Revisiting the performance of large jobs in the M/GI/1 queue

Adam Wierman

***Abstract*— In recent years, the response times (sojourn times) experienced by large job sizes have been the focus of a large number of papers. Though results about many scheduling disciplines have appeared, to this point, results characterizing the response time of large job sizes have been limited to either mean value analysis or fluid scalings. In this paper, we present a new framework that unites these prior results and provides new results characterizing the distributional behavior of large job sizes. Our new framework provides results for diffusion scalings and beyond. In addition, we illustrate the impact these new results have for the analysis of busy periods and for predicting response times of large customers upon arrival.**

I. Introduction

Traditionally, the performance of scheduling policies has been measured using mean response time (a.k.a. sojourn time, flow time), and more recently mean slowdown and the tail of response time. Under these measures, policies that give priority to small job sizes (a.k.a. service requirements) at the expense of larger job sizes perform quite well. For example, Shortest-Remaining-Processing-Time (SRPT) is known to optimize mean response time [24]. As a result, designs based on these policies have been suggested for a variety of computer systems in recent years, including web servers [9], [23], wireless networks [11], [15], peer-to-peer systems [20], operating systems [7], databases [17], [18], transport protocols [37], and beyond. However the adoption of these new designs has been slowed by fears about the fairness of policies that prioritize small jobs. Specifically, there are worries that large job sizes may be "starved" of service under a policy that favors small job sizes, which would result in *large job sizes having response times that are unfairly long and variable [2], [28], [29], [31].*

These worries have recurred nearly everywhere size based policies have been suggested. One example is the case of web servers, where recent designs have illustrated that giving priority to requests for small files can significantly reduce response times [9], [23]. However, it is important that this improvement does not come at the expense of providing large job sizes unfairly large response times, which are typically associated with the important requests. A second example is the case of operating systems. UNIX processes are assigned decreasing priority based on their current age – CPU usage so far. The worry is that this may create unfairness for old processes [7]. The same tradeoff has appears across diverse application areas, whenever a new design favors small requests.

To address these worries, there has been a growing amount of theoretical work studying the "fairness" of scheduling policies. The goal of these papers (and the current paper) is to characterize the performance of large jobs sizes: when are large jobs starved of service and how unfairly are large jobs treated?

The first answers to this question came in the setting of the M/GI/1 queue. Using mean-value-analysis (MVA), Bansal & Harchol-Balter [1] studied the response times of large jobs under SRPT. Then, Harchol-Balter, Sigman, & Wierman [10] generalized those results to prove that large job sizes receive asymptotically equivalent response times under a range of common scheduling policies. Specifically, for each of SRPT, Processor Sharing (PS), Preemptive Shortest Job First (PSJF), Foreground Background scheduling (FB), and Preemptive Last Come First Served (PLCFS), [10] proved that if the second moment of the service distribution is finite,

$$\lim_{x\to\infty} \frac{E[T(x)]}{x} = \frac{1}{1-\rho}, \tag{1}$$

where $T(x)$ is the conditional response time experienced by a job of size x and $\rho < 1$ is the stationary load of the queue.

This result was surprising for many people – it says that the performance of large job sizes is asymptotically

Adam Wierman is an Assistant Professor of Computer Science at the California Institute of Technology, 1200 E. California Blvd. MC 256-80, Pasadena, CA 91125, USA. Email: adamw@caltech.edu

equivalent under fair policies like PS and biased policies like SRPT and FB.[1] Resultantly, it spawned a large number of followup papers that extended the result to other scheduling policies, e.g. [21], [14] and removed the requirement of a finite second moment, e.g. [3], [14]. Also, (1) motivated the study of the behavior of $E[T(x)]/x$ away from the limit, e.g. [10], [22], [34], [35]. In fact, many results are still appearing on this topic.

Not only has (1) been extended to a wide variety of policies, but in many cases it has been shown to hold in the following, much stronger, setting:

$$\lim_{x\to\infty} \frac{T(x)}{x} = \frac{1}{1-\rho} \quad \text{a.s.} \tag{2}$$

This statement serves as a law-of-large-numbers (LLN) or fluid scaling for the performance of large jobs.

Results of the form of (2) are more scattered. The RHS of (2) has been shown to be an a.s. upper bound for all work conserving policies by Harchol-Balter, Sigman, & Wierman [10] when the job size distribution has a finite second moment. Further, without the assumption of a finite second moment, (2) has been shown to hold for PS and some variants by Guillemin, Robert, & Zwart [8] and for SRPT, PSJF, FB, and some variants (including the SMART class) by Wierman, Nuyens, & Zwart [19]. These results even hold without the assumption of Poisson arrivals – they hold in the GI/GI/1 queue.

