Performance and Stability Analysis of the Task Assignment based on Guessing Size Routing Policy

Eitan Bachmat
Department of Computer Science
Ben-Gurion University
Beer-Sheva, Israel, 84105.
ebachmat@cs.bgu.ac.il

Josu Doncel
Department of Applied Mathematics,
Statistics and Operations Research
University of the Basque Country, UPV/EHU
Leioa, Spain. 48940
josu.doncel@ehu.eus

Hagit Sarfati
Department of Industrial Engineering
Ben-Gurion University
Beer-Sheva, Israel, 84105.
hagitb@post.bgu.ac.il

***Abstract*—In a system formed by parallel servers and one dispatcher, we study the Task Assignment based on Guessing Size (TAGS) policy, an open loop task assignment policy where jobs are non-preemptive, servers are First-Come-First-Served and the size of incoming jobs is not known. This policy works as follows: all the incoming jobs are routed to the first server and jobs that complete service before s_1 units of time leave the system, but jobs that do not complete service before s_1 are killed and they are routed to the second server, where the service starts from scratch. Likewise, jobs that are executed in server i, if they complete service before s_i units of time, leave the system, whereas jobs that do not complete service before s_i units of time are killed and routed to the next server. For an arbitrary job size distribution, we provide a necessary and sufficient condition for the stability of a system operating under the TAGS policy. We also analyze the performance of the optimal TAGS policy, i.e., when the cutoffs $s_1, s_2, \dots$ are chosen to minimize the waiting time of jobs for an arbitrary job size distribution and we show that it is lower bounded by the performance of the TAGS policy where the maximum queue length is minimized divided by the number of servers minus one. For Bounded Pareto distributed job sizes, we consider the asymptotic regime where the largest job size tends to infinity and we show that, when the system load is less than one, the performance of the optimal TAGS policy is, at most, two times worst than the performance of the optimal SITA policy, which a routing policy where the size of jobs is known. This result shows that the penalty caused by not knowing the size of incoming jobs is upper bounded by a factor of 2. For a higher system load, we show that the order of magnitude of the performance of the optimal TAGS policy in the asymptotic regime depends on the number of spare servers, i.e., the difference between the number of servers in the system and the minimum number of servers to stabilize the system. According to our numerical experiments, when the largest job size is finite, the difference on the performance between the TAGS policy and the SITA policy can be extremely large when the system load is higher than one, whereas it is small when the system load is less than one.**

***Index Terms*—Queueing theory, Parallel Servers, Heavy-tailed distributions.**

I. Introduction

We consider a system formed by parallel servers and a single dispatcher that handles all the incoming traffic to the system. The performance of this kind of systems is clearly affected by the routing policy that is implemented, i.e., how incoming jobs are routed to the servers. The challenge for the designers of these systems is to perform this routing optimally, that is, to assign tasks to the servers in order to optimize a given performance function, such as the mean waiting time or the mean number of customers in the system. The question of which routing policy is optimal is still open for many models.

We study a system with h servers operating under the Task Assignment based on Guessing Size (TAGS) routing policy [13]. In this policy, service time distribution is divided in intervals determined by $h-1$ cutoffs $s_1, s_2, \dots, s_{h-1}$. Hence, all the incoming jobs are routed to the first server and jobs that complete service before s_1 units of time leave the system. On the other hand, jobs that do not complete service before s_1 units are killed and they are routed to the second server, where the service starts from scratch. In server i, jobs that complete service before s_i units of time leave the system, whereas jobs that do not complete service before s_i units of time are killed and routed to the next server. An illustration of a system with 4 servers operating under the TAGS policy is presented in Figure 1.

We note that, in the system that operates under the TAGS policy, jobs are non-preemptive, i.e., jobs are run-to-completion. In fact, in distributed data centers, jobs are submitted to a single server and, since the memory requirement of jobs is so huge, that run-to-completion is preferable to time-sharing [9]. Examples of distributed server system in which jobs are non-preemptive are given in Table 1 of [13]. Besides, the servers are First-Come-First-Served (FCFS), which is a common model, for example, in super-computing systems [20].

A system operating under the TAGS policy has several advantages with respect to other policies of the literature. First, the service time requirement of incoming tasks is not known for TAGS. This is an important difference with respect to the SITA policies [15], where the jobs of incoming tasks is known. Besides, the TAGS policy is an open-loop policy, i.e., it does not require to know the state of the servers to route jobs and, therefore, communication between the servers and the dispatcher is not needed. We remark that this is an important advantage with respect to very popular policies such as Join-the-Shortest-Queue or Power-of-Two. Lastly, it is known that

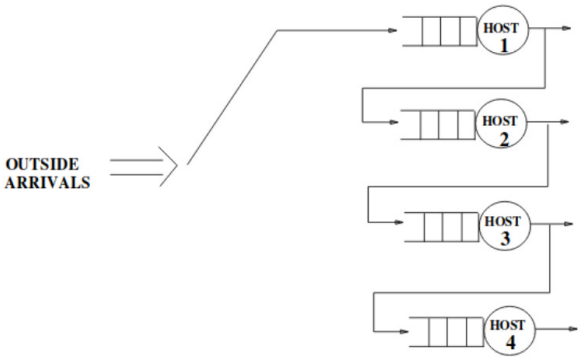

Fig. 1. A system with 4 servers operating under the TAGS policy

the job sizes distribution in data-centers is heavy-tailed, i.e., a small fraction of jobs consists of the half of the load [23]. Indeed, the author in [13] shows that the improvement of the performance of a system operating under the TAGS policies with respect to the performance of a system operating under other policies such as the Least-Work-Left is larger when the job size distribution is more heavy-tailed.

