Bounds on a Fair Policy with Near Optimal Performance

Adam Wierman [1] Mor Harchol-Balter [2]

November 2003
CMU-CS-03-198

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Abstract

Providing fairness and providing good response times are often viewed as conflicting goals in scheduling. Scheduling policies that provide low response times, such as Shortest Remaining Processing Time (SRPT), are sometimes not fair, while fair policies like Processor Sharing (PS) provide response times far worse than SRPT. This seemingly inevitable tension between providing fairness and providing good response times was eliminated at last year's ACM Sigmetrics conference with the introduction of a new scheduling policy, Fair Sojourn Protocol (FSP), that appears to provide both [9]. The FSP policy is provably fair, as seen directly from its definition, and simulations show that FSP has a very low mean response time, close to that of SRPT in many cases [9]. Unfortunately, analyzing the mean response time of the FSP policy has proven to be difficult, and thus the queueing performance of FSP has only be assessed via simulation.

In this work, we present the first queueing analysis of FSP. This analysis yields close upper and lower bounds on the mean response time and mean slowdown of the M/GI/1/FSP queue. Our upper bound shows that the improvement of FSP over PS is substantial: for *all* job size distributions, the mean response time and mean slowdown under FSP are a fraction $(1-\frac{\rho}{2})$ of that under PS, where ρ is the system load. For distributions with decreasing failure rate the improvement is even greater. We also prove that the mean response time of SRPT and FSP are quite close. Lastly, our bounds reveal that FSP has yet another desirable property: similarly to PS, the FSP policy is largely insensitive to the variability of the job size distribution.

[1] Carnegie Mellon University, Computer Science Department. Email: acw@cs.cmu.edu.
[2] Carnegie Mellon University, Computer Science Department. Email: harchol@cs.cmu.edu.

This work was supported by NSF Career Grant CCR-0133077, NSF ITR Grant 02-168, NSF CCR-0311383, and by IBM Corporation via Pittsburgh Digital Greenhouse Grant A002081, and an NSF Graduate Research Fellowship.

Keywords: Scheduling, queueing, FSP, fair sojourn protocol, PS, processor sharing, fairness, M/GI/1, slowdown, response time.

1 Introduction

In the late 1960's Schrage, Miller, Conway, and many others wrote a series of papers proving that by giving preference to short jobs, or more specifically, jobs with short remaining service times, one could greatly improve mean response time [5, 15, 17, 21, 24]. Here response time is defined as the time from when a job enters the system until it completes. In particular the Shortest-Remaining-Processing-Time (`SRPT`) scheduling algorithm, which always runs the job with shortest remaining size, was proven to yield optimal mean response time [20].

Despite the above results, there is a hesitancy to use such policies due to a fear of unfairness. It is commonly suggested that policies that bias towards small job sizes *starve* large jobs [3, 23, 25].

Recently, there has been a large amount of work studying unfairness analytically. Research has sought to quantify the amount of unfairness seen under common scheduling policies [1, 9, 10, 11, 19], as well as to understand unfairness properties across all scheduling policies [12, 27].

The definition of unfairness most commonly used is stated in terms of slowdown. Given an M/GI/1 system with a differentiable service distribution having finite mean and finite variance, let $T(x)$ denote the steady-state response time for a job of size x, and $\rho < 1$ be the system load. That is $\rho \stackrel{\text{def}}{=} \lambda E[X]$, where λ is the arrival rate of the system and X is a random variable distributed according to the service (job size) distribution $F(x)$ having density function $f(x)$. The slowdown seen by a job of size x is $S(x) \stackrel{\text{def}}{=} T(x)/x$, and the expected slowdown for a job of size x under scheduling policy P is $E[S(x)]^P$. The expected overall response time under scheduling policy P is $E[T]^P \stackrel{\text{def}}{=} \int_0^\infty E[T(x)]^P f(x)dx$ and the expected overall slowdown under scheduling policy P is $E[S]^P \stackrel{\text{def}}{=} \int_0^\infty E[S(x)]^P f(x)dx$.

Definition 1.1 *A job of size x is treated fairly under policy P iff $E[S(x)]^P \leq 1/(1-\rho)$. Further, a scheduling policy is fair iff it treats every job size fairly.*

Definition 1.2 *A job of size x is treated unfairly under policy P iff $E[S(x)]^P > 1/(1-\rho)$. Further, a scheduling policy is unfair iff there exists a job size x that is treated unfairly.*

The above definitions are motivated by the Processor-Sharing (`PS`) policy, which is fair in that every job achieves the same mean slowdown $(1/(1-\rho))$ under `PS` [28].

Many policies have been analyzed with respect to fairness. It has been shown that the `SRPT` policy is fair for all job size distributions when $\rho \leq 0.5$, however for higher loads `SRPT` becomes unfair [1, 27]. Further, many policies similar to `SRPT` have even worse fairness properties. In fact *all policies* that assign priority based on a job's size (e.g., `PSJF`) or based on a job's age[1] (e.g., `FB`) are "always unfair," meaning that they are unfair for all loads and all job size distributions [27]. Thus, even obvious approaches to curbing unfairness using hybrid algorithms that bias towards both small and sufficiently large job sizes (such as in [4, 8]) are still "always unfair."

Despite the growing amount of research, the search for a fair policy with near optimal performance proved elusive until last year at the 2003 ACM Sigmetrics conference when Friedman and Henderson presented a new policy called Fair-Sojourn-Protocol (`FSP`), which provides the first example of a fair policy that improves upon the performance of Processor-Sharing (`PS`) [9]. Until this time, the only "always fair" policies known were Processor-Sharing (`PS`) and Preemptive Last-Come-First-Served (`PLCFS`), both of which have the same mean response time.

The idea behind `FSP` is that it computes the times at which jobs would complete if the system were running `PS` and then orders the jobs in terms of their `PS` completion times. That is, `FSP` devotes the full processor to the (uncompleted) job with the earliest `PS` completion time. Thus, `FSP` can be thought of as performing `SRPT` on the remaining times of a virtual `PS` system. We defer a detailed description of the `FSP` algorithm to Section 2.

The `FSP` policy is provably fair, as seen directly from its definition – this will be explained below in Section 2. In addition, simulations show that `FSP` has very low mean response times, close to those of `SRPT` in many cases [9]. Unfortunately, analyzing the performance of the `FSP` policy has proven to be difficult, and thus the queueing performance of `FSP` has only be assessed via simulation.

The goal of this paper is to provide a first step towards analytically understanding the behavior of an M/GI/1/FSP queue, and towards beginning to understand the relative performance of `FSP` compared with the performance of other common policies. The paper is structured as follows. In Section 2 we define the `FSP` policy and describe its implementation. In Section 3, we derive upper and lower bounds on the expected response time seen by a job of size x under `FSP` using a novel analysis technique combining worst case bounds and stochastic analysis. In Section 4, we use these bounds to provide upper and lower bounds on the overall mean response time and overall mean slowdown under `FSP`. We find that the improvement of `FSP` over `PS` is substantial: for *all* job size distributions, the mean response time and mean slowdown under `FSP` are a fraction $(1 - \frac{\rho}{2})$ of that under `PS` , and for distributions with decreasing failure rate (see Section 4 for a definition of "decreasing failure rate"), the improvement is even greater. We also prove an analytic bound on the difference in mean response times under `FSP` and `SRPT`, which indicates that `FSP` achieves a near optimal

[1] A job's age is the service it has received so far.

mean response time. In Section 5, we verify the accuracy of our bounds using simulations, and illustrate our bounds on FSP evaluated over a wide range of job size distributions. We also discover another interesting benefit of the FSP policy: the mean response time and mean slowdown of FSP is largely insensitive to the coefficient of variation of the job size distribution, similarly to the PS policy.

2 Description of FSP

The easiest way to understand the FSP policy is to imagine that at any point in time you know the full state of a virtual PS queue, with the same arrival process. (Note, this won't actually be needed for the implementation below; however it makes it easy to understand.) Under the FSP policy, the job being run is always that job that the virtual PS queue would have completed first. Observe that the FSP policy, like PS, is work-conserving – it just avoids time-sharing by choosing to focus all attention on one job at a time.