Though (2) is a very strong statement about the behavior of large jobs, it is also clear that a lot of information about the distribution is hidden by the heavy scaling used (i.e. $1/x$). Specifically, it is very possible that two policies could have the same fluid limit of $T(x)$, but have the response times of large jobs exhibit very different distributional behavior.

The goal of this paper is to understand if the performance of large jobs is still equivalent under a wide range of scheduling policies when weaker scalings of $T(x)$ are considered. In order to accomplish this goal, we will present a general technique for studying $T(x)$ for large job sizes that will include both MVA and fluid scaling results in addition to diffusion scalings, i.e. $(T(x) - E[T(x)])/\sqrt{x}$ as $x \to \infty$, and other even weaker scalings. This technique sheds new light on the MVA and fluid scaling results and provides important insight into the distributional behavior of $T(x)$ for large job sizes. Additionally, we will illustrate that understanding the performance of large jobs provides surprising benefits, e.g. in the symbolic computation of busy period moments.

Our results use the cumulant moments to characterize the distributional behavior of $T(x)$. The calculation of cumulant moments is often delicate, but the recent stochastic calculus introduced by Duffield, Massey, & Whitt [6] for the analysis of time-varying traffic in telecommunication models provides us with a useful tool that we can adapt to our goal.

The remainder of the paper is organized as follows. We will first provide an overview of the scheduling policies studied and notation used in the paper in Section II. Also, in Section II, we will provide the reader a refresher on the properties of cumulant moments. Then, in Section III, we will present and discuss the main results of the paper, and their impact. Next, in Section IV, we will discuss the proof technique we use to attain our results. Unfortunately, due to limited space, we must omit the full details of the proofs. Finally, we conclude the paper with a brief discussion in Section V.

II. Preliminaries

Our focus in this paper is on work conserving, preempt-resume single server queues, *the M/GI/1 model unless otherwise stated*. The policies that we will consider are summarized in Table I. We define T and $T(x)$ to be the steady-state response time overall and for a job of size x respectively, and we let $\rho < 1$ be the system load. That is $\rho = \lambda E[X]$, where λ is the arrival rate of the system and X is a random variable distributed according to a continuous service (job size) distribution $F(x)$ having density function $f(x)$ defined for all $x \geq 0$. Let $\overline{F}(x) = 1 - F(x)$.

Busy periods will be very important to our analysis. Let B denote the length of a busy period, and recall that $E[B] = E[X]/(1-\rho)$ and $E[B^2] = E[X^2]/(1-\rho)^3$. Further, let $B(x)$ be a busy period started by x work.

Since we will be working with many priority based policies, we need to use transformations of the service distributions and busy periods during the analysis. In particular, one important quantity will be $X_{<x} = X|X < x$. Correspondingly, we define $\lambda_{<x} = \lambda F(x)$

[1] It should be noted that [10] proves that non-preemptive policies do not satisfy (1). Specifically, under all non-preemptive policies $E[T(x)]/x \to 1$ as $x \to \infty$.

FB	Foreground-Background preemptively serves those jobs that have received the least amount of service so far.
PLCFS	Preemptive Last Come First Served preemptively serves the most recent arrival.
PS	Processor Sharing serves all customers simultaneously, at the same rate.
PSJF	Preemptive Shortest Job First preemptively serves the job in the system with the smallest original size.
SRPT	Shortest Remaining Processing Time preemptively serves the job with the shortest remaining size.

TABLE I

A brief description of the scheduling policies discussed in this paper.

and $B_{<x}$ as the length of a busy period with arrival rate $\lambda_{<x}$ and service distribution $X_{<x}$.

Finally, the last quantity that we will introduce here are the *cumulant moments*. Cumulants have appeared only sporadically in queueing, tending to be used in large deviation limits. Formally, the cumulant moments of a random variable X, $\kappa_i[X]$ $i = 1, 2, \ldots$, are defined in terms of the moments of X, $E[X^i]$, as follows:

$$e^{\kappa_1[X]t + \frac{\kappa_2[X]t^2}{2!} + \cdots} = 1 + E[X]t + \frac{E[X^2]t^2}{2!} + \ldots$$

