Analysis of SRPT Scheduling: Investigating Unfairness

Nikhil Bansal   Mor Harchol-Balter

July 2000
CMU-CS-00-149

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Abstract

The Shortest-Remaining-Processing-Time (SRPT) scheduling policy has long been known to be optimal for minimizing mean response time. Despite this fact, SRPT scheduling is rarely used in practice. It is believed that the performance improvements of SRPT over other scheduling policies with respect to mean response time stem from the fact that SRPT unfairly penalizes the large jobs in order to help the small jobs. This belief has caused people to instead adopt "fair" scheduling policies such as Processor-Sharing (PS), which produces the same expected slowdown for jobs of all sizes.

This paper investigates formally via analysis the problem of unfairness in SRPT. The analysis assumes an M/G/1 model, but emphasizes heavy tailed job size distributions, as are characteristic of empirical workloads. We end with a trace-driven simulation experiment which agrees with the analysis, even though arrivals are no longer Poisson.

Keywords: M/G/1, processor sharing, shortest processing remaining time, time-sharing, priority, unfairness, starvation, slowdown, heavy-tailed behavior, high variance, single server, job scheduling.

1 Introduction

It has been long known that always giving service to the job with the shortest-remaining-processing-time (SRPT) is the optimal scheduling policy with respect to minimizing the mean response time. Yet, many existing schedulers time-share the processor equally among all jobs, giving each job an equal quantum of service. For example, a web-server today time shares between its many concurrent open connections, giving each an approximately equal share of processing time. In the limit, as the size of the quantum goes to zero, this "fair-share" scheduling policy is known as Processor Sharing (PS).

There are reasons why the optimal policy SRPT is not prevalent in practice. In some cases, it is because the size of a job (its processing requirement) is not known in advance, so SRPT cannot be applied. However, in several applications this is not the case, and it is possible to reasonably estimate the size of a job. For example, in the case of static web requests to a web server, a job's processing requirement is proportional to the file size requested, which is known by the server. Likewise in several database applications, the processing requirement for a query may be estimated in advance.

A second objection to switching to SRPT is that it is not clear whether the performance improvements of SRPT over traditional scheduling policies like PS are significant. Comparing SRPT with other policies is not easy given the complex nature of existing performance formulas for SRPT.

However, the foremost and very commonly cited objection to using SRPT is the fear that large jobs may starve under SRPT [1, 25, 26, 23]. It is often stated that the huge average performance improvements of SRPT over other scheduling policies stem from the fact that SRPT unfairly penalizes the large jobs in order to help the small jobs. It is believed that the performance of small jobs cannot be improved without hurting the large jobs (see Conservation Law, Section 2) and thus "large jobs starve under SRPT".

This paper will investigate the objections cited above. Before we can state our results, we need to define the performance metrics and the workloads which we use.

The performance metrics we use throughout are *response time* and *slowdown*. The response time of a job (a.k.a. turnaround time, flow time) is the time from when the job first arrives at the system until it departs the system. The slowdown of a job (a.k.a. stretch, normalized response time) is the ratio of its response time to its size. The slowdown metric is important because it helps to evaluate unfairness. For example, in an M/G/1 system with PS scheduling, all jobs (long and short) experience the same expected slowdown (hence PS is "fair").

It turns out that the job size distribution is important with respect to evaluating SRPT. We will therefore assume a general job size distribution. We will also concentrate on the special case of *heavy-tailed* job size distributions. Heavy-tailed (HT) job size distributions appear to fit many recent measurements of computing systems, as described in Section 3.

In this paper we will prove the following results about SRPT, all in the context of an M/G/1 queue with load $\rho < 1$.

On the topic of mean performance improvements:

1. Although it is well-known that SRPT scheduling optimizes mean response time, it is not known how SRPT compares with PS with respect to mean slowdown. We prove that SRPT scheduling always results in lower mean slowdown than PS scheduling (Theorem 1, Section 4).

2. Given that SRPT improves performance over PS both with respect to mean response time and mean slowdown, we next investigate the magnitude of the improvement. We find that for HT job size distributions the improvement is very significant under high loads. In fact mean response time improves by a factor of 3-5 under SRPT and mean slowdown improves by an order of magnitude under SRPT as compared with PS. In general we prove that for all job size distributions as the load approaches one, the mean response time under SRPT improves upon the mean response time under PS by at least a factor of 2 and likewise for mean slowdown. (Corollaries 1 and 2, Section 4).

On the topic of starvation we first show a very counter intuitive result.

1. The performance improvement of SRPT over PS does not usually come at the expense of the large jobs (Section 5.1, Theorem 2). In fact, we observe that for most HT job size distributions every single job, including the very largest, prefers SRPT to PS (unless the load is extremely close to 1).

2. While the above result does not hold at all loads, we show that no matter what the load, at least 99% of the jobs have a lower expected response time under SRPT than under PS, for HT job size distributions (Section 5.2, Corollary 3). In fact, these 99% of the jobs do significantly better. We show that these jobs have an average slowdown of at most 4, at any load (Section 5.2, Theorem 7), whereas their performance could be arbitrarily bad under PS as the load approaches 1. Similar, but weaker results are shown for general distributions (Section 5.2, Theorem 4 and 5).

3. While the previous result is concerned only with 99% of the jobs, we are able to prove a general upper bound on how much worse any job could fare under SRPT as opposed to PS for general distributions. (Section 5.2, Theorem 6). Our bounds show that jobs never do too much worse under SRPT than under PS. For example, our results show that for all job size distributions, the expected response time under SRPT for any job is never more than 3 times worse than under PS, when the load is 0.8, and never more than 5.5 times when the load is 0.9. In fact, if the load is less than half, then for every job size distribution, each job has a lower expected response time (hence expected slowdown) under SRPT than under PS (Section 5.2, Theorem 4).

4. The above results show an upper bound on how much worse a job could fare under SRPT as opposed to PS for general job size distributions. We likewise prove lower bounds on the performance of SRPT as compared with PS for general job size distributions. (Section 5.2, Theorem 8).

In the previous results we required assuming an M/G/1 system, because having arbitrary arrivals makes the adversary so powerful that it is difficult to come up with good bounds on the worst case starvation in that case. However, real arrivals do not arrive according to a Poisson Process [12].

Therefore, we also test our results in a trace-driven simulation of a single server fed by a web trace. The trace includes arrival times of HTTP requests at a web server, and the service requirement of each request. We make the following observations:

1. We compare the mean performance (mean response time and mean slowdown) under SRPT with that under PS. We find that the difference between PS and SRPT is surprisingly quite similar to that predicted by our previous results for the M/G/1 model.
2. We find "unfairness to big jobs" to be non-existent. In particular, even the biggest job performs better under SRPT as compared with PS. We find that the mean slowdown as a function of job size is very similar in the trace-driven simulation as compared with our analytical predictions.

Throughout this paper, for the sake of clarity, we compare SRPT with PS scheduling only. The reason for this is that PS has the properties that it is (1) "Ultimately" fair (equal slowdown for all jobs), (2) Insensitive to the variance of the job size distribution, which implies good performance, and (3) Ubiquitous. Some other alternative policies which we might consider are first-come-first-serve (FCFS), and non-preemptive last-come-first-serve (LCFS). These two policies initially appear to be fair in that they don't favor small or large jobs, however the policies are actually not fair since small jobs have much higher slowdown than large jobs under these policies. Furthermore, these policies have poor mean performance because they are sensitive to the second moment of the job size distribution.

