

Analysis of Fixed Priority SWAP Scheduling Policy for Real-time and Non Real-time Jobs

Michael Okopa
Makerere University
School of Computing and Information Sciences
Kampala, Uganda
okopamike@gmail.com

Tonny Bulega
Makerere University
School of Computing and Information Sciences
Kampala, Uganda
tbulega@cit.mak.ac.ug

Abstract—Scheduling of real-time jobs is challenging given the fact that they are delay sensitive. In order to improve the performance of such jobs, a number of scheduling policies have been proposed. SWAP, a non-preemptive threshold based policy has been suggested to approximate the Shortest Job First (SJF) policy which is a non-preemptive optimal policy. SWAP differentiates between short and large jobs such that short jobs are given precedence over large jobs. However, SWAP cannot guarantee the service quality of real-time jobs which are constrained in terms of deadlines especially under heavy load or congestion. In this paper, we analyze a variant of SWAP called Fixed Priority SWAP policy that can provide service differentiation so as to reduce the delay experienced by real-time jobs without hurting much the performance of non real-time jobs. The service differentiation attribute is based on the time sensitivity of the jobs, where more time sensitive jobs also called real-time jobs are given higher priority over less time sensitive jobs also termed non real-time jobs. We evaluate Fixed Priority SWAP under an M/G/1 queue for job size distributions with varying coefficient of variation (CoV) to analyze the impact of service differentiation on real-time and non real-time jobs. Our detailed numerical analysis reveals that Fixed Priority SWAP policy improves the performance of real-time jobs at the expense of little or no penalty for the non real-time jobs based on the threshold value.

Index Terms—Real-time; non real-time; response time; service differentiation.

I. INTRODUCTION

Service differentiation has been used in a number of networked environments to provide quality of service based on any given attribute. The service differentiation attributes can be selected depending on which traffic the operator defines as more important. For example, real-time traffic can be given higher priority than non real-time traffic to ensure that real-time communications are always reliable, also video traffic can be given higher priority over voice traffic since video traffic is less delay tolerant. Service differentiation is implemented by first classifying jobs by the system and then placing them into different priority queues. Jobs are then scheduled from the head of a higher priority queue, however jobs in the lower priority queues are given service only if all queues of higher priority are empty. The challenge with this is that if the volume of higher priority traffic becomes excessive, lower priority traffic can be starved. In routers this problem is overcome by allowing packets in a higher priority queue to be serviced

before packets in lower priority queues only if the amount of traffic in the high priority queue stays below a user-defined threshold [1].

In an effort to improve the performance of jobs, size based scheduling policies which use a notion of size to decide which job to service next have recently attracted a lot of attention for efficient resource allocation. The key idea behind size based scheduling policy is to favor short jobs while ensuring that large jobs are not starved. The degree to which large jobs suffer depends on the statistical characteristics of the job size distribution, that is how the mass is distributed among short and large jobs. Most size based policies are known to provide short response time for short jobs to the expense of significant penalty to large jobs when analyzed under M/M/1 queuing systems. However, under highly varying job sizes, some recent studies have shown that size-based scheduling policies offer impressive performance [4],[6].

Workloads have been observed to constitute around 99% of short jobs and the 1% which are the largest jobs account for more than 50% of the total amount of workload [3], where the term job is commonly used in scheduling theory to denote a piece of workload that arrives to the system all at once. Coefficient of variation CoV which is defined as the ratio of the standard deviation to the mean of a distribution is used as a common metric to measure the variability of a distribution [4], the higher the CoV value of a distribution the higher the variability of the distribution. Typical examples of CoV observed in the Internet traffic range between 5-20 [3].

The Shortest Job First (SJF), a non-preemptive scheduling policy that serves the job in the system with the smallest original size is found to be optimal [7]. A newly arriving shortest job must wait for the job in service to complete before it can receive service. However, the optimality of SJF depends on a priori knowledge of job service times, which is not available in systems where job service times are only known after the jobs have completed service [8].

Mi *et al.* [8] proposed a general scheduling policy called SWAP that approximates the behavior of the optimal SJF scheduling policy by using workload temporal dependence to forecast job service times without any a priori knowledge of upcoming job demands. SWAP uses a dynamic threshold value to decide which jobs are to be delayed at the end of the queue.

Jobs whose sizes are equal or less than the threshold value are classified as short, otherwise it is classified as large. Large jobs are delayed for a fixed number of times while short jobs are served immediately. The authors used simulations and found that SWAP closely approximates SJF. However, simulation techniques are restrictive in the sense that most often only programmers can comfortably adopt them and also it takes long to obtain results for a wide range of input parameters.

To complement the work on SWAP, Rai *et al.* [9] analytically modeled SWAP to approximate the behavior of the optimal SJF scheduling policy. Through analysis they found that SWAP closely approximates SJF for large jobs under both exponential and Bounded Pareto distributions. While SWAP offers reduced conditional average response time as compared to SJF to large job sizes, it penalizes short jobs more than SJF especially short jobs with smaller sizes under both exponential and Bounded Pareto distributions. However, avoiding a penalty for the largest jobs is a good feature of SWAP because these jobs constitute a large fraction of the total load for job size distributions with high variability property.

Our contribution is two-fold, first we propose a variant of SWAP called Fixed Priority SWAP policy that offers service differentiation so as to improve the performance of real-time jobs without appreciably degrading the performance of non real-time jobs. This is achieved by avoiding the interruption of the service of real-time jobs by non real-time jobs. Secondly, we evaluate Fixed Priority SWAP policy under M/G/1 for job size distributions with varying Cov .

The rest of the paper is organized as follows: in the next section, we present mathematical background that guides model derivation of Fixed Priority SWAP policy. In Section III, we review Fixed Priority SWAP scheduling policy and derive its models. We evaluate Fixed Priority SWAP in Section IV, and finally conclude the paper in Section V.