From this definition it follows that the cumulants of X can be generated from the cumulant generating function,

$$\mathcal{K}_X(s) = \log(\mathcal{L}_X(s)),$$

where $\mathcal{L}_X(s)$ is the Laplace transform of X. That is, $(-1)^i \mathcal{K}_X^{(i)}(0) = \kappa_i[X]$. Further, the cumulants of any distribution can be found from the raw moments as follows

$$\kappa_n[X] = E[X^n] - \sum_{j=1}^{n-1} \binom{n-1}{j} E[X^j] \kappa_{n-j}[X] \quad (3)$$

where $\kappa_1[X] = E[X]$ [13]. Immediately from (3), we can see that the cumulants capture many of the standard descriptive statistics. The first cumulant is the mean; the second cumulant is the variance; and the third cumulant is the third central moment and thus measures the skewness of the distribution. Beyond the third cumulant, the cumulants differ from the central and raw moments, see [13] for tables of the relationships between higher order cumulants, raw moments, and central moments.

Although not immediately evident from the definition, cumulants have many properties that both raw and central moments lack. For instance, letting c be a constant, $\kappa_1[X + c] = \kappa_1[X] + c$ but for $i \geq 2$, $\kappa_i[X + c] = \kappa_i[X]$. Thus, the first cumulant is shift-equivariant, but all others are shift-invariant. Other nice properties of cumulants include homogeneity and additivity. Homogeneity states that $\kappa_i[cX] = c^i \kappa_i[X]$. Additivity states that for independent random variables X and Y, $\kappa_i[X + Y] = \kappa_i[X] + \kappa_i[Y]$.

III. The response time of large jobs

The goal of this paper is to understand the asymptotics of $T(x)$ as $x \to \infty$ beyond the MVA and fluid limits. In order to attain this more delicate understanding, we make use of the cumulant moments. Interestingly, it turns out that $\kappa_i[T(x)]$ grows asymptotically linearly with x for all i under all work-conserving policies, which leads to the use of $\kappa_i[T(x)]/x$ as a metric for studying the asymptotic behavior of $T(x)$ as $x \to \infty$. Using this metric, we can obtain the following result, which we will sketch the proof of in Section IV.

Theorem 1 *In an M/GI/1 queue with $E[X^i] < \infty$, for $P \in \{\mathsf{PLCFS}, \mathsf{PS}, \mathsf{PSJF}, \mathsf{SRPT}, \mathsf{FB}\}$,*

$$\lim_{x \to \infty} \frac{\kappa_i[T(x)]^P}{x} = 1_{i=1} + \lambda E[B^i] \quad (4)$$

The usefulness of this result is a bit difficult to see upon first glance, so we will spend the rest of this section discussing the result.

First, it is useful to provide some intuition for the RHS of (4). Notice that the $1_{i=1}$ term is a direct consequence of the fact that we are using cumulants. In particular, $T(x)$ must include the job size itself x. Since this is a constant, it only affects the first cumulant. Additionally, $\lambda E[B^i]$ can be viewed as the ith cumulant of a compound Poisson process with arrival rate λ and jumps of size B. This observation is key to our proof approach in Section IV.

Second, Theorem 1 says that a wide range of policies provide asymptotically equivalent response times for large job sizes in a very strong sense, much stronger than either the MVA or fluid limits. Thus, large job sizes are treated asymptotically equivalently under fair policies like PS and biased policies like SRPT and PSJF.

We will illustrate the strength of Theorem 1 as compared with the prior literature in the next section, and then we will provide a few short examples of the impact of Theorem 1.

A. A generalization of prior work on fairness

We already mentioned that Theorem 1 provides a stronger statement about the asymptotic response times of large jobs than the MVA or fluid limits. To illustrate this concretely, it is useful to show that we can attain the MVA and fluid limits very easily from Theorem 1.

To obtain the MVA limit, we can simply observe that:

$$\begin{aligned}\lim_{x\to\infty}\frac{E[T(x)]^P}{x} &= \lim_{x\to\infty}\frac{\kappa_1[T(x)]^P}{x}\\ &= 1+\lambda E[B]\\ &= \frac{1}{1-\rho}\end{aligned}$$

To obtain the fluid scaling results, we need only slightly more work. First, we observe that:

$$\begin{aligned}\lim_{x\to\infty}\kappa_i\left[\frac{T(x)}{x}\right]^P &= \lim_{x\to\infty}\frac{\kappa_i[T(x)]^P}{x^i}\\ &= \begin{cases}\frac{1}{1-\rho}, & i=1;\\ 0, & i>1.\end{cases}\end{aligned}$$

Then, we note that the unique distribution with such a sequence of cumulant moments is a point mass at $1/(1-\rho)$, which is enough to guarantee a.s. convergence in this setting.

