq 0$, and in Eq. (5.15) for $\Delta_{max} < 0$. Moreover, $\varepsilon^{T_{\mathrm{I}}}$ and $\varepsilon^{T_{\mathrm{II}}}$ are computed, respectively, in Eqs. (5.27) and (5.22).

Proper choices of free parameters in Eq. (5.33) prove the theorem as follows. Choose α^{I}_{ds}, $\alpha^{\mathrm{II}}_{ds} = \alpha$. If $\Delta_{max} \geq 0$, then from Eq. (5.12), $\varepsilon_b(\sigma) = O(e^{-N\alpha\sigma})$. Set $\tau_s = \frac{T^{max}}{N^2}$, σ^{I}, and $\sigma^{\mathrm{II}} = N\sigma$. Then, from Eqs. (5.21) and (5.26), $T_{max} = O(N)$, $\tau_s = O(\frac{1}{N})$, and $\varepsilon^{T_{\mathrm{II}}}(\sigma^{\mathrm{II}}), \varepsilon^{T_{\mathrm{I}}}(\sigma^{\mathrm{I}}) = O(e^{-N\alpha\sigma})$ from Eqs. (5.27) and (5.22). If $\Delta_{max} < 0$ then from Eq. (5.15), $\varepsilon_b(\sigma) = O(e^{-N^2\alpha\sigma})$. Choose $\tau_s = \frac{T^{max}}{N^3}$ and $\sigma^{\mathrm{I}}, \sigma^{\mathrm{II}} = N^2\sigma$. This implies that $T_{max} = O(N^2)$ and $\tau_s = O(\frac{1}{N})$, respectively, from Eq. (5.21) and Eq. (5.26), and $\varepsilon^{T_{\mathrm{I}}}(\sigma^{\mathrm{I}}), \varepsilon^{T_{\mathrm{II}}}(\sigma^{\mathrm{II}}) = O(e^{-N^2\alpha\sigma})$ from Eqs. (5.27) and (5.22). Inserting these choices in Eq. (5.33) completes the proof.

□

This theorem shows that B^{I}_{ds} converges to B^{II}_{ds} almost surely. The speed of convergence is exponentially fast for $\Delta_{max} \geq 0$, and super exponentially fast for $\Delta_{max} < 0$. In the specific case of FIFO ($\Delta_{max} = 0$), the above theorem strengthen the results of [38] from *in probability* convergence to *almost sure* convergence for traffic sources with bounded peak-rate satisfying Eq. (5.3).

The above results imply that along with the independence of flows, scheduling is another important factor in viability of network decomposition. Although network can be decomposed in a many sources asymptotic regime for any schedulers as also addressed in [38] and [102], it might or might not be valid in a non-asymptotic regime depending on the scheduling algorithm used in the upstream network.

5.4 Numerical Examples

In this section, we examine our analytical results on network decomposition. We assume that the upstream network in Fig. 5.2a is homogeneous and consists of H identical Δ-schedulers with link capacity C. The total number of through and cross flows at each node is N (i.e., $N_h = N$ for any h), where N also serves as the scaling parameter. We choose ON-OFF Markov-Modulated Poisson Processes (MMPP) sources with $\lambda = 1ms^{-1}$, $\mu = 0.11ms^{-1}$ (same as the parameters for previous chapter numerical examples), for both through and cross flows. The capacity of any node in the upstream network is $C = Nc$, where $c = 0.1669$ Mbps. The above choices of traffic parameters and link capacities keeps the utilization at each node fixed to $\frac{N\rho_{av}}{Nc} = 90\%$ independent of N. Unless otherwise stated, we choose $\alpha_0 = N^2\alpha_{net}$ and $\alpha_h = \alpha_{net}$, where the bounds are optimized numerically over α_{net} and $\alpha_{net} \sim [10^{-10}, 10]$. In addition, we choose $\sigma_0 = \frac{\alpha_{net}}{\alpha_0}\sigma$ and $\sigma_h = \frac{\alpha_{net}}{\alpha_h}\sigma$ for any $h = 1, \ldots, H$. We use Eqs. (5.5) - (5.9) for the numerical results and optimize the bounds over free parameters numerically.

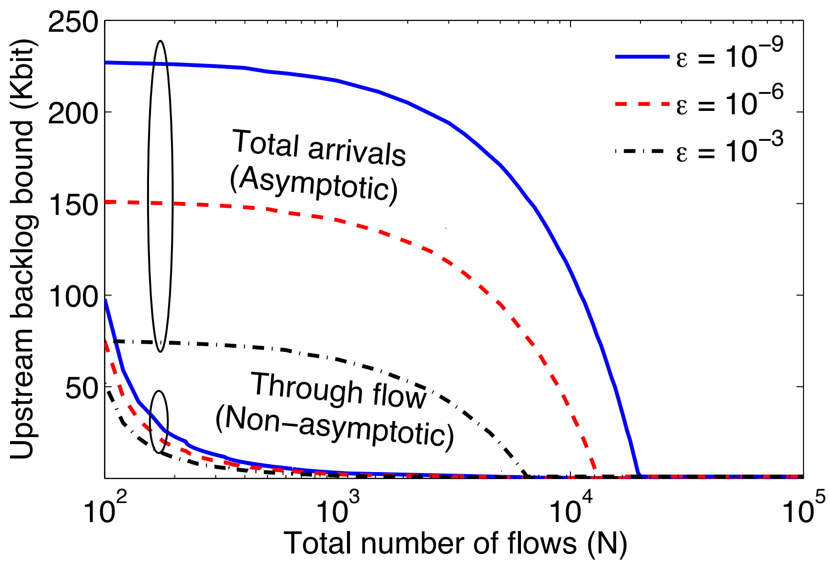


Figure 5.3: Comparing network decomposition criteria from the analyses of this chapter, and that of [38] with $n_0 = 1$, $C = 0.1169N$ Mbps, $U = 90\%$, $P = 1.5$ Mbps, $T^* = 10$ ms with $\varepsilon = 10^{-3}, 10^{-6}, 10^{-9}$.