The main contributions of this work are the following:

- We first analyze the stability of a system operating under the TAGS policy. For a given job size distribution, we show that a necessary and sufficient condition for the stability is that the system load is smaller than a critical value. We show that there is an upper-bound for the critical value for any job size distribution and we present the expression of the critical value for the Bounded Pareto job size distribution. Besides, we provide a job size distribution where the critical value coincides with the upper-bound.
- We study the performance of the optimal TAGS policy, that is, the performance when the cutoffs $s_1, \ldots, s_{h-1}$ are chosen so as to minimize the mean waiting time of jobs in the system. We show that the optimal performance and the performance obtained when the maximum mean queue length of the servers is minimized are related. From the obtained expression, we conclude that the former is lower bounded by the latter divided by the number of servers minus one. We also show that these results do not require to assume Poisson arrivals from outside.
- We consider the asymptotic regime where the largest job size tends to infinity and we assume that the job size distribution is Bounded Pareto. For this instance, we first compare the performance of a system operating under the TAGS policy with the performance of a system operating under the SITA policy where the cutoffs minimize the mean waiting time of jobs. We show that the performance of the TAGS policy is at most two times the performance of the SITA in the asymptotic regime and the load of the system is less than one. This implies that, for that case, the penalty for not knowing the sizes of the incoming tasks is upper bounded by 2. For higher loads, it is known from [13] that a system operating under the TAGS policy performs poorly. Besides, there are instances where the stability condition is not satisfied for TAGS and, as a result, the performance comparison between SITA and TAGS cannot be done for a general case. However, we provide an expression of the order of magnitude of performance of a system operating under the TAGS policy in the asymptotic regime which depends on the number of spare servers, i.e., the difference between the number of servers in the system and the minimum number of servers to stabilize the system. We also study several extensions of these asymptotic results. First, we show that they can be extended to heterogeneous servers and for a variant of the TAGS policy where jobs do not start from scratch when they are routed to the next server. Finally, we present a policy, that we call T+W, where some of the servers operate under the TAGS policy and the others under a work-conserving policy such as Least-Work-Left or random assignment.
- We analyze numerically the approximation proposed in [13], where it is assumed that the arrivals to all the servers are Poisson. Our experiments validate the accuracy of this approximation and also show that the approximation over-estimates the mean waiting time of jobs. We also study the minimium number of server required by a system operating under the TAGS policy to be stable and we present instances where it can be very high or, even worse, there are instances where the TAGS policy can not be stable for any number of servers.
- We also compare numerically the performance of the TAGS policy with the performance of SITA when r is finite. For two servers, we show that there are instances where the performance of the TAGS policy is more than two times the performance of SITA when the load of the system is small. For higher loads, we present instances where the performance the TAGS policy is extremely bad comparing to that of the SITA policy. As a consequence, the penalty for not knowing the sizes of the incoming tasks can be very high when the system load is larger than one, whereas when the load is smaller than one the performance of a system operating under the TAGS policy is, at most, 3 times more than the performance of a system operating under the SITA policy.

The rest of the article is organized as follows. In Section II, we put our work in the context of the existing literature. In Section III, we describe the model we study in this article and, in Section IV, we analyze the stability of a system that operates under the TAGS policy. We study the performance of the optimal TAGS policy in Section V. In Section VI we assume that the job size distribution is Bounded Pareto and we explore the asymptotic regime where the maximum job size tends to infinity. We present the numerical experiments we have performed in Section VII and we give the main conclusions of our work in Section VIII.

II. Related work

The analytical study of how to balance the load in a system with parallel queues optimally has been of great interest for researchers in Computer Science, see [14] for a recent book in this topic. Many existing routing policies are included in the SQ(d) framework, where for each incoming job, $d \geq 2$ servers are picked uniformly at random to observe their states and the job is routed to the server in the best observed state (the least number of customers or least workload, for instance). In terms of performance optimality, it is know that this kind of systems are very good [11], [12], [21], [19], [24], however the author in [22] showed that when the variability of the job size distribution is high this family of policies are not optimal. This is, in fact, the regime where TAGS outperforms the routing policies that belong the SQ(d) family of policies [13].

A related routing policy to TAGS is the Size Interval Task Assignment (SITA) policy. For this policy, each host serves jobs whose service demand is in a designated range [5]. Thus, the variance of the job executed in the servers decreases, which leads to a performance improvement when the number of servers increases [6]. The authors in [10] show that the SITA policy with optimal cutoffs minimizes mean response time, when the servers are non-observable and FCFS and the size of all the tasks is known. When the number of servers tends to infinity, the authors in [1] show that the optimal SITA policy equalizes the loads of the servers. The author in [3] introduces a task assignment policy where the size of incoming tasks is required, but the goal is to maximize the probability of satisfying the utilization requirements of incoming tasks. The main difference of the TAGS policy this respect to the latter policy and SITA is that the TAGS policy does not require to know the size of the incoming jobs.

III. Model Description

We consider a system formed by h servers with equal capacity modeled by FCFS queues and an input stream of jobs that follows a Poisson distribution with rate λ.[1] The size of the incoming tasks is given by a sequence of i.i.d. random variables denoted by X. Let $F(s) = \mathbb{P}(X < s)$ be the cumulative distribution function of the job size distribution, $\mathbb{E}[X]$ its mean and $\mathbb{E}[X^m]$ its m-th moment. We assume $F(\cdot)$ to be differentiable and we write $f(s) = \frac{dF(s)}{ds}$. Thus, the load of the system is defined as $\rho = \lambda\mathbb{E}(X)$.

Without loss of generality, we assume that the size of the smallest job is one and the size of the largest job is $r > 1$. The range of the job size distribution is defined as the ratio between the largest and the smallest job size, which in our case is r. Since X is a non-negative random variable such that f(s)=0 if $s < 1$ or $s > r$, it follows that

$$\begin{aligned}\mathbb{E}[X] &= \int_0^\infty (1 - F(s))ds \\ &= \int_0^1 1ds + \int_1^r (1 - F(s))ds + \int_r^\infty 0ds \\ &= 1 + \int_1^r (1 - F(s))ds.\end{aligned}$$

We consider a multi-server assignment policy called TAGS. Let $s_0 = 1$ and $s_h = r$. In the policy TAGS, the servers are numbered $1, \ldots, h$ and there is a vector of $h-1$ cutoff values $\mathbf{s} = (s_1, s_2, \ldots, s_{h-1})$ verifying that $s_0 < s_1 < s_2 < \cdots < s_{h-1} < s_h$. All incoming jobs are sent to server 1. If a job has been served before s_1 units of time in server 1, it leaves the system; otherwise, when the execution time equals s_1, it is stopped and sent to the end of the queue of server 2, where the execution starts from scratch. Thus, jobs that are executed in server i have been previously executed in servers $1, 2, \ldots, i-2$ and $i-1$, respectively, $s_1, s_2, \ldots, s_{i-2}$ and s_{i-1} units of time. Besides, if a job is being processed by the ith server and its execution time less than s_i time units, it leaves the system and, if not, it is stopped and put at the end of the queue of the next server. Jobs at the last server always run to completion.

