Scheduling for the tail: Robustness versus Optimality

Jayakrishnan Nair[1], Adam Wierman[2], and Bert Zwart[3]

[1]Department of Electrical Engineering, California Institute of Technology
[2]Computing and Mathematical Sciences Department, California Institute of Technology
[3]CWI Amsterdam, VU University Amsterdam, Eurandom & Georgia Tech

***Abstract*—When scheduling to minimize the sojourn time tail, the goals of optimality and robustness are seemingly at odds. Over the last decade, results have emerged which show that scheduling disciplines that are near-optimal under light (exponential) tailed workload distributions do not perform well under heavy (power) tailed workload distributions, and vice-versa. Very recently, it has been shown that this conflict between optimality and robustness is fundamental, i.e., no policy that does not learn information about the workload can be optimal across both light-tailed and heavy-tailed workloads. In this paper we show that one can exploit very limited workload information (the system load) in order to design a scheduler that provides robust performance across heavy-tailed and light-tailed workloads.**

I. Introduction

In the analysis of scheduling policies, the conventional performance metric has been the mean sojourn time (a.k.a. response time, flow time). In this context, it is well known that SRPT (Shortest Remaining Processing Time) scheduling minimizes the mean response time [1], regardless of the inter-arrival time and job size distributions.

However, guaranteeing good average case performance is typically insufficient. Motivated by quality of service considerations, we would also like to minimize the occurrence of very large sojourn times. Consequently, there has been considerable research interest in characterizing and optimizing the tail of the sojourn time distribution. Interestingly, this work highlights the following dichotomy: when the job size distribution is light-tailed, First Come First Served (FCFS) scheduling minimizes the sojourn time tail, whereas if the job size distribution is heavy-tailed, then scheduling policies like SRPT, Processor Sharing (PS), and Pre-emptive Last Come First Served (PLCFS) minimize the sojourn time tail.

Unlike in the case of optimizing the mean sojourn time, no scheduling policy is known to be optimal for the sojourn time tail across heavy-tailed and light-tailed job size distributions. Recently, Wierman & Zwart established that no such policy exists [2]. Specifically, they proved that no scheduling policy that does not learn information about the workload can be optimal for the sojourn time tail across heavy-tailed and light-tailed job size distributions; see Section III. Moreover, all scheduling policies that are known to be tail-optimal under heavy-tailed job sizes actually produce the worst possible tail behavior under light-tailed job sizes, and vice-versa.

To summarize, the literature understands how to design a scheduling policy that is optimal for the sojourn time tail for a given workload, but cannot design one that is *robust*, even minimally so, across heavy-tailed and light-tailed job size distributions.

In this paper, we show how to use partial workload information, specifically, the system load, to design a scheduler that is guarantees better than worst case sojourn time tail performance over a large class of heavy-tailed and light-tailed workloads. Additionally, the scheduler is optimal for the sojourn time tail over large subclasses of these workloads. We point out here that learning the load involves learning expectations, which is much easier than learning the tail of the job size distribution. Moreover, as we will see, our designs are robust to estimation errors in the load.

This paper is organized as follows. In Section II, we present a brief survey of the literature pertaining to sojourn time tail asymptotics in a GI/GI/1 setting, with an emphasis on the known optimality results. In Section III, we describe the recent result by Wierman & Zwart [2], which establishes the impossibility of non-learning scheduling policies that provide robustly optimal sojourn time tail performance across heavy-tailed and light-tailed job size distributions. We then describe our tail-robust scheduler design in Section III; for details, we refer the reader to [3]. Finally, we present a preliminary study of the performance of our tail-robust scheduler design with respect to expected sojourn time via simulations in Section V. In the remainder of this section, we introduce our model and notation.

For any non-negative random variable X, $F_X(\cdot)$ denotes the distribution function (d.f.) of X, i.e., $F_X(x) = P\left(X \leq x\right)$; $\Phi_X(\cdot)$ denotes the moment generating function of X, i.e., $\Phi_X(s) = \mathbb{E}\left[e^{sX}\right]$. For functions $\varphi(x)$ and $\xi(x)$, the notation $\varphi(x) \sim \xi(x)$ means $\lim_{x\to\infty} \frac{\varphi(x)}{\xi(x)} = 1$.

We will focus on the GI/GI/1 queue. Jobs arrive according to a renewal process; let A denote a generic inter-arrival time. Each job has an independent, identically distributed service requirement (size); let B denote a generic job size. The tuple (A, B) characterizes the workload. We will say that the workload is heavy-tailed (respectively, light-tailed) if the corresponding job size distribution is heavy-tailed (respectively, light-tailed). Without loss of generality, the server speed is taken to be unity. We make the following standard assumptions throughout: (i) load $\rho := \frac{\mathbb{E}[B]}{\mathbb{E}[A]} \in (0,1)$, (ii) $P(B > A) > 0$ (otherwise there would be no queueing). The *sojourn time* (response time) of a job refers to the time between its arrival and its departure. V_π denotes a random variable distributed as per the stationary sojourn time in a GI/GI/1 queue operating under scheduling discipline (policy) π.

Let Π denote the class of work-conserving, non-anticipative, and non-learning scheduling policies. By 'non-anticipative', we mean that the scheduling decision at time t cannot depend on arrival events after time t. By 'non-learning', we mean the scheduler cannot learn any distributional properties of the inter-arrival time or job size d.f.

II. Optimality

There is a vast body of literature analyzing the tail asymptotics of the stationary sojourn time distribution in a single server queue. The survey by Boxma & Zwart [4] provides an excellent overview. In this section, we present a brief review of the results with the specific goal of highlighting the optimality properties of common policies in the literature. Specifically, we focus on the following classic scheduling policies: FCFS, PLCFS, PS, and SRPT.