2 Previous work

2.1 Mean results

It has long been known that SRPT has the lowest mean response time of any scheduling policy [20],[24]. This result holds for any arrival sequence of jobs with arbitrary sizes. Rajaraman et al. showed further that the mean slowdown under SRPT is at most twice the optimal mean slowdown for any sequence of job arrivals [4].

Schrage and Miller first derived the expressions for the response times in an M/G/1 system using SRPT [21]. This was further generalized by Pechinkin *et al.* to disciplines where the remaining times are divided into intervals. The jobs with remaining times in the smaller interval are served first but those within the same interval are served in first-come-first server order [15]. The steady-state appearance of the M/G/1/SRPT queue was obtained by Schassberger [19].

Though the above formulas have been known for a long time, they are difficult to evaluate numerically, due to their complex form (many nested integrals). Hence, the comparison of SRPT to other policies was long neglected. More recently, SRPT has been compared with other policies by plotting the mean response times for specific job size distributions under specific loads [18, 16, 22, 21, 5]. A 7 year long study at University of Aachen under Schreiber [16, 22] involved extensive evaluation of SRPT for various job size distributions and loads. The survey paper by Schreiber [22] summarizes the results. They show that SRPT has significant mean response time improvements compared to other policies like FCFS, LFCS and PS. The improvements are observed to be especially significant for workloads with a large coefficient of variation [22, 18].

The above results comparing various scheduling policies are all plots for specific job size distributions and loads. Hence it is not clear whether the conclusions based on these plots hold for more general job size distributions and loads.

2.2 Unfairness results

With respect to starvation, it has often been cited that SRPT may lead to starvation of large jobs [1, 25, 26, 23]. Usually, examples of adversarial arrival sequences where a particular job starves are given to justify this. However, such worst case examples do not reflect the behavior of SRPT in the average case.

The term "starvation" is also used by people to indicate the *unfairness* of SRPT's treatment of long jobs. It is believed that since SRPT favors small jobs, long jobs should have a worse average performance under SRPT than under other policies. Perhaps this misconception that large jobs suffer unfairly under SRPT is due to the famous Kleinrock Conservation Law [9], [10, Page 197]. This law implies that if under some scheduling policy the response time of small jobs is reduced, then the response times for the large jobs would have to increase considerably. However, a careful examination of the proof reveals that the conservation law does not apply to policies which make use of size, for example SRPT.

Not much has been done to evaluate the problem of unfairness analytically. Recently, Bender et al. consider the metric *max slowdown* of a job, as indication of unfairness [1]. They show with an example that SRPT can have an arbitrarily large *max slowdown*. However, *max slowdown* is not an appropriate metric to measure unfairness. A large job may have an exceptionally long response time in some case, but it might do well most of the time. A more relevant metric which we use in our paper is the *max mean slowdown*.

There has also been work in the area of proposing new SRPT-like policies [2, 14] which try to reduce the problem of unfairness, while still favoring the short jobs to minimize the mean response time. These usually take the waiting time of a job so far into account, along with its remaining size in determining its priority. These policies are usually analytically intractable and have been evaluated by simulation only. However simulations show that they are promising.

3 Background on heavy tailed job size distributions

Many application environments show a mixture of job sizes spanning many orders of magnitude. Much previous work has used the *exponential* distribution to capture this variability, However, recent measurements indicate that for many applications the exponential distribution is a poor model and that a *heavy-tailed* distribution is more accurate. In general a heavy-tailed distribution is one for which

$$Pr\{X > x\} \approx x^{-\alpha},$$

where $0 < \alpha < 2$. The simplest heavy-tailed distribution is the *Pareto* distribution, with probability mass function

$$f(x) = \alpha k^{\alpha} x^{-\alpha-1}, \;\; \alpha, k > 0, \;\; x \geq k,$$

and cumulative distribution function

$$F(x) = Pr\{X \leq x\} = 1 - (k/x)^{\alpha}.$$

The *key property* of a heavy-tailed distribution is that a tiny fraction ($< 1\%$) of the very longest jobs comprise over half of the total load. We will refer to this as the *heavy-tailed property* throughout this paper.

Heavy-tailed distributions with $\alpha \approx 1$ appear to fit many recent measurements of computing systems [11, 7, 3, 8, 17]. Observe that lower α-values indicate greater variability in the job size distribution and stronger heavy-tailed properties. In fact $\alpha \leq 2$ indicates infinite variance. Thus, the above measurements indicate very high variability in job service requirements and a very heavy tail.

In practice, there is some upper bound on the maximum job size. Throughout this paper, we therefore model job sizes as being generated i.i.d from a distribution that is heavy-tailed, but has an upper bound. This truncated distribution is referred to as the *Bounded-Pareto* distribution [6]. It is characterized by three parameters: α, the exponent of the power law; k, the shortest possible job; and p, the largest possible job, The probability density function for the Bounded Pareto $B(k, p, \alpha)$ is defined as:

$$f(x) = \frac{\alpha k^{\alpha}}{1 - (k/p)^{\alpha}} x^{-\alpha-1} \quad k \leq x \leq p.$$

In this paper, $B(\alpha)$ will denote the distribution $B(k, p, \alpha)$ obtained by keeping the mean fixed (at 3000) and the maximum value fixed (at $p = 10^{10}$), which correspond to typical values taken from [3].

4 Mean analysis of M/G/1/SRPT

This section presents results on the mean response time and slowdown of M/G/1/SRPT. Throughout this section and the next we assume that the system is a single M/G/1 queue with arrival rate λ. We will assume that the job size distribution is continuous with probability density function $f(t)$. The cumulative job size distribution will be denoted by $F(t)$. We will denote $1 - F(t)$ by $\overline{F}(t)$. X will refer to the service time of a job. The load (utilization), ρ, of the server is

$$\rho = \lambda E[X].$$

The load made up by the jobs of size less than or equal to x, $\rho(x)$, is

$$\rho(x) = \lambda \int_0^x t f(t) dt.$$

The second moment of job sizes less than or equal to x, $m_2(x)$, is

$$m_2(x) = \int_0^x t^2 f(t) dt.$$

The expected response time for a job of size x under SRPT, $E[T(x)]_{SRPT}$, can be decomposed into the expected waiting time of the job, $E[W(x)]_{SRPT}$, and the expected residence time of the job $E[R(x)]_{SRPT}$, where $E[W(x)]$ is the expected time for a job of size x from when it first arrives to when it receives service for the first time, and $E[R(x)]_{SRPT}$ is the expected residence time (the time it takes for a job of size x to complete once it begins execution). The formulas for these expressions are given by [21]

$$E[T(x)]_{SRPT} = E[W(x)]_{SRPT} + E[R(x)]_{SRPT} \tag{1}$$

$$E[W(x)]_{SRPT} = \frac{\lambda(m_2(x) + x^2(1 - F(x)))}{2(1 - \rho(x))^2} \tag{2}$$

$$E[R(x)]_{SRPT} = \int_0^x \frac{dt}{1 - \rho(t)} \tag{3}$$

For PS the expected response time for a job of size x, $E[T(x)]_{PS}$, is given by [27]

$$E[T(x)]_{PS} = \frac{x}{1 - \rho} \tag{4}$$

For any policy, if $E[T(x)]$ is the expected response time for a job of size x, then the expected *slowdown*, $E[S(x)]$, is given by

$$E[S(x)] = \frac{T(x)}{x}$$

The mean response time and mean slowdown are given by $E[T] = \int_0^\infty E[T(x)]f(x)dx$ and $E[S] = \int_0^\infty E[S(x)]f(x)dx$ respectively.