Not only can we obtain prior results using Theorem 1, but we can attain interesting new results about weaker scalings of $T(x)$. The following corollary characterizes the diffusion limit of $T(x)$.

Corollary 1 *In an M/GI/1 queue with $E[X^i]<\infty$ for all i, for $P\in\{\mathsf{PLCFS},\mathsf{PS},\mathsf{PSJF},\mathsf{SRPT}\}$,*

$$\frac{T(x)^P-E[T(x)]^P}{\sqrt{x}}\xrightarrow{\mathrm{d}}Z,\quad \text{as } x\to\infty \tag{5}$$

where Z follows a normal distribution with mean 0 and variance $\lambda E[B^2]$.

Proof: We start by applying Theorem 1.

$$\begin{aligned}\lim_{x\to\infty}\kappa_i\left[\frac{T(x)-E[T(x)]}{\sqrt{x}}\right]^P &= \lim_{x\to\infty}\frac{\kappa_i[T(x)-E[T(x)]]^P}{x^{i/2}}\\ &= \begin{cases}\lambda E[B^2], & i=2;\\ 0, & i\neq 2.\end{cases}\end{aligned}$$

Now, we simply notice that the Normal distribution is the unique distribution with $\kappa_i=0$ for $i>2$, and that κ_1 and κ_2 specify the mean and variance. ∎

Corollary 1 can be viewed as a CLT for the performance of large job sizes, and illustrates that the performance of large job sizes is asymptotically equivalent at a finer scale than just the fluid limit. Further, this corollary extends recent work by Ward & Whitt [32] and Wierman & Harchol-Balter [35] addressing the question: How accurately can response times be predicted? This is often a relevant question for long job sizes because large delays can often cause customers to renege, and predictions of response times have been shown to reduce to likelihood of a customer to renege in many applications.

At this point, it should be clear that other, more delicate, scalings than even the diffusion scaling, can easily be obtained from Theorem 1. Further, according to all of these scalings, the response times of large jobs will be asymptotically equivalent under both fair (e.g. PS) and biased (e.g. SRPT) policies.

B. Tighter asymptotics for moments of $T(x)$

Beyond its importance for understanding fairness, Theorem 1 also provides new results for the asymptotics of the moments of $T(x)$, which is an important topic in its own right. For instance, in the case of PS, the asymptotic moments of $T(x)$ have been the topic of a number of papers, e.g. [40], [4] and the references therein.

Corollary 2 *Consider an M/GI/1 queue with $E[X^i]<\infty$. If P satisfies (4) and*

$$\lim_{x\to\infty}E[T(x)]^P-\frac{x}{1-\rho}=\delta,$$

then

$$E[T(x)^i]^P=\left(\frac{x}{1-\rho}\right)^i+(i(i-1)E[W]+i\delta)\left(\frac{x}{1-\rho}\right)^{i-1}+o(x^{i-1}) \tag{6}$$

where $E[W]=\frac{\lambda E[X^2]}{2(1-\rho)}$ is the stationary workload.

The proof follows from an inductive argument applying Theorem 1 in combination with (3).

To illustrate the use of Corollary 2, let us consider a few examples. First, note that under PS $\delta=0$. Applying Corollary 2 then gives a result that matches the tightest known asymptotics for the moments of $T(x)^{\mathsf{PS}}$ [40]. Further, $\delta=0$ under PLCFS and all other SYMMETRIC policies, as defined by Kelly [12].

Next, consider SRPT. From known formulas for $E[T(x)]^{\mathsf{SRPT}}$, e.g. in [25], it is easy to see that $\delta=$

$E[W]$. Applying Corollary 2 then gives asymptotics that are, to the best of our knowledge, tighter than the first order asymptotics that exist in the literature (see for example [39]). Interestingly, $\delta = E[W]$ under many other policies as well, e.g. FB, PSJF, and all SMART policies defined in [36].

C. Calculating the moments of the busy period

Typically, the symbolic calculation of the moments of busy periods is a computationally inefficient task. Many strides have been made toward increasing the efficiency of these calculations, see [5] and the references therein. Theorem 1 provides a new technique for simplifying these calculations. In particular, we can provide a bridge between the calculation of moments of busy periods, which is typically inefficient, and the calculation of moments of the workload process, which is typically efficient.