5.4.1 Asymptotic total backlog vs. non-asymptotic per-flow backlog

Showing the convergence of B^{I}_{ds} to B^{II}_{ds} in [38], is based on proving that the *total backlog* at the upstream node at the steady state in a many sources asymptotic is negligible. In this chapter, we replaced this requirement by a less stringent condition, and requiring negligible *through flow backlog* at the upstream network. The gain obtained by this replacement helps to investigate the viability of the decomposition in a non-asymptotic regime. To see the gain obtained by this replacement, we compare the asymptotic total backlog bound from Eq. (5.1) with the through flow backlog bound from Eq. (5.5) in a FIFO scheduler in Fig. 5.3.

To use Eq. (5.2), a moment generating function for MMPP traffic is needed. This has been computed and presented in [55] as follows

$$E[e^{\alpha A(t)}] = \left(\frac{\lambda}{\lambda+\mu}\ ,\ \frac{\mu}{\lambda+\mu}\right) exp\left[\begin{pmatrix} -\mu + P\alpha & \mu \\ \lambda & -\lambda \end{pmatrix} t\right] (1\ ,\ 1)\ .$$

We fix $n_0 = 1$ and vary N in $[10^2, 10^5]$ and compute the backlog bounds for each N. We repeat

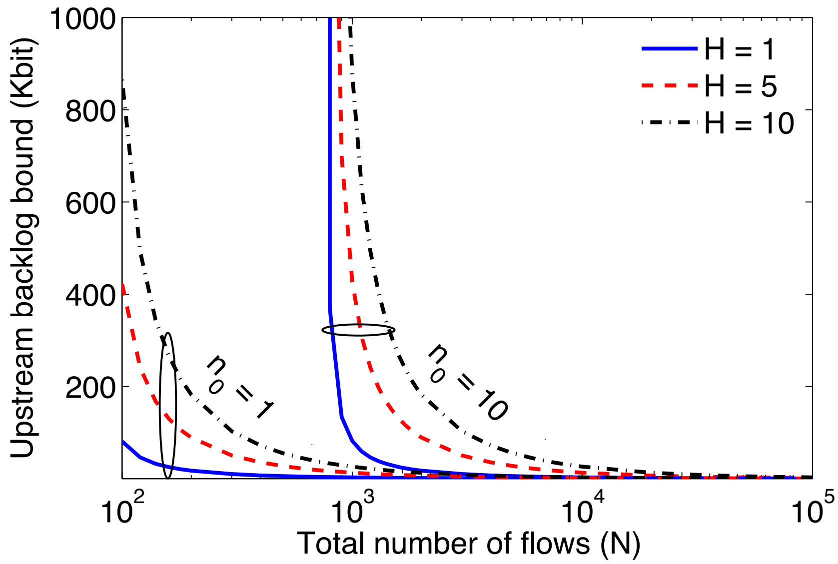


Figure 5.4: Probabilistic upstream backlog decay in N for different upstream path lengths, and traffic mixes with $\varepsilon = 10^{-6}$, $n_0 = 1, 10$, $C = 0.1169N$ Mbps, $U = 90\%$, $P = 1.5$ Mbps, $T^* = 10$ ms.

this example for three different violation probabilities: $\varepsilon = 10^{-3}$, 10^{-6}, and 10^{-9}.

The convergence speed of the through flow backlog to zero is faster than that of the asymptotic total backlog. The difference grows for smaller violation probabilities. This figure also corroborates the conjecture in [38] that negligibility of the *total upstream backlog* is not a *necessary* condition for the convergence of B^{I}_{ds} to B^{II}_{ds}.

5.4.2 The size of upstream network

In this section, we want to see if a through flow passes through a network consisting of H identical Δ-schedulers all satisfying many sources asymptotic criterion, then how distorted the final departure traffic would be. This is examined by computing the end-to-end through flow backlog bound in that network. The number of through flows is set to be $n_0 = 1, 10$ and for each, we compute the through flow upstream backlog bound (from this chapter) as a function of the scaling parameter N for three different path lengths ($H = 1, 2, 10$). All schedulers are assumed to be FIFO ($\Delta = 0$). Fig. 5.4 shows that the total backlog for any path length

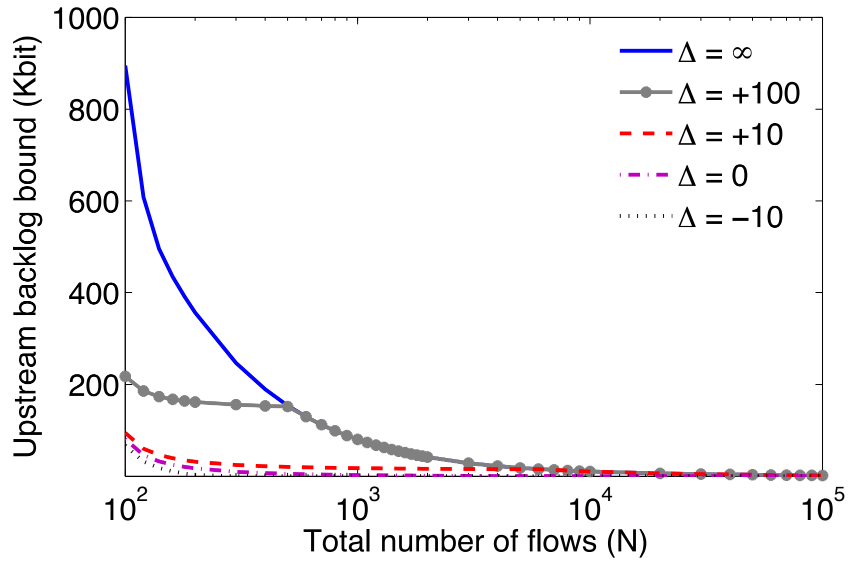


Figure 5.5: Probabilistic upstream backlog decay in N for $\Delta = -10, 0, 10, 100, \infty$ with $\varepsilon = 10^{-6}$, $n_0 = 1$, $C = 0.1169N$ Mbps, $U = 90\%$, $P = 1.5$ Mbps, $T^* = 10$ ms.

