Scheduling despite inexact job-size information

Adam Wierman
California Institute of Technology
1200 E. California Blvd.
Pasadena, CA 91125
acw@caltech.edu

Misja Nuyens
Statkraft
Lilleakerveien 6
Lilleaker, 0216 Oslo
misjanuyens@gmail.com

ABSTRACT

Motivated by the optimality of Shortest Remaining Processing Time (SRPT) for mean response time, in recent years many computer systems have used the heuristic of "favoring small jobs" in order to dramatically reduce user response times. However, rarely do computer systems have knowledge of exact remaining sizes. In this paper, we introduce the class of ϵ-SMART policies, which formalizes the heuristic of "favoring small jobs" in a way that includes a wide range of policies that schedule using inexact job-size information. Examples of ϵ-SMART policies include (i) policies that use exact size information, e.g., SRPT and PSJF, (ii) policies that use job-size estimates, and (iii) policies that use a finite number of size-based priority levels.

For many ϵ-SMART policies, e.g., SRPT with inexact job-size information, there are no analytic results available in the literature. In this work, we prove four main results: we derive upper and lower bounds on the mean response time, the mean slowdown, the response-time tail, and the conditional response time of ϵ-SMART policies. In each case, the results explicitly characterize the tradeoff between the accuracy of the job-size information used to prioritize and the performance of the resulting policy. Thus, the results provide designers an understanding of how accurate job-size information must be in order to achieve desired performance guarantees.

1. INTRODUCTION

Job-size information is a useful tool for improving the performance of schedulers. It has long been known that Shortest Remaining Processing Time first (SRPT) scheduling minimizes mean response time (sojourn time) [35] and is near optimal for weighted response time measures such as mean slowdown (stretch) [12]. Similarly, variants of SRPT such as Preemptive Shortest Job First (PSJF) are known to provide near optimal mean response time and mean slowdown.

However, the adoption of designs based on SRPT and PSJF has been slow due to fears about the fairness of these policies. Specifically, there are worries that large job sizes may be "starved" of service under a policy that gives priority to small job sizes. Recently, a resurgence of interest in SRPT, PSJF, and other size based policies, has resulted in interesting new results that have eased fairness concerns [37, 34, 38] and characterized the response-time tail of these policies [6, 5, 26]. This recent theoretical work has prompted many new computer system designs to "favor small jobs," e.g., web servers [14, 33, 19], wireless access points [15, 21], peer-to-peer networks [30], routers [31, 32], databases [22, 23], and beyond.

Unfortunately, there is a major disconnect between the theoretical work and the practical applications: the idealized policies studied analytically differ significantly from the policies implemented in real systems. There are three major differences that need to be addressed:

1. In designing systems, the goal is not simply to provide a small mean response time: other performance measures are also important, e.g., quality of service, slowdown, and fairness. Thus, idealized policies such as SRPT and PSJF are often tweaked by practitioners to perform well on secondary performance measures. For example, in web servers and routers, hybrid policies have been developed to minimize mean response time while providing fairness guarantees [13, 32, 11].
2. Information about the service demands (sizes) of jobs is hardly ever exact. Instead, estimates of job sizes are typically used. For instance, when serving static content, web servers have exact knowledge of the sizes of the files being served, but have inexact knowledge of network conditions. Thus, the web server only has an estimate of the true service demand [20, 33]. Similarly, estimates must be used for server side scheduling in peer-to-peer networks [30].
3. The overhead involved in distinguishing between the continuum of priority classes used by SRPT and PSJF typically causes system designers to discretize the policies, so that they use only a small number (5-10) of priority classes. Again, we can find examples of this in web servers and routers [14, 33, 32].

To this point, progress has been made towards bridging the first and third issues above. With respect to the latter, there is a large body of work analyzing the performance of policies that use a finite number of priority levels, see [8] and the references therein. With respect to the first issue, an emerging style of research attempts characterize the performance of hybrid policies by studying *classifications of scheduling policies* instead of individual idealized policies. The goal is to formalize a scheduling heuristic such

SIGMETRICS 2008 Annapolis, MD USA

as "favoring small jobs" and then study the impact of this *heuristic* instead of studying one specific policy that obeys the heuristic, e.g., [40, 38, 17, 16]. With this approach, the results characterize a wide range of hybrid policies, eliminating the need to study each hybrid policy individually. One such heuristic classification is the SMART class, introduced in [40], which formalizes the heuristic of "favoring small jobs" in a way that includes policies such as SRPT and PSJF in addition to a wide array of hybrid policies.

Though progress has been made towards addressing the first and third issues above, to the best of our knowledge, there is no analytic work addressing the second issue. The only studies to this point use simulation techniques, e.g., [20, 19] and the references therein.

In the current work, we will address all three issues above. Specifically, we introduce a generalization of the SMART class called the ϵ-SMART class, which includes hybrid policies, policies that schedule using job-size estimates, and policies that use only a finite number of priority levels. The ϵ-SMART class is defined using three simple properties (Definition 1 below) that formalize the notion of "favoring small jobs" in a way that guarantees that every job has priority over other jobs that are "significantly" larger. The key generalization that ϵ-SMART makes over the SMART classification is that it allows a job to have priority over a *somewhat* smaller job, where the *somewhat* is described by a function $\epsilon(\cdot)$. This parameter allows a tradeoff between the deviation from SRPT allowed and the strength of the performance bounds provable about policies in the classification. Further, the form of $\epsilon(\cdot)$ can be chosen so that ϵ-SMART can be used to study each of the three issues above, thus providing a unified analytic framework.

We prove a number of performance guarantees for ϵ-SMART policies. Specifically, in the M/GI/1 queue, we prove *simple and tight* bounds on the (i) mean response time, (ii) mean slowdown, (iii) response-time tail, and (iv) conditional response time (response time for a job of size x). The key contribution of these bounds is that they are all in terms of $\epsilon(\cdot)$, and thus explicitly relate the strength of the bias towards small jobs (i.e., the deviation from SRPT allowed) to the performance bounds. As a result, these bounds provide interesting new insights into design questions related to the second and third issue above.

With respect to the second issue above, it is clear that scheduling using job-size estimates increases mean response time, and that as estimation accuracy increases, the resulting mean response time decreases. But, improving estimates often comes only at the expense of significantly increased overhead. For example, in the case of web servers, improving estimates of job sizes requires improving estimates of network delays for every connection. Thus, designers need to understand the relationship between the improvement in accuracy and the reduction in mean response time. Our bounds provide an explicit characterization of this relationship, and how the relationship varies across performance measures. We will discuss this in detail in Section 5.1.

With respect to the third issue above, it is clear that the larger the number of priority classes, the smaller the mean response time. However, using more priority classes results in an increase in overhead. Thus, designers need to understand the improvement that comes from adding additional priority classes. Again, our bounds provide an explicit characterization of this improvement. We will discuss this in detail in Section 5.2.

The remainder of the paper is organized as follows. We first define the ϵ-SMART classification and discuss what policies are included in and excluded from the class (Section 2). Then, in Section 3, we introduce the model and notation that we use for our analysis of ϵ-SMART. In Section 4, we derive our bounds on the ϵ-SMART policies. The section is divided into three subsections where we subsequently analyze the conditional response time (Section 4.1), the mean response time (Section 4.2), and the response-time tail (Section 4.3). Following our analysis of ϵ-SMART, we discuss applications of the class in Section 5. Section 5.1 describes scheduling using job-size estimates and Section 5.2 describes scheduling using a finite number of priority levels. Finally, Section 6 includes some concluding remarks.

2. THE ϵ-SMART CLASSIFICATION

Although the heuristic of "favoring small jobs" has recently been used in many applications, the policies that result are very different due to (i) the need to optimize secondary performance measures, (ii) the need to prioritize using inexact job-size information, and (iii) the need to use a fixed, finite number of priority classes.

The ϵ-SMART classification formalizes the notion of "favoring small jobs" in a way that allows us to provide bounds on the impact of all of these variations at once. The class includes a wide range of variations on the idealized policies studied traditionally. The definition of ϵ-SMART represents a balance between the breadth of the policies included, and the tractability of performance bounds on the class. To highlight this balance, note that many of the policies in ϵ-SMART, e.g., policies that prioritize using estimates of job sizes, have not been analyzed in the literature before.

2.1 Defining ϵ-SMART

The ϵ-SMART classification is defined by three simple properties. In the definition, we denote jobs by a, b, or c, where job a has remaining size r_a and original size s_a. We also define job a to have priority over job b if job b can never run while job a is in the system.

Definition 1 *Let ϵ be a non-decreasing right-continuous function on $[0, \infty)$ such that $\epsilon(x) \geq x$ for all x and define $\epsilon^{-1}(x) = \inf\{y : \epsilon(y) \geq x\}$. A work conserving policy P is part of ϵ-SMART if it obeys the following properties at all times.*

Bias Property: *If $r_b > \epsilon(s_a)$, then job a has priority over job b.*

Consistency Property: *If job a ever receives service while job b is in the system, then job a has priority over job b at all times thereafter.*

Transitivity Property: *If an arriving job b preempts job c, then thereafter, until job c receives service, every arrival a with $\epsilon(s_a) < s_b$ is given priority over job c.*

This definition parallels and extends the definition of the SMART (SMAll Response Time) classification, introduced by Wierman, Harchol-Balter & Osogami in Sigmetrics 2005 [40]. In particular, the SMART class is a subclass of ϵ-SMART that can be obtained by setting $\epsilon(x) = x$. Thus, it is easy to see the wide array of policies included in ϵ-SMART and excluded from SMART.

As in the SMART classification, the Bias Property is designed to guarantee that the job being run by the server

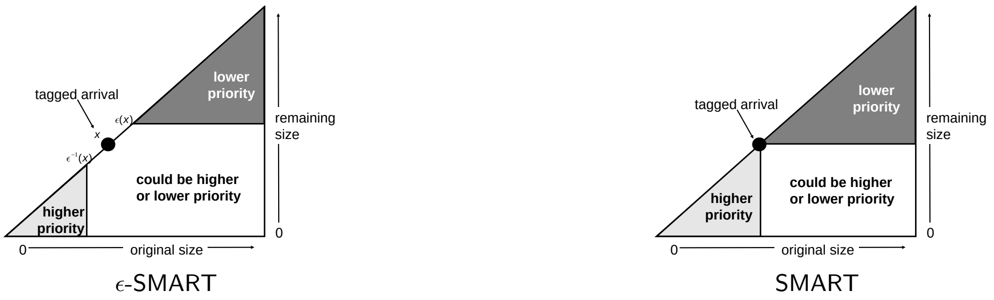


Figure 1: This diagram illustrates and contrasts the priority structures induced by the Bias Properties in the definition of ϵ-SMART (Definition 1) and SMART.

is, in some sense, one of the “smallest” jobs in the system. Refer to Figure 1 for an illustration of the ϵ-SMART Bias Property, and a contrast of the Bias Property under SMART and ϵ-SMART. The Consistency and Transitivity Properties enforce coherency in the notion of “small.” In particular, the Consistency Property says that if the scheduler decides that job a is “smaller” than job b, and thus serves job a, job a should stay “smaller” than b as it is worked on. Similarly, the Transitivity Property states that if job b is “smaller” than job c, then any arrival that is “smaller” than job b should also be “smaller” than job c.