To accomplish this, we make use of results from Zwart & Boxma [40] characterizing the moments of PS in terms of the moments of the convolutions of the workload distribution. These convolutions can be represented using cumulants easily and then calculated using Takács recursive formula [30]. Finally, Theorem 1 provides a bridge between the asymptotics of $T(x)^{\mathsf{PS}}$ and busy periods moments as follows:

$$E[B^i] = \frac{1}{\lambda} \lim_{x\to\infty} \frac{\kappa_i[T(x)]^{\mathsf{PS}}}{x}$$

Details of this approach can be found in [33].

IV. Proving Theorem 1

Now that we have spent some time understanding the applications of Theorem 1, we will move to a discussion of how to prove the result. In order to prove the result, we need to have the ability to sum over the arrivals during the service given to a tagged customer, the amount of resources in use and for how long they are used. We will do this by taking advantage of a stochastic integral introduced by Duffield, Massey, & Whitt [6]. Though the integral was developed for use in the analysis of time-varying traffic in telecommunication networks, Shalmon [27] independently used a representation of PS very related to this calculus to derive explicit formulas for the variance of the M/D/1 queue. We will see here that this calculus provides a useful technique for studying a wide variety of scheduling policies.

In this section we will first develop the stochastic integral, and then provide a few examples of how to analyze scheduling policies using the integral. Limited space has forced us to exclude some details, and show only a subset of the policies, but an extended version of this paper is under submission.

A. A stochastic calculus for cumulants

Let X_n, $n = 1, 2, \ldots$ be an i.i.d. sequence of service times and $\phi : \mathbb{R}^2 \to \mathbb{R}$ be a non-negative measurable function. Define a stochastic integral with respect to a nonhomogeneous Poisson process A as follows

$$\int_0^x \phi(X_{A(t)}, t) dA(t) = \sum_{n=1}^{A(x)} \phi(X_n, \hat{A}_n) \tag{7}$$

where $dA(t) = A(t) - A(t-)$ and $\hat{A}_n$ is the time of the nth arrival in the interval $(0, t]$. Notice that this can be defined as a sample path integration.

This integral has many nice properties, which are discussed in [6], [16]. However, for our purposes, the following result is all that is necessary. It illustrates that the stochastic integral captures the behavior of cumulant moments in a very intuitive manner. The result follows from Theorem 2.2 in [16].

Theorem 2 *If* $\int_0^x E[e^{s\phi(X,t)} - 1]\lambda(t)dt < \infty$ *for some* $s > 0$ *and*

$$T(x) \stackrel{\mathrm{d}}{=} x + \int_0^x \phi(X_{A(t)}, t) dA(t), \tag{8}$$

then

$$\frac{\kappa_i[T(x)]}{x} = 1_{i=1} + \frac{1}{x} \int_0^x E[\phi(X, t)^i]\lambda(t)dt \tag{9}$$

We will see in the following subsections that this result provides a very natural way in which to approach the analysis of a variety of scheduling policies.

B. Analyzing PLCFS

The case of PLCFS provides us with a useful intuitive understanding of Theorem 2.

Let us view $T(x)^{\mathsf{PLCFS}}$ according to the following branching decomposition, introduced by Shalmon in [26] to explain the Pollaczek-Khinchin formula. Consider a tagged job of size x. Then, $T(x)^{\mathsf{PLCFS}}$ is made up of x work from the tagged job and a busy periods worth of work for every arrival while the tagged job is being worked on. So, $T(x)^{\mathsf{PLCFS}}$ can be viewed as

a compound Poisson process, and letting $A(t)$ be the arrival process, we have

$$T(x)^{\mathsf{PLCFS}} \stackrel{\mathrm{d}}{=} x + \sum_{n=1}^{A(x)} B_i = x + \int_0^x B dA(t),$$

which gives

$$\begin{aligned} \frac{\kappa_i[T(x)]^{\mathsf{PLCFS}}}{x} &= 1_{i=1} + \frac{1}{x}\int_0^x E[B^i]\lambda dt \\ &= 1_{i=1} + \lambda E[B^i] \end{aligned}$$

This case provides the intuition for studying other policies, where our goal is to show that, asymptotically, all arrivals arrivals during service to the tagged job contribute a busy period to the response time.