II. MATHEMATICAL BACKGROUND

Let's denote the variables corresponding to real-time jobs by a superscript RT and for the non real-time jobs by a superscript NRT . Likewise, let the variables corresponding to short and large jobs be represented by subscripts s and l respectively. The size of real-time short and large jobs be referred to as x_s^{RT} and x_l^{RT} respectively. Similarly, the size of non real-time short and large jobs be referred to as x_s^{NRT} and x_l^{NRT} respectively.

We assume that a real-time job arrives at the system with probability p and a non real-time job arrives with probability $(1 - p)$. Let the average job arrival rate to the system be λ , then the average arrival rates of real-time and non real-time jobs are $\lambda^{RT} = p\lambda$ and $\lambda^{NRT} = (1 - p)\lambda$ respectively. A reasonable mean arrival rate of the real-time jobs is almost 30% of the total mean arrival, i.e, $p \leq 30\%$ [4].

Further assume that real-time and non real-time jobs maintain the same distribution as the aggregate of all the jobs in the system. If the probability density function pdf of all job classes is $f(x)$ then $f^{RT}(x) = f^{NRT}(x) = f(x)$.

Given the cumulative distribution function, $F(x) = \int_0^x f(t)dt$,

then the reliability function (or survival function) of x , $F^c(x) = 1 - F(x)$. The load corresponding to real-time jobs of sizes less than or equal to x_{th} , where x_{th} is the job size threshold, is given by $\rho_{x_s}^{RT} = p\rho_{x_s}$, where $\rho_{x_s} = \lambda \int_0^{x_s} tf(t)dt$. The load corresponding to real-time jobs of sizes greater than x_{th} is $\rho_{x_l}^{RT} = p\rho_{x_l}$, where $\rho_{x_l} = \lambda \int_{x_l}^{\infty} tf(t)dt$ and $\rho_{x_l} = (\rho - \rho_{x_s})$, where ρ = the total load in the system. Similarly, the load corresponding to non real-time jobs of sizes less than or equal to x_{th} is $\rho_{x_s}^{NRT} = (1 - p)\rho_{x_s}$ while the load corresponding to non real-time jobs of sizes greater than x_{th} is $\rho_{x_l}^{NRT} = (1 - p)\rho_{x_l}$. Define $\overline{x_s^n} = \int_0^{x_s} t^n f(t)dt$ to be the n^{th} moment of the job size distribution with size less than or equal to x . $\overline{x_s}$ is the first moment (mean) and $\overline{x_s^2}$ is the second moment of the job size distribution with size less than or equal to x_s . It follows that $\overline{x_l^n} = \int_{x_l}^{\infty} t^n f(t)dt$ is the n^{th} moment of the job size distribution with size greater than x . $\overline{x_l}$ is the mean and $\overline{x_l^2}$ is the second moment of the job size distribution with size greater than x .

Denote the system backlog due to real-time short and large jobs as $W(x_s^{RT})$ and $W(x_l^{RT})$ respectively while the system backlog due to non real-time short and large jobs as $W(x_s^{NRT})$ and $W(x_l^{NRT})$. The average waiting time of real-time and non real-time jobs of sizes less than or equal to x_{th} in an M/G/1/FCFS system and with arrival rate λ and load ρ values corresponding to the respective class of the job as given in [10] can be expressed as:

$$W(x_s^{RT}) = \frac{\lambda^{RT} \overline{x_s^2}}{2(1 - \rho_{x_s}^{RT})}; \quad W(x_s^{NRT}) = \frac{\lambda^{NRT} \overline{x_s^2}}{2(1 - \rho_{x_s}^{NRT})} \quad (1)$$

The corresponding average waiting time of real-time and non real-time jobs of size greater than x_{th} in an M/G/1/FCFS system is as follows:

$$W(x_l^{RT}) = \frac{\lambda^{RT} \overline{x_l^2}}{2(1 - \rho_{x_l}^{RT})}; \quad W(x_l^{NRT}) = \frac{\lambda^{NRT} \overline{x_l^2}}{2(1 - \rho_{x_l}^{NRT})} \quad (2)$$

We now define the expression for the conditional average response time under FCFS. An arriving job to a FCFS queue waits for all jobs it finds in the queue upon its arrival. Hence, the conditional average response time of a job of size x in an M/G/1/FCFS system is given as

$$T(x) = x + W(x), \quad (3)$$

where $W(x) = \frac{\lambda x^2}{2(1-\rho)}$ is the mean waiting time due to jobs in the system.

If an arriving job of size x finds only the jobs that are less than or equal to job size x_{th} in the M/G/1/FCFS queue, then its conditional average response time $T(x_{th})$ is given as

$$T(x_{th}) = x + W(x_{th}), \quad (4)$$

where $W(x_{th}) = \frac{\lambda x_{th}^2}{2(1-\rho_{x_{th}})}$. It follows that the conditional average response time of a job of size x greater than x_{th} in an M/G/1/FCFS system is given as

$$T(x_l) = x + W(x_l) \quad (5)$$

where $W(x_l) = \frac{\lambda x_l^2}{2(1-\rho_{x_l})}$. We numerically evaluate Fixed Priority SWAP models under the exponential and Bounded Pareto job size distributions as examples of workloads with varying coefficient of variation. The variability of a job size distribution is determined by its Coefficient of Variation CoV . Exponential distribution has a low variability since its $CoV = 1$, whereas a Bounded Pareto distribution has a high variability ($CoV > 1$). In this paper, we use the exponential distribution and Bounded Pareto $BP(10, 5 * 10^5, 1.1)$ distribution with mean values of 72.7 to numerically evaluate Fixed Priority SWAP models. The probability density function of an exponential distribution is given as:

$$f(x) = \mu e^{-\mu x}, \quad x \geq 0, \quad \mu \geq 0. \quad (6)$$

On the other hand, Bounded Pareto distributions have commonly been used to evaluate the performance of systems under heavy tailed workloads with high variance [6], [4]. We denote the Bounded Pareto distribution by $BP(k, P, \alpha)$ where k and P are the minimum and the maximum job sizes and α is the exponent of the power law. The pdf of the Pareto distribution is given as:

$$f(x) = \frac{\alpha k^\alpha}{1 - (k/P)^\alpha} x^{-\alpha-1} \quad k \leq x \leq P, \quad 0 \leq \alpha \leq 2. \quad (7)$$

The Pareto distributions that emerge in computer system applications typically have $\alpha \in (0.9; 1.3)$ [4].

