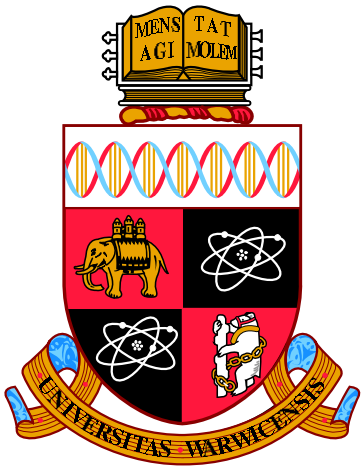


Stochastic Network Calculus with Martingales

by

Felix Heinrich Maria Poloczek

Thesis

Submitted to the University of Warwick

in partial fulfilment of the requirements

for admission to the degree of

Doctor of Philosophy

Department of Computer Science

The University of Warwick

June 2016

Ce vieillard qui avait usé sa vie à chercher une martingale, usait ses derniers jours à la mettre en œuvre, et ses dernières pièces à la voir échouer. La martingale est introuvable comme l'âme.

La Femme au collier de velours

Alexandre Dumas

Abstract

The practicality of the stochastic network calculus (SNC) is often questioned on grounds of looseness of its performance bounds. The reason for its inaccuracy lies in the usage of too elementary tools from probability theory, such as Boole's inequality, which is unable to account for correlations and thus inapropriate to properly model arrival flows.

In this thesis, we propose an extension of stochastic network calculus that characterizes its main objects, namely arrival and service processes, in terms of martingales. This characterization allows to overcome the shortcomings of the classical SNC by leveraging Doob's inequality to provide more accurate performance bounds. Additionally, the emerging *stochastic network calculus with martingales* is quite versatile in the sense that queueing related operations like multiplexing and scheduling directly translate into operations of the corresponding martingales. Concretely, the framework is applied to analyze the per-flow delay of various scheduling policies, the performance of random access protocols, and queueing scenarios with a random number of parallel flows.

Moreover, we show our methodology is not only relevant within SNC but can be useful also in related queueing systems. E.g., in the context of multi-server systems, we provide a martingale-based analysis of fork-join queueing systems and systems with replications.

Throughout, numerical comparisons against simulations show that the Martingale bounds obtained with Doob's inequality are not only remarkably accurate, but they also improve the Standard SNC bounds by several orders of magnitude.

Acknowledgments

First and foremost, I would like to thank my supervisor Florin Ciucu for all of his help over the last four years. Without his hard work none of my publications, and hence this thesis, would have been made possible.

Further, thanks go to Anja Feldmann for the opportunity to work as a research assistant in her group and for the amount of freedom she gave me to conduct my own research.

I am grateful to my co-author and Warwick office mate Amr Rizk for his ingenious ideas related to the use of submartingales in queueing theory and for long discussions on network calculus and related topics. Thanks also go to my further co-authors, Jens Schmitt and Oliver Hohlfeld.

Special thanks go to my former Berlin office mate Srivatsan Ravi for engaging in fruitful discussions on football, politics, and football politics. Further, to Arne Ludwig and Carlo Fürst for successfully establishing the concepts of "Eiszeit" and "Abstecher nach Flensburg", respectively. Thanks also go to the members of the "INET running club": Thomas Krenc, Niklas Semmler, and Damien Foucard.

Last but not least, I would like to thank my brother Ansgar and my good friends Sung-Gon, Alex, and Bartek for continuously reminding me that there is a life outside of academia.

Declarations

This thesis is submitted to the University of Warwick in support of the author's application for the degree of Doctor of Philosophy. It has been composed by the author and has not been submitted in any previous application for any degree.

The work presented (including data generated and data analysis) was carried out by the author except in the cases outlined below:

- the simulations in Chapters 3 and 7 were carried out by Florin Ciucu,
- the simulations in Chapter 6 were carried out by Amr Rizk,
- the submartingale representation in Chapter 6 is due to Amr Rizk,
- the extremal distributions from Subsection 5.2.3 are due to Florin Ciucu.

Parts of this thesis have been previously published by the author in the following publications:

Conference Papers:

[47] F. Ciucu, F. Poloczek, and J. Schmitt. Sharp per-flow delay bounds for bursty arrivals: The case of FIFO, SP, and EDF scheduling. In *IEEE Infocom*, pages 1896–1904, Apr. 2014.

[113] F. Poloczek and F. Ciucu. Scheduling analysis with martingales. *Performance Evaluation*, 79:56 – 72, Sept. 2014. Special Issue: Performance 2014.

[43] F. Ciucu, F. Poloczek, and O. Hohlfeld. On Capacity Dimensioning in Dynamic Scenarios: The Key Role of Peak Values. In *IEEE Lanman*, pages 1–6, May, 2014.

[42] F. Ciucu and F. Poloczek. On multiplexing flows: Does it hurt or not? In *IEEE Infocom*, pages 1122–1130, May 2015.

[114] F. Poloczek and F. Ciucu. Service-martingales: Theory and applications to the delay analysis of random access protocols. In *IEEE Infocom*, pages 945–953, May 2015.

[121] A. Rizk, F. Poloczek, and F. Ciucu. Computable bounds in fork-join queueing systems. In *ACM Sigmetrics*, pages 335–346, June 2015.

[44] F. Ciucu, F. Poloczek, and J. Schmitt. Stochastic upper and lower bounds for general markov fluids. In *International Teletraffic Congress (ITC)*. To appear.

Conference Posters:

[45] F. Ciucu, F. Poloczek, and J. Schmitt. Sharp bounds in stochastic network calculus. In *ACM Sigmetrics (Poster)*, pages 367–368, June 2013.

[115] F. Poloczek and F. Ciucu. Contrasting effects of replication in parallel systems: From overload to underload and back. In *ACM Sigmetrics (Poster)*, pages 375–376, June 2016.

Journals:

[120] A. Rizk, F. Poloczek, and F. Ciucu. Stochastic bounds in fork-join queueing systems under full and partial mapping. *Queueing Systems*. To appear.

Workshop Papers:

[112] F. Poloczek and F. Ciucu. A martingale-envelope and applications. In *Proc. of the ACM MAMA workshop*, 2013.

Technical Reports:

[46] F. Ciucu, F. Poloczek, and J. Schmitt. Sharp bounds in stochastic network calculus. *CoRR*, abs/1303.4114, 2013.

[116] F. Poloczek and F. Ciucu. Contrasting effects of replication in parallel systems: From overload to underload and back. *CoRR*, abs/1602.07978, 2016.

Sponsorship and Grant

This work is supportet in part by the *Deutsche Forschungsgemeinschaft (DFG)*, grant number Ci 195/1-1, “A Calculus for Networks with Flow Transformations (CAFLOTRA)”.

Abbreviations

a.s.	almost surely
CSMA/CA	carrier sense multiple access/collision avoidance
CCDF	complementary cumulative distribution function, $\bar{F}(x) = \mathbb{P}(X \geq x)$
EDF	earliest deadline first scheduling policy
$=_{\mathcal{D}}$	equality in distribution
FIFO	first in, first out scheduling policy
FJ	fork-join
FJR	fork-join with replication
i.i.d.	independent and identically distributed
$1_A(x)$	indicator function, $1_A(x) = 1$ if $x \in A$, and $1_A(x) = 0$ otherwise
MMOO	markov-modulated on-off
MAC	medium access control
$x \wedge y$	minimum, $x \wedge y = \min\{x, y\}$
MGF	moment generating function
MIMO	multiple input, multiple output
$[x]_+$	positive part, $[x]_+ = \max\{0, x\}$
r.v.	random variable

rng	range of function, $\operatorname{rng} f = \{y \mid \exists x : f(x) = y\}$
SP	static priority scheduling policy
SNC	stochastic network calculus
w.r.t.	with respect to

Contents

List of Figures

1

Introduction

Resource allocation is an old problem which perpetually reincarnates itself in resource sharing systems such as the telephone network, the Internet, or data centers. The first influential related analytical treatment was performed by Danish mathematician Agner Krarup Erlang who essentially looked at the problem of dimensioning the telephone network. One of Erlang's main results was a formula for the computation of the blocking probability that some shared resource is occupied; remarkably, amongst many applications, this formula has been used for nearly a century to dimension telephone networks.

Erlang's seminal work triggered the development of *queueing theory*, which has become an indispensable mathematical framework for the performance analysis of resource sharing systems. Over almost a century, this exact approach

to queueing theory (a.k.a. the classical approach) has been generalized to cover a broad class of networks, largely known by the product-form property (Baskett *et al.* [14], Kelly [87]). Besides its large scope, the class of product-form queueing networks is numerically tractable using convolution (Buzen [32]) or mean value analysis algorithms (Reiser and Lavenberg [118]).

Several alternatives to queueing theory have been developed to avoid the general limitation of Poisson arrivals of product-form networks. One is the *theory of effective bandwidth* (Kelly [89], Mazumdar [108]), which emerged in the 1990s as a unified framework to analyze the queueing behavior of broader classes of arrivals (e.g., deterministically regulated, Markovian, long-range dependent). The effective bandwidth is associated to an arrival flow and is essentially a number between the flow's average and peak rates, depending on some predefined Quality-of-Service constraint (e.g., margins on the buffer overflow probabilities). Unlike the classical approach, the performance metrics provided by the theory of effective bandwidth are typically given in *large buffer asymptotics* rather than in *exact results*, e.g., for the delay distribution W of some flow, the corresponding effective bandwidth approximation states that

$$\mathbb{P}(W > d) \sim \alpha e^{-\theta d} , \tag{1.1}$$

where α is the *asymptotic constant*, θ is the *asymptotic decay rate*, and $f(d) \sim g(d)$ means that $f(d)/g(d) \to 1$ as $d \to \infty$.

Another alternative to the classical approach is the *stochastic network calculus* (Chang [35], Jiang and Liu [81], Ciucu and Schmitt [48]), which can be considered as an extension of the effective bandwidth theory. Besides its ability to additionally deal with many scheduling algorithms and especially multi-queue scenarios, a fundamental difference of SNC (compared to the effective bandwidth theory) is that the results are provided as *probabilistic bounds*, e.g., for the delay distribution holds

$$\mathbb{P}(W > d) \leq \kappa e^{-\theta d} . \tag{1.2}$$

for some constant $\kappa > 0$.

The major advantage of SNC lies in two key features: *scheduling abstraction* and *convolution-form networks* (see Ciucu and Schmitt [48]). The former expresses the ability of SNC to compute per-flow (or per-class) queueing metrics for a large class of scheduling algorithms, in a unified manner, by decoupling scheduling from queueing analysis. Concretely, given an *arrival flow* A sharing a queueing system with other flows, the characteristics of the scheduling algorithm are first abstracted away in a so-called *service process* S; thereafter, the derivation of queueing metrics for the flow A is scheduling independent, i.e., independent of S. Furthermore, the per-flow results can be extended in a straightforward manner from a single queue to a large class of queueing networks, using convolution representations in a $(\min, +)$ algebra.

By relying on these two features, SNC could tackle several open queueing networks problems. The typical scenario involves the computation of non-asymptotic performance bounds of a single flow crossing a tandem network (i.e., a chain of queues which have to be traversed in order) and sharing the single queues with some other flows. Such scenarios were solved for a large class of arrival processes (see, e.g., Ciucu *et al.* [40, 29] and Fidler [59] for MMOO processes, and Liebeherr *et al.* [100] for heavy-tailed and self-similar processes). Another important solution was given for the delay distribution in a tandem (packet) network with Poisson arrivals and exponential packet sizes, by circumventing the so-called Kleinrock's independence assumption, which (artificially) assumes that the Poisson structure of the flows is immediately restored at each node in the network, (see Burchard *et al.* [28]). Other fundamentally difficult problems include the delay analysis of wireless channels under Markovian assumptions (see Zheng *et al.* [162]), the delay analysis of multi-hop fading channels (see Al-Zubaidy *et al.* [163]), bridging information theory and queueing theory by accounting for the stochastic nature and delay-sensitivity of real sources (see Lübben and Fidler [105]), or the computation of non-asymptotic per-flow capacity in ad-hoc networks (see Ciucu *et al.* [41]).

Based on its ability to solve some fundamentally hard queueing problems (in terms of bounds), SNC is justifiably proclaimed as a valuable alternative to the classical queueing theory (see Ciucu and Schmitt [48]). At the same time, SNC is also justifiably questioned on the tightness of its bounds. While the asymptotic tightness generally holds (see Chang [35, p. 291], and Ciucu *et al.* [40]), doubts on the bounds' numerical tightness shed skepticism on the practical relevance of SNC. This skepticism is supported by the fact that SNC largely employs the same probability methods as the effective bandwidth theory, which was argued to produce largely inaccurate results for non-Poisson arrival processes (see Abate *et al.* [2], Shroff and Schwartz [130]): E.g., in Choudhury *et al.* [39] it was convincingly conjectured through numerical experiments that delay bounds behave like

$$\mathbb{P}(W > d) \approx \kappa^{\#\text{flows}} e^{-\theta d} \ , \tag{1.3}$$

for some $0 < \kappa < 1$, whereas the corresponding constant α from Eq. (1.1) is oblivious to the number of flows. Hence, the bounds are "missing" an additional decay factor which is exponential in the number of flows.

From a technical point of view, the inaccuracy of the approximation from Eq. (1.2) stems from applying Boole's inequality, i.e.,

$$\mathbb{P}\left(\sup_n X_n \geq \sigma\right) \leq \sum_n \mathbb{P}\left(X_n \geq \sigma\right) \ . \tag{1.4}$$

to bound the supremum of a stochastic process X. It is known that this inequality is very loose, especially in non-Poisson scenarios (see Talagrand [137]).

One possibility to improve the bounds' accuracy, which was first undertaken by Kingman [92] to derive his classical GI/GI/1 bounds, and more recently extended by Duffield [55] to the analysis of Markov-Modulated On-Off (MMOO) processes, is to replace Boole's inequality (Eq. (1.4)) by Doob's inequality

$$\mathbb{P}\left(\sup_n X_n \geq \sigma\right) \leq \mathbb{E}[X_0]\sigma^{-1} \ , \tag{1.5}$$

which holds for the specific class of *(super-)martingales*. Besides the more technical advantage of providing a conceivably sharper inequality, there is also a conceptual similarity between supermartingales and queueing systems: A supermartingale roughly is a process such that for a given point in time any state in the future is expected to be less than the current. The same is true for the backlog- process in a queueing system: in order to guarantee finite performance metrics, the average arrivals have to be strictly less than the capacity, so that the increments are negative on average. This is typically ensured by a stability condition, like the one of Loynes.

The goal of this thesis is to systematically develop Kingman's martingale-based approach within the framework of stochastic network calculus. Concretely, the two key objects of SNC, i.e., the arrival- and the service processes A and S, will be characterized by suitably chosen *arrival-* and *service-martingales*, respectively. Whereas the arrival-martingales, enable the per-flow analysis of *random arrival flows* of queueing systems under scheduling, the service-martingales allow for the analysis of *random service models*, e.g., random access protocols. Concretely, arrival-martingales will be constructed for different arrival models including Markov-Modulated and autoregressive processes. In turn, service-martingale will be employed to model random access protocols like Aloha and CSMA/CA.

By exploiting Doob's inequality (Eq. (1.5)), we will see that the resulting performance metrics are throughout reasonably tight, hence revealing that the looseness of the state-of-the-art SNC bounds is generally not inherent to SNC itself, but due to the "temptatious" but "poisonous" elementary tools from probability theory (especially Eq. (1.4)) leveraged in its application.

Moreover, we will show that martingale-based techniques are not only useful in SNC but can be utilized in a more general queueing setup as well. Concretely, the related concept of a *submartingale* will be deployed to model the waiting- and response times of a multi-server queueing system where arriving jobs are either split into multiple subtasks or replicated to multiple servers and subsequently processed independently.

The simplicity of the obtained bounds together with its numerical accuracy could help to make the usage of martingales a valuable tool for the stochastic network calculus and related queueing theories.

1.1 Contributions

In this thesis, we provide the first *sharp* per-flow performance bounds for queueing systems with i.i.d., Markovian, and autoregressive arrivals under "First In, First Out" (FIFO), "Static Priority" (SP), and "Earliest Deadline First" (EDF) scheduling. The accuracy of these bounds contrasts the state-of-the-art bounds derived by the use of Boole's inequality (Eq. (1.4)) which will be shown to be off by several orders of magnitude; see Courcoubetis and Weber [49] for FIFO, Berger and Whitt [16] and Wischik [155] for Static Priority (SP), and Sivaraman and Chiussi [131] for Earliest-Deadline-First (EDF). Moreover, in our framework Eq. (1.3) holds in great generality, at the per-flow level for all considered scheduling policies. Hence, the bounds capture the exponential decay factor which was pointed out by Choudhury *et al.* [39] and by Botvich and Duffield [23].

In terms of service modelling, this thesis provides the first rigorous and accurate delay analysis in single-hop Aloha and CSMA/CA networks, subject to Markovian arrivals. By relying on a simplified CSMA/CA model proposed by Durvy *et al.* [56], which was argued to retain its key features, we extend a recent system theoretic approach by Ciucu *et al.* [41] by overcoming the limitations caused by the use of Boole's inequality.

We further investigate the often neglected "dynamic" queueing scenario with a *random* number of multiplexed flows. Assuming suitable independence assumptions, we extend the "folk theorem" in queueing theory, stating that determinism minimizes the queue size (see Rogozin [122], and Hajek [69]), to dynamic queues. In contrast, assuming a more realistic Markovian setup, the above folk theorem can fail. Concretely, we find that there is a phase transition in the flows' average lifetimes at which dynamic queue models yield (stochastically)

larger queues than the corresponding static queue models.

In the setup of parallel, i.e., multi-server, systems, we provide the first non-asymptotic and computable stochastic bounds on the waiting and response time in a fork-join queueing system in the most relevant scenarios. Concretely, we recover the $\mathcal{O}(\log K)$ asymptotic behavior (K being the number of servers) from Baccelli *et al.* [11], and Nelson and Tantawi [109]. Further, in the context of a replication queueing system, we first challenge the commonly used assumptions on statistical independence (see Gardner [65]) by providing some analytical arguments, that the benefits of replication are highly dependent on the corresponding correlation structure. Second, we develop a general analytical framework to compute stochastic bounds on the response time distributions in replication systems. In particular, our framework covers scenarios with Markovian arrivals, general service time distributions, and a correlation model amongst the original and replicated tasks.

1.2 Outline of the Thesis

In Chapter 2 the necessary background information for this thesis is provided: after briefly stating the probabilistic tools and techniques that are utilized in the sequel (Section 2.1), we give a short introduction to state-of-the-art stochastic network calculus (Section 2.2), including an outline of its major ideas (Subsection 2.2.1) and its current limitations (Subsection 2.2.2). In Chapter 3, we define the key object of this thesis, namely the *arrival-martingale*, as a novel characterization of arrival flows, and derive per-flow bounds of queueing metrics in systems under scheduling. In Chapter 4, we complement this setup with the characterization of the service processes (*service-martingale*), and show how it can be utilized to evaluate the performance of random access protocols like Aloha or CSMA/CA. In Chapter 5, we deploy this powerful martingale-methodology to investigate the impact of another source of randomness, namely the *random number of arrival flows*. Chapters 6 and 7 are devoted to the martingale-based

analysis of related queueing systems, namely *fork-join* queueing systems and systems with *replication*, respectively. Finally, in Chapter 8 we first give a brief discussion on the general quality of the bounds provided by the martingale-approach (Section 8.1), and lastly conclude the thesis (Section 8.2).

2

Background

2.1 Probability Theory

In this section we briefly state the probabilistic definitions and Theorems required for the remainder of this thesis. We assume throughout that all probabilistic objects are defined on a common *probability space* $(\Omega, \mathcal{A}, \mathbb{P})$. As usual, a *random variable* is a measurable function $X : \Omega \to \mathbb{R}$, its *expected value* is defined as

$$\mathbb{E}[X] = \int_{\Omega} X d\mathbb{P} .$$

Lemma 2.1 (Jensen's Inequality)**.** *Let X be a r.v., and $\varphi : \mathbb{R} \to \mathbb{R}$ a convex*

function. Then

$$\mathbb{E}\left[\varphi(X)\right] \geq \varphi(\mathbb{E}[X])$$

Proof. See e.g. [19, Eq. (5.33)]. □

A *stochastic process* is a sequence of random variables $(X_n)_{n\in I}$, where $I = \mathbb{N}$ or $I = \mathbb{Z}$.

Definition 2.2 (Stationarity)**.** *A stochastic process $(X_n)_{n\in\mathbb{N}}$ is* stationary *if its distribution is invariant under time-shifting, i.e., if for each $k \in \mathbb{N}$*

$$(X_n)_{n\in\mathbb{N}} =_{\mathcal{D}} (X_{k+n})_{n\in\mathbb{N}} \ .$$

Remark 2.3. *As a consequence of Kolmogorov's extension theorem (see e.g. [19, Theorem 36.1]), every stationary process $(X_n)_{n\in\mathbb{N}}$ can be extended to a stationary process $(X_n)_{n\in\mathbb{Z}}$.*

Definition 2.4 (Reversibility)**.** *For a stationary process $(X_n)_{n\in\mathbb{Z}}$, the* reversed process *$(X_n^r)_{n\in\mathbb{Z}}$ is defined as*

$$X_n^r := X_{-n} \ .$$

A process $(X_n)_{n\in\mathbb{Z}}$ is reversible*, if $(X_n^r)_{n\in\mathbb{Z}} =_{\mathcal{D}} (X_n)_{n\in\mathbb{Z}}$.*

Definition 2.5 (Filtration)**.** *A* filtration *$\mathcal{F} := (\mathcal{F}_n)_{n\in\mathbb{N}}$ is a sequence of increasing σ-algebras, i.e., $\mathcal{F}_n \subseteq \mathcal{F}_m$, for $n \leq m$.*

Definition 2.6 (Stopping Time)**.** *A random variable $N : \Omega \to \mathbb{N}$ is a* stopping time *w.r.t. a filtration $\mathcal{F}$, if for any $n \in \mathbb{N}$*

$$\{N = n\} \in \mathcal{F}_n \ .$$

The notion of a martingale is central for this thesis:

Definition 2.7 (Martingale)**.** *A stochastic process $(X_n)_n$ is a* martingale

w.r.t. the filtration $\mathcal{F}$ if for each $n \geq 1$

$$\mathbb{E}\left[X_n \mid \mathcal{F}_{n-1}\right] = X_{n-1} \ . \tag{2.1}$$

Further, $(X_n)_n$ is said to be a sub-(super-)martingale if in Eq. (2.1) we have $\geq$ ($\leq$) instead of equality.

If the filtration is not explicitly mentioned, it is to be understood as the generated filtration

$$\mathcal{F}_n = \sigma\left\{X_k \mid k \leq n\right\} \ .$$

Lemma 2.8 (Optional Stopping Theorem)**.** *Let $(X_n)_n$ be a martingale, and N a bounded stopping time, i.e., $N \leq n$ a.s., for some $n \geq 0$. Then*

$$\mathbb{E}\left[X_0\right] = \mathbb{E}\left[X_N\right] = \mathbb{E}\left[X_n\right] \ . \tag{2.2}$$

If X is only a sub-(super)-martingale, Eq. (2.2) is replaced by

$$\begin{aligned} \mathbb{E}\left[X_0\right] \leq \mathbb{E}\left[X_N\right] \leq \mathbb{E}\left[X_n\right] \quad & \textit{(submartingale), and} \\ \mathbb{E}\left[X_0\right] \geq \mathbb{E}\left[X_N\right] \geq \mathbb{E}\left[X_n\right] \quad & \textit{(supermartingale),} \end{aligned}$$

respectively.

Proof. See e.g. [19, Theorem 35.2]. □

Note that for any (possibly unbounded) stopping time N, the stopping time $N \wedge n$ is always bounded.

Lemma 2.9. *Let $(X_n)_n$ and $(Y_n)_n$ be independent (sub/super)-martingales, then the product $(X_n Y_n)_n$ is a (sub/super)-martingale as well.*

Proof. See e.g. [38, Theorem 2.1]. □

Although not a result of probability theory, the Perron-Frobenius theorem is included here as it is applied frequently in this thesis:

(a) single flow scenario (b) multiplexed flow scenario

Figure 2.1: Two queueing scenarios: (a) consists of a single flow A, whereas (b) has an additional cross-flow A'.

Lemma 2.10 (Perron-Frobenius Theorem)**.** *Let $A \in \mathbb{R}^{n\times n}$ be a real $n \times n$-matrix with only positive entries. Then A's maximal positive eigenvalue $\lambda(A)$ has a positive eigenvector, i.e., there is a vector $\overrightarrow{v} = (v_1, \ldots, v_n) \in \mathbb{R}^n$ such that*

$$A\overrightarrow{v} = \lambda(A)\overrightarrow{v} \ ,$$

and $v_1, \ldots, v_n > 0$.

Proof. See e.g. [77, Theorem 8.2.8]. □

We point out that Lemma 2.10 is only a special result of the classical Perron-Frobenius Theorem, as it only covers the case of strictly positive matrices.

2.2 Stochastic Network Calculus

In this section we give a brief introduction to Stochastic Network Calculus. In Subsection 2.2.1 we provide an overview of its general setup and its main ideas, which will form the basis for the remainder of this thesis. In Subsection 2.2.2 we outline the major techniques used so far to derive performance metrics and give an intuition why they lead to unsatisfactory results.

2.2.1 General Setup

We consider the queueing system from Figure 2.1: A data stream enters the system as an arrival flow A. After being stored in its buffer, a server processes the data with constant capacity $C > 0$ and the flow leaves the system as a departure flow D. The flow A may be the only flow under consideration (Figure 2.1(a)) or

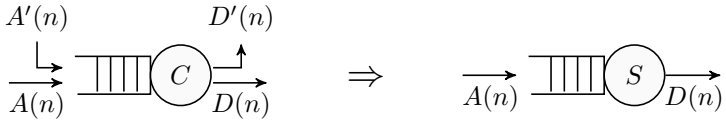


Figure 2.2: Scheduling abstraction: the cross-flow A' is encoded in the dynamic service process S.

may share the resource with some other flow A' (Figure 2.1(b)). In the latter case, the server additionally implements a *scheduling policy* which determines the priority allocated to the *through-flow* A and the *cross-flow* A'. This thesis is concerned with the *performance evaluation* of such a system under various assumption on its parameters; in particular, we are interested in estimating the backlog, i.e., the amount of data in the system, or the delay, i.e., the time a data unit stays in the system.

We assume a discrete-time scenario, the flows A and A' are given as bivariate stochastic processes

$$A(m,n) = \sum_{k=m+1}^{n} a_k \ , \quad A'(m,n) = \sum_{k=m+1}^{n} a'_k \ , \tag{2.3}$$

where $m, n \in \mathbb{Z}$, $m < n$, and the a_k and a'_k are nonnegative random variables describing the instantaneous arrivals at time k. Hence, $A(m,n)$ is the amount of data arriving at the system within the time interval $(m,n]$, by convention $A(m,n) := 0$, for $m \geq n$. We will frequently use the short-hand notation $A(n) := A(0,n)$ (and $A'(n) := A'(0,n)$).

We assume throughout that the $(a_n)_{n\in\mathbb{Z}}$ and $(a'_n)_{n\in\mathbb{Z}}$ are stationary stochastic processes (see Definition 2.2) defined on the set of integers $\mathbb{Z}$ (see Remark 2.3). Note that the definition of the *reversed process* (see Definition 2.4) extends to the bivariate processes A and A' by

$$A^r(m,n) := A(-n,-m) = \sum_{k=-n+1}^{-m} a_k \ , \text{ for } m < n \ ,$$

and analogously for A'^r.

The network calculus approach to address queueing systems starts with the *scheduling abstraction* (see Figure 2.2), i.e., with a transformation of the original queueing system (from Figure 2.1(b)) into an equivalent, but more amenable system, by encoding information about the capacity, the cross-flow, and the scheduling policy into a single *service process* $S(m,n)$. This service process S links *any* arrival process A with its corresponding departure process D through the inequality

$$D(n) \geq (A * S)(n) := \inf_{0 \leq m \leq n} \{A(0,m) + S(m,n)\} . \tag{2.4}$$

The service process S can be thought of as the departure process of a fictitious *saturated* arrival flow, i.e., a process A with $a_n = \infty$, for all $n \in \mathbb{Z}$. In some sense, the service process S is intimately related to the impulse-response of a linear and time invariant (LTI) system (for a discussion of this analogy see, e.g., [51, 24, 48]).

Service processes have been constructed for various scheduling policies, like "Static Priority", "First In, First Out", "Earliest Deadline First", etc. (see Chapter 3). Further, in Chapter 4, service processes will be constructed for random access protocols like Aloha and CSMA/CA. As for the arrival processes, we assume that the service processes are stationary in the sense that the distribution of $(S(m+k, n+k))_{m<n}$ is invariant to $k \in \mathbb{Z}$.

Through the coupling of A and D by Eq. (2.4), SNC is able to estimate the performance of the system. The performance metrics of interest are

1. the stationary *queue size* or *backlog*[1] Q, and
2. the stationary *virtual delay* W.

Queue Size Q: The queue size $Q(n)$ is defined as the amount of data within the system at time $n \in \mathbb{N}$, i.e.

$$Q(n) := A(n) - D(n) . \tag{2.5}$$

[1] Throughout this thesis, the terms "queue size" and "backlog" are used interchangeably

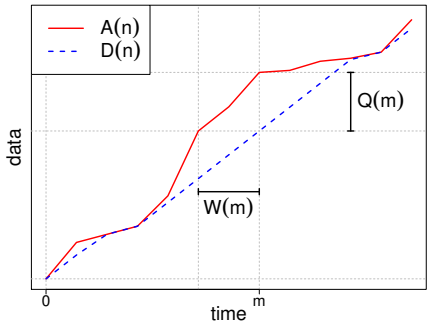


Figure 2.3: Queue size $Q(m)$ and virtual delay $W(m)$ as the vertical and horizontal distances between the curves A and D, respectively.

$Q(n)$ is the *vertical distance* between the curves A and D (see Figure 2.3). Although the queue size in Eq. (2.5) depends on the specific time parameter $n \in \mathbb{N}$, the following argument shows that it is possible to dispense with such temporal dependency: With the service process representation of the departure process from Eq. (2.4), one obtains

$$\begin{aligned} Q(n) &\leq A(n) - (A * S)(n) \\ &= A(n) - \inf_{0 \leq m \leq n} \{A(0,m) + S(m,n)\} \\ &= \sup_{0 \leq m \leq n} \{A(m,n) - S(m,n)\} \ . \end{aligned}$$

By the stationarity of the processes A and S (see Definition 2.2), it is possible to apply the time shift $n \rightsquigarrow 0$, such that the last line becomes

$$\begin{aligned} Q(n) &\leq \sup_{0 \leq m \leq n} \{A(m-n,0) - S(m-n,0)\} \\ &= \sup_{0 \leq m \leq n} \{A(-m,0) - S(-m,0)\} \\ &= \sup_{0 \leq m \leq n} \{A^r(m) - S^r(m)\} \ , \end{aligned} \tag{2.6}$$

where in the last line we utilized the reversed process representation (see Definition 2.4). By letting $n \to \infty$, one finally obtains (see also [35])

$$Q(n) \leq Q := \sup_{m \geq 0} \{A^r(m) - S^r(m)\} \ . \tag{2.7}$$

Assuming the stability condition $\mathbb{E}[a_1] < \mathbb{E}[S(1)]$, one can show that the *stationary queue size* Q is finite a.s. (see [35]). Throughout this thesis, the queue size is employed in the form of Eq. (2.7). We point out that Q is only an upper bound of the actual queue size of the system (at a certain point in time) as 1) Q includes the bound from the service process representation (Eq. (2.4)), and 2) Q involves the limit $n \to \infty$ between Eqs. (2.6) and (2.7).

Virtual Delay W: The virtual delay $W(n)$ is defined as the number of time steps a data unit would have stayed in the system had it departed at time $n \in \mathbb{N}$, i.e.,

$$W(n) := \inf\{k \in \mathbb{N} \mid A(n-k) \leq D(n)\} \ . \tag{2.8}$$

$W(n)$ is the *horizontal distance* between the curves A and D (see Figure 2.3). By monotonicity of the cumulative arrival process A one obtains with the service process representation from Eq. (2.4)

$$\begin{aligned} W(n) &\leq \inf\{k \in \mathbb{N} \mid A(n-k) \leq (A * S)(n)\} \\ &= \inf\left\{k \in \mathbb{N} \;\middle|\; A(n-k) \leq \inf_{0 \leq m \leq n} A(0,m) + S(m,n)\right\} \\ &= \inf\left\{k \in \mathbb{N} \;\middle|\; \sup_{0 \leq m \leq n} A(m, n-k) - S(m,n) \leq 0\right\} \ , \end{aligned}$$

(recall that by convention $A(m, n-k) = 0$ for $m \geq n-k$. Applying the time shift $n \rightsquigarrow 0$ this leads to:

$$\begin{aligned} W(n) &\leq \inf\left\{k \in \mathbb{N} \;\middle|\; \sup_{0\leq m\leq n} A(m-n,-k) - S(m-n,0) \leq 0\right\} \\ &= \inf\left\{k \in \mathbb{N} \;\middle|\; \sup_{0\leq m\leq n} A(-m,-k) - S(-m,0) \leq 0\right\} \\ &= \inf\left\{k \in \mathbb{N} \;\middle|\; \sup_{0\leq m\leq n} A^r(k,m) - S^r(m) \leq 0\right\} \ . \end{aligned}$$

By letting $n \to \infty$, one obtains

$$W(n) \leq W := \inf\left\{k \in \mathbb{N} \;\middle|\; \sup_{m\geq 0} A^r(k,m) - S^r(m) \leq 0\right\} \ ,$$

such that the following implication of events holds:

$$\begin{aligned} \{W \geq k\} &= \left\{\forall k' < k : \ \sup_{m\geq 0} A^r(k',m) - S^r(m) \ > \ 0\right\} \\ &\subseteq \left\{ \sup_{m\geq 0} A^r(k,m) - S^r(m) \ > \ 0\right\} \\ &\subseteq \left\{ \sup_{m\geq k} A^r(k,m) - S^r(m) \geq 0\right\} \ , \end{aligned} \tag{2.9}$$

where we used the monotonicity of A in the second and the positivity of S^r in the third line. Throughout this thesis, the *stationary virtual delay* W is employed in the form of Eq. (2.9).

2.2.2 Three Bounding Steps and One Pitfall

This section overviews the SNC approach to derive bounds for performance metrics. In addition to highlighting the underlying bounding steps, an elementary example proves that *careless bounding* can lend itself to impractical results.

We consider the queueing system from Figure 2.2, i.e., the arrival process $A(n)$ shares a server with capacity C with some other flow $A'(n)$. The information about C and $A'(n)$ is encoded in a service process $S(m,n)$. For the sake of simplicity, we confine ourselves to the case of the queue size Q, the derivations

for the virtual delay W are similar.

Recall from Eq. (2.6) that the queue size has an upper bound

$$\mathbb{P}(Q(n) \geq \sigma) \leq \sup_{0 \leq m \leq n} \{A^r(m) - S^r(m) \geq \sigma\} \ . \tag{2.10}$$

SNC typically continues with Eq. (2.10) by invoking Boole's inequality, i.e.,

$$\text{Eq. (2.10)} \ldots \leq \sum_{m=1}^{n} \mathbb{P}\left(A(m) - S(m) \geq \sigma\right) \ . \tag{2.11}$$

The probability events can be further estimated by using the Chernoff bound (i.e., $\mathbb{P}(X \geq x) \leq \mathbb{E}[e^{\theta X}]e^{-\theta x}$, for a r.v. X and $\theta > 0$):

$$\text{Eq. (2.11)} \ldots \leq \sum_{m=1}^{n} \mathbb{E}\left[e^{\theta(A(m)-S(m))}\right] e^{-\theta\sigma} \ , \tag{2.12}$$

for some $\theta > 0$. The expectation can be split into a product of expectations, according to the statistical independence properties of A and S, and the sum can be further reduced to some canonical form.

Eqs. (2.10)–(2.12) outline three major bounding steps. The first is "proprietary" to SNC, in the sense that it involves the specific construction of a "proprietary" service process S which decouples scheduling from analysis. The next two follow general purpose methods in probability theory, which are applied in the same form in the effective bandwidth theory.

In particular, the second step (i.e., Eq. (2.11)) reveals a convenient continuation of Eq. (2.10). The reason for this "temptatious" step to be consistently invoked in SNC stems from the "freedom" of seeking for bounds rather than exact results. As we will show, this "temptatious" step is also "poisonous" in the sense that it can lead to very loose bounds.

As a simple and yet illustrative example, let us consider the stationary process

$$A(n) = nX \ , \tag{2.13}$$

for all $n \geq 0$, where X is a Bernoulli random variable taking values in $\{0, 2\}$,

each with probabilities $1-\varepsilon > .5$ and $\varepsilon > 0$. Assume also that $S(m,n) = n-m$, i.e., a constant server with unit capacity. Clearly, for $\sigma > 0$, the backlog process satisfies for sufficiently large n

$$\mathbb{P}\left(Q(n) > \sigma\right) = \varepsilon \ .$$

In turn, the application of the bound from Eq. (2.11) lends itself to a trivial bound, i.e.,

$$\mathbb{P}\left(Q(n) > \sigma\right) \leq n\varepsilon \ ,$$

for $\sigma < 1$. The underlying reason behind this result is that Boole's inequality from Eq. (2.11) is agnostic to the statistical properties of the increments of the arrival process A. The construction of A from Eq. (2.13) illustrates thus the poor performance of Boole's inequality for arrivals with correlated increments.

In the next chapter, we will develop a framework that replaces Boole's inequality by Doob's inequality and show that, especially in correlated scenarios, this leads to accurate bounds.

3

Arrival Martingales

In this chapter we propose a novel representation of a queueing system's arrival flow by a suitable *arrival-martingale* and integrate it into the framework of stochastic network calculus. The crucial insight enabling the performance analysis is that typical queueing operations directly translate into operations of the respective martingales:

1. *Multiplexing* of flows translates into multiplying the corresponding martingales.

2. *Scheduling* translates into time-shifting the martingales, corresponding to the scheduled flows, at a specific shifting time parameter depending on the scheduling algorithm itself.

The second operation in particular highlights the instrumental role of the emerged SNC martingale framework to deal with the difficult problem of scheduling in a unified manner, roughly by decoupling scheduling through a *shifting parameter*; this shifting parameter can be tuned depending on scheduling, i.e., FIFO, SP, and EDF.

We apply our unified framework to the class of Markovian arrivals, and demonstrate for the first-time at the per-flow level that tail probabilities of the delay distribution exhibit an exponential decay in the number of flows (see Eq. (1.3)). Our results can be regarded as per-flow level extensions of the aggregate level results by Duffield [55].

We will also consider p-order autoregressive processes which can approximate the *whole* class of stationary processes (this property is typically referred to as Wold's decomposition, [26, p. 187]). Although autoregressive processes (of order $p = 1$) are also Markovian, their particular representation allows for a closed-form derivation of the performance bounds (the more general Markovian processes are subject to bounds in terms of implicit eigenvalues/vectors equations). More remarkably, unlike the results from [35, p. 340], which yield trivial (infinite) bounds when fitted for unbounded increment distributions, our results provide numerically accurate bounds.

For the rest of the chapter we first develop the theory of arrival-martingales in Section 3.1. Subsequently, in Section 3.2, we apply the emerging SNC framework to several classes of processes (independent increments, general Markovian arrivals, and p-order autoregressive processes).

3.1 A Calculus with Arrival-Martingales

We introduce our characterization of a queueing system by a certain supermartingale:

Definition 3.1 (Arrival-Martingale)**.** *The flow A admits arrival-martingales if for every $\theta \in (0, \theta_{max})$ there is a $K_a \geq 0$ and a function $h_a : \mathrm{rng}(a) \to \mathbb{R}^+$*

such that the process

$$M(n) := h_a(a_n)e^{\theta(A^r(n)-nK_a)} \ , \ n \geq 0 \ , \tag{3.1}$$

is a supermartingale.

The constant $\theta_{\max} > 0$ can be arbitrary, especially $\theta_{\max} = \infty$ is permitted. The index a, standing for "arrival", is needed as the definition is later (see Chapter 4) complemented with a similar definition for the service process. Note that the constant K_a and function h_a depend on $\theta > 0$.

An intuition for the definition is the following: In order to keep a queueing system in a stable regime, by Loynes' condition (see [104]), the average arrival rate has to be strictly less than the service rate. If one ignores the positivity constraint on the buffer, its expected increment (drift) is negative and thus the buffer content "resembles" a supermartingale. The conceptual reason for the exponential transform is that its shape directly determines the decay rate of queueing metrics (which for Markovian arrivals are exponential). From a technical point of view, the (convex) exponential transform assigns more weight to larger arrivals, reducing the negative drift and consequently the gap between the constructed supermartingale and a martingale. Moreover, since Doob's inequality does not differentiate between a supermartingale and a martingale, one looks to minimize the previous gap by maximizing the decay factor θ, which eventually determines the decay rate of the queueing metrics. Finally, the function h compensates for potential correlations among the increments; in particular, for i.i.d. increments, h is a constant.

Remark 3.2. *If Eq. (3.1) is a supermartingale, then by stationarity the "time-shifted" process*

$$h_a(a_{n+k})e^{\theta(A^r(k,n+k)-nK_a)}$$

is a supermartingale as well, for some fixed $k \geq 0$.

Let us now state an auxiliary definition which will become important in the general proofs of the performance metrics Q and W (see Theorem 3.4):

Definition 3.3 (Threshold)**.** *For h_a as in Definition 3.1 define the threshold H_a by*

$$H_a := \min \{h_a(x) \mid x > K_a\} \ .$$

H_a is the smallest value of $h_a(x)$ such that the instantaneous arrival is larger than the constant K_a. In many scenarios, the function h_a will be monotonically increasing such that we will have the simplification $H_a = h(K_a)$.

The next theorems and corollaries are the central results of this chapter, describing how arrival-martingales can be used to derive bounds on the performance metrics Q (queue size) and W (virtual delay). We start with the first scenario from Figure 2.1(a), i.e., considering the case of a single flow A, and a server with constant capacity $C > 0$:

Theorem 3.4 (Single Flow Bound)**.** *Assume that the flow A admits arrival-martingales, and let*

$$\theta^* := \sup \{\theta > 0 \mid K_a \leq C\} \ ,$$

then we have the following upper bound on the backlog and the virtual delay, respectively:

$$\mathbb{P}(Q \geq \sigma) \leq \frac{\mathbb{E}[h(a_n)]}{H_a} e^{-\theta^* \sigma} \ , \quad \mathbb{P}(W \geq k) \leq \frac{\mathbb{E}[h(a_n)]}{H_a} e^{-\theta^* kC} \ .$$

The proof of the theorem is basically a variant of Doob's inequality (see Eq. (1.5)). Adapted to the specific context, it will be used frequently in the sequel.