C. Analyzing SRPT

Our next example is SRPT. To analyze SRPT, we will write $T(x)$ as the sum of the waiting time experienced by a job of size x, $W(x)$, and the residence time experienced by a job of size x, $R(x)$. The waiting time of a job is the time between the arrival of the job and the moment the job first receives service. The residence time is the time from when a job first receives service until it exits the system. Using simple properties of cumulants, we have that

$$\kappa_i[T(x)]^{\mathsf{SRPT}} = \kappa_i[R(x)]^{\mathsf{SRPT}} + \kappa_i[W(x)]^{\mathsf{SRPT}}$$

Further, it is straightforward, though tedious, to show that $\kappa_i[W(x)]^{\mathsf{SRPT}}/x \to 0$ as $x \to 0$ by analyzing the moments of $W(x)$ and applying (3). We do not have space to include this derivation here. But, note that it is immediate when $E[X^{i+1}] < \infty$.

Using the above, we have that

$$\lim_{x\to\infty} \frac{\kappa_i[T(x)]^{\mathsf{SRPT}}}{x} = \lim_{x\to\infty} \frac{\kappa_i[R(x)]^{SRPT}}{x}.$$

Now, we will complete the analysis by using the stochastic calculus to approach the residence time under SRPT. Noting that when the tagged job has remaining size t under SRPT, exactly those arrivals with size $< t$ can interrupt the tagged job. Further, each of these arrivals begins a busy period including all arrivals of size $< t$ that must complete before the tagged job returns to service. It follows that

$$R(x)^{\mathsf{SRPT}} \stackrel{\mathrm{d}}{=} x + \int_0^x B_{<t} dA_{<t}(t), \tag{10}$$

which gives

$$\lim_{x\to\infty} \frac{\kappa_i[R(x)]^{\mathsf{SRPT}}}{x} = 1_{i=1} + \lim_{x\to\infty} \int_0^x E[B^i_{<t}]\lambda_{<t}(t)dt$$

Using this form, we can now easily upper and lower bound the cumulants as follows. These bounds show the usefulness of Theorem 1 since it allows us to apply simple stochastic bounds in order to attain bounds on the cumulants, which are typically more difficult to bound.

Let us start with the upper bound. In this case, we use the fact that $B_{<t} \leq_{st} B$, which gives

$$\begin{aligned} \lim_{x\to\infty} \frac{\kappa_i[R(x)]^{\mathsf{SRPT}}}{x} &\leq 1_{i=1} + \lim_{x\to\infty} \frac{1}{x}\int_0^x E[B^i]\lambda dt \\ &= 1_{i=1} + \lambda E[B^i] \end{aligned}$$

For the lower bound, we need to be a little more careful. We use the facts that (i) for $s < t$, $B_s \leq_{st} B_t$, (ii) $B_{<t} \to B$ as $t \to \infty$ and (iii) $\lambda_{<t} \to \lambda$ as $t \to \infty$. Let $0 < \varepsilon < 1$.

$$\begin{aligned} \lim_{x\to\infty} \frac{\kappa_i[R(x)]^{\mathsf{SRPT}}}{x} &\geq 1_{i=1} + \lim_{x\to\infty} \frac{1}{x}\int_{\varepsilon x}^x E[B^i_{<\varepsilon x}]\lambda_{<\varepsilon x} dt \\ &= 1_{i=1} + \lambda(1-\varepsilon)E[B^i] \end{aligned}$$

Since ε can be made arbitrarily small, this completes the proof.

D. Analyzing PS

To analyze PS, we will view $T(x)^{\mathsf{PS}}$ as a branching process. This view was first introduced implicitly by Yashkov [38], [39] and exploited explicitly by Shalmon [27]. The idea is to to view the response time for a tagged job j_x of size x as the sum of the lengths of branches in a random tree. Corresponding to each job, there is a branch of length equal to the size of the job. Let n_t be the number of branches at time t. When j_x arrives (at time $t = 0$) it sees $n_0 = N^{PS}$ existing branches. It is well known that N^{PS} is distributed geometrically with parameter ρ and that each job in the system has remaining size $\mathcal{E}$ [12].

Upon arrival, the tagged job starts a branch of length x. The process evolves as t grows by having the total arrival rate at each time t be $n_t\lambda$, split evenly among the n_t branches in the system at time t. The arrivals each form new branches attached to the branch they occurred during. One can see that this is equivalent to a PS queue by scaling the time in the branching process by n_t at each time t. Using this equivalence, it is clear that the response time of j_x is simply the sum of the lengths of the branches in the system between time 0 and time x.