In the next section, we discuss Fixed Priority SWAP scheduling and derive its models.

III. FIXED PRIORITY SWAP SCHEDULING MODELS

A. A review of Fixed Priority SWAP Scheduling

Fixed Priority SWAP is a two decision scheduling policy where the first decision is taken to give high or low priority to the jobs depending on the time sensitivity of the job. The second decision is taken to classify the jobs as short or large depending on the threshold value. The jobs in each priority class are then serviced in a SWAP order. In this particular case, jobs are served in a FCFS order until a large job is serviced when the entire queue is scanned and the size of each job identified. Short jobs are placed at the head of the queue before service resumes again. Low priority jobs which in this case are the non real-time jobs are serviced only if there are no high priority (real-time) jobs, i.e., all queues that correspond to real-time jobs are empty. The service of the non real-time jobs are not preempted on the arrival of the real-time jobs. The main goal of Fixed Priority SWAP scheduling policy in this case is to offer differentiated service between real-time and non real-time jobs. Fixed Priority SWAP scheduling policy improves the service of real-time jobs by avoiding the interruptions of the service of real-time jobs by non real-time jobs.

B. Fixed Priority SWAP(1)

We specifically focus on Fixed Priority (1) and SWAP(1) where classified large jobs are delayed once before receiving service. We model Fixed Priority(1) with each priority class having two queues as shown in Figure 1. In each priority class,

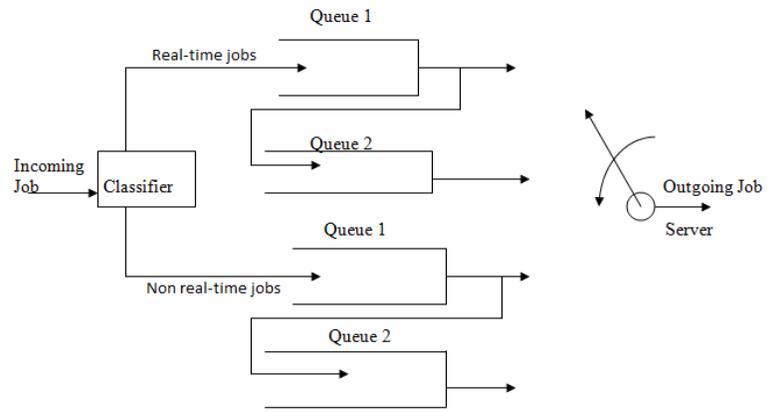


Fig. 1. Fixed Priority SWAP(1)

short jobs are serviced in queue 1 while scanned (delayed) large jobs are serviced in queue 2. Furthermore, in each priority class, all incoming jobs first arrive at queue 1 where they are serviced in FCFS order while comparing the service given to each job to the threshold value. If the job's service time is less than the threshold value, this job is classified as short otherwise its classified as large. If a large job is serviced, the entire queue is scanned and the size of each job in the queue is predicted. Large jobs are delayed once and placed in queue 2. After servicing scanned short jobs in queue 1 using FCFS order, the server then gives service to delayed large jobs in queue 2 also using FCFS order of service. The server then returns to the un scanned state again and serves incoming jobs using FCFS until a large job is served when the entire queue is scanned again. The state where served jobs in the queue are not scanned is called an un scanned state of the queue. Non real-time jobs are only served when the queues of real-time jobs are empty and a non real-time job is not preempted on the arrival of real-time jobs.

C. Models for arrivals at un scanned state

1) *Real-time jobs under Fixed Priority SWAP(1)*: Consider the arrival of a tagged real-time job to a Fixed Priority SWAP(1) queue in an un scanned state but just before a large job which triggers the scanning of the real-time queue is serviced. This tagged job will experience a conditional average response time which is the same as its conditional average response time under an isolated SWAP system [9]. If the tagged job is a real-time short job, it will be delayed by the mean residual life of the real-time large job that triggered the scanning of the queue and will then wait for all scanned real-time short jobs it finds in the queue before it receives service. The waiting time of the tagged job due to these jobs is given by $W(x_s^{RT})$ defined in equation 1. However, if the tagged job is a real-time large job, it will be delayed by the mean residual life of the real-time large job that triggered the scanning of the queue and also by the scanned real-time short and large jobs. The waiting time of the tagged job due to these jobs is given by $W(x_l^{RT})$ defined in equation 2. The resulting conditional

average response times under Fixed Priority SWAP(1) for real-time jobs at an scanned state are given as:

$$T(x) = \begin{cases} x + W(x_s^{RT}), & x \leq x_{th} \\ x + W(x_s^{RT}) + W(x_l^{RT}), & x > x_{th} \end{cases} \quad (8)$$

2) *Non real-time jobs under Fixed Priority SWAP(1):*

Assume an isolated non real-time Fixed SWAP(1) system. The mean waiting time of the non real-time short job, x_s^{NRT} in this isolated system is the same as the mean waiting time in a SWAP(1) system with mean arrival rate and load of λ^{NRT} and ρ^{NRT} respectively. We denote this waiting time by $W(x_s^{NRT})$. In the Fixed Priority SWAP(1) policy however, the tagged non real-time job will be delayed further by the service of the backlog due to short and large jobs that it finds in the real-time Fixed Priority SWAP(1) system upon its arrival denoted by $W(x_s^{RT})$ and $W(x_l^{RT})$ respectively. In addition, during its stay in the system, there will be on the average $\lambda^{RT}T(x)$ new arrivals of real-time jobs, each of which will delay it by an average of x_s^{RT} if the real-time job is short and x_l^{RT} if the real-time job is large and its service time x . $T(x)$ is the average time spent in the system by the non real-time job.