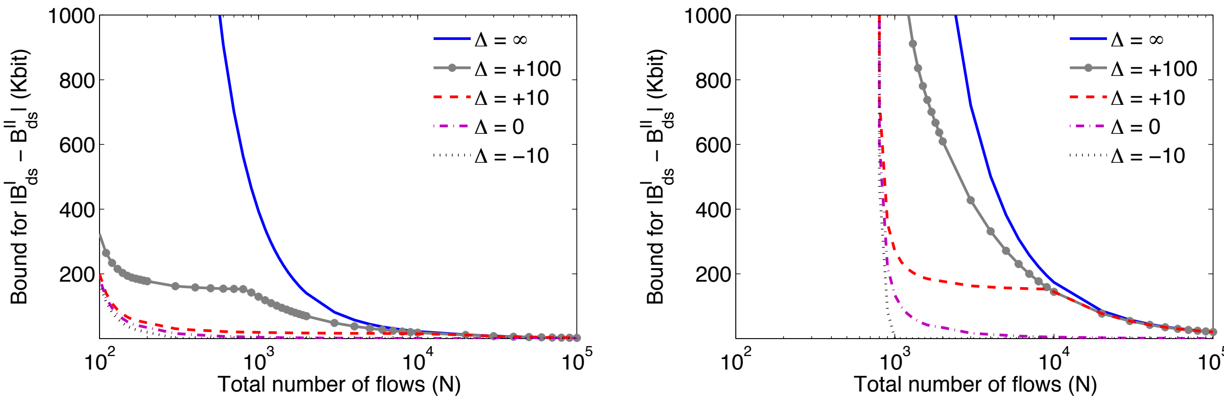
and traffic mixes will eventually decay to zero as the scaling parameter grows. However, the speed of convergence is highly dependent first on the traffic mix, and then on the upstream path length.

5.4.3 The effect of scheduling

In this example, we want to study the effect of the type of schedulers on the upstream through flow backlog. The number of through flows is $n_0 = 1$ and 10. We plot the through flow backlog as a function of the scaling parameter for different values of Δ, i.e., $-10, 0, 10, 100$, and ∞.

Fig. 5.5 shows that the speed of convergence of through flow backlog to zero is significantly affected by the scheduling in the upstream network. The through flow backlog is almost zero if N is few hundreds for some negative $\Delta \leq 10$ while it is non-negligible for $\Delta = 100$ even for $N = 1000$.

There is a break point in the corresponding curve for $\Delta = 100$, and the backlog bound after this break point matches that of BMux. The break point occurs when the minimum operation



(a) $n_0 = 1$ (b) $n_0 = 10$

Figure 5.6: Probabilistic upper bound on the backlog status at downstream node in the original, and simplified scenario with $\varepsilon = 10^{-6}$, $n_0 = n_c$ (at downstream node), $C = 0.1169N$ Mbps, $C_{ds} = 0.1669(n_0 + n_c)$, $U = 90\%$, $P = 1.5$ Mbps, and $T^* = 10$ ms.

in θ^* formulation in Eq. (5.8) switches from the second term to the first term. Note that θ^* always evaluates to the first term for BMux. The interpretation is that while N is increasing, the number of cross flows increases while the number of through flows is fixed. Thus, if cross flows have higher precedence than through flows ($\Delta > 0$), the scheduler will eventually converge to BMux since the number of cross flows becomes large compared to that of through flows.

This figure shows that the scheduling algorithms in the upstream network can impact the network decomposition in a non-asymptotic regime. The effect of scheduling can be as important as the effect of multiplexing gain in network decomposition and changes the convergence speed of B^{I}_{ds} to B^{II}_{ds} as shown in the analytical results.

5.4.4 A case for decomposition

In this example, we compare the total backlog of the downstream node in Scenario I, B^{I}_{ds}, with the total backlog in Scenario II, B^{II}_{ds}, by computing a probabilistic upper bound for $|B^{\mathrm{II}}_{ds} - B^{\mathrm{I}}_{ds}|$ from Eq. (5.33). The upstream network is a single Δ-scheduler with different choices of $\Delta =$

$-10, 0, 10, 100, \infty$. The details of the computations for the plots in Fig. 5.6a are presented as a pseudocode in Algorithm 3.

Fig. 5.6a considers one through flow ($n_0 = 1$), and one MMPP cross flow ($n_c = 1$) at the downstream node. The link capacity of the downstream node is $C_{ds} = (n_0 + n_c)0.1669$ Mbps which keeps the utilization at the downstream node to 90%. The figure shows that the queue size of Scenario I converges to that of Scenario II very fast especially for $\Delta \leq 10$. In fact, both scenarios seem to have the same probabilistic backlog bound even when N is few hundreds for $\Delta \leq 10$. For $\Delta \geq 100$, the difference is ignorable if N is as large as few 10^4.