The novelty of the ϵ-SMART classification is the inclusion of a function $\epsilon(x)$ that captures the strength of the bias towards small jobs in a formal way. If $\epsilon(x)$ is much larger than x, jobs can receive priority over much smaller jobs. When $\epsilon(x)$ is close to x, ϵ-SMART policies behave much like SRPT and PSJF. Thus, $\epsilon(x)$ can be thought of as a bound on the allowed deviation from SRPT. A last comment on Definition 1 is that it has been constructed so as to enforce only a *partial ordering* on the priorities of jobs. Thus, ϵ-SMART policies can, for instance, change how the policy makes decisions at arrival and departure instants. This is an important point to bring out because traditional analysis of scheduling policies assumes that policies obey one fixed priority rule, while ϵ-SMART policies may change their prioritization rule over time. This degree of freedom complicates much of the analysis of the ϵ-SMART class.

2.2 Examples of ϵ-SMART policies

Many common policies are part of the ϵ-SMART class. Of course, the ϵ-SMART class includes all SMART policies, which include SRPT, PSJF, and hybrids such as the RS policy, which assigns to each job the product of its remaining size and its original size and then gives highest priority to the job with lowest product. The inclusion of these hybrid policies is important because systems need to perform well for a combination of metrics, and thus variants of SRPT and PSJF are typically used in practice.

Apart from hybrid policies, ϵ-SMART includes many practical variations of policies that are excluded from SMART because of the rigidity of the Bias Property of SMART. For example, ϵ-SMART includes policies that have only a finite number of priority levels. In particular, ϵ-SMART includes preemptive threshold based policies where there are a finite number of thresholds $0 = t_1, \dots, t_k = \infty$ and a job of size s and remaining size r is assigned priority $p(s, r) = i$ if $p_1(s, r) \in [t_i, t_{i+1})$ for some static priority function $p_1(s, r) \in$ SMART such as $p_1(s, r) = s$ (i.e., PSJF). The inclusion of these policies is of particular practical importance because in many cases system designers simplify implementations by using only 5-10 priority levels. In Section 5.2, we will discuss how ϵ-SMART applies to this setting.

In addition to including threshold based policies, ϵ-SMART includes SMART policies that use inexact job-size information. The inclusion of these policies is of practical interest because exact job-size information is hardly ever available. To capture job-size estimates using ϵ-SMART one can use either $\epsilon(x) = x + \delta$, where δ bounds the error in the job size in an additive way, or $\epsilon(x) = (1 + \sigma)x$, where $1 + \sigma$ bounds the variation in the speed at which jobs are served. Note that, in both cases, no distributional assumptions are made on the job-size estimate errors – even adversarial errors are allowed. However, a strict upper bound on the errors of estimates is assumed. This is not realistic in many settings, but moving to probabilistic $\epsilon(x)$ is difficult, and is a topic we are still exploring. We will discuss the application of ϵ-SMART to understand policies using job-size estimates in Section 5.1.

The ϵ-SMART class also includes *time-varying policies*, i.e., policies that can change their priority rules over time based on system-state information or randomization. These policies are of practical importance because they allow system designers to perform *online multi-objective optimization.* Specifically, suppose a system designer wants to optimize a secondary objective while still providing small mean response times. In order to accomplish this, the system designer can implement a parameterized version of ϵ-SMART, such as prioritizing based on some family of functions $p_i(s, r)$ of size and remaining size, and then use machine learning techniques to search for the $p_i(s, r)$ that optimizes the secondary objective.

2.3 Policies excluded from ϵ-SMART

To this point we have only discussed the breadth of ϵ-SMART. However, it is also important to note that many policies are excluded from ϵ-SMART. Clearly, ϵ-SMART does not include policies that give priority to large jobs such as Longest Job First (LJF), Preemptive Longest Job First (PLJF), etc. In addition, ϵ-SMART does not include policies that only “weakly” prioritize small jobs. For example, ϵ-SMART does not include any non-preemptive policies, not even ones like Shortest Job First (SJF) that prioritize small jobs; nor does it include policies that do not use knowledge about the job sizes (blind policies), like Foreground-Background (FB).

The exclusion of these policies is a result of the tension between the *breadth* of the class and the *tightness* of the results provable about the class. In particular, excluding policies such as SJF and FB that bias weakly towards small job sizes is necessary in order to show that ϵ-SMART policies provide a near optimal mean response time across all service distributions and all loads. For example, though SJF can provide good mean response time when the second moment of the service distribution, $E[X^2]$ is small, the mean response time of SJF is arbitrarily larger than the optimal as $E[X^2] \to \infty$. Similarly, though FB can provide near optimal mean response time under service distributions having decreasing failure rates, when the service distribution has an increasing failure rate, FB can behave very poorly.

3. PRELIMINARIES

Though the definition of ϵ-SMART applies generally, to study the performance of ϵ-SMART policies, we need to choose a model simple enough to allow the analysis to be tractable. The setting we will use is a stationary preempt-resume M/GI/1 queue with generic service times (job sizes) X, having $E[X] < \infty$, and arrival rate λ. The system load ρ satisfies $\rho = \lambda E[X] < 1$. Let the service distribution, $F(x)$, be continuous, $\overline{F}(x) = 1 - F(x)$ be its tail, and $f(x)$ be its density. Define the right endpoint $x_U = \sup\{x : F(x) < 1\}$ and the left endpoint $x_L = \inf\{x : F(x) > 0\}$.

Let T^{P} denote a random variable that is distributed according to the response time under policy P. The response time (sojourn time) is the time between the arrival and the departure of a job. Let $T(x)^{\mathsf{P}}$ be distributed according to the conditional response time of a job of size x under policy P. Similarly, let S^P denote a random variable distributed according to the slowdown under policy P and $S(x)^P$ be the conditional slowdown under policy P. The slowdown (stretch) of a job of size x is defined as $T(x)/x$. The random variable W^{P}, called the *waiting time*, is distributed as the time a job waits before its service starts; $W(x)^{\mathsf{P}}$ denotes the waiting time for a job of size x. Finally, let R^{P}, the *residence time* of a job, be distributed as the time a job spends in the system after its service has started, and let $R(x)^{\mathsf{P}}$ denote the residence time of a job of size x. Hence, we may write $T(x)^{\mathsf{P}} \overset{\mathrm{d}}{=} R(x)^{\mathsf{P}} + W(x)^{\mathsf{P}}$.

Our analysis will depend heavily on the use of the following types of busy periods. Denote by B a random variable with the steady-state busy-period distribution. Let $B(y)$ be a random variable with the same distribution as a steady-state busy period that is started by a job of size y. Let B_x be distributed as the length of a steady-state busy period in the queue with generic service time $XI(X < x)$, where $I(A)$ denotes the indicator function of the event A. So, in this queue, the service time of a customer is zero with probability $P(X \geq x)$. We assume that those customers leave the queue immediately. Hence, the busy period B_x is made up of arrivals with sizes less than x. Furthermore, let $B_x(y)$ be distributed as a busy period in the queue with service time $XI(X < x)$ that is started by a job of size y.

In order to describe the moments of the busy period variations above, we will use the following notation. Let $m_i(x)$ be the ith moment of $XI(X < x)$, i.e., $m_i(x) = \int_0^x t^i f(t)dt$. Further, let $\rho(x) = \lambda m_1(x)$. Additionally, let $\widetilde{m_i}(x)$ be the ith moment of $min(X, x)$, i.e., $\widetilde{m_i}(x) = i \int_0^x t^{i-1}\overline{F}(t)dt$.

We will make use of the following asymptotic notation. Define $f(x) \sim g(x)$ as $x \to a$ to mean that $\lim_{x\to a} f(x)/g(x) = 1$. Further, we use the notation $f(x) = O(g(x))$ as $x \to a$ to indicate that $\limsup_{x\to a} f(x)/g(x) < \infty$, $f(x) = \Omega(g(x))$ as $x \to a$ to indicate that $\liminf_{x\to a} f(x)/g(x) > 0$, and $f(x) = \Theta(g(x))$ as $x \to a$ to indicate that $f(x) = O(g(x))$ and $f(x) = \Omega(g(x))$. Finally, we use $f(x) = o(g(x))$ as $x \to a$ to indicate that $\lim_{x\to a} f(x)/g(x) = 0$ and $f(x) = \omega(g(x))$ as $x \to a$ to indicate that $\lim_{x\to a} f(x)/g(x) = \infty$

Two policies that are of particular importance to the paper, both to the analysis and as the most common ϵ-SMART policies, are SRPT and PSJF. So, we will provide some background on these policies here.

SRPT: We will be interested in the conditional response time under SRPT, $T(x)^{\mathsf{SRPT}}$, and we will make use of the following characterization. Let D_x^{P} be the stationary amount of work that has priority over an arriving job of size x under P. Then, $W(x)^{\mathsf{SRPT}}$, is distributed as a busy period with an initial customer of size D_x^{SRPT} and generic service time $XI(X < x)$, i.e., $W(x)^{\mathsf{SRPT}} \overset{\mathrm{d}}{=} B_x(D_x^{\mathsf{SRPT}})$. Further, $R(x)^{\mathsf{SRPT}}$ is the sum of a sequence of busy periods, each started by dt work and generic service time $XI(X < t)$ where t is the remaining size of the tagged job, i.e., $R(x)^{\mathsf{SRPT}} \overset{\mathrm{d}}{=} \int_0^x B_t(dt)$. Specializing the above to $E[T(x)]^{\mathsf{SRPT}}$ and $Var[T(x)]^{\mathsf{SRPT}}$, we have:

$$
\begin{aligned}
E[T(x)]^{\mathsf{SRPT}} &= \int_0^x \frac{dt}{1-\rho(t)} + \frac{\lambda\widetilde{m_2}(x)}{2(1-\rho(x))^2} \\
Var[T(x)]^{\mathsf{SRPT}} &= \int_0^x \frac{\lambda m_2(t)}{(1-\rho(t))^3}dt + \frac{\lambda\widetilde{m_3}(x)}{3(1-\rho(x))^3} \\
&\quad + \frac{\lambda m_2(x)\lambda\widetilde{m_2}(x)}{(1-\rho(x))^4} - \frac{1}{4}\left(\frac{\lambda\widetilde{m_2}(x)}{(1-\rho(x))^2}\right)^2 .
\end{aligned}
$$

PSJF: We will again be interested in the conditional response time, $T(x)^{\mathsf{PSJF}}$, and we will make use of a similar characterization as in the case of SRPT. In particular, like under SRPT, $W(x)^{\mathsf{PSJF}}$ is distributed as a busy period with initial an customer of size D_x^{PSJF} and generic service time $XI(X < x)$, i.e., $W(x)^{\mathsf{PSJF}} \overset{\mathrm{d}}{=} B_x(D_x^{\mathsf{PSJF}})$. Further, $R(x)^{\mathsf{PSJF}}$ is a busy period started by x work including only arrivals of size $< x$, i.e., $R(x)^{\mathsf{PSJF}} \overset{\mathrm{d}}{=} B_x(x)$. Specializing the above to $E[T(x)]^{\mathsf{PSJF}}$ and $Var[T(x)]^{\mathsf{PSJF}}$, we have:

$$
\begin{aligned}
E[T(x)]^{\mathsf{PSJF}} &= \frac{x}{1-\rho(x)} + \frac{\lambda m_2(x)}{2(1-\rho(x))^2}, \\
Var[T(x)]^{\mathsf{PSJF}} &= \frac{\lambda x m_2(x)}{(1-\rho(x))^3} + \frac{\lambda m_3(x)}{3(1-\rho(x))^3} + \frac{3}{4}\left(\frac{\lambda m_2(x)}{(1-\rho(x))^2}\right)^2 .
\end{aligned}
$$

4. BOUNDING RESPONSE TIMES FOR ϵ-SMART POLICIES

We are now ready to start an analysis of the performance of ϵ-SMART policies in the M/GI/1 queue. We will provide bounds on the performance of ϵ-SMART policies with respect to three common performance metrics: the conditional response time $T(x)$ (Section 4.1), the mean response time $E[T]$ (Section 4.2), and the tail of response time $P(T > x)$ for $x \to \infty$ (Section 4.3). We will see that, despite its breadth, the ϵ-SMART classification is surprisingly tractable.