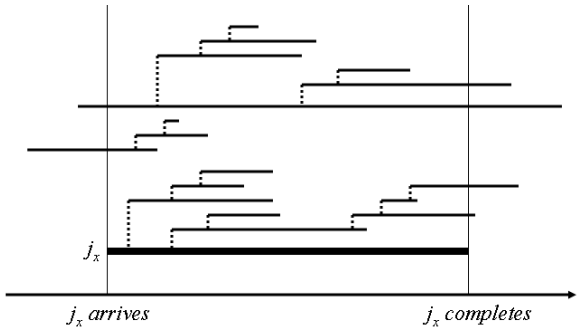

Fig. 1. An illustration of how to view PS as a branching process. In this diagram there are two jobs in the queue when the tagged job j_x arrives. The response time of j_x is the sum of the lengths of the branches between the arrival and completion instants of j_x.

If we denote the sum of the lengths of the branches of a tree started by a branch of length b at height 0 between time 0 and time a as $L_a(b)$, we can use this view of PS to write $T(x)^{PS}$ as follows:

$$T(x)^{PS} \stackrel{\mathrm{d}}{=} L_x(x) + \sum_{i=1}^{N^{PS}} L_x(\mathcal{E}) \tag{11}$$

As under SRPT, the effect of the work in the system at the arrival of the tagged job can be ignored in the limit. In particular, we have that

$$\lim_{x\to\infty} \frac{\kappa_i[T(x)]^{\mathsf{PS}}}{x} = \lim_{x\to\infty} \frac{\kappa_i[L_x(x)]}{x}.$$

We do not include the proof of this fact due to lack of space, but note that it is immediate if $E[X^{i+1}] < \infty$.

Now, we will see that the stochastic calculus is well suited for analyzing this branching process, despite the fact that it is typically a difficult process to understand. This observation was also used by Shalmon in [27] in his derivation of an explicit formula for the variance of response time in the M/D/1 PS queue. To apply the calculus, we notice that if a job arrives while the tagged job is receiving service and has remaining size t, then the arrival starts a subtree that contributes $L_t(X \wedge t)$ to the response time of the tagged job, where $X \wedge t = \min(X, t)$. So, we have

$$L_x(x) \stackrel{\mathrm{d}}{=} x + \int_0^x L_t(X \wedge t) dA(t)$$

Using Theorem 2, we can easily obtain an upper bound on the limit by noticing that $L_t(X \wedge t) \leq_{st} B$:

$$\begin{aligned}
\lim_{x\to\infty} \frac{\kappa_i[L_x(x)]}{x} &= 1_{i=1} + \lim_{x\to\infty} \frac{1}{x} \int_0^x E[L_t(X \wedge t)^i] \lambda dt \\
&\leq 1_{i=1} + \lim_{x\to\infty} \frac{1}{x} \int_0^x E[B^i] \lambda dt \\
&= 1_{i=1} + \lim_{x\to\infty} \lambda E[B^i]
\end{aligned}$$

The lower bound requires us to be a little more careful, and illustrates how much the fact that Theorem 2 allows the use of stochastic bounds simplifies the calculation. We will use the fact that $L_t(x \wedge t) \geq_{st} L_\gamma(X \wedge \delta)$ if $t \geq \gamma$ and $t \geq \delta$

$$\begin{aligned}
\lim_{x\to\infty} \frac{\kappa_i[L_x(x)]}{x} &= 1_{i=1} + \lim_{x\to\infty} \frac{1}{x} \int_0^x E[L_t(X \wedge t)^i] \lambda dt \\
&\geq 1_{i=1} + \lim_{x\to\infty} \frac{1}{x} \int_\delta^x E[L_t(X \wedge \delta)^i] \lambda dt \\
&\geq 1_{i=1} + \lim_{x\to\infty} \frac{1}{x} \int_\delta^x E[L_\gamma(X \wedge \delta)^i] \lambda dt \\
&= 1_{i=1} + \lambda E[L_\gamma(X \wedge \delta)^i]
\end{aligned}$$

We now let $\delta, \gamma \to \infty$ with $\delta > \gamma$ to make $E[L_\gamma(X \wedge \delta)^i] \to E[B^i]$ and complete the proof.

V. Concluding remarks

In this paper we have revisited the question of whether large job sizes are treated unfairly under policies that prioritize small jobs. Our goal has been to extend the literature beyond mean value and fluid limit results. Under these heavy handed scalings previous research had observed that a wide range of policies provide large job sizes asymptotically equivalent response times. Here, we have shown that large job sizes receive asymptotically equivalent response times under a range of policies, even when weaker scalings are considered (Theorem 1). Theorem 1 encompasses diffusion scalings and beyond. Further, Theorem 1 provides a new intuition about the performance of large job sizes as well as a number of surprising impacts beyond the performance of large job sizes, e.g. the calculation of busy period moments.