There are a number of different notions of optimality that have been considered in the literature, e.g., [4], [2]. Three of particular interest are the following. Note that the workload, specified by the distributions of A and B defines the probability measure P.

Definition 1. *For a class $\mathcal{P}$ of workloads, policy $\pi^* \in \Pi$ is*

1) ***weakly tail-optimal*** *if*
$$\limsup_{x\to\infty} \frac{P(V_{\pi^*} > x)^{1+\epsilon}}{P(V_\pi > x)} < \infty$$
for all $\epsilon > 0$, $P \in \mathcal{P}$ and $\pi \in \Pi$,

2) ***tail-optimal*** *if*
$$\limsup_{x\to\infty} \frac{P(V_{\pi^*} > x)}{P(V_\pi > x)} < \infty$$
for all $P \in \mathcal{P}$ and $\pi \in \Pi$,

3) ***strongly tail-optimal*** *if*
$$\limsup_{x\to\infty} \frac{P(V_{\pi^*} > x)}{P(V_\pi > x)} \le 1$$
for all $P \in \mathcal{P}$ and $\pi \in \Pi$.

It is easy to check that strong tail-optimality implies tail-optimality, which implies weak tail-optimality.

In the remainder of the section, we consider separately the cases of heavy-tailed and light-tailed job sizes, since the results tend to be qualitatively different across these two settings.

First, note that for any work conserving scheduling policy π, V_π may be stochastically bounded as follows:
$$B \le_{\mathrm{st}} V_\pi \le_{\mathrm{st}} Z^*,^1$$
where Z^* denotes the total time to emptiness of the queue in steady state, just after an arrival.

A. Tail asymptotics under heavy-tailed job sizes

Heavy-tailed job sizes have received significant interest over the past decade due to the fact that they are commonly observed in computer applications. A non-negative random variable X (or its d.f. F_X) is defined to be *heavy-tailed* if $\Phi_X(s) = \infty$ for all $s > 0$. Unfortunately, the class of all heavy-tailed distributions tends to be too broad to handle analytically, and so a majority of the literature focuses on a particular subclass of heavy-tailed distributions: regularly varying distributions. A random variable X (or its d.f. F_X) is said to be *regularly varying* with index $\theta > 1$ (denoted $X \in \mathcal{RV}(\theta)$) if $P(X > x) = x^{-\theta} L(x)$, where $L(x)$ is a slowly varying function, i.e., $L(x)$ satisfies $\lim_{x\to\infty} \frac{L(xy)}{L(x)} = 1 \ \forall\, y > 0$. Regularly varying distributions are a generalization of the class of Pareto/Ziph/power-law/scale-free distributions and constitute an important and analytically tractable subclass of the set of heavy-tailed distributions.

For the GI/GI/1 queue with regularly varying job sizes, i.e., $B \in \mathcal{RV}(\theta)$, the sojourn time tail asymptotics of nearly all common policies are understood. In particular, it is known that
$$\begin{aligned} &P(V_{PS} > x) \sim P(V_{SRPT} > x) \sim P(B > x(1-\rho)), \\ &P(V_{PLCFS} > x) \sim \mathbb{E}[N]\, P(B > x(1-\rho)), \\ &P(V_{FCFS} > x) \sim \tfrac{\rho}{1-\rho} \tfrac{1}{\theta-1} x P(B > x), \end{aligned} \tag{1}$$
where N denotes the number of jobs seen by the system in a busy period. See the survey by Boxma & Zwart [4] for details on the derivation of these results.

Observe that, since $P(B > x(1-\rho)) \sim (1-\rho)^{-\theta} P(B > x)$, (1) implies that under PS, SRPT, and PLCFS, the sojourn time tail is asymptotically a constant times the tail of the job size distribution. Therefore, *over the class of workloads with regularly varying job sizes, PS, SRPT and PLCFS are tail-optimal.*

On the other hand, the tail of V_{FCFS} is asymptotically one degree heavier than the tail of the job size distribution. In fact, all non-preemptive policies have the same tail asymptotics, up to a constant factor [5]. However, this is the worst possible tail behavior possible among work-conserving policies, since the random variable Z^* has a tail of the same degree [4].

As is typical of tail-asymptotics, the results summarized in (1) have a useful heuristic explanation: Under PS and SRPT, a tagged job has a very large sojourn time most likely because its own size B is very large. Such a tagged job receives service at average rate $1-\rho$, which makes its sojourn time $V \approx B/(1-\rho)$. Under PLCFS, the sojourn time of a tagged job equals the busy period started by it, which in turn is very large most likely because one job with a very large size arrived within that busy period. In contrast, under FCFS, a tagged job has a very large sojourn time most likely because of the presence of a job with a very large size in the queue ahead of it. This causes the sojourn time tail to be asymptotically proportional to the tail of the residual lifetime of the job size d.f., which is one degree heavier.

[1] For non-negative random variables X and Y, $X \le_{\mathrm{st}} Y$ if $P(X > x) \le P(Y > x)$ for all $x \ge 0$.

B. Tail asymptotics under light-tailed job sizes

Though heavy-tailed distributions have dominated the literature in the last decade, there are many situations that are naturally modeled by light-tailed distributions, e.g., in many manufacturing systems. A non-negative random variable X (or its d.f. F_X) is defined to be *light-tailed* if it is not heavy-tailed, i.e., if $\Phi_X(s) < \infty$ for some $s > 0$. When the job size distribution is light-tailed, the sojourn time d.f. is also typically light-tailed. We will describe the (logarithmic) asymptotic tail behavior of the sojourn time d.f. by its *decay rate*, defined as

$$\gamma(V) := \lim_{x\to\infty} -\frac{\log P\left(V > x\right)}{x},$$

when the limit exists.