To understand the power and efficiency of FSP, consider the simple scenario of 3 jobs of size $1-\varepsilon$, 1, $1+\varepsilon$ that all arrive at time 0 at the server. Under PS, all jobs would time-share the server, slowing each other down, and would all finish at about time 3. Under FSP, an ordering would be assigned to the jobs, and consequently the first arrival would finish at time $1-\epsilon$, the second at time $2-\epsilon$, and the third at time 3.

The above argument can be generalized to show that the FSP policy is "always fair" (every job finishes at least as early under FSP as compared with PS), since by its definition FSP is only reordering the work that is being done so as to be more efficient [9].

Observe that the FSP algorithm is related to the Weighted-Fair-Queueing (WFQ) algorithm. Both algorithms involve simulating PS, and FSP can be viewed as a *preemptive* version of WFQ where each packet forms its own stream. Note that this is askew from the way WFQ is used in practice, which is to evenly distribute bandwidth among connections in a network while adhering to packetized constraints. In such a setting, non-preemptive implementations are used. Further, due to the more general parameters of WFQ, analyses do not provide bounds on performance any better than to state that WFQ is not worse than PS by the length of the largest packet [6, 16]. The specific settings of WFQ that give rise to FSP have not been analyzed.

The implementation of FSP is not very complex, in fact it is quite similar to the implementation of the SRPT algorithm. As in SRPT, preemptions may only occur at moments when a new job arrives under FSP or when a job departs under FSP. Thus the total number of preemptions is at most twice the number of arriving jobs, which in practice is far less than the number of preemptions under implementations of PS (which involve time quantums). Aside from preemptions, as in SRPT, there are also priority updates needed under FSP. These priority updates occur only at "event times", where an event is an arrival or departure under FSP or under the virtual PS system. Again, the number of updates is clearly not great. It is at most three times the number of outside arrivals.

We now explain the priority update needed at event points. Let E_i denote the most recent prior event and let E_{i+1} denote the current event. Observe that during the time between any two events there is only one job, call it j, in service under FSP. During the time between E_i and E_{i+1}, call this time t, the remaining size of job j under FSP decreases by t. Further, at the moment of E_{i+1}, we need to decrease the remaining time of every job under the virtual PS system by t/n, where n is the number of jobs in PS. Observe that our definition of events ensures that the number of jobs under PS does not change between two consecutive events.

We have seen that the implementation of FSP is quite similar to that of SRPT. The SRPT scheduling policy has been implemented in many real-world applications, such as scheduling in Web servers [4, 11, 22]. The implementation of SRPT in [11] involves updating the priority of sockets in the Linux kernel, based on the remaining processing time required, and then draining these sockets into the network in order of their priority. An implementation of FSP would be equally simple, the only change would be that the priority updates would occur at the event points described above.

3 Bounds on the performance for a job of size x

In this section we will derive upper and lower bounds on $E[T(x)]^{FSP}$. As we have seen, the description of FSP relies heavily on decisions based on the current state of PS. Thus, it is natural to analyze FSP from the perspective of how it differs from PS.

We begin by noting that, by the definition of FSP, the following upper bound holds trivially.

Observation 3.1

$$E[T(x)]^{FSP} \le E[T(x)]^{PS} = \frac{x}{1-\rho}$$

In this section we will prove the following improvement on this bound.

Theorem 3.1 *In an M/GI/1 system,*

$$E[T(x)]^{FSP} \le \frac{x}{1-\rho} - \frac{x(\rho-\rho_x)}{1-\rho} - \frac{\lambda\int_0^x t\overline{F}(t)dt}{1-\rho}$$

where $X_x \stackrel{\text{def}}{=} \min(X, x)$ *and* $\rho_x \stackrel{\text{def}}{=} \lambda E[X_x] = \lambda \int_0^x \overline{F}(t)dt$, .[2]

Before proving this theorem we will need to recall one lemma from [9]. Let $\vec{r}^{PS}$ and $\vec{r}^{FSP}$ be vectors indicating the remaining work of each job under PS and FSP respectively. The vectors are ordered in the same way such that $r_1^{PS} \leq r_2^{PS} \leq \ldots \leq r_n^{PS}$. Thus, r_i^{PS} and r_i^{FSP} refer to the remaining work of the same job in PS and FSP respectively. Notice that (i) $\vec{r}^{FSP}$ is not necessarily ordered according to increasing remaining sizes and (ii) it is possible for $r_i^{FSP} = 0$ while $r_i^{PS} > 0$ for several values of i.

Lemma 3.1 *[9] For all* $m \leq n$,

$$\sum_{i=1}^{m} r_i^{PS} \geq \sum_{i=1}^{m} r_i^{FSP}$$

where n *is the number of jobs in the system.*

Proof: We prove the result using induction on the sequence of events in a busy period. An event is either a virtual service completion (a completion in PS), an arrival, or a service completion under FSP. Let $\vec{r}^{PS} \gg \vec{r}^{FSP}$ indicate that for all $m \leq n$, $\sum_{i=1}^m r_i^{PS} \geq \sum_{i=1}^m r_i^{FSP}$. We can see that the claim holds for the first arrival of a busy period; thus the base case holds trivially.

Now, suppose $\vec{r}^{PS} \gg \vec{r}^{FSP}$ at the time of event E_e, $e \geq 1$. Let $\vec{r}^{PS}$ and $\vec{r}^{FSP}$ be the vectors at the time of event E_e and let t be the time between the occurrence of event E_e and event E_{e+1}. Let $\vec{r}^{PS'}$ and $\vec{r}^{FSP'}$ be the updated vectors just before event E_{e+1}. Let i be the index of the first nonzero value in $\vec{r}^{FSP}$. Then

$$\begin{aligned}
r_j^{FSP'} &= r_j^{FSP} = 0,\ j = 1, \ldots, i-1 \\
r_i^{FSP'} &= r_i^{FSP} - t \\
r_j^{FSP'} &= r_j^{FSP},\ j = i+1, \ldots, n \\
r_j^{PS'} &= r_j^{PS} - t/n,\ \forall j
\end{aligned}$$

Notice that, because t is defined as the time between event E_e and E_{e+1}, $t \leq r_i^{FSP}$ and $t/n \leq r_j^{PS}$ for all j. We can now see that $\vec{r}^{PS'} \gg \vec{r}^{FSP'}$ is maintained immediately before event E_{e+1}, which completes the proof. ■

We are now ready to prove Theorem 3.1. Given the dependence of FSP on the details of PS, our analysis of FSP will be highly dependent on the following well known results about PS systems. Recall that the number of jobs in the system as seen by an arrival, N_{arr}^{PS}, and the number of jobs in system left behind by a departure, N_{dep}^{PS}, are both Geometric with mean $\rho/(1-\rho)$ [13]. Further, at both of these moments, the jobs in the system have independent, identically distributed ages and excesses distributed according to the equilibrium distribution, $\mathcal{E}$, where $f_{\mathcal{E}}(x) = \overline{F}(x)/E[X]$ [13], independent of the number of jobs in the system at that time.

Proof: (of Theorem 3.1) We will prove this bound by tracking the performance of a tagged job j_x of size x that arrives both into the FSP system and the virtual PS system. In our proof, we will look at work that must be done before j_x can leave under PS , but not under FSP . There are two such types of work. (i) There is work arriving after j_x arrives that interferes with j_x under PS but not under FSP. (ii) There is work already in the system at the time j_x arrives that interferes with j_x under PS but not under FSP. We will prove lower bounds on both types of work above. We can then subtract these lower bounds from $E[T(x)]^{PS}$ to obtain an upper bound on $E[T(x)]^{FSP}$.

We start with case (i). Any job that arrives while j_x is in the system under PS that does not complete before j_x leaves the system under PS will not be worked on under FSP while j_x is in the system. Thus, we can subtract all such work from $E[T(x)]^{PS}$. This work is lower bounded by the sum of the ages of jobs remaining in the PS system having age less than x under PS when j_x completes under PS. Notice that if a job has age greater than x in PS when j_x completes under PS that job must have arrived before x.