For a given vector of cutoffs $\mathbf{s}$, we denote by $W(\mathbf{s})$ the random variable of the waiting time of incoming jobs. For a given vector of cutoffs $\mathbf{s}$, we will be interested in analyzing the normalized mean waiting time, which is given by

$$\mathbb{E}[\bar{W}(\mathbf{s})] = \frac{\mathbb{E}[W(\mathbf{s})]}{\mathbb{E}[X]},$$

where $\mathbb{E}[W(\mathbf{s})]$ is the mean waiting time of jobs in the system.

Let $\mathbf{s}^{opt} = (s_1^{opt}, ..., s_{h-1}^{opt})$ be a vector of cutoffs that minimizes the mean waiting time of jobs among all possible cutoffs, i.e.,

$$\mathbf{s}^{opt} \in \arg\min_{\mathbf{s}} \mathbb{E}[W(\mathbf{s})].$$

To simplify notation, we write $\mathbb{E}[\bar{W}(\mathbf{s}^{opt})] = \mathbb{E}[\bar{W}^*]$ for the optimal normalized mean waiting time and $\mathbb{E}[W(\mathbf{s}^{opt})] = \mathbb{E}[W^*]$ for the mean waiting time.

The size of the jobs executed in server i is denoted by X_i. The probability that a job size belongs to the interval $[s_{i-1}, s_i]$ is given by p_i. Likewise, the probability for a job to be executed in server i is denoted by $\bar{p}_i$. From the definition of the TAGS policy, it follows that $p_i = F(s_i) - F(s_{i-1})$ and $\bar{p}_i = 1 - F(s_{i-1})$. The waiting time in server i is denoted by $W_i(\mathbf{s})$ and the load in server i by ρ_i.

Throughout the paper, we will use the notation $f \sim g$ to denote two quantities f, g whose ratio tends to 1 as r tends to infinity. We also use the notation $\lfloor x \rfloor$ for the floor of x.

A. Bounded Pareto distribution

A distribution X is said to be Bounded Pareto with parameters 1, r and α if its density has the form of the Pareto

[1] We remark that, as we will see later, some of the results hold also without assuming Poisson arrivals.

distribution with parameter α, but is restricted to a bounded domain $1 \leq s \leq r$. Let $a = \frac{1}{r}$. If $1 \leq s \leq r$, then

$$f(s) = \frac{\alpha s^{-\alpha-1}}{(1-a^{\alpha})},$$

and $f(s) = 0$ otherwise. Besides, the cumulative distribution function of the Bounded Pareto distribution is given by the following expression:

$$F(s) = \begin{cases} 0, & s \leq 1, \\ \frac{1-(1/s)^{\alpha}}{1-a^{\alpha}}, & 1 \leq s \leq r, \\ 1, & s \geq r. \end{cases}$$

Besides, when $\alpha \neq 1$, we have that

$$\mathbb{E}[X] = \frac{\alpha}{\alpha - 1} \frac{1 - a^{\alpha-1}}{1 - a^{\alpha}} \tag{1}$$

whereas when $\alpha = 1$

$$\mathbb{E}[X] = \frac{\ln(r)}{1 - \frac{1}{r}}. \tag{2}$$

We note that the Bounded Pareto distribution with $\alpha = -1$ coincides with the uniform distribution on the interval $[1, r]$.

The Bounded Pareto distribution with large range and $0 < \alpha < 2$ is know to be a good model for high variance job size distributions [16].

IV. Stability analysis

In this section, we analyze the load that ensures the stability of the system. A system operating under the TAGS policy satisfies that jobs never reenter the same server and therefore it can be described as a multi-class feed forward network, where jobs of class i correspond to the jobs which terminate service at server i. For such systems it is known, see [7], that stability is equivalent to the condition that each server in the network is sub-critical. The following result characterizes the conditions for the stability of the system.

Proposition 1. *Let $M(X) = \sup_s s(1 - F(s))$. Given a load ρ, there exists a number of servers h and a vector of cutoffs $\mathbf{s} = (s_1, ..., s_{h-1})$ such that the system is stable if and only if*

$$\rho < \frac{\mathbb{E}[X]}{M(X)}. \tag{3}$$

Proof. We first assume that the system with h servers is stable. Let s be the value at which $M(X)$ is achieved for a given job size distribution. Thus, we have that $s_{i-1} \leq s \leq s_i$ for server i. All jobs of size s or more will be executed in server i and, as a result, the server will spend at least s time units on each such job. Since the rate of jobs of size at least s is $\lambda(1-F(s))$, the system is stable if

$$\lambda s(1 - F(s)) < 1.$$

Using that $\rho = \lambda E[X]$, it results that (3) holds.

Conversely, we assume that (3) holds. Since jobs in server i spend at most s_i time units, the load of server i is upper bounded by $\lambda(1 - F(s_{i-1}))s_i$. Furthermore, using the definition of $M(X)$ and also that $\rho = \lambda\mathbb{E}[X]$

$$\lambda(1 - F(s_{i-1}))s_i \leq \lambda M(X)s_i/s_{i-1} = \rho M(X)/\mathbb{E}[X](s_i/s_{i-1}).$$

Fix any $t > 1$ such that $t\rho M(X)/E(X) < 1$ and let $s_i = t^i$, then by the above inequality shows that the load of server i is less than one. Finally, since for server h we have $r = t^h$, it follows that $h = \lfloor \log_t(p) \rfloor + 1$ servers are enough to be a stable system. □

This result says that, for a given job size distribution X, there is a critical load $\rho_{crit}(X) = \frac{\mathbb{E}[X]}{M(X)}$, that is, the system can be stable only when its load is smaller than $\rho_{crit}(X)$.

In the remainder of the article, we assume that the system operating under the TAGS policy is stable, that is, the load of the system is below the value of the critical load.

The following theorem provides an upper bound on the critical load when the job size distribution has range r.

Proposition 2. *For a job size distribution X with range r, we have that $\rho_{crit}(X) \leq 1 + \ln(r)$.*

Proof. See Appendix B. □

We now study the value of the critical load for particular job size distributions. We focus on the Bounded Pareto distribution in the next section.

A. Bounded Pareto distribution

We now assume that the job size distribution is Bounded Pareto with parameters 1, r and α. In the following result, we characterize the critical load for this distribution.