Fig. 5.6b repeats the same experiment as Fig. 5.6a, except that $n_0 = n_c = 10$. As assumed along this chapter, the number of through flows must be small compared to N to have a valid case of decomposition. This figure shows that violating this assumption can drastically impact the decomposition viability for large values of Δ, e.g., $\Delta = 100, \infty$. But the convergence of the backlog bounds in two scenarios happens for moderate values of $N < 10^4$ if $\Delta \leq 0$.

Algorithm 3 The algorithm used in Fig. 5.6

$\varepsilon \leftarrow 10^{-6}$

for $N = 10^2$ to 10^5 **do**

  for $\alpha = 10^{-10}$ to 10 **do**

    for all γ^{II} such that $0 \leq \gamma^{\text{II}} \leq \frac{C_{ds}-n_0\rho_0(\alpha)-n_c\rho_c(\alpha)}{2}$ **do**

      Compute σ^{II} by setting $\varepsilon^{T_{\text{II}}}(\sigma^{\text{II}})$ from Eq. (5.22) equal to $\frac{\varepsilon}{N^2+2}$

      Compute $T_{\text{II}}(\alpha, \gamma^{\text{II}})$ from Eq. (5.21)

    end for

  end for[This loop computes T_{II}]

  $T_{\text{II}} \leftarrow \min_{\alpha,\gamma^{\text{II}}} T_{\text{II}}(\alpha, \gamma^{\text{II}})$

  for $\alpha = 10^{-10}$ to 10 **do**

    $\alpha_0, \alpha_c \leftarrow \alpha$, $\alpha_1 \leftarrow \frac{\alpha}{N}$

    for all γ^{I} such that $0 \leq \gamma^{\text{I}} \leq \frac{C_{ds}-n_0\rho_0(\alpha_0)-n_c\rho_c(\alpha_c)}{3}$ **do**

      $\gamma_{us} \leftarrow C - n_1\rho_1(\alpha_1) - n_0\rho_0(\alpha_0) - \gamma^{\text{I}}$

      $K_0^{\text{I}} \leftarrow e\left(1 + \frac{n_0\rho_0(\alpha_0)}{\gamma^{\text{I}}} + \frac{n_1\rho_1(\alpha_1)}{\gamma_{us}}\right)$

      Compute σ^{I} by setting $\varepsilon^{T_{\text{I}}}(\sigma^{\text{I}})$ from Eq. (5.27) equal to $\frac{\varepsilon}{N^2+2}$

    end for

  end for[This loop computes T_{I}]

  $T_{\text{I}} \leftarrow \min_{\alpha,\gamma^{\text{I}}} T_{\text{I}}(\alpha, \gamma^{\text{I}})$

  $T_{max} = \max\{T_{\text{I}}, T_{\text{II}}\}$

  $\tau_s \leftarrow \frac{T_{max}}{N^2}$

  for $\alpha = 10^{-10}$ to 10 **do**

    $\alpha_0 \leftarrow N^2\alpha$, $\alpha_1 \leftarrow \alpha$

    for all γ such that $0 \leq \gamma \leq \frac{C-n_1\rho_1(\alpha_1)-n_0\rho_0(\alpha_0)}{H+1}$ **do**

      Compute σ_{net} by setting $\left\lceil \frac{T^{max}}{\tau_s} \right\rceil \varepsilon_b(\sigma_{net} - n_0 P_0 \tau_s)$ from Eq. (5.33) equal to $\frac{N^2\varepsilon}{N^2+2}$

      $X(\alpha, \gamma) \leftarrow b_{net}(\sigma_{net})$ from Eq. (5.7).

    end for

  end for

  return $Y(N) = \min_{\alpha,\gamma} X(\alpha, \gamma)$.

end for

Chapter 6

Conclusions and Future work

6.1 Conclusions

In this thesis, we have provided end-to-end delay and backlog analysis for a tandem of a class of schedulers which we call Δ-schedulers (and includes FIFO, SP, and EDF as special cases). We have advanced the Stochastic Network Calculus in performance analyses of a network of schedulers. The results of this thesis have gained new insights into the impact of scheduling algorithms on the end-to-end delay and backlog. Our main contributions are listed below.

1- In a single node scenario, Network Calculus can provide necessary and sufficient delay bound condition for Δ-schedulers.

We have formulated a tight service curve for Δ-schedulers (Theorem 3.1). The tightness of the service curve means that it can provide the necessary and sufficient single node delay bound constraint.

2- In an end-to-end scenario, Network Calculus can provide a scheduling analysis which is tight in some regimes.

Applying per-node Δ-scheduler service curves to the existing convolution theorem from [20], a statistical network service curve for a tandem of Δ-schedulers can be obtained. We have shown that the resulting end-to-end delay bounds in a tandem of Δ-schedulers with the existing convolution theorem are loose except for the case of BMux ($\Delta = -\infty$).

Our main contribution in multi-node scenarios is formulating a new statistical convolution theorem (Theorem 4.4). We have shown that in the special case of the deterministic setting of our formulations, the end-to-end upper delay bounds are close to the worst-case achievable delays (sometime matches) implying that our end-to-end bounds are tight good and tight in some regimes. Numerical comparisons of the new and existing convolution theorem shows the considerable gain achieved by Theorem 4.4. The numerical results also shows that the effect of scheduling does not diminish as the path length increases.

3- The existing conjecture on the departure characterization of CBR traffic in FIFO schedulers in Eq. (4.3) can be proved and extended it to multi-node scenarios and to more general traffic types.

We have shown that the existing conjecture on the CBR departure traffic rate in a FIFO scheduler in Eq. (4.3) is valid not only for CBR traffic, but also for general traffic sources in terms of long-term average rate. In another extension, the single node results are used for output characterization of general traffic on a long path. We have shown that the long-term average rate of a through flow in a tandem of FIFO schedulers converges to that in a tandem of SP with the lowest priority given to the through flow.

4- Network can be decomposed for a moderate number of traffic flows for some schedulers.