In describing the sojourn time asymptotics with light-tailed job sizes, the following function plays a key role. For $s \geq 0$, $\Psi(s) := -\Phi_A^{-1}\left(\frac{1}{\Phi_B(s)}\right)$. Let $A(x)$ denote the total work entering the system in the interval $(0, x]$, assuming an arrival at time 0. The following lemma gives an interpretation to the function $\Psi(\cdot)$.

Lemma 1 (Mandjes & Zwart [6])**.** *For* $s \geq 0$, $\lim_{x\to\infty} \frac{\log \mathbb{E}\left[e^{sA(x)}\right]}{x} = \Psi(s)$. *Further,* $\Psi(s)$ *is strictly convex and lower semi-continuous.*

For the GI/GI/1 queue with light-tailed job sizes,

$$\begin{aligned}\gamma(V_{FCFS}) &= \gamma_F := \sup\{s \geq 0 \ : \ \Psi(s) - s \leq 0\},\\ \gamma(V_{PLCFS}) &= \gamma_L := \sup_{s\geq 0}\{s - \Psi(s)\};\end{aligned}$$

see Nuyens & Zwart [7]. From the strict concavity of $s - \Psi(s)$, and since $\Psi'(0) = \rho$, it is easy to show that $\gamma_L < (1-\rho)\gamma_F$. This implies the stationary sojourn time tail under FCFS is ‘lighter’ than that under PLCFS. In fact, it has been proved by Ramanan & Stolyar [8] that *FCFS is weakly tail-optimal over the class of workloads with light-tailed job sizes*.

Moreover, it has been proved by Nuyens et al. [9] that $\gamma(Z^*) = \gamma_L$, implying that PLCFS produces the worst possible sojourn time decay rate under light-tailed job sizes. Under mild regularity conditions, we additionally have

$$\gamma(V_{PS}) = \gamma(V_{SRPT}) = \gamma_L.$$

As in the heavy-tailed case, there are useful heuristic explanations of the asymptotic results in this setting. In particular, across all the policies discussed, a tagged job experiences a large sojourn time due to a combination of one or more of the following effects:

(i) there is a large backlog in the system when the tagged job arrives,
(ii) the tagged job has a large size,
(iii) a large amount of work enters the system after the arrival of the tagged job.

Under FCFS, very large sojourn times are most likely caused by effect (i), whereas under PS, SRPT, and PLCFS, very large sojourn times are most likely caused by a combination of effects (ii) and (iii).

The above discussion highlights the dichotomy described in the introduction; the policies that produce the best possible sojourn time tail behavior under heavy-tailed job size distributions produce the worst possible sojourn time tail behavior under light-tailed job size distributions, and vice-versa.

III. Robustness versus Optimality

As we discussed in Section I, SRPT scheduling minimizes the mean sojourn time regardless of assumptions about the inter-arrival time and job size distributions. In other words, SRPT is optimal *and* robust for the mean. In this paper, we are interested in the problem of designing a scheduler that provides similar robustly optimal performance in minimizing the sojourn time tail. Stated formally, we seek a scheduling policy satisfies the following strong tail-robustness criterion:

Definition 2. *A policy* $\pi^* \in \Pi$ *is* ***strongly tail-robust****, if it is weakly tail-optimal over the class of all workloads with heavy-tailed as well as light-tailed job size distributions.*

As was discussed in Section II, none of the scheduling policies that have been analyzed in the literature is strongly tail-robust. In fact, the policies that are known to be tail-optimal under heavy-tailed (specifically, regularly varying) job sizes actually produce the worst possible sojourn time tail behavior under light-tailed job sizes, and vice-versa. Therefore, whether or not there even exists a strongly tail-robust scheduling policy has been a long-standing open question. This question was recently resolved in the negative by Wierman & Zwart [2]. They proved that there is a fundamental limit on the sojourn time tail performance of non-learning scheduling policies:

Theorem 2. *There does not exist a work-conserving, non-anticipative, and non-learning scheduling policy that is strongly tail-robust.*

In order to prove Theorem 2, [2] first proves a necessary condition for a policy $\pi \in \Pi$ to be tail-optimal over heavy-tailed workloads. The condition is based on the following intuition: *to be tail-optimal over heavy-tailed workloads, a scheduling policy should be able to maintain queue stability on the introduction of an infinite sized job*, which was suggested in a number of prior papers [10], [4], [9]. To formalize this intuition, assume that a job of size B_0 enters an empty system at time 0. For $t \geq 0$, let $R(t)$ denote the total service allocated in the interval $[0, t]$ to jobs arriving after time 0. Then, the following is a necessary condition for a policy $\pi \in \Pi$ to be tail-optimal over workloads with heavy-tailed workloads [2]:

$$\lim_{x\to\infty} P(R(x) > (\rho - \delta)x | B_0 > (1-\rho)yx) = 1 \qquad \forall\ \delta > 0,\ y > 1. \tag{2}$$

The reader may verify that all the scheduling policies that are known to be tail-optimal over heavy-tailed workloads satisfy this condition.

The proof of Theorem 2 is completed by showing that (2) is incompatible with optimality over light-tailed workloads.

Specifically, a probability measure with light-tailed job sizes is constructed using an exponential change of measure starting with a measure corresponding to a suitably defined heavy-tailed workload. It is then shown that condition (2) on the probability measure with heavy-tailed job sizes implies non-competitiveness for the constructed probability measure with light-tailed job sizes. See [2] for the details.