Proof. Consider first the queue size Q. Define the stopping time N by

$$N := \inf\{n \geq 0 \mid A^r(n) - nC \geq \sigma\} \ . \tag{3.2}$$

With the representation of the queue size Q from Eq. (2.7) (with $S(m, n) := (n - m)C$), it holds $\mathbb{P}(Q \geq \sigma) = \mathbb{P}(N < \infty)$. Applying the optional stopping theorem (Lemma 2.8) to the arrival-martingale (Eq. (3.1)) with parameter θ^*,

yields for every $m \in \mathbb{N}$:

$$\begin{aligned}\mathbb{E}[h(a_n)] &= \mathbb{E}[M(0)] \geq \mathbb{E}[M(N \wedge m)] \geq \mathbb{E}[M(N \wedge m)1_{N<m}] \\ &= \mathbb{E}[h(a_N)e^{\theta^*(A^r(N)-NK_a)}1_{N<m}] \\ &\geq \mathbb{E}[h(a_N)e^{\theta^*(A^r(N)-NC)}1_{N<m}] \\ &\geq H_a e^{\theta^*\sigma}\mathbb{P}(N < m) \ .\end{aligned}$$

In the last line we used the fact that by the inf-operator in Eq. (3.2), the last increment $a_N - C$ must be positive, i.e., $a_N > C \geq K_a$, and hence $h_a(a_N) \geq H_a$, a.s.. Now simply let $m \to \infty$.

For the virtual delay, recall from Eq. (2.9) that

$$\mathbb{P}\left(W \geq k\right) = \mathbb{P}\left(\sup_{n \geq k} A^r(k,n) - nC \geq 0\right)$$

Now define the stopping time N by

$$N := \inf\{n \geq k \mid A^r(k,n) - nC \geq 0\} \ , \tag{3.3}$$

such that $\mathbb{P}(W \geq k) = \mathbb{P}(N < \infty)$. Let

$$M(n) := h_a(a_n)e^{\theta^*(A^r(k,n)-(n-k)K_a)} \ , \quad n \geq k$$

be the time-shifted supermartingale from Remark 3.2. Similarly as for Q, by the optional stopping theorem for $m \geq k$ holds:

$$\begin{aligned}\mathbb{E}[h(a_n)] &= \mathbb{E}[M(k)] \geq \mathbb{E}[M(N \wedge m)] \geq \mathbb{E}[M(N \wedge m)1_{N<m}] \\ &= \mathbb{E}[h(a_N)e^{\theta^*(A^r(k,N)-(N-k)K_a)}1_{N<m}] \\ &\geq \mathbb{E}[h(a_N)e^{\theta^*(A^r(k,N)-(N-k)C)}1_{N<m}] \\ &\geq H_a e^{\theta^* kC}\mathbb{P}(N < m) \ .\end{aligned}$$

Now let $m \to \infty$. □

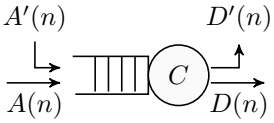


Figure 3.1: Multiplexed queueing scenario with a through-flow A and a cross-flow A'

Consider now the second scenario from Figure 3.1: two single flows A and A' with allocated capacities C_1 and C_2, respectively, are multiplexed into one queueing system with a shared total capacity of $C = C_1 + C_2$. The resulting system can be analyzed in two different ways: Firstly, for the aggregate system, both metrics Q and W can be estimated (*aggregate analysis*), and secondly, the virtual delay W for a single flow in the multiplexed system can be analyzed for several scheduling policies (*per-flow analysis*).

For both tasks, a technical definition is required:

Definition 3.5. *For two functions $h, h' : B \to \mathbb{R}_+$ ($B \subseteq \mathbb{R}$), define the* $(\min, \times)$*-convolution by*

$$(h \otimes h')(t) := \inf_{0 \le s \le t} h(s)h'(t-s) \ ,$$

for all $t \in B$.

Note that, by definition, for all a, b holds:

$$h \otimes h'(a+b) \le h(a)h'(b) \ . \tag{3.4}$$

3.1.1 Aggregate Analysis

We consider the queueing system as in Figure 3.1. The next theorem addresses the *aggregate analysis*, i.e., the analysis of aggregate arrivals $A + A'$:

Theorem 3.6 (Aggregate Flow Bound)**.** *Assume two independent arrivals A and A' admit arrival-martingales with parameters (h, K_a) and (h', K'_a), respectively. Then the aggregate flow $A+A'$ admits arrival-martingales with parameters $(h \otimes h', K_a + K'_a)$.*

Proof. Let a_n and a'_n denote the respective increment processes. Clearly, Eq. (3.4) implies for all n:

$$\begin{aligned} &h \otimes h'(a_n + a'_n)e^{\theta(A(n)+A'(n)-n\left(K_a+K'_a\right))} \\ &\leq h(a_n)e^{\theta(A(n)-nK_a)}h'(a'_n)e^{\theta(A'(n)-nK'_a)} \ , \end{aligned}$$

i.e., the product of the two arrival-martingales. By the independence assumption, this product is a supermartingale as well (see Lemma 2.9), and the proof is complete. □

The advantage of this theorem is that an aggregate flow can be handled in the same way as a single flow, e.g., for the constructed arrival-martingales, Theorem 3.4 can be evoked to derive the bounds on the backlog Q and the virtual delay W.

3.1.2 Per-Flow Analysis

We now turn to the *per-flow analysis* of flow A in the multiplexed queueing system equipped with a scheduling policy that determines the priority allocated to flows A, and A', respectively (Figure 3.1). The key element is the following technical lemma:

Lemma 3.7. *Assume the same situation as in Theorem 3.6. Then for every $l \geq 0$ and $\sigma > 0$ the following bound holds:*

$$\mathbb{P}\left(\sup_{n\geq l}\{A^r(l,n) + A'^r(0,n) - Cn\} \geq \sigma\right) \leq \frac{\mathbb{E}[h(a_n)]\mathbb{E}[h'(a'_n)]}{H_a}e^{-\theta(\sigma+lC_1)} \ ,$$

where H_a is the threshold from Definition 3.3 applied to the function $h \otimes h'$.

Proof. We proceed similarly as in the proof of Theorem 3.4. Consider the two supermartingales

$$\begin{aligned} M_1(n) &= h(a_n)e^{\theta(A^r(l,n)-(n-l)K_a)} \ , && n \geq l \text{ , and} \\ M_2(n) &= h(a'_n)e^{\theta\left(A'^r(n)-nK'_a\right)} \ , && n \geq 0 \end{aligned}$$

from the definition of the arrival martingales. By the independence assumption, the process

$$\tilde{M}(n) = M_1(n)M_2(n)$$

is a supermartingale in the *time-shifted* domain $\{l, l+1, l+2, \dots\}$. Let N denote a stopping time similar to the one from Eq. (3.2):

$$N := \inf\{n \geq l \mid A^r(l,n) + A'^r(0,n) - nC \geq \sigma\} \ . \tag{3.5}$$

Again, the desired probability is equal to $\mathbb{P}(N < \infty)$. By applying the optional stopping theorem (see Lemma 2.8), one has for $m \geq l$:

$$\begin{aligned}
\mathbb{E}[\tilde{M}(l)] \geq& \mathbb{E}[\tilde{M}(N \wedge m)] \\
\geq& \mathbb{E}[\tilde{M}(N \wedge m) 1_{N<m}] \\
=& \mathbb{E}[h(a_n)h'(a'_n)e^{\theta\left(A^r(l,n)-(n-l)C_1+A'^r(n)-nC_2\right)}1_{N<m}] \\
=& \mathbb{E}[h(a_n)h'(a'_n)e^{\theta\left(A^r(l,n)+A'^r(n)-nC+lC_1\right)}1_{N<m}] \\
\geq& H_a e^{\theta(\sigma+lC_1)}\mathbb{P}(N < m)
\end{aligned}$$

Now, by independence and the supermartingale property of M':

$$\begin{aligned}
\mathbb{E}[\tilde{M}(l)] =& \mathbb{E}[M_1(l)M_2(l)] = \mathbb{E}[M_1(l)]\mathbb{E}[M_2(l)] \\
\leq& \mathbb{E}[h(a_n)]\mathbb{E}[M_2(0)] = \mathbb{E}[h(a_n)]\mathbb{E}[h'(a'_n)] \ .
\end{aligned}$$

As above, we finally let $m \to \infty$ to complete the proof. □

The crucial parameter in Lemma 3.7 is the parameter l, indicating how many points in time the process A is delayed. This parameter can be adjusted according to the scheduling policy under consideration, or more precisely to the expression of the service process S depicted in Figure 2.2. We will next apply Lemma 3.7 and properly tune the parameter l for SP, FIFO, and EDF scheduling.

Recall from Eq. (2.9) that for the virtual delay holds

$$\mathbb{P}(W \geq k) \leq \mathbb{P}\left(\sup_{n \geq k}\{A^r(k,n) - S^r(n)\} \geq 0\right) . \tag{3.6}$$

Static Priority (SP) This scheduling policy always gives priority to the cross-flow A'. The service process $S(m,n)$ is given by (see [59]):

$$S(m,n) = [C(n-m) - A'(m,n)]_+ . \tag{3.7}$$

Corollary 3.8 (SP Per-Flow Bound)**.** *Consider the situation as in Theorem 3.6, with SP as the scheduling policy. Then for the virtual delay W for flow A holds:*

$$\mathbb{P}(W \geq k) \leq \frac{\mathbb{E}[h(a_n)]\mathbb{E}[h'(a'_n)]}{H_a} e^{-\theta C_1 k} .$$

Proof. In continuation of Eq. (3.6) with the service process as in Eq. (3.7):

$$\begin{aligned}
\mathbb{P}(W \geq k) &\leq \mathbb{P}\left(\sup_{n \geq k}\{A^r(k,n) - S^r(0,n)\} \geq 0\right) \\
&= \mathbb{P}\left(\sup_{n \geq k}\{A^r(k,n) - [Cn - A'^r(n)]_+\} \geq 0\right) \\
&\leq \mathbb{P}\left(\sup_{n \geq k}\{A^r(k,n) + A'^r(n) - Cn \geq 0\}\right) .
\end{aligned}$$

Now simply plug in the parameters $\sigma = 0$ and $l = k$ into Lemma 3.7. □

First In, First Out (FIFO) For FIFO, the service process $S(m,n)$ is given by (see [50]):

$$S(m,n) = [C(n-m) - A'(m,n-x)]_+ 1_{\{n-m>x\}} , \tag{3.8}$$

where $x \geq 0$ is a parameter freely chosen, but fixed. Note that for the specific choice of $x := 0$, one recovers the service process for SP from Eq. (3.7), corresponding to the fact that the through-flow's performance in a FIFO system is

upper-bounded by its performance in a SP system.

Corollary 3.9 (FIFO Per-Flow Bound)**.** *Consider the situation as in Theorem 3.6, with FIFO as the scheduling policy. Then for the virtual delay W holds:*

$$\mathbb{P}(W \geq k) \leq \frac{\mathbb{E}[h(a_n)]\mathbb{E}[h'(a'_n)]}{H_a} e^{-\theta C k} \ .$$

Proof. For the free parameter in the service process from Eq. (3.8) choose $x = k$. Then Eq. (3.6) continues to:

$$\begin{aligned}
\mathbb{P}(W \geq k) &\leq \mathbb{P}(\sup_{n \geq k}\{A^r(k,n) - S^r(0,n)\} \geq 0) \\
&= \mathbb{P}(\sup_{n \geq k}\{A^r(k,n) - [Cn - A'^r(n-k)]_+ 1_{\{n>k\}}\} \geq 0) \\
&\leq \mathbb{P}(\sup_{n \geq 0}\{A^r(n) + A'^r(n) - C(n+k)\} \geq 0) \ .
\end{aligned}$$

Now apply Lemma 3.7 with $l = 0$ and $\sigma = Ck$. □

Note the difference in the decay rate: Whereas for SP it is the per-flow capacity C_1, for FIFO we have the total capacity $C = C_1 + C_2$.

Earliest Deadline First (EDF) Now consider the case of EDF scheduling. An EDF server associates fixed relative deadlines d and d' with the flows A and A', respectively. All data units are served in the order of their remaining deadlines, even when they are negative (we do not consider data loss). Note that in the extreme cases $d' < d = \infty$ and $d = d'$, we recover the situation of SP and FIFO scheduling, respectively. The service process $S(m,n)$ is given by (see [101]):

$$S(m,n) = [C(n-m) - A'(m, n - x + \min\{x,y\})]_+ 1_{\{n-m>x\}} \ , \tag{3.9}$$

where $x \geq 0$ is again a free parameter, and $y := d - d'$ denotes the difference between the respective deadlines. It is convenient to distinguish between the

cases $y \geq 0$ and $y < 0$.

Let us first consider the case $y \geq 0$:

Corollary 3.10 (EDF Per-Flow Bound, $y \geq 0$)**.** *Assume EDF is used as scheduling policy, $y \geq 0$, and consider the situation as in Theorem 3.6. Then for the virtual delay W holds:*

$$\mathbb{P}(W \geq k) \leq \frac{\mathbb{E}[h(a_n)]\mathbb{E}[h'(a'_n)]}{H_a} e^{-\theta(Ck - C_2 \min\{k,y\})} \ .$$

Proof. Again, let $x := k$. Eq. (3.6) with the service process from Eq. (3.9) gives:

$$\begin{aligned}
\mathbb{P}\left(W \geq k\right) &\leq \mathbb{P}\left(\sup_{n \geq k} \{A^r(k,n) + A'^r(n - k + \min\{k,y\}) - Cn\} \geq 0\right) \\
&\leq \mathbb{P}\Bigg(\sup_{\tilde{n} \geq \min\{k,y\}} \{A^r(k, \tilde{n} + k - \min\{k,y\}) + A'^r(\tilde{n}) \\
&\qquad\qquad - C(\tilde{n} + k - \min\{k,y\}\} \geq 0\Bigg) \\
&\leq \mathbb{P}\Bigg(\sup_{\tilde{n} \geq \min\{k,y\}} \{A^r(\min\{k,y\}, \tilde{n}) + A'^r(\tilde{n}) - C\tilde{n}\} \\
&\qquad\qquad \geq C(k - \min\{k,y\})\Bigg) \ ,
\end{aligned}$$

where we used the substitution

$$\tilde{n} = n - k + \min\{k,y\}$$

in the third, and the stationarity of A^r in the fourth line. Now apply Lemma 3.7 with $l = \min\{k,y\}$, and $\sigma = C(k - \min\{k,y\})$; hereby note that $l \geq 0$ and $\sigma - cl = Ck - c' \min\{k,y\}$. □

Consider now the case $y = d - d' < 0$. This is more difficult as now $\min\{k,y\} = y < 0$, so that for

$$n_0 \in B := \{n \geq k \mid n < k - y\} \ ,$$

the argument $n_0 - k + \min\{k,y\}$ is negative as well. By definition (again from

[101]), for those $n_0 \in B$:

$$A'^r(n_0 - k + \min\{k, y\}) = 0 \ . \tag{3.10}$$

Corollary 3.11 (EDF Per-Flow Bound, $y < 0$)**.** *Assuming EDF scheduling with $y < 0$, for the virtual delay W holds:*

$$\mathbb{P}(W \geq k) \leq \frac{\mathbb{E}[h(a_n)]\mathbb{E}[h'(a_n)]}{H_a} e^{-\theta(Ck - C_2 y)} + \frac{\mathbb{E}[h(a_n)]}{H_a} e^{-\tilde{\theta} Ck} \ ,$$

where $\tilde{\theta}$ is the parameter such that the flow A admits an arrival-martingale with $K_a = C$. Note that as $C > C_1$, such a $\tilde{\theta}$ exists and is greater than θ.

Proof. By splitting up the probability in Eq. (3.6) using the Boole's inequality

$$\begin{aligned}\mathbb{P}(W \geq k) \leq & \mathbb{P}\Big(\sup_{n \geq k : n \notin B} \{A^r(k,n) - S^r(0,n)\} \geq 0\Big) \\ & + \mathbb{P}\Big(\sup_{n \geq k : n \in B} \{A^r(k,n) - S^r(0,n)\} \geq 0\Big) \ ,\end{aligned}$$

one has for the first probability:

$$\begin{aligned}& \mathbb{P}\Big(\sup_{n \geq k : n \notin B} \{A^r(k,n) - S^r(0,n)\} \geq 0\Big) \\ & \leq \mathbb{P}\Big(\sup_{n \geq k - y} \{A^r(k,n) + A'^r(n - k + y) - Cn\} \geq 0\Big) \\ & \leq \mathbb{P}\Big(\sup_{\tilde{n} \geq -y} \{A^r(\tilde{n}) + A'^r(-y, \tilde{n}) - C\tilde{n}\} \geq Ck\Big) \\ & \leq \frac{\mathbb{E}[h(a_n)]\mathbb{E}[h'(a'_n)]}{H_a} e^{-\theta(Ck - C_2 y)} \ .\end{aligned}$$

In the third line, stationarity and the substitution $\tilde{n} = n - k$ was used, and in the fourth line Lemma 3.7 was applied with $\sigma = Ck$, $l = -y$, and the roles of A and A' were interchanged.

For the second probability with Eq. (3.10):

$$\begin{aligned}
&\mathbb{P}(\sup_{n\geq k:n\in B}\{A^r(k,n)-S^r(0,n)\}\geq 0)\\
&\leq \mathbb{P}(\sup_{k\leq n<k-y}\{A^r(k,n)-Cn\}\geq 0)\\
&= \mathbb{P}(\sup_{0\leq \tilde{n}<-y}\{A^r(\tilde{n})-C(\tilde{n}+k)\}\geq 0)\\
&\leq \mathbb{P}(\sup_{\tilde{n}\geq 0}\{A^r(\tilde{n})-C\tilde{n}\}\geq Ck)\\
&\leq \frac{\mathbb{E}[h(a_n)]}{H_a}e^{-\tilde{\theta}Ck}\ ,
\end{aligned}$$

with the usual substitution $\tilde{n} = n - k$ and the stationarity assumption in the fourth line. In the last line, Theorem 3.4 with $\sigma = Ck$ were used. □

3.2 Applications

In this section we demonstrate the versatility of the proposed calculus with arrival-martingales to address several classes of arrival processes: with independent increments (Subsection 3.2.1), with Markovian increments (Subsection 3.2.2), and p-order autoregressive (Subsection 3.2.3).

3.2.1 Processes with Independent Increments

One of the simplest traffic model is given by a process with independent increments, i.e., $A(m,n) = \sum_{k=m+1}^{n} a_k$, where $(a_k)_k$ is a sequence of i.i.d. random variables with positive distribution. Although not realistic, this example is included here because it provides a good intuition on how the calculus with arrival-martingales works.

Lemma 3.12. *In the situation above, the flow A admits arrival-martingales.*

Proof. For $\theta > 0$ let $h_a \equiv 1$ and define K_a by

$$K_a = \log \mathbb{E}[e^{\theta a_1}]/\theta\ . \tag{3.11}$$

According to the i.i.d. assumption we have:

$$\begin{aligned}\mathbb{E}\left[h(a_{n+1})e^{\theta(A(n+1)-(n+1)K_a)} \;\middle|\; a_1,\ldots,a_n\right] &= e^{\theta(A(n)-nK_a)}E\left[e^{\theta a_{n+1}}\right]e^{-\theta K_a}\\ &= h(a_n)e^{\theta(A(n)-nK_a)}\ ,\end{aligned}$$

proving the arrival-martingale. □

Let the capacity $C > 0$ satisfy the two stability conditions

$$\mathbb{E}[a_1] < C < \sup a_1\ , \tag{3.12}$$

to avoid the trivial scenarios of no queueing at all and infinite queue size, respectively. Combining the martingale-envelope from Lemma 3.12 with the general theory from Section 3.1, the following bounds hold:

Corollary 3.13 (Bounds for i.i.d. Arrivals)**.** *Consider an i.i.d. arrival flow $(a_n)_n$, and a capacity C such that the condition from Eq. (3.12) holds. Then, with*

$$\theta^* := \sup\{\theta \geq 0 \mid K_a \leq C\}$$

for this single flow holds:

$$\mathbb{P}(Q \geq \sigma) \leq e^{-\theta^*\sigma}\ ,\ \textit{and}\quad \mathbb{P}(W \geq k) \leq e^{-\theta^* C_1 k}\ .$$

In a scenario with flows A and A' and capacity $C = C_1 + C_2$ (satisfying the corresponding stability conditions), for flow A holds in the multiplexed queueing system under scheduling:

$$\begin{aligned}\text{FIFO:}\quad & \mathbb{P}(W \geq k) \leq e^{-\theta^* Ck}\\ \text{SP:}\quad & \mathbb{P}(W \geq k) \leq e^{-\theta^* C_1 k}\\ \text{EDF1:}\quad & \mathbb{P}(W \geq k) \leq e^{-\theta^*(Ck - C_2\min\{k,y\})}\\ \text{EDF2:}\quad & \mathbb{P}(W \geq k) \leq e^{-\theta^*(Ck+C_2 y)} + e^{\tilde{\theta}C_2 k}\ ,\end{aligned}$$

where $y = d - d'$, $C = C_1 + C_2$, and $\tilde{\theta}$ is the parameter of flow A in the system with total capacity C.

EDF1 and EDF2 correspond to the cases $y \geq 0$ and $y < 0$, respectively (see Corollaries 3.10 – 3.11).

Proof. Use the arrival-martingales from Lemma 3.12. For the first part, apply Theorem 3.4. For the second apply Corollaries 3.8 – 3.11. □

Note that the aggregate analysis of the whole system (as in Subsection 3.1.1) is contained in the first part of Corollary 3.13, as the resulting aggregate flow $(a_n + a'_n)_n$ is still i.i.d.

Remark 3.14. *By definition, the parameter θ^* can assume any nonnegative value including 0, and ∞. Assuming the stability condition from Eq. (3.12), the following argument shows that in fact $0 < \theta^* < \infty$: Consider the two continuous functions*

$$\varphi_1(\theta) := \mathbb{E}[e^{\theta a_1}] \quad \text{and } \varphi_2(\theta) := e^{\theta C} \ .$$

Due to the first stability condition from Eq. (3.12) we know that

$$\frac{d}{d\theta}\varphi_1(\theta)\bigg|_{\theta=0} = \mathbb{E}[a_1] < C = \frac{d}{d\theta}\varphi_2(\theta)\bigg|_{\theta=0} \ ,$$

i.e., (since $\varphi_1(0) = \varphi_2(0) = 1$) there is $\varepsilon > 0$ such that $\varphi_1 < \varphi_2$ on $[0, \varepsilon]$. Due to the second stability condition, φ_1 will eventually become larger than φ_2, and so by continuity there exists $\theta^* > 0$ such that $\varphi_1(\theta^*) = \varphi_2(\theta^*)$.

In Figure 3.2 simulations of the i.i.d. scenario are displayed together with the corresponding bounds for SP and EDF[1]. The Martingale bounds (from Corollary 3.13) almost match the simulations, whereas the bounds computed with Boole's inequality are off by several orders of magnitude.

[1]For this figure (and remaining figures in this chapter), 100 independent simulations were run, each consisting of 10^9 packets. To ensure a stationary regime, the first 10^8 packets in each run were discarded. The resulting (empirical) CCDFs are presented as box-plots.

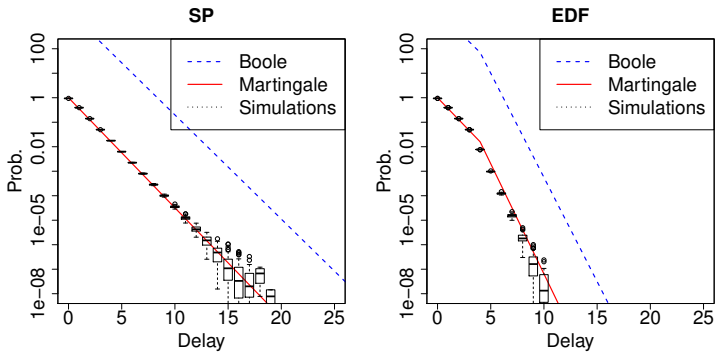


Figure 3.2: CCDF of the virtual delay (i.i.d.-case): $10 + 10$ exponentially distributed subflows with $\lambda = 1$, utilization $\rho = 0.95$, and, for EDF, $y = d - d' = 4$.

3.2.2 Processes with Markovian Increments

The previous independence assumption on the increments is now replaced by a *Markovian correlation structure*, i.e., the process $a_n := f(x_n)$ is driven by a Markov chain $(x_n)_{n\in\mathbb{N}}$ with state space $\mathcal{S} = \{1, 2, \ldots, s_{max}\}$. Here, $f : \mathcal{S} \to \mathbb{R}^+$ is an injective and deterministic function. To ensure stationarity, we assume x_n to be in steady state.

Let π denote its stationary distribution, and T the $s_{max} \times s_{max}$-transition matrix of the *reversed process*, i.e.,

$$\pi(i) = \mathbb{P}(x_n = i) \quad \text{and} \quad T(i,j) = \mathbb{P}\left(x_{n-1} = j \mid x_n = i\right) \ .$$

In many cases, the Markov chain is *reversible* and the matrix T coincides with the transition matrix of a_n itself. Now, for any $\theta \geq 0$, let T_θ denote the *exponentially transformed* transition matrix, i.e.,

$$T_\theta(i,j) = T(i,j)e^{\theta f(j)} \ , \tag{3.13}$$

clearly, $T = T_0$. The following martingale construction can be found in [55]:

Lemma 3.15. *In the situation above, the flow A admits arrival-martingales.*

Proof. Let $\theta > 0$ and let $\lambda(\theta)$ denote the spectral radius of T_θ and $v \in \mathbb{R}^{s_{max}}$ a

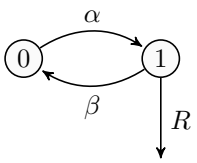


Figure 3.3: An arrival process modelled in terms of a Markov-Modulated On-Off (MMOO) process

corresponding eigenvector. Note that by the Perron-Frobenius Theorem $\lambda(\theta)$ is positive and v can be chosen to have positive components. With the function h_a defined by $h_a(f(i)) = v_i$ we can write for arbitrary $K > 0$:

$$\begin{aligned}
&\mathbb{E}\left[h_a(a_{n+1})e^{\theta(A(n+1)-(n+1)K)} \;\middle|\; x_1,\ldots,x_n\right] \\
&= e^{\theta(A(n)-nK)}\mathbb{E}\left[h_a(a_{n+1})e^{\theta f(x_{n+1})} \;\middle|\; x_n\right] e^{-\theta K} \\
&= e^{\theta(A(n)-nK)}\left(T^\theta v\right)(x_n)\, e^{-\theta K} \\
&= h_a\left(a_n\right) e^{\theta(A(n)-nK)}\lambda(\theta)e^{-\theta K} \;,
\end{aligned}$$

Substituting

$$K_a := \frac{\log \lambda(\theta)}{\theta} \tag{3.14}$$

for K proves the martingale property. □

As an application of Lemma 3.15 consider the arrival model as a *Markov Modulated On-Off Process* (MMOO) (Figure 3.3), i.e., a Markov chain x_n jumping between the two states 0 ("Off") and 1 ("On") with probabilities α and β, respectively. While in state 1 it transmits R data units per time unit, while in state 0 it does not transmit any data. Hence, $a_n := Rx_n$. The stationary distribution of a_n is given by:

$$\pi_0 := \mathbb{P}(a_n = 0) = \frac{\beta}{\alpha+\beta}, \quad \pi_1 := \mathbb{P}(a_n = R) = \frac{\alpha}{\alpha+\beta} \;, \tag{3.15}$$

and the process is reversible, i.e., $A = A^r$. We additionally assume that the

Markov chain satisfies the "burstiness condition"

$$\alpha < 1 - \beta \ ,$$

i.e., the probability of jumping to the "On"-state is strictly less than the probability of staying there. The advantage of this condition is that it is equivalent to the eigenvector v (as defined in Lemma 3.15) being monotonically increasing, i.e.,

$$v_0 < v_1 \Leftrightarrow \alpha < 1 - \beta \ , \tag{3.16}$$

for a proof see [27].

As an immediate consequence of Theorem 3.4 we now have:

Corollary 3.16 (Bounds for MMOO Arrivals)**.** *Consider the MMOO arrival flow as above and a capacity C satisfying $C > R\pi_1 = \mathbb{E}[a_n]$. With θ^* such that*

$$\log \lambda(\theta^*) = \theta^* C$$

for the backlog Q and the virtual delay W holds:

$$\mathbb{P}(Q \geq \sigma) \leq \kappa e^{-\theta^* \sigma} \ , \text{and} \quad \mathbb{P}(W \geq k) \leq \kappa e^{-\theta^* k C} \ ,$$

where $\kappa := \frac{\alpha + \beta v_0 / v_1}{\alpha + \beta}$. Moreover, for the constant holds $\kappa < 1$.

Proof. The existence of θ^* follows from the Perron-Frobenius Theorem (see Lemma 2.10), as

$$1 < \min_i \sum_j T^\theta_{i,j} \leq \lambda(\theta) \leq \max_i \sum_j T^\theta_{i,j} \leq e^{\theta \max_i f(x_i)} < \infty \ .$$

Apply Theorem 3.4 to the martingale-envelope constructed in Lemma 3.15. For the threshold H_a from Definition 3.3 holds $H_a = h_a(R) = h_a(f(1))$, such that

$$\frac{\mathbb{E}[h(a_n)]}{H_a} = \frac{\frac{\beta}{\alpha+\beta} h_a(0) + \frac{\alpha}{\alpha+\beta} h_a(R)}{h_a(R)} = \kappa \ .$$

The fact that $\kappa < 1$ follows from Eq. (3.16). □

We now consider the case of K such flows $(A_i)_{1\leq i\leq K}$, each with capacity $C_1 > 0$ being multiplexed in a system with capacity $C := KC_1 > 0$. Instead of writing down the transition matrix for the resulting process, we simply can apply Theorem 3.6 and Corollaries 3.8–3.11 to obtain bounds on the aggregate and per-flow analysis, respectively:

Corollary 3.17. *Consider the multiplexed queueing system with total capacity $C = KC_1$, and let θ^* and κ such that*

$$\log \lambda(\theta^*) = \theta^* C_1 \ , \quad \textit{and} \quad \kappa := \frac{\left(\pi_0 v_0 + \pi_1 v_1\right)^K}{v_0^{K-\lceil CR^{-1}\rceil} v_1^{\lceil CR^{-1}\rceil}} \ .$$

Then in the multiplexed queueing system with total capacity $C = KC_1$, it holds for the aggregate flow:

$$\mathbb{P}\left(Q \geq \sigma\right) \leq \kappa e^{-\theta^*\sigma} \ , \ \textit{and} \quad \mathbb{P}\left(W \geq k\right) \leq \kappa e^{-\theta^* C k} \ ,$$

and for a single flow comprising $K_1 < K$ subflows under scheduling:

$$\begin{aligned}
\text{FIFO:} \quad & \mathbb{P}\left(W \geq k\right) \leq \kappa e^{-\theta^* C k} \\
\text{SP:} \quad & \mathbb{P}\left(W \geq k\right) \leq \kappa e^{-\theta^* K_1 C_1 k} \\
\text{EDF1:} \quad & \mathbb{P}\left(W \geq k\right) \leq \kappa e^{-\theta^*\left(Ck - (K-K_1)C_1 \min\{k,y\}\right)} \\
\text{EDF2:} \quad & \mathbb{P}\left(W \geq k\right) \leq \kappa e^{-\theta^*\left(Ck + (K-K_1)C_1 y\right)} + \tilde{\kappa} e^{-\tilde{\theta} N C_1 k} \ ,
\end{aligned}$$

where $y := d - d'$, and EDF1 and EDF2 correspond to $y \geq 0$ and $y < 0$, respectively. For EDF2, $\tilde{\kappa}$ and $\tilde{\theta}$ denote the corresponding parameters in the queueing system which has the total capacity $C = KC_1$ but only the K_1 subflows as arrivals.

Proof. At least $\lceil CR^{-1}\rceil$ chains have to be in the "On"-state if the aggregate instantaneous arrival is larger than the capacity. Thus, by the monotonicity

property from Eq. (3.16), for the threshold from Definition 3.3 holds:

$$H_a = v_0^{K-\lceil CR^{-1} \rceil} v_1^{\lceil CR^{-1} \rceil} \ .$$

Now simply apply Theorem 3.6 and Lemmas 3.8 – 3.11 to the arrival-martingale of Lemma 3.15. □

It can be shown that the leading constant κ is exponential in K (see [27]) and thus the fundamental property of an exponential decay in the number of flows (see Eq. (1.3)) is captured. As a side remark, the corresponding leading constant from [35, p. 340], is greater than one.

We point out that while the bounds in Corollary 3.17 for the *aggregate flow* have already been obtained in [27], the *per-flow* bounds (i.e., for SP, FIFO, and EDF) represent the contribution of this chapter.

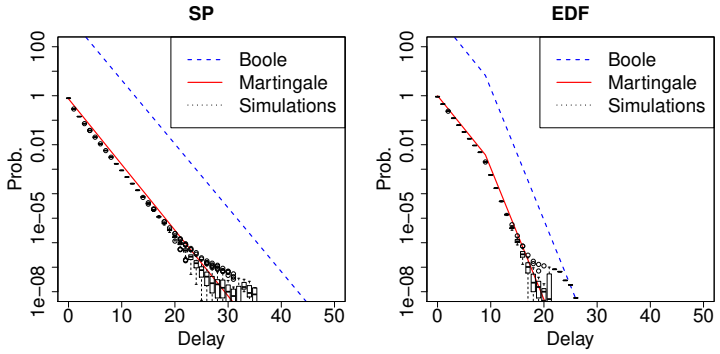
In Figure 3.4 simulations of the MMOO and the corresponding bounds for SP and EDF are displayed for different link utilizations. As in the case of independent increments, the Martingale bounds (from Corollary 3.17) are reasonably tight even at high utilizations (i.e., $\rho = 0.95$), whereas the bounds calculated with Boole's inequality (see Eq. (2.11)) are off by several orders of magnitude.

3.2.3 Autoregressive Arrival Models

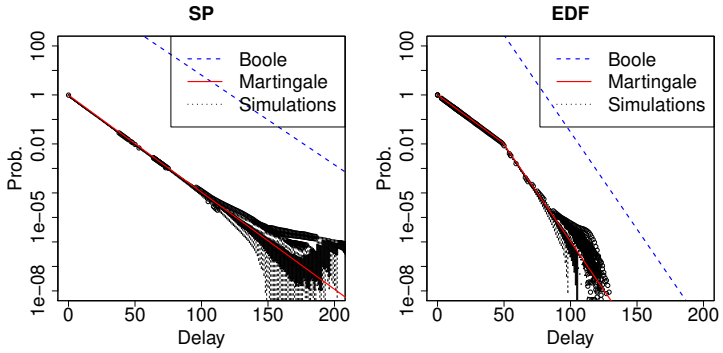
As a third example we consider autoregressive processes. Roughly, a p-order autoregressive process ($AR(p)$) evolves by rescaling the p previous values of the process and adding *Gaussian white noise*, i.e., uncorrelated Gaussian random variables.

We start with the formal definition of $AR(p)$. We assume throughout that the white noise is not only uncorrelated but independent.

Definition 3.18. *Let* $p \geq 1$, $Z_0, Z_1, Z_2, \cdots \sim \mathcal{N}_{0,1}$ *i.i.d.,* $\varphi_1, \ldots, \varphi_p \in [0, 1)$,



(a) Utilization $\rho = 0.75$, and $d - d' = 10 - 1 = 9$.



(b) Utilization $\rho = 0.95$, and $d - d' = 50 - 1 = 49$.

Figure 3.4: CCDF of the virtual delay (MMOO-case): $N_1 = \frac{1}{2}N = 10$, $\alpha = 0.1$, $\beta = 0.5$, and $R = 1$.

$\varphi = \sum_{k=1}^{p} \varphi_k$, *and* $\mu, \sigma > 0$. *If the relation*

$$a_n = \sum_{i=1}^{p} \varphi_i a_{n-i} + (1 - \varphi)\,\mu + (1 - \varphi)\,\sigma Z_n \tag{3.17}$$

holds, the process $(a_n)_n$ *is called the p-order autoregressive process,* $AR(p)$.

It can be shown (see, e.g., [26], p. 85) that if all the (complex) roots of the *characteristic polynomial*

$$\chi(z) = 1 - \sum_{i=1}^{p} \varphi_i z^i$$

lie outside the unit interval, i.e., $\chi(z) = 0 \Rightarrow |z| > 1$, then the process $AR(p)$ is stationary. We assume throughout that this condition is fulfilled. As above,

we apply Kolmogorov's theorem to obtain an extended process $(a_n)_{n\in\mathbb{Z}}$ which is still stationary and satisfying Eq. (3.17) (see Remark 2.3). Moreover, as $AR(p)$ is clearly a Gaussian process itself, it is also reversible (see [149, Theorem 1]), i.e., $A^r = A$.

Note that although $\mathbb{E}[a_n] = \mu$ for all $n \in \mathbb{Z}$, by the correlation of $AR(p)$ the variance $\mathbb{V}[a_n]$ is not equal to σ, but must be derived using the Yule-Walker-Equations (see again [26], p. 239).

As the instantaneous increment of an $AR(p)$ process depends on the p previous values and not only on the last one (as for the Markovian arrivals from Subsection 3.2.2) we need to slightly modify the definition of an arrival-martingale. The following notations are useful:

Notation 3.19. *Denote by $\overrightarrow{a_n}$ the p-dimensional vector*

$$\overrightarrow{a_n} := (a_n, a_n + a_{n-1}, \ldots, a_n + \cdots + a_{n-p+1}) = \left(\sum_{k=1}^{i} a_{n-k+1}\right)_{1\le i\le p} . \quad (3.18)$$

Further, for functions $h_1, \ldots, h_p$ let Πh denote the product function

$$\Pi h(x_1, \ldots, x_p) := \prod_{i=1}^{p} h_i(x_i) .$$

For brevity, we omit the parameter p in Notation 3.19, because its value is clear from the context.

Definition 3.20. *For $AR(p)$ arrival processes, in the definition of the arrival-martingale (Definition 3.1), Eq. (3.1) is replaced by:*

$$\Pi h\,(\overrightarrow{a_n})\, e^{\theta(A(n)-nC)} \ , \ n \ge 0 . \quad (3.19)$$

Note that for $p = 1$ the definition coincides with Definition 3.1. The reason for the unusual representation of the p previous values of $\overrightarrow{a_n}$ in Eq. (3.18) lies in the following fact:

Lemma 3.21. *For $\sigma > 0$, let $N := \inf\{n \ge 0 \mid A(n) - Cn \ge \sigma\}$ denote the*

stopping time in Eq. (3.2) from the proof of Theorem 3.4. Then for any $i \geq 1$,

$$\sum_{k=1}^{i} a_{N-k+1} > iC \ .$$

Proof. Assume that $\sum_{k=1}^{i} a_{N-k+1} \leq kC$ for some $i \geq 1$. Then

$$A_{N-i} - (N-i)\, C = (A_N - NC) - \left(\sum_{k=1}^{i} a_{N-k+1} \right) + iC > \sigma \ ,$$

contradicting the minimal property of N. □

The lemma generalizes the fact used in the proof of Theorem 3.4 that the last increment $a_N - C$ must be positive. In the next theorem, arrival-martingales for $AR(p)$ are constructed:

Lemma 3.22. *In the situation above, the autoregressive arrival process $A(m,n) = \sum_{k=m+1}^{n} a_k$ admits arrival-martingales.*

Proof. Let $\theta > 0$, $K_a := \mu + \frac{\sigma^2 \theta}{2}$ and define the functions $h_1, \ldots, h_p$ by

$$h_i(t) := e^{\theta \frac{\varphi_i}{1-\varphi} t} \ ,$$

i.e.,

$$\Pi h(\overrightarrow{a_n}) = e^{\frac{\theta}{1-\varphi} \sum_{i=1}^{p} \varphi_i \sum_{k=1}^{i} a_{n-k+1}} \ .$$

For $n \geq 0$, let $M_n := \Pi h(\overrightarrow{a_n}) e^{\theta(A_n - nK_a)}$. We show that M_n is a martingale.

Note that

$$\begin{aligned}
&\mathbb{E}\left[\Pi h(\overrightarrow{a_n}) e^{\theta(a_n - K_a)} \;\middle|\; Z_1, \ldots, Z_n\right] \\
&= \mathbb{E}\left[e^{\theta\left(\frac{1}{1-\varphi}\left(\sum_{i=1}^{p}\varphi_i \sum_{k=1}^{i} a_{n-k+1}\right) + a_n - K_a\right)} \;\middle|\; Z_1, \ldots, Z_n\right] \\
&= \mathbb{E}\left[e^{\theta\left(\frac{1}{1-\varphi}\left(\sum_{i=1}^{p}\varphi_i \sum_{k=2}^{i} a_{n-k+1}\right) + \frac{\varphi}{1-\varphi} a_n + a_n - K_a\right)} \;\middle|\; Z_1, \ldots, Z_n\right] \\
&= \mathbb{E}\left[e^{\theta\left(\frac{1}{1-\varphi}\left(\sum_{i=1}^{p}\varphi_i \sum_{k=1}^{i-1} a_{n-k}\right) + \sum_{i=1}^{p}\frac{\varphi_i}{1-\varphi} a_{n-i} + \mu + \sigma Z_n - K_a\right)} \;\middle|\; Z_1, \ldots, Z_n\right] \\
&= \mathbb{E}\left[e^{\theta\left(\frac{1}{1-\varphi}\left(\sum_{i=1}^{p}\varphi_i \sum_{k=1}^{i} a_{n-k}\right)\right)} e^{(\mu + \sigma Z_n - K_a)} \;\middle|\; Z_1, \ldots, Z_n\right] \\
&= \Pi h(\overrightarrow{a_{n-1}}) \mathbb{E}[e^{\theta(\sigma Z_n)}] e^{-\theta^2\sigma^2/2} \\
&= \Pi h(\overrightarrow{a_{n-1}}) \ .
\end{aligned}$$

Multiplying both sides by $e^{\theta(A_{n-1} - (n-1)K_a)}$ yields

$$\mathbb{E}\left[M_n \mid Z_1, \ldots, Z_n\right] = M_{n-1}$$

and the proof is complete. □

Note that for $p = 0$ we recover the case of independent increments as in Subsection 3.2.1. Let now

$$Y := \sum_{k=1}^{p} \varphi_k \sum_{i=1}^{k} a_{n-i+1} \ ,$$

Y is normally distributed with $\mathbb{E}[Y] = \mu \sum_{k=1}^{p} k\varphi_k$ (by stationarity, the distribution of Y is independent of n). Let $\nu^2 := \mathbb{V}[Y]$ denote its variance, which again can be calculated using the Yule-Walker-Equations.