Similarly, a tagged non real-time large job will be delayed by the service of the backlog that it finds in the non real-time queue due to both short and large jobs denoted by $W(x_s^{NRT})$, and $W(x_l^{NRT})$ respectively. Furthermore, the tagged non real-time large job will be delayed by the service of the backlog that it finds in the real-time Fixed Priority SWAP(1) system upon its arrival, $W(x_s^{RT})$ and $W(x_l^{RT})$. In addition, during its stay in the system, there will be on the average $\lambda^{RT}T(x)$ new arrivals of real-time jobs, each of which will delay it by an average of x_s^{RT} if the real-time job is short and x_l^{RT} if the real-time job is large and its service time x .

The resulting conditional average response time of non real-time jobs under Fixed Priority SWAP(1) at an scanned state are given as:

$$T(x) = \begin{cases} x + \frac{W(x_s^{NRT}) + W(x_s^{RT}) + W(x_l^{RT})}{(1 - \rho_{x^{RT}})}, & x \leq x_{th} \\ x + \frac{W(x_s^{NRT}) + W(x_l^{NRT}) + W(x_s^{RT}) + W(x_l^{RT})}{(1 - \rho_{x^{RT}})}, & x > x_{th} \end{cases} \quad (9)$$

where $\rho_{x^{RT}} = \rho_{x_s^{RT}} + \rho_{x_l^{RT}}$.

D. Models for arrivals at scanned state

1) *Real-time jobs under Fixed Priority SWAP(1):* We consider the arrival of a tagged real-time job to a scanned queue. The average response time of a real-time job under Fixed Priority SWAP(1) in the scanned state is the same as the average response time under SWAP(1) policy with the mean arrival rate λ^{RT} and at load ρ^{RT} at scanned state. Hence, the expression for the average response time of a real-time job under Fixed Priority SWAP(1) is similar to the conditional average response time under SWAP(1) given in [9]. The resulting conditional average response times of real-time jobs under Fixed Priority SWAP(1) in the scanned state are given

as:

$$T(x) = \begin{cases} x + 3W(x_s^{RT})/2 + W(x_l^{RT}) + \overline{F}F^c(x_l^{RT}), & x \leq x_{th} \\ x + 3W(x_s^{RT})/2 + 2W(x_l^{RT}) + \overline{F}F^c(x_l^{RT}), & x > x_{th} \end{cases} \quad (10)$$

2) *Non real-time jobs under Fixed Priority SWAP(1):*

Consider the arrival of a tagged non real-time job to a Fixed Priority SWAP(1) system in the scanned state. Assume an isolated non real-time Fixed SWAP(1) system. The mean waiting time of the non real-time job x^{NRT} in this isolated system is the same as the mean waiting time in a SWAP(1) system with appropriate mean arrival rate and load of λ^{NRT} and ρ^{NRT} respectively. In other words, under the Fixed Priority SWAP(1) policy, the tagged non real-time short job will be delayed by the service of the backlog due to scanned short jobs $3W(x_s^{RT})/2$, scanned large jobs $W(x_l^{RT})$ that it finds in the real-time Fixed Priority SWAP(1) system upon its arrival, including at least one large un scanned job, $\overline{F}F^c(x_l^{RT})$. This tagged non real-time short job will additionally be delayed by large jobs that were newly scanned along with real-time tagged large jobs by a mean waiting time of $W(x_l^{RT})$. In addition, during its stay in the system, there will be on the average $\lambda^{RT}T(x)$ new arrivals of real-time jobs, each of which will delay it by an average of x_s^{RT} if the new arrival is a real-time short job and x_l^{RT} if the new arrival is a real-time large job and its service time x .

Likewise, the tagged non real-time large job will be delayed further by non real-time large jobs that were newly scanned along itself by a mean waiting time of $W(x_l^{NRT})$.

The resulting average response time of non real-time jobs under Fixed Priority SWAP(1) at scanned state is given as:

$$T(x) = \begin{cases} x + \frac{3W(x_s^{NRT})/2 + W(x_l^{NRT}) + \overline{F}F^c(x_l^{NRT})}{(1 - \rho_{x^{RT}})} + \frac{3W(x_s^{RT})/2 + 2W(x_l^{RT}) + \overline{F}F^c(x_l^{RT})}{(1 - \rho_{x^{RT}})}, & \text{for } x \leq x_{th} \\ x + \frac{3W(x_s^{NRT})/2 + 2W(x_l^{NRT}) + \overline{F}F^c(x_l^{NRT})}{(1 - \rho_{x^{RT}})} + \frac{3W(x_s^{RT})/2 + 2W(x_l^{RT}) + \overline{F}F^c(x_l^{RT})}{(1 - \rho_{x^{RT}})}, & \text{for } x > x_{th} \end{cases} \quad (11)$$

In the next section, we present numerical results showing the performance of Fixed Priority SWAP(1) policy and its comparison with SWAP(1) scheduling policy.

IV. PERFORMANCE EVALUATION

In this section, we use the derived Fixed Priority SWAP(1) models to evaluate its performance. We specifically focus on how Fixed Priority SWAP(1) improves the mean response time of real time jobs, and the penalty experienced by non-real time jobs due to the service of real time jobs. We also investigate the impact of threshold values x_{th} on the performance of Fixed Priority SWAP(1). We use exponential and Bounded Pareto distributions presented in section II at low load value $\rho = 0.5$ and high load value of $\rho = 0.9$ to mitigate workloads with varying coefficient of distributions. Due to space limitations, we numerically evaluate Fixed Priority SWAP(1) models for

jobs that arrive to the scanned state only which we derived in Section III-D.