We will provide some discussion of the results throughout this section. Additionally, we will illustrate two important applications of the results in Section 5. It should be noted that the results in this section for $E[T]$, the response-time tail, and $T(x)$ generalize the results for the SMART classification from [40, 26, 41], while the bounds on $E[S]$ in this paper can be used to provide the first bounds on $E[S]$ for SMART policies.

4.1 Conditional response time

We will start our analysis of ϵ-SMART policies by bounding $T(x)$. The conditional response time for a job of size x, $T(x)$, has been used recently to understand the "fairness" and "predictability" of scheduling policies, see [37, 38, 39]. In addition, we will see that the analysis of $T(x)$ serves as a building block for the study of $E[T]$, $E[S]$, and $P(T > x)$.

To state our result, we need to define one more quantity. Let $R(x, h(x))^{\mathsf{P}}$ be the residence time for a job of size x under policy P that is viewed by the scheduler as having original size $h(x)$.

Theorem 4.1 *In an M/GI/1 queue, for all* P *in* ϵ-SMART*,*

$$R(x, \epsilon^{-1}(x))^{\mathsf{SRPT}} + W(\epsilon^{-1}(x))^{\mathsf{PSJF}} \leq_{st} T(x)^{\mathsf{P}} \leq_{st} R(x, \epsilon(x))^{\mathsf{PSJF}} + W(\epsilon(x))^{\mathsf{SRPT}}.$$

Thus,

$$\int_0^x \frac{dt}{1-\rho(\epsilon^{-1}(t))} + \frac{\lambda m_2(\epsilon^{-1}(x))}{2(1-\rho(\epsilon^{-1}(x)))^2} \leq E[T(x)]^{\mathsf{P}} \leq \frac{x}{1-\rho(\epsilon(x))} + \frac{\lambda \widetilde{m_2}(\epsilon(x))}{2(1-\rho(\epsilon(x)))^2}.$$

The proof of Theorem 4.1 follows from Lemmas 4.2 and 4.3 below, which prove stochastic bounds on the residence and waiting times of ϵ-SMART policies respectively. However, before proving these bounds, let us take a moment to discuss Theorem 4.1.

Observe that the bounds in Theorem 4.1 are a combination of the residence times and waiting times of variants of SRPT and PSJF that are in ϵ-SMART. Thus, the bounds are tight for both the waiting time and residence time. This is not surprising because the residence time and waiting time of SRPT and PSJF also played roles in the corresponding bounds on the SMART classification. The difference here is that the bounds do not depend on the residence/waiting time for a job of size x, but on the residence/waiting times for jobs of size $\epsilon(x)$ and $\epsilon^{-1}(x)$. Thus, we can explicitly see the impact of weakening the bias towards small jobs (increasing the gap between $\epsilon(x)$ and $\epsilon^{-1}(x)$) on the tightness of the bounds on $T(x)$.

Now, we will state and prove two lemmas from which Theorem 4.1 follows. First, we prove a bound on the residence time of ϵ-SMART policies.

Lemma 4.2 *In an M/GI/1 queue, for all* P *in* ϵ-SMART*,*

$$R(x, \epsilon^{-1}(x))^{\mathsf{SRPT}} \leq_{st} R(x)^{\mathsf{P}} \leq_{st} R(x, \epsilon(x))^{\mathsf{PSJF}}.$$

Proof. Consider a tagged job j_x of size x that has just begun to receive service at time t. As a consequence of the Consistency Property, all jobs in the system at time t have lower priority than j_x. Thus, only arriving jobs contribute to $R(x)^{\mathsf{P}}$.

To maximize $R(x)^{\mathsf{P}}$ among P in ϵ-SMART, P must give all arriving jobs smaller than $\epsilon(x)$ higher priority than j_x. Under this scenario, $R(x)^{\mathsf{P}} = R(x, \epsilon(x))^{\mathsf{PSJF}}$. To minimize $R(x)^{\mathsf{P}}$ among P in ϵ-SMART, P must, at all instants, give priority only to arriving jobs that are smaller than $\epsilon^{-1}(r)$, where r is the remaining size of j_x. In this scenario, $R(x)^{\mathsf{P}} = R(x, \epsilon^{-1}(x))^{\mathsf{SRPT}}$. □

Next, we prove a bound on the waiting time of ϵ-SMART policies. This proof is significantly more involved than the previous argument.

Lemma 4.3 *In an M/GI/1 queue, for all* P *in* ϵ-SMART*,* $W(\epsilon^{-1}(x))^{\mathsf{PSJF}} \leq_{st} W(x)^{\mathsf{P}} \leq_{st} W(\epsilon(x))^{\mathsf{SRPT}}$.

Proof. Consider a tagged job j_x of size x. Denote by D_x^{P} the work that is present in the queue when j_x arrives, and that has higher priority. Future arrivals may also have higher priority than j_x, thus creating a busy period of jobs that complete before j_x. Applying the Bias Property, we have that

$$B_{\epsilon^{-1}(x)}(D_x^{\mathsf{P}}) \leq_{st} W(x)^{\mathsf{P}} \leq_{st} B_{\epsilon(x)}(D_x^{\mathsf{P}}).$$

Now, in the remainder of the proof, we will argue that

$$D_{\epsilon^{-1}(x)}^{\mathsf{PSJF}} \leq_{st} D_x^{\mathsf{P}} \leq_{st} D_{\epsilon(x)}^{\mathsf{SRPT}}.$$

This is enough to complete the proof because $W(\epsilon^{-1}(x))^{\mathsf{PSJF}} = B_{\epsilon^{-1}(x)}(D_{\epsilon^{-1}(x)}^{\mathsf{PSJF}})$ and $W(\epsilon(x))^{\mathsf{SRPT}} = B_{\epsilon(x)}(D_{\epsilon(x)}^{\mathsf{SRPT}})$.

We will start by proving the lower bound, $D_{\epsilon^{-1}(x)}^{\mathsf{PSJF}} \leq_{st} D_x^{\mathsf{P}}$. Note that PSJF devotes the full service rate to jobs with original size less than $\epsilon^{-1}(x)$ when they exist. Letting Q_x be the work in the system made up of jobs with original size at most x, we have

$$D_{\epsilon^{-1}(x)}^{\mathsf{PSJF}} = Q_{\epsilon^{-1}(x)}^{\mathsf{PSJF}} \leq_{st} Q_{\epsilon^{-1}(x)}^{\mathsf{P}} \leq_{st} D_x^{\mathsf{P}},$$

where the final inequality is a consequence of the Bias Property. This completes the proof of the lower bound.

To prove the upper bound, $D_x^{\mathsf{P}} \leq_{st} D_{\epsilon(x)}^{\mathsf{SRPT}}$, we will rely heavily on the arguments used to prove Theorem 4.1 in [40], which states that $D_y^{\mathsf{Q}} \leq_{st} D_y^{\mathsf{SRPT}}$ for all $\mathsf{Q} \in \mathsf{SMART}$.

To begin, let $D_x^{\mathsf{P}}(t)$ represent the stochastic process of work under P that would have priority over j_x if j_x would arrive at time t. So, D_x^{P} is the stationary value of $D_x^{\mathsf{P}}(t)$.

Next, note that $D_x^{\mathsf{P}} \leq_{st} D_x^{\mathsf{P}^*}$, where P^* is an ϵ-SMART policy that assigns all new arrivals with size at most $\epsilon(x)$ higher priority than jobs of size exactly x. The priority structure of P, P^*, and SRPT are illustrated in Figure 2.

Now, we will prove that $D_x^{\mathsf{P}^*} \leq_{st} D_{\epsilon(x)}^{\mathsf{SRPT}}$. Intuitively, if we transform x to $\epsilon(x)$, then the Bias, Consistency, and Transitivity Properties of ϵ-SMART become those of SMART, and as mentioned above, SRPT maximizes $D_{\epsilon(x)}^{\mathsf{Q}}$ for $\mathsf{Q} \in \mathsf{SMART}$. The process $D_{\epsilon(x)}^{\mathsf{Q}}(t)$ has been characterized in [40] for $\mathsf{Q} \in \mathsf{SMART}$, and here we will use a similar characterization of $D_{\epsilon(x)}^{\mathsf{P}^*}(t)$.

Define *small* arrivals as those with original size smaller than $\epsilon(x)$ and call all other arrivals *large*. Note that small arrivals contribute to both $D_x^{\mathsf{P}^*}(t)$ and $D_{\epsilon(x)}^{\mathsf{SRPT}}(t)$ whenever they are in the system. In contrast, large arrivals can only contribute to $D_x^{\mathsf{P}^*}(t)$ and $D_{\epsilon(x)}^{\mathsf{SRPT}}(t)$ when they have remaining size smaller than $\epsilon(x)$. Further, we prove in Lemma 4.4 that large jobs can only start to contribute when $D_x^{\mathsf{P}^*}(t) = 0$ and there can be at most one large job contributing to $D_x^{\mathsf{P}*}(t)$. The same property is a defining feature of $D_{\epsilon(x)}^{\mathsf{Q}}(t)$ for $\mathsf{Q} \in \mathsf{SMART}$, see Lemma 4.1 in [40], and thus holds for SRPT in particular. Finally, observe that $D_x^{\mathsf{P}^*}(t)$ and $D_{\epsilon(x)}^{\mathsf{Q}}(t)$ are both reduced at the full service rate whenever they are positive.