Considering the single flow scenario from Figure 2.1(a) and Theorem 3.4, the following bounds hold:

Corollary 3.23. *For the autoregressive arrival model AR(p) with a capacity C satisfying $C > \mu$, let*

$$\theta^* = 2\frac{C - \mu}{\sigma^2} \ , \ \textit{and} \quad \kappa = e^{\frac{\theta^*(\mu - C)}{1-\varphi}\left(\sum_{i=1}^{p} i\varphi_i - \frac{\nu^2}{(1-\varphi)\sigma^2}\right)} \ .$$

Then for the backlog Q and virtual delay W hold

$$\mathbb{P}(Q \geq \sigma) \leq \kappa e^{-\theta^* \sigma} \text{ , and } \quad \mathbb{P}(W \geq k) \leq \kappa e^{-\theta^* C k} \ .$$

Proof. The only difference to the proof of Theorem 3.4 concerns the leading constant. Note that, by the monotonicity of the functions h_i, with Lemma 3.21 one obtains:

$$\Pi h(\overrightarrow{a_N}) \geq \Pi h(C, \ldots, pC) = e^{\frac{\theta}{1-\varphi} C \sum_{i=1}^{p} i\varphi_i} \ ,$$

Hence, for the leading constant holds:

$$\begin{aligned} \frac{\mathbb{E}[h(\overrightarrow{a_n})]}{\mathbb{E}[h(\overrightarrow{a_N})]} &\leq \frac{\mathbb{E}[e^{\frac{\theta^*}{1-\varphi}\left(\mu \sum_{i=1}^{p} i\varphi_i + \nu Z_0\right)}]}{e^{\frac{\theta^*}{1-\varphi}\left(\sum_{i=1}^{p} \varphi_i \sum_{k=1}^{i} a_{n-k+1}\right)}} \\ &= \frac{e^{\frac{\theta^*}{1-\varphi}\left(\mu \sum_{i=1}^{p} i\varphi_i + \frac{\theta^* \nu^2}{(1-\varphi)2}\right)}}{e^{\frac{\theta^*}{1-\varphi} \sum_{i=1}^{p} \varphi_i ic}} \\ &= e^{\frac{\theta^*(\mu - C)}{1-\varphi}\left(\sum_{i=1}^{p} i\varphi_i - \frac{\nu^2}{(1-\varphi)\sigma^2}\right)} = \kappa \ . \end{aligned}$$

The rest is exactly the same as in the proof of Theorem 3.4. □

Let us consider the special case of $p = 1$, i.e.,:

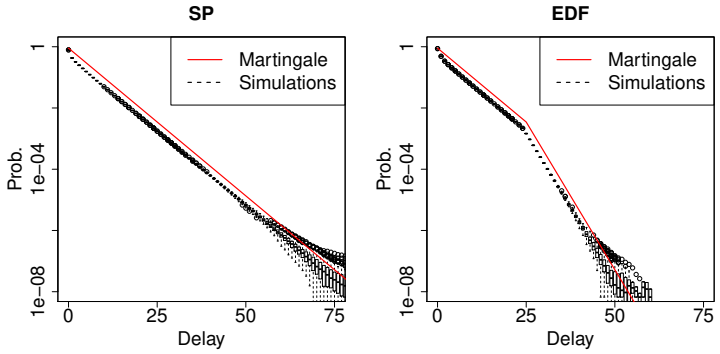
$$a_n = \varphi a_{n-1} + (1-\varphi)\mu + (1-\varphi)\sigma Z_n \ .$$

This special case allows an explicit calculation of the variance ν^2:

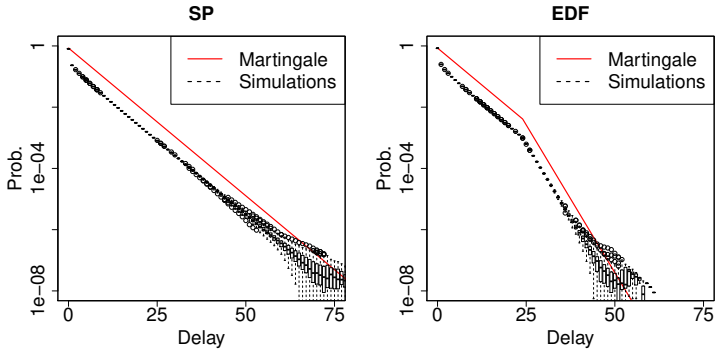
$$\nu^2 = \mathbb{V}[\varphi a_n] = \mathbb{V}[\varphi a_{n+1}] = \varphi^2 \mathbb{V}[\varphi a_n + \sigma\,(1-\varphi)\,Z_{n+1}] = \varphi^2 \left(\nu^2 + \sigma^2\,(1-\varphi)^2\right) \ ,$$

and thus $\nu^2 = \sigma^2 \frac{(1-\varphi)\varphi^2}{1+\varphi}$. The leading constant κ from Corollary 3.23 reduces to

$$\kappa = \frac{\mathbb{E}[h(a_n)]}{h(C)} = e^{\frac{\theta^*(\mu - C)}{1-\varphi}\left(\varphi - \frac{\nu^2}{(1-\varphi)\sigma^2}\right)} = e^{\frac{\theta^*(\mu - C)}{1-\varphi}\left(\varphi - \frac{\varphi^2}{1+\varphi}\right)} = e^{\frac{\theta^* \varphi(\mu - C)}{1-\varphi^2}} \ . \quad (3.20)$$



(a) $AR(1)$, $\varphi_1 = 0.6$



(b) $AR(2)$, $\varphi_1 = 0.4$, $\varphi_2 = 0.2$

Figure 3.5: CCDF of the virtual delay (autoregressive-case): $AR(1)$ ((a)) and $AR(2)$ ((b)), with parameters $\mu = 0.5$, $\sigma = 1$, utilization $\rho = 0.75$, and, for EDF, $y = d - d' = 24$.

Therefore, with regards to the queue size Q, the following bound holds:

$$\mathbb{P}(Q > \sigma) \leq e^{\frac{\theta^* \varphi (\mu - C)}{1-\varphi^2}} e^{-\theta^* \sigma} .$$

Note that, as $\mu - C < 0$, in this case $\kappa \in (0, 1]$. This bound improves the known results drastically: e.g., in [35], p. 340, an additional factor occurs, which depends on an upper bound on the increment process. As the *Gaussian white noise* is unbounded, the corresponding bound from [35] is trivial.

Now consider the aggregate scenario as in Figure 3.1: We assume that two homogeneous and independent autoregressive arrival flows are multiplexed.

Corollary 3.24 (Bounds for $AR(p)$ Arrivals)**.** *With the definitions as in*

Corollary 3.23 for the multiplexed queueing system with aggregate capacity $2C$ *holds:*

$$\mathbb{P}(Q \geq \sigma) \leq \kappa^2 e^{-\theta^* \sigma} \ , \ \textit{and} \quad \mathbb{P}(W \geq k) \leq \kappa^2 e^{-\theta^* 2Ck} \ ,$$

and for a single flow under scheduling:

$$\begin{aligned}
\text{FIFO:} \qquad & \mathbb{P}(W \geq k) \leq \kappa^2 e^{-\theta^* 2Ck} \\
\text{SP:} \qquad & \mathbb{P}(W \geq k) \leq \kappa^2 e^{-\theta^* Ck} \\
\text{EDF1:} \qquad & \mathbb{P}(W \geq k) \leq \kappa^2 e^{-\theta^* (2Ck - C\min\{k,y\})} \\
\text{EDF2:} \qquad & \mathbb{P}(W \geq k) \leq \kappa^2 e^{-\theta^* (2Ck + Cy)} + \tilde{\kappa} e^{-\tilde{\theta} 2Ck} \ .
\end{aligned}$$

Again, $y := d - d'$*, and EDF1 and EDF2 correspond to* $y \geq 0$ *and* $y < 0$*, respectively;* $\tilde{\kappa}$ *and* $\tilde{\theta}$ *denote the constants* κ *and* θ *with* C *exchanged by* $2C$*.*

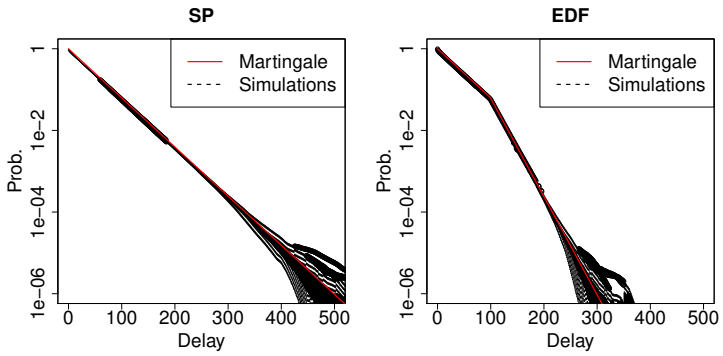
Proof. By definition of h_i in Lemma 3.22:

$$h_i \otimes h_i(t) = h_i(t)^2 \ . \tag{3.21}$$

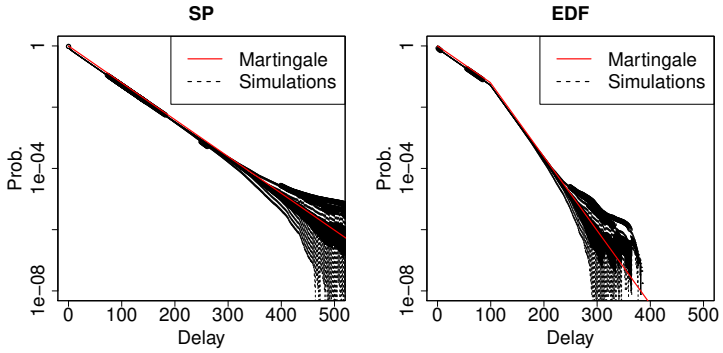
The results corresponding to Theorem 3.6 and Corollaries 3.8 – 3.11 with the modified arrival-martingale from Definition 3.20 are proved analogously. □

Note that, as the sum of independent autoregressive processes is still autoregressive, the aggregate bounds in the first part of Corollary 3.24 could also be obtained by applying Corollary 3.23 to the *single* flow $A_n + A'_n$. As the corresponding κ is independent of the number of flows, applying Eq. (3.21) iteratively leads to bounds retaining the fundamental exponential decay property from Eq. (1.3).

In Figures 3.5 and 3.6, simulations of the $AR(p)$ and the corresponding bounds for SP and EDF are displayed for different link utilizations. Unlike in the two previous arrival models, Boole's inequality could not be evoked to obtain bounds, since the sum on the right hand side in Eq. (2.11) seems not to converge.



(a) $AR(1)$, $\varphi_1 = 0.6$



(b) $AR(2)$, $\varphi_1 = 0.4$, $\varphi_2 = 0.2$

Figure 3.6: CCDF of the virtual delay (autoregressive-case): $AR(1)$ ((a)) and $AR(2)$ ((b)), with parameters $\mu = 0.5$, $\sigma = 1$, utilization $\rho = 0.95$, and, for EDF, $y = d - d' = 99$.

3.3 Summary

In this chapter we have proposed a novel characterization of arrival models by a certain supermartingale ("arrival-martingale") and developed a related *unified calculus* dealing with flows' multiplexing and scheduling. The crucial result of this calculus is that the scheduling operation translates into a time shifting operation of the underlying martingale-envelopes, enabling thus the derivation of tight per-flow performance bounds by leveraging a variant of Doob's inequality. We applied this calculus to Markovian and p-order autoregressive arrival flows and derived bounds on the per-flow delay distributions for several scheduling policies (FIFO, SP, and EDF). In certain burstiness scenarios, the obtained per-flow bounds capture for the first-time a fundamental exponential decay factor in

the number of flows. Moreover, the bounds almost match simulations, improving over classic results (e.g., FIFO: [49, 35], SP: [16, 155], EDF: [131]) by arbitrary orders of magnitude, especially at hight utilizations.

In the next chapter, we complement the derived arrival-martingale calculus with a parallel *service-martingale* calculus allowing for the analysis of more advanced service models like the Aloha or CSMA/CA protocol.

4

Service Martingales

In this chapter we extend the martingale methodology from Chapter 3 in order to fit the delay analysis of Aloha and CSMA/CA networks with Markovian arrivals. The novel element of the proposed extension is the concept of a *service-martingale* which models the Markovian service, characteristic to a multiaccess channel such as CSMA/CA, in the martingale domain. By combining service-martingales with the arrival-martingales defined in Chapter 3, we obtain sharp stochastic bounds on the backlog and delay distributions of a Markovian source over Aloha and CSMA/CA multiaccess channels.

A key benefit of our proposed methodology integrating arrival- and service-martingales is its *modularity*: Indeed, we provide conceivably straightforward applications to both simple and complex MAC scenarios. The first

(simple) scenario is standard and involves the analysis of a tagged bursty source sharing a MAC channel. We then consider two complex extensions by additionally accounting for 1) in-source scheduling, i.e., the tagged source consists of multiple flows scheduled according to a SP (Static Priority) policy before being transmitted over the shared channel, and 2) spatial multiplexing MIMO (multiple-input multiple-output), i.e., the tagged source is transmitted over multiple shared MAC channels. A qualitative insight of the obtained stochastic bounds is that MIMO reduces the delays of bursty sources exponentially (in the number of channels), and, more interestingly, that it is subject to a fundamental power-of-two phenomena.

The rest of this chapter is organized as follows. After discussing related work (Section 4.1), we introduce the concept of service-martingales in Section 4.2, and derive general performance metrics (backlog and delay) for a source modelled by arrival-martingales. In Section 4.3 we apply these results to a Markovian tagged source transmitting over Aloha and CSMA/CA channels; numerical results illustrate the remarkable tightness of the obtained stochastic bounds. In Section 4.4 we provide further applications to scenarios with in-source SP scheduling and spatial multiplexing MIMO.

4.1 Related Work

Classical works concerned with the throughput and delay analysis of random access protocols (e.g., Aloha or CSMA) rely on strong assumptions. One is that the point process comprising of both newly generated and retransmitted (due to collisions) packets is a Poisson process (Abramson [3], Kleinrock and Tobagi [94], and more recently Yang and Yum [159]). A related assumption is that, at each source, packets arrive as a *blocked* Poisson process, in the sense that *at most* one packet can be backlogged at any source (Tobagi [141] or Beuerman and Coyle [17]); this model is related to the infinite source model in which each source generates a single packet during its lifetime (Lam [97]). Another related and simplifying assumption is to discard the buffered packets at the beginning

of a transmission period for a source (Takagi and Kleinrock [135]).

Such conceivably unnatural assumptions enable a tractable analysis but preclude the analysis of realistic bursty sources, i.e., non-Poisson. In particular, the obtained results only capture the *access delay*, and not the other component of the actual delay, i.e., the *queueing delay*. For an elaborate discussion on fundamental drawbacks of ignoring data burstiness in the context of the multiaccess channel, in connection to information theory, see Gallager [64] and Ephremides and Hajek [57].

More recent literature addresses the throughput or delay analysis of the prevalent 802.11 CSMA/CA protocol. Some influential works include Bianchi [18], Cali *et al.* [33], Carvalho and Garcia-Luna-Aceves [34], which share the common assumption of saturated sources (i.e., ignoring burstiness). An approximate queueing analysis accounting for random arrivals is undertaken in Tickoo and Sikdar [140], by approximating the probability of non-empty queues as if the system behaved as an M/M/1 queue. A related approximation of the probability that a source finds itself empty upon a successful transmission is considered by Garetto and Chiasserini [66]. Another work addressing non-saturated arrivals is Alizadeh-Shabdiz and Subramaniam [4]; in addition to enforcing a technical independence assumption from [18], the analysis crucially relies on an M/G/1 approximation of the network, i.e., the arrival process is again assumed to follow a Poisson process.

While such existing results clearly provide valuable insights into the behavior of the notoriously difficult CSMA/CA protocol, the state-of-the-art literature lacks a mathematically rigorous (and also accurate) analysis under random arrivals, especially non-Poisson/bursty. The goal of this chapter is to fill this gap by providing the first rigorous and accurate delay analysis in single-hop Aloha and CSMA/CA networks, subject to Markovian arrivals. A crucial feature of the proposed analysis is that it rigorously accounts for buffering and consequently it captures the total (i.e., access plus queueing) delay experienced by a tagged Markovian source.

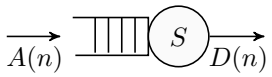


Figure 4.1: A server with an arrival process A, service process S, and departure process D

4.2 Theory

We assume the same situation as in Chapter 3 (as it was described in detail in Section 2.2): A flow $A(m,n) = \sum_{k=m+1}^{n} a_k$ arrives at a server characterized by a service process $S(m,n) = \sum_{k=m+1}^{n} s_k$ (see Figure 4.1). The service process S links A to its corresponding departure process by the inequality

$$D(n) \geq (A * S)(n) := \min_{0 \leq m \leq n} \{A(m) + S(m,n)\} \ , \tag{4.1}$$

(see Eq. (2.4)). The increment processes $(a_k)_k$ and $(s_k)_k$ are assumed to be stationary and independent of each other.

We give the central definition of this chapter concerning service modelling:

Definition 4.1 (Service-Martingales)**.** *The service process S admits service-martingales if for every $\theta > 0$ there is a $K_s \geq 0$ and a function $h_s : \mathrm{rng}(s) \to \mathbb{R}^+$ such that the process*

$$h_s(s_n) e^{\theta(nK_s - S^r(n))} \ , \ n \geq 0 \ , \tag{4.2}$$

is a supermartingale.

Again, the parameters K_s and h_s implicitly depend on θ; the augmented notation $K_s(\theta)$ and $h_s(\theta)$ is omitted for brevity, when clear from the context.

Arrival- and service-martingales relate to each other by a sign change of θ, and closely resemble with the concepts of effective bandwidth and capacity, respectively. The crucial difference is that while the effective bandwidth and capacity are defined in terms of the moment generating function (MGF) and Laplace

transform of $A(n)$ and $S(n)$, respectively, the arrival- and service-martingales are defined as stochastic processes and not as (deterministic) numbers, albeit in terms of similar exponential transforms.

The analogous result to Remark 3.2 also holds for service-martingales:

Remark 4.2. *If (4.2) is a supermartingale, then by stationarity the "time-shifted" process*

$$h_s(s_{n+k})e^{\theta(nK_s - S^r(k,n+k))}$$

is also a supermartingale, for some fixed $k \geq 0$.

Let us now state an auxiliary definition which extends Definition 3.3 by taking into account the service-martingale:

Definition 4.3 (Threshold)**.** *For h_a and h_s as in Definitions 3.1 and 4.2 define the threshold*

$$H_{as} := \min\{h_a(x)h_s(y) \mid x - y > 0\} \ .$$

Intuitively, H_{as} is the smallest value of $h_a(x)h_s(y)$ such that the instantaneous arrival (i.e., x) is larger than any value of the stochastic process driving the service process (i.e., y).

For the rest of this section we assume that the arrival flow A and the service process S admit arrival- and service-martingales, respectively. The corresponding parameters are denoted by K_a and h_a for the arrival-, and by K_s and h_s for the service-martingales. Recall that these parameters implicitly depend on the value of θ.

Again, the performance metrics of interest are the (stationary) backlog distribution as defined in Eq. (2.7):

$$Q =_{\mathcal{D}} \sup_{n \geq 0}\{A^r(n) - S^r(n)\} \ ,$$

and the virtual delay from Eq. (2.9) with

$$\mathbb{P}\left(W \geq k\right) \leq \left\{ \sup_{n \geq k} A^r(k,n) - S^r(n) \geq 0 \right\} \ .$$

Theorem 4.4 (Backlog)**.** *Assume that the statistically independent processes A and S admit arrival- and service-martingales, respectively. Let*

$$\theta^* := \sup\left\{\theta \geq 0 \mid K_a \leq K_s\right\} \ ,$$

and H_{as} as in Definition 4.3. Then the following backlog bound holds for any $\sigma \geq 0$

$$\mathbb{P}(Q \geq \sigma) \leq \frac{\mathbb{E}[h_a(a_0)]\mathbb{E}[h_s(s_0)]}{H_{as}} e^{-\theta^* \sigma} \ .$$

Proof. Let θ^* as defined, and the corresponding parameters K_a, h_a, K_s, and h_s (all depending on θ^*). By the independence assumption, the process

$$h_a(a_n) h_s(s_n) e^{\theta^*(A(n) - nK_a + nK_s - S(n))}$$

is a supermartingale (see Lemma 2.9). As by definition (of θ^*) $K_s - K_a \geq 0$,

$$M(n) := h_a(a_n) h_s(s_n) e^{\theta^*(A(n) - S(n))}$$

is a supermartingale as well. Now proceed similarly as in the proof of Theorem 3.4 by defining the stopping time N as the first time when $A(n) - S(n)$ exceeds σ, i.e.,

$$N := \min\left\{n \mid A(n) - S(n) \geq \sigma\right\} \ ,$$

again, $\mathbb{P}(Q \geq \sigma) = \mathbb{P}(N < \infty)$. By the optional stopping theorem (see Lemma 2.8)

applied to the stopping time $N \wedge n$ (for $n \geq 0$) we have

$$\begin{aligned}\mathbb{E}[h_a(a_0)]\mathbb{E}[h_s(s_0)] &= \mathbb{E}[M(0)] = \mathbb{E}[M(N \wedge n)] \\ &\geq \mathbb{E}[M(N \wedge n)1_{\{N \leq n\}}] \\ &= \mathbb{E}[h_a(a_N)h_s(s_N)e^{\theta^*(A(N)-S(N))}1_{\{N \leq n\}}] \\ &\geq H_{as}e^{\theta^*\sigma}\mathbb{P}(N \leq n) \ .\end{aligned}$$

For the last step note that by the minimality of N, $a_N > s_N$ and so with Definition 4.3: $h_a(a_N)h_s(s_N) \geq H_{as}$. The proof completes by letting $n \to \infty$. □

Theorem 4.5 (Delay)**.** *In the situation of Theorem 4.4, the following stochastic bound holds for the virtual delay*

$$\mathbb{P}(W \geq k) \leq \frac{\mathbb{E}[h_a(a_0)]\mathbb{E}[h_s(s_0)]}{H_{as}}e^{-\theta^* K_s k} \ .$$

Proof. Let θ^* as defined, and the corresponding parameters K_a, h_a, K_s, and h_s (again, all depending on θ^*). Given the representation for the virtual delay from Eq. (2.9), we can write:

$$\begin{aligned}\mathbb{P}\left(W \geq k\right) &\leq \mathbb{P}\left(\sup_{n \geq k}\{A^r(k,n) - S^r(n)\} \geq 0\right) \\ &\leq \mathbb{P}\left(\sup_{n \geq k}\{A^r(k,n) - (n-k)K_a + nK_s - S^r(n)\} \geq kK_s\right) \ .\end{aligned}$$

Using Remark 4.2 and the independence assumption, it follows that

$$h_a(a_n)h_s(s_n)e^{\theta(A^r(k,n)-(n-k)K_a+nK_s-S^r(n))}$$

is also a supermartingale (in the time-domain $\{k, k+1, \ldots\}$). Therefore, by invoking the same arguments as in the proof of Theorem 4.4, the above inequalities

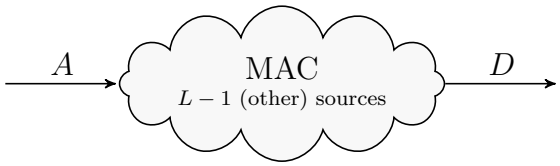


Figure 4.2: A tagged source L, comprising of arrival and departure processes A and D, respectively, competing on a MAC shared channel (Aloha or CSMA/CA) with $L-1$ other sources

continue to:

$$\begin{aligned}\mathbb{P}\left(W \geq k\right) &\leq \frac{\mathbb{E}[h_a(a_k)]\mathbb{E}[h_s(s_k)e^{\theta^*(kK_s - S(0,k))}]}{H_{as}} e^{-\theta^* K_s k} \\ &\leq \frac{\mathbb{E}[h_a(a_0)]\mathbb{E}[h_s(s_0)]}{H_{as}} e^{-\theta^* K_s k} \ ,\end{aligned}$$

where we lastly used the stationarity of $(a_n)_n$ and the property that the expectation of supermartingales is non-increasing. □

4.3 Applications: Aloha and CSMA/CA

In this section we apply the previous theoretical results to analyze the queueing performance of a bursty source, denoted by L, and transmitting over an Aloha or CSMA/CA shared channel together with $L-1$ other (saturated) sources (see Figure 4.2).

In both cases we consider a bursty source L being modelled by a Markov-Modulated On-Off (MMOO) process $(a_n)_n$ as in Section 3.2.2 (see Figure 4.3).

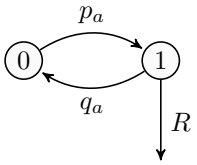


Figure 4.3: The arrival process for source L, modelled in terms of a Markov-Modulated On-Off (MMOO) process

Recall that the transition matrix of the Markov chain is given by

$$T_a = \begin{pmatrix} 1-p_a & p_a \\ q_a & 1-q_a \end{pmatrix} ,$$

for a's steady state distribution holds

$$\mathbf{\Pi_a} = \left(\frac{q_a}{p_a+q_a}, \frac{p_a}{p_a+q_a} \right) ,$$

and the cumulative arrival process can be represented as

$$A(n) = \sum_{k=1}^{n} f(a_k) , \tag{4.3}$$

where $f(0) = 0$, $f(1) = R$, and $R > 0$ is the peak rate transmitted while the source is in state "1" (i.e., the "On" state). Arrival martingales for the source A were constructed in Lemma 3.15.

In the following we consider the two cases when the source L shares an Aloha or CSMA/CA channel with $L-1$ other (saturated) sources denoted by $\{1, 2, \ldots, L-1\}$.

4.3.1 Aloha

With the (slotted) Aloha protocol, in each time slot a source transmits with a fixed probability $p_{tr} > 0$, independently from the other sources and also from previous transmissions. Thus, the probability of a successful transmission is given by

$$p_{suc} := p_{tr} \left(1 - p_{tr}\right)^{L-1} .$$

During the interval of a successful transmission the link provides an ideal capacity $C > 0$. In any other interval, due to a successful transmission of another source or a collision, no capacity is provided (for source L). The service process

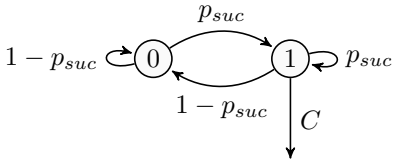


Figure 4.4: The service process for source L, modelled in terms of a process with independent increments corresponding to an Aloha link

for the source L is thus given by

$$S_{\text{aloha}}(m, n) := \sum_{k=m+1}^{n} s_k \ ,$$

were the the (instantaneous) service rates s_k are i.i.d. and are distributed according to:

$$s_k := \begin{cases} C & \mathbb{P} = p_{tr} \left(1 - p_{tr}\right)^{L-1} \\ 0 & \mathbb{P} = 1 - p_{tr} \left(1 - p_{tr}\right)^{L-1} \end{cases}$$

(see Figure 4.4 and also Ciucu *et al.* [41]).

Service-martingales for S_{aloha} can be obtained in a way similar to the i.i.d. arrival model of Lemma 3.12:

Lemma 4.6. *The service process S_{aloha} for the Aloha protocol admits service-martingales.*

Proof. As arrival- and service-martingales relate to each other by a sign change of θ, replace in Eq. (3.11) the moment generating function by the Laplacian, i.e., let

$$K_s := \log \mathbb{E}\left[e^{-\theta s_1}\right] / \left(-\theta\right) \ , \tag{4.4}$$

and proceed as in the proof of Lemma 3.12. □

We now state the main result for the Aloha model. Let $T_{a,\theta}$ denotes the exponentially transformed transition matrix of T_a as in Eq. (3.13).

Corollary 4.7 (Bounds for Aloha)**.** *Assume the stability condition $E[a_1] <$*

$E[s_1]$ and let

$$\theta^* := \sup\left\{\theta > 0 \;\middle|\; \lambda_a(\theta) = \mathcal{L}_s(\theta)^{-1}\right\} \ ,$$

where $\lambda_a(\theta)$ denotes the maximal positive eigenvalue of $T_{a,\theta}$, and

$$\mathcal{L}_s(\theta) := 1 - p_{tr}\left(1-p_{tr}\right)^{L-1} + p_{tr}\left(1-p_{tr}\right)^{L-1} e^{-\theta C}$$

is the Laplace transform of s_k. Let further h_a be a (positive) eigenvector of T_{a,θ^*} corresponding to $\lambda_a(\theta)$, and $h_s \equiv 1$. Then the following bounds hold for the backlog and delay of source L:

$$\mathbb{P}(Q \geq \sigma) \leq \frac{\mathbb{E}[h_a(a_0)]}{H_{as}} e^{-\theta^*\sigma} \ , \ \textit{and} \quad \mathbb{P}(W \geq k) \leq \frac{\mathbb{E}[h_a(a_0)]}{H_{as}} e^{-\theta^* K_s k} \ ,$$

where H_{as} is defined as in Definition 4.3.

Proof. Note first that θ^* is well-defined (i.e., the supremum is taken over a non-empty set) because

$$\frac{d}{d\theta}\lambda_a(\theta)\bigg|_{\theta=0} = \mathbb{E}[a_1] < \mathbb{E}[s_1] = \frac{d}{d\theta}\mathcal{L}_s(\theta)^{-1}\bigg|_{\theta=0} \ .$$

Note also that the more explicit definition of θ^* follows from Theorem 4.4, whereby the values K_a and K_s are from Eq. (3.14) and Eq. (4.4), respectively, i.e.,

$$\theta^* := \sup\left\{\theta > 0 \;\middle|\; \frac{\log \lambda_a(\theta)}{\theta} \leq \frac{\log E\left[e^{-\theta s_1}\right]}{-\theta}\right\} \ .$$

The replacement of the inequality by an equality is possible due to the continuity of the eigenvalues and the Laplace transform. The rest of the proof follows from Theorems 4.4 and 4.5 using the constructed arrival- and service-martingales, respectively. □

To illustrate the accuracy of the obtained delay bounds, we quickly provide several numerical results in Figures 4.5 and 4.6, by varying both the utilization and also the number of sources. The bounds are shown as continuous lines and the simulation results are shown as box-plots.

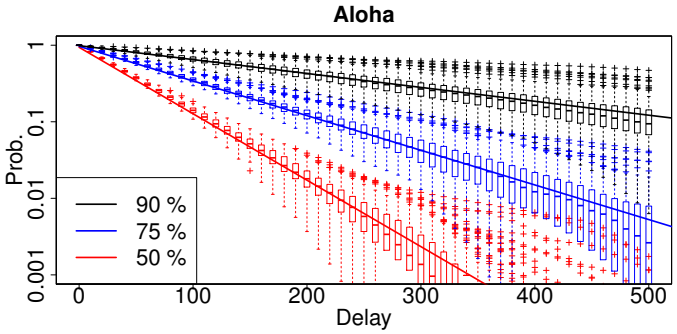


Figure 4.5: CCDF of the virtual delay of source L (Aloha-case): probabilities $p_a = 0.1$, $q_a = 0.5$, $p_{tr} = 0.2$, $L = 10$ sources, and utilizations $\rho = 0.5, 0.75, 0.9$ (bottom to top), respectively

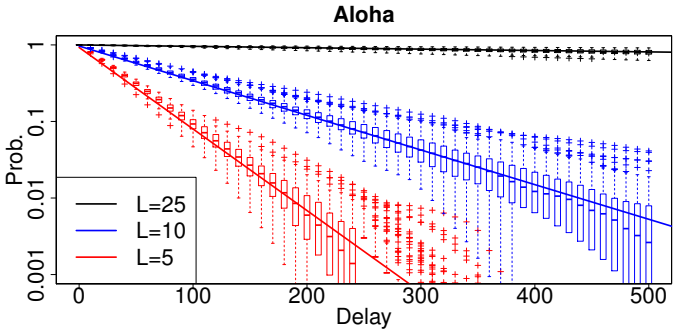


Figure 4.6: CCDF of the virtual delay of source L (Aloha-case): probabilities $p_a = 0.1$, $q_a = 0.5$, $p_{tr} = 0.2$, $\rho = 0.75$, and number of sources $L = 5, 10, 25$ (bottom to top), respectively

4.3.2 CSMA/CA

We adopt the CSMA/CA model from Durvy *et al.* [56] in terms of a Markov chain $(s_n)_n$, as depicted in Figure 4.7. Due to its tree structure, the Markov chain is reversible (see Kelly [88, Lemma 1.5]). The source L can transmit (subject to current buffer occupancy) at some peak rate $C > 0$ (i.e., ideal channel's capacity) while in state L, whereas all sources are in backoff mode while in state 0.

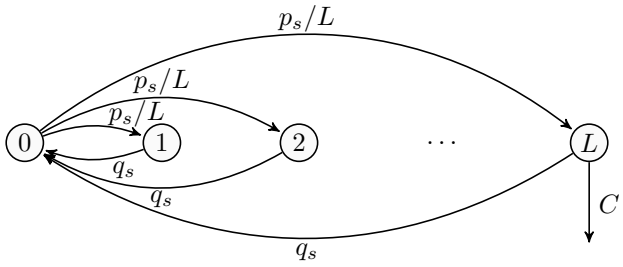


Figure 4.7: The service process for source L, modelled in terms of a Markov process corresponding to a CSMA/CA link

The transition matrix of the chain $(s_n)_n$ is given by

$$T_s = \begin{pmatrix} 1-p_s & \frac{p_s}{L} & \dots & \frac{p_s}{L} \\ q_s & 1-q_s & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ q_s & 0 & \dots & 1-q_s \end{pmatrix} ,$$

whereas the steady-state distribution of s is given by

$$\mathbf{\Pi_s} = \left(\frac{q_s}{p_s+q_s}, \frac{p_s}{L(p_s+q_s)}, \dots, \frac{p_s}{L(p_s+q_s)} \right) .$$

Using the methodology from [41], the service process $S_{\text{csma}}(m,n)$ of link L can be represented by

$$S_{\text{csma}}(m,n) := \sum_{k=m+1}^{n} C 1_{\{s_k=L\}} = \sum_{k=m+1}^{n} f(s_k) , \tag{4.5}$$

where $f(L) := C$, and $f(i) := 0$ for $i < L$. Finally, let $T_{s,\theta}$ denote the exponentially transformed transition matrix as in Definition 3.13.

Service-martingales for S_{csma} can be obtained in a way similar to the Markovian case of Subsection 3.2.2:

Lemma 4.8. *The service process S_{csma} for the CSMA/CA protocol admits service-martingales.*

Proof. For $\theta > 0$, let $\lambda_s(-\theta)$ denote the maximal positive eigenvalue of $T_{s,-\theta}$,

h_s a corresponding eigenvector, and

$$K_s := \frac{\log \lambda_s(-\theta)}{-\theta}$$

The rest is as in the proof of Lemma 3.15. □

We now state the main result for the CSMA/CA scenario.

Corollary 4.9 (Bounds for CSMA/CA)**.** *Assume the stability condition* $\mathbb{E}[a_1] < \mathbb{E}[s_1]$ *and let*

$$\theta^* := \sup\left\{\theta > 0 \;\middle|\; \frac{\log \lambda_a(\theta)}{\theta} = \frac{\log \lambda_s(-\theta)}{-\theta}\right\} .$$

Let also h_a *and* h_s *be corresponding (positive) eigenvectors of* T_{a,θ^*} *and* T_{s,θ^*}*, respectively. Then the following bounds hold for the backlog and delay of source L:*

$$\begin{aligned}\mathbb{P}(Q \geq \sigma) &\leq \frac{\mathbb{E}[h_a(a_0)]\mathbb{E}[h_s(s_0)]}{H_{as}} e^{-\theta^*\sigma} \\ \mathbb{P}(W \geq k) &\leq \frac{\mathbb{E}[h_a(a_0)]\mathbb{E}[h_s(s_0)]}{H_{as}} e^{-\theta^* K_s k} ,\end{aligned}$$

where H is defined as in Definition 4.3.

Proof. Note that θ^* is well-defined (i.e., the supremum is taken over a non-empty set) because

$$\frac{d}{d\theta}\lambda_a(\theta)\bigg|_{\theta=0} = \mathbb{E}[a_1] < \mathbb{E}[s_1] = \frac{d}{d\theta}\left(\lambda_s(-\theta)\right))^{-1}\bigg|_{\theta=0} .$$

For the rest of the proof simply apply Theorems 4.4 and 4.5 to the constructed arrival- and service-martingales. The replacement of the inequality by an equality is due to the same argument as in the proof of Corollary 4.7. □

As for Aloha, we quickly provide several numerical results in Figures 4.8 and 4.9; the figures confirm that the stochastic delay bounds are very accurate

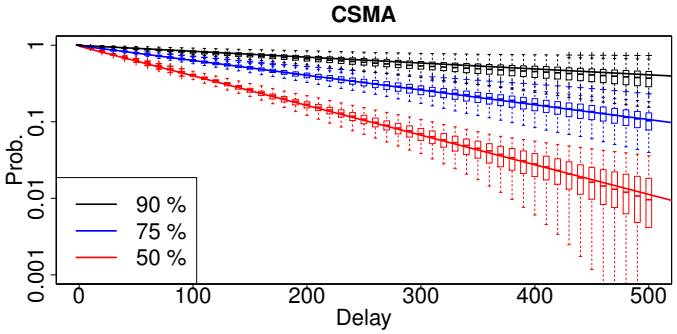


Figure 4.8: CCDF of the virtual delay of source L (CSMA/CA-case): probabilities $p_a = 0.1$, $q_a = 0.5$, $p_s = 0.8$, $q_s = 0.2$, $L = 10$ sources, and utilizations $\rho = 0.5, 0.75, 0.9$ (bottom to top), respectively

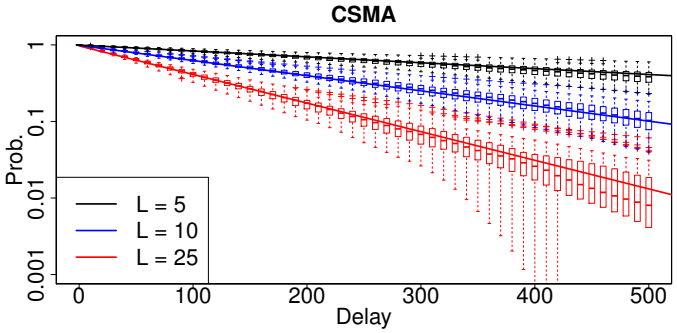


Figure 4.9: CCDF of the virtual delay of source L (CSMA/CA-case): probabilities $p_a = 0.1$, $q_a = 0.5$, $p_s = 0.8$, $q_s = 0.2$, utilization $\rho = 0.75$, and number of sources $L = 5, 10, 25$ (bottom to top) flows, respectively

for a broad range of scenarios (note that at large values of the tail delay, the box plots widen due to the availability of fewer data points in the simulations).

Finally, we note that for both Aloha and CSMA/CA, the arrival and service processes are independent. That is due to the fact that the construction of the service process is oblivious to the arrival process, and in particular it holds for saturated arrivals; such constructions are conservative since the network nodes do not rely on backlog state information from neighborhood nodes, and thus the channel may be underutilized.

4.4 Further Applications: Scheduling and MIMO

In this section we present more complex applications of the general results from Section 4.2. Concretely, we extend the CSMA/CA scenario from Subsection 4.3.2 in two directions: 1) accounting for in-source scheduling (Subsection 4.4.1), and 2) accounting for spatial multiplexing MIMO (Subsection 4.4.2).

4.4.1 In-Source Scheduling

We generalize the basic scenario from Section 4.3.2 by assuming that the tagged source L comprises multiple flows, whose transmissions are first scheduled before being sent over the CSMA/CA channel. Without loss of generality we assume only two flows, whose arrivals and departures are denoted by A and D, and A' and D', respectively, and a Static Priority (SP) scheduling policy (see Figure 4.10).

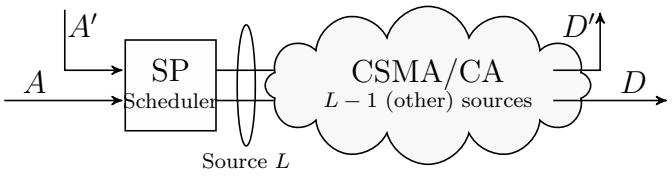


Figure 4.10: A tagged source L, comprising of two arrival flows A and A', which are scheduled according to an SP policy before being transmitted over the channel

The arrival processes A and A' of the source L are statistically independent, and are assumed to have the same parameters as in Section 4.3 for the arrival-martingales, i.e., $K_a = K'_a$ and $h_a(\cdot) = h'_a(\cdot)$; let also $T_{a,\theta}$ be the corresponding exponential column-transform (of a single flow).

In this scheduled system, we are interested in the performance metrics (i.e., backlog and delay) for the flow A. Because the service process $S_{\text{csma}}(m,n)$ from Eq. (4.5) is an *exact service process*, in the sense that Eq. (2.4) is in fact satisfied with equality (see Ciucu *et al.* [41]), it follows that the overall service

process available to the flow A is given by

$$S_A(m,n) = S_{\text{csma}}(m,n) - A'(m,n) \ .$$

This service process, known in the (stochastic) network calculus literature as the leftover service process (see also Chang [35] and Fidler [60]), can be thought of as a combination of the service processes for SP (Eq. (3.7)) and CSMA/CA (Eq. (4.5)).

Concerning the service process $S(n)$, recall that it admits service-martingales with parameters $h_s(\cdot)$ and K_s; let T_s^θ be the corresponding column-transform.

Corollary 4.10 (Bounds for SP + CSMA/CA)**.** *Assume the stability condition* $2E[a_1] < E[s_1]$ *and let*

$$\theta^* := \sup\left\{\theta > 0 \;\middle|\; \left(\lambda_a(\theta)\right)^2 = \lambda(\theta)^{-1}\right\} \ .$$

Let also h_a *and* h_s *be corresponding (positive) eigenvectors of* T_{a,θ^*} *and* T_{s,θ^*}*, respectively. Then the following bounds hold for the backlog and delay of the (sub-)arrival flow* A *of source* L*:*

$$\begin{aligned}\mathbb{P}(Q \geq \sigma) &\leq \frac{\mathbb{E}[h_a(a_0)]^2\mathbb{E}[h_s(s_0)]}{H_{as}} e^{-\theta^*\sigma} \\ \mathbb{P}(W \geq k) &\leq \frac{\mathbb{E}[h_a(a_0)]^2\mathbb{E}[h_s(s_0)]}{H_{as}} e^{-\theta^*(K_s - K'_a)k} \ ,\end{aligned}$$

where

$$H_{as} := \min\left\{h_a(x)h'_a(x')h_s(y) \mid x + x' - y > 0\right\} \ .$$

Proof. Note first that θ^* is well-defined using the same argument from Corollary 4.9. Next we slightly adapt Theorems 4.4 and 4.5 for the constructed arrival- and service-martingales. The key observation (in the case of the delay) is that by the independence assumption of A, A', and S, the product

$$h_a(a_n)h_a(a'_n)h_s(s_n)e^{\theta\left(A(k,n)-(n-k)K_a+A'(n)-nK'_a+nK_s-S(n)\right)}$$

is a supermartingale. Note also that $A'(k,n)$ is shifted with respect to both $A'(n)$ and $S(n)$, whence the term $K_s - K_a$ in the asymptotic decay rate of the delay. Finally, the definition of θ^* from Theorem 4.4 becomes

$$\theta^* := \sup\{\theta > 0 \mid 2K_a \leq K_s\} \ ,$$

which completes the proof. □

Corollary 4.10 generalizes the SP delay bounds from Corollary 3.8 for a constant-rate server; similar generalizations are immediate in the case of the other scheduling FIFO and EDF. Corollary 4.10 reveals the *modularity* feature of the proposed methodology, in the sense of jointly analyzing interconnected systems such as in-source scheduling and MAC protocols; a further convincing example is provided next.