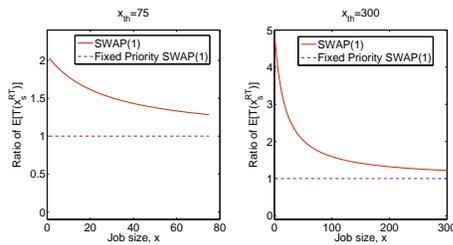


Fig. 2. Ratio of $E[T(x)]$ under SWAP(1) to $E[T(x_s^{RT})]$ under Fixed Priority SWAP(1) of real-time short jobs in the scanned state as a function of job size for $Exp(\frac{1}{\mu} = 72.7)$, at load $\rho = 0.5$ and $p = 0.3$ when the thresholds are $x_{th} = 75$ and $x_{th} = 300$.

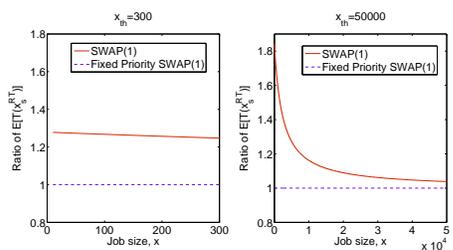


Fig. 3. Ratio of $E[T(x)]$ under SWAP(1) to $E[T(x_s^{RT})]$ under Fixed Priority SWAP(1) of real-time short jobs in the scanned state as a function of job size for BP(10, $5 * 10^5$, 1.1), at load $\rho = 0.5$ and $p = 0.3$ when the thresholds are $x_{th} = 300$ and $x_{th} = 50000$.

Figure 2 shows the ratio of the average response time of real-time jobs under SWAP(1) to their average response time under Fixed Priority SWAP(1) for the exponential distribution at a low load of $\rho = 0.5$. We observe that SWAP(1) reaches as high as 2 at a low threshold value of $x_{th} = 75$, which means that the maximum average response time under SWAP(1) is 2 times higher than the average response time under Fixed Priority SWAP(1). We further observe that at a high threshold value of $x_{th} = 300$, the ratio of the average response time of real-time jobs under SWAP(1) to their average response time under Fixed Priority SWAP(1) reaches as high as above 4. Fixed priority SWAP(1) offers a better performance to real-time jobs compared to SWAP(1) due to the fact that Fixed Priority SWAP(1) avoids the interruption of the service of the real-time jobs by non real-time jobs. The performance difference between SWAP(1) and Fixed Priority SWAP(1) in terms of reducing the average response time of real-time jobs is less as the job sizes increase.

We now look at the improvement of Fixed Priority SWAP(1) in terms of reducing the average response time of real-time jobs for the BP distribution considered at a low load of $\rho = 0.5$. This is shown in Figure 3 where we plot the ratio of the average response time between the policies. We see that, while SWAP(1) reaches as high as above 1.2 at low threshold values, SWAP(1) reaches as high as above 1.6 at high threshold values. This still confirms the fact that Fixed Priority SWAP(1) reduces the average response time of real-time jobs compared

to SWAP(1) especially at high threshold values.

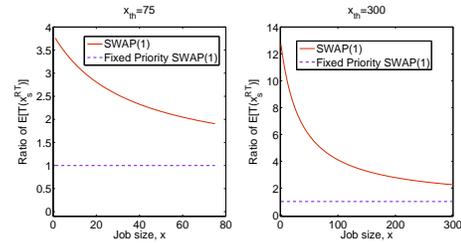


Fig. 4. Ratio of $E[T(x)]$ under SWAP(1) to $E[T(x_s^{RT})]$ under Fixed Priority SWAP(1) of real-time short jobs in the scanned state as a function of job size for $Exp(\frac{1}{\mu} = 72.7)$, at load $\rho = 0.9$ and $p = 0.3$ when the thresholds are $x_{th} = 75$ and $x_{th} = 300$.

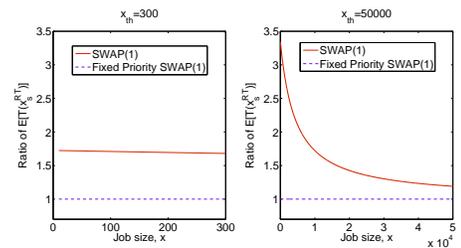


Fig. 5. Ratio of $E[T(x)]$ under SWAP(1) to $E[T(x_s^{RT})]$ under Fixed Priority SWAP(1) of real-time short jobs in the scanned state as a function of job size for BP(10, $5 * 10^5$, 1.1), at load $\rho = 0.9$ and $p = 0.3$ when the thresholds are $x_{th} = 300$ and $x_{th} = 50000$.

Figure 4 shows the ratio of $E[T(x)]$ under SWAP(1) to $E[T(x_s^{RT})]$ under Fixed Priority SWAP(1) of real-time short jobs in the scanned state at high load $\rho = 0.9$ for exponentially distributed workloads, it can be observed that real-time short jobs experience lower average response time under Fixed Priority SWAP(1) than under SWAP(1) regardless of the threshold value. This is noticed where at low threshold values, SWAP(1) reaches as high as above 3.5 while at high threshold values SWAP(1) reaches as high as above 12. Again the performance of Fixed Priority SWAP(1) at high load is much better at high threshold values just like at low load.

Similarly, real-time short jobs experience lower average response time under Fixed Priority SWAP(1) than under SWAP(1) for Bounded Pareto distribution for all threshold values at high load $\rho = 0.9$, as seen in Figure 5. This is observed where at low threshold values, SWAP(1) reaches as high as above 1.5 while at high threshold values SWAP(1) reaches as high as above 3.

Figure 6 compares the ratio of the average response time of real-time large jobs under SWAP(1) to their average response time under Fixed Priority SWAP(1) for the exponential distribution at a low load of $\rho = 0.5$. We see that for both considered threshold values, Fixed Priority SWAP(1) offers a lower average response time to real-time large jobs in the scanned state as compared to SWAP(1). However, at large job sizes there is no difference between the average response time offered by SWAP(1) and Fixed Priority SWAP(1) to real-time large jobs.