Observe that for a given load ρ, all jobs have the same slowdown under PS, since, $E[S(x)]_{PS} = \frac{1}{1-\rho}$ for any x. Thus PS is ultimately "fair".

We now show that the mean performance advantages of SRPT over PS are significant. In Figure 1, we plot the mean response time of SRPT versus PS as a function of load[1]. In this figure we assume a heavy tailed (HT) workload, as is consistent with empirical measurements of real workloads (see Section 3). Specifically we consider a Bounded Pareto job size distribution B($\alpha = 1.1$) (see Section 3). Figure 1 shows the mean response time under PS versus SRPT. Observe that the the difference between SRPT and PS increases as the load is increased. At a load of 0.8, the mean response time under PS is 3 times that under SRPT and 6 times at load of 0.95. We investigate a variety of different job size distributions (not shown here) and find almost identical behavior. Clearly since SRPT

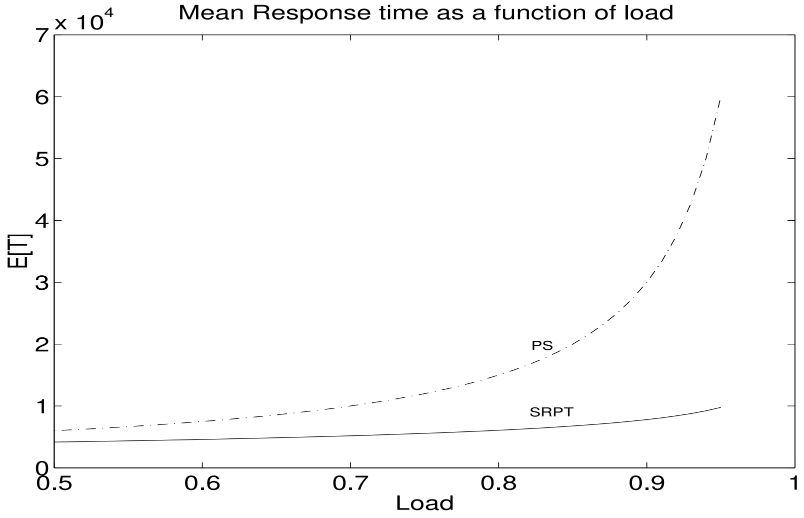

Figure 1: *Response Time for Bounded Pareto distribution, B(1.1).*

is optimal with respect to mean response time, the mean response time under SRPT is lower than that under PS for any job size distribution. We will prove that for any job size distribution, the mean response time under SRPT is less than half that under PS, as the load approaches 1 (See Corollary 1 below).

We now look at slowdown under SRPT versus PS. Figure 2 shows the mean slowdown for B($\alpha = 1.1$) as a function of load. The improvement is much more dramatic. Even

[1] Due to the highly nested nature of the $E[T]_{SRPT}$ formula, these plots were only possible by using a symbolic math package such as Mathematica^{TM}.

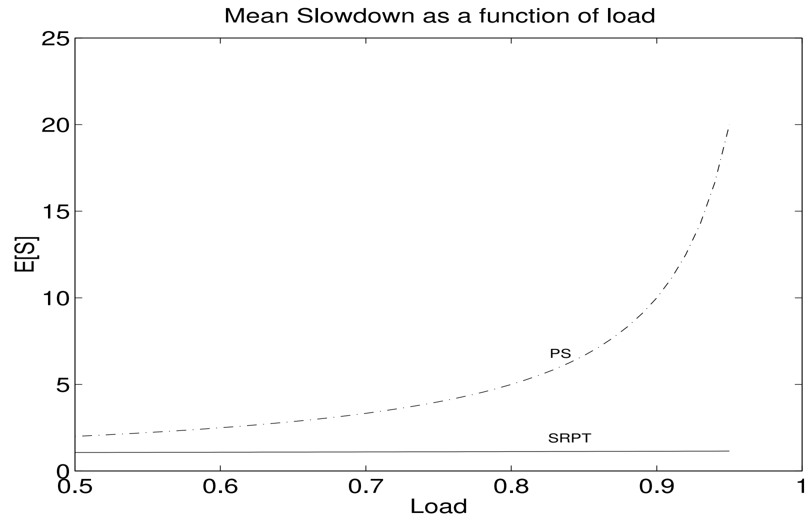

Figure 2: *Slowdown for Bounded Pareto distribution B(1.1).*

at a load of 0.95, where the mean slowdown under PS is 20, the mean slowdown under SRPT is close to 1. In fact, we investigated other job size distributions as in the case of mean response time. We find that slowdown improvement is more sensitive to job size distribution than the mean response time improvement. However, the plots for mean slowdown look similar to Figure 2 for a wide range of heavy-tailed distributions (not shown here). We show in Theorem 7, Section 5.2 that the extreme improvement witnessed in Figure 2, is a consequence of the heavy tailed property mentioned earlier.

For general job size distributions, we prove that SRPT has better mean slowdown than PS at any load. Furthermore, the mean slowdown under SRPT is less than half that under PS as load approaches 1.

Theorem 1 *Given a load ρ, then for any distribution of job sizes, then,*

$$E[T]_{SRPT} \leq h(\rho) E[T]_{PS}$$
$$E[S]_{SRPT} \leq h(\rho) E[S]_{PS}$$

where

$$h(\rho) = \frac{\rho}{2} - \frac{(1-\rho)\log(1-\rho)}{\rho}$$

In particular, for any load ρ, $E[S]_{SRPT} \leq E[S]_{PS}$.

The proof of Theorem 1 will be given in Section 5.2, since it requires analysis not yet developed.

Observing that $h(\rho) \to \frac{1}{2}$, as $\rho \to 1$, we get,

Corollary 1 *For any job size distribution, as the load approaches 1,* $\rho \to 1$, $E[T]_{SRPT} \leq \frac{1}{2}E[T]_{PS}$.

Corollary 2 *For any job size distribution, as the load approaches 1,* $\rho \to 1$, $E[S]_{SRPT} \leq \frac{1}{2}E[S]_{PS}$.

It is easy to see that the factor of two improvement in Corollaries 1 and 2 is in fact tight, given the assumption of general distributions. To see this, observe that for the constant job size distribution, SRPT is identical to FCFS. As the load approaches 1, it can be seen that $E[T]_{SRPT} = E[T]_{FCFS} \approx \frac{1}{2}E[T]_{PS}$.