To summarize, Theorem 2 implies that it is impossible for a non-learning scheduling policy to be strongly tail-robust across all light-tailed and heavy-tailed workloads. Moreover, among the scheduling policies that have been analyzed in the literature, those that produce optimal sojourn time tail behavior under heavy-tailed workloads produce the worst possible sojourn time tail behavior under light-tailed workloads, and vice-versa.

This state of the art raises several questions about the fundamental limits of tail-robust scheduling: What tradeoffs between optimality and robustness are achievable? Is it possible for a non-learning scheduler to provide 'close to optimal' sojourn time tail performance over large subsets of heavy-tailed and light-tailed workloads? Can partial workload information, e.g., system load, be used to be achieve tail-robustness?

Most of the above questions are still open, however, in the following section, we describe recent work that takes a step towards answering the last question above. In particular, we describe a scheduler that uses limited processor sharing (LPS) to guarantee a weak form of tail-robustness over large subclasses of heavy-tailed and light-tailed workloads using only an estimate of the system load.

IV. Tail-robust scheduling via Limited Processor Sharing

In this section, we describe how a tail-robust scheduler can be designed using Limited Processor Sharing (LPS-c). The results reported in this section appear in [3].

Under LPS-c, there is a limited multiprogramming level c, which determines the maximum number of jobs that may simultaneously receive service. Specifically, if there are n jobs in the system, then the server capacity is shared equally among the $\min(n, c)$ jobs which arrived earliest.

LPS-c is a natural candidate for our goal of tail-robust scheduling because, as c grows from 1 to ∞, LPS-c transitions from FCFS, which is optimal under light-tailed job sizes, to PS, which is optimal under heavy-tailed job sizes. We seek to design an intermediate value of c that is tail-robust. To achieve tail-robustness, it turns out that the choice of c must incorporate some workload information. We show that a tail-robust c can be designed as a function of only the system load ρ. We believe this is not a serious limitation, since estimating the load is also important to guarantee system stability. Moreover, as we shall see, our design is robust to estimation errors in ρ.

Before we state our main results, we define the following weak notion of tail-robustness.

Definition 3. *For a class $\mathcal{P}$ of workloads, policy π^* is* weakly tail-robust *if there exists a policy $\pi \in \Pi$ such that*

$$\lim_{x \to \infty} \frac{P(V_{\pi^*} > x)}{P(V_\pi > x)} = \infty$$

for all $P \in \mathcal{P}$.

Thus, a policy π^* is weakly tail-robust if it produces 'better-than-worst-case' sojourn time tail performance over the class of workloads under consideration. The schedulers we describe in this section are weakly tail-robust over a large class of heavy-tailed as well as light-tailed workloads.

A. Main result

We now state our main result concerning the tail-robust choice of c.

Proposition 3 ([3])**.** *Consider the GI/GI/1 queue. Let $\Theta \in (1, 2]$. LPS-c with $c = \left\lfloor \frac{\lceil \frac{\Theta}{\Theta-1} \rceil - 1}{1-\rho} \right\rfloor + 1$ is*

(a) *weakly tail-robust over the class of workloads where the job size distribution is regularly varying with index $\theta > 1$,*

(b) *weakly tail-robust over the class of workloads where the job size distribution is phase-type,*

(c) *weakly tail-optimal over the class of workloads where the job size distribution is regularly varying with index $\theta \geq \Theta$,*

(d) *weakly tail-optimal over the class of light-tailed workloads satisfying*

$$\frac{\gamma(B)}{\gamma(V_{FCFS})} \geq \left\lfloor \frac{\left\lceil \frac{\Theta}{\Theta-1} \right\rceil - 1}{1-\rho} \right\rfloor + 1$$

or $\gamma(B) = \infty$.

Note that our scheduler has a design parameter $\Theta \in (1, 2]$. Irrespective of the value of Θ, the scheduler guarantees better than worst case sojourn time tail performance over workloads with regularly varying job sizes as well as phase-type distributed job sizes. Moreover, the scheduler guarantees optimal tail performance over a large subset of these workloads. The parameter Θ allows one to tradeoff between the optimality regions in the light-tailed and heavy-tailed regimes. Specifically, increasing Θ decreases the set of heavy-tailed workloads over which the scheduler guarantees optimal sojourn time tail performance, while increasing the corresponding set among the light-tailed workloads.

The case $\Theta = 2$ is of special interest. Note that this is the most 'light-tailed centric' choice of Θ. In this case, $c = \left\lfloor \frac{1}{1-\rho} \right\rfloor + 1$. The optimality region among heavy-tailed workloads includes those with regularly varying job sizes with finite variance. The optimality region among the light-tailed workloads includes those that satisfy $\frac{\gamma(B)}{\gamma(V_{FCFS})} \geq \left\lfloor \frac{1}{1-\rho} \right\rfloor + 1$. It is interesting to note that the M/M workloads (inter-arrival times as well as job sizes are exponentially distributed) are on the boundary of this region, since, for the M/M/1 queue, $\gamma(B)/\gamma(V_{FCFS}) = \frac{1}{1-\rho}$.

A practical remark about our tail-robust scheduler design is that, although it requires learning the system load ρ, it is robust to estimation errors. Specifically, so long as the estimate used is an upper bound on the true load, the weak tail-robustness guarantees given by statements (a) and (b) of Proposition 3 still hold. Only the optimality regions among the heavy-tailed and light-tailed workloads change.

The proof of Proposition 3 depends on an analysis of the sojourn time tail asymptotics under in an LPS-c queue with both heavy-tailed and light-tailed job sizes. We describe next the relevant results from these analyses.