4.4.2 MIMO

Here we generalize the basic scenario from Section 4.3.2 by considering a spatial multiplexing MIMO (multiple input multiple-output) scenario (see, e.g., Heath and Paulraj [74]), in which the source L is served by J CSMA/CA channels (see Figure 4.11). To keep the analysis tractable, we assume the independence of the channels and disregard fading effects.

The source L has the same arrival process as in Section 4.3, in particular with the parameters K_a and $h_a(\cdot)$ for the corresponding arrival-martingales. Furthermore, by extending the notations from Section 4.3.2, we assume that the service on each channel $j = 1, 2, \ldots, J$ is modulated by i.i.d. Markov processes $(s_{j,n})_n$ (with the same parameters as in Section 4.3.2). For the particular case of MIMO spatial multiplexing, the *overall* service process $S_j(m,n)$ of link L can be represented by

$$S(m,n) := \sum_{j=1}^{J} S_j(m,n) := \sum_{j=1}^{J} \sum_{k=m+1}^{n} C1_{\{s_{j,k}=L\}} \ , \tag{4.6}$$

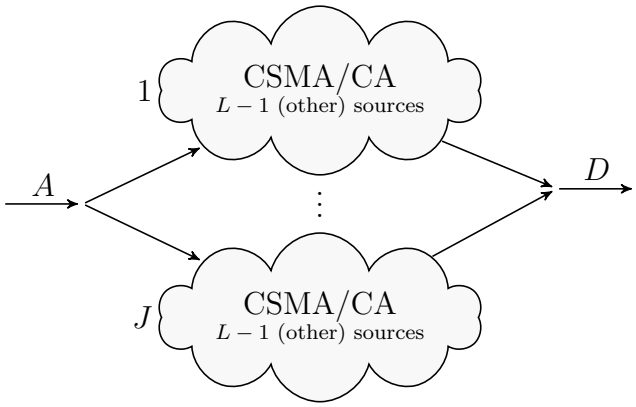


Figure 4.11: Spatial multiplexing MIMO: the tagged source L is transmitted over J independent MAC channels

where $S_j(m,n)$ is the service process for channel j.

Each service process $S_j(n)$ admits service-martingales with parameters $h_s(\cdot)$ and K_s (due to the i.i.d. assumption across the modulated Markov processes). Let also $T_{a,\theta}$ and $T_{s,\theta}$ be the corresponding exponential column-transforms for the arrival and service processes.

Corollary 4.11 (Bounds for MIMO)**.** *Assume the stability condition* $E[a_1] < JE[s_1]$ *and let*

$$\theta^* := \sup\left\{\theta > 0 \;\middle|\; \lambda_a(\theta) = (\lambda_s(\theta))^{-J}\right\} \ ,$$

where $sp(\cdot)$ *denotes the maximal positive eigenvalue. Let also* h_a *and* h_s *be corresponding (positive) eigenvectors of* T_{a,θ^*} *and* T_{s,θ^*}*, respectively. Then the following bounds hold for the backlog and delay of source* L*:*

$$\mathbb{P}(Q \geq \sigma) \leq \frac{\mathbb{E}[h_a(a_0)]\mathbb{E}[h_s(s_0)]^J}{H_J} e^{-\theta^*\sigma}$$
$$\mathbb{P}(W \geq k) \leq \frac{\mathbb{E}[h_a(a_0)]\mathbb{E}[h_s(s_0)]^J}{H_J} e^{-\theta^* K_s k} \ ,$$

where

$$H_J := \min\left\{ h_a(x)\prod_{j=1}^{J} h_s(y_j) \;\middle|\; x - \sum_{j=1}^{J} y_j > 0 \right\} \ .$$

Proof. As in Corollary 4.9, θ^* is well-defined. We make the key observation that

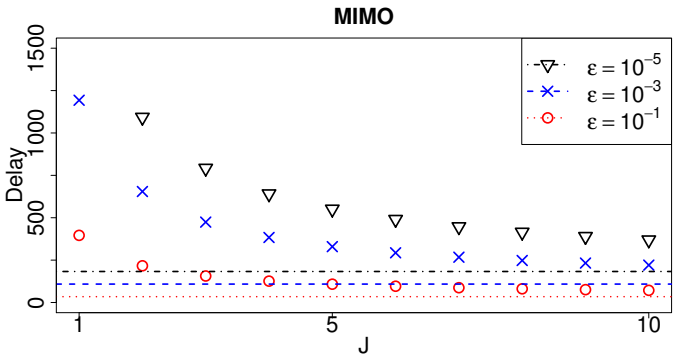


Figure 4.12: The tail delays from Corollary 4.11 as a function of the number of channels J ($p_a = 0.1$, $q_a = 0.5$, $p_s = 0.8$, $q_s = 0.2$, utilization $\rho = 0.75$, and $\varepsilon = 10^{-5}, 10^{-3}, 10^{-1}$); the bottom horizontal lines correspond to the tail delays under deterministic service (the corresponding bounds are computed with Theorem 4.5)

by the independence assumption on the Markov processes $(s_{j,n})_n$, the product

$$\prod_{j=1}^{J} h_j(s_{j,n}) e^{\theta(JK_s - S(n))}$$

is a service-martingale for the overall service process S. Consequently, the definition of θ^* from Theorem 4.4 becomes

$$\theta^* := \sup\{\theta > 0 \mid K_a \leq JK_s\} \ .$$

The rest proceeds as in Corollary 4.9. □

Let us now analyze the impact of the number of channels J, in particular on the probabilistic delay of source L. Due to the implicit definition of θ^* from Corollary 4.11 in terms of eigen-values/vectors, a quantitative result is conceivably difficult to be obtained. We thus resort to a numerical experiment, using the same numerical values as in Section 4.3.2. Concretely, in Figure 4.12, we illustrate the tail delay for three violation probabilities (i.e., $\varepsilon = 10^{-5}, 10^{-3}, 10^{-1}$) as a function of the number of channels J, and for a normalized utilization $\rho = 0.75$ (for each J). The key observation is the exponential decay of the delay, an effect which is more pronounced for smaller (and thus more practical) values of ε.

The figure also includes the corresponding delays in a scenario with deterministic (and normalized) service, for the three values of ε (i.e., the three horizontal bottom lines, which are invariant to J). As expected, for each ε, the tail delays converge to the horizontal line corresponding to a deterministic service; especially for small values of ε, the convergence is however very slow and not visible in the current plot. While we limit J to 10 for both practical considerations and the readability of the plot, we point out that for $\varepsilon = 10^{-5}$ the convergence is still not visible at $J = 100$, but only around $J = 1000$ (i.e., an impractical regime).

Overall, the figure convincingly indicates that MIMO spatial multiplexing manifests its power for small values of J only. Concretely, for realistic small values of ε, there is a dramatic decrease in delay when increasing the number of channels from $J = 1$ to $J = 2$. The delays continue to decrease by further increasing J, but at much smaller rates.

4.5 Summary

In this chapter we have developed the first rigorous and accurate methodology to compute queueing performance metrics (i.e., backlog and delay) for bursty sources sharing a MAC (bursty) channel: the sources are modelled using the arrival-martingale model from Chapter 3, whereas the available service for the source at the shared channel is modelled using the service-martingale model. By leveraging the modelling power of the proposed martingale methodology we have shown that the obtained stochastic bounds are remarkably tight in the case of Markov-modulated sources, and Aloha and CSMA/CA channels. We have also shown that our methodology offers an attractive modularity feature, in the sense that we could extend basic results to much more complex scenarios accounting for in-source SP scheduling or MIMO spatial multiplexing.

5

The Impact of Randomness in the Number of Flows

The common challenge faced by all queueing approaches, when modelling some unpredictable resource sharing based system, is capturing the system's inherent randomness. While capturing randomness is essential in modelling, different randomness models can lead to very different insights on actual system behavior. Consider for instance a simple example of a router with capacity C which is being modelled by the classic M/M/1 queue: packets arrive as a Poisson process with rate λ, and their sizes are exponentially distributed with average $1/\mu$. Under

the stability condition $\lambda/(\mu C) < 1$, the packets' average delay is

$$\mathbb{E}\Big[delay\Big] = \frac{1}{\mu C - \lambda} \ . \tag{5.1}$$

If we consider next the much simpler averaged-out D/D/1 model, in which the interarrival times are constant (i.e., equal to $1/\lambda$) and packet sizes are constant as well (i.e., equal to $1/\mu$). Under the same stability condition, the packets' average delay becomes

$$\mathbb{E}\Big[delay\Big] = \frac{1}{\mu C} \ . \tag{5.2}$$

Note the different quantitative results predicted by the two models, with the observation that the "more random" one predicts higher delays. Such stochastic ordering properties, formalizing the manifestation of the folk principle that "*deterministm minimizes the queue*", have been studied in the context of queueing systems (see the related work section) and even for risk management (see, e.g., Asmussen *et al.* [7]).

Let us consider a more complex queueing model subject to flows' multiplexing and which explicitly accounts for the number of parallel flows at time n, denoted throughout by $F(n)$. While there is an overwhelming work on *static queues* whereby $F(n)$ is a constant (e.g., the results provided in Chapter 3), much less is known on *dynamic queues* whereby $F(n)$ is a stochastic process[1]. Moreover, since communication networks are more accurately modelled by dynamic queues (e.g., the number of parallel flows traversing an Internet router *actually is* a stochastic process) the goal of this chapter is to provide an analytical understanding on the role of randomness in $F(n)$ on the queue size (e.g., How fast does it grow?). In particular, this chapter attempts to provide insights into the question "What is the *joint impact* of stochastic models, for both $F(n)$ and the flows' themselves, on the queue size?".

To answer such a fundamental question we consider two randomness

[1] We use the terminologies *static queue* when the number of parallel flows is deterministic and *dynamic queue* when the number of flows is random. While not standard and perhaps confusing, the terminology is preferred as a convenient shorthand.

models: One is subject to strong i.i.d. assumptions (paralleling Subsection 3.2.1), enabling a tractable analytical study on the impact of various distributions of $F(n)$ on the queue size. The second more realistic case is when $F(n)$, and also the flows, have a Markov structure (paralleling Subsection 3.2.2). While stochastic bounds on the queue size can also be derived, as in the i.i.d. case, they are expressed in terms of eigen-values/vectors hampering an explicit analytical investigation; for this reason, numerical evaluations will be invoked.

By using convexity arguments, the simplicity of the i.i.d. case enables showing that the best-case distribution from the perspective of the queue size is the intuitively obvious *constant distribution*, extending thus the folk principle that "determinism minimizes the queues" from static to dynamic queues. The second extremal property concerns the corresponding worst-case distribution, i.e., which law of $F(n)$ maximizes the queue size? It is shown that this is a *bimodal distribution*, with mass on the extremes of $F(n)$'s range and therefore maximizing all the moments. This result also agrees with parallel results from static queues concerning extremal properties of bimodal distributions (see Section 5.1.3). Another immediate result is that strong conditions on ordering distributions are needed, in contrast to parallel results from M/G/k queues. The perhaps most fundamental insight is that the above folk principle can fail, in the more realistic case when $F(n)$ is Markov-modulated. Concretely, we find that there is a transition in the flows' average lifetimes at which dynamic queue models yield (stochastically) larger queues than the corresponding (normalized) static queue models.

These overall insights raise the important caveat that approximating (realistic) dynamic queues by static queues (i.e., replacing the stochastic process $F(n)$ by its mean $\mathbb{E}[F(n)]$) can yield very misleading results, which can either overestimate or underestimate the "true" results.

The rest of this chapter is structured as follows: First we overview related work. In Section 5.2 we treat dynamic queues under i.i.d. multiplexing, and in Section 5.3 under more realistic Markovian assumptions.

5.1 Related Work

Here we overview previous work related to the main topics of this chapter, i.e., the relevance of studying dynamic queues (Subsection 5.1.1, stochastic orderings concerning queueing metrics (Subsection 5.1.2), and extremal distributions for minimizing/maximizing queues (Subsection 5.1.3).

5.1.1 Dynamic Queues and Analytical Approaches

The importance of accounting for the elastic nature of Internet traffic, determined by a dynamic or random number of parallel flows, has been recognized in the context of bandwidth sharing. Massoulié and Roberts showed that randomness in the number of parallel flows can have unpredictable consequences on the throughput of long-lived flows, irrespective of the assigned weights to the parallel flows [107]. In a similar setting, Bonald and Massoulié demonstrated that network stability is insensitive to a broad range of fair allocations [20], generalizing a result of de Veciana *et al.* for weighted max-min fairness [144]. A more recent study of Liu *et al.* showed that stability is actually sensitive to the settings of α fairness, in networks with non-convex and time-varying rate regions [102], generalizing an earlier result of Bonald and Proutière [21]. Another notable insensitivity result is that in dynamic scenarios with flows arriving as a Poisson process, the first moments of the number of flows and the flows' throughput do not depend on the flow size distribution or on the properties of the flows' arrivals (Fred *et al.* [63]).

A general way to model randomness in the number of flows is through a queue with bulk arrivals, i.e., the $G^{[F]}/G/1$ queue, whereby customers arrive in batches of random size F according to a renewal process, and customers have some service time distribution. In the case of Poisson renewals, exact solutions exist for various queueing metrics (e.g., Laplace transforms for waiting times) and various scheduling of the batches: FIFO (Burke [30]), with priorities (Takagi and Takahashi [136]), or PS (Bansal [13]); for more general renewals solutions are given numerically (Schleyer [125]) or in terms of bounds (Yao *et al.* [160]).

For an excellent treatment of queues with bulk arrivals see Chaudhry and Templeton [36]. Our contribution herein is to analyze very general distributions (subject to a finite moment generating function (MGF)).

Other analytical approaches address queueing models with fluid arrivals. For instance, the classical Anick-Mitra-Sondhi model [6], with a fixed number of flows producing arrivals at some rates according to the states of Markov On-Off processes, can be regarded as a queue with a binomial number of flows. Queueing in related fluid models can be analyzed exactly in terms of spectral representations, at a cost of high computational complexity due to a combinatorial explosion in the number of states [133]. The advantage of our approach is that it provides *simple* (convex) upper and lower bounds on queueing metrics, which further permit the immediate analysis of extremal properties.

5.1.2 Stochastic Orderings

Stochastic orderings, setting partial orders for queueing metrics, were previously addressed in static scenarios. An elementary example on the role of the variability of underlying distributions was just illustrated in Eqs. (5.1) and (5.2). More generally, in M/G/k queues, the average delay was shown to be an increasing function of the variance of the service time distribution (see Whitt [151, 152]). Extensions of this monotonicity property were considered by Asmussen and O'Cinneide in [8] for Markov-modulated M/G/1 queues. For single queues with Markov-modulated Poisson processes, and under some monotonicity assumptions on the generator of a Markov chain modulating the intensity, Bäuerle and Rolski [15] proved that the queues increase by scaling down the generator. In the case of networks with Poisson arrivals, it was shown that exponential packet sizes yield smaller delays than averaged-out sizes but not in full generality (for a counterexample see Harchol-Balter and Wolfe [72]). When the arrivals are not Poisson however, the monotonicity property fails in some cases even for single queues (see, e.g., Ross [124]).

This chapter shows that the monotonicity of the variance alone of the

number of flows $F(n)$ is *not* sufficient to infer stochastic orderings on the queue size; instead, a sufficient condition is given by the monotonicity of the MGF. In the light of related work, our result thus indicates that queuing metrics are much more sensitive to the variability of the number of flows than of the flows themselves; this claim is further supported by the emphasized sensitivity of dynamic queues to peak rather than average-values.

5.1.3 Extremal Distributions

A "folk theorem" in queueing theory states that, when the average inter-arrival (service) time is fixed, the constant inter-arrival (service) time distribution *minimizes* queueing metrics such as average waiting time. This result was proved for renewal processes (see Rogozin [122]) and also for more general arrival processes with exponential service times (see Hajek [69] and Humblet [78]). A related variant of the underlying intuitive principle that "determinism minimizes the waiting" is that round-robin server assignment outperforms random server assignment (see Makowski and Philips [106]).

In turn, bimodal distributions maximize queue lengths in GI/M/1 queues (Whitt [153]), in G/M/1 queues with bulk arrivals (Lee and Tsitsiklis [99]), and more recently in queues with bulk arrivals and finite buffers (Bušić *et al.* [31]). We will show that these extremal properties characteristic to static queues extend to dynamic queues as well.

5.2 I.I.D. Multiplexing

We first consider multiplexing under strong i.i.d. assumptions of the flows. This simplified case enables an analytical study on the impact of the distribution of the number of parallel flows on the queue size. For the more realistic Markov-modulated multiplexing case, which is only amenable to a numerical study, see the next section.

We consider the single-queue scenario from Chapter 3 (as depicted in

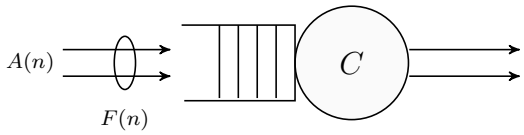


Figure 5.1: A server with constant rate C serving a single queue with input $A(n)$ consisting of $F(n)$ parallel flows.

Figure 5.1): The queue has an infinite sized buffer, whereas the server has a constant capacity C and serves the arrivals in a work-conserving manner.

After introducing the arrival model, we will derive upper and lower bounds on the queue size, and then discuss on extremal distributions of $F(n)$ relative to achievable queue sizes; the obtained analytical insights will be finally complemented by some illustrative numerical results.

5.2.1 Arrival Model

The time model is discrete. The number of parallel flows active at time n is represented by a stationary stochastic process $F(n)$. The cumulative arrival process $A(n)$, counting the number of data units (e.g., packets) over the time interval $[0, n]$ is defined recursively as

$$A(n) = A(n-1) + \sum_{i=1}^{F(n)} a_i(n) \ , \tag{5.3}$$

with the initial condition $A(0) = 0$. The instantaneous arrival process at time n is represented by the random vector $\mathbf{a}(n) = (a_1(n), a_2(n), \dots)$. When clear from the context, we will refer to the elements of $F(n)$ by F, and to the elements of $\mathbf{a}(n)$ simply by a.

For some $\theta > 0$, we assume that the moment generating functions (MGFs)

$$\phi_a(\theta) := \mathbb{E}\left[e^{\theta a}\right] \text{ and } \phi_F(\theta) := \mathbb{E}\left[e^{\theta F}\right]$$

are finite. Moreover, we assume that the elements of $\mathbf{a}(n)$ and $F(n)$ are each i.i.d., and jointly independent.

5.2.2 The Queue Distribution

Since the increment process $A(n) - A(n-1)$ is reversible, the stationary queue size Q can be written as (see Eq. (2.7))

$$Q =_{\mathcal{D}} \sup_{n \geq 0} \{A(n) - Cn\} \ .$$

The next theorem provides upper and lower bounds for the distribution of Q:

Theorem 5.1. *(Q's Distribution, i.i.d.-case) Consider the arrival process from Eq. (5.3) and assume that the elements of* $\mathbf{A}$ *are i.i.d. with MGF* $\phi_a(\theta)$*, and the elements of* $\mathbf{F}$ *are i.i.d. with MGF* $\phi_F(\theta)$*; also,* $\mathbf{A}$ *and* $\mathbf{F}$ *are independent. Consider a queue with service rate* C *and let*

$$\theta^* := \sup \{\theta \geq 0 \mid \phi_F(\log \phi_a(\theta)) = \phi_C(\theta)\} \ . \tag{5.4}$$

Then we have the upper bound for all $\sigma \geq 0$

$$\mathbb{P}\Big(Q \geq \sigma\Big) \leq e^{-\theta^* \sigma} \ . \tag{5.5}$$

If in addition there exists the constants a_{max} *and* N_{max} *such that* $a_1(1) \leq a_{max}$ *almost surely (a.s.),* $F(1) \leq N_{max}$ *a.s., and* $N_{max} a_{max} > C$*, then we have the lower bound for all* $\sigma \geq 0$

$$\mathbb{P}\Big(Q \geq \sigma\Big) \geq e^{-\theta^*(N_{max} a_{max} - C)} e^{-\theta^* \sigma} \ .$$

The upper and lower bounds are asymptotically *exact* (i.e., the following limit $\lim_{\sigma \to \infty} \frac{1}{\sigma} \log \mathbb{P}(Q > \sigma) = \theta^*$ holds) since the two exponential bounds have the same decay rate θ^*. We remark that the theorem immediately extends to the case of a queue with random instantaneous capacities $(C(1), C(2), \dots)$, if these are i.i.d.; the only modification is that $\phi_C(\theta)$ in Eq. (5.4) is to be replaced by $\phi_{C(1)}(\theta)$. In the theorem, we do not explicitly impose the stability condition $\mathbb{E}[a]\mathbb{E}[F] = \phi_a'(0)\phi_F'(0) < C$. Unless this is true then $\theta^* = 0$ in Eq. (5.4). Also,

for the lower bound, the condition $N_{\max} a_{\max} > C$ avoids the trivial situation of no queueing.

To prove the upper bound we apply Kingman's technique for GI/GI/1 queues based on an exponential martingale [92]. To prove the lower bound we rely on some additional ideas from Ross [123] and Chang [35].

Proof. The proof for the upper bound is a variant of the proofs of Theorem 3.4, and Lemma 3.12, taking into account the additional randomness that stems from the process $F(n)$. Let $x \geq 0$. With θ^* as in the theorem we construct the random process

$$X_n = e^{\theta^*(A(n)-Cn)}$$

for all $n \geq 0$. Let also the associated filtration of σ-algebras

$$\mathcal{F}_n = \sigma(\mathbf{a}(1), \ldots, \mathbf{a}(n), F(1), \ldots, F(n)) \ ,$$

where $\mathbf{a}(n)$'s denote the vectors $(a_1(n),\ a_2(n),\ \cdots)$.

The key to the proof is to show that X_n is a martingale. For some $n \geq 1$ we can write for the conditional expectation

$$\begin{aligned} \mathbb{E}\left[X_n \mid \mathcal{F}_{n-1}\right] &= \mathbb{E}\left[X_{n-1} e^{\theta^*\left(\sum_{i=1}^{F(n)} a_i(n) - C\right)} \;\middle|\; \mathcal{F}_{n-1}\right] \\ &= X_{n-1}\mathbb{E}\left[e^{\theta^*\left(\sum_{i=1}^{F(n)} a_i(n) - C\right)}\right] \ , \end{aligned}$$

using that X_{n-1} is $\mathcal{F}_{n-1}$-measurable and the independence assumptions on $\mathbf{A}$ and $\mathbf{F}$. Further conditioning on $F(n)$ we can compute the last expectation

$$\begin{aligned} \mathbb{E}\left[e^{\theta^* \sum_{i=1}^{F(n)} a_i(n)}\right] &= \sum_{m \geq 0} \phi_a(\theta^*)^n \mathbb{P}\left(F(n) = m\right) \\ &= \phi_F\left(\log \phi_a(\theta^*)\right) \ , \end{aligned}$$

after using the independence properties again. With this we can continue above

$$\mathbb{E}\left[X_n \mid \mathcal{F}_{n-1}\right] = X_{n-1}\phi_C(-\theta^*)\phi_F\left(\log \phi_a(\theta^*)\right) = X_{n-1} \ ,$$

using the definition of θ^*. Therefore the sequence X_n is a martingale (relative to $\mathcal{F}_n$). The proof for the upper bound follows exactly as in the proof of Theorem 3.4.

To prove the lower bound we further let $y \geq 0$ and denote

$$N := \inf\{n \geq 0 \mid A(n) - Cn \geq \sigma\} \ , \text{ and}$$
$$N_y := \min\{N, \inf\{n \geq 0 \mid A(n) - Cn \leq -y\}\} \ .$$

N_y is the first time to exit the interval $[-y, \sigma]$. Note that N_y is a finite stopping time relative to $\mathcal{F}_n$. By the optional stopping theorem (see Lemma 2.8), the process $(X_{N_y \wedge n})_n$ is a martingale, which is bounded and hence uniformly integrable. Thus, $X_{T_y \wedge n} \to X_{T_y}$ a.s. and in L^1 (see [154, Theorem 13.7]), and we have

$$\begin{aligned}\mathbb{E}[X_0] =& \mathbb{E}\left[X_{N_y \wedge 0}\right] = \mathbb{E}\left[X_{N_y}\right] \\ =& \mathbb{E}\left[X_{N_y} \mid A(N_y) \geq CN_y + \sigma\right] \mathbb{P}(A(N_y) \geq CN_y + \sigma) \\ &+ \mathbb{E}\left[X_{N_y} \mid A(N_y) \leq CN_y - y\right] \mathbb{P}(A(N_y) \leq CN_y - y) \ . \end{aligned} \tag{5.6}$$

Note further the implications of events

$$\begin{aligned}\{A(N_y) \geq CN_y + \sigma\} &\Rightarrow \{N_y = N\} \\ &\Rightarrow \{A(N_y - 1) < C(N_y - 1) + \sigma\} \\ &\Rightarrow \{A(N_y) \leq CN_y + N_{\max} a_{\max} - C + \sigma\} \ ,\end{aligned}$$

where we used the definition of N and the bounding constants from the theorem. We can thus bound the previous sum as

$$\mathbb{E}[X_0] \leq e^{\theta^*(N_{\max} a_{\max} - C + \sigma)} \mathbb{P}(A(N_y) \geq CN_y + \sigma) + e^{-\theta^* y} \ .$$

Letting $y \to \infty$ yields

$$\mathbb{E}[X_0] \leq e^{\theta^*(N_{\max} a_{\max} - C + \sigma)} \mathbb{P}(N < \infty) \ .$$

The lower bound from the theorem follows immediately from $\mathbb{P}(N < \infty) = \mathbb{P}(Q \geq \sigma)$ and $\mathbb{E}[X_0] = 1$, which completes the proof. □

5.2.3 Extremal Distributions

Given the bounds from Theorem 5.1, we can identify the best/worst-case distributions for $F(n)$ which minimize/maximize the queue size. Then we discuss conditions under which a particular distribution is "better" or "worse" than another.

To formalize the underlying stochastic ordering, and thus the meaning of "better" and "worse", we say that a queue Q_1 is smaller than another queue Q_2 if the corresponding decay rates θ_1 and θ_2 (e.g., defined in Eq. (5.4)) satisfy

$$\theta_1 \geq \theta_2 \ ,$$

i.e., if the tail probability of Q_1 decays faster than the tail probability of Q_2.

Best-Case Distribution

First we briefly show the intuitive result that the best-case distribution of F is the constant one. What is more interesting is that neither of the distributions of F and a dominates the other, when jointly accounting for both.

Given the i.i.d. assumption, Jensen's inequality (see Lemma 2.1)

$$e^{\theta \mathbb{E}[X]} \leq \mathbb{E}\left[e^{\theta X}\right]$$

(for some r.v. X) yields that

$$\phi_{\mathbb{E}[F]}(\log \phi_a(\theta)) \leq \phi_F(\log \phi_a(\theta)) \ .$$

The left-hand side corresponds to the composition of MGFs from the definition of θ from Eq. (5.4) when there is no randomness in the number of parallel flows, i.e., when the elements of $F(n)$ are equal to a single constant. In turn, the right-hand side accounts for randomness in $F(n)$. Because of the inequality above, it follows that the value of θ^* from Eq. (5.4) decreases when accounting for randomness, which further means that the queue increases correspondingly. The best-distribution is thus the constant, which in particular minimizes all the moments.

Finally, we point out the interesting fact that none of the randomness in the number of parallel flows, or at the flow level, dominates the other. That is because there is no general ordering between the terms

$$\phi_{\mathbb{E}[F]}\left(\log \phi_a(\theta)\right) \quad \text{and} \quad \phi_F\left(\log \phi_{\mathbb{E}[a]}(\theta)\right) \ .$$

Indeed, using Jensen's inequality, the left term is the smallest when a is non-random (i.e., $a = \mathbb{E}[a]$) and F is random. In turn, the left term is the largest when F is non-random (i.e., $F = \mathbb{E}[F]$) and a is random. This fundamental lack of monotonicity suggests that, even for the purpose of deriving bounds on the queue size distribution, both the randomness in the number of flows and at the flow level must be jointly accounted for. In other words, simplifying the queueing model by averaging-out either F or a can lend itself to incorrect results.

Worst-Case Distribution

According to Theorem 5.1, the problem of determining the distribution of F which maximizes the queue reduces to solving for

$$\underset{F,\ \text{fixed}\ \mathbb{E}[F]}{\operatorname{argmax}} \ \mathbb{E}\left[e^{\theta F}\right] \ , \tag{5.7}$$

for all $\theta > 0$. The next Lemma gives the solution:

Lemma 5.2. *(*Worst-case distribution*) Assuming that F has the support $\{0, 1, \ldots, F_{max}\}$, the solution of Eq. (5.7) is the bimodal distribution with*

$$\pi_0 = 1 - \frac{\mathbb{E}[F]}{F_{max}} \quad \textit{and} \quad \pi_{F_{max}} = \frac{\mathbb{E}[F]}{F_{max}} \ .$$

Proof. Assume that there exists $0 < i < F_{\max}$ such that $\pi_i := \mathbb{P}(F = i) > 0$. Denoting $x = \frac{F_{\max} - i}{F_{\max}} \pi_i$, let us observe that

$$\pi_0 + \pi_i e^{\theta i} + \pi_m e^{\theta F_{\max}} \leq \pi_0 + x + (\pi_{F_{\max}} + \pi_i - x)\, e^{\theta F_{\max}} \ . \tag{5.8}$$

Indeed, showing this inequality reduces to showing that the function

$$f(i) := \frac{e^{\theta F_{\max}} - e^{\theta i}}{F_{\max} - i}$$

is monotonically increasing over $i \in \{0, 1, \ldots, F_{\max} - 1\}$. This can be shown immediately by extending $f(\cdot)$ to continuous time, differentiating, and using the inequality $e^z \geq z + 1$ for $z \geq 0$.

Therefore, Eq. (5.8) shows that a "worse" distribution can be obtained by appropriately spreading the distribution mass to the extremes. Note that the new distribution retains the average value $\mathbb{E}[F]$ since

$$i\pi_i + m\pi_{F_{\max}} = F_{\max} \left(\pi_{F_{\max}} + \pi_i - x\right) \ .$$

The proof is complete by repeatedly spreading the mass, as in Eq. (5.8), for all $0 < i < F_{\max}$ for which $\pi_i > 0$. □

We note that the bimodal distribution was found to attain the maximum over a partial order set according to convex ordering (see Shaked and Shanthikumar [129], Theorem 3.A.24, p. 125); in our case, the ordering is restricted to MGFs only.

5.2.4 Ordering Distributions

The constant best-case distribution and the bimodal worst-case distribution identified earlier are clearly unrealistic from a practical point of view. It is thus of interest to analyze the relationship between different (and more realistic) distributions from the point of view of being "better" or "worse".

Following the presented arguments, an immediate sufficient condition for a distribution F_1 to be "better" than a distribution F_2 (subject to the condition $\mathbb{E}[F_1] = \mathbb{E}[F_2]$) is an ordering on the MGFs, i.e.,

$$\mathbb{E}\left[e^{\theta F_1}\right] \leq \mathbb{E}\left[e^{\theta F_2}\right] \ , \tag{5.9}$$

for all $\theta > 0$. This can be seen from the construction of the optimal θ from, e.g., Eq. (5.4) in Theorem 5.1.

The condition from Eq. (5.9) is clearly strong as it implicitly involves all the moments of F_1 and F_2. In the light of the discussion from Section 5.1.2 that an ordering on the variance (of packet distributions) is sufficient for ordering the queue sizes in M/G/k queues, we point out that a similar condition on the variance is not sufficient in the current context (mainly due to non-Poisson input). To quickly illustrate this negative fact, by counterexamples, let $C = 3$, F_1 the Uniform distribution with support $\{0, 1, 2, 3, 4\}$ and F_2 having the same support, the same average $\mathbb{E}[F_1] = \mathbb{E}[F_2] = 2$, and the mass $\pi_1 = 0.5$, $\pi_2 = 0.25$, and $\pi_4 = 0.25$. One can show that $Var[F_1] > Var[F_2]$ and

$$\sup\left\{\theta \mid e^{\theta C} = \mathbb{E}\left[e^{\theta F_1}\right]\right\} > \sup\left\{\theta \mid e^{\theta C} = \mathbb{E}\left[e^{\theta F_2}\right]\right\} \ , \tag{5.10}$$

i.e., F_1 is "better" than F_2.

In turn, by changing the mass of F_2 to $\pi_1 = 0.5$ and $\pi_4 = 0.5$, one can show that $Var[F_1] < Var[F_2]$ but F_1 is "worse" than F_2. To conclude, the variance alone of F is not a sufficient indicator for ordering the queues. Moreover, in the light of the above counterexamples, it is conceivable that the sufficient condition from Eq. (5.9), which imposes an ordering on the MGFs, is

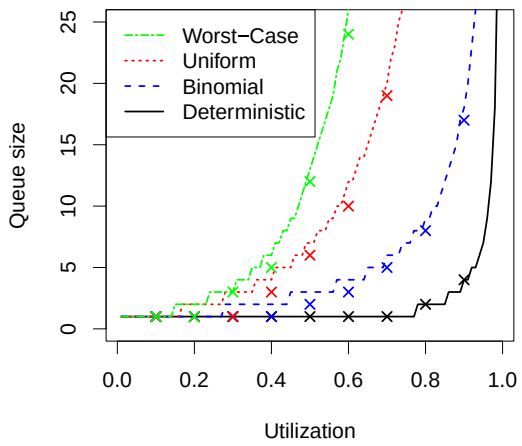


Figure 5.2: Impact of several distributions for the number of parallel flows F on the queue size. Analytical bounds are depicted with lines, whereas corresponding simulation results are depicted with the "×" symbol.

also necessary.

5.2.5 Numerical Results

We now provide numerical evidence on the discrepancy between static and dynamic queues, by varying the distribution of the number of parallel flows F and also the corresponding peak-to-mean ratios.

To keep the analysis concise, we consider a homogeneous scenario in which the elements of a are Bernoulli random variables taking the values 0 and 1 with probabilities $1-p$ and p, respectively. Figure 5.2 illustrates the queue size x, for a fixed violation probability $\varepsilon = 10^{-3}$, and as a function of the utilization factor; the other parameters are $\mathbb{E}[F] = 10$, $F_{\max} = 20$, $C = 9$, and p is scaled accordingly for each utilization value. The worst-case distribution is the one from Lemma 5.2. The figure indicates that the impact of F's distribution on the queue size can be substantial (e.g., as large as many orders of magnitude). Moreover, simulation results (depicted with the "×" symbol, for each distribution) indicate that our analytical bounds are quite tight.

In Figure 5.3 we illustrate the impact of several distributions on the queue size, especially when varying the peak-to-mean ratio (the same parameters

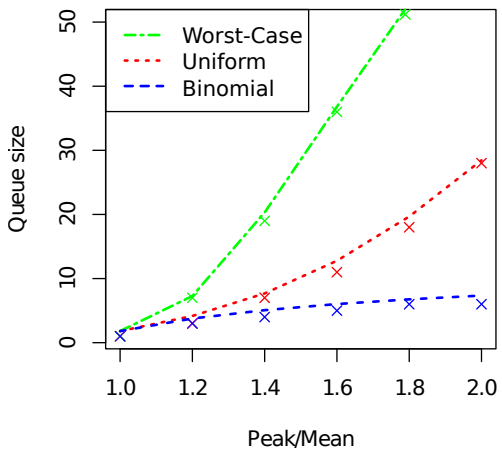


Figure 5.3: Impact of several distributions for the number of parallel flows F on the queue size, depending on the peak-to-mean ratio.

are used as in Figure 5.2, except for scaling the peak and fixing the utilization to 75%). The figure provides strong evidence that approximating dynamic by static queues can be arbitrarily misleading for queueing metrics, even for moderate values of the peak-to-mean ratio.

As a side remark, the obtained results uncover several fundamental similarities and differences amongst the concepts of capacity when defined in 1) information theory (e.g., as the channel capacity), 2) static, and 3) dynamic queues (e.g., as the required capacity to guarantee some queueing constraints). All three corresponding maximal capacities are attained by the intuitively obvious constant distribution, which in particular has zero entropy. In turn, while the minimal channel capacity is attained by the uniform distribution (which maximizes the entropy), the two queueing minimal capacities are attained by bimodal distributions; this conceptual difference stems from the different scalar measures of a distribution used in information theory (i.e., the entropy) and queues (i.e., moments accounting for actual values).

5.3 Markov-Modulated Multiplexing (MMM)

In this section we consider the Markov-Modulated Multiplexing (MMM) case, i.e., $F(n)$ is modulated by a Markov process. While MMM is more realistic than i.i.d. multiplexing, the implicit nature of the obtained stochastic bounds only allows for qualitative insights on the behavior of dynamic queues using numerical results.

5.3.1 Arrival Model

To model MMM we consider a number of $F_{\max}$ Markov-Modulated sources. For each source, transmissions are modulated by a Markov chain with state space $\mathcal{S} = \{0, 1, \text{IA}\}$ (see Figure 5.4).

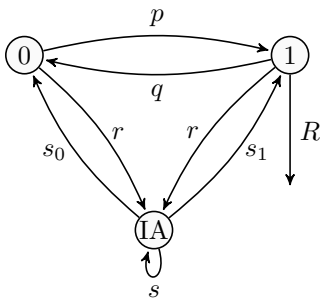


Figure 5.4: A Markov process modulating the arrival process of a source

The upper two states correspond to a typical Markov-Modulated On-Off (MMOO) source (see Figure 3.3 in Subsection 3.2.2) which is idle while in state "0" and transmits at constant rate R while in state "1". The extra state "IA" models the situation that the MMOO source may be inactive, i.e., it is no longer considered present. The difference between the states "0" and "IA" is that $r << q$, i.e., it is much less likely for the source to enter the inactive state than the idle state. From the inactive state, the source reactivates according to the (conditional) steady-state probability vector of the MMOO source, i.e.,

$$\pi_{\mathbf{act}} = \left(\frac{q}{p+q}, \frac{p}{p+q} \right) ,$$

(see Eq. (3.15)) such that $s_0 = \frac{q}{p+q}(1-s)$ and $s_1 = \frac{p}{p+q}(1-s)$. The transition matrix of the entire Markov chain is

$$T = \begin{pmatrix} (1-p)(1-r) & p(1-r) & r \\ q(1-r) & (1-q)(1-r) & r \\ \frac{q}{p+q}(1-s) & \frac{p}{p+q}(1-s) & s \end{pmatrix} . \tag{5.11}$$

To summarize, the number of parallel flows (i.e., the number of Markov chains *not* delving in the "IA" state) is a (Markov) process $F(n)$ with support $\{0, 1, \ldots, F_{\max}\}$. The fundamental difference from the i.i.d. multiplexing model from Eq. (5.3) is that MMM allows for the dynamic multiplexing of bursty sources (e.g., MMOO processes). In particular, we point out that the model from Eq. (5.3) cannot be simply extended to bursty sources by relaxing the condition that the elements of $\mathbf{A}$ are i.i.d.; for instance, in the case of MMOO sources in Eq. (5.3), their Markovian structure would be ambiguous due to dynamically changing $F(n)$. On the other hand, the proposed MMM model restricts the distribution of $F(n)$ to a binomial, albeit the dynamical structure (i.e., driven by an implicit Markov chain) of $F(n)$ is captured.

5.3.2 The Queue Distribution

Let $(a_i(n))_n$, $i \in \{1, \ldots, F_{max}\}$, denote F_{max} independent copies of Markov-Modulated sources as in Figure 5.4. Then, the (cumulative) arrival process $A(n)$ is recursively given by

$$A(n) = A(n-1) + \sum_{i=1}^{F_{max}} f(a_i(n)) , \tag{5.12}$$

where

$$f(x) := \begin{cases} R & x = 1 \\ 0 & x \in \{0, \mathrm{IA}\} \end{cases} .$$

It is easy to check that the stationary distribution of each source is given

by the probability vector

$$\pi = \left(\frac{q(1-s)}{(p+q)(r+1-s)}, \ \frac{p(1-s)}{(p+q)(r+1-s)}, \ \frac{r}{r+1-s} \right) \ .$$

Further, the balance equations

$$\pi_i T(i,j) = \pi_j T(j,i) \ , \quad i,j \in \mathcal{S}$$

hold so that the sources $a_i(n)$, and hence the increment process $A(n) - A(n-1)$, are reversible. Consequently, the stationary queue length Q has again the representation (see Eq. (2.7))

$$Q = \sup_{n \geq 0} \{A(n) - Cn\} \ .$$

Recall the definition of the exponentially transformed transition matrix (see Eq. (3.13)):

$$T_\theta(i,j) := T_\theta(i,j) e^{\theta f(j)} \ , \quad i,j \in \mathcal{S} \ ,$$

for $\theta \geq 0$. Further, $\lambda(\theta)$ denotes the maximal positive eigenvalue and ν a corresponding positive eigenvector.

The next theorem provides upper and lower bounds on Q's distribution:

Theorem 5.3. *(Q's* Distribution, MMM-case*) Consider the arrival model from Eq. (5.12) and a constant server capacity $C > 0$. Let*

$$\theta^* := \sup \left\{ \theta \geq 0 \ \middle| \ \lambda(\theta) = e^{\theta C F_{max}^{-1}} \right\} \ ,$$

then the following bounds on the backlog hold for $\sigma > 0$:

$$\mathbb{P}(Q \geq \sigma) \leq H_u e^{-\theta^* \sigma} \ , \ \textit{and} \quad \mathbb{P}(Q \geq \sigma) \geq H_l e^{-\theta^* \sigma} \ ,$$

where

$$H_u = \frac{\left(\pi_0\nu_0 + \pi_1\nu_1 + \pi_{\mathrm{IA}}\nu_{\mathrm{IA}}\right)^{F_{max}}}{\nu_1^{\lceil CR^{-1}\rceil} + \min\{\nu_{\mathbf{0}}, \nu_{\mathrm{IA}}\}^{F_{max}-\lceil CR^{-1}\rceil}} \ , \ \textit{and}$$
$$H_l = \frac{\left(\pi_0\nu_0 + \pi_1\nu_1 + \pi_{\mathrm{IA}}\nu_{\mathrm{IA}}\right)^{F_{max}}}{\max_s \nu_s^{F_{max}} e^{\theta^*(RF_{max}-C)}} \ .$$

Note that the definition of θ^* resembles the one from Theorem 5.1 with the only difference that the MGF is replaced by the eigenvalue. We also note that $\theta^* = 0$ when the queue is not stable, and that the upper and lower bounds are asymptotically exact since they have the same decay rate θ^*.