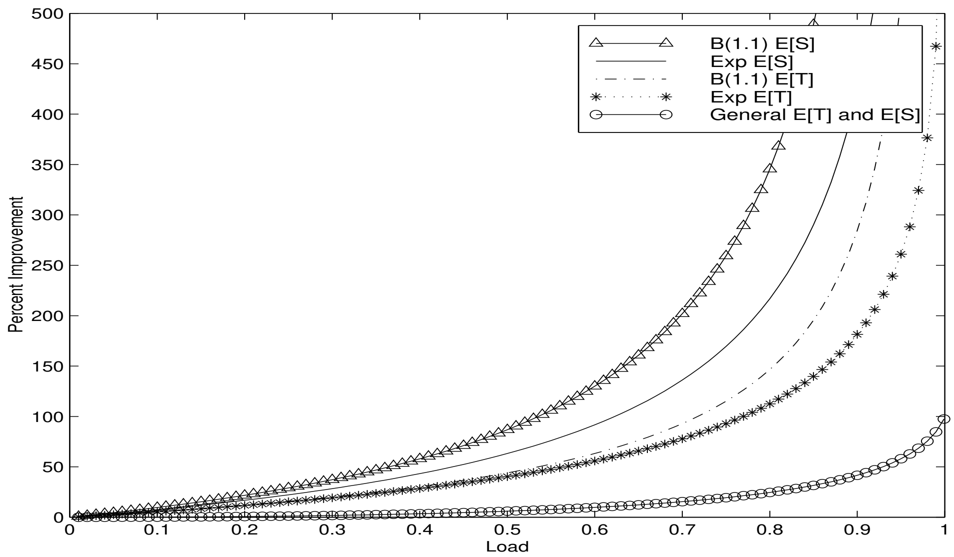

Figure 3: *This figure summarizes our results on the improvement of SRPT over PS with respect to mean slowdown and mean response time for various distributions. The lowest curve is a lower bound for general job size distributions, which is a graphical representation of Theorem 1.*

Figure 3 depicts pictorially all of the results mentioned in this section. This figure shows the percentage improvement of SRPT over PS. Begin by observing the bottom-most curve (consisting of circles). This curve gives a lower bound on the percentage improvement of SRPT over PS for general job size distributions. The curve shows that when load is very high, the percentage improvement of SRPT over PS is at least 100%. This improvement holds for both mean response time and mean slowdown. However, for most job size distributions, the improvement is much greater than the lower bound. Consider next the curve second from the bottom consisting of stars. This curve shows the improvement in mean response time of SRPT over PS for an exponential job size distribution. For load 0.8, the improvement is already a 100%, and for load 0.9, the improvement is close to

200%. The dashed curve (third from bottom) shows the improvement in mean response time of SRPT over PS for the $B(1.1)$ distribution. At a load of 0.8, the improvement is about 150%, and at a load of 0.9 the improvement is over 300%. The improvement in mean slowdown is even more exaggerated as shown by the top two curves.

At this point it is tempting to assume that the huge mean slowdown improvements of SRPT as seen in Figure 2 are due to disproportionately helping the many small jobs and sacrificing the fewer big jobs. In the next section we will show that this is in fact not the case.

5 Unfairness Analysis

It is commonly believed that it is not possible to improve the performance of a job without hurting the performance of some other job. In particular, if one switches to a new scheduling policy which improves the mean performance of some jobs, then it must do so at the cost of hurting some other jobs. We first dispel this myth.

Section 5.1 motivates our main results. We first show with an example that there exist job size distributions such that every job can do better under SRPT than under PS. We also give intuition as to why this might be true. We then show the main analytical results on unfairness in Section 5.2.

5.1 All jobs can do better

We saw in Figures 1 and 2 that for $B(\alpha = 1.1)$ at load 0.9, mean response time and mean slowdown under SRPT is substantially less than under PS. We now ask whether this mean improvement comes at the cost of severely penalizing large jobs. Figure 4 below shows the slowdown as function of job size, at load 0.9. Specifically we show the expected slowdown for a job in each percentile of the job size distribution (where 100 percentile indicates the very largest job). Observe that, each job has an expected slowdown of 10 under PS, but surprisingly every single job has a smaller slowdown (hence response time) under SRPT. Even the largest job has a slowdown of 9.54 under SRPT. Thus, even the largest job spends a lesser amount of time under SRPT than under PS.

We state this important observation as a theorem.

Theorem 2 *There exist job size distributions such that every job does better under SRPT than under PS.*

Proof: The proof follows from Figure 4. ∎

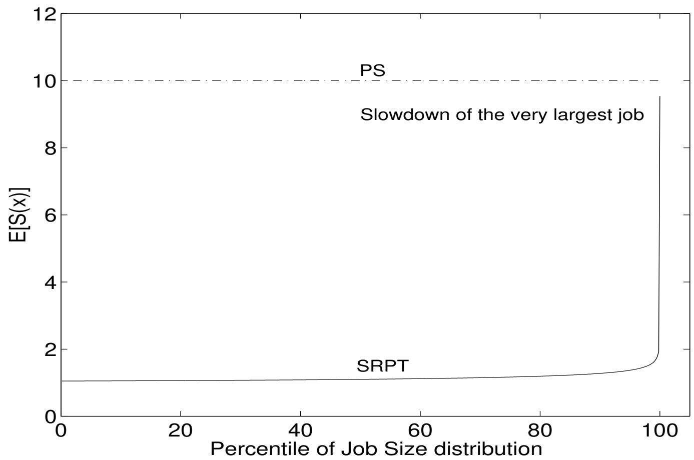

Figure 4: *Expected slowdown as a function of job size for $B(1.1)$ distribution, at load $\rho = 0.9$. Even the largest job prefers SRPT to PS*

Surprisingly, this fact is true for many job size distributions, in particular heavy-tailed ones. This is important because HT distributions are representative of today's empirical workloads (Section 3).

For example, if the job size distribution is B(1.1) every job does better under SRPT as long as the load is below 0.96. In fact for B(1.5) every job does better for loads up to 0.999. In general we observe the following:

Observation: For many heavy-tailed job size distributions $E[T(x)]_{SRPT} \leq E[T(x)]_{PS}$, for all x, unless ρ is extremely close to 1.

It is clear that under SRPT slowdown should be better for small jobs, since SRPT helps the small jobs. However, it is not at all clear why this should also be the case for large jobs. Intuitively, the following explains why this should be true:

Under SRPT, a job is affected only by the other jobs in the system which have a smaller remaining size than itself. Once a job begins execution, its remaining size diminishes with time. Thus the load seen by the job gets smaller as the job is worked upon. In contrast under PS, throughout its execution, a job is affected by all the other jobs present in the system. Thus the load that the job sees does not change with time. Thus it makes sense that, the expected *residence time* of a job under SRPT is smaller than its expected response time under PS. This difference is especially significant for heavy-tailed distributions where the large jobs make up most of the load.

The above argument has shown intuitively why the *residence time* under SRPT is smaller than the response time under PS. To argue about *response time* under SRPT, however, we also need to take into account the waiting time under SRPT. Although the waiting time under SRPT may be large for big jobs, it turns out that the waiting time is often not great enough to overturn the above conclusion. In the next section, we will provide formal proofs which take all these details into account.

5.2 Unfairness analysis for general job size distributions

Theorem 2 shows the existence of job size distributions for which every job prefers SRPT to PS under most loads. We now extend this result along many directions. First, we show a similar but weaker result that holds for all job size distributions.

Theorem 3 *For any job size distribution, if the load is not more than half then every job under SRPT has a lower expected response time than under PS.*

The proof of this theorem will follow from Theorem 4. The condition that the load is lower than half in Theorem 3 is rather restrictive. However, if we relax the restriction that *every* job performs better, then we get the following stronger result which holds at all loads.

Theorem 4 *For any job size distribution f and any load $\rho < 1$,*

$$E[T(x)]_{SRPT} \leq E[T(x)]_{PS}$$

for every job of size x such that $\rho(x) \leq \frac{1}{2}$ (i.e. jobs of size $\leq x$ comprise less than half the load).