To total this work, we can take advantage of the fact that the number of jobs in the system when j_x departs the PS system is Geometric with mean $\rho/(1-\rho)$ and each job has age distributed according to the equilibrium distribution $\mathcal{E}$ independently of both the number of jobs in the system and the sizes of the other jobs in the system. We, however, are only interested in the jobs having age less than x at the time x departs. Let $I(t)$ denote the indicator random variable that takes 1 when t is true and 0 otherwise. Then we have

$$\begin{aligned}
E[T(x)]^{FSP} &\leq E[T(x)]^{PS} - E\left[\sum_{i=1}^{N_{dep}^{PS}} \mathcal{E} I(\mathcal{E} < x)\right] \\
&= \frac{x}{1-\rho} - \sum_{i=1}^{\infty} i E[\mathcal{E} I(\mathcal{E} < x)] P(N_{dep}^{PS} = i) \\
&= \frac{x}{1-\rho} - E[\mathcal{E} I(\mathcal{E} < x)] \sum_{i=1}^{N_{dep}^{PS}} i P(N_{dep} = i) \\
&= \frac{x}{1-\rho} - E[N_{dep}^{PS}] E[\mathcal{E} I(\mathcal{E} < x)] \\
&= \frac{x}{1-\rho} - \frac{\rho}{1-\rho}\left(\int_0^x t \frac{\overline{F}(t)}{E[X]} dt\right) \\
&= \frac{x}{1-\rho} - \frac{\lambda \int_0^x t\overline{F}(t)dt}{1-\rho} \qquad (1)
\end{aligned}$$

We now move to case (ii), which will complete the proof of the theorem. We will again track the performance of a tagged job j_x of size x. Let t_0 be the time that j_x arrives into the FSP system and the virtual PS system. We have

[2] One can view ρ_x as being the load in a queue where, when a job of greater than size x arrives, its size is immediately truncated to x.

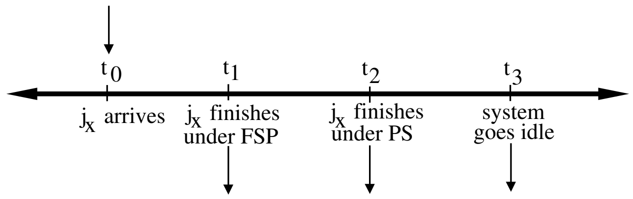


Figure 1: A diagram of the time points used for case (ii) in the proof of Theorem 3.1

already bounded the work in arriving jobs that `PS` does and `FSP` avoids; we are now going to bound the work in the system at t_0 that `PS` does and `FSP` avoids. In doing this, we will ignore all arrivals in the system that occur after j_x to simplify our arguments about this work. Since we are only analyzing the amount of work present when j_x arrives this does not affect the accuracy or our calculations.

Let t_1 be the time that `FSP` finishes j_x; t_2 be the time that `PS` finishes j_x; and t_3 be the first time when everyone who was in the system at t_0 has completed. Notice that t_3 is the same under `FSP` and `PS` since we are not allowing any new arrivals into either system and both policies are work conserving. Further notice that $t_0 \leq t_1 \leq t_2 \leq t_3$. These time points are summarized in Figure 1.

We are interested in bounding $t_2 - t_1$. Let $r_i(t_j)^{PS}$ be the i-th smallest remaining size at time t_j under `PS` and $r_i(t_j)^{FSP}$ be remaining size of the corresponding job at time t_j under `FSP`. Notice that

$$
\begin{aligned}
t_3 - t_1 &= \sum_{\{i : r_i(t_0)^{PS} > x\}} r_i^{FSP}(t_0) \\
&\geq \sum_{\{i : r_i(t_0)^{PS} > x\}} r_i(t_0)^{PS}
\end{aligned}
$$

and, since $t_3 - t_2$ is simply the work remaining in the `PS` system when j_x completes, we have

$$
\begin{aligned}
t_2 - t_1 &= (t_3 - t_1) - (t_3 - t_2) \\
&= (t_3 - t_1) - \sum_{\{i : r_i(t_0)^{PS} > x\}} (r_i(t_0)^{PS} - x) \\
&\geq \sum_{\{i : r_i(t_0)^{PS} > x\}} x
\end{aligned}
$$

Notice that this is the sum of x work for each job in the `PS` system having remaining size greater than x at t_0.

Finally, we can move back to the stochastic realm and again take advantage of the fact that the number of jobs in the system when x arrives is Geometric with mean $\rho/(1-\rho)$ and each job has age distributed according to the equilibrium distribution $\mathcal{E}$ independently of both the number of jobs in the system and the sizes of the other jobs in the system. Thus, subtracting from Equation 1 we have

$$
\begin{aligned}
E[T(x)]^{FSP} &\leq \frac{x}{1-\rho} - \frac{\lambda \int_0^x t\overline{F}(t)dt}{1-\rho} \\
&\quad -E\left[\sum_{i=1}^{N_{arr}} xI(\mathcal{E} > x)\right] \\
&= \frac{x}{1-\rho} - \frac{\lambda \int_0^x t\overline{F}(t)dt}{1-\rho} \\
&\quad -E[N_{arr}]E[xI(\mathcal{E} > x)] \\
&= \frac{x}{1-\rho} - \frac{\lambda \int_0^x t\overline{F}(t)dt}{1-\rho} \\
&\quad -\frac{\rho}{1-\rho}\left(\frac{x\int_x^\infty \overline{F}(x)}{E[X]}\right) \\
&= \frac{x}{1-\rho} - \frac{\lambda \int_0^x t\overline{F}(t)dt}{1-\rho} - \frac{x(\rho-\rho_x)}{1-\rho}
\end{aligned}
$$

■

We will see in Section 5 that Theorem 3.1 is tight enough to provide useful performance bounds on both the overall expected time in system and the overall expected slowdown. Already though, we can see that for small job sizes `FSP` significantly outperforms `PS`. Further, under `FSP` large jobs outperform `PS`, though only slightly. Thus, the performance improvement given to the small jobs does not sacrifice fairness.

It is also interesting at this point to compare $E[T(x)]^{FSP}$ to $E[T(x)]^{SRPT}$. We know that `SRPT` is sometimes unfair, thus for some large job sizes x, $E[T(x)]^{SRPT} \geq E[T(x)]^{PS} \geq E[T(x)]^{FSP}$. Further, we know that `SRPT` is optimal with respect to overall mean response time. Thus, there must be some other small job sizes x where $E[T(x)]^{SRPT} \leq E[T(x)]^{FSP}$.

Let us now make some remarks about the tightness Theorem 3.1.

Remark 3.1 *Notice that the slowdown for a job of size x under* `FSP` *can be expressed as follows:*

$$
\begin{aligned}
E[S(x)]^{FSP} &\leq \frac{1}{1-\rho} - \frac{\rho-\rho_x}{1-\rho} - \frac{\lambda\int_0^x t\overline{F}(t)dt}{x(1-\rho)} \\
&= \left(1 + \frac{\rho_x}{1-\rho}\right) - \frac{\lambda\int_0^x t\overline{F}(t)dt}{x(1-\rho)}
\end{aligned}
$$

We can see that the bound in Theorem 3.1 is tight in the limiting cases.

$$
\begin{aligned}
\lim_{x\to 0} E[S(x)]^{FSP} &\leq 1 \\
\lim_{x\to\infty} E[S(x)]^{FSP} &\leq \frac{1}{1-\rho}
\end{aligned}
$$

The fact that the limit as $x \to \infty$ is tight follows from the fact that every Always Fair policy has limiting slowdown of

$1/(1-\rho)$ *as* $x \to \infty$ *[27], and the fact that the limit as* $x \to 0$ *is tight follows from the fact that a slowdown of 1 is optimal.*

Remark 3.2 *A second remark is that the bound in Theorem 3.1 is tight in the case of deterministic service distributions. That is, if all arriving jobs have size b, we know that no job arriving after x will be worked on while x is in the system, and all jobs that arrived before x will complete before x is worked on. Thus, in an M/D/1 queue* FSP *is simply performing* FCFS.