Proof. By Lemma 3.15, the processes

$$X_n^i := \nu_{a_i(n)} e^{\theta^*\left(\sum_{k=1}^n f(a_i(k)) - CF_{max}^{-1}n\right)} \ ,$$

(for fixed $0 \le i \le F_{max}$) are martingales. By the independence assumption on the F_{max} arrivals the product

$$X_n := \prod_{i=1}^{F_{max}} X_n^i = \prod_{i=1}^{F_{max}} \nu_{a_i(n)} e^{\theta^*(A(n)-Cn)}$$

is a martingale as well (see Lemma 2.9). Now similarly as in the proof of Theorem 3.4 define the stopping time

$$N = \inf\{n \ge 0 \mid A(n) - Cn \ge \sigma\}$$

and then apply the optional stopping theorem to $N \wedge n$, implying that

$$\mathbb{E}[X_0] = \mathbb{E}[X_{N\wedge n}] \ge \mathbb{E}[X_{N\wedge n} I_{\{N\le n\}}] \ge e^{\theta^*\sigma}\mathbb{E}[\prod_{i=1}^{F_{max}} \nu_{a_i(N)} I_{\{N\le n\}}] \ .$$

As in the proof of Corollary 3.17, at time N at least $\lceil CR^{-1}\rceil$ chains are trans-

mitting. Therefore:

$$\prod_{i=1}^{F_{max}} \nu_{a_i(N)} \geq \nu_1^{\lceil CR^{-1} \rceil} + \min\{\nu_{\mathbf{0}}, \nu_{\mathrm{IA}}\}^{F_{max} - \lceil CR^{-1} \rceil} = \frac{\mathbb{E}[X_0]}{H_u}$$

The upper bound then follows as in the proof of Theorem 5.1 by letting $n \to \infty$ and observing that

$$\mathbb{P}(Q \geq \sigma) = \mathbb{P}(N < \infty) \ .$$

For the lower bound, define the stopping time

$$N_\tau = \min\{N, \inf\{n \geq 0 \mid A(n) - Cn \leq -\tau\}\}$$

for some $\tau \geq 0$. Using the same arguments as in the proof of Theorem 5.1 we have

$$\begin{aligned}
\mathbb{E}[X_0] =& \mathbb{E}\left[X_{N_\tau} \mid A(N_\tau) - CN_\tau \geq \sigma\right] \mathbb{P}(A(N_\tau) - CN_\tau \geq \sigma) \\
&+ \mathbb{E}\left[X_{N_\tau} \mid A(N_\tau) - CN_\tau \leq -\tau\right] \mathbb{P}(A(N_\tau) - CN_\tau \leq -\tau) \\
\leq& \max_s \nu_s^{F_{max}} e^{\theta^*(RF_{max} - C + \sigma)} \mathbb{P}(A(N_\tau) - CN_\tau \geq \sigma) + \max_s \nu_s^{F_{max}} e^{-\theta^* \tau} \ .
\end{aligned}$$

Now simply let $\tau \to \infty$:

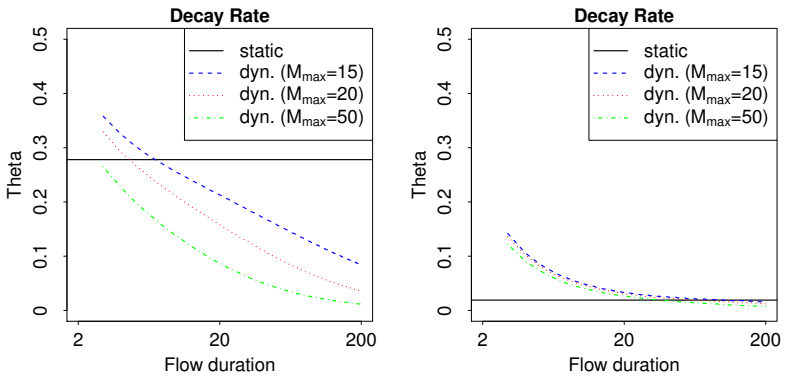
$$\mathbb{E}[X_0] \leq \max_s \nu_s^{F_{max}} e^{\theta^*(RF_{max} - C + \sigma)} \mathbb{P}(N < \infty) = \frac{\mathbb{E}[X_0]}{H_l} e^{\theta^* \sigma} \mathbb{P}(N < \infty) \ ,$$

which completes the proof. □

5.3.3 Numerical Results

As in Section 5.2, we next discuss the discrepancy between static and dynamic queues. Recall that the *exponential decay rate* θ^* from Theorem 5.3 is the same for the upper and lower bounds, respectively, and is thus the dominating factor for the decay of the overflow probability $\mathbb{P}(Q \geq \sigma)$.

We consider a similar numerical settings as in Section 5.2.5 with an average $F_{avg} = 10$ of homogeneous Markov Modulated sources, as in Figure 5.4,



(a) $p = 0.1$, $q = 0.5$ (b) $p = 0.01$, $q = 0.05$

Figure 5.5: Decay rate θ as a function of the flows' average lifetime r^{-1} for both static and dynamic (dyn.) scenarios ($\rho = 0.75$, $F_{avg} = 10$, RC^{-1} is rescaled for each r^{-1}; the x-axis is shown on a log-scale)

which are active (i.e., dwelling in the states 0 and 1). Formally,

$$\pi_{\text{IA}} = \frac{r}{r + 1 - s} = 0.25 \ . \tag{5.13}$$

The parameter r determines the flow's average lifetime (which equals r^{-1}). Its range is the interval $[0, \frac{1}{3}]$; for $r = 0$ the queues are static, whereas for $r > \frac{1}{3}$ the parameter s cannot be scaled such that Eq. (5.13) holds. The ratio RC^{-1} is scaled such that the link utilization $\rho = 0.75$ remains constant in all cases, i.e.,

$$RC^{-1} = \frac{\rho}{\pi_1 F_{max}} \quad \text{and} \quad RC^{-1} = \frac{\rho}{(\pi_{\mathbf{act}})_1 \, F_{avg}}$$

in the dynamic and static cases, respectively.

In Figure 5.5 we illustrate the dominating factor θ^* from Theorem 5.3, of the probability of $\mathbb{P}(Q \geq \sigma)$, for various average lifetimes r^{-1} of the flows. Compared to 5.5(a), the scenario from 5.5(b) captures burstier flows (by decreasing the transition probabilities by a factor of 10). In both figures we consider a static scenario (i.e., $F_{max} = 10$) and three (properly normalized) dynamic (dyn.) scenarios by varying $F_{max} = 15, 20, 50$.

Figure 5.5(a) highlights the expected behavior that randomness in the number of flows "hurts" the system's performance: Unless the flows are very

short-lived (i.e., $r^{-1} \geq 5$) the backlog in the dynamic case is on average larger than its deterministic counterpart. Interestingly, for $r^{-1} \leq 4$ the performance actually benefits from randomization. This is due to the fact that for very short-lived flows, the (beneficial) property of multiplexing roughly *independent* flows (as the Markov structure lasts very shortly) outruns the (detrimental) effect of the *bursty* sources.

This transition effect, i.e., the actual value of the flows' average lifetime at which dynamic multiplexing "hurts", depends on the flows' own burstiness. This can be seen from Figure 5.5(b) where the transition occurs at much larger average lifetimes (and at which the flows remain roughly independent since the flows' Markov structure survives for around the average dwelling time in one of the states).

In conclusion, the figures indicate that for reasonable (i.e., not very short) average flows' lifetimes, flows' multiplexing "hurts" the queue size. Moreover, the discrepancy between static and dynamic queues depends on the flows' own burstiness and also the distribution/support of the number of flows, and can be arbitrarily large as shown in Figure 5.5(a) for large F_{max} and long flows.

5.4 Summary

In this chapter we utilized the powerful martingale-methodology from Chapters 3 and 4 to investigate the queueing behavior in typically neglected but highly relevant dynamic queues characterized by a *random number of parallel flows*. Under some strong i.i.d. assumptions, enabling a tractable analysis, we have first shown that dynamic queues retain some extremal properties from static queues, i.e., capacities are maximized by constant distributions and are minimized by bimodal distributions. While the i.i.d. case confirms that "*determinism minimizes the queues*", we have shown that this folk principle fails in the more realistic case when the number of parallel flows has a Markov structure. Concretely, we have shown that there is a transition of the flows' average lifetime, below which dynamic queues are smaller than static queues. While our observations

jointly depend on the overall statistics, they nevertheless provide a convincing argument that current approximations of dynamic by static queues can be very misleading, and that a rigorous analysis of queueing scenarios with a dynamic number of flows is necessary.

6

Fork-Join Queueing Systems

The performance analysis of Fork-Join (FJ) systems received new momentum with the recent wide-scale deployment of large-scale data processing that was enabled through emerging frameworks such as MapReduce [54]. The main idea behind these big data analysis frameworks is an elegant divide and conquer strategy with various degrees of freedom in the implementation. The open-source implementation of MapReduce, known as Hadoop [150], is deployed in numerous production clusters, e.g., Facebook and Yahoo [86].

The basic operation of MapReduce is depicted in Figure 6.1. In the *map phase*, a job is split into multiple tasks that are mapped to different workers (servers). Once a specific subset of these tasks finish their executions, the corresponding *reduce phase* starts by processing the combined output from all the

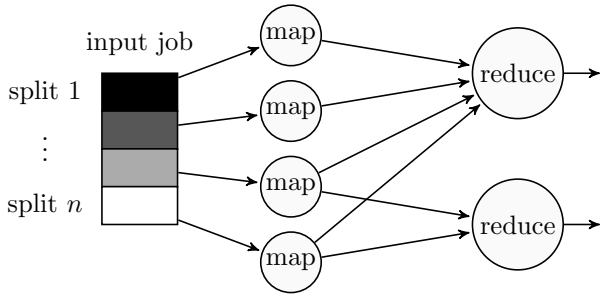


Figure 6.1: Schematic illustration of the basic operation of MapReduce.

corresponding tasks. In other words, the reduce phase is subject to a fundamental synchronization constraint on the finishing times of all involved tasks.

A natural way to model one reduce phase operation is by a *basic* FJ queueing system with K servers. Jobs, i.e., the input unit of work in MapReduce systems, arrive according to some point process. Each job is split into K (map) tasks (or *splits*, in the MapReduce terminology), which are simultaneously sent to the K servers. At each server, each task requires a random service time, capturing the variable task execution times on different servers in the map phase. A job leaves the FJ system when all of its tasks are served; this constraint corresponds to the specification that the reduce phase starts no sooner than when all of its map tasks complete their executions.

Concerning the execution of tasks belonging to different jobs on the same server, there are two operational modes. In the *non-blocking* mode, the servers are work-conserving in the sense that tasks immediately start their executions once the previous tasks finish theirs. In the *blocking* mode, the mapped tasks of a job simultaneously start their executions, i.e., servers can be idle when their corresponding queues are not empty. The non-blocking execution mode prevails in MapReduce due to its conceivable efficiency, whereas the blocking execution mode is employed when the `jobtracker` (the node coordinating and scheduling jobs) waits for all machines to be ready to synchronize the configuration files before mapping a new job; in Hadoop, this can be enforced through the coordination service `zookeeper` [150].

In this chapter we analyze the performance of the FJ queueing model in four practical scenarios by considering two broad arrival classes (driven by either renewal or non-renewal processes) and the two operational modes (i.e., blocking and non-blocking) described above. The key contribution, to the best of our knowledge, are the first non-asymptotic and computable stochastic bounds on the waiting and response time distributions in the most relevant scenario, i.e., non-renewal (Markov modulated) job arrivals and the non-blocking operational mode. Under all scenarios, the bounds are numerically tight especially at high utilizations. This inherent tightness is due to a suitable martingale representation of the underlying queueing system similar to the one of Chapters 3 and 4. The simplicity of the obtained stochastic bounds enables the derivation of scaling laws, e.g., delays in FJ systems scale as $\mathcal{O}(\log K)$ in the number of parallel servers K, for both renewal and non-renewal arrivals, in the non-blocking mode; more severe delay degradations hold in the blocking mode, and, moreover, the stability region depends on the same fundamental factor of $\log K$.

In addition to the direct applicability to the dimensioning of MapReduce clusters, there are other relevant types of parallel and distributed systems such as production or supply networks. In particular, by slightly modifying the basic FJ system corresponding to MapReduce, the resulting model suits the analysis of window-based transmission protocols over multipath routing. By making several simplifying assumptions such as ignoring the details of specific protocols (e.g., multipath TCP), we can provide a fundamental understanding of multipath routing from a queueing perspective. Concretely, we demonstrate that sending a flow of packets over two paths, instead of one, does generally reduce the steady-state response times. The surprising result is that by sending the flow over more than two paths, the steady-state response times start to increase. The technical explanation for such a rather counterintuitive result is that the $\log K$ resequencing price at the destination quickly dominates the tempting gain in the queueing waiting time due to multipath transmissions.

The rest of this chapter is structured as follows: We first discuss related

work on FJ systems and related applications. Then we analyze full mapping, i.e., a mapping of jobs to K servers in Sections 6.2 (renewal input) and 6.3 (non-renewal input). The analysis of partial mapping, i.e., a mapping of jobs to $H < K$ servers follows in Section 6.4. In Section 6.5 we apply the obtained results on the steady-state response time distributions to the analysis of multipath routing from a queueing perspective.

6.1 Related Work

We first review analytical results on FJ systems, and then results related to the two application case studies considered in this chapter, i.e., MapReduce and multipath routing.

The significance of the Fork-Join queueing model stems from its natural ability to capture the behavior of many parallel service systems. The performance of FJ queueing systems has been subject of multiple studies such as [11, 109, 143, 90, 95, 12, 25]. In particular, [11] notes that an exact performance evaluation of general FJ systems is remarkably hard due to the synchronization constraints on the input and output streams. More precisely, a major difficulty lies in finding an exact closed form expression for the joint steady-state workload distribution for the FJ queueing system. However, a number of results exist given certain constraints on the FJ system. The authors of [62] provide the stationary joint workload distribution for a two-server FJ system under Poisson arrivals and independent exponential service times. For the general case of more than two parallel servers there exists a number of works that provide approximations [109, 143, 95, 98] and bounds [11, 12] for certain performance metrics of the FJ system. Given renewal arrivals, [12] significantly improves the lower bounds from [11] in the case of heterogeneous phase-type servers using a matrix-geometric algorithmic method. The authors of [95] provide an approximation of the sojourn time distribution in a renewal driven FJ system consisting of multiple G/M/1 nodes; they show that the approximation error diminishes at extremal utilizations. Refined approximations for the mean sojourn time in

two-server FJ systems that take the first two moments of the service time distribution are given in [90]; numerical evidence is further provided on the quality of the approximation for different service time distributions.

The closest related work to ours is [11], which provides computable lower and upper bounds on the expected response time in FJ systems under renewal assumptions with Poisson arrivals and exponential service times; the underlying idea is to artificially construct a more tractable system, yet subject to stochastic ordering relative to the original one. Our corresponding first order upper bound recovers the $\mathcal{O}(\log K)$ asymptotic behavior of the one from [11], and also reported in [109] in the context of an approximation; numerically, our bound is slightly worse than the one from [11] due to our main focus on computing bounds on the whole distribution (first order bounds are secondarily obtained by integration). Moreover, we show that the $\mathcal{O}(\log K)$ scaling law also holds in the case of Markov modulated arrivals. In a parallel work [91] to ours, the authors adopt a network calculus approach to derive stochastic bounds in a non-blocking FJ system, under a strong assumption on the input; for related constructions of such arrival models see [81].

The work in [82, 83] studies FJ systems where jobs leave the system when a subset $H \leq K$ of its tasks are finished. This system is similar to the partial mapping FJ system that we study in Section 6.4, however, with subtle yet fundamental differences. The FJ system presented in [82, 83] is based on the assumption that when H tasks finish execution, the finished job *purges* the unfinished $K - H$ tasks out their corresponding queues. The authors of [82, 83] provide upper bounds for the mean response times in such systems under Poisson arrivals and general service distributions. In Section 6.4, we consider instead injective task mapping, i.e., jobs are *only* forked onto a subset of servers $H \leq K$. For this type of FJ systems we provide bounds on the steady state waiting and response time distributions under round-robin and random task placement.

Concerning concrete applications of FJ systems, in particular MapReduce, there are several empirical and analytical studies analyzing its perfor-

mance. For instance, [161, 9] aim to improve the system performance via empirically adjusting its numerous and highly complex parameters. The targeted performance metric in these studies is the job response time, which is in fact an integral part of the business model of MapReduce based query systems such as [110] and time priced computing clouds such as Amazon's EC2 [1]. For an overview on works that optimize the performance of MapReduce systems see the survey article [111]. Using a similar idea as in [11], the authors of [138] derive asymptotic results on the response time distribution in the case of renewal arrivals; such results are further used to understand the impact of different scheduling models in the reduce phase of MapReduce. Using the model from [138] the work in [139] provides approximations for the number of jobs in a tandem system consisting of a map queue and a reduce queue in the heavy traffic regime. The work in [145] derives approximations of the mean response time in MapReduce systems using a mean value analysis technique and a closed FJ queueing system model from [142].

Concerning multipath routing, the works [10, 73] provided ground for multiple studies on different formulations of the underlying resequencing delay problem, e.g., [70, 157]. Factorization methods were used in [10] to analyze the disordering delay and the delay of resequencing algorithms, while the authors of [73] conduct a queueing theoretic analysis of an M/G/∞ queue receiving a stream of numbered customers. In [70, 157] the multipath routing model comprises Bernoulli thinning of Poisson arrivals over K parallel queueing stations followed by a resequencing buffer. The work in [70] provides asymptotics on the conditional probability of the resequencing delay conditioned on the end-to-end delay for different service time distributions. For $K = 2$ and exponential interarrival and service times, [157] derives a large deviations result on the resequencing queue size. Our work differs from these works in that we consider a model of the basic operation of window-based transmission protocols over multipath routing, motivated by the emerging application of multipath TCP [117]. We point out, however, that we do not model the specific operation of any par-

ticular multipath transmission protocol. Instead, we analyze a generic multipath transmission protocol under simplifying assumptions, in order to provide a theoretical understanding of the overall response times comprised of both queueing and resequencing delays.

Relative to the existing literature, our key theoretical contribution is to provide *computable* and non-asymptotic bounds on the *distributions* of the steady-state waiting and response times under both *renewal* and *non-renewal* input in FJ systems. The consideration of non-renewal input is particularly relevant, given recent observations that job arrivals are subject to temporal correlations in production clusters. For instance, [37, 85] report that job, respectively, flow arrival traces in clusters running MapReduce exhibit various degrees of burstiness.

6.2 FJ Systems with Renewal Input

We consider a FJ queueing system as depicted in Figure 6.2. Jobs arrive at the input queue of the FJ system according to some point process with interarrival times t_i between the i and $i+1$ jobs. Each job i is split into K tasks that are mapped through a bijection to K servers. A task of job i that is serviced by some server n requires a random service time $x_{k,i}$. A job leaves the system when all of its tasks finish their executions, i.e., there is an underlying synchronization constraint on the output of the system. We assume that the families $\{t_i\}$ and $\{x_{k,i}\}$ are independent.

In the sequel we differentiate between two cases, i.e., *a*) non-blocking and *b*) blocking servers. The first case corresponds to work-conserving servers, i.e., a server starts servicing a task of the next job (if available) immediately upon finishing the current task. In the latter case, a server that finishes servicing a task is blocked until the corresponding job leaves the system, i.e., until the last task of the current job completes its execution. This can be regarded as an additional synchronization constraint on the input of the system, i.e., all tasks of a job start receiving service simultaneously. We will next analyze *a*) and *b*)

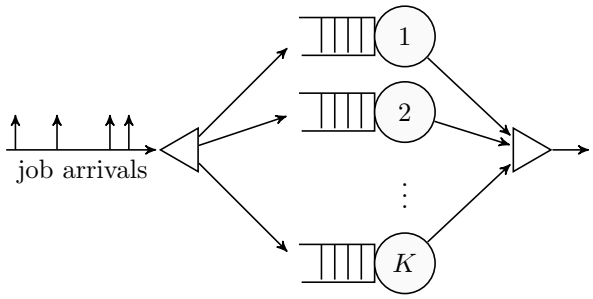


Figure 6.2: A schematic Fork-Join queueing system with K parallel servers. An arriving job is split into K tasks, one for each server. A job leaves the FJ system when all of its tasks are served. An arriving job is considered waiting until the service of the last of its tasks starts, i.e., when the previous job departs the system.

for renewal arrivals.

6.2.1 Non-Blocking Systems

Consider an arrival flow of jobs with renewal interarrival times t_i, and assume that the waiting time of the first job is $w_1 = 0$. Given K parallel servers, the *waiting time* w_j of the jth job is defined as

$$w_j = \max\left\{0, \max_{1\le n\le j-1}\left\{\max_{k\in[1,K]}\left\{\sum_{i=1}^{n} x_{k,j-i} - \sum_{i=1}^{n} t_{j-i}\right\}\right\}\right\} , \qquad (6.1)$$

for all $j \geq 2$, where $x_{k,j}$ is the service time required by the task of job j that is mapped to server k. We count a job as waiting until its last task starts receiving service. Similarly, the *response times* of jobs, i.e., the times until the last corresponding tasks have finished their executions, are defined as $r_1 = \max_k x_{k,1}$ for the first job, and for $j \geq 2$ as

$$r_j = \max_{0\le n\le j-1}\left\{\max_{k\in[1,K]}\left\{\sum_{i=0}^{n} x_{k,j-i} - \sum_{i=1}^{n} t_{j-i}\right\}\right\} , \qquad (6.2)$$

where by convention $\sum_{i=1}^{0} t_i = 0$; for brevity, we will denote $\max_k := \max_{k\in[1,K]}$.

We assume that the task service times $x_{k,j}$ i.i.d.. The stability condition for the FJ queueing system is given as $\mathbb{E}[x_{1,1}] < \mathbb{E}[t_1]$. By stationarity and

reversibility of the i.i.d. processes $x_{k,j}$ and t_j, there exists a distribution of the steady-state waiting time w and steady-state response time r, respectively, which have the representations

$$w =_{\mathcal{D}} \max_{n\geq 0}\left\{\max_{k}\left\{\sum_{i=1}^{n} x_{k,i} - \sum_{i=1}^{n} t_i\right\}\right\} \tag{6.3}$$

and

$$r =_{\mathcal{D}} \max_{n\geq 0}\left\{\max_{k}\left\{\sum_{i=0}^{n} x_{k,i} - \sum_{i=1}^{n} t_i\right\}\right\} \ , \tag{6.4}$$

respectively. Note that the only difference in Eq. (6.3) and Eq. (6.4) is that for the latter the sum over the $x_{k,i}$ starts at $i = 0$ rather than at $i = 1$.

The following theorem provides stochastic upper bounds on w and r. The corresponding proof will rely on submartingale constructions and the optional stopping theorem (see Lemma 2.8).

Theorem 6.1. (Renewals, Non-Blocking) *Given a FJ system with K parallel non-blocking servers that is fed by renewal job arrivals with interarrivals t_j. If the task service times $x_{k,j}$ are i.i.d., then the steady-state waiting and response times w and r are bounded by*

$$\mathbb{P}\left[w \geq \sigma\right] \quad \leq \quad K e^{-\theta_{nb}\sigma} \tag{6.5}$$

$$\mathbb{P}\left[r \geq \sigma\right] \quad \leq \quad K\mathbb{E}\left[e^{\theta_{nb}x_{1,1}}\right] e^{-\theta_{nb}\sigma} \ , \tag{6.6}$$

where θ_{nb} (with the subscript "nb" standing for non-blocking) is defined by

$$\theta_{nb} := \sup\left\{\theta > 0 \;\middle|\; \mathbb{E}\left[e^{\theta x_{1,1}}\right]\mathbb{E}\left[e^{-\theta t_1}\right] = 1\right\} \ . \tag{6.7}$$

We remark that the stability condition $\mathbb{E}\left[x_{1,1}\right] < \mathbb{E}\left[t_1\right]$ guarantees the existence of a positive solution in Eq. (6.7) (see the argument in Remark 3.14).

Proof. Consider the waiting time w. We first prove that for each $k \in [1, K]$ the

process

$$z_k(n) = e^{\theta_{nb} \sum_{i=1}^{n} (x_{k,i} - t_i)}$$

is a martingale with respect to the filtration

$$\mathcal{F}_n := \sigma \left\{ x_{k,m}, t_m \,|\, m \leq n, k \in [1, K] \right\} \ .$$

The independence assumption of $x_{k,j}$ and t_j implies that

$$\begin{aligned} \mathbb{E}\left[z_k(n) \mid \mathcal{F}_{n-1} \right] &= \mathbb{E}\left[e^{\theta_{nb} \sum_{i=1}^{n} (x_{k,i} - t_i)} \,\middle|\, \mathcal{F}_{n-1} \right] \\ &= \mathbb{E}\left[e^{\theta_{nb} (x_{k,n} - t_n)} \right] e^{\theta_{nb} \sum_{i=1}^{n-1} (x_{k,i} - t_i)} \\ &= e^{\theta_{nb} \sum_{i=1}^{n-1} (x_{k,i} - t_i)} \\ &= z_k(n-1) \ , \end{aligned} \tag{6.8}$$

under the condition on θ_{nb} from the theorem.

Next we prove that the process

$$z(n) = \max_k z_k(n) \tag{6.9}$$

is a submartingale w.r.t. $\mathcal{F}_n$. Given the martingale property of each of the z_n and the monotonicity of the conditional expectation we can write for $j \in [1, K]$:

$$\mathbb{E}\left[\max_k z_k(n) \,\middle|\, \mathcal{F}_{n-1} \right] \geq \mathbb{E}\left[z_j(n) \mid \mathcal{F}_{n-1} \right] = z_j(n-1) \ ,$$

where the inequality stems from $\max_k z_k(n) \geq z_j(n)$ for $j \in [1, K]$ a.s., whereas the subsequent equality stems from the martingale property Eq. (6.8) for $z_k(n)$ for all $k \in [1, K]$. Hence, we can write

$$\mathbb{E}\left[z(n) \mid \mathcal{F}_{n-1} \right] \geq \max_k z_k(n-1) = z(n-1) \ , \tag{6.10}$$

which proves the submartingale property.

To derive a bound on the steady-state waiting time distribution, let $\sigma > 0$ and define the stopping time N as usually by

$$N := \inf \left\{ n \geq 0 \,\middle|\, \max_k \sum_{i=1}^{n} (x_{k,i} - t_i) \geq \sigma \right\} , \tag{6.11}$$

such that with the representation of w from Eq. (6.3): $\{N < \infty\} = \{w \geq \sigma\}$. Now, using the optional stopping theorem (see Lemma 2.8) for submartingales with $n \geq 1$:

$$\begin{aligned} K &= \sum_{k \in [1,K]} \mathbb{E}\left[e^{\theta_{nb} \sum_{i=1}^{n} (x_{k,i} - t_i)} \right] \\ &\geq \mathbb{E}\left[\max_k e^{\theta_{nb} \sum_{i=1}^{n} (x_{k,i} - t_i)} \right] \\ &= \mathbb{E}\left[z(n) \right] \\ &\geq \mathbb{E}\left[z(N \wedge n) \right] \\ &\geq \mathbb{E}\left[z(N) 1_{N<n} \right] \\ &\geq e^{\theta_{nb}\sigma} \mathbb{P}\left[N < n \right] , \end{aligned} \tag{6.12}$$

where we used the condition on θ_{nb} from the theorem in the first line, Boole's inequality in the second line, and the optional stopping theorem for submartingales in the fourth line. In the last line we used the definition of the stopping time K. The proof completes by letting $n \to \infty$.

For the response time r, define the processes

$$\tilde{z}_k(n) = e^{\theta_{nb}\left(\sum_{i=0}^{n} x_{k,i} - \sum_{i=1}^{n} t_i\right)} ,$$

which differs from the z_k only in the range of the sum of the service times $x_{k,i}$. Then we proceed as for the derivation of the bound on the waiting time w. The only difference in the derivation is that inequality Eq. (6.12) translates to

$$K\mathbb{E}\left[e^{\theta_{nb} x_{1,1}} \right] \geq \mathbb{E}\left[\max_k e^{\theta_{nb}\left(\sum_{i=0}^{n} x_{k,i} - \sum_{i=1}^{n} t_i\right)} \right] . \qquad \square$$

Fixing the right hand sides in Eq. (6.5) and Eq. (6.6) to ε, we find that the corresponding quantiles on the waiting and response times grow with the number of parallel servers K as $\mathcal{O}(\log K)$, a law which was already demonstrated in the special case of Poisson arrival and exponential service times, and for first moments, in [109], and more generally in [11]. This scaling result is essential for dimensioning FJ systems such as MapReduce computing clusters, as it explains the impact of a MapReduce server pool size K on the job waiting/response times. Note that this result depends on the assumption that the tasks' service times $x_{k,i}$ are fixed, i.e., the "job size" $\sum_{k\in[1,K]} x_{k,i}$ increases in K. By properly rescaling the service times (e.g., by considering $\frac{x_{k,i}}{K}$), a higher value of θ_{nb} in Eq. (6.7) is obtained, and therefore in Theorem 6.1 – for sufficiently large σ – the beneficial effect of a higher decay rate outruns the detrimental effect of an increased constant K.

We note that the bound in Theorem 6.1 can be computed for different arrival and service time distributions as long as the MGF (moment generating function) and Laplace transform from Eq. (6.7) are computable. Given a scenario where the job interarrival process and the task size distributions in a MapReduce cluster are not known a priori, estimates of the corresponding MGF and Laplace transforms can be obtained using recorded traces, e.g., using the method from [68].

Next we illustrate two immediate applications of Theorem 6.1.

Example 1: Exponentially distributed interarrival and service times

Consider that the interarrival times t_i and service times $x_{k,i}$ are exponentially distributed with parameters λ and μ, respectively; note that when $K = 1$ the system corresponds to the M/M/1 queue. The corresponding stability condition becomes $\mu > \lambda$. Using Theorem 6.1, the bounds on the steady-state waiting and response time distributions are

$$\mathbb{P}\left[w \geq \sigma\right] \leq K e^{-(\mu-\lambda)\sigma} \tag{6.13}$$

and

$$\mathbb{P}\left[r \geq \sigma\right] \leq \frac{K}{\rho} e^{-(\mu-\lambda)\sigma} \ , \tag{6.14}$$

where the exponential decay rate $\mu - \lambda$ follows by solving $\frac{\mu}{\mu-\theta}\frac{\lambda}{\lambda+\theta} = 1$, i.e., the instantiation of Eq. (6.7).

Next we briefly compare our results to the existing bound on the mean response time from [11], given as

$$\mathbb{E}\left[r\right] \leq \frac{1}{\mu-\lambda} \sum_{k=1}^{K} \frac{1}{k} \ . \tag{6.15}$$

By integrating the tail of Eq. (6.14) we obtain the following upper bound on the mean response time

$$\mathbb{E}\left[r\right] \leq \frac{\log(K/\rho)+1}{\mu-\lambda} \ .$$

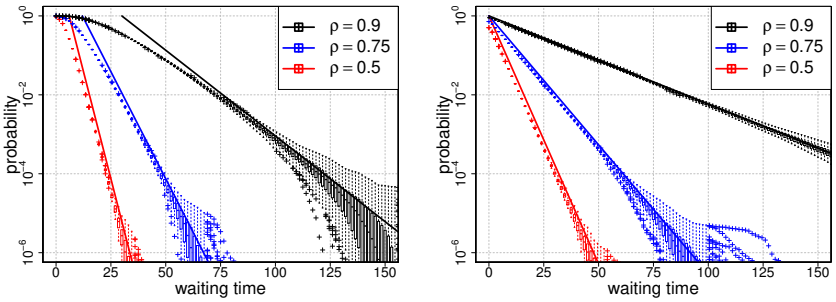
Compared to Eq. (6.15), our bound exhibits the same $\log K$ scaling law but is numerically slightly looser; asymptotically in K, the ratio between the two bounds converges to one. A key technical reason for obtaining a looser bound is that we mainly focus on deriving bounds on distributions; through integration, the numerical discrepancies accumulate.

For the numerical illustration of the tightness of the bounds on the waiting time distributions from Eq. (6.13) we refer to Figure 6.3.(a); the numerical parameters and simulation details are included in the caption.

Example 2: Exponentially distributed interarrival times and constant service times

We now consider the case of i.i.d. exponentially distributed interarrival times t_i with parameter λ, and deterministic service times $x_{k,i} = 1/\mu$, for all $i \geq 0$ and $k \in [1, K]$; note that when $N = 1$ the system corresponds to the M/D/1 queue.

The condition on the asymptotic decay rate θ_{nb} from Theorem 6.1 be-



(a) Non-Blocking (b) Blocking

Figure 6.3: Bounds on the waiting time distributions vs. simulations (renewal input): (a) the non-blocking case Eq. (6.13) and (b) the blocking case Eq. (6.22). The system parameters are $K = 20$, $\mu = 1$, and three utilization levels $\rho = \{0.9, 0.75, 0.5\}$ (from top to bottom). Simulations include 100 runs, each accounting for 10^7 slots.

comes

$$\frac{\lambda}{\lambda + \theta_{nb}} = e^{-\frac{\theta_{nb}}{\mu}} \ ,$$

which can be numerically solved; upper bounds on the waiting and response time distributions follow then immediately from Theorem 6.1.

6.2.2 Blocking Systems

Here, we consider a blocking FJ queueing system, i.e., the start of each job is synchronized amongst all servers. We maintain the i.i.d. assumptions on the interarrival times t_i and service times $x_{n,i}$. The waiting time and response time for the jth job can then be written as

$$w_j = \max\left\{0, \max_{1\le n\le j-1}\left\{\sum_{i=1}^{n}\max_k x_{k,j-i} - \sum_{i=1}^{n} t_{j-i}\right\}\right\}$$
$$r_j = \max_{0\le n\le j-1}\left\{\sum_{i=0}^{n}\max_k x_{k,j-i} - \sum_{i=1}^{n} t_{j-i}\right\} \ .$$

Note that the only difference to Eq. (6.1) and Eq. (6.2) is that the maximum over the number of servers now occurs inside the sum.

It is evident that the blocking system is more conservative than the

non-blocking system in the sense that the waiting time distribution of the non-blocking system is dominated by the waiting time distribution of the blocking system. Moreover, the stability region for the blocking system, given by $\mathbb{E}[t_1] > \mathbb{E}[\max_n x_{n,1}]$, is included in the stability region of the corresponding non-blocking system (i.e., $\mathbb{E}[t_1] > \mathbb{E}[x_{1,1}]$).

Analogously to Eq. (6.3), the steady-state waiting and response times w and r have now the representations

$$w =_{\mathcal{D}} \max_{n\geq 0}\left\{\sum_{i=1}^{n}\max_{k} x_{k,i} - \sum_{i=1}^{n} t_i\right\} \tag{6.16}$$

$$r =_{\mathcal{D}} \max_{n\geq 0}\left\{\sum_{i=0}^{n}\max_{k} x_{k,i} - \sum_{i=1}^{n} t_i\right\} \ . \tag{6.17}$$

The following theorem provides upper bounds on w and r:

Theorem 6.2. (Renewals, Blocking) *Given a FJ queueing system with K parallel blocking servers that is fed by renewal job arrivals with interarrivals t_j and i.i.d. task service times $x_{k,j}$. The distributions of the steady-state waiting and response times are bounded by*

$$\begin{aligned} \mathbb{P}[w \geq \sigma] &\leq e^{-\theta_b \sigma} \\ \mathbb{P}[r \geq \sigma] &\leq \mathbb{E}\left[e^{\theta_b x_{1,1}}\right] e^{-\theta_b \sigma} \ , \end{aligned} \tag{6.18}$$

where θ_b (with the subscript "b" standing for blocking) is defined by

$$\theta_b := \sup\left\{\theta > 0 \;\middle|\; \mathbb{E}\left[e^{\theta \max_k x_{k,1}}\right]\mathbb{E}\left[e^{-\theta t_1}\right] = 1\right\} \ . \tag{6.19}$$

Before giving the proof we note that, in general, Eq. (6.19) can be numerically solved. Moreover, for small values of K, θ_b can be analytically solved.

Proof. Consider the waiting time w. We proceed similarly as in the proof of Theorem 6.1. Letting $\mathcal{F}_k$ as above, we first prove that the process

$$y(n) = e^{\theta_b \sum_{i=1}^{n}(\max_k x_{k,i} - t_i)}$$

is a martingale w.r.t. $\mathcal{F}_n$ using a technique from [93]. We write

$$\begin{aligned}\mathbb{E}\left[y(n) \mid \mathcal{F}_{n-1}\right] &= \mathbb{E}\left[e^{\theta_b \sum_{i=1}^{n}(\max_k x_{k,i} - t_i)} \;\middle|\; \mathcal{F}_{n-1}\right] \\ &= e^{\theta_b \sum_{i=1}^{n-1}(\max_k x_{k,i} - t_i)} \mathbb{E}\left[e^{\theta_b(\max_k x_{k,1} - t_1)}\right] \\ &= e^{\theta_b \sum_{i=1}^{n-1}(\max_k x_{k,i} - t_i)} \\ &= y(n-1)\ ,\end{aligned}$$

where we used the independence and renewal assumptions for $x_{n,i}$ and t_i in the second line, and finally the condition on θ_b from Eq. (6.19). The proof for w completes as in the proof of Theorem 3.4 by applying the optional stopping theorem to the stopping time

$$N := \inf\left\{k \geq 0 \;\middle|\; \sum_{i=1}^{k}\left(\max_n x_{n,i} - t_i\right) \geq \sigma\right\}\ . \tag{6.20}$$

The proof for the response time r is analogous. □

Example 3: Exponentially distributed interarrival and service times

Consider interarrival and service times t_i and $x_{k,i}$ that are exponentially distributed with parameters λ and μ, respectively. In [119] it was shown that

$$\max_k L_k =_{\mathcal{D}} \sum_{k=1}^{K} \frac{L_k}{k}$$

for i.i.d. exponentially distributed random variables L_k, so that the stability condition $\mathbb{E}[t_1] > \mathbb{E}[\max_k x_{k,1}]$ becomes

$$\frac{1}{\lambda} > \frac{1}{\mu}\sum_{k=1}^{K}\frac{1}{k}\ . \tag{6.21}$$

By applying Theorem 6.2, the bounds on the steady-state waiting and response time distributions are

$$\mathbb{P}[w \geq \sigma] \leq e^{-\theta_b \sigma} \tag{6.22}$$

and

$$\mathbb{P}\left[r \geq \sigma\right] \leq \frac{\mu}{\mu - \theta_b} e^{-\theta_b \sigma} \ ,$$

where θ_b can be numerically solved from the condition

$$\prod_{k=1}^{K} \frac{k\mu}{k\mu - \theta_b} \frac{\lambda}{\lambda + \theta_b} = 1 \ .$$

For quick numerical illustrations we refer back to Figure 6.3.(b).

The interesting observation is that the stability condition from Eq. (6.21) depends on the number of servers K. In particular, as the right hand side grows in $\log K$, the system becomes unstable (i.e., waiting times are infinite) for sufficiently large K.

Example 4: Exponentially distributed interarrival and constant service times

If the service times are deterministic, i.e., $x_{k,i} = 1/\mu$ for all $i \geq 0$ and $k \in [1, K]$, the representations of w and r from Eq. (6.16) and Eq. (6.17) match their non-blocking counterparts from Eq. (6.3) and Eq. (6.4) and hence the corresponding stability regions and stochastic bounds are equal to those from Example 2.

6.3 FJ Systems with Non-renewal Input

In this section we consider the more realistic case of FJ queueing systems with non-renewal job arrivals. This model is particularly relevant given the empirical evidence that clusters running MapReduce exhibit various degrees of burstiness in the input [37, 85]. Moreover, numerous studies have demonstrated the burstiness of Internet traces, which can be regarded in particular as the input to multipath routing.

We model the interarrival times t_i using a Markov modulated process similar to the one from Subsection 3.2.2. Concretely, consider a two-state modulating Markov chain c_k, as depicted in Figure 6.4, with a transition matrix T

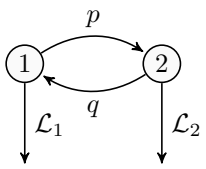


Figure 6.4: Markov modulating chain c_k for the job interarrival times.

given by

$$T = \begin{pmatrix} 1-p & p \\ q & 1-q \end{pmatrix} , \tag{6.23}$$

for some values $0 < p, q < 1$. In state $i \in \{1, 2\}$ the interarrival times are given by i.i.d. random variables L_i with distribution $\mathcal{L}_i$. We assume that L_1 is stochastically smaller than L_2, i.e.,

$$\mathbb{P}\left[L_1 \geq t\right] \leq \mathbb{P}\left[L_2 \geq t\right] , \tag{6.24}$$

for any $t \geq 0$. Additionally, we assume that the Markov chain c_k satisfies the same burstiness condition as in Eq. (3.16), namely

$$p < 1 - q , \tag{6.25}$$

i.e., the probability of jumping to a different state is less than the probability of staying in the same state.

Analogously to Eq. (3.13), the exponential transform of the transition matrix T is defined as

$$T_\theta := \begin{pmatrix} (1-p)\,\mathbb{E}\left[e^{-\theta L_1}\right] & p\,\mathbb{E}\left[e^{-\theta L_2}\right] \\ q\,\mathbb{E}\left[e^{-\theta L_1}\right] & (1-q)\,\mathbb{E}\left[e^{-\theta L_2}\right] \end{pmatrix} ,$$

for some $\theta > 0$. Let $\Lambda(\theta)$ denote the maximal positive eigenvalue of T_θ, and the vector $h = (h(1), h(2))$ denote a corresponding eigenvector. By the Perron-Frobenius Theorem, $\Lambda(\theta)$ is equal to the spectral radius of T_θ such that h can

be chosen with strictly positive components.

As in the case of renewal arrivals, we will next analyze both non-blocking and blocking FJ systems.

6.3.1 Non-Blocking Systems

We first analyze a non-blocking FJ system fed with arrivals that are modulated by a stationary Markov chain as in Figure 6.4. We assume that the task service times $x_{k,j}$ are i.i.d. and that the families $\{t_i\}$ and $\{x_{k,i}\}$ are independent. Note that both the definition of w_j from Eq. (6.1) and the representation of the steady-state waiting time w in Eq. (6.3) remain valid, due to stationarity and reversibility; the same holds for the response times.

The next theorem provides upper bounds on the steady-state waiting and response time distributions in the non-blocking scenario with Markov modulated interarrivals.