B. Tail asymptotics under heavy-tailed job sizes

In this section, we consider the case of regularly varying job sizes. We will describe the (logarithmic) asymptotics of the sojourn time tail using its *tail-index*, defined as

$$\Gamma(V) := \lim_{x\to\infty} -\frac{\log P(V > x)}{\log(x)},$$

when the limit exists. Note that a greater tail index implies a lighter tail. For technical reasons, we make the following assumption in our analysis.

Assumption 1. *$c\rho$ is not an integer, i.e., $\lfloor c\rho \rfloor < c\rho$.*

The following theorem describes the sojourn time tail index under LPS-c; see [3] for the proof.

Theorem 4. *Consider the GI/GI/1 queue. Under Assumption 1, if $B \in \mathcal{RV}(\theta)$ for $\theta > 1$, then*

$$\Gamma(V_{LPS-c}) = \min\left\{\theta, (\theta-1)(c - \lfloor c\rho \rfloor)\right\}. \tag{3}$$

It is natural to view the GI/GI/1 LPS-c queue as a work-conserving version of a GI/GI/c FCFS queue, where each server has speed $1/c$. Theorem 4 implies the (logarithmic) sojourn time tail asymptotics in these two queues are identical. In fact, our proof of Theorem 4 relies on the parallels between these two queues. Define $k = k_c := c - \lfloor c\rho \rfloor$. We refer to k as the number of 'spare slots,' since k is the minimum number of infinite sized jobs that must be introduced into the LPS-c queue to make it unstable. This definition parallels the notion of 'spare servers' which is used in the context of the GI/GI/c FCFS queue and has been shown to determine the moment conditions for the waiting time (delay) in that system [11]. In fact, in both the GI/GI/1 LPS-c queue and the GI/GI/c FCFS queue, a tagged job faces a very long waiting time most likely because of k very large jobs in queue ahead of it.

Let us now interpret (3). For regularly varying job sizes, (1) implies that

1) $\Gamma(V_{FCFS}) = \theta - 1$ (this is the worst possible tail index among policies in Π),
2) $\Gamma(V_{PS}) = \theta$ (this is the optimal tail index among policies in Π).

Now, (3) implies that

1) for $c = 1$, $\Gamma(V_{LPS-c}) = \theta - 1$,
2) with increasing c, $\Gamma(V_{LPS-c})$ increases, i.e., the sojourn time tail gets lighter,
3) for c large enough that $k_c > \theta/(\theta-1)$ we have that $\Gamma(V_{LPS-c}) = \theta$.

Therefore, as c increases, the sojourn time tail behavior under LPS-c transitions from that under FCFS (worst case) to that under PS (optimal). This is consistent with our view of LPS-c as a hybrid version of FCFS and PS. Also, with heavy-tailed job sizes, LPS-c should be designed with 'large enough' c.

Finally, we sketch how Theorem 4 is used to prove statements (a) and (c) of Proposition 3. (3) implies that LPS-c is weakly tail-robust over regularly varying workloads if and only if $k_c \geq 2$, which can be shown to hold for $c \geq \left\lfloor \frac{1}{1-\rho} \right\rfloor + 1$. It is easy to see that our tail-robust designs satisfy this condition. Also, it can be shown that setting $c = \left\lfloor \frac{\lceil \frac{\Theta}{\Theta-1} \rceil - 1}{1-\rho} \right\rfloor + 1$ implies that for workloads with regularly varying job sizes with index $\theta \geq \Theta$, $\Gamma(V_{LPS-c}) = \Gamma(V_{PS}) = \theta$, which is sufficient to guarantee weak tail-optimality over this class.

C. Tail asymptotics under light-tailed job sizes

In this section, we consider the case of light-tailed job sizes. As in Section II-B, we will describe the (logarithmic) tail asymptotics of the sojourn time using its decay rate $\gamma(V)$. Note that a larger decay rate implies a lighter tail. The following theorem describes the sojourn time decay rate under LPS-c; see [3] for the proof.

Theorem 5. *Consider the GI/GI/1 queue. If $\gamma(B) \in (0, \infty)$,*

$$\gamma(V_{LPS-c}) = \min_{a\in[0,1]} f_c(a), \tag{4}$$

$$\begin{aligned} \textit{where} \quad f_c(a) := \; & a\gamma(V_{FCFS}) + \frac{(1-a)\gamma(B)}{c} \\ & + \sup_{s\geq 0}\left[(1-a)s\left(1 - \frac{1}{c}\right) - \Psi(s)\right]. \end{aligned} \tag{5}$$

Otherwise, if $\gamma(B) = \infty$, then $\gamma(V_{LPS-c}) = \gamma(V_{FCFS})$.

To build an understanding of (4), it is useful to begin by interpreting it in the context of effects (i), (ii), and (iii) described in Section II-B that could lead to a long sojourn time. Intuitively, the effects (i), (ii) and (iii) contribute respectively to the first, second, and third term in (5). The variable a captures the relative contribution of these effects to a large sojourn time. If a is close to 1, then effect (i) dominates, whereas if a is close to 0, then effects (ii) and (iii) dominate. The minimization operation in (4) implies that the most dominant combination of these effects determines the sojourn time decay rate. Therefore, one should interpret the value of $a_c^* := \arg\min_{a\in[0,1]} f_c(a)$ as providing a description of *how* large sojourn times are caused.

We now sketch how Theorem 5 is used to complete the proof of Proposition 3. To prove statements (b) and (d) of Proposition 3, we need to deduce the following properties of the LPS-c decay rate characterization (4).