This can also be seen by manipulating the bound on FSP. *Let* $E[X] = b$ *for an M/D/1 queue.*

$$\begin{aligned} E[T(b)]^{FSP} &= E[T(b)]^{FCFS} \\ &= b + \frac{\lambda b^2}{2(1-\rho)} \\ &= b + \frac{\rho b}{1-\rho} - \frac{\lambda b^2}{2(1-\rho)} \\ &= b + \frac{b\rho_b}{1-\rho} - \frac{\lambda \int_0^b t\overline{F}(t)dt}{1-\rho} \end{aligned}$$

To understand why the bound in Theorem 3.1 is not exact when the service distribution is not deterministic consider the following. The proof is broken into two cases: (i) the work from arrivals that are guaranteed to not be worked on under FSP but are worked on under PS; and (ii) the work in the system that is worked on under PS but not under FSP. In both cases the analysis only bounds the work. In the first case, there can be arrivals into the system after the tagged job completes under FSP that complete under PS before the tagged job. These arrivals are not included in our bound. In case (ii) we are not exact because of the dependence on Lemma 3.1, which only provides a bound on the desired quantity of work.

We now turn our attention to deriving a simple lower bound for $E[T(x)]^{FSP}$ that will be useful when we lower bound $E[S]^{FSP}$.

Theorem 3.2 *Define* $\rho(x) \stackrel{\text{def}}{=} \lambda \int_0^x tf(t)dt$[3] *and* $m_2(x) = \int_0^x t^2 f(t)dt$. *Then, in an M/GI/1 system,*

$$E[T(x)]^{FSP} \geq x(1+\rho(x)) + \frac{\lambda}{2}\left(x^2\overline{F}(x) - m_2(x)\right)$$

Proof: This lower bound follows from the fact that at least the following three pieces of work must be finished while a job j_x of size x is in the system: (i) x work to complete j_x; (ii) at least $\int_0^x \lambda E[X : X < t]dt = \int_0^x \rho(t)dt$ work from arrivals, since when j_x has remaining size t an arrival with size smaller than t will complete before j_x under FSP; (iii) the remaining size (up to size x) of the job in service when j_x arrives. This last quantity is easily seen to be $\frac{\lambda}{2}\left(\int_0^x t^2 f(t)dt + x^2\overline{F}(x)\right)$, which is the expected excess of jobs with remaining size at most x. We now simplify this expression as follows:

$$\begin{aligned} E[T(x)]^{FSP} &\geq x + \int_0^x \rho(t)dt + \frac{\lambda}{2}\left(m_2(x) + x^2\overline{F}(x)\right) \\ &= x + \lambda \int_0^x \int_0^t sf(s)dsdt \\ &\quad + \frac{\lambda}{2}\left(m_2(x) + x^2\overline{F}(x)\right) \\ &= x + \lambda \int_0^x sf(s)\int_s^x dtds \\ &\quad + \frac{\lambda}{2}\left(m_2(x) + x^2\overline{F}(x)\right) \\ &= x + \lambda\left(x\int_0^x sf(s)ds - \int_0^x s^2 f(s)ds\right) \\ &\quad + \frac{\lambda}{2}\left(m_2(x) + x^2\overline{F}(x)\right) \\ &= x + x\rho(x) - \lambda m_2(x) \\ &\quad + \frac{\lambda}{2}\left(x^2\overline{F}(x) + m_2(x)\right) \\ &= x(1+\rho(x)) + \frac{\lambda}{2}\left(x^2\overline{F}(x) - m_2(x)\right) \end{aligned}$$

■

It should be immediately obvious that this bound is quite loose for large job sizes. What is surprising is that it already gives a good lower bound on $E[S]^{FSP}$.

4 Bounds on mean performance

We will now quantify the overall performance improvements attained by FSP. We will use the upper bound on $E[T(x)]^{FSP}$ derived in Theorem 3.1 and the lower bound in Theorem 3.2 to begin to understand the overall mean performance of FSP.

Theorem 4.1 *In an M/GI/1/FSP system with an arbitrary service distribution:*

$$\begin{aligned} E[T]^{FSP} &\geq \left(\frac{1-\rho}{\rho}\right)\log(1-\rho)E[T]^{PS} \\ E[T]^{FSP} &\leq \left(1-\frac{\rho}{2}\right)E[T]^{PS} \end{aligned}$$

Proof: We will first calculate an upper bound on the ratio of $E[T]^{FSP}$ to $E[T]^{PS}$. The proof follows from manipulation of the bound proved for FSP in Theorem 3.1. Recall that $\mathcal{E}$ is the equilibrium distribution, $\mu = 1/E[X]$, where X represents the service distribution, and that $E[T(x)]^{PS} = x/(1-\rho)$.

[3] Note the difference between $\rho(x)$ and the previously introduced ρ_x. While ρ_x is the load in a system where arrivals of greater than size x are truncated to size x, $\rho(x)$ is the load in a system where arrivals of greater than size x are discarded.

$$
\begin{aligned}
\frac{E[T]^{FSP}}{E[T]^{PS}} &= \frac{1-\rho}{E[X]}\int_{x=0}^{\infty} E[T(x)]^{FSP} f(x)dx \\
&\leq \frac{1-\rho}{E[X]}\int_{x=0}^{\infty}\left(\frac{x}{1-\rho} - \frac{x(\rho-\rho_x)}{1-\rho}\right. \\
&\quad \left. - \frac{\lambda\int_{t=0}^{x} t\overline{F}(t)dt}{1-\rho}\right) f(x)dx \\
&= 1-\mu\lambda\left(\int_{x=0}^{\infty} x\int_{t=x}^{\infty}\overline{F}(t)dt f(x)dx\right. \\
&\quad \left. + \int_{x=0}^{\infty} f(x)\int_{t=0}^{x} t\overline{F}(t)dtdx\right) \\
&= 1-\mu\lambda\left(\int_{t=0}^{\infty}\overline{F}(t)\int_{x=0}^{t} xf(x)dxdt\right. \\
&\quad \left. + \int_{t=0}^{\infty} t\overline{F}(t)\int_{x=t}^{\infty} f(x)dxdt\right) \\
&= 1-\mu\lambda\int_{t=0}^{\infty}\overline{F}(t)\left(\int_{x=0}^{t} xf(x)dx + t\overline{F}(t)\right)dt \\
&= 1-\mu\lambda\int_{t=0}^{\infty}\overline{F}(t)\int_{x=0}^{t}\overline{F}(x)dxdt \\
&= 1-\rho\int_{t=0}^{\infty}\frac{\overline{F}(t)}{E[X]}\frac{\int_{x=0}^{t}\overline{F}(x)dx}{E[X]}dt \\
&= 1-\rho\int_{t=0}^{\infty} f_{\mathcal{E}}(t)F_{\mathcal{E}}(t)dt \\
&= 1-\frac{\rho}{2}\left(F_{\mathcal{E}}(t)\right)^2\Big|_{t=0}^{\infty} \\
&= 1-\frac{\rho}{2}
\end{aligned}
$$

The lower bound follows from Theorem 5.7 in [26], where this lower bound was shown to hold for SRPT, which is the optimal policy for overall mean response time. Since SRPT is optimal, $E[T]^{SRPT} \leq E[T]^{FSP}$ ∎

Notice that every step in the calculation of the upper bound above, other than the initial bounding of $E[T(x)]^{FSP}$, is exact. Thus, this upper bound is tight for deterministic distributions, since our upper bound on $E[T(x)]^{FSP}$ is tight for deterministic distributions. Further, it is interesting to notice that the bounds on $E[T]^{FSP}$ do not depend on the variability of the service distribution. Thus, they bound the effect that variability can have on the mean response time. This observation will be investigated further in Section 5.

We can illustrate the tightness of these bounds analytically as follows.