Now, we can compare $D_x^{\mathsf{P}^*}(t)$ and $D_{\epsilon(x)}^{\mathsf{SRPT}}(t)$. First, note that the only difference in the evolution of these processes

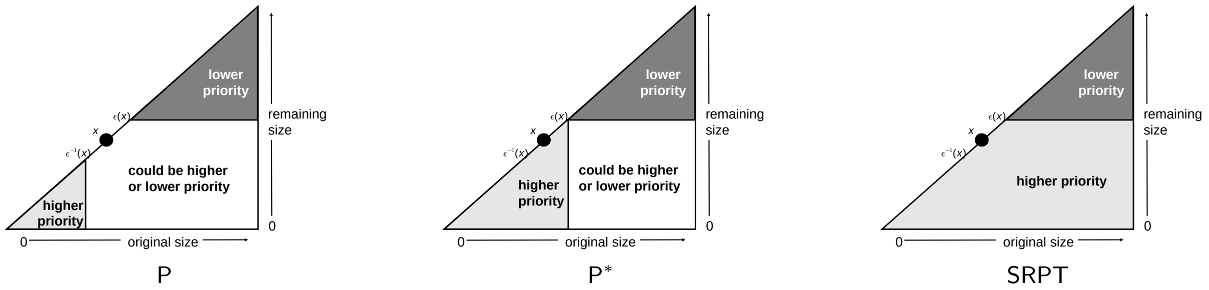


Figure 2: An illustration of the priority structure used in the three steps in the proof of Lemma 4.3.

lies in the contribution of large jobs, which can only begin when $D_x^{\mathsf{P}^*}(t) = 0$ and $D_{\epsilon(x)}^{\mathsf{SRPT}}(t) = 0$. Further, note that every large job will eventually contribute $\epsilon(x)$ work to $D_{\epsilon(x)}^{\mathsf{SRPT}}(t)$, but that this need not be the case under P^*. Thus, large jobs contribute stochastically more to the process $D_{\epsilon(x)}^{\mathsf{SRPT}}(t)$ than to $D_x^{\mathsf{P}^*}(t)$. Noting that contributions from small arrivals are stochastically identical is then enough to complete the proof that $D_x^{\mathsf{P}^*} \leq_{st} D_{\epsilon(x)}^{\mathsf{Q}}$. These remaining arguments are similar to those for the SMART class, so we refer the reader to [40] for the details. □

We now prove the lemma used in the above proof.

Lemma 4.4 *Under any ϵ-SMARTpolicy, at all times t, there is only one job with original size larger than $\epsilon(x)$ that can have priority over an arriving job of size x. Further, as long as such a job is in the system, no other jobs with original size larger than $\epsilon(x)$ can receive service.*

PROOF. Suppose job a has size $s_a > \epsilon(x)$ and reaches the point where it would have priority over an arriving tagged job j_x of size x at time t_a, and that job a is the only job with original size smaller than $\epsilon(x)$ at time t_a. We will show that no other job with original size larger than $\epsilon(x)$ can receive service while a is in the system. Without receiving service, no other job with size larger than $\epsilon(x)$ can become higher priority than an arriving j_x. So, this is sufficient to complete the proof.

First, note that because a was being served at time t_a, it must have priority over every job in the system at time t_a due to the Consistency Property. Next, note that all arrivals of original size larger than $\epsilon(x)$ after time t_a have lower priority than j_x due to the Bias Property. Consequently, they have lower priority than a due to the Transitivity property. Thus, they cannot receive service until a completes. □

4.2 Mean response time and mean slowdown

Though the stochastic bounds we derived in the previous section are of interest in their own right, probably the most important application of the bounds is to understand the mean response time, $E[T]$, and mean slowdown, $E[S]$, under ϵ-SMART policies. In this section, we will decondition Theorem 4.1 in order to obtain simple bounds on both $E[T]$ and $E[S]$. These bounds illustrate the tradeoff between the strictness of the bias towards small jobs and the efficiency of the resulting policy, as measured by the "competitive ratio" of policies in the class with respect to $E[T]$ and $E[S]$. A policy is called *c-competitive* with respect to $E[T]$ ($E[S]$) when $E[T] \leq cE[T]^{Optimal}$ ($E[S] \leq cE[S]^{Optimal}$). In the case of $E[T]$, $E[T]^{Optimal} = E[T]^{\mathsf{SRPT}}$. With respect to $E[S]$, no online policy can optimize $E[S]$ across all service distributions and loads, but it has been proven that $E[S]^{\mathsf{SRPT}} \leq 2E[S]^{Optimal}$ [12].

Theorem 4.5 *Consider an M/GI/1 queue with* P *in* ϵ-SMART. *Suppose there exists:*

(I) a $\sigma \geq 0$ satisfying $\epsilon(x) \leq (1+\sigma)x$ for all x,

(II) a $0 \leq \gamma < 1$ satisfying $(1-\gamma)(1-\rho(x)) \leq 1-\rho(\epsilon(x))$.

Then P *is $2\left(\frac{1+\sigma}{1-\gamma}\right)^2$-competitive with respect to $E[T]$, i.e.,*

$$E[T]^{\mathsf{SRPT}} \leq E[T]^P \leq 2\left(\frac{1+\sigma}{1-\gamma}\right)^2 E[T]^{\mathsf{SRPT}}.$$

Further, P *is $6\left(\frac{1+\sigma}{1-\gamma}\right)^2$-competitive with respect to $E[S]$.*

The first thing to notice about Theorem 4.5 is that it provides an explicit view of the tradeoff between minimizing mean response time and weakening the bias towards small jobs. In particular, as the bias towards small jobs strengthens (γ and σ decrease), the competitive ratio decreases. Further, if we take $\gamma = \sigma = 0$, we get back Theorem 4.1 from [40], which says that all SMART policies are 2-competitive with respect to $E[T]$. Note that, the factor of 2 is tight in this setting, it can be achieved under deterministic job size distributions. Similarly, when $\gamma = \sigma = 0$, we see that SMART policies are 6-competitive with respect to $E[S]$. This is interesting because no bound on $E[S]$ for SMART policies previously existed in the literature.

Further, γ and σ provide intuitive parameters for understanding the strength of the bias towards small jobs. Condition I indicates how large a job can be and still possibly get priority over a job of size x. Condition II bounds the percentage of the load made up by jobs larger than size x that can have priority over a job of size x. So, Conditions I and II present complementary formulations of how much the Bias Property can be weakened: significantly larger jobs are allowed to have priority without decreasing the mean response time too much, as long as the larger jobs do not make up too much load. The tradeoff between these two conditions will vary depending on the service distribution. In Section 5.1 we will see that, in the practical case of the Pareto distributions, the two conditions are actually equivalent, i.e., only the constraint $\epsilon(x) \leq (1+\sigma)x$ is necessary.

Finally, the constant competitive ratio given by Theorem 4.5 guarantees that ϵ-SMARTpolicies significantly outperform more traditional policies like Processor Sharing (PS) and First Come First Served (FCFS) at high loads. Specifically, as we let $\rho \to 1$, both PS and FCFS have $E[T] =$

$\Theta(1/(1-\rho))$ and $E[S] = \Omega(1/(1-\rho))$. In contrast, the following theorem shows that ϵ-SMART policies have strictly smaller growth rates with ρ when the service distribution is unbounded. This means that as $\rho \to 1$, the improvement of ϵ-SMART policies over FCFS and PS becomes arbitrarily large for both $E[T]$ and $E[S]$.

Theorem 4.6 *Consider an M/GI/1 queue with* P *in* ϵ-SMART *with* $\epsilon(x)$ *such that (I) and (II) hold. If* $x_U = \infty$, *then as* $\rho \to 1$, $E[T]^{\mathsf{P}} = o\left(\frac{1}{1-\rho}\right)$ *and* $E[S]^{\mathsf{P}} = o\left(\frac{1}{1-\rho}\right)$

The growth rate of $E[T]^{\mathsf{P}}$ for P in ϵ-SMART can be found explicitly in a few special cases. In particular, we have $E[T]^{\mathsf{P}} = \Theta(E[T]^{\mathsf{SRPT}})$, and the growth rate of $E[T]^{\mathsf{SRPT}}$ as $\rho \to 1$ has been calculated for exponential [2] and Pareto job sizes [3]:

(i) if X is exponentially distributed, then

$$E[T]^{\mathsf{P}} = \Theta\left(\frac{1}{(1-\rho)\log\left(1/(1-\rho)\right)}\right)$$

(ii) if $X \sim Pareto(\alpha, x_L)$ with $\alpha > 1$, i.e., $\overline{F}(x) = (x/x_L)^{\alpha}$ for some $x_L > 0$, then

$$E[T]^{PSJF} = \begin{cases} \Theta\left(\log\left(\frac{1}{1-\rho}\right)\right), & \text{if } \alpha < 2 \\ \Theta\left(\log^2\left(\frac{1}{1-\rho}\right)\right), & \text{if } \alpha = 2 \\ \Theta\left((1-\rho)^{-\frac{\alpha-2}{\alpha-1}}\right), & \text{if } \alpha > 2. \end{cases}$$

We will now prove Theorem 4.5. Due to space constraints, the proof of Theorem 4.6 is given in the Appendix.

Proof. (of Theorem 4.5) We first derive an auxiliary result:

$$\begin{aligned} \frac{\widetilde{m_2}(\epsilon(x))}{\widetilde{m_2}(x)} &= 1 + \frac{\int_x^{\epsilon(x)} t\overline{F}(t)dt}{\int_0^x t\overline{F}(t)dt} \leq 1 + \frac{\overline{F}(x)\int_x^{\epsilon(x)} tdt}{\overline{F}(x)\int_0^x tdt} \\ &\leq 1 + \frac{\epsilon(x)^2 - x^2}{x^2} \leq (1+\sigma)^2. \qquad (1) \end{aligned}$$

Now, by Theorem 4.1, we have that

$$\begin{aligned} E[T(x)]^P \leq &\frac{1-\rho(x)}{1-\rho(\epsilon(x))}E[R(x)]^{\mathsf{PSJF}} \\ &+ \left(\frac{1-\rho(x)}{1-\rho(\epsilon(x))}\right)^2\left(\frac{\widetilde{m_2}(\epsilon(x))}{\widetilde{m_2}(x)}\right)E[W(x)]^{\mathsf{SRPT}}. \end{aligned}$$

Applying Condition II and (1) then gives

$$\begin{aligned} E[T(x)]^P &\leq \frac{E[R(x)]^{\mathsf{PSJF}}}{1-\gamma} + \left(\frac{1+\sigma}{1-\gamma}\right)^2 E[W(x)]^{\mathsf{SRPT}} \\ &\leq \left(\frac{1+\sigma}{1-\gamma}\right)^2\left(E[R(x)]^{\mathsf{PSJF}} + E[W(x)]^{\mathsf{SRPT}}\right). \quad (2) \end{aligned}$$

We now apply Theorem 5.1 from [40], which guarantees that $\int_0^\infty (E[R(x)]^{\mathsf{PSJF}} + E[W(x)]^{\mathsf{SRPT}})f(x)dx \leq 2E[T]^{\mathsf{SRPT}}$ to complete the proof for $E[T]^{\mathsf{P}}$.