The possibility of queuing analysis of the nodes inside a network and regardless of the network has been investigated profoundly in the literature. The existing analyses assume BMux or FIFO links and are mostly under a many sources asymptotic. In Chapter 5, we have extended this concept to a non-asymptotic regime and to all Δ-schedulers. We have shown that although a network can be decomposed for all schedulers, eventually, as the number of traffic flows increases, in a non-asymptotic regime, we might still be able to decompose the network depending on the scheduling algorithms used in the network. Indeed, we have shown (by an example) that a network might be decomposed for some schedulers (including FIFO) even when the number of flows is only few hundreds. More precisely, the backlog analyses of the nodes without considering the network effect converges to that which accounts for the network effect as the number of flows increases. The convergence speed is a function of scheduling and varies from exponential (for $\Delta \geq 0$) to super exponential (for $\Delta < 0$).

6.2 Future Work

The results of this thesis can be used to continue research in two directions. The first one is extending this results to more general cases. Some interesting generalization opportunities are outlined in the following:

1- Other network scenarios:

In this thesis, we have analyzed a tandem network. An interesting, currently open problem is how tight we can analyze other network scenarios and how scheduling impacts the end-to-end delay and backlog in those scenarios. For instance, the authors of [6], [66] provide end-to-end deterministic FIFO analyses for a large set of network topologies. It is shown that the tightness of the resulting end-to-end delay bounds from Network Calculus depends on the network scenarios. For instance, it is proved that the bounds are tight in a sink-tree network while they are not tight in some nested network scenarios.

2- Other scheduling algorithms:

Our scheduling analysis includes some schedulers, while it excludes others. It would be interesting to expand our results to a more general group of schedulers and study their end-to-end behaviors as well. In particular, GPS is an interesting candidate to examine the end-to-end delay in a tandem of rate schedulers.

3- Other traffic types:

Although our convolution theorem (Theorem 4.4) and per-node service curve formulation (Theorem 3.1) can be applied to any SBB traffic, we only consider EBB as an example in the thesis. An interesting future work is to compute end-to-end delay and backlog bounds for FBM traffic (which is SBB, but not EBB) and check to see if the scheduling has similar impacts for this type of traffic.

The second direction for the continuing research in this field is to use the results of this thesis to gain new insights into the scheduling behavior. One of the interesting future studies of this sort is departure characterization in overloaded regime for Δ-schedulers. The departure characterization in FIFO schedulers in the overloaded regime in [29] is based on the FIFO service curve derived by Cruz in Eq. (3.17). We can apply our service curve formulation in Theorem 3.1 to see if a similar departure characterization exists for Δ-schedulers.

Bibliography

[1] M. Andrews. Probabilistic end-to-end delay bounds for earliest deadline first scheduling. In *Proc. of IEEE INFOCOM*, pages 603 – 612, March 2000.

[2] M. Andrews and L. Zhang. Minimizing end-to-end delay in high-speed networks with a simple coordinated schedule. In *Proc. of IEEE INFOCOM*, pages 380 – 388, March 1999.

[3] R. R. Bahadur and R. R. Rao. On deviations of the sample path mean. *The Annals of Mathematical Statistics*, 31(4):1015 – 1027, 1960.

[4] F. Baskett, K.M. Chancy, R. R. Muntz, and F. G. Palacios. Open, closed, and mixed networks of queues with different classes of customers. *Journal of the ACM*, 22(2):248 – 260, April 1975.

[5] J. Beran, R. Sherman, M. S. Taqqu, and W. Willinger. Long-range dependence in variable-bit-rate video traffic. *IEEE Transactions on Communications*, 43(2,3,4):1566 – 1579, February 1995.

[6] L. Bisti, L. Lenzini, E. Mingozzi, and G. Stea. Estimating the worst-case delay in FIFO tandems using network calculus. In *Proc. of ValueTools*, pages 1 – 10, October 2008.

[7] T. Bonald, A. Proutiere, and J. W. Roberts. Statistical performance guarantees for streaming flows using expedited forwarding. In *Proc. of IEEE INFOCOM*, pages 1104 – 1112, April 2001.

[8] T. Bonald and J. Roberts. Scheduling network traffic. *ACM SIGMETRICS Performance Evaluation Review*, 34(4):29 – 35, March 2007.

[9] R. R. Boorstyn, A. Burchard, J. Liebeherr, and C. Oottamakorn. Statistical service assurances for traffic scheduling algorithms. *IEEE Journal on Selected Areas in Communications*, 18(12):2651 – 2664, December 2000.

[10] J. Y. Le Boudec and P. Thiran. *Network Calculus*. Springer Verlag, Lecture Notes in Computer Science, LNCS 2050, 2001.

[11] P. T. Brady. A technique for investigating on-off patterns in speech. *Bell System Technical Journal*, 44(1):1 – 22, January 1965.

[12] E. Brockmeyer, H.L. Halstrom, and A. Jensen. *The Life and Works of A.K. Erlang*. Danish Academy of Technical Sciences, Denmark, 1948.

[13] A. Burchard. Personal Communications. June 2010.

[14] A. Burchard, J. Liebeherr, and F. Ciucu. On superlinear scaling of network delays. *IEEE/ACM Transactions on Networking*.

[15] A. Burchard, J. Liebeherr, and S. D. Patek. A min-plus calculus for end-to-end statistical service guarantees. *IEEE Transaction on Information Theory*, 52(9):4105 – 4114, September 2006.

[16] C.S. Chang. *Performance guarantees in communication networks*. Springer Verlag, 2000.

[17] V. Cholvi, J. Echage, and J-Y Le Boudec. Worst case burstiness increase due to FIFO multiplexing. *Performance Evaluation*, 49(1-4):491 – 506, September 2002.