Theorem 6.3. (Non-Renewals, Non-Blocking) *Given a FJ queueing system with K parallel non-blocking servers, Markov modulated job interarrivals t_j according to the Markov chain depicted in Figure 6.4 with transition matrix Eq. (6.23), and i.i.d. task service times $x_{k,j}$. The steady-state waiting and response time distributions are bounded by*

$$\mathbb{P}[w \geq \sigma] \leq Ke^{-\theta_{nb}\sigma} \tag{6.26}$$

$$\mathbb{P}[r \geq \sigma] \leq K\mathbb{E}\left[e^{\theta_{nb}x_{1,1}}\right] e^{-\theta_{nb}\sigma} \ , \tag{6.27}$$

where θ_{nb} is defined by

$$\theta_{nb} := \sup\left\{\theta > 0 \;\middle|\; \mathbb{E}\left[e^{\theta x_{1,1}}\right] \Lambda(\theta) = 1\right\} \ .$$

Proof. Consider the filtration

$$\mathcal{F}_n := \sigma\left\{x_{k,m}, t_m, c_m \,|\, m \leq n, k \in [1,K]\right\} \ ,$$

that includes information about the state c_k of the Markov chain. Now, we construct the process $z(n)$ as

$$\begin{aligned} z(n) &= h(c_n) e^{\theta_{nb}\left(\max_k \sum_{i=1}^{n} x_{k,i} - \sum_{i=1}^{n} t_i\right)} \\ &= \left(e^{\theta_{nb}\left(\max_k \sum_{i=1}^{n} x_{k,i} - nD\right)}\right) \left(h(c_n) e^{\theta_{nb}\left(nD - \sum_{i=1}^{n} t_i\right)}\right) \end{aligned} \tag{6.28}$$

with the deterministic parameter

$$D := \theta_{nb}^{-1} \log\left(\mathbb{E}\left[e^{\theta_{nb} x_{1,1}}\right]\right) \ .$$

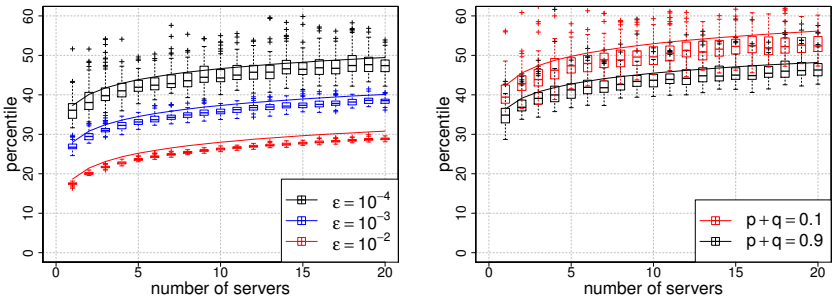
Note the similarity of $z(n)$ to Eq. (6.9) except for the additional function h.

Next we show that both terms of Eq. (6.28) are submartingales. In the first step we note that by the definition of D:

$$\mathbb{E}\left[e^{\theta_{nb}\left(\sum_{i=1}^{n} x_{k,i} - kD\right)} \,\middle|\, \mathcal{F}_{n-1}\right] = e^{\theta_{nb}\left(\sum_{i=1}^{n-1} x_{k,i} - (n-1)D\right)} \ ,$$

hence, following the line of argument in Eq. (6.10) the left factor of Eq. (6.28), which accounts for the additional $\max_k$, is a submartingale. The second term follows as in the proof of the service-martingale in Lemma 4.8. As the process $z(n)$ is a product of two independent submartingales, it is a submartingale itself w.r.t. $\mathcal{F}_n$. We use the stopping time N defined in Eq. (6.11) and apply the optional stopping theorem. On the one hand we can write for every $k \in \mathbb{N}$

$$\begin{aligned} \mathbb{E}\left[z(n)\right] &\geq \mathbb{E}\left[z(N \wedge n)\right] \\ &\geq \mathbb{E}\left[z(N \wedge n) 1_{N<n}\right] \\ &= \mathbb{E}\left[\max_k h(c_N) e^{\theta_{nb}\left(\sum_{i=1}^{N} x_{k,i} - \sum_{i=1}^{N} t_i\right)} 1_{N<n}\right] \\ &\geq e^{\theta_{nb}\sigma} \mathbb{E}\left[h(c_N) 1_{N<n}\right] \\ &= e^{\theta_{nb}\sigma} \mathbb{E}\left[h(c_N) \mid N < n\right] \mathbb{P}\left[N < n\right] \ . \end{aligned} \tag{6.29}$$



(a) Impact of ε (b) Impact of the burstiness factor $p+q$

Figure 6.5: The $\mathcal{O}(\log K)$ scaling of waiting time percentiles w^ε for Markov modulated input (the non-blocking case Eq. (6.26)). The system parameters are $\mu = 1$, $\lambda_2 = 0.9$, $\rho = 0.75$ (in both (a) and (b)) $p = 0.1$, $q = 0.4$ (in (a)), three violation probabilities ε (in (a)), $\varepsilon = 10^{-4}$ and only two burstiness parameters $p+q$ (in (b)) (for visual convenience). Simulations include 100 runs, each accounting for 10^7 slots.

On the other hand we can upper bound the term

$$\begin{aligned}\mathbb{E}\left[z(n)\right] &= \mathbb{E}\left[\max_k e^{\theta_{nb}\left(\sum_{i=1}^n x_{k,i} - nD\right)}\right]\mathbb{E}\left[h(c_n)e^{\theta_{nb}\left(nD-\sum_{i=1}^n t_i\right)}\right] \\ &\leq K\mathbb{E}\left[h(c_1)\right] \ .\end{aligned}$$

Letting $n \to \infty$ in Eq. (6.29) leads to

$$\mathbb{P}\left[N < \infty\right] \leq \frac{\mathbb{E}\left[h(c_1)\right]}{\mathbb{E}\left[h(c_N) \mid N < \infty\right]} K e^{-\theta_{nb}\sigma} \ . \tag{6.30}$$

In Lemma 6.4 below it is shown that the distribution of the random variable $(c_N \mid N < n)$ is stochastically smaller than the stationary distribution of the Markov chain. Given the burstiness condition in Eq. (6.25) and that the function h is monotonically decreasing [27], we can further upper bound the prefactor in Eq. (6.30) as

$$\frac{\mathbb{E}\left[h(c_1)\right]}{\mathbb{E}\left[h(c_N) \mid N < \infty\right]} \leq 1 \ ,$$

which completes the proof. The proof for the response time r is analogous. □

The stochastic ordering used in the proof of Lemma 6.3 is given by the

following Lemma:

Lemma 6.4. *Let c_n be the Markov chain from Figure 6.4 and N be the stopping time from Eq. (6.11). Then the distribution of $(c_N \mid N < \infty)$ is stochastically smaller than the steady-state distribution of c_n, i.e.,*

$$\mathbb{P}\left[c_N = 2 \mid N < \infty\right] \leq \mathbb{P}\left[c_1 = 2\right] \ ,$$

or, equivalently,

$$\mathbb{E}\left[h(c_N) \mid N < \infty\right] \geq \mathbb{E}\left[h(c_n)\right] \ ,$$

for all monotonically decreasing functions h on $\{1, 2\}$.

Proof. Using Bayes' rule and the stationarity of the process c_n, it holds:

$$\begin{aligned}
\mathbb{P}\left[c_N = 2 \mid N < \infty\right] &= \mathbb{P}\left[N < \infty\right]^{-1} \mathbb{P}\left[c_N = 2,\ N < \infty\right] \\
&= \mathbb{P}\left[N < \infty\right]^{-1} \sum_{n=1}^{\infty} \mathbb{P}\left[c_N = 2,\ N = n\right] \\
&= \mathbb{P}\left[N < \infty\right]^{-1} \mathbb{P}\left[c_1 = 2\right] \sum_{n=1}^{\infty} \mathbb{P}\left[N = n \mid c_n = 2\right]
\end{aligned}$$

Since L_1 is stochastically smaller than L_2 (see Eq. (6.24)), we have for any $n \geq 1$

$$\begin{aligned}
\mathbb{P}[N = n \mid c_n = 2] &= \mathbb{P}\left[t_n \leq \max_k \sum_{i=1}^{n} x_{k,i} - \sum_{i=1}^{n-1} t_i - \sigma, \max_k \sum_{i=1}^{n-1} (x_{k,i} - t_i) < \sigma \,\middle|\, c_n = 2\right] \\
&\leq \mathbb{P}\left[t_n \leq \max_k \sum_{i=1}^{n} x_{k,i} - \sum_{i=1}^{n-1} t_i - \sigma, \max_k \sum_{i=1}^{n-1} (x_{k,i} - t_i) < \sigma\right] \\
&= \mathbb{P}\left[N = n\right] \ .
\end{aligned}$$

Hence $\mathbb{P}\left[c_N = 2 \mid N < \infty\right] \leq \mathbb{P}\left[c_1 = 2\right]$, which completes the proof. □

Remark 6.5. *Note that, if the burstiness condition Eq. (6.25) is not fulfilled then we can still upper bound the prefactor in Eq. (6.30) using the trivial upper bound*

$$\frac{\mathbb{E}\left[h(c_1)\right]}{\mathbb{E}\left[h(c_N) \mid N < \infty\right]} \leq \frac{\mathbb{E}\left[h(c_1)\right]}{\min_n h(c_n)} \ .$$

Figure 6.5 displays the bounds on the waiting time percentiles w^{ε}, for various violation probabilities ε, in the FJ system with non-renewal input. The bounds closely match the corresponding simulation results, shown as box-plots, while also exhibiting the $\mathcal{O}(\log K)$ scaling behavior (which can be also derived from both Eq. (6.26) and Eq. (6.27), as in Section 6.2).

6.3.2 Blocking Systems

Now we turn to the blocking variant of the FJ system that is fed by the same non-renewal arrivals as in the previous section. We consider exponential distributions $\mathcal{L}_m$ for $m \in [1, 2]$. The main result is:

Theorem 6.6. (Non-Renewals, Blocking) *Given a FJ system with K blocking servers, Markov modulated job interarrivals t_j, and i.i.d. task service times $x_{k,j}$. The steady-state waiting and response time distributions are bounded by*

$$\begin{aligned} \mathbb{P}[w \geq \sigma] &\leq e^{-\theta_b \sigma} \\ \mathbb{P}[r \geq \sigma] &\leq \mathbb{E}\left[e^{\theta_b x_{1,1}}\right] e^{-\theta_b \sigma} \ , \end{aligned} \tag{6.31}$$

where θ_b is defined by

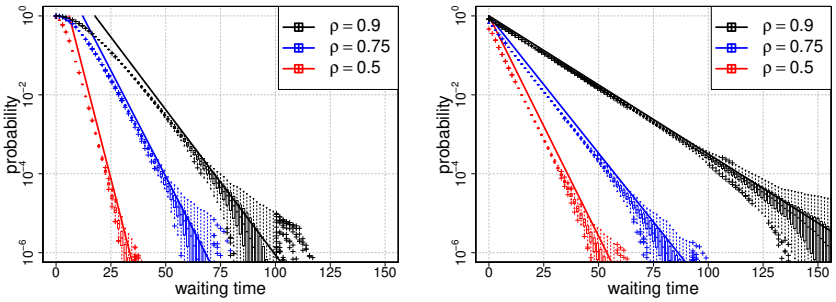
$$\theta_b := \sup\left\{\theta > 0 \;\middle|\; \mathbb{E}\left[e^{\theta \max_k x_{k,1}}\right] \Lambda(\theta) = 1\right\} \ .$$

Again, the positive solution for θ_b is guaranteed under the stronger stability condition $\mathbb{E}[t_1] > \mathbb{E}[\max_n x_{n,1}]$ and the Perron-Frobenius Theorem.

Proof. Let $D := \theta_b^{-1} \log \mathbb{E}\left[e^{\theta_b \max_k x_{k,1}}\right]$ and define the process y by:

$$\begin{aligned} y(n) &= h(c_n) e^{\theta_b \left(\sum_{i=1}^{n} \max_k x_{k,i} - \sum_{i=1}^{n} t_i\right)} \\ &= \left(e^{\theta_b \left(\sum_{i=1}^{n} \max_k x_{k,i} - nD\right)}\right)\left(h(c_n) e^{\theta_b \left(nD - \sum_{i=1}^{n} t_i\right)}\right) \ . \end{aligned}$$

Similarly to the proofs of Theorem 6.2 and Theorem 6.3 one shows that both the first and second factor of y are martingales, and hence y is a martingale.



(a) Non-Blocking (b) Blocking

Figure 6.6: Bounds on the waiting time distributions vs. simulations (non-renewal input): (a) the non-blocking case Eq. (6.26) and (b) the blocking case Eq. (6.31). The parameters are $K = 20, \mu = 1, p = 0.1, q = 0.4, \lambda_1 \in \{0.4, 0.72, 0.72\}$ and $\lambda_2 \in \{0.9, 0.9, 1.62\}$ leading to utilizations $\rho \in \{0.5, 0.75, 0.9\}$. Simulations include 100 runs, each accounting for 10^7 slots.

We use the stopping time N in Eq. (6.20) and write

$$\begin{aligned}
\mathbb{E}\left[h(c_1)\right] &= \mathbb{E}\left[y(0)\right] \\
&\geq \mathbb{E}\left[y(N \wedge n)\right] \\
&\geq \mathbb{E}\left[y(N \wedge n) 1_{N<n}\right] \\
&= \mathbb{E}\left[e^{\theta_b(\sum_{i=1}^{N} \max_k x_{k,i} - \sum_{i=1}^{N} t_i)} h(c_N) 1_{N<n}\right] \\
&\geq e^{\theta_b \sigma} \mathbb{E}\left[h(c_N) \mid N < \infty\right] \mathbb{P}\left[N < n\right] \ .
\end{aligned}$$

Taking $n \to \infty$ we obtain the bound

$$\mathbb{P}\left[N < \infty\right] \leq \frac{\mathbb{E}\left[h(c_1)\right]}{\mathbb{E}\left[h(c_K) \mid K < \infty\right]} e^{-\theta_b \sigma} \leq e^{-\theta_b \sigma} \ ,$$

where we used Lemma 6.4 for the last inequality. The proof for r is analogous.

□

A close comparison of the waiting time bound in the non-renewal case Eq. (6.31) to the corresponding bound in the renewal case Eq. (6.18) reveals that the decay factors θ_b depend on similar conditions, whereby the MGF of the interarrival times in Eq. (6.18) is replaced by the maximal positive eigenvalue

of the modulating Markov chain in Eq. (6.31). Moreover, given the ergodicity of the underlying Markov chain, the blocking system with non-renewal input is subject to the same degrading stability region (in $\log K$) as in the renewal case (recall Eq. (6.21)).

For quick numerical illustrations of the tightness of the bounds on the waiting time distributions in both the non-blocking and blocking cases we refer to Figure 6.6.

So far we have contributed stochastic bounds on the steady-state waiting and response time distributions in FJ systems fed with either renewal and non-renewal job arrivals. The key technical insight was that the stochastic bounds in the non-blocking model grow as $\mathcal{O}(\log K)$ in the number of parallel servers K under non-renewal arrivals, which extends a known result for renewal arrivals [109, 11]. The same fundamental factor of $\log K$ was shown to drive the stability region in the blocking model. A concrete application follows next.

6.4 Partial Mapping

In this section we consider FJ queueing systems where jobs are mapped to a subset of $H \leq K$ servers. This model captures a crucial aspect of the operation of parallel systems, i.e., the amount of resources provided to some job is not necessarily the entire amount of resources available. This corresponds, for example, to batch systems, where servers are grouped into resource pools and incoming jobs are assigned to one such pool. In general, partial mapping provides a basis for service differentiation and isolation within parallel systems. In the following we regard two contrasting types of partial mapping, i.e., a rigid round-robin mapping and a random partial mapping of jobs to $H \leq K$ servers. The subsequent analysis of the fan-out ratio H/K on the system performance provides a reference for dimensioning such server pools. In the following, we restrict the exposition to the more interesting case of non-blocking servers since most of the derivations rely on results from Sections 6.2 and 6.3.

6.4.1 Round-robin Partial Mapping, Dyadic System

We consider a dyadic FJ system where the number of servers is given as $K = 2^W$ (with $W \geq 1$) and a job is split into $H = 2^V$ tasks (with $1 \leq V \leq W$). The assignment of tasks to servers follows a round-robin scheme such that the first job is assigned to servers $1, \ldots, H$, the second to the servers $H+1, \ldots, 2H$, etc.

In the following, we consider job arrivals as renewal processes similar to Section 6.2. For the analysis it is sufficient to look only at an equivalent "FJ subsystem" that consists of only H servers and adjust the job interarrival times $\bar{t}_n$ to that system accordingly:

$$\bar{t}_n := \sum_{i=1}^{2^{(W-V)}} t_{(n-1)2^{(W-V)}+i} \ .$$

Note that for the extremal case $V = W$ we recover the scenario from Section 6.2, i.e., $\bar{t}_n = t_n$.

The Laplace transform of the job interarrival times $\bar{t}_n$ to one subsystem is obtained directly from the Laplace transform of the original job interarrival times t_n and the number of subsystems:

$$\mathbb{E}\left[e^{-\theta \bar{t}_1}\right] = \mathbb{E}\left[e^{-\theta t_1}\right]^{2^{W-V}} = \mathbb{E}\left[e^{-\theta t_1}\right]^{\frac{K}{H}} \ .$$

The steady-state waiting time distribution now has the following representation:

$$w =_{\mathcal{D}} \max_{n \geq 0} \left\{ \max_{1 \leq k \leq H} \left\{ \sum_{i=1}^{n} x_{k,i} - \sum_{i=1}^{n} \bar{t}_i \right\} \right\} \tag{6.32}$$

and the response time:

$$r =_{\mathcal{D}} \max_{n \geq 0} \left\{ \max_{1 \leq k \leq H} \left\{ \sum_{i=0}^{n} x_{k,i} - \sum_{i=1}^{n} \bar{t}_i \right\} \right\} \ . \tag{6.33}$$

The next theorem provides upper bounds on the steady-state waiting and response time distributions in the non-blocking scenario with partial round-robin mapping and renewal interarrivals.

Theorem 6.7. (ROUND-ROBIN MAPPING, RENEWALS, NON-BLOCKING) *Given a FJ queueing system with $K = 2^W$ non-blocking servers and partial round-robin mapping of jobs to $H = 2^V$ servers with $1 \leq V \leq W$. The system is fed by renewal job arrivals with interarrivals t_j. If the input job size is normalized such that the MGF of the task service time is given as $\mathbb{E}\left[e^{\theta x_{k,i}/H}\right]$, with the service times $x_{k,i}$ being i.i.d., then the steady-state waiting and response times w and r are bounded by*

$$\mathbb{P}\left[w \geq \sigma\right] \leq He^{-\theta\sigma} \ ,$$
$$\mathbb{P}\left[r \geq \sigma\right] \leq H\mathbb{E}\left[e^{\theta x_{1,1}}\right] e^{-\theta\sigma} \ ,$$

where θ is defined by

$$\theta := \sup\left\{\theta > 0 \;\middle|\; \mathbb{E}\left[e^{\theta x_{1,1}/H}\right] \mathbb{E}\left[e^{-\theta t_1}\right]^{\frac{N}{H}} = 1\right\} \ . \tag{6.34}$$

Proof. The proof goes along the same arguments of the proof of Theorem 6.1, however, with modified MGF and Laplace transform for the task service times $x_{k,i}$ and the job interarrival times t_i, respectively. □

The rationale behind the normalization of the input job size such that the MGF of the task service time is given as $\mathbb{E}\left[e^{\theta x_{k,i}/H}\right]$ is to compare different fan-out factors H such that the mean task service time is $\mathbb{E}\left[x\right]/H$.

Example 5: Exponentially distributed interarrival and service times

In the case of exponentially distributed interarrival times with parameter λ the job interarrival times at one subsystem have an Erlang $E_{\frac{K}{H}}$ distribution. We assume the tasks are exponentially distributed with a mean $1/H\mu$. The condition Eq. (6.34) from Theorem 6.7 becomes

$$\left(\frac{H\mu}{H\mu - \theta}\right)\left(\frac{\lambda}{\lambda + \theta}\right)^{\frac{N}{H}} = 1 \ . \tag{6.35}$$

In Figure 6.7 we show simulation box-plots as well as corresponding

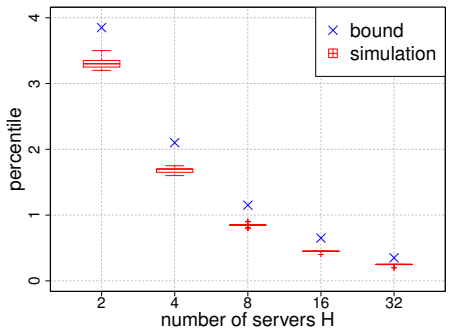


Figure 6.7: Round-robin partial mapping: Bound on the waiting time percentile w^{ε} for renewal arrivals and increasing number of servers (fan-out) H. The system parameters are $\mu = 1, \lambda = 0.75, \varepsilon = 10^{-3}$ and the overall number of servers is $K = 2^8$.

bounds on the waiting time percentile w^{ε} from Theorem 6.7 for an increasing number of fan-out servers H. Observe the diminishing gain in terms of waiting time reduction with increasing the server fan-out.

6.4.2 Random Partial Mapping

Here, we consider a system that randomly maps a job to H out of K available servers based on a uniform distribution over the set $\{A \subseteq \{1, \ldots, K\} \mid |A| = H\}$ of server combinations with cardinality H. We bound the job waiting and response time in this system using the following abstraction which considers the probability of assigning a task to a specific server. Note that the probability for a task dedicated to a certain server is given by $p_d = H/K$. Now, if we focus on only one server of this FJ system, the task service times at that server can be represented by the compound distribution

$$\bar{x}_{k,i} = \begin{cases} x_{k,i} & \text{with probability} \quad p_d \\ 0 & \text{with probability} \quad 1 - p_d \ , \end{cases} \tag{6.36}$$

since a job that is not assigned to this server can be considered to have a service time equal to 0. Hence, one server of this FJ system with random partial mapping can be modelled as if it is part of a FJ system with full mapping as in

Section 6.2, but with the modified service times $\bar{x}_{k,i}$. Note that due to the selection of the subset with fixed cardinality H, the $(\bar{x}_{k,i})_k$ are no longer independent. Their MGF can be computed as:

$$\mathbb{E}\left[e^{\theta \bar{x}_{k,i}}\right] = (1 - p_d) + p_d \mathbb{E}\left[e^{\theta x_{k,i}}\right] \ .$$

The representations for the waiting and response time, respectively, become

$$w =_{\mathcal{D}} \max_{n \geq 0} \left\{ \max_{1 \leq k \leq H} \left\{ \sum_{i=1}^{n} \bar{x}_{k,i} - \sum_{i=1}^{n} t_i \right\} \right\} \ , \tag{6.37}$$

and

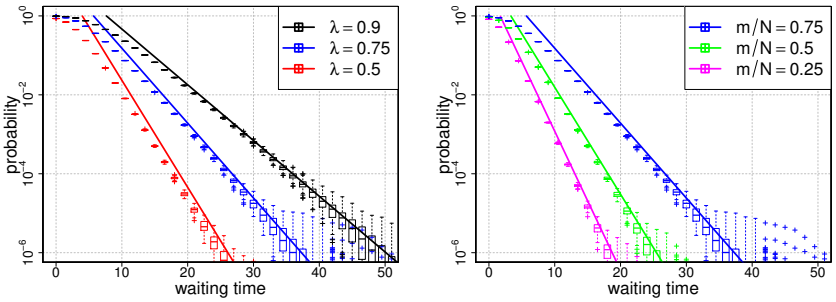
$$r =_{\mathcal{D}} \max_{n \geq 0} \left\{ \max_{1 \leq k \leq H} \left\{ x_{k,0} + \sum_{i=1}^{n} \bar{x}_{k,i} - \sum_{i=1}^{n} t_i \right\} \right\} \ . \tag{6.38}$$

Note the asymmetry for the response time in (6.38). For $i \geq 1$ we consider the modified service times $\bar{x}_{k,i}$ as the corresponding server is only selected with probability p_d. In turn, for $i = 0$, we need to consider the unmodified service time $x_{0,i}$ as we only look at those servers which have been selected for mapping.

The following theorems provide upper bounds on the steady-state waiting and response time distributions in the non-blocking scenarios with partial random mapping for renewal and Markov-modulated interarrivals, respectively.

Theorem 6.8. (Random Mapping, Renewals, Non-Blocking) *Given a FJ queueing system with K servers and random partial mapping of jobs to $H \leq K$ servers based on a uniform distribution over the set $\{A \subseteq \{1, \ldots, K\} \mid |A| = H\}$ of server combinations with cardinality H. The system is fed with renewal job arrivals. If the task service times $x_{k,j}$ are i.i.d., then the steady-state waiting and response times w and r are bounded by*

$$\mathbb{P}\left[w \geq \sigma\right] \leq H e^{-\theta^* \sigma} \ ,$$
$$\mathbb{P}\left[r \geq \sigma\right] \leq H \mathbb{E}\left[e^{\theta^* x_{1,1}}\right] e^{-\theta^* \sigma} \ ,$$



(a) Impact of λ (b) Impact of the fan-out ratio H/N

Figure 6.8: Bounds on the waiting time distributions vs. simulation box-plots for renewal input with random server mapping. The parameters are $K = 16, \mu = 1$. (a) Here, we fix the fan-out ratio to $H = 12$ and change the job arrival rate $\lambda \in \{0.5, 0.75, 0.9\}$ while in (b) we fix the arrival rate to $\lambda = 0.75$ and vary the fan-out ratio $H/K \in \{0.25, 0.5, 0.75\}$. Simulations include 100 runs, each accounting for 10^6 slots.

where θ is the solution of

$$\theta^* := \left\{\theta > 0 \;\middle|\; \left((1-p_d) + p_d \mathbb{E}\left[e^{\theta x_{n,i}}\right]\right) \mathbb{E}\left[e^{-\theta t_1}\right] = 1\right\} \ . \tag{6.39}$$

Proof. The proof goes along similar steps as for Theorem 6.7, however, using the process

$$z_k(n) = e^{\theta^* \sum_{i=1}^{n} (\bar{x}_{k,i} - t_i)}$$

which is a martingale for each $k \leq K$ under the criterion (6.39) on θ^*. □

Note that the observed correlation of the $(\bar{x}_{k,i})_k$ does not cause any problems in the proof as the submartingale construction does not require independence. In fact, even the processes $z_k(n)$ from the proof of Theorem 6.1 were not independent due to the common interarrival times t_i.

Figure 6.8 shows a numerical illustration of the tightness of the bounds on the waiting time distribution from Theorem 6.8. The illustrated results are for the example of exponentially distributed interarrival and service times with parameters λ and μ, respectively.

By combining the above consideration of the compound service time distribution with the results from Section 6.3, one can extend the analysis of random partial mapping to the case of non-renewal input.

Theorem 6.9. (Random Mapping, Non-Renewals, Non-Blocking) *Given a FJ queueing system with K parallel non-blocking servers, Markov modulated job interarrivals t_j as in Section 6.3, and task service times $\bar{x}_{k,i}$ that are described by Eq. (6.36). Jobs are randomly mapped to servers according to a uniform distribution over the set of server combinations with cardinality H. The steady-state waiting and response time distributions are bounded by*

$$\begin{aligned} \mathbb{P}\left[w \geq \sigma\right] &\leq He^{-\theta^* \sigma} \ , \\ \mathbb{P}\left[r \geq \sigma\right] &\leq H\mathbb{E}\left[e^{\theta^* x_{1,1}}\right] e^{-\theta^* \sigma} \ , \end{aligned}$$

where θ^* is defined by

$$\theta^* := \sup\left\{\theta > 0 \;\middle|\; \left((1-p_d) + p_d \mathbb{E}\left[e^{\theta x_{1,1}}\right]\right) \Lambda(\theta) = 1\right\} \ .$$

(Recall that $\Lambda(\theta)$ was defined as a spectral radius of T_θ in Section 6.3).

Proof. The proof follows analogously to the proof of Theorem 6.3 with the difference that $x_{k,i}$ is replaced by $\bar{x}_{k,i}$ and K by H, respectively. □

Remark 6.10. *Random number of servers H: One variation of the system that is considered in Section 6.4.2 is a random mapping of arriving jobs to a random number of servers $1 \leq H \leq N$ based on a uniform distribution over the power set $2^A \setminus \{\varnothing\}$ with $A = \{1, \ldots, N\}$. In this case the steady state waiting and response times are bounded by*

$$\begin{aligned} \mathbb{P}\left[w \geq \sigma\right] &\leq Ke^{-\theta^* \sigma} \ , \\ \mathbb{P}\left[r \geq \sigma\right] &\leq K\mathbb{E}\left[e^{\theta^* x_{1,1}}\right] e^{-\theta^* \sigma} \ , \end{aligned}$$

where θ^* is the solution of (6.39) with $p_d = 2^{N-1}/(2^N - 1)$.

6.5 Application to Window-based Protocols over Multipath Routing

In this section we slightly adapt and use the non-blocking FJ queueing system from Section 6.2.1 to analyze the performance of a *generic* window-based transmission protocol over multipath routing. While this problem has attracted much interest lately with the emergence of multipath TCP [117], it is subject to a major difficulty due to the likely overtaking of packets on different paths. Consequently, packets have to additionally wait for a *resequencing delay*, which directly corresponds to the synchronization constraint in FJ systems. We note that the employed FJ non-blocking model is subject to a convenient simplification, i.e., each path is modelled by a single server/queue only.

As depicted in Figure 6.9, we consider an arrival flow containing l batches of K packets, with $l \in \mathbb{N}$, at the fork node A. In practice, a *packet* as denoted here may represent an entire train of consecutive datagrams. The incoming packets are sent over multiple paths to the destination node B, where they need to be eventually reordered. We assume that the batch size corresponds to the transmission window size of the protocol, such that one packet traverses a single path only. For example, the first path transmits the packets $\{1, K+1, 2K+1, \dots\}$, i.e., packets are distributed in a round-robin fashion over the K paths. We also assume that packets on each path are delivered in a (locally-) FIFO order, i.e., there is no overtaking on the same path.

In analogy to Section 6.2.1, we consider a batch waiting until its last packet starts being transmitted. When the transmission of the last packet of batch j begins, the previous batch has already been received, i.e., all packets of the batch $j-1$ are *in order* at node B.

We are interested in the response times of the batches, which are upper bounded by the largest response time of the packets therein. The arrival time of a batch is defined as the latest arrival time of the packets therein, i.e., when the batch is entirely received. Formally, the response time of batch

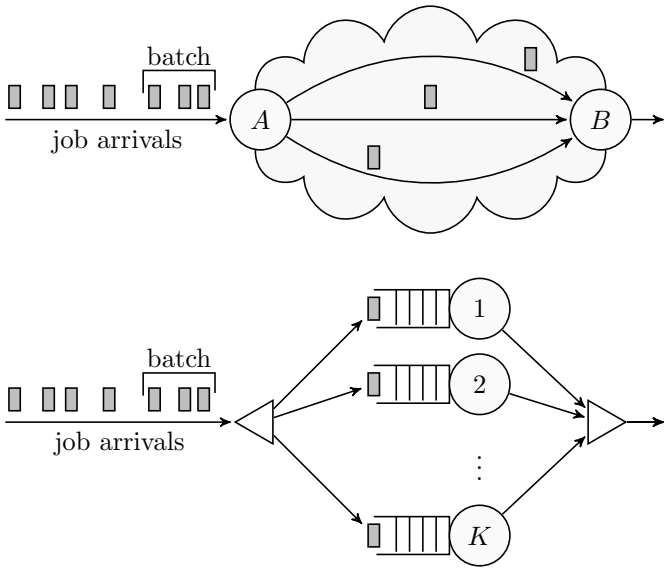


Figure 6.9: A schematic description of the window-based transmission over multipath routing; each path is modelled as a single server/queue.

$j \in \{lK + 1 \mid l \in \mathbb{N}\}$ can be given by slightly modifying Eq. (6.2), i.e.,

$$r_j = \max_{0 \le n \le j-1} \left\{ \max_k \left\{ \sum_{i=0}^{n} x_{k,j-i} - \sum_{i=1}^{n} t_{k,j-i} \right\} \right\} .$$

The corresponding steady-state response time has the modified representation

$$r =_{\mathcal{D}} \max_{n \ge 0} \left\{ \max_k \left\{ \sum_{i=0}^{n} x_{k,i} - \sum_{i=1}^{n} t_{k,i} \right\} \right\} .$$

The modifications account for the fact that the packets of each batch are asynchronously transmitted on the corresponding paths (instead, in the basic FJ systems, the tasks of each job are simultaneously mapped). In terms of notations, the $t_{k,i}$'s now denote the interarrival times of the packets transmitted over the same path k, whereas $x_{k,i}$'s are i.i.d. and denote the transmission time of packet i over path k; as an example, when the arrival flow at node A is Poisson, $t_{k,i}$ has an Erlang E_K distribution for all k and i.

We next analyze the performance of the considered multipath routing for both renewal and non-renewal input.

Renewal Arrivals

Consider first the scenario with renewal interarrival times. Similarly to Section 6.2.1 we bound the distribution of the steady-state response time r using a submartingale in the time domain $j \in \{lK + 1 \mid l \in \mathbb{N}\}$. Following the same steps as in Theorem 6.1, the process

$$z_k(n) = e^{\theta^* \left(\sum_{i=0}^{n} x_{k,i} - \sum_{i=1}^{n} t_{k,i}\right)}$$

is a martingale with

$$\theta^* := \sup \left\{ \theta > 0 \;\middle|\; \mathbb{E}\left[e^{\theta x_{1,1}}\right] \mathbb{E}\left[e^{-\theta t_{1,1}}\right] = 1 \right\} \ ,$$

where we used the filtration

$$\mathcal{F}_n := \sigma \left\{ x_{k,m}, t_{k,m} \mid m \leq n, k \in [1, K] \right\} \ .$$

Note that $\mathbb{E}\left[e^{-\theta t_{1,1}}\right]$ denotes the Laplace transform of the interarrival times of packets transmitted over each path. The proof that $\max_k z_k(n)$ is a submartingale follows a similar argument as in Eq. (6.10). Hence, we can bound the distribution of the steady-state response time as

$$\mathbb{P}\left[r \geq \sigma\right] \quad \leq \quad K\mathbb{E}\left[e^{\theta^* x_{1,1}}\right] e^{-\theta^* \sigma} \ , \tag{6.40}$$

with the condition on θ^* from above.

Non-Renewal Arrivals

Next, consider a scenario with non-renewal interarrival times t_i of the packets arriving at the fork node A in Figure 6.9, as described in Section 6.3. On every path $k \in [1, K]$ the interarrivals are given by a sub-chain $(c_{k,n})_n$ that is driven by the K-step transition matrix $T^K = (\alpha_{i,j})_{i,j}$ for T given in Eq. (6.23). Similarly as in the proof of Theorem 6.3, we will use an exponential transform $(T^K)_\theta$ of

the transition matrix that describes each path k, i.e.,

$$(T^K)_\theta := \begin{pmatrix} \alpha_{1,1}\beta_1 & \alpha_{1,2}\beta_2 \\ \alpha_{2,1}\beta_1 & \alpha_{2,2}\beta_2 \end{pmatrix} ,$$

with $\alpha_{i,j}$ defined above and β_1, β_2 being the elements of a vector β of conditional Laplace transforms of K consecutive interarrival times t_i. The vector β is given by

$$\beta := \begin{pmatrix} \beta_1 \\ \beta_2 \end{pmatrix} = \begin{pmatrix} \mathbb{E}\left[e^{-\theta^* \sum_{i=1}^K t_i} \;\middle|\; c_1 = 1\right] \\ \\ \mathbb{E}\left[e^{-\theta^* \sum_{i=1}^K t_i} \;\middle|\; c_1 = 2\right] \end{pmatrix} ,$$

and can be computed given the transition matrix T from Eq. (6.23) via an exponential row transform [35, Example 7.2.7] denoted by

$$\tilde{T}_{\theta^*} := \begin{pmatrix} (1-p)\mathbb{E}\left[e^{-\theta^* L_1}\right] & p\mathbb{E}\left[e^{-\theta^* L_1}\right] \\ \\ q\mathbb{E}\left[e^{-\theta^* L_2}\right] & (1-q)\mathbb{E}\left[e^{-\theta^* L_2}\right] \end{pmatrix} ,$$

yielding $\beta = (\tilde{T}_{\theta^*})^K \begin{pmatrix} 1 \\ 1 \end{pmatrix}$.

Denote $\Lambda(\theta^*)$ and $h = (h(1), h(2))$ as the maximal positive eigenvalue of the matrix $(T^K)_{\theta^*}$ and the corresponding right eigenvector, respectively. Mimicking the proof of Theorem 6.3, one can show for every path k that the process

$$z_k(n) = h(c_{k,n}) e^{\theta^* \left(\sum_{i=0}^n x_{k,i} - \sum_{i=1}^n t_{k,i}\right)}$$

is a martingale with the definition

$$\theta^* := \sup\left\{\theta > 0 \;\middle|\; \mathbb{E}\left[e^{\theta x_{1,1}}\right] \Lambda(\theta) = 1\right\} . \tag{6.41}$$

Given the martingale representation of the processes $z_k(n)$ for every path

k, the process

$$z(n) = \max_k z_k(n)$$

is a submartingale following the line of argument in Eq. (6.10). We can now use Eq. (6.30) and the remark at the end of Section 6.3.1 to bound the distribution of the steady-state response time r as

$$\mathbb{P}\left[r \geq \sigma\right] \leq \frac{\mathbb{E}\left[h(c_{1,1})\right]}{h(2)} K \mathbb{E}\left[e^{\theta^* x_{1,1}}\right] e^{-\theta^* \sigma} \ , \tag{6.42}$$

where we also used that h is monotonically decreasing and θ^* as defined in Eq. (6.41).

As a direct application of the obtained stochastic bounds (i.e., Eq. (6.40) and Eq. (6.42)), consider the problem of optimizing the number of parallel paths K subject to the batch delay (accounting for both queueing and resequencing delays). More concretely, we are interested in the number of paths K minimizing the overall average batch delay. Note that the path utilization changes with K as

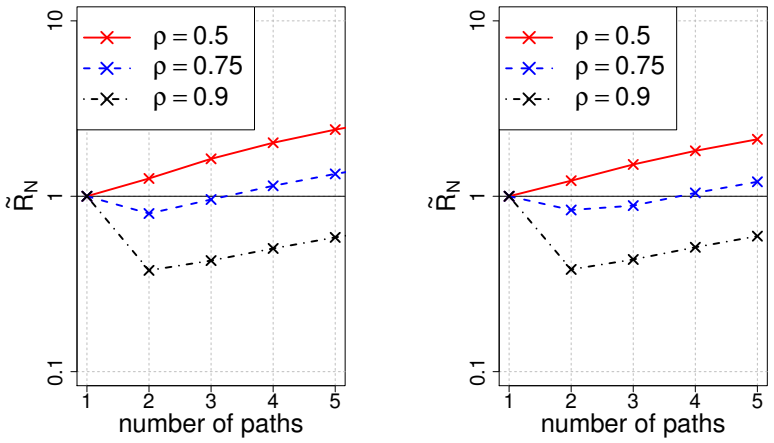
$$\rho = \frac{\lambda}{K\mu} \ ,$$

since each path only receives $\frac{1}{K}$ of the input. In other words, the packets on each path are delivered much faster with increasing K, but they are subject to the additional resequencing delay (which increases as $\log K$ as shown in Section 6.2.1).

To visualize the impact of increasing K on the average batch response times we use the ratio

$$\tilde{R}_K := \frac{\mathbb{E}\left[r_K\right]}{\mathbb{E}\left[r_1\right]} \ ,$$

where, with abuse of notation, $\mathbb{E}\left[r_K\right]$ denotes a bound on the average batch response time for some K, and $\mathbb{E}\left[r_1\right]$ denotes the corresponding baseline bound for $K = 1$; both bounds are obtained by integrating either Eq. (6.40) or Eq. (6.42) for the renewal and the non-renewal case, respectively. Note that the quantity



(a) Renewal (b) Non-renewal

Figure 6.10: Multipath routing reduces the average batch response time when $\tilde{R}_K < 1$; smaller $\tilde{R}_K$ corresponds to larger reductions. Baseline parameter $\mu = 1$ and non-renewal parameters: $p = 0.1, q = 0.4$, $\lambda_1 = \{0.39, 0.7, 0.88\}, \lambda_2 = 0.95$, yielding the utilizations $\rho = \{0.5, 0.75, 0.9\}$ (from top to bottom).

$\tilde{R}_K$, as a ratio of two *upper* bounds, is meaningful only if the corresponding bounds are assumed to be reasonably tight.

In the renewal case, with exponentially distributed interarrival times with parameter λ, and homogeneous paths/servers where the service times are exponentially distributed with parameter μ, we obtain

$$\tilde{R}_K = \left(\frac{\log(N\mu/(\mu-\theta^*))+1}{\log(1/\rho)+1}\right)\left(\frac{\mu-\lambda}{\theta^*}\right) , \tag{6.43}$$

where θ^* is defined as

$$\theta^* := \sup\left\{\theta > 0 \;\middle|\; \frac{\mu}{\mu-\theta}\left(\frac{\lambda}{\lambda+\theta}\right)^K = 1\right\} .$$

In the non-renewal case we obtain the same expression for $\tilde{R}_K$ as in Eq. (6.43) except for the additional prefactor $\frac{\mathbb{E}[h(c_1(1))]}{h(2)}$ prior to K; moreover, θ is the implicit solution from Eq. (6.41).

Figure 6.10 illustrates $\tilde{R}_K$ as a function of K for several utilization levels

ρ for both renewal (a) and non-renewal (b) input; recall that the utilization on each path is $\frac{\rho}{K}$. In both cases, the fundamental observation is that at small utilizations (i.e., roughly when $\rho \leq 0.5$), multipath routing increases the response times. In turn, at higher utilizations, response times benefit from multipath routing but only for 2 paths. While this result may appear as counterintuitive, the technical explanation (in (a)) is that the waiting time in the underlying $E_K/M/1$ queue quickly converges to $\frac{1}{\mu}$, whereas the resequencing delay grows as $\log K$; in other words, the gain in the queueing delay due to multipath routing is quickly dominated by the resequencing delay price.

6.6 Summary

In this chapter we have provided the first computable and non-asymptotic bounds on the waiting and response time distributions in Fork-Join queueing systems under full and partial server mapping. We have analyzed four practical scenarios comprising of either work-conserving or non-work-conserving servers, which are fed by either renewal or non-renewal arrivals. In the case of work-conserving servers, we have shown that delays scale as $\mathcal{O}(\log K)$ in the number of parallel servers K, extending a related scaling result from renewal to non-renewal input. In turn, in the case of non-work-conserving servers, we have shown that the same fundamental factor of $\log K$ determines the system's stability region. Given their inherent tightness, our results can be directly applied to the dimensioning of Fork-Join systems such as MapReduce clusters and multipath routing. A highlight of our study is that multipath routing is reasonable from a queueing perspective for two routing paths only.

7

Replication in Parallel Systems

Despite a significant increase in network bandwidth and computing resources, major online service providers (and not only) still face extremely volatile revenues due to the high variability of latencies (aka response times/delays), especially in their tails (e.g., the 95^{th}-percentile). Several well-cited and convincing studies reported significant potential revenue loss by Google, Bing, or Amazon, were the latencies higher [127, 76, 132]; a typical cited argument is that an additional 100ms in latency would cost Amazon 1% of sales.

Given the late abundance of computing resources, a natural and yet very simple way to improve latencies is *replication*, a concept which was traditionally used to improve the reliability of fault-tolerant systems [126]. In the context of a multi-server (parallel) system, the idea is merely to replicate a task into multiple

copies/replicas, and to execute each replica on a different server. By leveraging the statistical variability of the servers themselves, as execution platforms, it is expected that some replicas would finish much faster than others; for a discussion of various system/OS factors affecting execution times see [53]. The key gain of executing multiple replicas is not to reduce the average latency, but rather the latency tail which is recognized as critically important for ensuring a consistently fluid/natural responsiveness of systems. Therefore, replication can be regarded as being instrumental to the development of "latency tail-tolerant systems", similarly to its role in fault-tolerant systems [53].