Corollary 4.1 *In an M/GI/1 system with an arbitrary service distribution*

$$
\begin{aligned}
E[T]^{FSP} &\leq E[T]^{SRPT} + \sum_{n=2}^{\infty}\left(\frac{1}{2}-\frac{1}{n+1}\right)\rho^n E[X] \\
&\leq E[T]^{SRPT} + \frac{\rho^2}{2}E[T]^{PS}
\end{aligned}
$$

That is, FSP *is within an additive factor of $\frac{\rho^2}{2}E[T]^{PS}$ from the optimal mean response time.*

Proof : We will first show that the upper and lower bounds in Theorem 4.1 differ additively by $\delta(\rho)E[T]^{PS}$ where

$$\delta(\rho) \stackrel{\text{def}}{=} \left(\frac{\rho^2}{3\cdot 2} + \frac{\rho^3}{4\cdot 3} + \ldots\right)$$

Thus $E[T]^{FSP}$ is within $\delta(\rho)E[T]^{PS}$ of $E[T]^{SRPT}$, the optimal mean response time.

Recall that $-\log(1-\rho) = \sum_{i=1}^{\infty}\frac{\rho^i}{i}$. Thus,

$$-\left(\frac{1-\rho}{\rho}\right)\log(1-\rho) = 1 - \frac{\rho}{2\cdot 1} - \frac{\rho^2}{3\cdot 2} - \frac{\rho^3}{4\cdot 3} - \ldots$$

Comparing this to the upper bound on $E[T]^{FSP}$ we notice that the two differ only by lower order terms of ρ, specifically $\delta(\rho)$. Further, $-\left(\frac{1-\rho}{\rho}\right)\log(1-\rho)E[T]^{PS}$ is not only a lower bound for FSP, it is a lower bound on the performance of SRPT . Thus, we have shown that $E[T]^{FSP}$ differs from the optimal $E[T]$ only by lower order terms.

We will now complete the proof by analyzing $\delta(\rho)E[T]^{PS}$. Recall that $\frac{1}{1-\rho} = \sum_{i=0}^{\infty}\rho^i$. Further,

$$\sum_{i=2}^{n}\frac{1}{k(k+1)} = \sum_{i=2}^{n}\frac{1}{k} - \frac{1}{k+1} = \frac{1}{2} - \frac{1}{n+1}$$

Thus,

$$\delta(\rho)E[T]^{PS} = \sum_{n=2}^{\infty}\left(\frac{1}{2}-\frac{1}{n+1}\right)\rho^n E[X]$$

Finally, we can observe that

$$\sum_{n=2}^{\infty}\left(\frac{1}{2}-\frac{1}{n+1}\right)\rho^n E[X] \leq \frac{\rho^2}{2}E[T]^{PS}$$

∎

This corollary tells us that, despite the fact that FSP sacrifices some performance to guarantee fairness, it still achieves near optimal performance. In Section 5, we will use simulations to further investigate both the accuracy of our bounds on the performance of FSP and the comparison of FSP to SRPT.

We now move to bounding the overall mean slowdown.

	Coxian				Bounded Pareto			
$C^2[X]$	β	μ_1	μ_2	$E[\min(X_1, X_2)]$	α	k	p	$E[\min(X_1, X_2)]$
1/16	-	-	-	-	5.1231	0.8048	10^7	0.8918
1/8	-	-	-	-	4.0000	0.7500	10^7	0.8571
1/4	-	-	-	-	3.2361	0.6903	10^7	0.8171
1/2	-	-	-	-	2.7320	0.6340	10^7	0.7748
1	0.0000	1.0000	1.0000	0.5000	2.4135	0.5857	10^7	0.7354
2	0.4142	3.4142	0.5858	0.3091	2.2166	0.5489	10^7	0.7029
3	0.3019	3.5811	0.4189	0.2567	-	-	-	-
4	0.2377	3.6733	0.3267	0.2277	2.0868	0.5208	10^7	0.6765
5	0.1962	3.7321	0.2679	0.2092	-	-	-	-
6	0.1670	3.7728	0.2272	0.1963	-	-	-	-
7	0.1454	3.8028	0.1972	0.1869	-	-	-	-
8	0.1287	3.8257	0.1743	0.1797	1.9903	0.4767	10^7	0.6538
9	0.1155	3.8439	0.1561	0.1739	-	-	-	-
10	0.1047	3.8586	0.1414	0.1693	-	-	-	-
16	-	-	-	-	1.9110	0.4767	10^7	0.6328
32	-	-	-	-	1.8413	0.4569	10^7	0.6123
64	-	-	-	-	1.7772	0.4373	10^7	0.5916
128	-	-	-	-	1.7167	0.4175	10^7	0.5702

Table 1: A summary of the service distributions used for simulations. Note that empty regions of the table indicate only that these $C^2[X]$ values were not used in our plots, not that there is no distribution with that $C^2[X]$ value.

Theorem 4.2 *In an M/GI/1/FSP system with an arbitrary service distribution:*

$$E[S]^{FSP} \geq \left(1 - \rho + (1-\rho)\frac{\lambda}{2}E[\min(X_1, X_2)]\right) E[S]^{PS}$$
$$E[S]^{FSP} \leq \left(1 - \rho + \frac{\lambda}{2}E[\min(X_1, X_2)]\right) E[S]^{PS}$$

where X_1 and X_2 are independent random variables from identical to the service distribution.

To better understand this result and its proof consider that under an exponential service distribution $\lambda E[\min(X_1, X_2)] = \rho/2$; however under a deterministic service distribution $\lambda E[\min(X_1, X_2)] = \rho$. Thus, under an exponential service distribution we get bounds of

$$\left(1 - \frac{3\rho}{4} - \frac{\rho^2}{4}\right) E[S]^{PS} \leq E[S]^{FSP} \leq \left(1 - \frac{3\rho}{4}\right) E[S]^{PS}$$

and under the deterministic distribution we obtain looser bounds of

$$\left(1 - \frac{\rho}{2} - \frac{\rho^2}{2}\right) E[S]^{PS} \leq E[S]^{FSP} \leq \left(1 - \frac{\rho}{2}\right) E[S]^{PS}$$

It is interesting to note that the $(1 - \rho/2)$ factor is tight for deterministic distributions.

To further understand the distribution of $\min(X_1, X_2)$ we prove the following theorem.

Theorem 4.3 *If X, X_1, X_2 are independent and identically distributed random variables having a decreasing failure rate*[4] *then*

$$E[\min(X_1, X_2)^i] \leq \frac{1}{2^i}E[X^i]$$

Proof : Let $F(x)$ and $f(x)$ be the distribution function and density function of X respectively. Observe that

$$\overline{F}_{min}(x) = \overline{F}(x)^2$$
$$f_{min}(x) = 2f(x)\overline{F}(x)$$

We can therefore write the failure (hazard) rate of $\min(X_1, X_2)$, $\mu_{min}(t)$, in term of the failure rate of X, $\mu(t)$, as

$$\mu_{min}(t) = f_{min}(t)/\overline{F}_{min}(t) = \frac{2f(x)\overline{F}(x)}{\overline{F}(x)^2} = 2\mu(t)$$

Further, we can write

$$\overline{F}_{min}(x) = e^{-\int_0^x \mu_{min}(t)dt}$$

[4]The failure rate of a distribution is defined as $\mu(t) \stackrel{\text{def}}{=} f(t)/\overline{F}(t)$, and a distribution has decreasing failure rate (DFR) if $\mu(t)$ is non-increasing. Examples of distributions with decreasing failure rates are Weibull and Pareto distributions. In general, DFR distributions tend to be highly variable.

Now, we can derive the desired bound:

$$\begin{aligned} E[\min(X_1,X_2)^i] &= i\int_{x=0}^{\infty} x^{i-1}\overline{F}_{min}(x)dx \\ &= i\int_{x=0}^{\infty} x^{i-1}e^{-\int_{t=0}^{x}\mu_{min}(t)dt}dx \\ &= i\int_{x=0}^{\infty} x^{i-1}e^{-\int_{t=0}^{x}2\mu(t)dt}dx \\ &\leq i\int_{x=0}^{\infty} x^{i-1}e^{-\int_{t=0}^{2x}\mu(t)dt}dx \\ &= i\int_{x=0}^{\infty} x^{i-1}\overline{F}(2x)dx \\ &= \frac{i}{2^i}\int_{u=0}^{\infty} u^{i-1}\overline{F}(u)du \\ &= \frac{1}{2^i}E[X^i] \end{aligned}$$

■

We now return to the proof of Theorem 4.2.