[18] G. L. Choudhury, D. M. Lucantoni, and W. Whitt. Squeezing the most out of ATM. *IEEE Transactions on Communications*, 44(2):203 – 217, February 1996.

[19] F. Ciucu. Network calculus delay bounds in queueing networks with exact solutions. In *Proc. of ITC-20*, pages 495 – 506, June 2007.

[20] F. Ciucu, A. Burchard, and J. Liebeherr. Scaling properties of statistical end-to-end bounds in the network calculus. *IEEE Transactions on Information Theory*, 52(6):2300 – 2312, June 2006.

[21] F. Ciucu and J. Liebeherr. A case for decomposition of FIFO networks. In *Proc. of IEEE INFOCOM*, pages 1071 – 1079, April 2009.

[22] D. Clark, S. Shenker, and L. Zhang. Supporting real-time applications in an integrated services packet network: architecture and mechanism. In *Proc. of ACM SIGCOMM*, pages 14 – 26, August 1992.

[23] C. Courcoubetis and R. Weber. Buffer overflow asymptotics for a switch handling many trafic sources. *Journal of Applied Probability*, 33(3):886 – 903, 1996.

[24] M. E. Crovella and A. Bestavros. Self-similarity in world wide web traffic: evidence and possible causes. *IEEE/ACM Transactions on Networking*, 5(6):835 – 846, December 1997.

[25] R. L. Cruz. A calculus for network delay, part I : Network elements in isolation. *IEEE Transactions of Information Theory*, 37(1):114 – 121, January 1991.

[26] R. L. Cruz. A calculus for network delay, part II : Network analysis. *IEEE Transactions on Information Theory*, 37(1):132 – 141, January 1991.

[27] R. L. Cruz. Quality of service guarantees in virtual circuit switched networks. *IEEE Journal on Selected Areas in Communications*, 13(6):1048 – 1056, August 1995.

[28] R. L. Cruz. SCED+: efficient management of quality of service guarantees. In *Proc. of IEEE INFOCOM*, pages 625 – 634, April 1998.

[29] F. Cuicu, O. Hohlfeld, and L. Y. Chen. On the convergence to fairness in overloaded FIFO systems. In *Proc. of IEEE INFOCOM*, April 2011.

[30] A. Varma D. Stiliadis. Latency-rate servers: A general model for analysis of traffic scheduling algorithms. *IEEE/ACM Transactions on Networking*, 6(5):611 – 624, March 1998.

[31] J. N. Daigle and J. D. Langford. Models for analysis of packet voice communications systems. *IEEE Journal on Selected Areas in Communications*, 4(6):847 – 855, September 1986.

[32] G. de Veciana, C. Courcoubetis, and J. Walrand. Decoupling bandwidths for networks: a decomposition approach to resource management. In *Proc of IEEE INFOCOM*, pages 466 – 473, June 1994.

[33] A. Demers, S. Keshav, and S. Shenker. Analysis and simulation of a fair queueing algorithm. *SIGCOMM Computer Communication Review*, 19(4):1 – 12, September 1989.

[34] A. Elwalid, D. Heyman, T. V. Lakshman, D. Mitra, and A. Weiss. Fundamental bounds and approximations for ATM multiplexers with applications to video teleconferencing. *IEEE Jounal on Selected Areas of Communications*, 13:1004 – 1016, August 1995.

[35] A. Elwalid and D. Mitra. Analysis, approximations and admission control of a multiservice multiplexing system with priorities. In *Proc. of IEEE INFOCOM*, pages 463 – 472, April 1995.

[36] A. Elwalid and D. Mitra. Design of generalized processor sharing schedulers which statistically multiplex heterogeneous QoS classes. In *Proc. of IEEE INFOCOM*, pages 1220 – 1230, March 1999.

[37] A. Elwalid, D. Mitra, and R. H. Wentworth. A new approach for allocating buffers and bandwidth to heterogeneous, regulated traffic in an ATM node. *IEEE Journal on Selected Areas in Communications*, 13:1115 – 1127, August 1995.

[38] D. Y. Eun and N. B. Shroff. Network decomposition: theory and practice. *IEEE Transactions on Networking*, 13(3):526 – 539, June 2005.

[39] M. Fidler. An end-to-end probabilistic network calculus with moment generating functions. In *Proc. of IWQoS*, pages 261 – 270, June 2006.

[40] M. Fidler. Survey of deterministic and stochastic service curve models in the network calculus. *IEEE Communications Surveys Tutorials*, 12(1):59 – 86, 1st Quarter 2010.

[41] M. Fidler and S. Recker. Conjugate network calculus: A dual approach applying the legendre transform. *Computer Networks*, 50(8):1026 – 1039, June 2006.

[42] J. R. Gallardo, D. Makrakis, and L. Orozco-Barbosa. Use of α-stable self-similar stochastic processes for modeling traffic in broadband networks. *Performance Evaluation*, 40(1-3):71 – 98, March 2000.

[43] L. Georgiadis, R. Guerin aand V. Peris, and K. N. Sivarajan. Efficient network QoS provisioning based on per node traffic shaping. *IEEE Transactions on Networking*, 4(4):482 – 501, August 1996.

[44] Y. Ghiassi-Farrokhfal and J. Liebeherr. Output characterization of constant bit rate traffic in FIFO networks. *IEEE Communications Letters*, 13(8):618 – 620, August 2009.

[45] Y. Ghiassi-Farrokhfal, J. Liebeherr, and A. Burchard. The impact of link scheduling on long paths: Statistical analysis and optimal bounds. In *Proc. of IEEE INFOCOM*, April 2011.

[46] D. Gross, J. F. Shortle J. M. Thompson, and C. M. Harris. *Fundamentals of Queueing Theory*. John Wiley & Sons, Hoboken, NJ, 4'th edition, 2008.