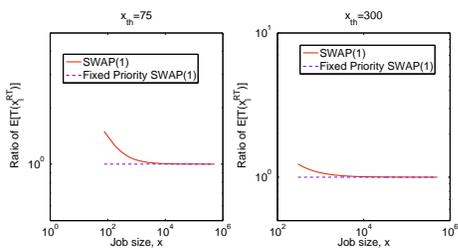


Fig. 6. Ratio of $E[T(x)]$ under SWAP(1) to $E[T(x_l^{RT})]$ under Fixed Priority SWAP(1) of real-time large jobs in the scanned state as a function of job size for $Exp(\frac{1}{\mu} = 72.7)$, at load $\rho = 0.5$ and $p = 0.3$ when the thresholds are $x_{th} = 75$ and $x_{th} = 300$.

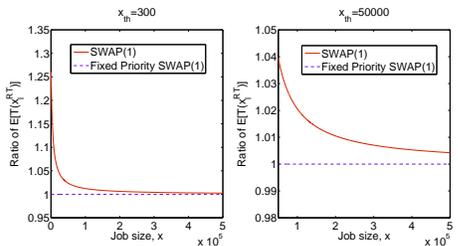


Fig. 7. Ratio of $E[T(x)]$ under SWAP(1) to $E[T(x_l^{RT})]$ under Fixed Priority SWAP(1) of real-time large jobs in the scanned state as a function of job size for BP(10, $5 * 10^5, 1.1$), at load $\rho = 0.5$ and $p = 0.3$ when the thresholds are $x_{th} = 300$ and $x_{th} = 50000$.

Similarly, Figure 7 shows results of Fixed Priority SWAP(1) in comparison to SWAP(1) under the Bounded Pareto distribution at low load $\rho = 0.5$ and at scanned state. It can be observed from the figure that Fixed Priority SWAP(1) offers a lower average response time to real-time large jobs regardless of the threshold values. Unlike at low threshold values where SWAP(1) and Fixed Priority SWAP(1) offer the same average response time to the largest large real-time jobs, at high threshold values Fixed Priority SWAP(1) offers a lower average response time to the largest large real-time jobs as compared to SWAP(1).

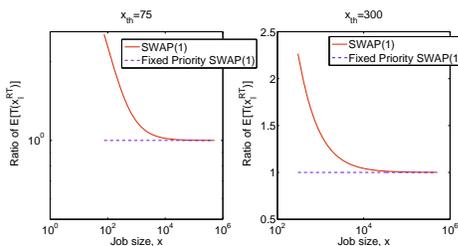


Fig. 8. Ratio of $E[T(x)]$ under SWAP(1) to $E[T(x_l^{RT})]$ under Fixed Priority SWAP(1) of real-time large jobs in the scanned state as a function of job size for $Exp(\frac{1}{\mu} = 72.7)$, at load $\rho = 0.9$ and $p = 0.3$ when the thresholds are $x_{th} = 75$ and $x_{th} = 300$.

Figures 8 and 9 show the performance of real-time large jobs at high load $\rho = 0.9$. We observe from Figure 8 that for the exponential distribution, the performance of real-time large jobs in the scanned state under Fixed Priority SWAP(1) is better than under SWAP(1) at all threshold values. It can also be noted that Fixed Priority SWAP(1) reduces the average

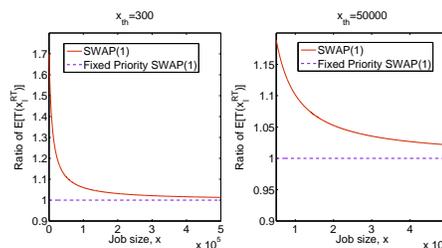


Fig. 9. Ratio of $E[T(x)]$ under SWAP(1) to $E[T(x_l^{RT})]$ under Fixed Priority SWAP(1) of real-time large jobs in the scanned state as a function of job size for BP(10, $5 * 10^5, 1.1$), at load $\rho = 0.9$ and $p = 0.3$ when the thresholds are $x_{th} = 300$ and $x_{th} = 50000$.

response time of real-time large jobs more at high threshold values. Mean while, the largest large real-time jobs experience the same average response time under both SWAP(1) and Fixed Priority SWAP(1).

For the BP distribution (see Figure 9), we observe that Fixed Priority SWAP(1) performs better than SWAP(1) at all considered threshold values. It is noticed that at low threshold values, SWAP(1) reaches as far as above 1.5 and at high threshold values SWAP(1) reaches as far as 1.15. For both considered threshold values, the performance of real-time large jobs under SWAP(1) and Fixed Priority SWAP(1) are closer as job sizes increase.

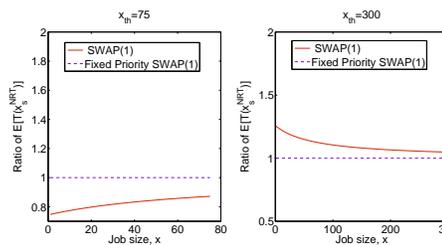


Fig. 10. Ratio of $E[T(x)]$ under SWAP(1) to $E[T(x_s^{NRT})]$ under Fixed Priority SWAP(1) of Non real-time short jobs in the scanned state as a function of job size for $Exp(\frac{1}{\mu} = 72.7)$, at load $\rho = 0.5$ and $p = 0.3$ when the thresholds are $x_{th} = 75$ and $x_{th} = 300$.