Proof : (of Theorem 4.2) The proof follows from manipulation of the bounds proved for FSP in Theorem 3.1 and Theorem 3.2. We will first calculate an upper bound on the ratio of $E[S]^{FSP}$ to $E[S]^{PS}$.

$$\begin{aligned} \frac{E[S]^{FSP}}{E[S]^{PS}} &= (1-\rho)\int_{x=0}^{\infty} E[S(x)]^{FSP}f(x)dx \\ &\leq (1-\rho)\int_{x=0}^{\infty}\left(\frac{1}{1-\rho}-\frac{\rho-\rho_x}{1-\rho}\right. \\ &\quad \left.-\frac{\lambda\int_0^x t\overline{F}(t)dt}{x(1-\rho)}\right)f(x)dx \\ &= 1-\rho+\int_{x=0}^{\infty}\rho_x f(x)dx \\ &\quad -\lambda\int_{x=0}^{\infty}\frac{f(x)}{x}\int_{t=0}^{x}t\overline{F}(t)dtdx \\ &= 1-\rho+\lambda\int_{x=0}^{\infty}f(x)\int_{t=0}^{x}\overline{F}(t)dtdx \\ &\quad -\lambda\int_{x=0}^{\infty}\frac{f(x)}{x}\left(\int_{s=0}^{x}s^2f(s)ds+x^2\overline{F}(x)\right)dx \\ &\leq 1-\rho+\lambda\int_{t=0}^{\infty}\overline{F}(t)\int_{x=t}^{\infty}f(x)dxdt \\ &\quad -\lambda\int_{x=0}^{\infty}xf(x)\overline{F}(x)dx \\ &= 1-\rho+\lambda\int_{t=0}^{\infty}\overline{F}(t)\overline{F}(t)dt \\ &\quad -\frac{\lambda}{2}\int_{x=0}^{\infty}x\left(2f(x)\overline{F}(x)\right)dx \\ &= 1-\rho+\frac{\lambda}{2}E[\min(X_1,X_2)] \end{aligned}$$

The lower bound on $E[S]^{FSP}$ follows in the same manner by integrating the lower bound on $E[S(x)]^{FSP}$ derived in Theorem 3.2.

$$\begin{aligned} E[S]^P &\geq \int_{x=0}^{\infty}\left(1+\rho(x)+\frac{\lambda}{2x}\left(x^2\overline{F}(x)-m_2(x)\right)\right)f(x)dx \\ &\geq 1+\int_{x=0}^{\infty}\left(\rho(x)+\frac{\lambda}{2}x\overline{F}(x)-\frac{1}{2}\rho(x)\right)f(x)dx \qquad (2) \\ &= 1+\frac{1}{2}\int_{x=0}^{\infty}f(x)\rho(x)dx+\frac{\lambda}{2}\int_{x=0}^{\infty}xf(x)\overline{F}(x)dx \\ &= 1+\frac{\lambda}{2}\int_{x=0}^{\infty}x2f(x)\overline{F}(x)dx \\ &= 1+\frac{\lambda}{2}E[\min(X_1,X_2)] \\ &= \left(1-\rho+(1-\rho)\frac{\lambda}{2}E[\min(X_1,X_2)]\right)E[S]^{PS} \end{aligned}$$

where (2) follows from the observation that

$$\lambda m_2(x)=\lambda\int_0^x t^2f(t)dt\leq x\rho(x)$$

■

In order to illuminate the effect of the service distribution on our bound for $E[S]^{FSP}$, we now present some corollaries of Theorems 4.2 and 4.3.

Corollary 4.2 *In an M/GI/1/FSP system with an arbitrary service distribution the additive difference between the upper and lower bounds in Theorem 4.2 is less than $(\rho^2/2)E[S]^{PS}$. Further, if the service distribution has a decreasing failure rate, the additive difference between the bounds is less than $(\rho^2/4)E[S]^{PS}$.*

Proof : The proof follows from Theorem 4.2 by noticing that the difference between the bounds is $\rho\lambda E[\min(X_1,X_2)]/2$. Further, for an arbitrary service distribution $E[\min(X_1,X_2)]\leq E[X]$ and for a distribution with decreasing failure rate Theorem 4.3 tells us that $E[\min(X_1,X_2)]\leq E[X]/2$. ■

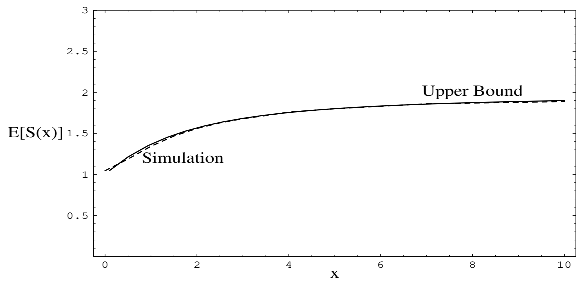
Corollary 4.3 *In an M/GI/1/FSP system with an arbitrary service distribution:*

$$E[S]^{FSP}\leq\left(1-\frac{\rho}{2}\right)E[S]^{PS}$$

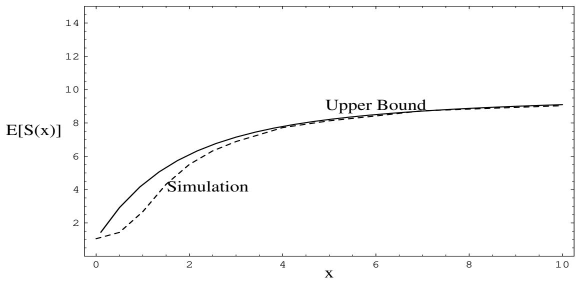
Further, in an M/GI/1/FSP system where the service distribution has a decreasing failure rate:

$$E[S]^{FSP}\leq\left(1-\frac{3\rho}{4}\right)E[S]^{PS}$$

Proof : The proof follows immediately from Theorem 4.2 by noticing that for independent X_1, X_2, $E[\min(X_1,X_2)]\leq E[X]$ under all service distributions. We can guarantee even better performance if the service distribution has a decreasing failure rate by applying Theorem 4.3. ■



(a) $\rho = .5$



(b) $\rho = .9$

Figure 2: These plots compare simulations of FSP with the upper bound on expected slowdown under FSP proven in Theorem 3.1. The dotted line corresponds to the simulation of FSP, and the solid line corresponds to the upper bound proven on $E[S(x)]^{FSP}$. The service distribution is taken as an Exponential with mean 1. The system load is 0.5 in column (a) and 0.9 in column (b).

5 Accuracy of bounds

In this section, we will evaluate the accuracy of the bounds presented in Section 4 by comparing the analytical results derived in this work to simulations of an FSP queue. We will first describe the setup of our simulations. Next, we will illustrate the accuracy of the bounds on $E[T(x)]^{FSP}$ derived in Section 3. Then, we will illustrate the tightness of the bounds derived for $E[T]^{FSP}$ and $E[S]^{FSP}$ in Section 4. Here, we will also compare the performance of FSP to the performance of SRPT, the policy that minimizes the overall mean response time [20] .