Lemma 6. *Consider the GI/GI/1 queue. Assuming $\gamma(B) \in (0,\infty)$, $\gamma(V_{LPS-c})$ is monotonically decreasing in c. Moreover,*

$$\lim_{c\to\infty} \gamma(V_{LPS-c}) = \gamma(V_{PS}).$$

If $\gamma(V_{FCFS}) < \gamma(B)$, then

1) *for $c \leq \frac{\gamma(B)}{\gamma(V_{FCFS})}$, $\gamma(V_{LPS-c}) = \gamma(V_{FCFS})$,*
2) *for $c > \frac{\gamma(B)}{\gamma(V_{FCFS})}$, $\gamma(V_{LPS-c})$ is strictly decreasing in c.*

The statements of the above lemma are proved in [3]. Lemma 6 implies that for light-tailed workloads satisfying $\gamma(V_{FCFS}) < \gamma(B)$, $\gamma(V_{LPS-c}) > \gamma(V_{PS})$ for all c. This implies that LPS-c is weakly tail-robust over this class of workloads, which includes the set of workloads with phase-type distributed job sizes. Also, whenever the condition $c \leq \frac{\gamma(B)}{\gamma(V_{FCFS})}$ holds, $\gamma(V_{LPS-c}) = \gamma(V_{FCFS})$. This implies that LPS-$c$ is weakly tail-optimal over the class of workloads that satisfy this condition.

V. Tail-robust design: performance in expected sojourn time

In the preceding sections, we restricted our focus on the design of scheduling policies that provide good sojourn time tail performance. In practice however, we would like a scheduler to guarantee good performance not just with respect to the tail of the sojourn time distribution, but also its mean. In other words we would like to minimize the occurrence of very large sojourn times, as well as minimize the average sojourn times. In addition, we would also like the scheduler to be robust to the modeling of the workload. The design of scheduling policies that satisfy all these criteria remains a hard and open problem.

In this section, we investigate the performance of our proposed tail-robust scheduler designs with respect to mean sojourn time. Unfortunately, there is no known analysis for the mean sojourn time in an LPS-c queue even in the M/GI/1 setting. Therefore, we resort to simulations.

Consider the M/GI/1 system. Let

$$C^2(B) := \frac{\mathbb{E}\left[B^2\right]}{\mathbb{E}\left[B\right]^2} - 1$$

denote the squared coefficient of variation of the job size distribution. It is well known that for the mean sojourn time, FCFS outperforms PS when the job size d.f. is less variable than the exponential d.f. (specifically, when $C^2(B) < 1$), whereas PS outperforms FCFS when the job size d.f. is more variable than the exponential d.f. (specifically, when $C^2(B) > 1$).

Since LPS-c can be viewed as a hybrid of PS and FCFS, we might expect our tail-robust design to provide intermediate (robust) expected sojourn time performance across these two regimes. Indeed, it is easy to prove that when the failure rate (a.k.a. hazard rate) of the job size d.f. is monotone, the expected sojourn time under LPS-c is intermediate between that under FCFS and PS, as stated by the following lemma.

Lemma 7. *Consider the GI/GI/1 queue. For all $c \in \mathbb{N}$,*

1) *if the job size d.f. has a non-increasing failure rate, then*

$$\mathbb{E}\left[V_{PS}\right] \leq \mathbb{E}\left[V_{LPS-(c+1)}\right] \leq \mathbb{E}\left[V_{LPS-c}\right] \leq \mathbb{E}\left[V_{FCFS}\right],$$

2) *if the job size d.f. has a non-decreasing failure rate, then*

$$\mathbb{E}\left[V_{FCFS}\right] \leq \mathbb{E}\left[V_{LPS-c}\right] \leq \mathbb{E}\left[V_{LPS-(c+1)}\right] \leq \mathbb{E}\left[V_{PS}\right].$$

This lemma is proved by invoking a stochastic ordering result for the LPS-c queue by Nuyens and van der Weij [12]. We prove Lemma 7 in the appendix.

In our first experiment, we fix the values of ρ and $\mathbb{E}\left[B\right]$, and set $c = \left\lfloor \frac{1}{1-\rho} \right\rfloor + 1$. We study the variation of $\mathbb{E}\left[V\right]$ versus $C^2(B)$ under FCFS, LPS-c, and PS. For each value of $C^2(B)$, we model the job size d.f. by a phase-type distribution by matching the first the second moment using the method suggested by Sauer & Chandy [13]: for $C^2(B) < 1$, the d.f. is modeled by a Generalized Erlang distribution, and for $C^2(B) > 1$, the d.f. is modeled by a Hyper-exponential distribution.

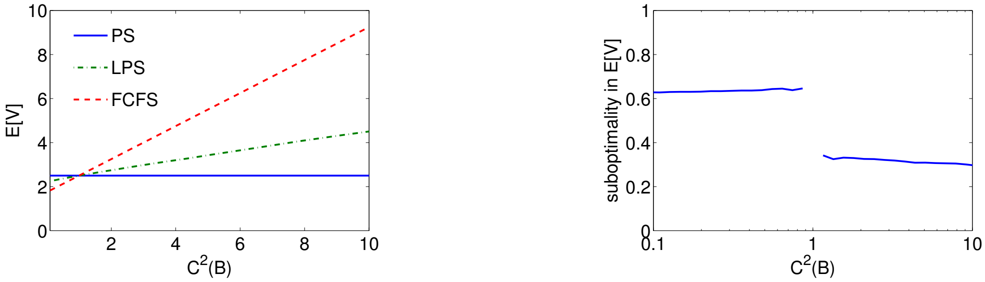
Fig. 1 shows the results for the low-load case: $\rho = 0.6$, $\mathbb{E}\left[B\right] = 1$. Fig. 1(a) is a plot of $\mathbb{E}\left[V\right]$ versus $C^2(B)$. Note that $\mathbb{E}\left[V_{LPS-c}\right]$ appears to grow linearly with $C^2(B)$, with a slope that is intermediate between that for PS (slope = 0) and FCFS (slope = $\frac{\mathbb{E}[B]}{2}\left(1 + \frac{\rho}{1-\rho}\right)$). In Fig 1(b), we plot the relative suboptimality in expected sojourn time under LPS-c relative to PS and FCFS. Specifically, we plot