Ongoing work is exploring another possible impact of Theorem 1. We can view Theorem 1 as a statement that the largest job sizes under a range of policies finish after (nearly) all other jobs that arrive while they are in the system. Interestingly, this is the same intuition that is often used in the analysis of the response time tail when the service distribution is heavy-tailed. This hints that it may be possible to use Theorem 1 to characterize $Pr(T > x)$.

Another important contribution of this paper is the proof technique used to attain Theorem 1. The proof of Theorem 1 makes use of a recently developed stochastic integral for nonhomogeneous Poisson processes. This integral was originally developed for use in analyzing systems with non-stationary load, however we have shown here that the framework is well-suited for the analysis of scheduling policies. In particular, it provides a useful tool for analyzing policies that have a natural branching structure in how work can be assigned, as we saw in the cases of SRPT and PS. It should be evident to the reader that the analysis will extend to many other priority based policies, and a topic of current research is formalizing the branching structure in a policy that will guarantee that Theorem 1 holds.

VI. Acknowledgements

This work effectively began during an extended visit to the EURANDOM Institute at Eindhoven University of Technology in the Netherlands and has benefitted greatly from discussions with Onno Boxma, Richard Boucherie, and Sing-Kong Cheung. Additionally, I would like to thank Michael Shalmon for sharing his insights regarding the branching structure of PS and of PLCFS.

References

[1] N. Bansal and M. Harchol-Balter. Analysis of SRPT scheduling: Investigating unfairness. In *Proc. of ACM Sigmetrics*, 2001.

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

[3] P. Brown. Comparing FB and PS scheduling policies. *Perf. Eval. Rev.*, 34(3):18–20, 2006.

[4] S. K. Cheung, H. van den Berg, and R. J. Boucherie. Decomposing the queue length distribution of processor-sharing models into queue lengths of permanent customer queues. *Perf. Eval.*, 62(1-4):100–116, 2005.

[5] S. Drekic and J. E. Stafford. Symbolic computation of moments in priority queues. *J. on Computing*, 14:261–277, 2002.

[6] N. Duffield, W. Massey, and W. Whitt. A nonstationary offered-load model for packet networks. *Telecommunication Systems*, 16(3,4):271–296, 2001.

[7] H. Feng, V. Misra, and D. Rubenstein. PBS: a unified priority-based cpu scheduler. In *Proc. of ACM Sigmetrics*, 2007.

[8] F. Guillemin, P. Robert, and B. Zwart. Tail asymptotics for processor sharing queues. *Adv. in Appl. Prob.*, 36:525–543, 2004.

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

[10] 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.

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

[12] F. Kelly. *Reversibility and Stochastic Networks*. John Wiley & Sons, 1979.

[13] M. Kendall. *The Advanced Theory of Statistics*. Griffin, London, 1945.

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

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

[16] W. Massey. The analysis of queues with time-varying rates for telecommunication models. *Telecommunication Systems*, 21(2-4):173–204, 2002.

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

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

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

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

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

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

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

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

[25] 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.

[26] M. Shalmon. The GI/GI/1 queue and its variations via the lcfs preemptive resume discipline. *Prob. Eng. Inform. Sciences*, 2:215–230, 1988.

[27] M. Shalmon. Explicit formulas for the variance of conditional sojourn times in M/D/1-PS. *Oper. Res. Letters*, 35:463–466, 2007.

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

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

[30] L. Takács. A single-server queue with poisson input. *Oper. Res.*, 10:388–397, 1962.

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

[32] A. Ward and W. Whitt. Predicting response times in processor-sharing queues. In *Proc. of the Fields Institute Conf. on Comm. Networks*, 2000.

[33] A. Wierman. *Scheduling for today's computer systems: Bridging theory and practice*. PhD thesis, Carnegie Mellon University, 2007.

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

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

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

[37] S. Yang and G. de Veciana. Enhancing both network and user performance for networks supporting best effort traffic. *Trans. on Networking*, 12(2):349–360, 2004.

[38] S. Yashkov. Processor sharing queues: Some progress in analysis. *Queueing Sys.*, 2:1–17, 1987.

[39] S. Yashkov. Mathematical problems in the theory of shared-processor systems. *J. of Soviet Mathematics*, 58:101–147, 1992.

[40] A. Zwart and O. Boxma. Sojourn time asymptotics in the M/G/1 processor sharing queue. *Queueing Sys.*, 35:141–166, 2000.