[47] M. Grossglauser and S. Keshav. On CBR service. In *Proc. of IEEE INFOCOM*, pages 129 – 137, March 1996.

[48] R. Guerin, H. Ahmadi, and M. Naghshineh. Equivalent capacity and its application to bandwidth allocation in high-speed networks. *IEEE Journal on Selected Areas in Communications*, 9(7):968 – 981, September 1991.

[49] J. Guibert. Overflow probability upperbound in fluid queues with general onn/off sources. *Journal of Applied Probability*, 31(3):1134 – 1139, September 1994.

[50] Y. Jiang and Y. Liu. *Stochastic Network Calculus*. Springer Verlag, 2008.

[51] A. Karasaridis and D. Hatzinakos. Network heavy traffic modeling using α-stable self-similar processes. *IEEE Transactions on Communications*, 49(7):1203 – 1214, July 2001.

[52] M. Katevenis, S. Sidiropoulos, and C. Courcoubetis. Weighted round-robin cell multiplexing in a general-purpose ATM switch chip. *IEEE Journal on Selected Areas in Communications*, 9(8):1265 – 1279, October 1991.

[53] F. P. Kelly. Networks of queues with customers of different types. *Journal of Applied Probability*, 3(12):542 – 554, September 1975.

[54] F. P. Kelly. Effective bandwidth of multiclass queues. *Queueing Systems*, 9(1-2):5 – 16, March 1991.

[55] F. P. Kelly. Notes on effective bandwidths. In *Stochastic Networks: Theory and Applications. (Editors: F.P. Kelly, S. Zachary and I.B. Ziedins) Royal Statistical Society Lecture Notes Series, 4*, pages 141 – 168. Oxford University Press, 1996.

[56] G. Kesidis, J. Walrand, and C.-S. Chang. Effective bandwidths for multiclass markov fluids and other ATM sources. *IEEE/ACM Transactions on Networking*, 1(4):424 – 428, August 1993.

[57] L. Kleinrock. *Queueing Systems*, volume II: Application. Wiley Interscience, 1975.

[58] L. Kleinrock. *Queueing Systems*, volume I: Theory. Wiley Interscience, 1975.

[59] E. W. Knightly. H-bind: A new approach to providing statistical performance guarantees to vbr traffic. In *Proc. of IEEE INFOCOM*, pages 1091 – 1099, March 1996.

[60] E. W. Knightly. Second moment resource allocation in multi-service networks. In *Proc. of ACM SIGMETRICS*, pages 181–191, June 1997.

[61] E. W. Knightly. Enforceable quality of service guarantees for bursty traffic streams. In *Proc. of IEEE INFOCOM*, pages 635 – 642, March 1998.

[62] E.W. Knightly and N.B. Shroff. Admission control for statistical QoS: Theory and practice,. *IEEE Network*, 13:20 – 29, Mar./Apr 1999.

[63] A. Kumar, D. Manjunath, and J. Kuri. *Communication Networking: An Analytical Approach*. Elsevier, 2004.

[64] J. Kurose. On computing per-session performance bounds in high-speed multi-hop computer networks. In *Proc. of ACM SIGMETRICS*, pages 128 – 139, June 1992.

[65] N. Laskin, I. Lambadaris, F. Harmantzis, and M. Devetsikiotis. Fractional lvy motion and its application to network traffic modeling. *Computer Networks*, 40(3):363 – 375, October 2002.

[66] L. Lenzini, L. Martorini, E. Mingozzi, and G. Stea. Tight end-to-end per-flow delay bounds in FIFO multiplexing sink-tree networks. *Performance Evaluation*, 63(9-10):956 – 987, October 2006.

[67] C. Li, A. Burchard, and J. Liebeherr. A network calculus with effective bandwidth. *IEEE/ACM Transactions on Networking*, 16(6):1442 – 1453, December 2008.

[68] C. Li and E.W. Knightly. Coordinated multihop scheduling: a framework for end-to-end services. *IEEE/ACM Transactions on Networking*, 10(6):776 – 789, December 2002.

[69] J. Liebeherr. *The Internet as a Large-Scale Complex System*, pages 179 – 201. University Press, SFI Book Series, 2005.

[70] J. Liebeherr, A. Burchard, and F. Ciucu. Non-asymptotic delay bounds for networks with heavy-tailed traffic. In *Proc. of IEEE INFOCOM*, pages 1 – 9, March 2010.

[71] J. Liebeherr, M. Fidler, and S. Valaee. A system-theoretic approach to bandwidth estimation. *IEEE/ACM Transactions on Networking*, 18(4):1040 – 1053, August 2010.

[72] J. Liebeherr, Y. Ghiassi-Farrokhfal, and A. Burchard. Does link scheduling matter on long paths? In *Proc. of IEEE ICDCS*, pages 199 – 208, June 2010.

[73] J. Liebeherr, Y. Ghiassi-Farrokhfal, and A. Burchard. On the impact of link scheduling on end-to-end delays in large networks. *IEEE Journal on Selected Areas in Communications*, 29(5):1009 – 1020, May 2011.

[74] J. Liebeherr, S.D. Patek, and A. Burchard. Statistical per-flow service bounds in a network with aggregate provisioning. In *Proc. of IEEE INFOCOM*, pages 1680 – 1690, April 2003.

[75] J. Liebeherr, D.E. Wrege, and D. Ferrari. Exact admission control for networks with a bounded delay service. *IEEE/ACM Transactions on Networking*, 4(6):885 – 901, December 1996.

[76] N. Likhanov and R. R. Mazumdar. Cell loss asymptotics in buffers fed with a large number of independent stationary sources. In *Proc. of IEEE INFOCOM*, pages 339–346, March 1998.