Proposition 3. *For the Bounded Pareto distribution with parameters 1, r and α,*

- *if $\alpha \neq 1$, then $\rho_{crit} = (1 - a^{1-\alpha})(1 - \alpha)^{-1/\alpha}$,*
- *if $\alpha = 1$, then $\rho_{crit} = \frac{r \ln(r)}{r-1}$.*

Proof. For the Bounded Pareto distribution with $\alpha \neq 1$, the supremum of $s(1 - F(s))$ is achieved when $s = r(1 - \alpha)^{\alpha}$ and therefore

$$M(X) = r^{1-\alpha} \frac{(1-\alpha)^{1/\alpha}}{1 - a^{\alpha}} \frac{\alpha}{1 - \alpha}.$$

Dividing (1) by the above expression, it results that

$$\rho_{crit} = (1 - a^{1-\alpha})(1 - \alpha)^{-1/\alpha}.$$

For $\alpha = 1$, since $s(1-F(s))$ is a decreasing function of s, the supremum of $s(1-F(s))$ is given when $s = 1$ and, therefore, $M(X) = 1$. As a result, we have that $\rho_{crit} = \mathbb{E}[X]$ and using (2) the desired result follows. □

We have the following corollary that characterizes the critical load for fixed α as r tends to infinity.

Corollary 4. *For the Bounded Pareto distribution with parameters 1, r and α, when $r \to \infty$,*

- *if $\alpha \neq 1$, then $\rho_{crit} \to (1 - \alpha)^{-1/\alpha}$,*

- *if* $\alpha = 1$*, then* $\frac{\rho_{crit}}{\ln(r)} \to 1$.

In Proposition 3, we show that, when $\alpha = 1$, the critical load is $\frac{r}{r-1}\ln(r)$, which is very close to the upper bound given in Proposition 2. Besides, from Corollary 4, it follows that the critical load when $\alpha = 1$ coincides with the upper bound of Proposition 2 when $r \to \infty$. We now present a distribution where the value of $\rho_{crit}(X)$ is equal to $1 + \ln(r)$, that is, the upper bound of the critical load is tight.

B. A distribution with a tight upper-bound for the critical load

We consider a job size distribution X with a probability mass function defined as follows:

$$f(x) = \begin{cases} \frac{1}{x^2}, & \text{if } x \in [1, r), \\ \frac{1}{r}, & \text{if } x = r, \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

Interestingly, for this distribution, we have that $x(1 - F(x)) = 1$, for all $x \in [1, r]$ and, as a result, $M(X) = 1$. Furthermore, for the mean job size, it results that

$$\mathbb{E}[X] = 1 + \int_1^r x f(x) dx = 1 + \int_1^r x \frac{1}{x^2} dx = 1 + \ln(r).$$

As a consequence of this reasoning and using that $\rho_{crit}(X) = \mathbb{E}[X]/M(X)$, the following result follows.

Proposition 5. *If X is the job size distribution defined in* (4)*, then $\rho_{crit}(X) = 1 + \ln(r)$.*

V. Bounds on the normalized mean waiting time

In this section, we are interested in the performance achieved when the vector of cutoffs is $\mathbf{s}^{opt}$, that is, when the cutoffs are chosen so as to minimize the mean waiting time of jobs in the system. In the following result, we show how the normalized mean waiting when the vector of cutoffs is $\mathbf{s}^{opt}$ and when the vector of cutoffs minimizes the maximum mean queue length among servers are related.

Proposition 6. *Let $\mathbf{s}^{que}$ be the vector of cutoffs that minimizes the maximum mean queue length of the servers. Then,*

$$\mathbb{E}[\bar{W}(\mathbf{s}^{que})] \leq h\mathbb{E}[\bar{W}^*] + h - 1. \tag{5}$$

Proof. See Appendix A. □

This result gives an upper bound for the performance of a system where the vector of cutoffs minimizes the maximum mean queue length. Moreover, from this result, we can easily derive a lower-bound for the optimal normalized mean waiting time that depends on the number of servers and the normalized mean waiting time when the vector of cutoffs minimizes the maximum mean queue length, i.e.,

$$\mathbb{E}[\bar{W}^*] \geq \frac{\mathbb{E}[\bar{W}(\mathbf{s}^{que})]}{h} - \frac{h-1}{h},$$

and using that $\frac{h-1}{h} \leq 1$, we get the following result.

Corollary 7.

$$\mathbb{E}[\bar{W}^*] \geq \frac{\mathbb{E}[\bar{W}(\mathbf{s}^{que})]}{h} - 1.$$

Using that, for all $\mathbf{s}$, $\mathbb{E}[\bar{W}(\mathbf{s})] = \frac{\mathbb{E}[W(\mathbf{s})]}{\mathbb{E}[X]}$, an analogous result of Proposition 6 is obtained for the mean waiting time of jobs, i.e.,

$$\mathbb{E}[W(\mathbf{s}^{que})] \leq h\mathbb{E}[W^*] + \mathbb{E}[X](h-1).$$

We remark that the result of Proposition 6 holds for an arbitrary arrival process, i.e., a sequence of i.i.d. random variables with common distribution, not necessarily Poisson. The only requirement for this result to hold is that it is satisfied the Little's Law.

We now present that a similar result to that of Proposition 6 is given for SITA policies, a size-aware policy where the service times are divided into intervals and all the jobs with size in a given interval are dispatched to the same queue. The proof is identical to that of Proposition 6 and therefore we omit it.

Remark 8. *Consider a system operating under the SITA policy. Then,*

$$\mathbb{E}[\bar{W}(\mathbf{s}^{que})] \leq h\mathbb{E}[\bar{W}^*],$$

where $\mathbf{s}^{que}$ is the vector of cutoffs that minimizes the maximum mean queue length of the servers.

VI. Asymptotic analysis for Bounded Pareto distribution

In this section, we analyze the normalized mean waiting time of jobs when the job size distribution is Bounded Pareto with parameters 1, r and α, where $\alpha \in (0, 2)$ and r tends to infinity. As in [13], we assume Poisson arrivals to all the servers for analytical tractability. As we will see in section VII, the performance of the system under this assumption is very accurate. We first focus on the case where $\rho < 1$ and then we analyze the performance of a system operating under the TAGS policy when the load is larger than 1. Finally, we present some extensions of the results obtained in this section.

A. The case $\rho < 1$

We first consider that the total system utilization satisfies that $\rho < 1$. In the following result, we compare the optimal mean waiting time of a system operating under the TAGS policy and of a system operating under the SITA policy.

Theorem 9. *Let $\rho < 1$. For Bounded Pareto distributed job sizes with $r \to \infty$, the mean waiting time in a TAGS system with optimal cutoffs is at most two times larger than the mean waiting time of a SITA system with optimal cutoffs.*

Proof. See Appendix C. □

We know that the SITA policy requires the knowledge of the size of incoming tasks, whereas the TAGS policy it does not. Therefore, a system operating under the SITA policy always outperforms a system operating under the TAGS policy. However, an important conclusion from the result of Theorem 9 is that, in the asymptotic regime, the penalty for not knowing job sizes is upper bounded by a factor of 2, for any value of α and any number of servers.