5.1 Simulation Setup

Throughout this section we will be simulating an M/GI/1/FSP queue. In these simulations we will use either two-stage Coxian or bounded Pareto service distributions. Recall that a two-stage Coxian random variable, C, is defined as $C = X_1$ with probability $1-\beta$ and $C = X_1 + X_2$ with probability β where $\beta \in [0,1]$ and X_1, X_2 follow exponential distributions with rates $\mu_1 > 0$ and $\mu_2 > 0$ respectively. A bounded Pareto distribution is defined as:

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

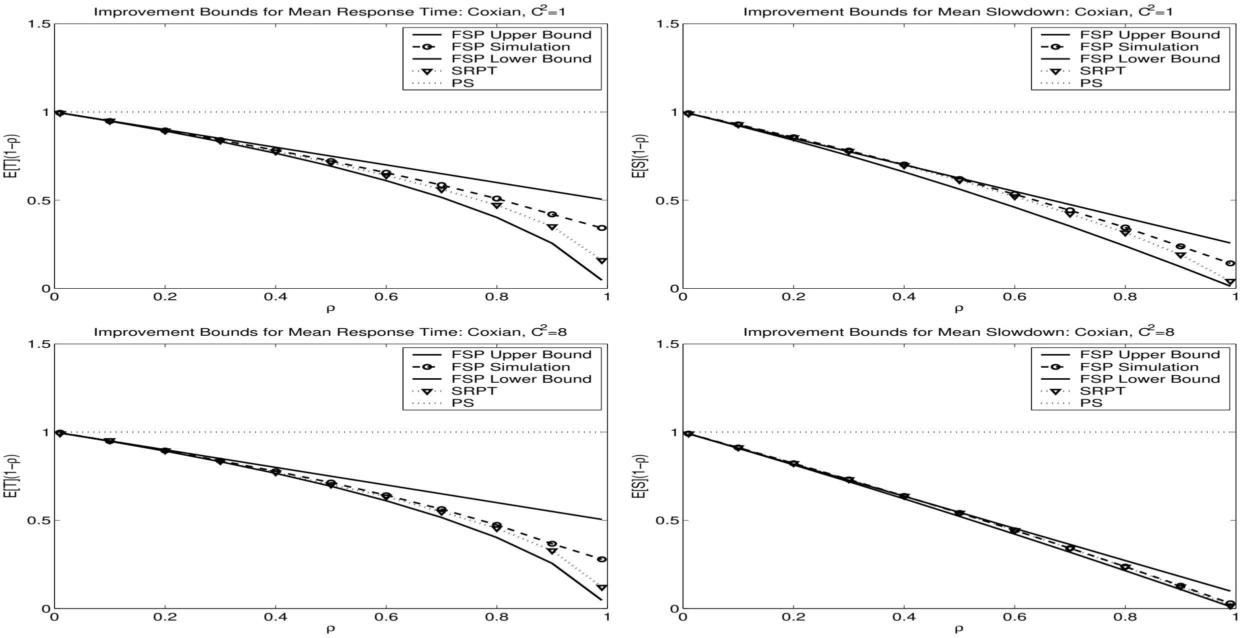
for $x \in [k,p]$.

These service distributions allow us to isolate the effects of the variability of the service distribution and tail behavior. In order to accomplish this, we fix the mean of the service distribution to 1, and then vary the second moment of the service distribution. The square coefficient of variation of the service distribution ($C^2[X] = E[X^2]/E[X]^2 - 1$) can be set almost arbitrarily. This corresponds to changing the variability of the service distribution. Notice that when the variability of a Coxian distribution is adjusted, the tail behavior of the distribution is unchanged. Under a bounded Pareto distribution however, a change in variability is the result of a change in the tail behavior of the distribution. The parameters used for these distributions in the simulations described here are summarized in Table 1. The appropriate parameters for the bounded Pareto distributions were obtained by fixing the upper bound of the service distribution, which leaves two unknown parameters, and then numerically solving the two equations $E[X] = 1$ and $C^2[X] = K$ for each K of interest. The procedure for finding the appropriate parameters of the Coxian distributions is symmetric.

Notice that the weight of the tail of the bounded Pareto distribution is allowed to vary from far lighter than that of an exponential to far heavier than that of an exponential. Thus, since Coxian distributions all have exponential tails, we have isolated the effect of the variability of the distribution from the effect of the behavior of the tail of the distribution. Investigating the effect of the weight of the tail of the service distribution is important in light of many recent measurements that have observed job size distributions which are well-modeled by bounded Pareto distributions or other heavy tailed distributions [2, 7, 14, 18].

5.2 Simulation Results

We first investigate the tightness of the bounds on $E[S(x)]^{FSP}$ derived in Section 3. Figure 2 compares these bounds with simulations of FSP . In order to investigate the performance of FSP for individual job sizes, jobs in the simulations are grouped into bins corresponding to small ranges of job sizes, and then the aggregate performance of each bin is plotted. The bin sizes used were of width 0.25. Figure 2 shows that the upper bound proven in Theorem 3.1 is quite accurate for all loads when the service distribution is exponential. This is not surprising since the bound is exact in the



(a) Overall mean response time

(b) Overall mean slowdown

Figure 3: These plots show a comparison between our bounds on FSP and the performance of SRPT. Notice that the performance of FSP is comparable both in terms of mean response time and mean slowdown. The service distributions in these plots are Coxian with mean 1 and $C^2 = 1, 8$ in the first and second rows respectively.

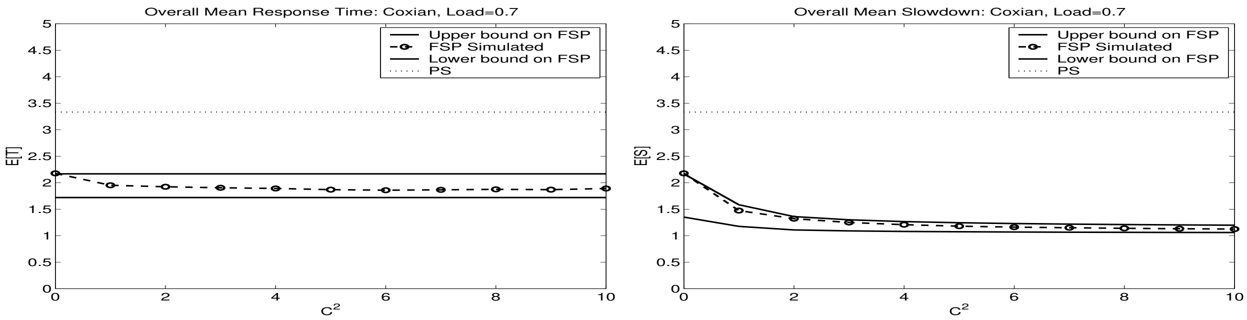
case of a deterministic service distribution. However, when the service distribution is highly variable, the upper bound becomes loose under high load; although the upper bound is still accurate for low load.

We now move to investigating the bounds on $E[T]^{FSP}$ and $E[S]^{FSP}$ derived in Section 4. Recall that in order to evaluate these bounds we need to understand $E[\min(X_1, X_2)]$ for X_1 and X_2 independent and identically distributed according to the service distribution. These minimums are listed in Table 1. Note that the different tail behavior of the Coxian and bounded Pareto distributions causes these values to differ greatly between the two classes of distributions

In order to investigate the overall performance of FSP we show three sets of plots. Figure 3 illustrates the performance of FSP when compared with PS and SRPT under varying loads for distributions with both low and high variability. Figure 4 investigates the performance of FSP under Coxian distributions with varying C^2 for low, medium, and highly loaded systems. Figure 5 shows the same information under bounded Pareto service distributions covering a wider range of variability.

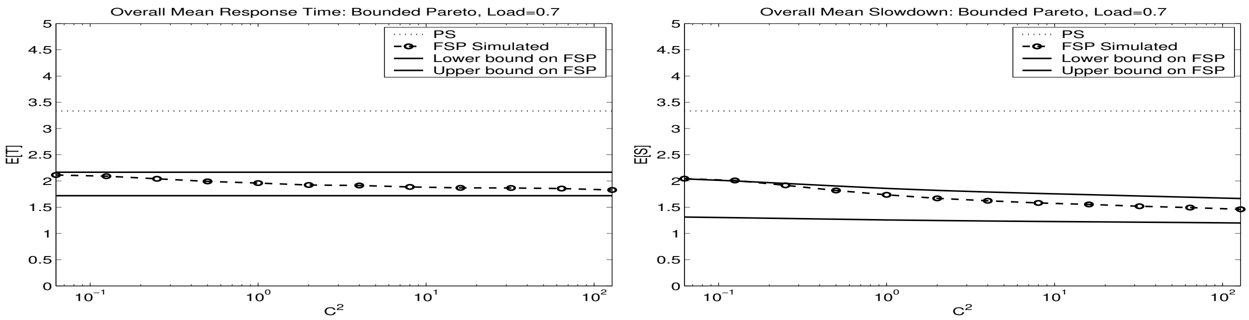
Figure 3 shows the mean response time and mean slowdown of FSP, SRPT, and PS normalized by $(1 - \rho)$ in the cases of both an exponential distribution and a highly variable distribution. This figure illustrates the huge performance gains FSP achieves over PS under medium and high loads. Further, we see that FSP nearly matches the performance of SRPT. This is a very important observation because FSP is providing near optimal mean response time without sacrificing unfairness. It is also interesting to notice the tightness of the bounds proven on $E[T]^{FSP}$ and $E[S]^{FSP}$ in Theorems 4.1 and 4.2. In all cases the bounds on FSP are quite tight up to medium-high loads. However, the bounds on $E[S]^{FSP}$ under highly variable distributions are tight for quite high loads as well.