To complete the proof for $E[S]^{\mathsf{P}}$, we need a few more calculations. First, we have

$$\begin{aligned} E[R(x)]^{\mathsf{PSJF}} - E[R(x)]^{\mathsf{SRPT}} &= \frac{x}{1-\rho(x)} - \int_0^x \frac{dt}{1-\rho(t)} \\ &= \int_0^x \frac{\rho(x)-\rho(t)}{(1-\rho(x))(1-\rho(t))}dt \\ &\leq \frac{x\rho(x) - \int_0^x \rho(t)dt}{(1-\rho(x))^2} \\ &= \frac{\lambda m_2(x)}{(1-\rho(x))^2} \leq 2E[W(x)]^{\mathsf{SRPT}}. \end{aligned}$$

Combining the above with (2) yields

$$\begin{aligned} E[S]^{\mathsf{P}} &\leq \left(\frac{1+\sigma}{1-\gamma}\right)^2 \int_0^\infty \frac{1}{x}\left(E[R(x)]^{\mathsf{SRPT}} + 3E[W(x)]^{\mathsf{SRPT}}\right)dF(x) \\ &\leq 3\left(\frac{1+\sigma}{1-\gamma}\right)^2 E[S]^{\mathsf{SRPT}} \leq 6\left(\frac{1+\sigma}{1-\gamma}\right)^2 E[S]^{Optimal}, \end{aligned}$$

which finishes the proof. □

4.3 The response-time tail

We now move to characterizing the response time tail, $P(T > x)$ as $x \to \infty$, under ϵ-SMART policies. Understanding the distribution of response time is fundamental when considering QoS and capacity planning applications. By studying the tail of the response-time distribution, we characterize the likelihood of large delays under ϵ-SMART policies. Before we can state our results, we need to introduce the classes of job-size distributions we will be studying.

4.3.1 Distributional assumptions

We consider two classes of service distributions in this section, one class of heavy-tailed distributions and one class of light-tailed distributions. The class of heavy-tailed distributions we consider is the following:

Definition 2 *We say that the tail* $\overline{F}(x)$ *of a distribution is* ***regularly varying with index*** α, $\overline{F} \in \mathcal{RV}(\alpha)$, *when*

$$\overline{F}(x) = L(x)x^{-\alpha},$$

where $L(x)$ *is a* ***slowly varying function****, i.e.,* $L(x)$ *is such that for all* $y > 0$, $\lim_{x\to\infty} \frac{L(yx)}{L(x)} = 1$.

This class is a generalization of Pareto distributions, see [4] for more details. The class of light-tailed distributions we consider is the following:

Definition 3 *We say that* $\overline{F} \in \mathcal{LT}$ *when* $\overline{F}$ *obeys the following properties:*

(A) $Ee^{sB} < \infty$ for some $s > 0$.

(B) $P(B = x_U) = 0$.

The light-tailed distributions that satisfy both of these assumptions include distributions with infinite endpoints (e.g., exponential, gamma, and certain Weibull distributions), as well as all continuous distributions with finite support (e.g., uniform and beta distributions).

When studying the case of light-tailed job sizes, we will describe the logarithmic behavior of the tail of the sojourn time distribution using the *decay rate*.

Definition 4 *The* ***(asymptotic) decay rate*** $\gamma(X)$ *of a random variable* X *is defined by*

$$\gamma(X) = \lim_{x\to\infty} \frac{-\log P(X>x)}{x},$$

given that the limit exists.

Informally, for large x, one may write $P(X>x) \approx e^{-\gamma(X)x}$. It should be noted that a smaller decay rate corresponds to a heavier tail of the distribution.

4.3.2 Results

Let us now state our two main results about the response-time tail under ϵ-SMART policies. We will discuss the impact of the results first, and then provide the proofs at the end of the section. We start with our result for heavy-tailed service distributions.

Theorem 4.7 *Consider an M/GI/1 queue with* P *in* ϵ-SMART *and* $\overline{F} \in \mathcal{RV}(\alpha)$ *with* $\alpha \in (1,2)$. [1] *If there exists a* $0 < \delta < \sqrt{\alpha - 1}$ *such that* $\epsilon(x) = o(x^{(2-\delta)/(3-\alpha+\delta)})$ *as* $x \to \infty$*, then*

$$P(T^{\mathsf{P}} > y) \sim P(T^{\mathsf{SRPT}} > y) \sim P(X > (1-\rho)y).$$

Theorem 4.7 illustrates that the response-time tail of all ϵ-SMART policies is asymptotically equivalent to that of SRPT under heavy-tailed job sizes as long as $\epsilon(x)$ is "accurate enough," i.e., as long as $\epsilon(x) = o(x^{(2-\delta)/(3-\alpha+\delta)})$. Not only do all ϵ-SMART policies have asymptotically equivalent response-time tails, but the response-time tail is proportional to the tail of the job size distribution, which is the best possible (up to a multiplicative constant). Thus, Theorem 4.7 can be viewed as providing a concrete bound on how much the bias towards small jobs can be weakened while still ensuring the resulting policy behaves similarly to SRPT, and thus asymptotically optimally.

Now, we move to light-tailed service distributions.

Theorem 4.8 *Consider an M/GI/1 queue with* $\overline{F} \in \mathcal{LT}$ *and* P *in* ϵ-SMART*. Then,*

(i) if $x_U = \infty$ *and* $\epsilon^{-1}(x) \to \infty$ *as* $x \to \infty$*,* $\gamma(T^{\mathsf{P}}) = \gamma(T^{\mathsf{SRPT}}) = \gamma(B)$*,*

(ii) if $x_U < \infty$*,* $\gamma(B) \le \gamma(T^{\mathsf{P}}) \le \gamma(B_{\epsilon^{-1}(x_U)})$*.*

Theorem 4.8 again shows that, when $x_U = \infty$, the response-time tail of ϵ-SMART policies under light-tailed job-size distributions is also asymptotically equivalent to that of SRPT. In this case, the restriction on $\epsilon(x)$ is even weaker: all that is needed is $\epsilon^{-1}(x) \to \infty$ as $x \to \infty$. However, when $x_U < \infty$, the response-time tail of ϵ-SMART policies can differ from that of SRPT: it can be lighter or heavier. See [27] for the decay rate of SRPT. Finally, note that the expressions for $\gamma(B)$ and $\gamma(B_y)$ can in general be solved numerically using $\gamma(B_y) = \sup_{s\ge 0}\left[s - \lambda(Ee^{sB_y} - 1)\right]$, see [27] for details.

It is important to observe the contrast in the behavior of the response-time tail of ϵ-SMART policies under heavy-tailed and light-tailed service distributions. Under heavy-tailed job sizes, the response-time tail is near optimal, in the sense that no policy can have a response-time tail more than a multiplicative constant smaller. However, under light-tailed job sizes, the tail is nearly as heavy as possible, since no work conserving policy can have a tail heavier than that of a busy period. This seems to be a general tendency: policies that behave (near) optimally for heavy-tailed service times, can behave very poorly for light-tailed distributions, see for example [5, 6, 26].

A last comment about Theorems 4.7 and 4.8 is that it seems likely that the results can be extended to GI/GI/1 queues, using the approach of [26]; however, the proofs will become much more complicated.

Now, we will conclude the section with the proofs of Theorems 4.7 and 4.8. First, we will prove the result for heavy-tailed job sizes (Theorem 4.7) and then the result for light-tailed job sizes (Theorem 4.8). The proof of Theorem 4.7 relies on a technique introduced by Núñez-Queija in [25] and developed further in [5]. In particular, we will use the following sufficient conditions.

Condition 1 *There exists a* $g > 0$ *such that* $E[T(x)]^P/x \to g$ *as* $x \to \infty$.

Condition 2 *Let* $\overline{F} \in \mathcal{RV}(\alpha)$*. There exists* $\kappa > \alpha$ *such that*

$$P(T(x)^P - E[T(x)]^P > t) \le \frac{h(x)}{t^\kappa}$$

with $h(x) = o(x^{\kappa-\delta})$ *for some* $\delta > 0$.

Condition 3 $T(x)$ *is stochastically increasing in* x.

These conditions are sufficient to guarantee the following result, from [5].

Theorem 4.9 *If* $\overline{F} \in \mathcal{RV}(\alpha)$ *and Conditions 1 - 3 hold, then*

$$P(T(x)^P > gx) \sim P(X > x) \quad \text{as } x \to \infty.$$

As in [25], in our proof we will limit ourselves to $1 < \alpha < 2$ and $\kappa = 2$ and use Chebyshev's inequality to reduce Condition 2 to the following form:

$$P(T(x)^P - E[T(x)]^P > t) \le \frac{Var[T(x)]^{\mathsf{P}}}{t^2}.$$

Thus, we need only show that $Var[T(x)]^{\mathsf{P}} = o(x^{2-\delta})$ for some $\delta > 0$.

Proof. (of Theorem 4.7) To prove the result, we will make use of the upper and lower bounds in Theorem 4.1: let $U(x) = R(x, \epsilon(x))^{\mathsf{PSJF}} + W(\epsilon(x))^{\mathsf{SRPT}}$ and define likewise $L(x) = R(x, \epsilon^{-1}(x))^{\mathsf{SRPT}} + W(\epsilon^{-1}(x))^{\mathsf{PSJF}}$. Further, let U and L be the corresponding deconditioned bounds, so that $L \le_{st} T^{\mathsf{P}} \le_{st} U$. We will prove that Conditions 1-3 hold for both U and L, from which it follows that $P(U > x) \sim P(L > x) \sim P(X > (1-\rho)x)$. Since U and L stochastically upper and lower bound T^{P}, this will complete the proof.

To begin, note that Condition 3 holds trivially for both the $U(x)$ and $L(x)$.

To show that Condition 1 holds, we will prove that both $E[U(x)]/x$ and $E[L(x)]/x$ converge to $1/(1-\rho)$ as $x \to \infty$. Since $x \le \epsilon(x)$, we have that $\epsilon(x) \to \infty$ as $x \to \infty$. By

[1] This result can be extended to distributions of "intermediate regular variation" with $\alpha > 1$, as discussed in [25], but we limit ourselves to $\mathcal{RV}$ and $1 < \alpha < 2$ to ease the readability of the paper. The case of infinite variance is of practical interest.