Theorem 4 implies Theorem 3, since $\rho \leq \frac{1}{2}$ directly implies $\rho(x) \leq \frac{1}{2}$ for all x. The proof of Theorem 4 will follow from a more general Theorem 5 below.

Theorem 4 becomes especially useful if we relate the load percentiles and the job percentiles (i.e. $\rho(x)$ and $F(x)$). The *heavy-tailed property* stated in Section 3 implies that less than 1% of the very largest jobs make up more than half the load. Thus Theorem 4 implies that at least 99% of the jobs have smaller response times under SRPT than under PS no matter what the load. Thus we have Corollary 3.

Corollary 3 *For distributions with the heavy-tailed property, at least 99% of the jobs have a lower response time under SRPT than under PS at any load.*

Observe however that even for the "light-tailed" exponential ($c = 1$) and hyper-exponential distributions with $c \approx 3$ (where c is the coefficient of variation), Theorem 4 implies that more than 81% and 95% respectively of the jobs do better at any load.

We now state and prove a generalization of Theorem 4 which likewise holds for any job size distribution and load $\rho < 1$.

Theorem 5 *For any job size distribution f and load ρ,*

$$E[T(x)]_{SRPT} \leq E[T(x)]_{PS}$$

for all jobs of size x such that

$$2(1-\rho(x))^2 \geq (1-\rho) \tag{5}$$

Proof: $E[T(x)]_{PS}$ can be written as $\int_0^x \frac{dt}{1-\rho}$. And, $E[R(x)]_{SRPT} = \int_0^x \frac{dt}{1-\rho(t)}$.
Since $\rho \geq \rho(t)$ for any t, the expected residence time for any job under SRPT is smaller than the expected response time under PS. We will bound this difference and obtain conditions under which the difference more than compensates for the waiting time under SRPT.

$$\begin{aligned}
E[T(x)]_{PS} - E[R(x)]_{SRPT} &= \int_0^x \frac{dt}{1-\rho} - \int_0^x \frac{dt}{1-\rho(t)} \\
&= \int_0^x \frac{(\rho-\rho(t))dt}{(1-\rho(t))(1-\rho)} \\
&\geq \int_0^x \frac{(\rho-\rho(t))dt}{1-\rho} \qquad [\text{Since} \quad (1-\rho(t)) \leq 1] \\
&= \frac{x(\rho-\rho(x)) + \lambda m_2(x)}{1-\rho} [\text{Since} \quad \rho'(t) = \lambda t f(t)] \qquad (6) \\
&\geq \frac{\lambda x^2(1-F(x)) + \lambda m_2(x)}{1-\rho} \qquad (7)
\end{aligned}$$

Line (7) follows from Line (6) since:

$$\begin{aligned}
x(\rho-\rho(x)) &= \lambda x \int_x^\infty t f(t) dt \\
&\geq \lambda x^2 \int_x^\infty f(t) dt \\
&= \lambda x^2 (1-F(x))
\end{aligned}$$

Comparing the expression for $E[W(x)]_{SRPT}$ in equation (2) with (7) it is clear that,

$$E[T(x)]_{PS} - E[R(x)]_{SRPT} \geq E[W(x)]_{SRPT}$$

whenever the condition (5) is met.
Thus, $E[T(x)]_{PS} \geq E[T(x)]_{SRPT}$ if $2(1-\rho(x))^2 \geq (1-\rho)$. ∎

Proof: (**Theorem 4**) If $\rho(x) \leq \frac{1}{2}$, then $2(1-\rho(x)) \geq 1$.
Observe that for all x, $(1-\rho(x)) \geq (1-\rho)$.
Multiplying both the inequalities we get, $2(1-\rho(x))^2 \geq (1-\rho)$ and the result follows from Theorem 5. ∎

Theorems 3 and 4 show that for all job size distributions,

1. If the load is below half, then all jobs have a lower expected response time under SRPT as compared to PS.

2. Even if the load is above half, a majority of the jobs have better expected response times under SRPT.

But what about the small fraction of jobs which have a higher slowdown under SRPT than under PS, how bad can their starvation be? We will show that for a fixed load, no job can do arbitrarily badly on the average. Theorem 6 establishes an bound on the ratio of the expected response time of a job of size x under SRPT to that under PS.

Theorem 6 *For all job size distributions f, for all loads ρ, for all x,*

$$E[T(x)]_{SRPT} \leq \frac{1-\rho}{1-\rho(x)}\left[\frac{\rho}{2(1-\rho(x))}+1\right]\cdot E[T(x)]_{PS} \tag{8}$$

In particular,

$$E[T(x)]_{SRPT} \leq \left[\frac{\rho}{2(1-\rho)}+1\right]\cdot E[T(x)]_{PS} \tag{9}$$

Before we can prove this theorem, we need one observation:

Lemma 6.1

$$\int_0^x tf(t)dt + x\cdot\overline{F}(x) \leq E[X]$$

Proof:

$$\begin{aligned} E[X] &= \int_0^x tf(t)dt + \int_x^\infty tf(t)dt \\ &\geq \int_0^x tf(t)dt + x\overline{F}(x) \end{aligned}$$

∎

Proof: **(Theorem 6)**

$$\begin{aligned} E[T(x)]_{SRPT} &= \frac{\lambda\int_0^x t^2 f(t)dt + \lambda x^2\overline{F}(x)}{2(1-\rho(x))^2} + \int_0^x \frac{dt}{1-\rho(t)} \\ &\leq \frac{\lambda\int_0^x t^2 f(t)dt + \lambda x^2\overline{F}(x)}{2(1-\rho(x))^2} + \frac{x}{1-\rho(x)} \quad [\text{Since } (1-\rho(x)) \leq (1-\rho(t)), \text{for } t \leq x] \\ &\leq \frac{\lambda x\int_0^x tf(t)dt + \lambda x^2\overline{F}(x)}{2(1-\rho(x))^2} + \frac{x}{1-\rho(x)} \end{aligned}$$

$$
\begin{aligned}
&= \frac{x}{1-\rho(x)}\left[\frac{\lambda\int_0^x tf(t)dt + \lambda x\overline{F}(x)}{2(1-\rho(x))}+1\right] \\
&\leq \frac{x}{1-\rho(x)}\left[\frac{\lambda E[X]}{2(1-\rho(x))}+1\right] \qquad \text{[By Lemma 6.1]} \\
&= \frac{x}{1-\rho}\frac{1-\rho}{1-\rho(x)}\left[\frac{\rho}{2(1-\rho(x))}+1\right] \\
&= E[T(x)]_{PS}\frac{1-\rho}{1-\rho(x)}\left[\frac{\rho}{2(1-\rho(x))}+1\right]
\end{aligned}
$$

Thus equation (8) follows.

We observe that the expression $\frac{1-\rho}{1-\rho(x)}\left[\frac{\rho}{2(1-\rho(x))}+1\right]$ is maximized when x is the largest job (i.e. $\rho(x)=\rho$), in which case we get $E[T(x)]_{SRPT} \leq (\frac{\rho}{2(1-\rho)}+1)E[T(x)]_{PS}$. ■

Theorem 6 shows that for a given a load ρ, the expected response time for a job cannot be arbitrarily worse than that under PS. For example, if $\rho = 0.8$, no job has an expected response time more than 3 times that under PS, and never more than 5.5 times that under PS when the load is 0.9. In reality however, the factor is much better, since our analysis is not tight and it holds for all job size distributions. Stronger results can be obtained for specific job size distributions.