Figures 4 and 5 also illustrate the huge performance gains obtained by FSP. It is interesting to notice that the bounds on $E[S]^{FSP}$ become tighter as the variability of the service distribution increases. Surprisingly, these plots indicate that both $E[T]^{FSP}$ and $E[S]^{FSP}$ are somewhat independent of both the variability in the service distribution and the weight of the tail of the service distribution. Specifically, $E[T]^{FSP}$ and $E[S]^{FSP}$ do not change much once the square coefficient of variation is larger than 1 regardless of the weight of the tail of the service distribution. It is known that PS has this property; however, due to the bias FSP gives to short job sizes, it is surprising that the variability of the service distribution has such a small effect on $E[T]^{FSP}$ and $E[S]^{FSP}$. This has some practical ramifications if it can be proven to hold in general. First, for analysis, this indicates that performance gains seen in M/M/1 systems are good approximations for those seen in M/G/1 systems, even when G is highly variable. Second, for simulations, this observation



(a) Overall mean response time (b) Overall mean slowdown

Figure 4: These plots show a comparison between the bounds proven for FSP and simulation results for FSP. The service distribution in these plots is a Coxian with mean 1, and varying coefficient of variation. The system load is 0.7.



(a) Overall mean response time (b) Overall mean slowdown

Figure 5: These plots show a comparison between the bounds proven for FSP and simulation results for FSP. The service distribution in these plots is a Bounded Pareto with mean 1, and varying coefficient of variation. The system load is 0.7.

indicates that we need not take the time to simulate highly variable service distributions, which require much longer simulation times than low variability distributions.

6 Conclusion

The advent of the FSP scheduling policy has sparked considerable interest. FSP is the first policy to exhibit fairness for all job sizes while improving mean response times over PS. In addition, FSP is practical to implement.

This paper provides the first analysis of the M/GI/1/FSP queue by proving upper and lower bounds on its mean response time and mean slowdown. The bounds are obtained using a novel technique that blends worst-case bounds and stochastic arguments to understand how FSP differs from PS. Despite the complexity of the argument, the resulting bounds are very simple, stated in terms of just ρ, the load, and the minimum of two random variables representing job size. These upper and lower bounds are close in that they differ additively by only $\frac{\rho^2}{2}E[T]^{PS}$ in both the case of mean response time and mean slowdown. Further, if the service distribution has a DFR, the bounds on mean slowdown are only separated by $\frac{\rho^2}{4}E[T]^{PS}$. Thus, in all cases, the bounds differ only by lower order functions of ρ.

The simplicity of the resulting bounds is important because it allows easy comparison of the performance of FSP to that of other policies. In particular, our bounds show that the improvement of FSP over PS is substantial: for *all* job size distributions the mean response time and mean slowdown under FSP are a fraction $(1-\frac{\rho}{2})$ of that under PS, and for distributions with decreasing failure rate, the fraction becomes $(1-\frac{3\rho}{4})$ in the case of mean slowdown. Furthermore, we prove that for *all* job size distributions FSP is only an additive factor of $\frac{\rho^2}{2}E[T]^{PS}$ worse than optimal (SRPT) with respect to mean response time.

The simplicity of the upper and lower bounds also allows us to observe that, surprisingly, the variability of the service distribution does not have much effect on the performance of FSP. In fact, because our analytic bounds on mean response time are independent of the variability of the service distribution, we can see that the maximum effect variability can have on mean response time is an additive factor of $\frac{\rho^2}{2}E[T]^{PS}$. Although, it is well known that PS is insensitive to the variability of the service distribution, it is surprising, given the bias FSP gives to small job sizes, that FSP ia also largely unaffected the service distribution's variability.

References

[1] N. Bansal and M. Harchol-Balter. Analysis of SRPT scheduling: Investigating unfairness. In *Proceedings of ACM Sigmetrics Conference on Measurement and Modeling of Computer Systems*, 2001.

[2] P. Barford and M. Crovella. Generating representative web workloads for network and server performance evaluation. In *Proceedings of ACM Sigmetrics Conference on Measurement and Modeling of Computer Systems*, 1998.

[3] 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.

[4] 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.

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

[6] A. Demers, S. Keshav, and S. Shenkar. Analysis and simulation of a fair queueing algorithm. *Journal of Internetworking*, 1:3–26, 1990.

[7] A. B. Downey. Evidence for long-tailed distributions in the internet. In *Proceedings of ACM SIGCOMM Internet Measurment Workshop*, 2001.

[8] H. Feng and V. Misra. Mixed scheduling disciplines for network flows (the optimality of FBPS). In *Workshop on MAthematical performance Modeling and Analysis (MAMA 2003)*, 2003.

[9] E. Friedman and S. Henderson. Fairness and efficiency in web server protocols. In *Proceedings of ACM Sigmetrics Conference on Measurement and Modeling of Computer Systems*, 2003.

[10] M. Gong and C. Williamson. Quantifying the properties of SRPT scheduling. In *IEEE/ACM International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*, 2003.

[11] M. Harchol-Balter, B. Schroeder, N. Bansal, and M. Agrawal. Implementation of SRPT scheduling in web servers. *ACM Transactions on Computer Systems*, 21(2), May 2003.

[12] M. Harchol-Balter, K. Sigman, and A. Wierman. Asymptotic convergence of scheduling policies with respect to slowdown. *Performance Evaluation*, 49(1-4):241–256, 2002.

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

[14] W. Leland, M. Taqqu, W. Willinger, and D. Wilson. On the self-similar nature of ethernet traffic. In *Proceedings of SIGCOMM '93*, pages 183–193, September 1993.

[15] T. O'Donovan. Direct solutions of M/G/1 priority queueing models. *Revue Francaise d'Automatique Informatique Recherche Operationnelle*, 10:107–111, 1976.

[16] A. Parekh and R. Gallager. A generalized processor sharing approach to flow control in integrated services networks: the single node case. *IEEE/ACM Transactions on Networking*, 1:344–357, 1993.

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

[18] D. L. Peterson. Data center I/O patterns and power laws. In *CMG Proceedings*, December 1996.

[19] I. Rai, G. Urvoy-Keller, and E. Biersack. Analysis of LAS scheduling for job size distributions with high variance. In *Proceedings of ACM Sigmetrics Conference on Measurement and Modeling of Computer Systems*, 2003.

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

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

[22] B. Schroeder and M. Harchol-Balter. Web servers under overload: How scheduling can help. In *International Teletraffic Congress (ITC 2003)*, 2003.

[23] A. Sliberschatz, P. Galvin, and G. Gagne. *Operating System Concepts, Sixth Edition*. John Wiley & Sons, 2002.

[24] D. 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, Fourth Edition*. Prentice Hall, 2001.

[26] A. Wierman and M. Harchol-Balter. Bounding the performance of SMART scheduling policies. Technical Report CMU-CS-03-199, Carnegie Mellon University, November 2003.

[27] A. Wierman and M. Harchol-Balter. Classifying scheduling policies with respect to unfairness in an M/GI/1. In *Proceedings of ACM Sigmetrics Conference on Measurement and Modeling of Computer Systems*, 2003.

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