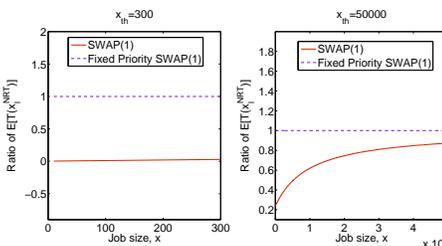


Fig. 11. Ratio of $E[T(x)]$ under SWAP(1) to $E[T(x_s^{NRT})]$ under Fixed Priority SWAP(1) of Non real-time short jobs in the scanned state as a function of job size for BP(10, $5 * 10^5, 1.1$), at load $\rho = 0.5$ and $p = 0.3$ when the thresholds are $x_{th} = 300$ and $x_{th} = 50000$.

Figures 10 and 11 show the performance of non real-time short jobs at low load $\rho = 0.5$. We observe from Figure 10 that for exponentially distributed job sizes, the performance of non real-time short jobs in the scanned state under Fixed Priority

SWAP(1) is worse than under SWAP(1) at low threshold values. However, at high threshold values non real-time short jobs see an improvement in average response time under Fixed Priority SWAP(1) compared to SWAP(1). We observe from Figure 11 that for the BP distribution, non real-time short jobs perform worse under Fixed Priority SWAP(1) compared to SWAP(1) at all considered threshold values. We further note that the performance of non real-time short jobs under Fixed Priority SWAP(1) and SWAP(1) are closer at high threshold values and for the largest non real-time short jobs.

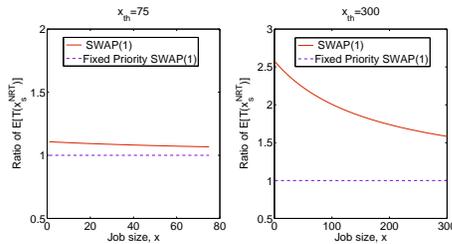


Fig. 12. Ratio of $E[T(x)]$ under SWAP(1) to $E[T(x_s^{NRT})]$ under Fixed Priority SWAP(1) of Non real-time short jobs in the scanned state as a function of job size for $Exp(\frac{1}{\mu} = 72.7)$, at load $\rho = 0.9$ and $p = 0.3$ when the thresholds are $x_{th} = 75$ and $x_{th} = 300$.

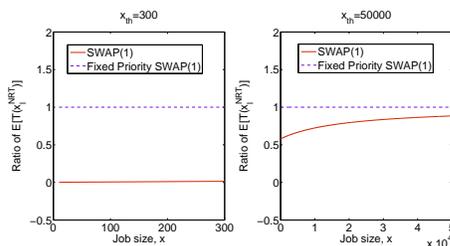


Fig. 13. Ratio of $E[T(x)]$ under SWAP(1) to $E[T(x_s^{NRT})]$ under Fixed Priority SWAP(1) of Non real-time short jobs in the scanned state as a function of job size for BP(10, $5 * 10^5, 1.1$), at load $\rho = 0.9$ and $p = 0.3$ when the thresholds are $x_{th} = 300$ and $x_{th} = 50000$.

Figure 12 shows the ratio of the average response time of non real-time short jobs under SWAP(1) to their average response time under Fixed Priority SWAP(1) for exponentially distributed job sizes at a high load of $\rho = 0.9$. We observe that SWAP(1) reaches as high as above 1 at a low threshold value of $x_{th} = 75$, while at a high threshold value of $x_{th} = 300$, the ratio of the average response time of non real-time jobs under SWAP(1) to their average response time under Fixed Priority SWAP(1) reaches as high as above 2.5. Therefore, Fixed priority SWAP(1) still offers a better performance to non real-time jobs compared to SWAP(1). We note that the performance of SWAP(1) and Fixed Priority SWAP(1) with respect to non real-time jobs are closer at low threshold values compared to high threshold values.

Figure 13 shows the ratio of the average response time of non real-time short jobs under SWAP(1) to their average response time under Fixed Priority SWAP(1) for the BP distribution at high load of $\rho = 0.9$. We observe that Fixed Priority SWAP(1) offers a higher average response time to non

real-time short jobs compared to SWAP(1) at all considered threshold values. However, the average response time of non real-time short jobs under SWAP(1) and Fixed Priority SWAP(1) are closer at high threshold values.

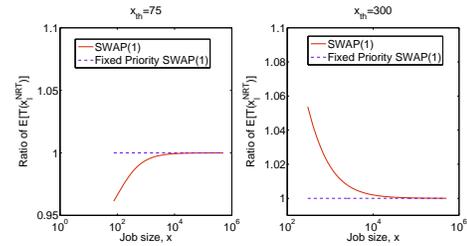


Fig. 14. Ratio of $E[T(x)]$ under SWAP(1) to $E[T(x_s^{NRT})]$ under Fixed Priority SWAP(1) of Non real-time large jobs in the scanned state as a function of job size for $Exp(\frac{1}{\mu} = 72.7)$, at load $\rho = 0.5$ and $p = 0.3$ when the thresholds are $x_{th} = 75$ and $x_{th} = 300$.

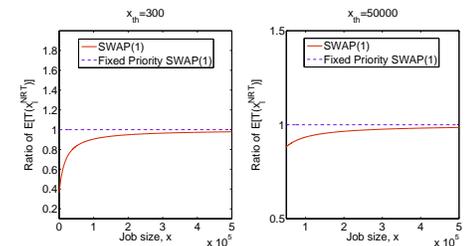


Fig. 15. Ratio of $E[T(x)]$ under SWAP(1) to $E[T(x_s^{NRT})]$ under Fixed Priority SWAP(1) of Non real-time large jobs in the scanned state as a function of job size for BP(10, $5 * 10^5, 1.1$), at load $\rho = 0.5$ and $p = 0.3$ when the thresholds are $x_{th} = 300$ and $x_{th} = 50000$.

Figures 14 and 15 show the performance of non real-time large jobs at low load $\rho = 0.5$. We observe from Figure 14 that for exponentially distributed job sizes, the performance of non real-time large jobs in the scanned state under Fixed Priority SWAP(1) is worse than under SWAP(1) at low threshold values. However, at high threshold values non real-time large jobs experience an improvement in their average response time under Fixed Priority SWAP(1) compared to SWAP(1).