$$\frac{\mathbb{E}\left[V_{LPS-c}\right] - \min(\mathbb{E}\left[V_{PS}\right], \mathbb{E}\left[V_{FCFS}\right])}{\max(\mathbb{E}\left[V_{PS}\right], \mathbb{E}\left[V_{FCFS}\right]) - \min(\mathbb{E}\left[V_{PS}\right], \mathbb{E}\left[V_{FCFS}\right])}$$

versus $C^2(B)$. The plot is almost constant over the regimes $C^2(B) < 1$ and $C^2(B) > 1$, as is also suggested by the almost linear plot of $\mathbb{E}\left[V_{LPS-c}\right]$ versus $C^2(B)$ in Fig. 1(a). Further, we see that our particular tail-robust choice of c seems to provide relatively better performance for expected sojourn time in the $C^2(B) > 1$ (high variability) regime. Fig 2 shows the corresponding plots in the high-load case: $\rho = 0.9$, $\mathbb{E}\left[B\right] = 1$. Note that the plots are qualitatively similar to the low-load case.

In our second experiment, we choose 3 different job size distributions, and study the variation of $\mathbb{E}\left[V\right]$ with the load (which we scale by scaling the arrival rate). We compare FCFS, PS, and LPS-c (with $c = \left\lfloor \frac{1}{1-\rho} \right\rfloor + 1$). Fig. 3 summarizes our results. For parts (a) and (b), we model the job size d.f. as a phase-type d.f. by matching the first two moments, as in [13]. As before, we observe that:

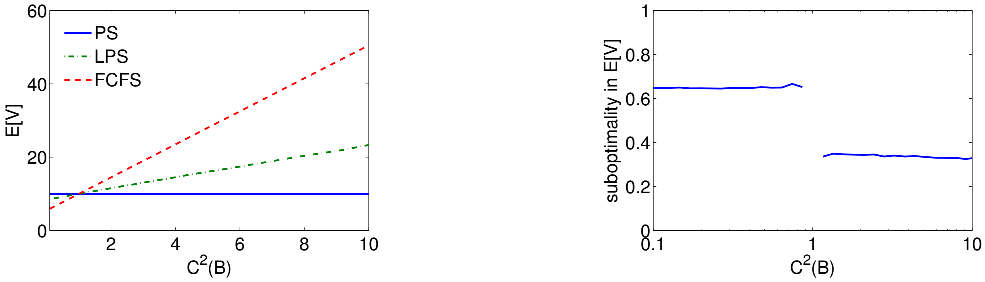
1) The expected sojourn time under LPS-c is intermediate between that under FCFS and PS.
2) With our particular tail-robust choice of c, $\mathbb{E}\left[V_{LPS-c}\right]$ is closer to $\mathbb{E}\left[V_{PS}\right]$ than $\mathbb{E}\left[V_{FCFS}\right]$. This means the 'relative suboptimality', as defined above under LPS-c is smaller for high-variability job size distributions ($C^2(B) > 1$).



(a)

(b)

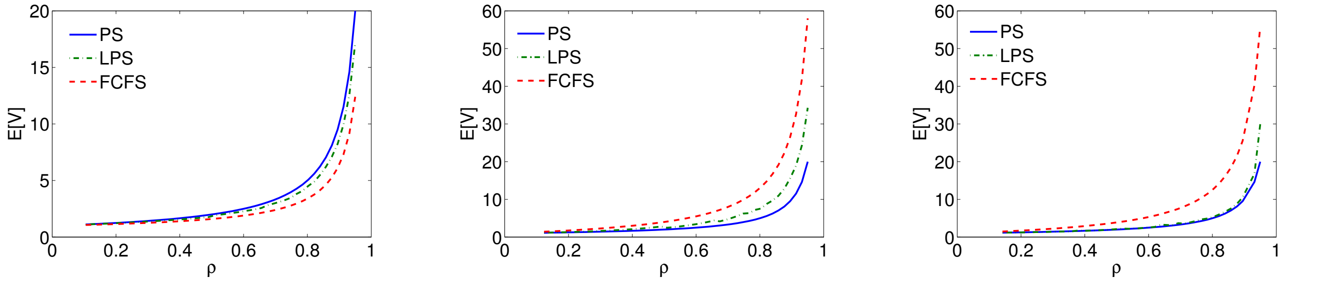
Fig. 1. $\rho = 0.6$, $\mathbb{E}[B] = 1$



(a)

(b)

Fig. 2. $\rho = 0.9$, $\mathbb{E}[B] = 1$



(a) $\mathbb{E}[B] = 1$, $C^2(B) = 1/5$

(b) $\mathbb{E}[B] = 1$, $C^2(B) = 5$

(c) B ∼Pareto with index $\alpha = 2.1$, $\mathbb{E}[B] = 1$ $\Rightarrow$ $C^2(B) = 4.76$

Fig. 3. $E[V]$ versus ρ

Appendix
Proof of Lemma 7

The proof Lemma 7 follows easily from the following two lemmas.