Though the bound in (9), $\frac{\rho}{2(1-\rho)}+1$, is somewhat weak, the stronger bound as a function of x, shown in equation (8), is quite useful. For example, at $\rho = 0.9$, equation (8) implies that $E[T(x)]_{SRPT} < \frac{1}{3}E[T(x)]_{PS}$, for all jobs of size x such that $\rho(x) \leq \frac{\rho}{2}$. Coupled with the heavy-tailed property, the consequences are striking. The heavy-tailed property implies that at at least 99% of the jobs perform at least three times better under SRPT as compared with PS. So, at most only 1% of the remaining jobs can do worse under SRPT. Applying equation (8) again we see that these 1% jobs can be at most 5.5 times worse under SRPT than under PS. Thus the improvements in the mean slowdown would be close to a factor of at least 3, since it is dominated by the improvements for the small jobs. At higher loads, the results are even more striking. At $\rho = 0.99$, the expected slowdown for jobs such that $\rho(x) \leq \frac{\rho}{2}$ (99% of the jobs for heavy-tailed distributions) under SRPT is at least about 25 times better than that under PS for any distribution. So, the mean slowdown will be significantly better for heavy-tailed distributions at this load. Theorem 7 below makes this observation precise.

Theorem 7 *For any job size distribution and any load $\rho < 1$,*

$$E[S(x)]_{SRPT} \leq 2 + 2\rho$$

for all jobs of size x such that $\rho(x) \leq \frac{1}{2}$. Hence, for HT job size distributions at least 99% of the jobs have an expected slowdown of at most 4, irrespective of the system load.

Proof: Follows directly from equation (8), Theorem 6. ∎

Theorem 7 and Theorem 6 explain why Figure 2 is typical of all heavy-tailed distributions.

Figure 5 summarizes our results on the performance improvement under SRPT for the majority of the jobs (i.e. for jobs of size x such that $\rho(x) \leq \frac{\rho}{2}$).

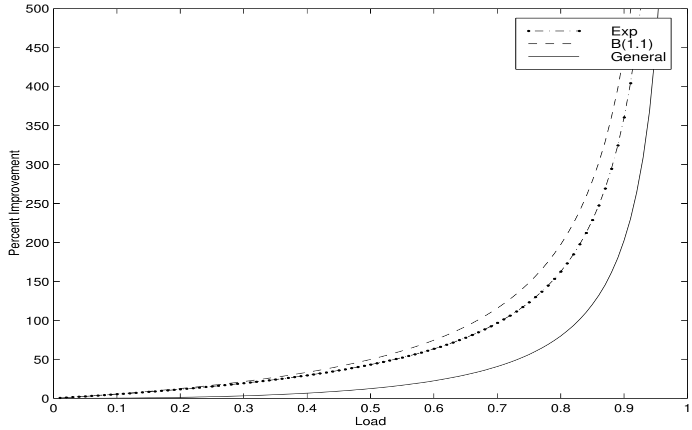

Figure 5: *This figure shows the percentage improvement in the expected slowdown experienced by the majority of jobs, (i.e., jobs of size x such that $\rho(x) \leq \frac{\rho}{2}$). The solid line indicates a lower bound for all distributions.*

For general job size distributions, we need not have the huge mean slowdown improvements we observe for HT distributions. We will now use the previous theorem to prove Theorem 1.

Proof: (**Theorem 1**) Let $g(\rho(x))$ denote the improvement factor in Theorem 6.

$$g(\rho(x)) = \frac{1-\rho}{1-\rho(x)}\left[\frac{\rho}{2(1-\rho(x))} + 1\right]$$

So, $E[T(x)]_{SRPT} \leq g(\rho(x))E[T(x)]_{PS}$.

$$\begin{aligned}
E[T]_{SRPT} &= \int_0^\infty E[T(x)]_{SRPT}\, f(x)dx \\
&\leq \int_0^\infty g(\rho(x))E[T(x)]_{PS}\, f(x)dx \\
&= \int_0^\rho \frac{1}{\lambda}\frac{g(\rho(x))}{1-\rho}d(\rho(x)) \qquad \text{Since}, d\rho(x) = \lambda x f(x)dx \\
&= \frac{1}{\rho}E[T]_{PS}\int_0^\rho g(\rho(x))d(\rho(x))
\end{aligned}$$

Integrating g by parts, we get

$$\int_0^\rho g(\rho(x))d\rho(x) = \frac{\rho^2}{2} - (1-\rho)\log(1-\rho) \tag{10}$$

Thus, it follows that $E[T]_{SRPT} \leq h(\rho)E[T]_{PS}$.

For slowdown, we similarly obtain,

$$\begin{aligned}
E[S]_{SRPT} &= \int_0^\infty E[S(x)]_{SRPT}\ f(x)dx \\
&\leq \int_0^\infty E[S(x)]_{PS}\ g(\rho(x))f(x)dx \text{ [Dividing both sides of equation (8) by } x\text{]} \\
&= \int_0^\rho \frac{1}{1-\rho} g(\rho(x))\frac{1}{\lambda x} d(\rho(x))
\end{aligned}$$

Observe that $g(\rho(x))$ is increasing in $\rho(x)$ and $\frac{1}{x}$ is decreasing in $\rho(x)$. We now apply the Chebyshev Integral Inequality [13], which states that if $u(y), v(y)$ are non-negative functions which are non-decreasing and non-increasing respectively, then

$$(b-a)\int_a^b u(y)v(y)dy \leq \int_a^b u(y)dy \int_a^b v(y)dy \tag{11}$$

Setting $y = \rho(x), u(\rho(x)) = g(\rho(x)), v(\rho(x)) = \frac{1}{x}, a = 0$ and $b = \rho$ we get,

$$\begin{aligned}
E[S]_{SRPT} &\leq \frac{1}{1-\rho}\frac{1}{\rho}\int_0^\rho g(\rho(x))d\rho(x) \int_0^\rho \frac{d\rho(x)}{\lambda x} \\
&= \frac{1}{\rho}\int_0^\rho \frac{1}{1-\rho(x)}\left[\frac{\rho}{2(1-\rho(x))} + 1\right] d\rho(x) \int_0^\infty f(x)dx \\
&= \frac{h(\rho)}{1-\rho}
\end{aligned}$$

Thus, $E[S]_{SRPT} \leq h(\rho)E[S]_{PS}$.

We now show that $h(\rho) \leq 1$ for all $\rho \leq 1$.

$$\frac{d(h(\rho)}{d\rho} = \frac{1}{2} + \frac{1}{\rho} + \frac{\log(1-\rho)}{\rho^2}$$

Using the identity,

$$\log(1-x) = -x - \frac{x^2}{2} - \frac{x^3}{3} - \ldots$$

we get that

$$\frac{d(h(\rho))}{d\rho} \leq 0, \quad \forall \quad 0 \leq \rho \leq 1$$

Thus $h(\rho)$ is decreasing in ρ. Observing that $h(0) = 1$, it follows that $h(\rho) \leq 1$, for all ρ.

Thus $E[S]_{SRPT} \leq E[S]_{PS}$. ■

Figure 3 plots the worst case improvement factor of SRPT over PS with respect to mean slowdown, as a function of ρ.