From the result of Theorem 9 and taking into account that the vector of cutoffs that minimizes the mean waiting time also minimizes the normalized mean waiting time, we conclude that when r tends to infinity, the normalized mean waiting time in a TAGS system with optimal cutoffs is at most two times larger the normalized mean waiting time of a SITA system with optimal cutoffs.

B. The case $\rho > 1$

We now study the performance of a system when the load is higher than one. First, we define the number of spare servers as $\tilde{h} = h - i + 1$, where i is the minimum number of servers to be stable the system. We present a procedure to obtain this value next.

Remark 10. *Given $\rho < \frac{\mathbb{E}[X]}{M(X)}$ we can determine the exact number of servers needed for constructing a stable system with load ρ by the following procedure. Let*

$$\rho_i = \lambda p_i \mathbb{E}[X_i] + \lambda s_i (1 - F(s_i)).$$

Fixing s_{i-1} it is easy to see that ρ_i is a non decreasing function in s_i. Let $\tilde{s}_1$ be such that $\rho_1 = 1$. More generally, for a given $\tilde{s}_{i-1}$, we compute $\tilde{s}_i$ to satisfy that $\rho_i = 1$. If there is no value for which $\rho_i = 1$, it is enough to consider a system formed by i servers to ensure stability.

We know, see the numerical section, that for higher loads, the difference on the performance between SITA and TAGS can be extremely large. Therefore, we focus on the TAGS policy to study the optimal normalized mean waiting time in the asymptotic regime.

In the following result we give an expression of the order of magnitude of the ratio of the performance of a system operating under the TAGS policy for $\rho > 1$.

Proposition 11. *When $\rho > 1$,*

$$\mathbb{E}[\bar{W}^*] = \Theta(r^{\frac{|2\alpha-2|}{q^{\tilde{h}}-1}}), \tag{6}$$

where $q = \frac{\alpha}{2-\alpha}$ if $\alpha > 1$, $q = \frac{2-\alpha}{\alpha}$ if $\alpha < 1$ and $q = 1$ if $\alpha = 1$.

Proof. See Appendix D. □

We observe that the order of magnitude of the performance of a system operating under the TAGS policy depends on the number of servers and the minimum number of servers to stabilize the system only through $\tilde{h}$, i.e., the number of spare servers.

C. Extensions

1) Heterogeneous servers: We now present how the results of Theorem 9 and of Proposition 11 extend to a system with heterogeneous servers. For this case, the job size distribution is given with respect to some reference server. Each server in the system, say server i, has an associated power coefficient c_i. A job which takes t units of time on the reference server, takes t/c_i units of time on server i. In this setting, the optimization is performed in two stages: on the one hand, one must choose the vector of cutoffs $\mathbf{s}$ and, on the other hand, which server gives service to jobs whose size is at least s_{i-1}. Moreover, the low load condition ($\rho < 1$) is replaced by the condition that the strongest server (largest c_i) can handle the entire load.

The proof of Theorem 9 does not require the assumption that the servers are homogeneous. To generalize the proof of Proposition 11 to heterogeneous servers, we need to redefine the number of spare servers. For $\alpha < 1$ this is done by ordering the servers in increasing order of the capacities and checking how many servers are in a low load system. When $\alpha > 1$, we proceed in an analogous manner, but ordering the servers in decreasing order of the capacities. In fact, the load burden in the case $\alpha < 1$ falls on the large job servers, whereas for $\alpha > 1$ the opposite is true.

2) Variants of the TAGS policy: We now study the performance of a variant of the policy TAGS where jobs are resumed on the next server from the point in which they were stopped in the previous server [8], [4].

This assumption clearly improves the stability of the TAGS policy. However, when the job size distribution is Bounded Pareto and $\rho < 1$, the asymptotic performance of both policies coincide. To see this, we consider that $s_i/s_{i-1} \to \infty$ and define $\tilde{s}_{i-1} = (1+\varepsilon)s_{i-1}$, where $\varepsilon > 0$. We consider a server in a system operating under the TAGS policy which handles jobs in the range $[\tilde{s}_{i-1}, s_i]$ and from each job whose original size was $s \geq s_{i-i}$ we subtracted $\tilde{s}_{i-1}$ work. Hence, the traffic of server i for the latter system is smaller than that of server i in a system implementing the TAGS policy where work is resumed. Using the same arguments as in Theorem 9, we can show that, in the asymptotic regime, the contribution to the waiting time of server i when handles jobs ranging in size between $\tilde{s}_{i-1}$ and s_i is the same as the original TAGS policy and, therefore, our conclusion follows.

3) The T+W policy: The numerical experiments performed in section VII-C suggest that the value of $\tilde{h}$ can be substantially improved at higher loads. For this purpose, we present a policy that combines TAGS with work preserving policies like LWL or random assignment, which are better at consuming load than TAGS. We call this policy T+W. The basic idea is to divide the set of servers in two types: in the first group of servers, jobs are routed according to a work preserving policy and handles jobs whose size s satisfies that $s(1-F(s))$ is large; and in the second group of servers, the remaining jobs are routed according to TAGS assignment. We now present how this policy can be implemented for Bounded Pareto distributed job sizes.

We first assume $\alpha < 1$. Let $\tilde{s}$ be such that the load of jobs whose size is greater than $\tilde{s}$ is 1. All the jobs are routed to $\lfloor \rho \rfloor + 1$ servers according to a work conserving policy. If the execution of a job exceeds $\tilde{s}+\varepsilon$ unit (for $\varepsilon > 0$) time units, it is killed and sent to the remaining servers where it is processed according to the TAGS policy.

Similarly, when $\alpha > 1$, we define $\tilde{s}$ to be such that the load of a server in a TAGS system serving jobs ranging in size in the interval $[1, \tilde{s}]$ is precisely 1. The remaining load,

which is formed by jobs whose size is in the interval $[\tilde{s}, p]$, is routed to the LWL or random assignment system, which requires $\rho] + 1$ servers to be stable. The remaining servers manage all incoming jobs using the TAGS policy, where the last cutoff is $\tilde{s} - \varepsilon$. Jobs which are of size greater than $\tilde{s} - \varepsilon$ are sent to begin service from scratch in the work preserving system.

It is easy to see that the stability of the improvement of the stability T+W policy with respect to TAGS is huge and the main reason for this is that jobs are routed to $\tilde{\rho} + 1$ servers work preserving policies like LWL or random assignment.