While the idea of using redundant requests is not new, as it has been used to demonstrate significant speedups in parallel programs [67, 75], it has become very attractive with its implementation in the MapReduce framework through the so-called "backup-tasks" [54]. Thereafter there has been a surge of very high-quality empirical work which has convincingly demonstrated the benefits of using redundancy for significant latency improvement, both in the mean and also top percentiles. Such works include latency reductions in Google's distributed systems [52], in DNS queries and database servers [146], key-value storage systems [134], cloud storage systems [156], or significant speed-ups of small jobs in data-centers [5] or short TCP flows [158].

Such empirical work has been complemented by several excellent analytical studies (see the Related Work section), which have provided fundamental insight into the benefits of replication. Constrained by analytical tractability, most of these works make several strong assumptions: not only the arrivals are Poisson and the service times are exponentially distributed (i.e., typical assumptions in the queueing literature), but the service times of the replicas plus the corresponding original tasks are statistically independent. By challenging these assumptions, especially the last two, we first provide some elementary analytical arguments, along with some simulation results, that the benefits of replication are highly dependent on both the distributional and correlation structures of the service times. A convincing example is that the stability region of a system is not

monotonous in the replication factor. For instance, by adding a replica server an overloaded system can be stabilized, an advantage which however vanishes by adding additional replica servers.

In this chapter, we provide a general analytical framework to compute stochastic bounds on the response time distributions in replication systems. In particular, our framework covers scenarios with Markovian arrivals, general service time distributions (subject to a finite moment generating function), and a correlation model amongst the original and replicated tasks. Using back-of-the-envelope calculations, our results can be immediately used for engineering purposes (e.g., to determine the optimum number of replicated servers to minimize the top percentiles of latencies). Similar to Chapter 6 our methodology relies on martingales-based techniques. According to several numerical/simulation illustrations, our results exhibit a similar high accuracy, including the challenging case of Markovian arrivals.

To concretely illustrate the applicability of our results we consider two applications. The first is to improve the performance of FJ queueing systems through replication, thus extending the model from 6. In particular, we design an elementary replication policy which can significantly improve not only delay quantiles (e.g., by a factor of roughly 2), but more fundamentally the stability region of a FJ system by a logarithmic factor $\mathcal{O}(\ln K)$ in the number of servers K; our analysis provides a theoretical understanding of the benefits of using back-up tasks in MapReduce, as a proposal to alleviate the problem of stragglers [54]. Albeit such a theoretical benefit is obtained under strong exponential and statistical independence assumptions, simulation results show that the underlying numerical benefits carry over to realistic scenarios subject to correlations amongst replicas. The second application investigates the analytical trade-off between resource usage and response times under replication, a matter which has recently been addressed through Google and Bing empirical studies. The key analytical insight is that increasing resource usage through replication yields a substantial reduction of response time upper quantiles if the

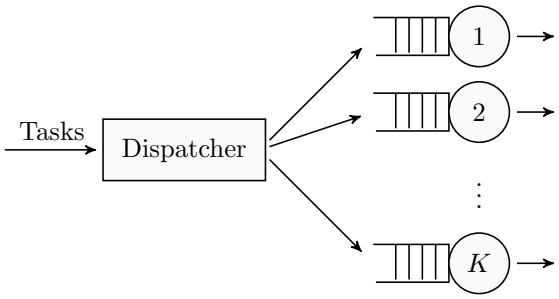


Figure 7.1: A parallel system with K servers; tasks are dispatched to the servers in a possibly replicated manner (i.e., the same task to multiple servers)

service times of the replicas are sufficiently independent (i.e., subject to a low correlation factor, to be later formally described).

The rest of the chapter is organized as follows. In Section 7.1 we introduce the analytical models and discuss related work. In Section 7.2 we provide several insights into the benefits of replication, by following elementary models and derivations. In Section 7.3 we provide our general theoretical framework dealing with both Poisson and Markovian arrivals, and also independent and correlated replicas (i.e., four scenarios). In Section 7.4 we investigate the two applications of our analytical framework.

7.1 Replication Models and Related Work

We consider a parallel system with K homogeneous servers with identical speeds (see Figure 7.1). A stream of tasks arrives at a dispatcher according to some stationary point process; the interarrival times are denoted by t_i with the mean $E[t_1] = \frac{1}{\lambda}$, whereas their number within the (continuous) time interval $(0, t]$ is denoted by $N(t)$. This process can have a Markov structure, to be more precisely defined in Section 7.3.2.

The service times of the tasks are denoted by x_i and are drawn from some general distribution subject to a finite moment generating function; the average is set to $E[x_1] = \frac{1}{\mu}$. For numerical purposes, we will occasionally use the

analytically convenient Pareto distribution, which can be approximated within our theoretical framework through a hyperexponential distribution.

The *utilization* of one server, in a system without replicas where tasks are symmetrically distributed, is denoted by

$$\rho := \frac{\lambda}{K\mu} \ .$$

In general, it is assumed for stability that $\rho < 1$. However, in a system with replication, the expression of the utilization ρ may change depending on various factors (e.g., the distribution of tasks' service times) whereas the stability condition may fail (such occurrences will be specifically indicated).

7.1.1 Tasks Assignment Policies

A crucial design component in the parallel server system is the task assignment policy, i.e., how are the incoming tasks assigned to the K servers for processing? While many such policies have been analytically and empirically studied, we focus on few relevant ones in terms of both performance and overhead:

- **Random**: Each task is dispatched, uniformly at random, to one of the K servers; in the particular case of a Poisson (overall) arrival stream, the tasks arrived at some server follow a Poisson distribution with rate $\frac{\lambda}{K}$.
- **Round-Robin**: Tasks are deterministically dispatched in a circular fashion to the K servers, i.e., task i is assigned to server i mod K (with the convention that 0 stands for K); in the case of a Poisson stream, the interarrival times at some server follow an Erlang $E(K, \lambda)$ distribution.
- **G/G/K**: Unlike the previous two schemes, which immediately dispatch the incoming tasks, and whereby tasks enqueue at the assigned servers, in $G/G/K$ it is the responsibility of each server to fetch a single task, from a centralized queue at the dispatcher, once they become idle.

- **(Full-)Replication** (K-replication factor): Each incoming task i is replicated to all the K servers[1]; the corresponding service times are denoted by $x_{i,j}$ for $j = 1, \ldots, K$. Alike in *Random* and *Round-Robin*, each server maintains a local (FIFO) queue.

- **Partial-Replication** (k-replication factor): Besides *full* replication, a task may be replicated to only $k \leq K$ servers; we will assume that both K and k are powers of 2, and that consecutive blocks of k replicas are allocated to the K servers in a round-robin manner. We call the underlying strategy (strict) *Partial-Replication* when $1 < k < K$, and *No-Replication* when $k = 1$.

In terms of analytical tractability, *Random* and *Round-Robin* are significantly more amenable than $G/G/K$; in fact, exact results are known for $G/G/K$ only in the case of Poisson arrivals and exponential service times (in which case the model is denoted by $M/M/K$). However, $G/G/K$ yields significantly better performance (i.e., much smaller response times of the tasks) than *Random* and *Round-Robin*, especially in the case of high variability of the tasks' service times; in turn *Round-Robin* slightly outperforms *Random* (for an excellent related discussion see [71], pp. 408–430).

It is to be noted however that the superiority of $G/G/K$ is (partly) due to the availability of additional system information, i.e., each task is "informed" about which server is idle such that it can minimize its response time. In turn, amongst policies which are oblivious to such information, *Round-Robin* was shown to be optimal for exponential [58, 147] and increasing failure rate distributions [103]; for a recent state-of-the-art queueing analysis of Round-Robin see [79].

A more sophisticated replication strategy was proposed in the context of massively parallel data processing systems in which (large) jobs are forked/split into (smaller) tasks, each assigned to a server; once a fraction of the tasks finish their executions, each of the remaining (and straggling) tasks are further

[1] For the sake of clarification, the original task is called a replica as well.

replicated. This model appeared in the MapReduce specification [54], and was formally studied in terms of the underlying response time / resource usage trade-off, albeit by disregarding queueing effects in [148]. Another strategy used by Google is to defer the start of executing the second replica for some suitable time, in order to reduce resource usage [53].

7.1.2 Purging/Cancellation Models

Before discussing the relative performance of *Replication* to other policies, we first define how replication strategies deal with residual resources. From a technical perspective, the following distinction is similar to the one of *blocking* and *non-blocking* from Chapter 6:

- **Purging**: A task is considered to complete (and hence its response time is determined) when the fastest replica finishes its execution; at the same time, the residual replicas are all purged/cancelled from the system (with some negligible related cost).
- **Non-Purging**: A task response time is determined as in the *Purging* case, but the remaining replicas leave the system no sooner than their execution end.

Purging is clearly more efficient from a purely *task response-time* perspective, as it frees resources once the first replica completes; this operation demands however synchronization overhead amongst the servers. One basic reason for this superiority is that in the *Non-Purging* model the utilization increases k-fold for a k-replication factor, for any task service time distribution; in particular, a 2-replication factor requires the replica-free system to have a utilization under 50% (otherwise the response times get unbounded). In turn, the growth of the utilization is less pronounced in the *Purging* model, depending on the type of distribution of the service times; in fact, and perhaps counter-intuitively, there is no increase in the case of the exponential distribution regardless the replication factor (for a follow-up discussion see 7.2.2).

Besides the advantage of a better queueing performance, the *Purging* model is much easier to analyze. In fact, the only analytical study of *Non-Purging* is considered in [146]; besides the classical and simplifying assumptions of Poisson arrivals and exponential service times, the underlying queueing analysis critically relies on an artificial statistical independence assumption amongst the queues. Using this assumption, it is shown that below a utilization threshold of 33%, a 2-replication factor strategy does improve the response time despite the inherent doubling of the utilization.

A generalized version of *Partial-Replication* considers the situation when the fastest $l \leq k$ replicas finish their execution (the residual ones being subsequently purged); a practical use of this generalization is in coded distributed storage systems [128]. The central result is that under arrivals with *independent increments*, and exponential (or "*heavier*") service times, *Full-Replication* minimizes the (average) response times. In turn, in the case of "*lighter*" service times and 100% utilization, a replication factor greater than one is detrimental. The underlying proofs use an ingenious coupling argument, but do not provide quantitative results.

Another set of qualitative results, on the superiority of *Full-Replication* for a specific type of service time distributions (including the exponential) is presented in [96]. Interestingly, under a discrete time model with geometric service time distributions, is is shown in [22] through quantitative results that *No-Replication* is optimal (for an explanation of the apparent contradiction between exponential and geometric service time distributions, with respect to the optimality of the replication model, see [96]).

Recently, an *Early Purging* model, in which residual replicas are purged once the first one starts its execution, has been mentioned in [53] and further analyzed in [84]; besides reducing the resource usage, it was shown that this model can also significantly reduce response times despite the apparent loss of diversity, at high utilizations.

The perhaps most fundamental related result obtained so far is a re-

cent exact analysis under the purging model [65]. While the analysis critically relies on the Poisson/exponential models, a key analytical contribution is capturing multi-class arrivals (i.e., different arrival streams are served by different sets of (replicated) servers). The elegance of the results lends itself to several fundamental and contriving insights into the properties of replication, especially accounting for the multi-class feature of the model.

More general stochastic bounds in replication systems are obtained in [61], including the very challenging multi-stage case, by leveraging the analytical power of the stochastic network calculus methodology. While the underlying arrival and service models from [61] are more general than ours, the crucial difference is in handling the underlying correlation structures: concretely, while [61] deals with arbitrary correlation structures yielding stochastic bounds holding in great generality, we exploit the specific correlation structures through the martingale methodology.

7.2 Elementary analytical Insights

Here we complement the previous discussion by providing several motivating examples. After quickly contrasting the task assignment policies introduced earlier, under the Poisson/exponential models, we explore more general service time distributions. The key insight is that the stability region of replicated systems is not necessarily monotonous in the number of replicas; depending on the service distribution, any of the policies *No-Replication*, *Full-Replication*, or *Partial-Replication* can yield the largest stability region.

7.2.1 The M/M model

For some immediate analytical insight, consider the classical example of Poisson arrivals and exponential service times. Due to a lack of closed-form formulas for all considered policies, for large number of servers, we assume that $K = 2$; recall that the (server) utilization is $\rho = \frac{\lambda}{2\mu}$.

The average response times for the four policies (i.e., *Random*, *Round-Robin*, *M/M/2*, and *Replication*) are, respectively,

$$\begin{aligned}
\mathbb{E}\left[T_{Rnd}\right] &= \frac{1}{\mu(1-\rho)} \\
\mathbb{E}\left[T_{RR}\right] &= \frac{2}{\mu\left(1-4\rho+\sqrt{1+8\rho}\right)} \\
\mathbb{E}\left[T_{MM2}\right] &= \frac{1}{\mu\left(1-\rho^2\right)} \\
\mathbb{E}\left[T_{Rep}\right] &= \frac{1}{2\mu(1-\rho)} \ .
\end{aligned}$$

Note that *Replication* induces an $M/G/1$ queueing model, in which the service time is the first order statistics of two i.i.d. random variables (in the current case being an exponential with half of the mean of the original). Immediate comparisons reveal that the minimum ("best") response time is attained by *Replication*; a key reason is that the gain of sampling the minimum of exponential random variables, together with the *Purging* model, significantly dominates the cost of temporary redundant resource usage. In turn, the maximum ("worst") response time is attained by *Random*; the relative performance of *Round-Robin* and *M/M/2* depends on the value of ρ. Lastly, we point out that the superiority of *Replication* immediately extends to larger values of K.

More general results in terms of lower and upper bounds on the average response time in the case of a variant of *Replication*, in which only the fastest $l \leq K$ tasks are required to complete (whilst the residual tasks are purged) (and which was *qualitatively* studied in [128]), appeared in [83]; in particular, it was shown that *Replication* outperforms the corresponding $M/M/K$ model. Further upper bounds were derived in the case of general service time distributions, using existing bounds on the first two moments of the l^{th} order statistics.

7.2.2 Beyond the M model

In the previous example with exponential service times, the stability region is invariant to the replication factor; the reason is that the 1^{st} order statistic of

K (independent) exponential random variables $exp(\mu)$ is an exponential random dom variable $exp(K\mu)$. The next elementary examples show that any strategy amongst *No-Replication*, *Full-Replication*, or *Partial-Replication* can yield the strictly largest stability regions (and hence "best" response times, at least in some subset of the stability region; a follow-up discussion will be given in Section 7.3.3). A fundamental reason is the assumption of independent service times of the replicas, which motivates the need for accounting for some correlation structures.

Recall that in the *No-Replication* scenario, a necessary and sufficient condition for stability (or, equivalently, for finite response times) is

$$\mathbb{E}[x_1] < K\mathbb{E}[t_1] \ .$$

In the case of *Full-Replication*, the corresponding stability condition is given by

$$\mathbb{E}\left[\min\left\{x_1, \ldots, x_n\right\}\right] < \mathbb{E}[t_1] \ ,$$

whereas in the case of *Partial-Replication* with replication factor k by

$$\mathbb{E}\left[\min\left\{x_1, \ldots, x_k\right\}\right] < \frac{K}{k}\mathbb{E}[t_1] \ . \tag{7.1}$$

Denoting the CCDF of x_i by

$$f(x) := \mathbb{P}(x_1 \geq x) \ ,$$

we observe from the previous stability conditions that the "best" replication-factor k is

$$\underset{k}{\operatorname{argmin}}\, k \int f^k(x)dx \ . \tag{7.2}$$

We next present examples of different distributions for x_i resulting in "best" scenarios for each of the three replication strategies.

No-Replication: Uniform

Assume uniformly distributed service times, i.e., $x_i \sim \mathcal{U}_{[0,1]}$. The following argument shows that in this case replication is detrimental, i.e.,

$$\mathbb{E}[x_1] < k\mathbb{E}[\min\{x_1, \ldots, x_k\}] \ ,$$

for any $k \geq 2$:

$$\begin{aligned} k\mathbb{E}[\min\{x_1, \ldots, x_k\}] =& k \int_0^\infty \mathbb{P}(\min\{x_1, \ldots, x_k\} \geq x)\, dx \\ =& k \int_0^\infty \mathbb{P}(x_1 \geq x)^k\, dx \\ =& \int_0^1 k x^k dx = \frac{k}{k+1} > \frac{1}{2} = \mathbb{E}[x_1] \ . \end{aligned}$$

The same argument additionally shows that *Partial-Replication* is better than *Full-Replication*. This result extends the qualitative observation from [128] (i.e., Theorem 4 therein, restricted to a 100% utilization, and hence an unstable regime) to any (stable) utilization.

Full-Replication: Weibull

Let the x_i now be Weibull distributed, i.e., $f(x) = e^{-(x/\lambda)^\alpha}$. For $\alpha < 1$, a higher degree of replication is "better", as shown below:

$$\begin{aligned} k\mathbb{E}[\min\{x_1, \ldots, x_k\}] =& k \int_0^\infty \mathbb{P}(\min\{x_1, \ldots, x_k\} \geq x) dx \\ =& k \int_0^\infty e^{-k(x/\lambda)^\alpha} dx \\ =& k \frac{\lambda}{k^{1/\alpha}} \Gamma(1 + 1/\alpha) \ . \end{aligned}$$

By the assumption on α, the last term is monotonically decreasing in k. Note that in the special case of exponentially distributed x_i, i.e., $\alpha = 1$, replication is neither beneficial nor detrimental (from the point of view of the stability region), as pointed out earlier. This result also extends the qualitative observation

from [128] (i.e., Theorem 3) to any (stable) utilization.

Partial Replication: Pareto

Lastly we consider the Pareto distribution, i.e., $f(x) = x^{-\alpha}$ for $x \geq 1$. For a suitably chosen $\alpha > 1$, it can be shown that (strict) *Partial-Replication* can become "better" than both *Full-Replication* and *No-Replication*:

$$\begin{aligned} k\mathbb{E}\left[\min\left\{x_1, \ldots, x_k\right\}\right] =& k \int_0^\infty \mathbb{P}\left(\min\left\{x_1, \ldots, x_k\right\} \geq x\right) dx \\ =& k + k \int_1^\infty x^{-k\alpha} dx = k + \frac{k}{k\alpha - 1} \ . \end{aligned}$$

It is clear that for sufficiently small $\alpha > 1$, the minimal value is attained for $k = 2$.

This last example highlights that the performance of replication strategies heavily depends on the replication factor k, the service time distribution, and other underlying assumptions. In particular, performance is not monotonic in k, and thus an optimization framework is desirable (related results, on the actual response time distributions as a function of k will be provided in the next section).

For complementary numerical results illustrating the counterintuitive effect of k, consider the Pareto distribution with the assumption of independent service times of the k replicas. Let $K = 4$, arrival rate $\lambda = 1$, $\alpha = 1.1$ (for the Pareto distribution), yielding a utilization $\rho = 2.75$ (i.e., 275%). By plotting the simulated latencies of the first 10^4 packets, Figure 7.2 shows that while the system without replication is in overload, a replication factor of $k = 2$ stabilizes the system (reducing the utilization to 0.91), whereas a replication factor of 4 puts the system back in overload (increasing the utilization to 1.29).

The non-monotonic behavior in k disappears when the service times are sufficiently correlated. Indeed, by taking the service times of the replicas as

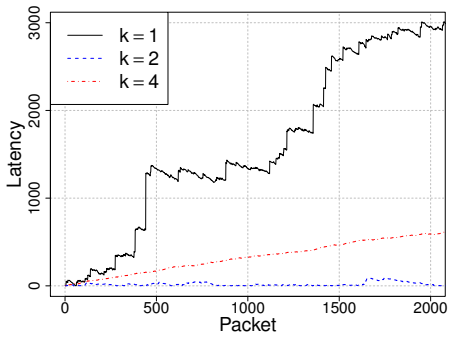


Figure 7.2: From overload ($k = 1$) to underload ($k = 2$) and back ($k = 4$) ($K = 4$, $\alpha = 1.1$, $\lambda = 1$, and utilization $\rho = 2.75$ (for the non-replicated $k = 1$ case))

$y + x_i$ (where the x_i are Pareto distributed, and $y \geq 0$ is arbitrary), it holds:

$$\begin{aligned} k\mathbb{E}[\min\{y + x_1, \ldots, y + x_k\}] &= k\mathbb{E}[y] + k\mathbb{E}[\min\{x_1, \ldots, x_k\}] \\ &= k\mathbb{E}[y] + k + \frac{k}{k\alpha - 1} \\ &= k\left(\mathbb{E}[y] + \frac{k\alpha}{k\alpha - 1}\right) , \end{aligned}$$

so that (for a suitably chosen $\alpha > 1$, and a sufficiently large value of $\mathbb{E}[y]$) the optimal value of k in Eq. (7.2) is 1 (i.e., *No-Replication* is "best").

7.3 Theory

We assume a queueing system with K servers and interarrival times between jobs i and $i + 1$ denoted by t_i. Upon its arrival, job i is replicated to $k \leq K$ servers where they are processed with service times $x_{i,1}, \ldots, x_{i,k}$, respectively. We throughout assume that K is an integral multiple of k. Further, the jobs are assigned to the $\frac{K}{k}$ batches in a round robin scheme, i.e., the interarrival times for one batch can be described as:

$$\tilde{t}_i := \sum_{j=0}^{K/k-1} t_{(i-1)\frac{K}{k}+j} \ .$$

The following recursion describes the response time r_{i+1} of job $i+1$, i.e., the time between the job's arrival and its service being complete:

$$r_1 := \min_{j \leq k} x_{1,j} \ , \quad r_{i+1} := \min_{j \leq k} \{x_{i+1,j}\} + \max\{0, r_i - \tilde{t}_i\} \ ,$$

resulting in a representation of the *steady-state* response time r as:

$$r =_{\mathcal{D}} \max_{n \geq 1} \left\{ \sum_{i=1}^{n+1} \min_{j \leq k} \{x_{i,j}\} - \sum_{i=1}^{n} \tilde{t}_i \right\} \ , \tag{7.3}$$

where the empty sum is by convention equal to 0. Note that, essentially the only difference between the response time as defined above (Eq. (7.3)) and the response time in the FJ scenario (Eq. (6.17)) is that the inner max-operator is exchanged by the min.

Depending on the correlation between either the interarrival times and the service times, respectively, we consider four different scenarios: In Subsection 7.3.1, all random variables t_i, $x_{i,j}$ are assumed to be independent. In Subsection 7.3.2, the interarrival times are driven by a certain Markov chain, whereas in Subsection 7.3.3 the service times are correlated through a common additive factor. Finally, in Subsection 7.3.4, a combination of both correlation models is considered.

7.3.1 Independent Arrivals, Independent Replication

As stated above, we consider the scenario of *independent replication*, i.e., the set $\{t_i, x_{i,j} \mid i \geq 1, \ j \leq k\}$ forms an independent family of random variables.

The next Theorem provides an upper bound on the CCDF of r as defined in Eq (7.3):

Theorem 7.1. *Let θ_{ind} be defined by*

$$\theta_{ind} := \sup \left\{ \theta \geq 0 \ \middle| \ \mathbb{E}\left[e^{\theta \min_{j \leq k}\{x_{i,j}\}}\right] \mathbb{E}\left[e^{-\theta t_i}\right]^{\frac{K}{k}} \leq 1 \right\} \ .$$

Then the following bound on the response time holds for all $\sigma \geq 0$:

$$\mathbb{P}(r \geq \sigma) \leq \mathbb{E}\left[e^{\theta_{ind} \min_{j \leq k}\{x_{1,j}\}}\right] e^{-\theta_{ind}\sigma} \ .$$

Note that, given the stability condition from Eq. (7.1), $\theta_{\text{ind}} > 0$ as

$$\frac{d}{d\theta}\, \mathbb{E}\left[e^{\theta \min_{j \leq k}\{x_{i,j}\}}\right] \mathbb{E}\left[e^{-\theta t_i}\right]^{\frac{K}{k}}\Big|_{\theta=0} = \mathbb{E}\left[\min_{j \leq k}\{x_{i,j}\}\right] - \frac{K}{k}\mathbb{E}\left[t_i\right] < 0 \ .$$

Proof. Define the process $M(n)$ by

$$M(n+1) := e^{\theta_{\text{ind}}\left(\sum_{i=1}^{n+1} \min_{j \leq k}\{x_{i,j}\} - \sum_{i=1}^{n} \tilde{t}_i\right)} \ .$$

As in the proof of Theorem 6.2 one shows that $M(n)$ is a martingale. Now define the stopping N as

$$N := \min\left\{ n \geq 0 \;\middle|\; \sum_{i=1}^{n} \min_{j \leq k}\{x_{i,j}\} - \sum_{i=1}^{n-1} \tilde{t}_i \geq \sigma \right\} \ ,$$

and proceed as in the proof of Theorem 3.4. □

We point out that the proof essentially follows the bounding technique for GI/GI/1 queues from [92], also used in the proof of Theorem 6.2.

7.3.2 Markovian Arrivals, Independent Replication

We now turn to the more realistic scenario where the interarrival times are correlated: A two-state Markov chain $Z(n)$ alternates between *active* and *inactive* periods; while in the active state, exponentially distributed interarrival times are generated with parameter λ_{act}, and the chain turns inactive with probability $p > 0$. In the inactive state, *one* interarrival time (exponentially distributed, parameter $\lambda_{\text{inact}} < \lambda_{\text{act}}$) is generated, and the chain jumps back to the active state (see Figure 7.3) (Note that this is essentially a special case of the Markov

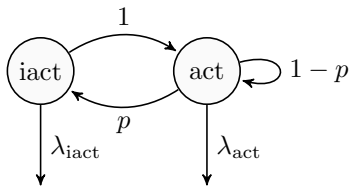


Figure 7.3: Two-state Markov chain $Z(n)$

chain from Figure 6.4.). Formally, let

$$t_{i,\text{act}} \sim \text{Exp}(\lambda_{\text{act}}) \ , \quad t_{i,\text{iact}} \sim \text{Exp}(\lambda_{\text{iact}})$$

be i.i.d. random variables and define the sequence of interarrival times t_i by

$$t_i := t_{i,Z(i)} \ .$$

The steady state distribution π of the Markov chain is given by

$$\pi_{\text{act}} = \frac{1}{1+p} \ , \quad \text{and} \quad \pi_{\text{iact}} = \frac{p}{1+p} \ ,$$

such that for the average of the interarrival times holds

$$\mathbb{E}[t_i] = \left(\lambda_{\text{act}}^{-1} + p\lambda_{\text{iact}}^{-1}\right) / \, (1+p) \tag{7.4}$$

Note that the transition matrix of $Z(n)$ is given by:

$$T := \begin{pmatrix} 0 & 1 \\ p & 1-p \end{pmatrix} \ .$$

In order to state the main result of this section, we need an exponential transform of T similar to the one in Eq. (3.13):

Definition 7.2. *For $0 \leq \theta < \lambda_{iact}$, let T_θ denote the following matrix:*

$$T_\theta := \begin{pmatrix} 0 & \frac{\lambda_{act}}{\lambda_{act}+\theta} \\ p\frac{\lambda_{iact}}{\lambda_{iact}+\theta} & (1-p)\frac{\lambda_{act}}{\lambda_{act}+\theta} \end{pmatrix} .$$

Further, let $\xi(\theta)$ denote the maximal positive eigenvalue of T_θ, and $h = (h_{act}, h_{iact})$ be a corresponding eigenvector.

The following Theorem is the analogous result to Theorem 7.1 (note that the service times $x_{i,j}$ are still assumed to be i.i.d.):

Theorem 7.3. *Let $1 \leq k \leq K$ and θ_{mkv} be defined by*

$$\theta_{mkv} := \sup \left\{ \theta \geq 0 \;\middle|\; \mathbb{E}\left[e^{\theta \min_{j \leq k}\{x_{i,j}\}} \right] \xi^{\frac{K}{k}}(\theta) \leq 1 \right\} .$$

Then, for the system with replication to k out of K servers, the following bound on the response time holds for all $\sigma > 0$:

$$\mathbb{P}(r \geq \sigma) \leq \mathbb{E}\left[e^{\theta_{mkv} \min_{j \leq k}\{x_{i,j}\}} \right] e^{-\theta_{mkv}\sigma} .$$

Proof. Proceeding similarly as in the proof of Theorem 7.1, define the process $M(n)$ by

$$M(n) := h_{Z(n\frac{K}{k}-1)} e^{\theta_{\mathrm{mkv}}\left(\sum_{i=1}^{n} \tilde{x}_i - \sum_{i=1}^{n-1} \tilde{t}_i\right)} .$$

$M(n)$ is a martingale: By induction over $\frac{K}{k} - 1$ one shows that:

$$\mathbb{E}\left[e^{-\theta_{\mathrm{mkv}}\tilde{t}_{n+1}} \;\middle|\; Z\left(n\frac{K}{k} - 1\right) \right] = \left(T_{\theta_{\mathrm{mkv}}}^{\frac{K}{k}}\right)_{Z(n\frac{K}{k}-1),\mathrm{iact}} + \left(T_{\theta_{\mathrm{mkv}}}^{\frac{K}{k}}\right)_{Z(n\frac{K}{k}-1),\mathrm{act}} .$$

Now:

$$\begin{aligned}
&\mathbb{E}\left[h_{Z((n+1)\frac{K}{k}-1)}e^{\theta_{\mathrm{mkv}}\left(\tilde{x}_{n+1}-\tilde{t}_n\right)} \;\middle|\; Z\left(n\frac{K}{k}-1\right)=\mathrm{act}\right] \\
&= \mathbb{E}\left[e^{\theta_{\mathrm{mkv}}\min_{j\le k}\{x_{n,j}\}}\right]\left(T^{\frac{K}{k}}_{\theta_{\mathrm{mkv}}}h\right)_{\mathrm{act}} \\
&= \mathbb{E}\left[e^{\theta_{\mathrm{mkv}}\min_{j\le k}\{x_{n+1,j}\}}\right]\,\xi^{\frac{K}{k}}(\theta_{\mathrm{mkv}})h_{\mathrm{act}} \\
&= h_{\mathrm{act}}\ ,
\end{aligned}$$

and similarly one obtains:

$$\mathbb{E}\left[h_{Z((n+1)\frac{K}{k}-1)}e^{\theta_{\mathrm{mkv}}\left(\tilde{x}_{n+1}-\tilde{t}_n\right)} \;\middle|\; Z\left(n\frac{K}{k}-1\right)=\mathrm{iact}\right]=h_{\mathrm{iact}}\ ,$$

so that:

$$\mathbb{E}\left[h_{Z((n+1)\frac{K}{k}-1)}e^{\theta_{\mathrm{mkv}}\left(\tilde{x}_{n+1}-\tilde{t}_n\right)} \;\middle|\; Z\left(n\frac{K}{k}-1\right)\right]=h_{Z(n)}\ .$$

Now multiply both sides by $e^{\theta_{\mathrm{mkv}}\left(\sum_{i=1}^{n}\min_{j\le k}\{x_{i,j}\}-\sum_{i=1}^{n-1}t_i\right)}$. The proof completes along the same kind of lines as in the proof of Theorem 7.1. □

7.3.3 Independent Arrivals, Correlated Replication

We now address the more realistic scenario when the replicas $x_{i,j}$ are no longer independent; we consider the following correlation model (from [83]):

$$x_{i,j}=\delta y_i+(1-\delta)\,y_{i,j}\ , \tag{7.5}$$

where the random variables y_i and $y_{i,j}$ are i.i.d., and $\delta\in[0,1]$. Here, the parameter δ describes the degree of correlation amongst the replicas: $\delta=0$ corresponds to the i.i.d. case from Section 7.3.1, whereas for $\delta=1$ the K servers are entirely synchronized so that no replication gain is achieved.

The interarrival times t_i are first assumed to be i.i.d. as in Section 7.3.1.

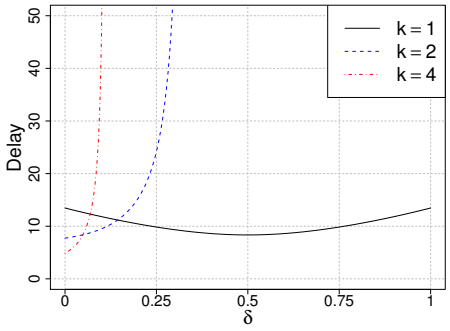


Figure 7.4: Delay for the 99%-percentile as a function of the degree of correlation δ ($\lambda = 4 * 0.75$, $\mu = 1$, $K = 4$, $k = 1, 2, 4$)

Theorem 7.4. *Let θ_{cor} be defined by*

$$\theta_{cor} := \sup \left\{ \theta \geq 0 \;\middle|\; \mathbb{E}\left[e^{\theta\delta y_i}\right] \mathbb{E}\left[e^{\theta(1-\delta)\min_{j\leq k}\{y_{i,j}\}}\right] \mathbb{E}\left[e^{-\theta t_i}\right]^{\frac{K}{k}} \leq 1 \right\} .$$

Then the following bound on the response time holds for all $\sigma \geq 0$:

$$\mathbb{P}(r \geq \sigma) \leq \mathbb{E}\left[e^{\delta\theta_{cor} y_i}\right] \mathbb{E}\left[e^{(1-\delta)\theta_{cor}\min_{j\leq k}\{y_{i,j}\}}\right] e^{-\theta_{cor}\sigma} .$$

Proof. Entirely analogous to the proof of Theorem 7.1. □

To illustrate the impact of the correlation parameter δ we consider the special case when y_i and $y_{i,j}$ are exponentially distributed with parameter μ. Clearly,

$$\min_{j\leq k}\{y_{i,j}\} \sim \text{Exp}(k\mu) ,$$

so that $\theta_{\text{cor}} > 0$ is the solution of

$$\frac{\mu}{\mu - \delta\theta} \; \frac{k\mu}{k\mu - (1-\delta)\,\theta} \; \frac{\lambda}{\lambda + \theta} = 1 .$$

Further, Figure 7.4 illustrates the 99%-percentile of the delay as a function of the degree of correlation δ for several numbers of replicas k. Strictly from the point of view of the stability region, as it was also considered in Sec-

tion 7.2.2, we observe that replication (both $k = 2$ and $k = 4$) is detrimental as the corresponding systems quickly become unstable. In contrast, from the point of view of delays, replication can be beneficial within a subset of the corresponding stability region notwithstanding its strict inclusion in the stability region of the non-replicated system. This fundamental observation can be intuitively explained in that for larger values of the degree of correlation δ, the servers become more synchronized and consequently no significant *replication gain* can be achieved; a further follow-up discussion concerning a convergence result depending on δ will be given in Section 7.4.1. As a side remark, the symmetry in the delay for $k = 1$ is due to the underlying Erlang distribution, which minimizes its variance at $\delta = .5$.

7.3.4 Markovian Arrivals, Correlated Replication

We briefly state the results for the combination of the scenario from Sections 7.3.2 and 7.3.3:

Theorem 7.5. *With the same notation as in Sections 7.3.2 and 7.3.3, let* $\theta_{mkv,cor}$ *be defined by*

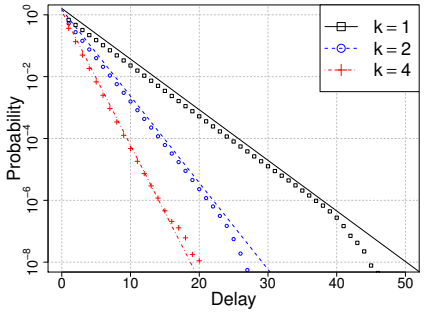
$$\theta_{mkv,cor} := \sup\left\{\theta \geq 0 \;\middle|\; \mathbb{E}\left[e^{\theta\delta y_i}\right]\mathbb{E}\left[e^{\theta(1-\delta)\min_{j\leq k}\{y_{i,j}\}}\right]\xi^{\frac{K}{l}}(\theta) \leq 1\right\}.$$

Then the following bound on the response time holds for all $\sigma \geq 0$*:*

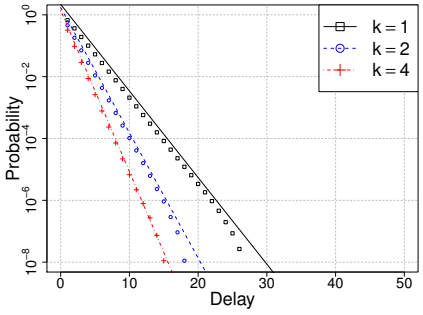
$$\mathbb{P}(r \geq \sigma) \leq \mathbb{E}\left[e^{\delta\theta_{mkv,cor}y_i}\right]\mathbb{E}\left[e^{(1-\delta)\theta_{mkv,cor}\min_{j\leq k}\{y_{i,j}\}}\right]e^{-\theta_{mkv,cor}\sigma}.$$

Proof. Entirely analogous to the proofs of Theorems 7.1 and 7.3. □

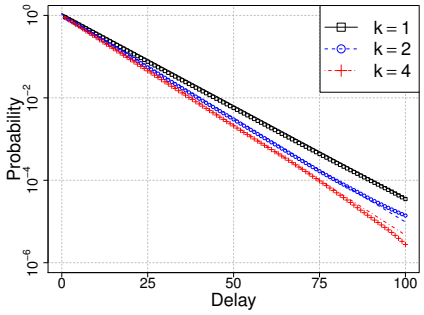
To numerically compare our stochastic bounds from Theorems 7.1, 7.4, 7.3, and 7.5 to simulation results we refer to Figures 7.5(a)–(d), respectively. In all four scenarios, addressing combinations of independent/correlated arrivals and replications, jobs are replicated to $k = 1, 2, 4$ out of a total number of $K = 4$ servers. The parameters of the respective models are chosen such that



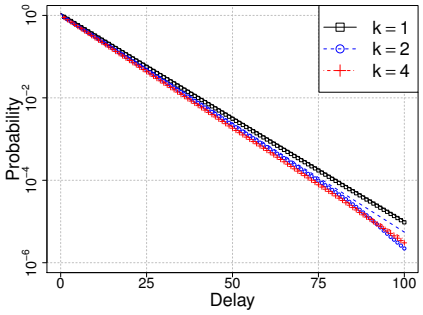
(a) Poisson (Theorem 7.1)



(b) Poisson with correlation ($\delta = .5$) (Theorem 7.4)



(c) Markov ($p = 0.1$, $\lambda_{\text{iact}} = 0.3$, $\lambda_{\text{act}} = 30$) (Theorem 7.3)



(d) Markov with correlation (as in (c) and $\delta = .5$) (Theorem 7.5)

Figure 7.5: Stochastic bounds vs. simulation results accounting for 10^9 packets ($K = 4$, $\rho = .75$, $\mu = 1$)

the (server) utilization remains constant, i.e., $\rho = 0.75$. In particular, in Figure 7.5(a), both the interarrival- and service times are exponentially distributed with parameters $\lambda = 4 \times 0.75 = 3$ and $\mu = 1$. In Figure 7.5(b), the interarrival times are again exponential with $\lambda = 4 \times 0.75 = 3$, the correlation factor is $\delta = 0.5$, whereas the components y_i and $y_{i,j}$ of the service times $x_{i,j}$ from Eq. (7.5) are exponential with parameter

$$\mu' := \delta + (1 - \delta) \,/ k \ ,$$

such that $\mathbb{E}[x_{i,j}] = 1$. In Figure 7.5(c), the parameters for the Markov chain are $p = 0.1$, $\lambda_{\text{act}} = 30$, $\lambda_{\text{iact}} = 0.3$, whereas the services times are exponential with parameter $\mu = 1$. According to Eq. (7.4) the average of the interarrival times

is $E[t_i] = 1/3$, such that $\rho = 0.75$. Finally, in Figure 7.5(d), the parameters for the service times from Figure 7.5(b) are combined with the parameters for the interarrival times from Figure 7.5(c). We remark that in all four scenarios the stochastic bounds from Theorems 7.1, 7.4, 7.3, and 7.5 are remarkably accurate.

7.4 Applications

In this section we present two practical applications of our theoretical framework. The first concerns integrating replication with a fork-join queueing model (see Chapter 6); a major outcome is the construction of an intuitive class of assignment policies which can fundamentally improve response times. The second investigates the analytical trade-off between resource usage and response times, an issue which was subject to several measurement studies involving Google and Bing traces.

7.4.1 Fork-Join with Replication (FJR)

In this section we consider replication in the context of a FJ queueing system as in Chapter 6, i.e., arriving jobs are split into K different tasks which are mapped to K servers to be processed independently. A job is considered finished once *all* of its corresponding tasks have finished. We consider the special case of a *blocking* system (see Subsections 6.2.2 and 6.3.2) whereby jobs cannot be forked before all of the tasks of the previous job have left the system.

The obvious drawback of this blocking model is that it is no longer work-conserving: servers can become idle once some but not all tasks of one job are complete. Moreover, the stability condition of the system becomes a function of the number of servers.

Consider for instance the case of Poisson arrivals with rate λ and exponential and identically distributed service times x_i, $i = 1, \ldots, K$, with rate μ. As the distribution of the maximum of i.i.d. exponential random variables satisfies

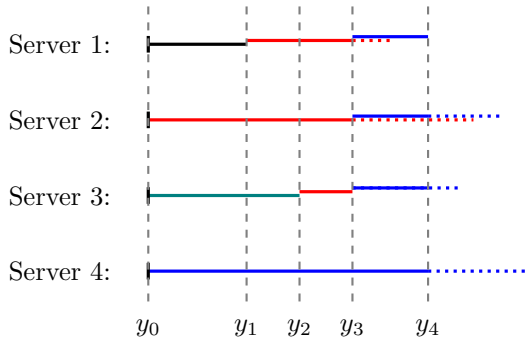


Figure 7.6: FJR policy; different colors denote different tasks, dotted lines indicate tasks which have been purged.

$\max_{i=1}^{K} x_i =_{\mathcal{D}} \sum_{i=1}^{K} \frac{x_i}{i}$ [119], the stability condition is roughly

$$\frac{\lambda}{\mu} \ln K < 1 \ . \tag{7.6}$$

To overcome the issue of decaying stability regions (in the number of servers K) we propose the following task assignment policy which suitably triggers replicas on top of the standard FJ model.

Policy FJR (Fork-Join with Replication): Once a server finishes its task, it immediately replicates a remaining task from another running server. When either the original task or one of its replica has finished, the others are immediately purged.

FJR can be regarded as a concrete implementation of backup-tasks in MapReduce (which is not explicitly presented in the original MapReduce description [54]). Our policy is quite flexible in that the executing task to be replicated can be chosen randomly (yet independently of the current state); moreover, as multiple servers can become idle at the same time (due to the underlying purging model), each can replicate any executing tasks. Intuitively, this flexibility is due to the underlying assumption of exponentially distributed and independent service times.

The main result of the FJR policy is the following:

Theorem 7.6. *The* overall *service time x of jobs processed by FJR follows an*

Erlang$(K, K\mu)$*-distribution. Consequently, the corresponding stability condition is*

$$\frac{\lambda}{\mu} < 1 \ .$$

Proof. Let $y_1 < y_2 < \ldots < y_K$ denote the times where the tasks (original or replica) finish (see Figure 7.6). Obviously, it holds $x = y_K$. We first show (with the convention $y_0 \equiv 0$) that the family

$$\{y_i - y_{i-1} \mid i \geq 1\}$$

is independent and identically exponentially distributed with parameter $K\mu$.

For $i = 1$, this follows directly from the well known fact that the minimum over K independent, exponential random variables with rate μ is exponentially distributed with rate $K\mu$.