Theorem 4.1, we can write

$$\lim_{x\to\infty} \frac{E[U(x)]}{x} \leq \lim_{x\to\infty} \frac{1}{x}\left(\frac{x}{1-\rho(\epsilon(x))} + \frac{\lambda\widetilde{m_2}(\epsilon(x))}{2(1-\rho(\epsilon(x)))^2}\right)$$
$$= \frac{1}{1-\rho} + \frac{\lambda}{2(1-\rho)^2}\lim_{x\to\infty}\frac{\widetilde{m_2}(\epsilon(x))}{x}.$$

Using that $\epsilon^{-1}(x) \to \infty$ as $x \to \infty$ since $\epsilon(x) = o(x^{(2-\delta)/(3-\alpha+\delta)})$, we can write

$$\lim_{x\to\infty} \frac{E[L(x)]}{x} \leq \lim_{x\to\infty} \frac{1}{x}\left(\int_0^x \frac{dt}{1-\rho(\epsilon^{-1}(t))} + \frac{\lambda m_2(\epsilon^{-1}(x))}{2(1-\rho(\epsilon^{-1}(x)))^2}\right)$$
$$\leq \frac{1}{1-\rho} + \frac{\lambda}{2(1-\rho)^2}\lim_{x\to\infty}\frac{m_2(\epsilon^{-1}(x))}{x}$$

We need only show that in both expressions the second term disappears in the limit. Since $m_2(\epsilon^{-1}(x)) \leq \widetilde{m_2}(\epsilon(x)) = o(x^{1-\delta})$, it is enough to show that $\widetilde{m_2}(\epsilon(x)) = o(x)$. To prove this, begin by noticing that $\widetilde{m_2}(\epsilon(x)) = o(\epsilon(x)^{2-\alpha})$, and that, by assumption, $\epsilon(x) = o(x^{(2-\delta)/(3-\alpha+\delta)})$. Using the assumption $\delta \leq \sqrt{\alpha - 1}$, a little algebra now yields that $(2-\alpha)(2-\delta) \leq (1-\delta)(3-\alpha+\delta)$. This implies $\widetilde{m_2}(\epsilon(x)) = o(x^{1-\delta})$.

Next, we will verify that Condition 2 holds. It follows from the analysis in [25] that, for $\eta > 0$, the following asymptotics hold:

$$Var[R(x,\epsilon(x))]^{\mathsf{PSJF}} \leq Var[R(\epsilon(x))]^{\mathsf{PSJF}} = o(\epsilon(x)^{3-\alpha+\eta}),$$
$$Var[W(\epsilon(x))]^{\mathsf{SRPT}} = o(\epsilon(x)^{3-\alpha+\eta}),$$
$$Var[R(x,\epsilon^{-1}(x))]^{\mathsf{SRPT}} \leq Var[R(x)]^{\mathsf{SRPT}} = o(x^{3-\alpha+\eta}),$$
$$Var[W(\epsilon^{-1}(x))]^{\mathsf{PSJF}} \leq Var[W(x)]^{\mathsf{PSJF}} = o(x^{3-\alpha+\eta}).$$

Choosing $\eta = \delta$, we have

$$Var[U(x)] = o(\epsilon(x)^{3-\alpha+\varepsilon}) = o(x^{2-\delta}).$$

Choosing $\eta = \delta + \alpha - 1$ yields the same result for $Var[L(x)]$.

Thus, conditions 1-3 hold both for $U(x)$ and $L(x)$, and we can conclude that

$$P(L > x) \sim P(U > x) \sim P(X > (1-\rho)x).$$

As a consequence, $P(T > x) \sim P(X > (1-\rho)x)$, which finishes the proof. □

Next, we will prove Theorem 4.8. This proof follows along similar lines as the related results for SMART and FB [26], so we shall refer to those papers for some technical auxiliary results.

PROOF. (of Theorem 4.8) Let T be the response time of a tagged job with (random) service time X arriving at time 0. First, note that for all work-conserving disciplines, $\gamma(T^{\mathsf{P}}) \geq \gamma(B)$, see [26]. So, to prove the statement we will argue only about upper bounds on $\gamma(T^{\mathsf{P}})$, and hence, lower bounds on $P(T > x)$ as $x \to \infty$.

First, we consider the case that $x_U = \infty$, where we will show that $\gamma(T) \leq \gamma(B)$. To do so, let A be last interarrival time before 0, and let X_0 be the service time of the last job arriving before time 0. Let $y < x_U$ and $0 < \delta < y$. If $A \leq \delta$, $X \geq y$, and $X_0 < \epsilon^{-1}(y)$, then, under any ϵ-SMART policy P, at time 0, X_0 will have priority over the tagged job with service time X. Thus, as soon as X_0 is served, it starts a busy period of jobs that will also have priority over X if their service time is less than $\epsilon^{-1}(y) - \delta$, by the Bias property. Thus, we have

$$P(T^{\mathsf{P}} > x) \geq P(T^{\mathsf{P}} > x, X \geq y, A \leq \delta, X_0 < \epsilon^{-1}(y))$$
$$\geq P(X \geq y)P(A \leq \delta)P(X_0 < \epsilon^{-1}(y))P(B_{\epsilon^{-1}(y)-\delta} \geq x).$$

Note that we assume $P(X_0 < \epsilon^{-1}(y)) > 0$, because otherwise all policies fall into ϵ-SMART and no useful bounds are possible.

Now, since $P(A \leq \delta) > 0$, taking limits gives us

$$\lim_{x\to\infty} \frac{1}{x}\log P(T^{\mathsf{P}} > x) \geq \lim_{x\to\infty}\frac{1}{x}P(B_{\epsilon^{-1}(y)-\delta} \geq x) \qquad (3)$$
$$= \gamma(B_{\epsilon^{-1}(y)-\delta}).$$

Finally, we note that, by assumption, $\epsilon^{-1}(y) - \delta \to \infty$ as $y \to \infty$, and use that $\gamma(B_x)$ converges to $\gamma(B)$ as $x \to \infty$, see [27], to conclude that $\gamma(T^{\mathsf{P}}) \leq \gamma(B)$.

Finally, we consider the case that $x_U < \infty$. Since $\gamma(B_x)$ converges to $\gamma(B_y)$ as $x \to y$, it follows from (3) that $\gamma(T^{\mathsf{P}}) \leq \gamma(B_{\epsilon^{-1}(x_U)})$. □

Before leaving this topic, it is worth noting that the inequalities in the second part of Theorem 4.8 are achieved by some policies in ϵ-SMART. For example, if jobs are served according to SRPT, with the exception that no distinction is made between service times in the interval $[x_M, x_U]$, for some $x_M < x_U$. The behavior of the decay rate of this discipline is then the same as that of SRPT with an alternative service distribution, namely where all the probability mass from the interval $[x_\varepsilon^{-1}(x_U), x_U]$ is concentrated in one point. The decay rate of such an SRPT queue is $\gamma(B_{\epsilon^{-1}(x_U)})$, which is strictly larger than $\gamma(B)$, see [27].

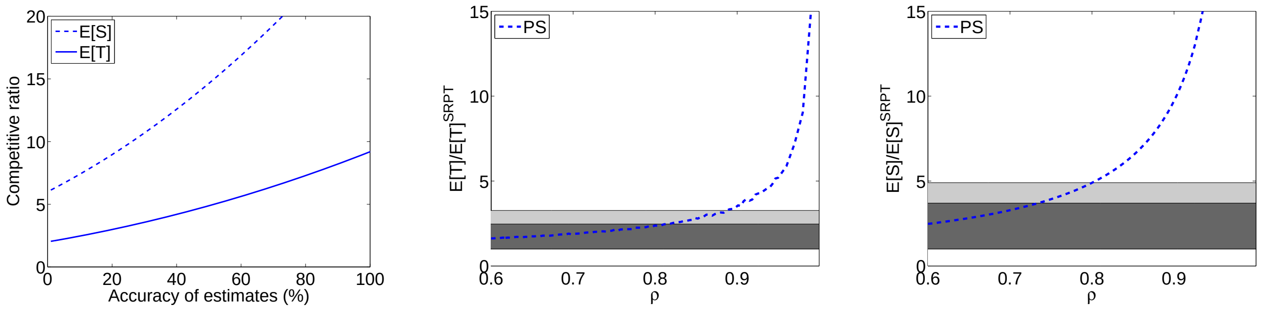
5. APPLICATIONS OF ϵ-SMART

So far, we have discussed the class of ϵ-SMART policies in an abstract way, referring very little to the type of inexact job-size information being used to schedule. In this section, we will consider two practical causes for inexact job-size information, and specialize our results to these cases.

To ground our discussion in this section, we will consider a specific application: a static web server. However, the same issues we will discuss have also been raised in the setting of server side peer-to-peer scheduling [30], wireless networks [15, 21], and beyond.

At a high level, the operation of a web server that serves primarily static requests is very simple. In order to fulfill a request, the web server must retrieve the file and then send the file over the outgoing link. Typically the amount of bandwidth at the web server is the bottleneck device since purchasing more bandwidth is much more expensive than upgrading the disks or CPUs at the web server [24, 7]. Even a modest web server can saturate a T3 or 100Mbps Ethernet connection. Thus, much of the delay experienced by requests for files is a result of queueing for bandwidth.

In standard web server designs, such as Apache [36] and Flash [29] servers, the bandwidth is allocated by cycling through the queued files, giving each a small slice of service. We can model this behavior as Processor Sharing (PS), which gives an equal share of the service capacity to each job in the queue at all times. However, recent designs, e.g., [14, 33, 20, 19, 13], have achieved dramatic reductions in user response times by using variations of SRPT to allocate bandwidth. But, the implemented policies differ from pure SRPT in two main ways:



(a) Accuracy of estimate (σ) vs. competitive ratio of $E[T]$ and $E[S]$

(b) ρ vs. $E[T]/E[T]^{\mathsf{SRPT}}$

(c) ρ vs. $E[S]/E[S]^{\mathsf{SRPT}}$

Figure 3: An illustration of the impact of the accuracy of job-size estimates. Plot (a) shows the relationship between the accuracy of the estimates and the worst-case competitive ratios for $E[T]$ and $E[S]$ of the resulting policy. Plots (b) and (c) contrast the attainable $E[T]$ and $E[S]$ under ϵ-SMART policies with PS as a function of load. In (b) and (c), ϵ-SMART policies with accuracy 10% ($< 25\%$) all fall in the dark (light) grey region. In (a)-(c), the service distribution is Pareto with mean 1 and $\alpha = 1.2$.

1. The file size of a request is taken to be the service demand of the request, while the true service demand of a request also includes many other factors, such as the CPU time and, more importantly, the network path. Thus, the file size is only an estimate of the true service demand.

2. Only a finite number of priority classes are used (typically 5-10), instead of the continuum of classes used by SRPT. Thus, jobs are given priority based on whether their remaining size is "small," "medium," "large," etc., instead of based on their exact remaining size.

We will discuss these two issues in the next two sections. They will provide two examples of choices of $\epsilon(x)$ that are of practical interest; however it is important to note that many other forms of $\epsilon(x)$ are also interesting to study.

5.1 Using job-size estimates

The fact that the true service demand of a request to a web server differs from the file size (which is used to schedule) has a significant impact on the performance of designs based on SRPT. This fact was observed in a number of papers [33, 20]. As a result, designers have searched for ways to improve estimates of the true service demand, and come to the conclusion that it is important to supplement the file size information with information about the round trip time (RTT) that the request will experience in the network [33, 19]. This is because the RTT will limit the bandwidth of the connection, and thus have a significant impact on the service demand of the request. However, measuring the RTT accurately requires significant overhead, that increases with the accuracy desired. So, an important design tradeoff to understand is the balance between the accuracy of job-size estimates and response times.