VII. Numerical experiments

A. The approximation equations

The mean waiting time of jobs in a system operating under the TAGS policy with Poisson arrivals has no exact analytical formula. The reason for this is that the input stream to the second server and beyond is not Poisson. We analyze the approximation suggested in [13] that consists of assuming Poisson arrivals to all servers. Indeed, under this assumption, an approximation to the average waiting time can be computed using the Pollaczek-Khinchine equation for an M/G/1 queue. The author in [13] also suggested that the approximation will over-estimate the average waiting time since the input streams, to all but the first server, tend to be more regular than Poisson, having near constant inter-arrivals.

To analyze the true performance of TAGS and to compare it with the approximation equations, we developed a simulation of a system operating under the TAGS policy. For each run, we consider an arrival traffic of 10^8 jobs and the Bounded Pareto job size distributions with different values of α, varying from 0 to 2, and different numbers of servers. The total load of the system is $\rho = h/2$, where h is the number of servers. The smallest job size was of size 1 and the largest job size was $r = 10^4$. This value was chosen because for $\alpha = 2$, the probability of a job of size greater than s is about s^2, hence the probability for a job of size greater than 10^4 is approximately 10^{-8}. This means that for α close to 2, and 10^8 jobs in a simulation run we would not get jobs substantially greater than 10^4, therefore, there was no point in choosing a larger value for r. The values of $s_1, \ldots, s_{h-1}$ for the system were chosen to be close to optimal for minimizing average waiting time in the approximate equations.

The results of Table VII-A show that the average waiting time value which is computed using the approximation equations is always close to the value computed from the simulations. Moreover, as conjectured in [13], the computed value always over-estimates the actual average waiting time. As expected, the computed values are closest to the simulation results when the number of servers is small. The computed values for $h = 2$ are essentially identical to the simulated values, except for the case $\alpha = 1.8$ where there was a 10% difference. For larger values of h the error can be as large as 20%, a value which we still consider to be very reasonable.

TABLE I
Comparison of actual waiting time from simulations of TAGS systems, with the estimate from the approximation formulas, which assume Poisson arrivals at all servers.

h	α	Simulated $\mathbb{E}[W^*]$	Calculated $\mathbb{E}[W^*]$
2	0.2	1368.72	1408.87
2	0.4	577.91	583.34
2	0.6	214.06	215.64
2	0.8	79.60	79.86
2	1	33.83	34.02
2	1.2	18.07	17.99
2	1.4	11.38	11.38
2	1.6	7.83	7.88
2	1.8	5.25	5.73
3	0.2	2323.63	2657.09
3	0.4	704.26	769.79
3	0.6	185.89	197.60
3	0.8	52.70	54.32
3	1	19.89	20.35
3	1.2	11.76	12.03
3	1.4	9.74	10.12
3	1.6	9.09	10.20
3	1.8	9.59	11.14
4	0.2	11180.74	13982.31
4	0.4	1468.58	1789.23
4	0.6	255.05	287.54
4	0.8	54.47	58.84
4	1	18.82	19.72
4	1.2	12.17	13.01
4	1.4	13.28	15.15
4	1.6	22.72	28.05
4	1.8	143.89	181.70
5	0.6	484.40	597.45
5	0.8	68.04	77.26
5	1	21.04	22.81
5	1.2	15.24	17.31
5	1.4	28.93	36.34
6	0.8	98.86	117.45
6	1	25.71	29.04
6	1.2	22.46	27.19
7	0.8	173.79	221.62
7	1	33.70	39.98
7	1.2	43.60	55.88
8	1	50.17	60.72

The largest errors occur for the extreme values of α, away from the central value $\alpha = 1$, where the errors are smallest.

We have also performed experiments with the larger value of the maximum job size and we have observed that, in all the cases, the obtained results follow the pattern presented in Table VII-A, and therefore we can conclude that the approximate equations are fairly accurate and conservative.

B. The minimum number of servers to stabilize the system

We consider a system operating under the TAGS policy when the job size distribution is Bounded Pareto. In Table VII-B, we represent the minimum number of servers needed to stabilize this system for different values of α, from 0 to 2, and different values of ρ higher than one. The symbol NA, denotes the case where $\rho \geq \rho_{crit}$ and hence, a system operating under the TAGS policy cannot be stable regardless of the number of servers.

As it can be observed in Table VII-B, when the load is equal to one, the minimum number of servers to stabilize the

TABLE II
THE VALUES OF THE MINIMUM NUMBER OF SERVERS TO STABILIZE A SYSTEM, WITH THE JOB SIZE DISTRIBUTION BOUNDED PARETO WITH $r = 10^4$ AND α VARYING FROM 0 TO 2 AND DIFFERENT VALUES OF ρ FROM 1 TO 4.

α	$\rho = 1$	$\rho = 1.5$	$\rho = 2$	$\rho = 2.5$	$\rho = 3$	$\rho = 3.5$	$\rho = 4$
0.1	2	3	4	9	NA	NA	NA
0.2	2	2	4	7	34	NA	NA
0.3	2	2	4	6	14	NA	NA
0.4	2	2	4	5	9	33	NA
0.5	2	2	3	5	7	13	NA
0.6	2	2	3	4	6	9	14
0.7	2	2	3	4	5	7	9
0.8	2	2	3	4	5	6	7
0.9	2	2	3	3	4	5	6
1.0	2	2	3	3	4	5	6
1.1	2	2	3	3	4	5	6
1.2	2	2	3	4	4	6	7
1.3	2	2	3	4	5	7	11
1.4	2	2	3	4	6	NA	NA
1.5	2	2	3	5	NA	NA	NA
1.6	2	2	3	6	NA	NA	NA
1.7	2	2	4	NA	NA	NA	NA
1.8	2	2	4	NA	NA	NA	NA
1.9	2	2	5	NA	NA	NA	NA

system is always 2. However, when the load of the system increases, the situation changes and the minimum number of servers to stabilize the system changes with α. For instance, when $\rho = 2$, the minimum number of servers is 4 for $\alpha = 0.1$, 3 for $\alpha = 1$ and 5 for $\alpha = 1.9$.

We see in Table VII-B that there are instances where the minimum number of servers to stabilize the system is very high. For example, when $\rho = 3$ and $\alpha = 0.2$, the obtained value is 34 and when $\rho = 3.5$ and $\alpha = 0.4$, it is 33.