We observe from Figure 15 that for the BP distribution, non real-time large jobs perform worse under Fixed Priority SWAP(1) compared to SWAP(1) at all considered threshold values. We further observe that the performance of non real-time large jobs under Fixed Priority SWAP(1) and SWAP(1) are closer for the largest non real-time large jobs at both low and high threshold values.

Figures 16 and 17 show the performance of non real-time large jobs at high load $\rho = 0.9$. We observe from Figure 16 that for the exponential distribution, the performance of non real-time large jobs in the scanned state under Fixed Priority SWAP(1) is better than under SWAP(1) at both considered threshold values. It is further observed that SWAP(1) reaches as far as about 1.3 at low threshold while at high threshold SWAP(1) reaches as far as about 1.6, showing that Fixed Priority SWAP(1) performs even better at high threshold values. We observe from Figure 17 that for the BP distribution at high load

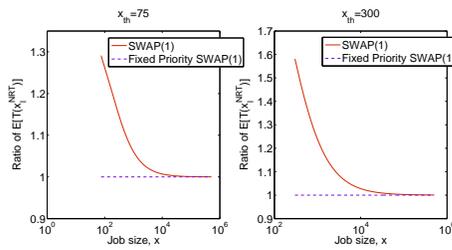


Fig. 16. Ratio of $E[T(x)]$ under SWAP(1) to $E[T(x_s^{NRT})]$ under Fixed Priority SWAP(1) of Non real-time large jobs in the scanned state as a function of job size for $Exp(\frac{1}{\mu} = 72.7)$, at load $\rho = 0.9$ and $p = 0.3$ when the thresholds are $x_{th} = 75$ and $x_{th} = 300$.

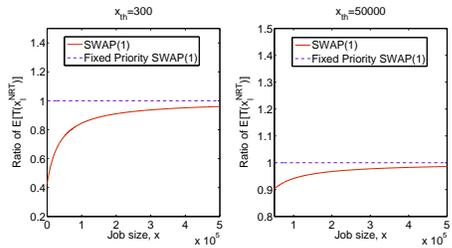


Fig. 17. Ratio of $E[T(x)]$ under SWAP(1) to $E[T(x_s^{NRT})]$ under Fixed Priority SWAP(1) of Non real-time large jobs in the scanned state as a function of job size for $BP(10, 5 * 10^5, 1.1)$, at load $\rho = 0.9$ and $p = 0.3$ when the thresholds are $x_{th} = 300$ and $x_{th} = 50000$.

$\rho = 0.9$, non real-time large jobs perform worse under Fixed Priority SWAP(1) compared to SWAP(1) at all considered threshold values. We observe further that the performance of non real-time large jobs under Fixed Priority SWAP(1) and SWAP(1) are closer for the largest non real-time large jobs at both threshold values but even closer at high threshold values.

V. CONCLUSION

We analysed a variant of SWAP(1) policy called Fixed Priority SWAP(1) scheduling policy under workload with varying distributions so as to improve the average response time of real-time jobs without hurting much the non real-time jobs. The numerical results obtained from the derived models show that Fixed Priority SWAP(1) improves the performance of real-time jobs without hurting much the non real-time jobs. We further observe that the performance of non real-time jobs under Fixed Priority SWAP(1) and SWAP(1) are generally closer at high threshold values regardless of the distribution. However, the potential bottleneck to the implementation of Fixed priority SWAP(1) as compared to SWAP(1) policy is the computational overhead necessary to maintain the priority queues. In this paper, we presented numerical results of Fixed Priority SWAP(1) when classified large jobs are delayed once at scanned state only due to space limitations. In the future, we will numerically investigate Fixed Priority SWAP(1) at un scanned state as well. We will also validate these models using simulations.

REFERENCES

[1] C. Semeria, "Supporting Differentiated Service Classes: Queue Scheduling Disciplines," Juniper Networks, Inc., 2001.

[2] W. Gong, Y. Liu, V. Misra, and D. Towsley, "On the Tails of Web File Size Distributions," in *Proc. 39-th Allerton Conference on Communication, Control, and Computing*, Boulder, Colorado, Oct. 2001.

[3] M. Crovella, R. Frangioso, and M. Harchol-Balter, "Connection Scheduling in Web Servers," in *Proc. USENIX Symposium on Internet Technologies and Systems (USITS 99)*, Boulder, Colorado, Oct. 1999, pp. 243–254.

[4] I. A. Rai, G. Urvoy-Keller, and E. Biersack, "Analysis of LAS Scheduling for Job Size Distributions with High Variance," in *Proc. ACM SIGMETRICS*, Jun. 2003, pp. 218–228.

[5] N. Bansal, "On the average sojourn time under M/M/1/SRPT," *Operations Research Letters*, pp. 195–200, 2005, 33(2).

[6] N. Bansal and M. Harchol-Balter, "Analysis of SRPT Scheduling: Investigating Unfairness," in *Proc. Sigmetrics 2001 / Performance 2001*, Jun. 2001, pp. 279–290.

[7] A. Wierman, "Scheduling for todays computer systems: Bridging theory and practice," Carnegie Mellon University, 2007, PhD thesis.

[8] N. Mi, G. Casale, and E. Smirni, "Scheduling for Performance and Availability in Systems with Temporal Dependent Workloads," in *Proc. The International Conference on Dependable Systems and Networks*, 2008, pp. 336–345.

[9] I. A. Rai and M. Okopa, "Modeling and Evaluation of SWAP Scheduling policy Under Varying Job Size Distributions," in *Proc. The Tenth International Conference on Networks, IARIA, 2011*, Jan. 2011, pp. 115–120.

[10] L. Kleinrock, *Queueing Systems, Volume I & II. Computer Applications*. John Wiley & Sons, 1976.