So far, we have shown two types of results with respect to starvation. We either show that *all* jobs do well for *most* loads. Or, *most* jobs do well for *all* loads. A natural question to ask at this point is, whether there are job size distributions for which all jobs do well at all loads.

We show that this is not the case. When load approaches 1, the largest job will perform worse under SRPT for any job size distribution (which has a well-defined largest job).

Theorem 8 *For every bounded distribution,* $\exists \rho < 1$ *such that*

$$E[T(l)]_{SRPT} > E[T(l)]_{PS}$$

where l *is the size of the largest job.*

Proof: We will lower bound $E[T(l)]_{SRPT}$ and show that there exists a $\rho < 1$ such that $E[T(l)]_{SRPT} > \frac{l}{1-\rho} = E[T(l)]_{PS}$.

The waiting time of the largest job under SRPT is simply $\frac{m_2(l)}{2(1-\rho)^2}$. To lower bound the residence time under SRPT we use the Chebyshev Integral Inequality, (11), with $y = x$,$u(x) = \frac{1}{1-\rho(x)}$, $v(x) = 1 - \rho(x)$, $a = 0$ and $b = l$. Note that u and v satisfy the conditions in (11) since $\rho(x)$ is non-decreasing in x. Thus we get,

$$1 \leq \frac{\int_0^l (1-\rho(x))dx \int_0^l \frac{dx}{1-\rho(x)}}{l^2}$$

Equivalently,

$$\text{Residence Time(l)} = \int_0^l \frac{dx}{1-\rho(x)} \geq \frac{l^2}{l - l\rho + \lambda m_2(l)}$$

So,

$$\begin{aligned} E[T(l)]_{SRPT} - E[T(l)]_{PS} &\geq \frac{\lambda m_2(l)}{2(1-\rho)^2} + \frac{l^2}{l - \rho l + \lambda m_2(l)} - \frac{l}{1-\rho} \\ &= \frac{\lambda m_2(l)}{1-\rho}\left(\frac{1}{2(1-\rho)} - \frac{1}{(1-\rho(1-d))}\right) \text{ [where } d = \frac{m_2(l)}{lE[X]}\text{]} \\ &= \frac{\lambda m_2(l)}{2(1-\rho)^2(1-\rho(1-d))}(\rho(1+d) - 1) \end{aligned}$$

Since both l and $E[X]$ are finite, $d > 0$. Thus for any $\rho < 1$, such that $\rho(1+d) > 1$, $E[T(l)]_{SRPT} - E[T(l)]_{PS} > 0$.

Hence, for any job size distribution, there is a load (less than 1) such that, the largest job has a higher expected response time under SRPT than under PS. ■

6 Trace-driven simulation comparing SRPT with PS

In Sections 4 and 5 we analyze the mean performance of SRPT (mean response time and mean slowdown) and the unfairness properties of SRPT under an M/G/1 model. In this section we investigate whether these results are also consistent with more realistic arrival processes. We consider a trace-driven simulation experiment of a single server fed by a trace of static requests to a Web server. The trace contains both the arrival times and the job sizes, where we assume that the job's size (in seconds) is proportional to the "file size" (in bytes) associated with the request. This experiment is described in Section 6.1.

In the above experiment, we ignore the overhead due to cost of preemptions. We are allowed to do this because the expected number of preemptions is provably higher under PS than under SRPT, as shown in Section 6.2.

6.1 Trace-driven simulation experiment

The data for our trace-driven simulation experiment is obtained from the Internet Traffic Archives[2]. The data consists of HTTP requests observed on the Home IP modem bank at UC Berkeley during 4 consecutive hours on Nov 17, 1996[3]. We exclude the requests which correspond to requests of size less than 128 bytes, since they indicate that the server did not process the request. This modified trace contains about 75,000 requests.

A quick examination of the trace data shows that it closely fits a heavy-tailed Pareto distribution $B(\alpha = 1.2, k = 1024, p = 4700000)$ with mean 8700 and squared coefficient of variation $C^2 = 35$.

We simulated a single processor with a single CPU. To create the desired system load, we scale the service times by the appropriate factor.

Figure 6 is produced by subdividing the 75,000 requests into 25 intervals of 3000 requests each. We then measure the mean slowdown for each interval. The system load is 0.9. Figure 6 shows that the mean slowdown under PS fluctuates between 5 and 30, with a mean at about 10. This is consistent with the M/G/1/PS formula (4), although the arrival process now is not Poisson. The mean slowdown under SRPT is about 1.5, with much less fluctuation. This is again consistent with the results predicted from the M/G/1/SRPT formula (1) (assuming the distribution to be $B(\alpha = 1.2, k = 1024, p = 4700000)$, which is

[2] http://ita.ee.lbl.gov/html/traces.html

[3] UC Berkeley Home IP Web Traces, collected by Steven D. Gribble.

a good analytic approximation to the workload obtained from the trace).

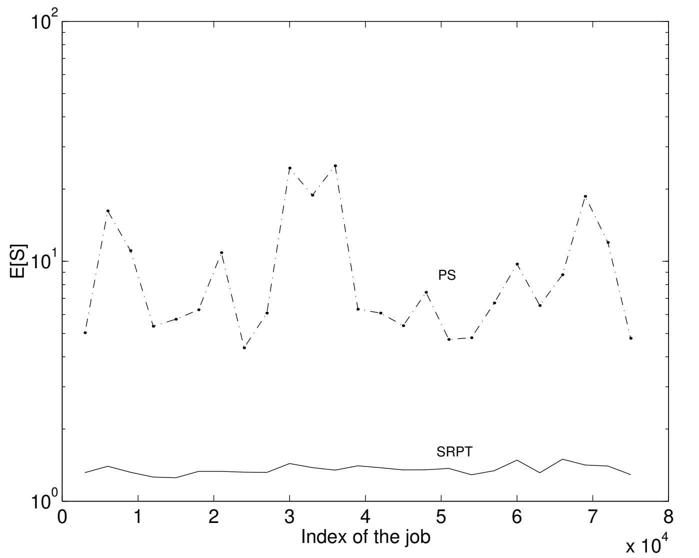

Figure 6: The figure shows simulation results for the mean slowdown during the course of 75,000 jobs under SRPT and under PS. Jobs have been grouped into 25 intervals, to show the fluctuation of mean slowdown with time.

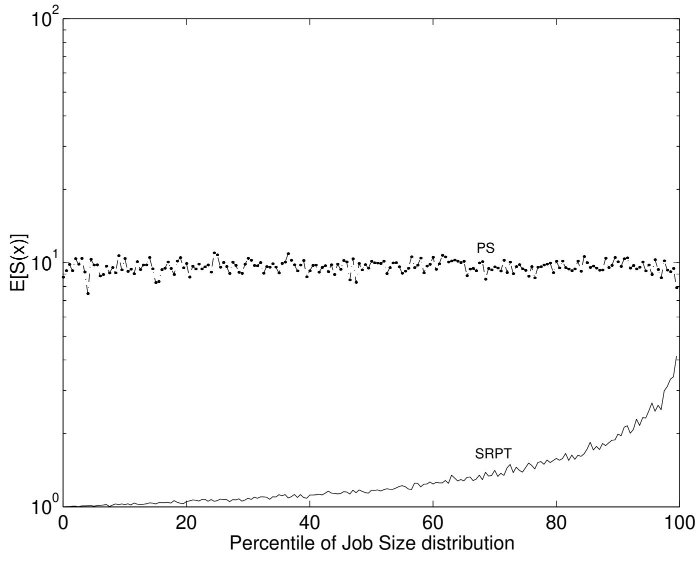

Figure 7: This figure shows simulation results of the expected slowdown as a function of job size under SRPT and under PS. Even jobs in the 100th-percentile of the job size distribution, have an expected slowdown under SRPT which is significantly below that of PS.