Prior to our work, this tradeoff has never been addressed analytically. Instead, simulation studies were used, e.g., [20, 19]. We can now use ϵ-SMART to provide an explicit analytic characterization of this tradeoff.

In particular, we will use a simple model of TCP throughput to highlight the impact of the RTT on the service demand of requests to the web server. The throughput (and thus service rate) given to a flow is known to be inversely proportional to the RTT, see for example, [28, 18]. Thus, we can *roughly* approximate the true service demand, S, of a request as $S = \alpha f RTT$, where f is the file size and α is a constant that depends on the loss rate in the network and the details of TCP. Now, the scheduler of a web server will use some estimate of $\alpha f RTT$, say $\widehat{\alpha} f \widehat{RTT}$. Suppose, now that the estimator is such that the error is bounded. Specifically, suppose $\widehat{\alpha}\widehat{RTT} < (1+\sigma)\alpha RTT$ and $\alpha RTT < (1+\sigma)\widehat{\alpha}\widehat{RTT}$. Note that a multiplicative error is appropriate because one would expect the accuracy of the estimate to be proportional to the length of the RTT. However, the assumption of a strict upper bound on the error σ is clearly an approximation (which current work is attempting to remove).

Given this scenario, we can immediately apply our results to characterize the impact of the error in job-size estimates by using $\epsilon(x) \le (1 + \sigma)x$. Further, we will assume that the true service demands follow a Pareto distribution with infinite variance. This is a natural assumption given the large number of studies that have found heavy-tailed service demands in networks, e.g., [1, 9, 10].

Corollary 5.1 *Consider an M/GI/1 queue with* P *in* ϵ-SMART *and* $X \sim$ *Pareto*(α) *for some* $1 < \alpha < 2$. *If* $\epsilon(x) \le (1+\sigma)x$, *then*

$$E[T]^{\mathsf{P}} \le 2(1+\sigma)^{2\alpha} E[T]^{\mathsf{SRPT}}, \tag{4}$$

$$E[S]^{\mathsf{P}} \le 6(1+\sigma)^{2\alpha} E[S]^{Optimal}, \quad \textit{and} \tag{5}$$

$$P(T^{\mathsf{P}} > y) \sim P(T^{\mathsf{SRPT}} > y). \tag{6}$$

Corollary 5.1 guarantees that running SRPT on (bounded) job-size estimates will provide a near optimal $E[T]$, $E[S]$, and response-time tail. Further, it provides an explicit characterization of the improvement in worst-case performance that results from improving the accuracy of job-size estimates. It is important to point out that there are no distributional assumptions made about the errors of the estimates. In particular, the result holds even if adversarial errors are assumed.

We illustrate the bounds on $E[T]$ and $E[S]$ from Corollary 5.1 in Figure 3, where we also compare the performance of ϵ-SMART policies with PS, the traditional web server design. Interestingly, we see that even very poor estimates are useful under high load. This is a consequence of Theorem 4.6, which says that the heavy-traffic behavior of all ϵ-SMART policies is superior to that of PS. However, when the load is not high, $E[T]^{\mathsf{PS}}$ and $E[S]^{\mathsf{PS}}$ are smaller than the corresponding bounds on ϵ-SMART policies if job-size estimates are not accurate enough. Note, however, that the upper bounds on ϵ-SMART policies are very loose for low loads, since they must hold independently of the load and service distribution. Further, it is important to remember that Corollary 5.1 holds even under adversarial errors in job-size estimates; thus one would expect $E[T]$ and $E[S]$ to be significantly below the upper bound in practice.

Proof. (Corollary 5.1) We first calculate the following:

$$\frac{1-\rho(\epsilon(x))}{1-\rho(x)} \geq 1 - \frac{\rho(\epsilon(x))-\rho(x)}{\rho-\rho(x)} = 1 - \frac{\frac{1}{x^{\alpha-1}} - \frac{1}{\epsilon(x)^{\alpha-1}}}{\frac{1}{x^{\alpha-1}}}$$
$$= (x/\epsilon(x))^{\alpha-1} \geq (1+\sigma)^{1-\alpha}.$$

Applying Theorem 4.5 with $1-\gamma = (1+\sigma)^{1-\alpha}$ then completes the proof of (4) and (5).

To prove (6), we use Theorem 4.7. We first note that a $Pareto(\alpha)$ distribution is regularly varying with rate α. For all $0 < \delta < (\alpha-1)/2$, we have $\epsilon(x) = (1+\sigma)x = o(x^{(2-\delta)/(3-\alpha+\delta)})$. Noting that $(\alpha-1)/2 \leq \sqrt{\alpha-1}$ finishes the proof. □

5.2 Limited number of priority classes

Web server designs use a limited number of priority classes due to the overhead associated with maintaining different priority classes. Thus, limiting the number of priority classes as much as possible is desirable. However, as the number of priority classes drops, $E[T]$ can increase significantly. Thus, there is a design tradeoff between limiting the number of priority classes and achieving a desired $E[T]$.

Though policies using a limited number of priority classes have been studied extensively in the literature, to the best of our knowledge, there are no simple bounds that relate the number of priority classes to the mean response time attainable. The reason for this is the complexity of the formulas available for $E[T]$ under these policies. However, ϵ-SMART can help to provide results in this area.

In particular, policies that use a limited number of priority classes fall into the ϵ-SMART classification. To illustrate this, let us define a k-class preemptive priority queue as follows. A k-class preemptive uses $k+1$ thresholds, $0 = t_0 < t_1 < \ldots < t_k = \infty$. Jobs in with size in $[t_i, t_{i+1})$ are called *class-i jobs*, and are given preemptive priority over class $j > i$ jobs. To see that k-class policies are in ϵ-SMART, we can define $\epsilon(x) = \sum_{i=1}^{k} t_i I(x \in [t_{i-1}, t_i))$. Note that instead of prioritizing based on size, we could also have prioritized based on any SMART policy, and the resulting k-class policy would fall in ϵ-SMART.

Given that k-class policies are in ϵ-SMART, we can now apply Theorems 4.5, 4.7, and 4.8 to characterize the achievable performance. It turns out the the cases of unbounded ($x_U = \infty$) and bounded ($x_U < \infty$) service distributions need to be handled separately.

Unbounded service distributions

In the case of unbounded service distributions, we cannot apply Theorems 4.5, 4.7, and 4.8 directly. There are two reasons for this. First, there exists no finite σ such that $\epsilon(x) < (1+\sigma)x$ for all x. For this condition to hold, the number of priority classes would need to be infinite. Thus, Theorem 4.5 does not apply. Second, $\epsilon(x) = \infty$ for $x > t_{k-1}$, thus Theorems 4.7 and 4.8 do not apply.

It may seem problematic that our results do not apply in this case, but it is necessary: when $E[X^2] = \infty$, all k-class priority queues have $E[T] = \infty$, and thus cannot be constant competitive since $E[T]^{\mathsf{SRPT}} < \infty$. Further, the response-time tail of k-class priority queues is strictly heavier than that of SRPT in this setting. Thus, if we were to loosen restrictions on $\epsilon(x)$ so that results could apply to k-class priority queues under unbounded service distributions, the performance guarantees would be much weaker.

Bounded service distributions

By contrast, when the service distribution is bounded, Theorems 4.5 and 4.8 apply directly to k-class policies.

In particular, we can immediately apply Theorem 4.8 to attain a bound on the response-time tail of the k-class policy. Further, it is also easy to apply Theorem 4.5. Given a set of thresholds $\{t_i\}$, we can calculate the σ and γ such that Conditions I and II hold as follows:

$$\sigma = \max_{1\leq i\leq k} \left\{ \frac{t_i}{t_{i-1}} \right\} - 1$$
$$\gamma = 1 - \min_{1\leq i\leq k} \left\{ \frac{1-\rho(t_i)}{1-\rho(t_{i-1})} \right\}.$$

With this choice of σ and γ, it follows that the k-class policy is $2\left(\frac{1+\sigma}{1-\gamma}\right)^2$-competitive. Clearly, the tradeoff between σ and γ is very dependent on the service distribution, but given the service distribution and the thresholds, calculating the competitive ratio is straightforward. Further, this result give new insight into the improvement (in terms of worst-case $E[T]$) that comes from increasing the number of priority classes. For example, it is possible to show that adding thresholds at $(t_i + t_{i+1})/2$ that split each original class in two (i.e., doubling the number of priority classes), will decrease σ by a factor of 2.

6. CONCLUDING REMARKS

The heuristic of "favoring small jobs" is a commonly accepted way of providing small response times, and is being increasingly applied in the design of modern computer systems. However, the idealized policies studied analytically differ significantly from the policies applied in practice. In this paper, we have introduced the ϵ-SMART classification as a tool for understanding the impact of these differences.

Specifically, the ϵ-SMART class is defined so as to include both the idealized policies that "favor small jobs", such as SRPT and PSJF, as well as the variations of these idealized policies that are used in real systems. The ϵ-SMART class includes (i) hybrid policies, (ii) policies that prioritize by means of job-size estimates, and (iii) policies that use only a finite number of priority classes.

The main contribution of this paper is to supply *simple and tight* performance bounds on the mean response time, mean slowdown, response-time tail, and the conditional response time of policies in the ϵ-SMART class. It turns out that in each case, the practical variations included in ϵ-SMART provide performance similar to SRPT. Further, the results show an explicit relationship between the degree of variation from SRPT (as expressed by the function $\epsilon(x)$) and the resulting performance.

We have also discussed in detail two important applications of the ϵ-SMART classification. One application provides the first bounds in the literature on the performance of size-based scheduling polices that use inexact job-size information. These bounds explicitly show the relationship between the (worst-case) mean response time and the accuracy of job-size estimates. In a second application, we give simple bounds on the performance of policies that use only a finite number of priority levels. These bounds provide new insight into the relationship between mean response time and the number of priority classes used.

This work is only a first step towards an analytic understanding of the impact of inexact job-size information; there remain a large number of interesting questions for future work. For example, for the ϵ-SMART classification, it would be interesting to extend the results for the response-time tail to the GI/GI/1 queue, and determine if and how the restrictions on $\epsilon(x)$ change. Further, it would be interesting to extend the ϵ-SMART class to include probabilistic (rather than deterministic) functions $\epsilon(x)$. This would allow, for instance, the error in job-size estimates to follow an unbounded distribution, which is a more accurate model of real system behavior. Finally, it would also be interesting to derive results for specific ϵ-SMART policies that are of interest, such as SRPT on job-size estimates.

7. REFERENCES

[1] M. Arlitt and C. Williamson. Web server workload characterization: the search for invariants. In *Proc. of ACM Sigmetrics*, 1996.

[2] N. Bansal. On the average sojourn time under M/M/1/SRPT. *Oper. Res. Letters*, 22(2):195–200, 2005.

[3] N. Bansal and D. Gamarnik. Handling load with less stress. *Queueing Systems*, 54(1):45–54, 2006.

[4] N. Bingham, C. Goldie, and J. Teugels. *Regular Variation.* Cambridge University Press, 1987.