We show in Table VII-B that, for some instances, the load is larger than $\rho_{crit}(X)$ and, therefore, the system operating under the TAGS policy can not be stable. For instance, when ρ is equal to 3.5 and 4 and α is larger or equal than 1.4, a system operating under the TAGS policy can not be stable. The author in [13] has also observed that the TAGS policy performs poorly when the load of the system is higher than one and they study the server expansion of the TAGS system, that is, the number of servers to be added in a system to stabilize the system. In our illustration, we show that there are instances where the system cannot be stabilize by increasing the number of servers.

From the results shown in Table VII-B, one can conclude that the performance comparison that has been carried out in Theorem 9 cannot be done for any value of $\rho > 1$. Therefore, if we aim to study the performance of a system operating under the TAGS system, we need to check that the stability condition $\rho < \rho_{crit}(X)$ is satisfied.

C. Performance comparison with SITA for $r < \infty$

In Theorem 9, we have shown that the ratio of the performance of a system operating under the optimal TAGS policy over the performance of a system operating under the optimal SITA in the asymptotic regime is upper bounded by two, i.e,

$$\frac{\mathbb{E}[W^*]}{\min_{\mathbf{s}} \mathbb{E}[W^{SITA}(\mathbf{s})}] \leq 2.$$

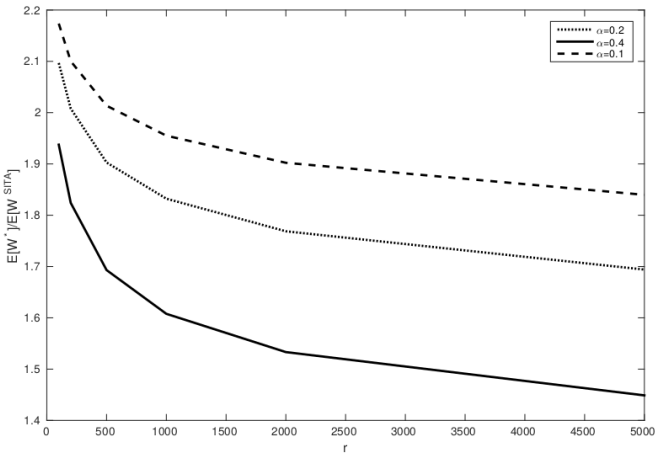

Fig. 2. The ratio of the mean waiting time of TAGS over the mean waiting time of SITA when r varies from 100 to 5000 for different values of α. $h = 2$ and $\rho = 0.5$.

In this section, we focus on the above ratio of performances when r is finite. First, we consider a system with $\rho < 1$. In Figure 2, we consider a system with two servers and $\rho = 0.5$. We study the performance ratio when r varies from 100 to 5000 and different values of α. We observe that when r is 100 or 200 and α is equal to 0.1 or 0.2, the performance ratio is higher than two. Hence, this illustration thus shows that the result of Theorem 9 does not hold for a finite r. We have performed further numerical experiments to explore the upper bound of this performance ratio and the performance ratio seems to be upper bounded by 3 for two hosts when ρ is less than one and r is finite. Unfortunately, we have not succeeded in showing this upper bound and thus we conjecture that the performance ratio in a system with two servers is upper bounded by 3 when ρ is less than one and r is finite. Hence, proving this conjecture, at least until now, remains as an unsolved problem.

We also study this performance ratio when $\rho > 1$. We have computed the mean waiting time of jobs in a system operating under the SITA-E policy (i.e, the SITA policy where the load of the servers is equalized) for the parameters considered in Table VII-A. We know from [18] that the optimal SITA balances the load of the servers for the Bounded Pareto distribution with $\alpha = 1$. Therefore, the performance ratio we present in Table VII-C coincides with $\frac{\mathbb{E}[W^*]}{\min_{\mathbf{s}} \mathbb{E}[W^{SITA}(\mathbf{s})}]$ for this case. For the rest of the cases, since $\mathbb{E}[W^{SITA-E}] \geq \min_{\mathbf{s}} \mathbb{E}[W^{SITA}(\mathbf{s})]$, it follows that the values we show in Table VII-C are lower bounds of the values of the performance ratio we are investigating in this section.

As it can be seen in Table VII-C, for $\alpha = 1$, the difference on the performance of TAGS and SITA increases with the number of servers. In fact, when the number of servers is 7 and 8, the performance ratio is, respectively, 60.72 and 114.15.

TABLE III
MEAN WAITING TIME OF THE SITA-E POLICY AND THE PERFORMANCE OF TAGS OVER THE PERFORMANCE OF SITA-E FOR THE PARAMETERS CONSIDERED IN TABLE VII-A.

h	α	$\mathbb{E}[W^{SITA-E}]$	$\mathbb{E}[W^*]/\mathbb{E}[W^{SITA-E}]$
2	0.2	91.66	14.93
2	0.4	38.56	14.98
2	0.6	16.31	13.12
2	0.8	9.11	8.73
2	1	10.507	3.21
2	1.2	17.59	1.03
2	1.4	16.266	0.69
2	1.6	9.0235	0.86
2	1.8	4.15	1.26
3	0.2	37.618	61.76
3	0.4	16.733	42.09
3	0.6	7.33	25.36
3	0.8	3.738	14.12
3	1	3.042	6.54
3	1.2	4.88	2.41
3	1.4	6.335	1.53
3	1.6	4.557	1.99
3	1.8	2.432	3.95
4	0.2	20.358	549.21
4	0.4	9.272	158.39
4	0.6	4.195	60.79
4	0.8	2.174	25.05
4	1	1.586	12.06
4	1.2	2.209	5.51
4	1.4	3.301	4.02
4	1.6	2.806	8.09
4	1.8	1.659	86.73
5	0.6	2.715	178.42
5	0.8	1.442	47.25
5	1	1.0274	20.49
5	1.2	1.273	12
5	1.4	3.013	9.61
6	0.8	1.03	95.98
6	1	0.731	35.17
6	1.2	0.843	26.64
7	0.8	0.775	224.24
7	1	0.555	60.72
7	1.2	0.608	71.71
8	1	0.4395	114.15

For $\alpha \neq 1$, we have found instances where the performance ratio is extremely high. For instance, when $\alpha = 0.2$ and the number of servers is 4, the ratio $\mathbb{E}[W^*]/\mathbb{E}[W^{SITA-E}]$ is 549.21, which means that the performance of a system operating under the optimal TAGS policy is more than 549.21 times the performance of a system operating under the optimal SITA policy.