[77] C. D. Locke. *Best-effort decision-making for real-time scheduling*. PhD thesis, Carnegie Mellon University, Pittsburgh, PA, USA, 1986.

[78] B. Maglaris, D. Anastassiou, P. Sen, G. Karlsson, and J.D. Robbins. Performance models of statistical multiplexing in packet video communications. *IEEE Transactions on Communications*, 36(7):834 – 844, July 1988.

[79] P. Mannersalo and I. Norros. Gps schedulers and gaussian traffic. In *Proc. of IEEE INFOCOM*, pages 1660 – 1667, June 2002.

[80] S. Mao and S. S. Panwar. A survey of envelope processes and their applications in quality of service provisioning. *IEEE Communications Surveys and Tutorials*, 8(1-4):2 – 20, 3rd Quarter 2006.

[81] A. K. Parekh and R. G. Gallager. A generalized processor sharing approach to flow control in integrated services networks: the single-node case. *IEEE/ACM Transactions on Networking*, 1(3):344 – 357, June 1993.

[82] V. Paxson and S. Floyd. Wide area traffic: the failure of poisson modeling. *IEEE/ACM Transactions on Networking*, 3(3):226 – 244, June 1995.

[83] V. V. Petrov. On the probabilities of large deviations for sums of independent random variables. *Theory of Probabilities and its Applications*, 34(2):287 – 298, March 1965.

[84] F. L. Presti, Z. Zhang, and D. Towsley. Bounds, approximations and applications for a two-queue gps system. In *Proc. of IEEE INFOCOM*, pages 1310 – 1317, March 1996.

[85] H. Sariowan. *A service-curve approach to performance guarantees in integrated-service networks*. PhD thesis, University of California, San Diego, 1996.

[86] M. Schwartz. *Broadband Integrated Networks*. Prentice Hall, 1996.

[87] V. Sivaraman and F. Chiussi. End-to-end statistical delay guarantees using earliest deadline first (EDF) packet scheduling. In *Proc. of IEEE GLOBECOM*, pages 1307 – 1312, December 1999.

[88] V. Sivaraman and F. M. Chiussi. Statistical analysis of delay bound violations at an earliest deadline first (EDF) scheduler. *Performance Evaluation*, 36-37:457–470, August 1999.

[89] V. Sivaraman and F. M. Chiussi. Statistical analysis of delay bound violations at an earliest deadline first (EDF) scheduler. *Performance Evaluation*, 36 – 37:457 – 470, 1999.

[90] V. Sivaraman and F. M. Chiussi. Providing end-to-end statistical delay guarantees with earliest deadline first scheduling and per-hop traffic shaping. In *Proc of IEEE INFOCOM*, pages 603–612, March 2000.

[91] V. Sivaraman, F. M. Chiussi, and M. Gerla. Traffic shaping for end-to-end delay guarantees with EDF scheduling. In *Proc. of IWQoS*, pages 10 – 18, June 2000.

[92] P. Skelly, M. Schwartz, and S. Dixit. A histogram-based model for video traffic behavior in an ATM multiplexer. *IEEE/ACM Transaction on Networking*, 1(4):446 – 459, August 1993.

[93] D. Starobinski and M. Sidi. Stochastically bounded burstiness for communication networks. In *Proc. of IEEE INFOCOM*, pages 36 – 42, Macrh 1999.

[94] I. Stoica and H. Zhang. Providing guaranteed services without per flow management. In *Proc. of ACM SIGCOMM*, pages 81 – 94, August 1999.

[95] M. Vojnovic and J.-Y. Le Boudec. Bounds for independent regulated inputs multiplexed in a service curve network element. *IEEE Transactions on Communications*, 51(5):735 – 740, May 2003.

[96] D. J. Wischik. The output of a switch, or, effective bandwidths for networks. *Queueing Systems: Theory and Applications*, 32(4):383 – 396, 1999.

[97] J. y. Qiu, C. Cetinkaya, C. Li, and E. W. Knightly. Inter-class resource sharing using statistical service envelopes. In *Proc. of IEEE INFOCOM*, pages 36 – 42, March 1999.

[98] O. Yaron and M. Sidi. Performance and stability of communication networks via robust exponential bounds. *IEEE/ACM Transactions on Networking*, 1(3):372 – 385, June 1993.

[99] O. Yaron and M. Sidi. Generalized processor sharing networks with exponentially bounded burstiness arrivals. In *Proc. of IEEE INFOCOM*, pages 628 – 634, June 1994.

[100] Q. Yin, Y. Jiang, S. Jiang, and P. Y. Kong. Analysis on generalized stochastically bounded bursty traffic for communication networks. In *Proc. of Local Computer Networks*, pages 141 – 149, November 2002.

[101] Y. Ying, F. Guillemin, R. Mazumdar, and C. Rosenberg. Buffer overflow asymptotics for multiplexed regulated traffic. *Performance Evaluation*, 65(8):555–572, 2008.

[102] Y. Ying, R. Mazumdar, C. Rosenberg, and F. Guillemin. The burstiness behavior of regulated flows in networks. In *Proc. of Networking 2005*, pages 918 – 929, May 2005.

[103] H. Zhang and D. Ferrari. Rate-controlled static-priority queueing. In *Proc. of IEEE INFOCOM*, pages 227 – 236, March 1993.

[104] H. Zhang and E. W. Knightly. Providing end-to-end statistical performance guarantees with bounding interval dependent stochastic models. In *Proc. of ACM SIGMETRICS*, pages 211 – 220, May 1994.

[105] L. Zhang. Virtual clock: a new traffic control algorithm for packet switching networks. In *Proc. of ACM SIGCOMM*, pages 19 – 29, September 1990.