[5] S. Borst, O. Boxma, R. Nunez-Queija, and B. Zwart. The impact of the service discipline on delay asymptotics. *Performance Evaluation*, 54:175–206, 2003.

[6] O. Boxma and B. Zwart. Tails in scheduling. *Perf. Eval. Rev.*, 34(4):13–20, 2007.

[7] A. Cockcroft. Watching your web server. http://www.theunixinsider.com, 1996.

[8] R. W. Conway, W. L. Maxwell, and L. W. Miller. *Theory of Scheduling.* Addison-Wesley Publishing Company, 1967.

[9] M. Crovella and A. Bestavros. Self-similarity in world wide web traffic: Evidence and possible causes. *Trans. on Networking*, 5(6):835–846, 1997.

[10] A. B. Downey. A parallel workload model and its implications for processor allocation. In *Proc. of High Performance Distributed Computing*, pages 112–123, August 1997.

[11] E. Friedman and S. Henderson. Fairness and efficiency in web server protocols. In *Proc. of ACM Sigmetrics*, 2003.

[12] J. Gehrke, S. Muthukrishnan, R. Rajaraman, and A. Shaheen. Scheduling to minimize average stretch online. In *40th Annual symposium on Foundation of Computer Science*, pages 433–443, 1999.

[13] M. Gong and C. Williamson. Simulation evaluation of hybrid SRPT scheduling policies. In *Proc of IEEE MASCOTS*, 2004.

[14] M. Harchol-Balter, B. Schroeder, M. Agrawal, and N. Bansal. Size-based scheduling to improve web performance. *ACM Transactions on Computer Systems*, 21(2), May 2003.

[15] M. Hu, J. Zhang, and J. Sadowsky. A size-aided opportunistic scheduling scheme in wireless networks. In *Globecom*, 2003.

[16] A. Kherani. Sojourn times in (discrete) time shared systems and their continuous time limits. In *Proc. of ValueTools*, 2006.

[17] A. A. Kherani and R. Nunez-Queija. TCP as an implementation of age-based scheduling: fairness and performancs. In *IEEE Infocom*, 2006.

[18] A. Kumar. Comparative performance analysis of versions of TCP in a local network with a lossy link. *IEEE/ACM Trans./ Networking*, 6:485–498, 1998.

[19] D. Lu, P. Dinda, Y. Qiao, and H. Sheng. Effects and implications of file size/service time correlation on web server scheduling policies. In *Proc. of IEEE Mascots*, 2005.

[20] D. Lu, H. Sheng, and P. Dinda. Size-based scheduling policies with inaccurate scheduling information. In *Proc. of IEEE Mascots*, 2004.

[21] R. Mangharam, M. Demirhan, R. Rajkumar, and D. Raychaudhuri. Size matters: Size-based scheduling for MPEG-4 over wireless channels. In *SPIE & ACM Proceedings in Multimedia Computing and Networking*, pages 110–122, 2004.

[22] D. McWherter, B. Schroeder, N. Ailamaki, and M. Harchol-Balter. Priority mechanisms for OLTP and transactional web applications. In *Int. Conf on Data Engineering*, 2004.

[23] D. McWherter, B. Schroeder, N. Ailamaki, and M. Harchol-Balter. Improving preemptive prioritization via statistical characterization of OLTP locking. In *Int. Conf on Data Engineering*, 2005.

[24] Microsoft. The arts and science of web server tuning and internet information servers 5.0. microsoft technet - insights and answers for it professionals. http://www.microsoft.com/technet/, 2001.

[25] R. Nunez-Queija. Queues with equally heavy sojourn time and service requirement distributions. *Ann. Oper. Res*, 113:101–117, 2002.

[26] M. Nuyens, A. Wierman, and B. Zwart. Preventing large sojourn times using SMART scheduling. *Oper. Res.*, in press.

[27] M. Nuyens and B. Zwart. A large-deviations analysis of the GI/GI/1 SRPT queue. *Queueing Sys.*, 54(2):85–97, 2006.

[28] J. Padhye, V. Firoiu, D. Towsley, and J. Kurose. Modeling TCP throughput: a simple model and its empirical validation. *ACM Computer Communication Review*, 28:303–314, 1998.

[29] V. S. Pai, P. Druschel, and W. Zwaenepoel. Flash: An efficient and portable web server. In *Proc. of USENIX*, 1999.

[30] Y. Qiao, D. Lu, R. Bustamante, and P. Dinda. Looking at the server side of peer-to-peer systems. Technical Report NWU-CS-04-37, Northwestern University, 2004.

[31] I. A. Rai, G. Urvoy-Keller, and E. Biersack. Analysis of LAS scheduling for job size distributions with high variance. In *Proc. of ACM Sigmetrics*, 2003.

[32] I. A. Rai, G. Urvoy-Keller, M. Vernon, and E. W. Biersack. Performance modeling of LAS based scheduling in packet switched networks. In *Proc. of ACM Sigmetrics-Performance*, 2004.

[33] M. Rawat and A. Kshemkalyani. SWIFT: Scheduling in web servers for fast response time. In *Symp. on Net. Comp. and App.*, 2003.

[34] D. Raz, H. Levy, and B. Avi-Itzhak. A resource-allocation queueing fairness measure. In *Proc. of ACM Sigmetrics-Performance*, 2004.

[35] L. E. Schrage. A proof of the optimality of the shortest remaining processing time discipline. *Operations Research*, 16:678–690, 1968.

[36] The Apache software foundation. The Apache web server.

[37] A. Wierman. Fairness and classifications. *Perf. Eval. Rev.*, 34(4):4–12, 2007.

[38] A. Wierman and M. Harchol-Balter. Classifying scheduling policies with respect to unfairness in an M/GI/1. In *Proc. of ACM Sigmetrics*, 2003.

[39] A. Wierman and M. Harchol-Balter. Classifying scheduling policies with respect to higher moments of response time. In *Proc. of ACM Sigmetrics*, 2005.

[40] A. Wierman, M. Harchol-Balter, and T. Osogami. Nearly insensitive bounds on SMART scheduling. In *Proc. of ACM Sigmetrics*, 2005.

[41] C. Yang, A. Wierman, S. Shakkottai, and M. Harchol-Balter. Tail asymptotics for policies favoring short jobs in a many-flows regime. In *Proc. of ACM Sigmetrics*, 2006.

APPENDIX

Due to space constraints, we could not include the following proof in the paper. However, we include it here for the benefit of the reviewers.

Proof. (of Theorem 4.6) Since conditions (I) and (II) hold, Theorem 4.5 implies that P is constant competitive. Since PSJF is an ϵ-SMART policy, Theorem 4.5 implies that $E[T]^{\mathsf{SRPT}} = \Theta(E[T]^{\mathsf{PSJF}})$, so that $E[T]^{\mathsf{P}} = \Theta(E[T]^{\mathsf{PSJF}})$, and similarly for $E[S]$. Thus, we can limit ourselves to analyzing PSJF.

We will first focus on $E[T]^{\mathsf{PSJF}}$. Starting with $E[R]^{\mathsf{PSJF}}$, we can easily see that

$$E[R]^{\mathsf{PSJF}} = \int_0^\infty \frac{xf(x)}{1-\rho(x)}dx = \frac{1}{\lambda}\log\left(\frac{1}{1-\rho}\right)$$

Next, we focus on the waiting time. First, note that for any service distribution with finite mean and $x_U = \infty$, we have $m_2(x) = o(x)$. Thus, for every $\varepsilon > 0$, there is a $y(\varepsilon)$ such that for all $x > y(\varepsilon)$, $m_2(x)/x < \varepsilon$. Now, recall that:

$$E[W]^{\mathsf{PSJF}} = \int_0^\infty \frac{\lambda m_2(x)f(x)}{2(1-\rho(x))^2}dx$$

We will first truncate the integral at $y(\varepsilon)$:

$$\begin{aligned}\int_0^{y(\varepsilon)} \frac{\lambda m_2(x)f(x)}{2(1-\rho(x))^2} &\le \frac{\lambda m_2(y(\varepsilon))}{2(1-\rho(y(\varepsilon)))^2}\int_0^{y(\varepsilon)} f(x)dx \\ &= \frac{\lambda m_2(y(\varepsilon))F(y(\varepsilon))}{(1-\rho(y(\varepsilon)))^2},\end{aligned}$$

which is simply $O(1)$ as $\rho \to 1$.

Since $\rho'(x) = \lambda x f(x)$, we have, as $\rho \to 1$,

$$\begin{aligned}E[T]^{PSJF} &= \int_{y(\varepsilon)}^\infty \frac{\lambda m_2(x)f(x)}{2(1-\rho(x))^2}dx + O(1) \\ &= \frac{1}{2}\int_{y(\epsilon)}^\infty \frac{\rho'(x)}{(1-\rho(x))^2}\frac{m_2(x)}{x}dx + O(1) \\ &\le \frac{1}{2}\frac{\varepsilon}{1-\rho} + O(1).\end{aligned}$$

Since $\varepsilon > 0$ was arbitrary, this completes the proof for $E[T]$.

We now move to $E[S]$. As with $E[T]$, it is enough to show that $E[S]^{\mathsf{PSJF}} = o(1/(1-\rho))$. Let $\varepsilon > 0$. Starting with the slowdown of residence time, we have

$$\begin{aligned}\int_0^\infty \frac{1}{x}\frac{xf(x)}{1-\rho(x)}dx &= \int_0^{1/\varepsilon} \frac{f(x)}{1-\rho(x)}dx + \frac{1}{\lambda}\int_{1/\varepsilon}^\infty \frac{1}{x}\frac{\rho'(x)}{1-\rho(x)}dx \\ &\le \frac{F(1/\varepsilon)}{1-\rho(1/\varepsilon)} + \frac{\varepsilon}{\lambda}\log\left(\frac{1}{1-\rho}\right) \\ &= O(1) + \frac{\varepsilon}{\lambda}\log\left(\frac{1}{1-\rho}\right),\end{aligned}$$

which gives us that the slowdown of the residence time is $o(1/(1-\rho))$. The argument is similar for the slowdown of the waiting time. Here, we use that $\lambda m_2(x) \le x\rho(x)$:

$$\begin{aligned}\int_0^\infty \frac{1}{x}\frac{\lambda m_2(x)f(x)}{2(1-\rho(x))^2}dx &\le \int_0^{1/\varepsilon} \frac{\rho(x)f(x)}{2(1-\rho(x))^2}dx + \frac{1}{\lambda}\int_{1/\varepsilon}^\infty \frac{1}{x}\frac{\rho'(x)\rho(x)}{2(1-\rho(x))^2}dx \\ &\le \frac{\rho(1/\varepsilon)F(1/\varepsilon)}{2(1-\rho(1/\varepsilon))^2} + \frac{\varepsilon}{2\lambda}\frac{1}{1-\rho} \\ &= O(1) + \frac{1}{2\lambda}\frac{\epsilon}{1-\rho}.\end{aligned}$$

This implies that the slowdown of the waiting time is $o(1/(1-\rho))$. Combining the two pieces gives us that $E[S] = o(1/(1-\rho))$, and completes the proof. □