Lemma 8. *Consider the GI/GI/1 queue. If the job size d.f. has a non-increasing failure rate, then $\mathbb{E}[V_{LPS-c}]$ is monotonically decreasing in c. If the job size d.f. has a non-decreasing failure rate, then $\mathbb{E}[V_{LPS-c}]$ is monotonically increasing in c.*

This lemma is proved in [12].

Lemma 9. *In the GI/GI/1 queue,*

$$V_{LPS-c} \xrightarrow{c\uparrow\infty} V_{PS} \text{ in distribution.}$$

Proof: We wish to prove that for all $x \geq 0$,

$$\lim_{c\to\infty} P\left(V_{LPS-c} > x\right) = P\left(V_{PS} > x\right). \tag{6}$$

Let N denote the number of jobs entering the system during a busy period. For a scheduling policy $\pi \in \Pi$, Let $N_\pi(x)$ denote the number of jobs that experience a sojourn time $>$ x during a busy period under policy π. It is well known that

$$P\left(V_\pi > x\right) = \frac{\mathbb{E}\left[N_\pi(x)\right]}{\mathbb{E}\left[N\right]}. \tag{7}$$

Since N is almost surely finite,

$$\begin{aligned}\mathbb{E}\left[N_{PS}(x)\right] &= \lim_{c\to\infty} \mathbb{E}\left[N_{PS}(x) \mid N < c\right] P\left(N < c\right)\\ &= \lim_{c\to\infty} \mathbb{E}\left[N_{LPS-c}(x) | N < c\right] P\left(N < c\right).\end{aligned} \tag{8}$$

The last step above uses the fact that conditioned on the event $N < c$, all jobs in the busy period experience the same sojourn time under PS and LPS-c.

We now note that

$$\begin{aligned} & \mathbb{E}\left[N_{LPS-c}\right] \geq \mathbb{E}\left[N_{LPS-c} | N < c\right] P\left(N < c\right) \\ \Rightarrow & \liminf_{c \to \infty} \mathbb{E}\left[N_{LPS-c}\right] \geq \lim_{c \to \infty} \mathbb{E}\left[N_{LPS-c} | N < c\right] P\left(N < c\right) \\ & = \mathbb{E}\left[N_{PS}(x)\right]. \end{aligned} \tag{9}$$

The last step above uses (8). Also,

$$\begin{aligned} \mathbb{E}\left[N_{LPS-c}\right] &= \mathbb{E}\left[N_{LPS-c} | N < c\right] P\left(N < c\right) \\ & \quad + \mathbb{E}\left[N_{LPS-c} | N > c\right] P\left(N > c\right) \\ &\leq \mathbb{E}\left[N_{LPS-c} | N < c\right] P\left(N < c\right) \\ & \quad + \mathbb{E}\left[N | N > c\right] P\left(N > c\right) \end{aligned}$$

Since $\lim_{c \to \infty} \mathbb{E}\left[N | N > c\right] P\left(N > c\right) = 0$, using (8), we get

$$\limsup_{c \to \infty} \mathbb{E}\left[N_{LPS-c}\right] \leq \mathbb{E}\left[N_{PS}(x)\right]. \tag{10}$$

(9) and (10) imply that

$$\lim_{c \to \infty} \mathbb{E}\left[N_{LPS-c}\right] = \mathbb{E}\left[N_{PS}(x)\right],$$

which, invoking (7), implies (6). This completes the proof. ■

References

[1] L. E. Schrage, "A proof of the optimality of the shortest remaining processing time discipline." *Operations Research*, vol. 16, 1968.

[2] A. Wierman and B. Zwart, "Is tail-optimal scheduling possible?" Under submission.

[3] J. Nair, A. Wierman, and B. Zwart, "Tail-robust scheduling via limited processor sharing," *Performance Evaluation*, To appear.

[4] O. Boxma and B. Zwart, "Tails in scheduling," *Performance Evaluation Review*, vol. 34, no. 4, pp. 13–20, 2007.

[5] V. Anantharam, "Scheduling strategies and long-range dependence," *Queueing Systems Theory Appl.*, vol. 33, no. 1-3, pp. 73–89, 1999, Queues with heavy-tailed distributions.

[6] M. Mandjes and B. Zwart, "Large deviations of sojourn times in processor sharing queues," *Queueing Syst. Theory Appl.*, vol. 52, no. 4, pp. 237–250, 2006.

[7] M. Nuyens and B. Zwart, "A large-deviations analysis of the GI/GI/1 SRPT queue," *Queueing Syst. Theory Appl.*, vol. 54, no. 2, pp. 85–97, 2006.

[8] K. Ramanan and A. L. Stolyar, "Largest weighted delay first scheduling: large deviations and optimality," *Annals of Applied Probability*, vol. 11, pp. 1–48, 2001.

[9] M. Nuyens, A. Wierman, and B. Zwart, "Preventing large sojourn times using SMART scheduling," *Operations Research*, vol. 56, no. 1, pp. 88–101, 2008.

[10] S. Borst, R. Nunez-Queija, and B. Zwart, "Sojourn time asymptotics in processor-sharing queues," *Queueing Systems*, vol. 53, no. 1-2, pp. 31–51, 2006.

[11] A. Scheller-Wolf and R. Vesilo, "Structural interpretation and derivation of necessary and sufficient conditions for delay moments in FIFO multiserver queues," *Queueing Syst. Theory Appl.*, vol. 54, no. 3, pp. 221–232, 2006.

[12] M. Nuyens and W. van der Weij, "Monotonicity in the limited processor-sharing queue," *Stochastic Models*, vol. 25, no. 3, pp. 408–419, 2009.

[13] C. H. Sauer and J. M. Chandy, "Approximate analysis of central server models," *IBM J. Res. Dev.*, vol. 19, no. 3, pp. 301–313, 1975.