Now, suppose $1 \leq l \leq K$ tasks finish, or are purged, at time y_i. Denote by $z_1, \ldots, z_l$ the corresponding service times of the respective replicas starting at y_i. For the remaining $K - l$ servers, denote by $z_{l+1}, \ldots, z_K$ the service times of the current tasks and by $s_{l+1}, \ldots, s_K$ the length of time they started before y_i. Now we can write

$$\begin{aligned} y_{i+1} - y_i = \min\big\{ & z_1, \ldots, z_l, z_{l+1} - s_{l+1}, \ldots, z_K - s_K \\ & \big| \ z_{l+1} - s_{l+1}, \ldots, z_K - s_K > 0 \big\} \ . \end{aligned}$$

Note that the family $\{z_1, \ldots, z_K\}$ is independent from one another and from $\{s_{l+1}, \ldots s_K\}$.

Now, with

$$A := \{z_{l+1} - s_{l+1}, \ldots, z_K - s_K > 0\} \ ,$$

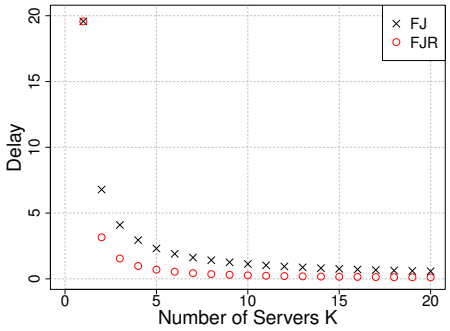


Figure 7.7: Improving the 99%-percentile of delays in FJ systems by replication

$\vec{s} := (s_{l+1}, \ldots, s_K)$, and $f(.)$ the common density of $\vec{s}$:

$$\begin{aligned}
\mathbb{P}(y_{i+1} - y_i \geq \sigma) &= \mathbb{P}\left(\min\left\{z_1, \ldots, z_l, z_{l+1} - s_{l+1}, \ldots, z_K - s_K\right\} \geq \sigma \mid A\right) \\
&= e^{-l\mu\sigma} \int e^{-\mu\left(\sum_{j=l+1}^{K} \sigma + s_j\right)} f(\vec{s}) d\vec{s} \Big/ \mathbb{P}(A) \\
&= e^{-K\mu\sigma} \int e^{-\mu \sum_{j=l+1}^{K} s_j} f(\vec{s}) d\vec{s} \Big/ \mathbb{P}(A) \\
&= e^{-K\mu\sigma} \int \mathbb{P}(z_{l+1} > s_{l+1}, \ldots, z_K > s_K) f(\vec{s}) d\vec{s} \Big/ \mathbb{P}(A) \\
&= e^{-K\mu\sigma} \ ,
\end{aligned}$$

so that $y_i - y_{i-1}$ is exponentially distributed for any $1 \leq i \leq K$. It follows that

$$x = y_K = \sum_{i=1}^{K} y_i - y_{i-1}$$

has an Erlang distribution with parameters K and $K\mu$. Therefore $\mathbb{E}[x] = \frac{1}{\mu}$, which completes the proof. □

It is evident that the stability region of FJR improves the stability region of the standard FJ queueing model (given in Eq. (7.6)) by a logarithmic factor. Figure 7.7 shows the 99^{th} percentile of the delays as a function of K ($\mu = 1$ and Poisson arrivals with rate such $\rho = 0.75$ when $K = 1$; the utilization consequently decays for larger K). The numerical benefit of FJR is that it roughly halves the FJ delays.

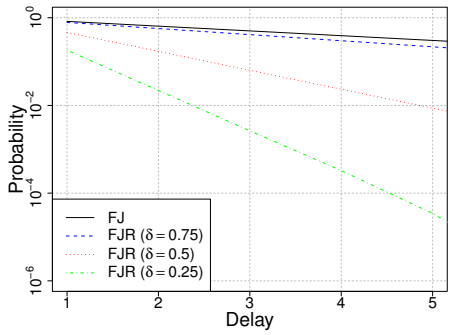


Figure 7.8: Convergence of FJR to FJ in terms of the degree of correlation δ ($K = 4$).

While the fundamental improvements achieved by the FJR policy, relative to the standard FJ model, are remarkable, we point out that they are mainly due to the exponential and independence assumptions on the triggered replicas. Unfortunately, a clean analysis in the case of correlated replicas (even of the form $(1 - \delta)x_i + \delta x$, with x and x_i's being exponentially distributed) appears prohibitive. For this reason, we resort to simulations to illustrate that the benefits of FJR (proven in the ideal i.i.d. and exponential case) carry over to more practical scenarios with correlated replicas.

Concretely, Figure 7.8 shows the bounds on the delay distributions for FJ and three FJR scenarios, depending on the degree of correlation δ (the service times of an original and its replicated tasks are $(1 - \delta)x_i + \delta x$, with x and x_i's being exponentially distributed with rate $\mu = 1$; Poisson arrivals such that the utilization for FJ is $\rho = 0.9$ (the corresponding utilizations for FJR are not analytically determined)). The figure essentially illustrates the convergence of FJR to FJ; we remark in particular that FJ is invariant to δ, whereas FJR behaves identically as FJ when $\delta = 1$ (i.e., when the replicas are identical to the originals).

7.4.2 Resource Usage vs. Response Times

For the second application we investigate the analytical trade-off between resource usage and response times under replication. This application is motivated by empirical observations from Google [53] and Bing [80] traces that a slight increase in the resource budget may yield substantial reductions of the upper quantiles of response times. For example, [80] reports that the 99^{th} percentile of the delay improves by as much as 40% under a 5% increase of the resource budget. To compensate for the inherent increase of resource usage under replication, the schemes from [53, 80] defer the execution time of the replicas until the original request has been outstanding for a given *replication offset* Δ.

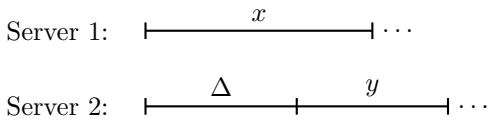


Figure 7.9: Replication with deferred execution times: a replica (at Server 2) may start no sooner than ($\Delta \geq 0$) after the starting time of the original (at Server 1).

Consider a scenario with two servers. Jobs arrive with rate λ at the first server with interarrival times t_i and service times $x_i =_{\mathcal{D}} x$; if the processing time of a job is larger than some fixed Δ, then the job is replicated at the second server with service times $y_i =_{\mathcal{D}} y$ (see Figure 7.9 for a time-line illustration of a generic job with execution time x and its replica, should $x > \Delta$). Whenever either of the original job or its replica finishes execution, the residual service time of the other is cancelled (i.e., the purging replication model).

The utilization at the first server is thus given by

$$\rho_1 = \lambda \mathbb{E}\left[\min\{x, \Delta + y\}\right] \ , \tag{7.7}$$

whereas the utilization at the second is

$$\rho_2 = \lambda \mathbb{E}\left[\min\left\{|x-\Delta|, y\right\}\right] \ . \tag{7.8}$$

We note that unlike previous models, where the utilization is server independent, the current model is subject to different server utilizations due to the lack of symmetry in dispatching the load.

The measure for *resource usage* is the total utilization at the two servers and is denoted by u to avoid confusion

$$u := \rho_1 + \rho_2 \ .$$

Aiming for explicit results, we assume for convenience the independent replication model and the exponential service model, i.e., $x \sim \exp(\mu)$ and $y \sim \exp(\mu)$, with $\mu = 1$. Given the statistical independence of x_i's and y_i's, straightforward computations of integrals yield

$$\begin{aligned} \rho_1 &= \frac{\lambda}{\mu} - \frac{\lambda}{2\mu} e^{-\mu\Delta} \text{ and} \\ \rho_2 &= \frac{\lambda}{2\mu} e^{-\mu\Delta} \ , \end{aligned}$$

which means that the resource usage $u = \frac{\lambda}{\mu}$ is invariant to the choice of Δ.

In turn, Δ can have a major impact on the response times: for instance, if $\mu < \lambda < 2\mu$ then the response times can be either unbounded for sufficiently large values of Δ, and in particular when $\Delta = \infty$ (i.e., no replicas are executed), or finite for some values of Δ.

In fact, an immediate application of Theorem 7.1 yields that the response time is non-decreasing in Δ. Thus, the optimal choice of Δ, which minimizes both the resource usage and the response times, is $\Delta = 0$. The explanation for the seemingly sharp contrast between this theoretical result and the empirical results from [53, 80] is the underlying independence assumption of the replication model.

7.5 Summary

In this chapter we have developed an analytical framework to compute stochastic bounds on the response time distribution in quite general replicated queueing systems. Unlike existing models, ours cover practical scenarios including correlated interarrivals, general service time distributions, and not necessarily independent service times for original tasks and their replicas. By employing the powerful martingale methodology, we were able to derive numerically accurate bounds by exploiting the specific correlation structures of the underlying processes. Remarkably, we have shown both analytically and through simulations that the choices of the underlying models and assumptions play a fundamental role concerning the effects of replication in parallel systems, thus motivating our general framework. In terms of applications, we have developed a novel task replication policy in fork-join systems which is similar to the implementation of back-up tasks in MapReduce. For the analytically convenient Poisson arrivals and i.i.d. exponential service times model, our policy improves the performance of the standard fork-join model by a fundamental logarithmic factor.

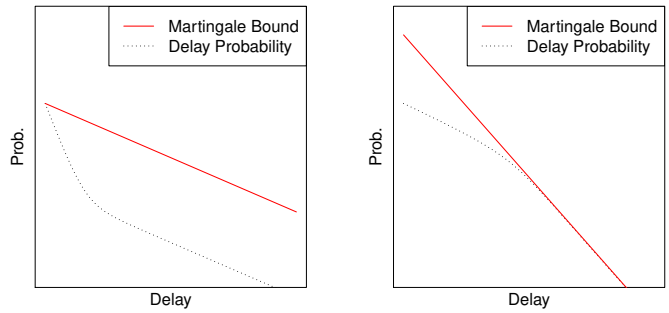
8

Concluding Remarks

8.1 On the Accuracy of the Martingale-Bounds

The crucial step in the derivations of all the performance bounds in the preceding chapters consists in invoking (some variant of) Doob's inequality, either for (super-)martingales (Chapters 3–7) or for submartingales (Chapter 5 and Chapter 6). In this section we discuss the tightness of the martingale-based method and provide some insight into reasons for differences of the bounds' accuracy.

Although the bounds illustrated in the Figures of Chapter 3 are seemingly accurate, the bounds degrade with the level of *correlations within the arrivals*. This trend can be particularly noticed for 1-order vs. 2-order autoregressive processes (see Figure 3.5(a) vs. 3.5(b)); the same could be observed by



(a) Initial rate larger than asymptotic rate

(b) Initial rate smaller than asymptotic rate

Figure 8.1: Possible CCDF of the delay. Depending on the flows' burstiness the martingale (exponential) bounds are inevitably loose for small or large delays.

reducing the scale of the x-axis in Figures 3.6(a) and 3.6(b). One explanation is that on a logarithmic y-axis the simulations throughout are seemingly convex, i.e., the probabilities in an initial phase decay faster than asymptotically (see Figure 8.1(a)), this behavior has been in fact convincingly shown to hold for bursty flows in [39]. In contrast, as the arrival- and service-martingales are based on exponential transforms, they *can only* render bounds of the form of the (generalized) exponential distribution (i.e., $\mathbb{P} \leq \kappa e^{-\theta x}$), whence the straight lines in the plots. In other words, the longer the "initial phase" of the true distribution is, or more generally the level of long-range correlations, the larger the gap is between the distribution and the obtained bounds.

A possible approach to reduce this inherent gap would be to use hyperexponential rather than exponential transforms, i.e., functions of the form $p_1 e^{\theta_1 x} + p_2 e^{\theta_2 x}$, where the parameters p_1, λ_1 and p_2, λ_2 are scaled accordingly to the initial and the tail periods, respectively.

The diametrically opposite situation occurs in the FJ queueing system (Chapter 6): for non-blocking systems, the simulations now have a concave shape on a logarithmic y-axis (see Figure 6.3(a)), and hence the (exponential) bounds are to some extend inaccurate in the initial phase but reasonably tight asymptotically (see Figure 8.1(b)). Moreover, for both blocking and non-blocking systems,

the bounds become more accurate at higher utilizations (see Figures 6.3 and 6.6). This behavior can be explained by the *correlation within the servers*: For the leading constant K (the number of servers) from Theorem 6.1, the union bound was utilized (see the second line of Eq. (6.12)), which is known to provide better estimates if the r.v.'s under consideration are rather uncorrelated. As a high link utilization translates into a comparably smaller impact of the common interarrival times t_i (i.e., the "dependent part"), the $(x_{k,i} - t_i)_{k\in[1,K]}$ become "more uncorrelated" and hence the gap between simulations and bounds is reduced.

8.2 Conclusion

In this thesis, we developed a general framework that combines the stochastic network calculus methodology with the powerful probabilistic tool of martingales. Concretely, the characteristics of a queueing system were captured by arrival- and service-martingales (Definitions 3.1 and 4.1), retaining the "modularity" property of SNC that information about the arrival and the service are encoded in two different objects. Whereas the arrival-martingales enable the analysis of queueing systems under scheduling (Chapter 3), and provide its first sharp per-flow delay bounds (Corollaries 3.8–3.11), the service-martingales allow for the analysis of more sophisticated service models like random access protocols (Chapter 4). Here, we provided the first rigorous and accurate delay analysis of Aloha and CSMA/CA networks, subject to Markovian arrivals (Corollaries 4.7 and 4.9).

Moreover, we demonstrated the versatility of the martingale approach by considering related queueing systems: For queueing systems with a random number of parallel flows (Chapter 5) we gave evidence that the "folk theorem" of queueing theory ("determinism minimizes the queue size"), can actually fail (Theorem 5.3). In the scenario of multi-server systems we provided non-asymptotic and computable bounds on the performance of fork-join queueing systems (Chapter 6) and systems with replications (Chapter 7), respectively.

The bounds provided in this thesis improve the corresponding bounds de-

rived with Boole's inequality by several orders of magnitude (see e.g, Figure 3.4); moreover, simulations indicate that they are reasonably tight, especially at hight utilizations (see e.g., Figure 4.8). Thus, we convincingly demonstrated that the inaccuracy of (state-of-the-art) SNC is mainly due to inappropriate probabilistic tools leveraged in its application, rather than to SNC itself. The revised *stochastic network calculus with martingales* could disprove the skepticism towards its practical relevance, and help establishing SNC as a valuable tool to the performance analysis of queueing systems.

Bibliography

[1] Amazon Elastic Compute Cloud EC2. `http://aws.amazon.com/ec2`.

[2] J. Abate, G. L. Choudhury, and W. Whitt. Waiting-time tail probabilities in queues with long-tail service-time distributions. *Queueing Systems*, 16(3-4):311–338, Sept. 1994.

[3] N. Abramson. The Aloha system: another alternative for computer communications. In *Proceedings of AFIPS Joint Computer Conferences*, pages 281–285, 1970.

[4] F. Alizadeh-Shabdiz and S. Subramaniam. Analytical models for single-hop and multi-hop ad hoc networks. In *ACM Broadnets*, pages 449–458, Oct. 2004.

[5] G. Ananthanarayanan, A. Ghodsi, S. Shenker, and I. Stoica. Effective straggler mitigation: Attack of the clones. In *10th USENIX Conference on Networked Systems Design and Implementation (NSDI)*, pages 185–198, 2013.

[6] D. Anick, D. Mitra, and M. Sondhi. Stochastic theory of a data-handling system with multiple sources. *Bell Systems Technical Journal*, 61(8):1871–1894, Oct. 1982.

[7] S. Asmussen, A. Frey, T. Rolski, and V. Schmidt. Does Markov-modulation increase the risk? *ASTIN Bulletin*, 25(1):49–66, May 1995.

[8] S. Asmussen and C. O'Cinneide. On the tail of the waiting time in a markov-modulated M/G/1 queue. *Operations Research*, 50(3):559–565, May-June 2002.

[9] S. Babu. Towards automatic optimization of MapReduce programs. In *Proc. of ACM SoCC*, pages 137–142, 2010.

[10] F. Baccelli, E. Gelenbe, and B. Plateau. An end-to-end approach to the resequencing problem. *J. ACM*, 31(3):474–485, June 1984.

[11] F. Baccelli, A. M. Makowski, and A. Shwartz. The Fork-Join queue and related systems with synchronization constraints: Stochastic ordering and computable bounds. *Adv. in Appl. Probab.*, 21(3):629–660, Sept. 1989.

[12] S. Balsamo, L. Donatiello, and N. M. Van Dijk. Bound performance models of heterogeneous parallel processing systems. *IEEE Trans. Parallel Distrib. Syst.*, 9(10):1041–1056, Oct. 1998.

[13] N. Bansal. Analysis of the M/G/1 processor-sharing queue with bulk arrivals. *Operations Research Letters*, 31(5):401 – 405, Sept. 2003.

[14] F. Baskett, K. M. Chandy, 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, Apr. 1975.

[15] N. Bäuerle and T. Rolski. A monotonicity result for the workload in Markov-modulated queues. *Journal of Applied Probability*, 35(3):741–747, Sept. 1998.

[16] A. W. Berger and W. Whitt. Effective bandwidths with priorities. *IEEE/ACM Transactions on Networking*, 6(4):447–460, Aug. 1998.

[17] S. Beuerman and E. Coyle. The delay characteristics of CSMA/CD networks. *IEEE Transactions on Communications*, 36(5):553–563, May 1988.

[18] G. Bianchi. Performance analysis of the IEEE 802.11 distributed coordination function. *IEEE Journal on Selected Areas in Communications*, 18(3):535–547, Mar. 2000.

[19] P. Billingsley. *Probability and Measure (3^{rd} Edition)*. Wiley, 1995.

[20] T. Bonald and L. Massoulié. Impact of fairness on internet performance. In *ACM Sigmetrics*, pages 82–91, 2001.

[21] T. Bonald and A. Proutière. Flow-level stability of utility-based allocations for non-convex rate regions. In *40th Annual Conference on Information Sciences and Systems*, pages 327–332, 2006.

[22] S. Borst, O. Boxma, J. F. Groote, and S. Mauw. Task allocation in a multi-server system. *Journal of Scheduling*, 6(5):423–436, Sept. 2003.

[23] D. Botvich and N. Duffield. Large deviations, economies of scale, and the shape of the loss curve in large multiplexers. *Queueing Systems*, 20(3-4):293–320, Sept. 1995.

[24] J.-Y. Le Boudec and P. Thiran. *Network Calculus*. Springer Verlag, Lecture Notes in Computer Science, LNCS 2050, 2001.

[25] O. Boxma, G. Koole, and Z. Liu. Queueing-theoretic solution methods for models of parallel and distributed systems. In *Proc. of Performance Evaluation of Parallel and Distributed Systems. CWI Tract 105*, pages 1–24, 1994.

[26] P. J. Brockwell and R. A. Davis. *Time Series: Theory and Methods*. Springer-Verlag, 1991.

[27] E. Buffet and N. G. Duffield. Exponential upper bounds via martingales for multiplexers with Markovian arrivals. *Journal of Applied Probability*, 31(4):1049–1060, Dec. 1994.

[28] A. Burchard, J. Liebeherr, and F. Ciucu. On superlinear scaling of network delays. *IEEE/ACM Transactions on Networking*, 19(4):1043–1056, Aug. 2011.

[29] A. Burchard, J. Liebeherr, and S. D. Patek. A min-plus calculus for end-to-end statistical service guarantees. *IEEE Transactions on Information Theory*, 52(9):4105–4114, Sept. 2006.

[30] P. J. Burke. Delays in single-server queues with batch input. *INFORMS-Operations Research*, 23(4):830–833, July–Aug. 1975.

[31] A. Bušić, J.-M. Fourneau, and N. Pekergin. Worst case analysis of batch arrivals with the increasing convex ordering. In A. Horváth and M. Telek, editors, *Formal Methods and Stochastic Models for Performance Evaluation*, volume 4054 of *Lecture Notes in Computer Science*, pages 196–210. Springer, 2006.

[32] J. P. Buzen. Computational algorithms for closed queueing networks with exponential servers. *Communications of the ACM*, 16(9):527–531, Sept. 1973.

[33] F. Cali, M. Conti, and E. Gregori. Dynamic tuning of the IEEE 802.11 protocol to achieve a theoretical throughput limit. *IEEE/ACM Transactions on Networking*, 8(6):785–799, Dec. 2000.

[34] M. Carvalho and J. Garcia-Luna-Aceves. Delay analysis of IEEE 802.11 in single-hop networks. In *IEEE International Conference on Network Protocols (ICNP)*, pages 146–155, Nov. 2003.

[35] C.-S. Chang. *Performance Guarantees in Communication Networks*. Springer Verlag, 2000.

[36] M. L. Chaudhry and J. G. C. Templeton. *A First Course in Bulk Queues*. John Wiley and Sons, 1983.

[37] Y. Chen, S. Alspaugh, and R. Katz. Interactive analytical processing in big data systems: A cross-industry study of mapreduce workloads. *Proc. VLDB Endow.*, 5(12):1802–1813, Aug. 2012.

[38] A. Cherny. Some particular problems of martingale theory. In Y. Kabanov, R. Liptser, and J. Stoyanov, editors, *From Stochastic Calculus to Mathematical Finance*, pages 109–124. Springer, 2006.

[39] G. Choudhury, D. Lucantoni, and W. Whitt. Squeezing the most out of ATM. *IEEE Transactions on Communications*, 44(2):203–217, Feb. 1996.

[40] 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.

[41] F. Ciucu, R. Khalili, Y. Jiang, L. Yang, and Y. Cui. Towards a system theoretic approach to wireless network capacity in finite time and space. In *IEEE Infocom*, pages 2391–2399, 2014.

[42] F. Ciucu and F. Poloczek. On multiplexing flows: Does it hurt or not? In *IEEE Infocom*, pages 1122–1130, May 2015.

[43] F. Ciucu, F. Poloczek, and O. Hohlfeld. On capacity dimensioning in dynamic scenarios: The key role of peak values. In *IEEE Lanman*, May 2014.

[44] F. Ciucu, F. Poloczek, and J. Schmitt. Stochastic upper and lower bounds for general markov fluids. In *International Teletraffic Congress (ITC)*. To appear.

[45] F. Ciucu, F. Poloczek, and J. Schmitt. Sharp bounds in stochastic network calculus. In *ACM Sigmetrics (Poster)*, pages 367–368, June 2013.

[46] F. Ciucu, F. Poloczek, and J. Schmitt. Sharp bounds in stochastic network calculus. *CoRR*, abs/1303.4114, 2013.

[47] F. Ciucu, F. Poloczek, and J. Schmitt. Sharp per-flow delay bounds for bursty arrivals: The case of FIFO, SP, and EDF scheduling. In *IEEE Infocom*, pages 1896–1904, Apr. 2014.

[48] F. Ciucu and J. Schmitt. Perspectives on network calculus - No free lunch but still good value. In *ACM Sigcomm*, 2012.

[49] C. Courcoubetis and R. Weber. Effective bandwidths for stationary sources. *Probability in Engineering and Informational Sciences*, 9(2):285–294, Apr. 1995.

[50] R. L. Cruz. SCED+: Efficient management of quality of service guarantees. In *IEEE Infocom*, pages 625–634, Apr. 1998.

[51] R. L. Cruz and C. Okino. Service guarantees for window flow control. In *34th Allerton Conference on Communications, Control and Computating*, Oct. 1996.

[52] J. Dean. [Online] Achieving rapid response times in large online services. Mar. 2012. Berkeley AMPLab Cloud Seminar, `http://research.google.com/people/jeff/latency.html`.

[53] J. Dean and L. A. Barroso. The tail at scale. *Communications of the ACM*, 56(2):74–80, Feb. 2013.

[54] J. Dean and S. Ghemawat. Mapreduce: Simplified data processing on large clusters. *Communications of the ACM*, 51(1):107–113, Jan. 2008.

[55] N. G. Duffield. Exponential bounds for queues with Markovian arrivals. *Queueing Systems*, 17(3-4):413–430, Sept. 1994.

[56] M. Durvy, O. Dousse, and P. Thiran. Self-organization properties of CSMA/CA systems and their consequences on fairness. *IEEE Transactions on Information Theory*, 55(3):931–943, Mar. 2009.

[57] A. Ephremides and B. E. Hajek. Information theory and communication networks: An unconsummated union. *IEEE Transactions on Information Theory*, 44(6):2416–2434, Oct. 1998.

[58] A. Ephremides, P. Varaiya, and J. Walrand. A simple dynamic routing problem. *IEEE Transactions on Automatic Control*, 25(4):690–693, Aug. 1980.

[59] M. Fidler. An end-to-end probabilistic network calculus with moment generating functions. In *IEEE International Workshop on Quality of Service (IWQoS)*, pages 261–270, 2006.

[60] M. Fidler. A survey of deterministic and stochastic service curve models in the network calculus. *IEEE Communications Surveys & Tutorials*, 12(1):59–86, Feb. 2010.

[61] M. Fidler and Y. Jiang. Non-asymptotic delay bounds for (k, l) fork-join systems and multi-stage fork-join networks. *CoRR*, abs/1512.08354, 2015.

[62] L. Flatto and S. Hahn. Two parallel queues created by arrivals with two demands I. *SIAM J. Appl. Math.*, 44(5):1041–1053, Oct. 1984.

[63] S. B. Fred, T. Bonald, A. Proutiere, G. Régnié, and J. W. Roberts. Statistical bandwidth sharing: a study of congestion at flow level. In *ACM Sigcomm*, pages 111–122, 2001.

[64] R. G. Gallager. A perspective on multiaccess channels. *IEEE Transactions on Information Theory*, 31(2):124–142, Mar. 1985.

[65] K. Gardner, S. Zbarsky, S. Doroudi, M. Harchol-Balter, and E. Hyytiä. Reducing latency via redundant requests: Exact analysis. In *ACM Sigmetrics*, pages 347–360, 2015.

[66] M. Garetto and C.-F. Chiasserini. Performance analysis of 802.11 WLANs under sporadic traffic. In *IFIP Networking*, pages 1343–1347, 2005.

[67] G. D. Ghare and S. T. Leutenegger. Improving speedup and response times by replicating parallel programs on a snow. In *10th International Conference on Job Scheduling Strategies for Parallel Processing (JSSPP)*, pages 264–287, 2004.

[68] R. J. Gibbens. Traffic characterisation and effective bandwidths for broadband network traces. *J. R. Stat. Soc. Ser. B. Stat. Methodol.*, pages 169–179, 1996.

[69] B. Hajek. The proof of a folk theorem on queuing delay with applications to routing in networks. *Journal of the ACM*, 30(4):834–851, Oct. 1983.

[70] Y. Han and A. Makowski. Resequencing delays under multipath routing - Asymptotics in a simple queueing model. In *Proc. of IEEE INFOCOM*, pages 1–12, Apr. 2006.

[71] M. Harchol-Balter. *Performance Modeling and Design of Computer Systems. Queueing Theory in Action*. Cambridge University Press, 2012.

[72] M. Harchol-Balter and D. Wolfe. Bounding delays in packet-routing networks. In *ACM Symposium on Theory of Computing (STOC)*, pages 248–257, 1995.

[73] G. Harrus and B. Plateau. Queueing analysis of a reordering issue. *IEEE Trans. Softw. Eng.*, 8(2):113–123, Mar. 1982.

[74] R. Heath and A. Paulraj. Switching between diversity and multiplexing in MIMO systems. *IEEE Transactions on Communications*, 53(6):962–968, June 2005.

[75] E. Heymann, M. Senar, E. Luque, and M. Livny. Evaluation of strategies to reduce the impact of machine reclaim in cycle-stealing environments. In *First IEEE/ACM International Symposium on Cluster Computing and the Grid*, pages 320–328, 2001.

[76] T. Hoff. [Online] Latency is everywhere and it costs you sales - how to crush it. July 2009. `http://highscalability.com/blog/2009/7/25/latency-is-everywhere-and-it-costs-you-sales-how-to-crush-it.html`.

[77] R. A. Horn and C. R. Johnson. *Matrix Analysis*. Cambridge University Press, New York, NY, USA, 2nd edition, 2012.

[78] P. A. Humblet. Determinism minimizes waiting time in queues. Technical report, MIT Laboratory for Information and Decision Systems, LIDS-P-1207, 1982.

[79] E. Hyytiä and S. Aalto. Round-robin routing policy: Value functions and mean performance with job- and server-specific costs. In *Proceedings of the 7th International Conference on Performance Evaluation Methodologies and Tools (ValueTools)*, pages 69–78, 2013.

[80] V. Jalaparti, P. Bodik, S. Kandula, I. Menache, M. Rybalkin, and C. Yan. Speeding up distributed request-response workflows. In *ACM SIGCOMM 2013*, pages 219–230, 2013.

[81] Y. Jiang and Y. Liu. *Stochastic Network Calculus*. Springer, 2008.

[82] G. Joshi, Y. Liu, and E. Soljanin. Coding for fast content download. In *Proc. of the Allerton Conference on Communication, Control, and Computing*, pages 326–333, 2012.

[83] G. Joshi, Y. Liu, and E. Soljanin. On the delay-storage trade-off in content download from coded distributed storage systems. *IEEE Journal on Selected Areas in Communications (JSAC)*, 32(5):989–997, May 2014.

[84] G. Joshi, E. Soljanin, and G. Wornell. Queues with redundancy: Latency-cost analysis. In *ACM Sigmetrics Workshop on MAthematical performance Modeling and Analysis (MAMA)*, 2015.

[85] S. Kandula, S. Sengupta, A. Greenberg, P. Patel, and R. Chaiken. The nature of data center traffic: Measurements & analysis. In *Internet Measurement Conference (IMC)*, IMC '09, pages 202–208, New York, NY, USA, 2009. ACM.

[86] S. Kavulya, J. Tan, R. Gandhi, and P. Narasimhan. An analysis of traces from a production MapReduce cluster. In *Proc. of IEEE/ACM CCGRID*, pages 94–103, May 2010.

[87] F. P. Kelly. Networks of queues with customers of different types. *Journal of Applied Probability*, 3(12):542–554, Sept. 1975.

[88] F. P. Kelly. *Reversibility and Stochastic Networks*. Wiley, Chichester, 1979.

[89] 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.

[90] B. Kemper and M. Mandjes. Mean sojourn times in two-queue Fork-Join systems: Bounds and approximations. *OR Spectr.*, 34(3):723–742, July 2012.

[91] G. Kesidis, B. Urgaonkar, Y. Shan, S. Kamarava, and J. Liebeherr. Network calculus for parallel processing. In *Proc. of the ACM MAMA workshop*, 2015.

[92] J. F. C. Kingman. A martingale inequality in the theory of queues. *Cambridge Philosophical Society*, 60(2):359–361, Apr. 1964.

[93] J. F. C. Kingman. Inequalities in the theory of queues. *Journal of the Royal Statistical Society, Series B*, 32(1):102–110, 1970.

[94] L. Kleinrock and F. A. Tobagi. Packet switching in radio channels: Part I–carrier sense multiple-access modes and their throughput-delay characteristics. *IEEE Transactions on Communications*, 23(12):1400–1416, Dec. 1975.

[95] S.-S. Ko and R. F. Serfozo. Sojourn times in G/M/1 Fork-Join networks. *Naval Res. Logist.*, 55(5):432–443, May 2008.

[96] G. Koole and R. Righter. Resource allocation in grid computing. *Journal of Scheduling*, 11(3):163–173, June 2007.

[97] S. Lam. *Packet Switching in a Multi-Access Broadcast Channel with Application to Satellite Communication in a Computer Network*. PhD thesis, University of California at Los Angeles, Los Angeles, CA, 1974.

[98] A. S. Lebrecht and W. J. Knottenbelt. Response time approximations in Fork-Join queues. In *Proc. of UKPEW*, July 2007.

[99] D. C. Lee and J. N. Tsitsiklis. The worst bulk arrival process to a queue. Technical report, MIT Laboratory for Information and Decision Systems, LIDS-P-2116, 1992.

[100] J. Liebeherr, A. Burchard, and F. Ciucu. Delay bounds in communication networks with heavy-tailed and self-similar traffic. *IEEE Transactions on Information Theory*, 58(2):1010–1024, Feb. 2012.

[101] 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.

[102] J. Liu, A. Proutière, Y. Yi, M. Chiang, and H. Poor. Stability, fairness, and performance: A flow-level study on nonconvex and time-varying rate regions. *IEEE Transactions on Information Theory*, 55(8):3437–3456, Aug. 2009.

[103] Z. Liu and D. Towsley. Optimality of the round-robin routing policy. *Journal of Applied Probability*, 31(2):466–475, 1994.

[104] R. M. Loynes. The stability of a queue with non-independent inter-arrival and service times. *Mathematical Proceedings of the Cambridge Philosophical Society*, 58(03):497–520, July 1962.

[105] R. Lübben and M. Fidler. Non-equilibrium information envelopes and the capacity-delay-error-tradeoff of source coding. *CoRR*, abs/1107.3087, 2011.

[106] A. M. Makowski and T. Philips. Simple proofs of some folk theorems for parallel queues. Technical report, Institute for Systems Research, ISR-TR-1989-37, 1989.

[107] L. Massoulié and J. Roberts. Bandwidth sharing and admission control for elastic traffic. *Telecommunication Systems*, 15(1-2):185–201, Nov. 2000.

[108] R. R. Mazumdar. *Performance Modeling, Loss Networks, and Statistical Multiplexing*. Synthesis Lectures on Communication Networks. Morgan & Claypool Publishers, 2009.

[109] R. Nelson and A. Tantawi. Approximate analysis of Fork/Join synchronization in parallel queues. *IEEE Trans. Computers*, 37(6):739–743, June 1988.

[110] R. Pike, S. Dorward, R. Griesemer, and S. Quinlan. Interpreting the data: Parallel analysis with Sawzall. *Sci. Program.*, 13(4):277–298, Oct. 2005.

[111] I. Polato, R. R, A. Goldman, and F. Kon. A comprehensive view of Hadoop research - a systematic literature review. *J. Netw. Comput. Appl.*, 46(0):1 – 25, Nov. 2014.

[112] F. Poloczek and F. Ciucu. A martingale-envelope and applications. In *Proc. of the ACM MAMA workshop*, 2013.

[113] F. Poloczek and F. Ciucu. Scheduling analysis with martingales. *Performance Evaluation*, 79:56 – 72, Sept. 2014. Special Issue: Performance 2014.

[114] F. Poloczek and F. Ciucu. Service-martingales: Theory and applications to the delay analysis of random access protocols. In *IEEE Infocom*, pages 945–953, May 2015.

[115] F. Poloczek and F. Ciucu. Contrasting effects of replication in parallel systems: From overload to underload and back. In *ACM Sigmetrics (Poster)*, pages 375–376, June 2016.

[116] F. Poloczek and F. Ciucu. Contrasting effects of replication in parallel systems: From overload to underload and back. *CoRR*, abs/1602.07978, Feb. 2016.

[117] C. Raiciu, S. Barre, C. Pluntke, A. Greenhalgh, D. Wischik, and M. Handley. Improving datacenter performance and robustness with multipath TCP. *SIGCOMM Comput. Commun. Rev.*, 41(4):266–277, Aug. 2011.

[118] M. Reiser and S. S. Lavenberg. Mean-value analysis of closed multichain queuing networks. *Journal of the ACM*, 27(2):313–322, Apr. 1980.

[119] A. Rényi. On the theory of order statistics. *Acta Math. Acad. Sci. Hungarica*, 4:191–232, 1953.

[120] A. Rizk, F. Poloczek, and F. Ciucu. Stochastic bounds in fork-join queueing systems under full and partial mapping. *Queueing Systems*, to appear.

[121] A. Rizk, F. Poloczek, and F. Ciucu. Computable bounds in fork-join queueing systems. In *ACM Sigmetrics*, pages 335–346, June 2015.

[122] B. Rogozin. Some extremal problems in the theory of mass service. *Theory of Probability & Its Applications*, 11(1):144–151, 1966.

[123] S. M. Ross. Bounds on the delay distribution in GI/G/1 queues. *Journal of Applied Probability*, 11(2):417–421, June 1974.

[124] S. M. Ross. Average delay in queues with non-stationary Poisson arrivals. *Journal of Applied Probability*, 15(3):602–609, Sept. 1978.

[125] M. Schleyer. An analytical method for the calculation of the waiting time distribution of a discrete time G/G/1-queueing system with batch arrivals. *OR Spectrum*, 29(4):745–763, Oct. 2007.

[126] F. B. Schneider. Implementing fault-tolerant services using the state machine approach: A tutorial. *ACM Computing Surveys*, 22(4):299–319, Dec. 1990.

[127] E. Schurman and J. Brutlag. The user and business impact of server delays, additional bytes and HTTP chunking in web search. *O'Reilly Velocity Web Performance and Operations Conference*, June 2009.

[128] N. Shah, K. Lee, and K. Ramchandran. When do redundant requests reduce latency ? In *51st Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 731–738, Oct. 2013.

[129] M. Shaked and J. G. Shanthikumar. *Stochastic Orders*. Springer, 2007.

[130] N. B. Shroff and M. Schwartz. Improved loss calculations at an ATM multiplexer. *IEEE/ACM Transactions on Networking*, 6(4):411–421, Aug. 1998.

[131] V. Sivaraman and F. M. Chiussi. Statistical analysis of delay bound violations at an earliest deadline first scheduler. *Performance Evaluation*, 36(1):457–470, Aug. 1999.

[132] S. Souders. [Online] Velocity and the bottom line. July 2009. `http://radar.oreilly.com/2009/07/velocity-making-your-site-fast.html`.

[133] T. E. Stern and A. I. Elwalid. Analysis of separable Markov-modulated rate models for information-handling systems. *Advances in Applied Probability*, 23(1):105–139, Mar. 1991.

[134] C. Stewart, A. Chakrabarti, and R. Griffith. Zoolander: Efficiently meeting very strict, low-latency slos. In *10th International Conference on Autonomic Computing (ICAC 13)*, pages 265–277, 2013.

[135] H. Takagi and L. Kleinrock. Throughput analysis for persistent CSMA systems. *IEEE Transactions on Communications*, COM-33(7):627–638, July 1985.

[136] H. Takagi and Y. Takahashi. Priority queues with batch Poisson arrivals. *Operations Research Letters*, 10(4):225–232, June 1991.

[137] M. Talagrand. Majorizing measures: The generic chaining. *Annals of Probability*, 24(3):1049–1103, July 1996.

[138] J. Tan, X. Meng, and L. Zhang. Delay tails in MapReduce scheduling. *SIGMETRICS Perform. Eval. Rev.*, 40(1):5–16, June 2012.

[139] J. Tan, Y. Wang, W. Yu, and L. Zhang. Non-work-conserving effects in MapReduce: Diffusion limit and criticality. *SIGMETRICS Perform. Eval. Rev.*, 42(1):181–192, June 2014.

[140] O. Tickoo and B. Sikdar. Queueing analysis and delay mitigation in IEEE 802.11 random access MAC based wireless networks. In *IEEE Infocom*, pages 1404–1413, Mar. 2004.

[141] F. A. Tobagi. Distributions of packet delay and interdeparture time in slotted Aloha and carrier sense multiple access. *Journal of the ACM*, 29(4):907–927, Oct. 1982.

[142] E. Varki. Mean value technique for closed Fork-Join networks. *SIGMETRICS Perform. Eval. Rev.*, 27(1):103–112, May 1999.

[143] S. Varma and A. M. Makowski. Interpolation approximations for symmetric Fork-Join queues. *Perform. Eval.*, 20(1–3):245–265, May 1994.

[144] G. de Veciana, T.-J. Lee, and T. Konstantopoulos. Stability and performance analysis of networks supporting services with rate control - could the internet be unstable? In *IEEE Infocom*, pages 802–810, 1999.

[145] E. Vianna, G. Comarela, T. Pontes, J. Almeida, V. Almeida, K. Wilkinson, H. Kuno, and U. Dayal. Analytical performance models for MapReduce workloads. *Int. J. Parallel Prog.*, 41(4):495–525, Aug. 2013.

[146] A. Vulimiri, P. B. Godfrey, R. Mittal, J. Sherry, S. Ratnasamy, and S. Shenker. Low latency via redundancy. In *Proceedings of the Ninth*

ACM Conference on Emerging Networking Experiments and Technologies (CoNext), pages 283–294, 2013.

[147] J. Walrand. *An introduction to queueing networks*. Prentice Hall, 1988.

[148] D. Wang, G. Joshi, and G. W. Wornell. Using straggler replication to reduce latency in large-scale parallel computing (extended version). *CoRR*, abs/1503.03128, 2015.

[149] G. Weiss. Time-reversibility of linear stochastic processes. *Journal of Applied Probability*, 12(4):831–836, Dec. 1975.

[150] T. White. *Hadoop: The Definitive Guide*. O'Reilly, 1st edition, 2009.

[151] W. Whitt. The effect of variability in the GI/G/s queue. *Journal of Applied Probability*, 17(4):1062–1071, 1980.

[152] W. Whitt. Comparison conjectures about the M/G/s queue. *Operations Research Letters*, 2(5):203–210, Dec. 1983.

[153] W. Whitt. On approximations for queues, I: Extremal distributions. Technical report, Institute for Systems Research, ISR-TR-1989-37, 1989.

[154] D. Williams. *Probability with Martingales*. Cambridge University Press, 1991.

[155] D. Wischik. Sample path large deviations for queues with many inputs. *Annals of Applied Probability*, 11(2):379–404, May 2001.

[156] Z. Wu, C. Yu, and H. V. Madhyastha. CosTLO: Cost-effective redundancy for lower latency variance on cloud storage services. In *12th USENIX Symposium on Networked Systems Design and Implementation (NSDI 15)*, pages 543–557, May 2015.

[157] Y. Xia and D. Tse. On the large deviation of resequencing queue size: 2-M/M/1 case. *IEEE Trans. Inf. Theory*, 54(9):4107–4118, Sept. 2008.

[158] H. Xu and B. Li. Repflow: Minimizing flow completion times with replicated flows in data centers. In *IEEE Infocom*, pages 1581–1589, 2014.

[159] Y. Yang and T.-S. P. Yum. Delay distributions of slotted Aloha and CSMA. *IEEE Transactions on Communications*, 51(11):1846–1857, Nov. 2003.

[160] D. D. W. Yao, M. L. Chaudhry, and J. G. C. Templeton. On bounds for bulk arrival queues. *European Journal of Operational Research*, 15(2):237 – 243, Feb. 1984.

[161] M. Zaharia, A. Konwinski, A. D. Joseph, R. Katz, and I. Stoica. Improving MapReduce performance in heterogeneous environments. In *Proc. of USENIX OSDI*, pages 29–42, Dec. 2008.

[162] K. Zheng, F. Liu, L. Lei, C. Lin, and Y. Jiang. Stochastic performance analysis of a wireless finite-state Markov channel. *IEEE Transactions on Wireless Communications*, 12(2):782–793, Feb. 2013.

[163] H. Al-Zubaidy, J. Liebeherr, and A. Burchard. A (min, $\times$) network calculus for multi-hop fading channels. In *IEEE Infocom*, 2013.