To investigate unfairness, we now plot the expected slowdown for each job as a function of the percentile of job size. Figure 7 shows that jobs of all sizes have a consistent mean slowdown of 10 under PS, which is in exact agreement with the result for M/G/1/PS. Under SRPT, the expected slowdown of jobs in the bottom 30 percentile is barely above 1; the expected slowdown of jobs in the 95th percentile is about 2; and the expected slowdown of jobs in the 100th percentile is about 4, which is still less than half that of PS. Again, these observed values of expected slowdown under SRPT agree very closely with the analytically predicted values for SRPT.

6.2 Preemption overhead

While implementing a scheduling policy with preemptions (like SRPT, PS, FB ...) the overhead associated with preemptions is a cause for concern in real systems. In real systems, PS is implemented as round robin where each process gets a finite time quantum. Observe that the number of preemptions depends on the size of this time quantum.

However, under SRPT, the amortized number of preemptions per job is at most two. This is true irrespective of the arrival process. To see this, observe that a job may be preempted under SRPT only when a new job arrives into the system or an existing job is completed. Since any job arrives and completes exactly once, then total number of preemptions is never more than twice the number of job arrivals.

Since most jobs require more than two time quantums of service, under any reasonable implementation of PS the average number of preemptions per job will be more than two.

7 Conclusion

The main contribution of this paper is to show that the long held belief that large jobs do worse under SRPT as compared to a fair policy like PS is not necessarily true. We prove that under moderate system load, for any job size distribution, *all* jobs prefer SRPT to PS. As the load increases, this statement is only true for job size distributions with the heavy-tailed property. However the situation is not as bad as one might think for general distributions. Even under conditions of higher load, for general distributions, we show that the majority of jobs are insensitive to the higher load. For the remaining jobs, we prove absolute bounds on how high the expected slowdown under SRPT can be as compared with PS.

While it was known that SRPT is optimal with respect to mean response time, the degree of the improvement in mean response time of SRPT over PS was not known for general job size distributions. Also, no comparison of SRPT over PS existed with respect to mean slowdown. We obtain bounds on the improvement factor of SRPT over PS under general job size distributions, as a function of load for both mean response time and mean slowdown.

Our analysis highlights the fact that the job size distribution plays a very significant role in the choice of the scheduling policy. Whereas PS is insensitive to the job size distribution, the performance of SRPT improves significantly under more heavy-tailed distributions. Observations were made earlier about the good behavior of SRPT for workloads with high variance. We show that this is really a consequence of the heavy-tailed property.

References

[1] M. Bender, S. Chakrabarti, and S. Muthukrishnan. Flow and stretch metrics for scheduling continous job streams. In *Proceedings of the 9th Annual ACM-SIAM Symposium on Discrete Algorithms*, 1998.

[2] L. Cherkasova. Scheduling strategies to improve response time for web applications. In *High-performance computing and networking: international conference and exhibition*, pages 305–314, 1998.

[3] Mark E. Crovella and Azer Bestavros. Self-similarity in World Wide Web traffic: Evidence and possible causes. In *Proceedings of the 1996 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*, pages 160–169, May 1996.

[4] J.E. Gehrke, S. Muthukrishnan, R. Rajaraman, and A. Shaheen. Scheduling to minimize average stretch online. In *40th Annual symposium on Foundation of Computer Science*, pages 433–422, 1999.

[5] Mor Harchol-Balter, M. Crovella, and S. Park. The case for srpt schduling in web servers. Technical Report MIT-LCS-TR-767, MIT Lab for Computer Science, October 1998.

[6] Mor Harchol-Balter, Mark Crovella, and Cristina Murta. On choosing a task assignment policy for a distributed server system. *IEEE Journal of Parallel and Distributed Computing*, 59:204 – 228, 1999.

[7] Mor Harchol-Balter and Allen Downey. Exploiting process lifetime distributions for dynamic load balancing. In *Proceedings of SIGMETRICS '96*, pages 13–24, 1996.

[8] Gordon Irlam. Unix file size survey - 1993. Available at `http://www.base.com/gordoni/ufs93.html`, September 1994.

[9] L. Kleinrock, R.R. Muntz, and J. Hsu. Tight bounds on average response time for time-shared computer systems. In *Proceedings of the IFIP Congress*, volume 1, pages 124–133, 1971.

[10] Leonard Kleinrock. *Queueing Systems*, volume II. Computer Applications. John Wiley & Sons, 1976.

[11] W. E. Leland and T. J. Ott. Load-balancing heuristics and process behavior. In *Proceedings of Performance and ACM Sigmetrics*, pages 54–69, 1986.

[12] W.E. Leland, M.S. Taqqu, W. Willinger, and D.V. Wilson. On the self-similar nature of Ethernet traffic (extended version). *IEEE/ACM Transactions on Networking*, 2:1–15, 1994.

[13] D.S. Mitrinovic. *Anayltic Inequalities*. Springer-Verlag, 1970.

[14] E. Modiano. Scheduling algorithms for message transmission over a satellite broadcast system. In *Proceedings of IEEE MILCOM '97*, pages 628–634, 1997.

[15] A.V. Pechinkin, A.D. Solovyev, and S.F. Yashkov. A system with servicing discipline whereby the order of remaining length is serviced first. *Tekhnicheskaya Kibernetika*, 17:51–59, 1979.

[16] R. Perera. The variance of delay time in queueing system M/G/1 with optimal strategy SRPT. *Archiv fur Elektronik und Uebertragungstechnik*, 47:110–114, 1993.

[17] David L. Peterson and David B. Adams. Fractal patterns in DASD I/O traffic. In *CMG Proceedings*, December 1996.

[18] J. Roberts and L. Massoulie. Bandwidth sharing and admission control for elastic traffic. In *ITC Specialist Seminar*, 1998.

[19] R. Schassberger. The steady-state appearance of the M/G/1 queue under the discipline of shortest remaining processing time. *Advances in Applied Probability*, 22:456–479, 1990.

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

[21] L.E. Schrage and L.W. Miller. The queue M/G/1 with the shortest processing remaining time discipline. *Operations Research*, 14:670–684, 1966.

[22] F. Schreiber. Properties and applications of the optimal queueing strategy SRPT - a survey. *Archiv fur Elektronik und Uebertragungstechnik*, 47:372–378, 1993.

[23] A. Silberschatz and P. Galvin. *Operating System Concepts, 5th Edition.* John Wiley & Sons, 1998.

[24] D.R. Smith. A new proof of the optimality of the shortest remaining processing time discipline. *Operations Research*, 26:197–199, 1976.

[25] W. Stallings. *Operating Systems, 2nd Edition.* Prentice Hall, 1995.

[26] A.S. Tanenbaum. *Modern Operating Systems.* Prentice Hall, 1992.

[27] Ronald W. Wolff. *Stochastic Modeling and the Theory of Queues.* Prentice Hall, 1989.