VIII. CONCLUSION

We have studied the performance of a system operating under the TAGS policy. For a given system load and any job size distribution, we have shown a necessary and sufficient condition for the stability of the system. The main conclusion of this result is that, for a given job size distribution X, there exists a critical load $\rho_{crit}(X)$, i.e. , the system is stable if and only if $\rho < \rho_{crit}(X)$. We have shown that $\rho_{crit}(X)$ is upper bounded by one plus the logarithm of the largest job size. Besides, we have computed the value of $\rho_{crit}(X)$ for the Bounded Pareto distribution and we have provided a distribution where the upper bound is attained. We have studied the performance of the optimal TAGS policy, i.e., when the cutoffs $s_1, s_2, \ldots$ minimize the mean waiting time of jobs, and we have compared it with the performance of the TAGS policy when the cutoffs minimize the maximum queue length of the serves. We have shown the relation between the performance of both systems, which allows us to provide a lower bound of the former and an upper bound of the latter. We have analyzed the performance of a system operating under the TAGS policy when the job size distribution is Bounded Pareto in the asymptotic regime where the maximum job size tends to finite (and the smallest job size is equal to one). First, we have shown that, when the system load is smaller than one, the performance of the optimal TAGS policy is, at most, two times worst than the performance of the optimal SITA policy. The main conclusion of this result is that the price of not knowing the job sizes of incoming jobs is upper bounded by 2 when the load is small. However, when the system load is higher than one, the performance of the TAGS policy is large comparing with that of the SITA policy. However, we have proved that the order of magnitude the performance of the TAGS policy in the asymptotic regime and we present a routing policy, that we call T+W, that implements the TAGS policy in some servers and a work conserving policy, such as Least-Work-Left, in the remaining ones. We have also compared the performance of SITA and TAGS when the largest job size is finite and the numerical experiments we have performed hat confirm that, when the system load is low, the difference on the performance of both systems is small, whereas when the system load is larger than one it is extremely large.

For future work, it would be interesting to perform an analytical study of the performance of TAGS policy for Bounded Pareto job size distribution when the largest job size is finite. Another possible future research is to analyze the routing policy T+W for an arbitrary job size distribution as well as to compute the difference on the performance of implementing this policy with respect to the TAGS policy. Finally, we would like to investigate the influence on the performance of the system operarting under the TAGS policy when we increase proportionately the number of servers and the arrival rate.

IX. ACKNOWLEDGMENTS

The authors would like to thank Gadi Rabinovich for many helpful discussions and support. They also thank Ofer Zeitouni and Mauray Bramson for helpful comments.

Eitan Bachmat was partially supported by an IBM faculty award.

Josu Doncel has received funding from the Department of Education of the Basque Government through the Consolidated Research Group MATHMODE (IT1294-19), from the Marie Sklodowska-Curie grant agreement No 777778 and from the Spanish Ministry of Economy and Competitiveness project MTM2016-76329-R.

APPENDIX

A. Proof of Proposition 6

For a job that finishes service in server i and a vector of cutoffs $\mathbf{s}$, we define $T_i(\mathbf{s})$ as the sum of the the execution times in the first $i-1$ servers. Hence, $T_i(\mathbf{s}) = \sum_{j=1}^{i-1} s_j$. Moreover, for a job of size s such that $s_{i-1} \leq s \leq s_i$,

$$T_i(\mathbf{s}) \leq (i-1)s_{i-1} \leq (i-1)s \leq (h-1)s,$$

and therefore

$$\sum_{i=1}^{h} p_i T_i(\mathbf{s}) \leq (h-1)\mathbb{E}[X]. \tag{7}$$

Let $b_i(\mathbf{s}) = \bar{p}_i W_i(\mathbf{s})$, for $i = 1, \ldots, h$. For a vector of cutoffs $\mathbf{s}$, the mean waiting time of jobs in the system is

$$\mathbb{E}[W(\mathbf{s})] = \sum_{i=1}^{h} b_i(s) + \sum_{i=1}^{h} p_i T_i(\mathbf{s}).$$

For the optimal mean waiting time, we have that

$$\mathbb{E}[W^*] \geq \max\{b_1(\mathbf{s}^{opt}), \ldots, b_h(\mathbf{s}^{opt})\}$$

Now, we notice that, for a given vector of cutoffs $\mathbf{s}$, the queue length of server i is given by $\lambda b_i(\mathbf{s})$. As a result, by definition of $\mathbf{s}^{que}$,

$$\max\{b_1(\mathbf{s}^{que}), \ldots, b_h(\mathbf{s}^{que})\} \geq \max\{b_1(\mathbf{s}^{que}), \ldots, b_h(\mathbf{s}^{que})\}.$$

Furthermore, we have that

$$\max\{b_1(\mathbf{s}^{que}), \ldots, b_h(\mathbf{s}^{que})\} \geq \frac{1}{h}\sum_{i=1}^{h} b_i(\mathbf{s}^{que}).$$

Therefore, we have shown that $h\mathbb{E}[W^*] \geq \sum_{i=1}^{h} b_i(\mathbf{s}^{que})$. Taking into account that

$$\sum_{i=1}^{h} b_i(\mathbf{s}^{que}) = \mathbb{E}[W(\mathbf{s}^{que})] - \sum_{i=1}^{h} p_i T_i(\mathbf{s}^{que}),$$

and also the property of (7), if we divide both side of the above expression by $\mathbb{E}[X]$, the desired result follows.

B. Proof of Proposition 2

Prior to prove the result of Proposition 2, we present the following lemmas:

Lemma 12. *For any job size distribution X of range r, there exists a continuous distribution Y of the same range such that $|\rho_{crit}(X) - \rho_{crit}(Y)| < \varepsilon$, for all $\varepsilon > 0$.*

Proof. Let X be a bounded distribution of range r with distribution F. We aim to show that for any $\varepsilon > 0$, it can be approximated by a continuous distribution Y of range r satisfying that $|\rho_{crit}(X) - \rho_{crit}(Y)| < \varepsilon$.

First, we decompose the range interval $[1, r]$ into n equal sub-intervals, with endpoints $1 = x_0, x_1, ..., x_n = r$. Consider the distribution Y_n which linearly extrapolates X between its endpoint values on each sub-interval $[x_i, x_{i+1}]$. Since X and Y_n are both monotone non decreasing functions they are Riemann integrable and it follows from the definition that $\lim_n \mathbb{E}[Y_n] = \mathbb{E}[X]$. We claim that $\lim_n M(Y_n) = M(X)$ as well.

Since the probability distribution of Y_n is given by F_{Y_n}, for any $s \in [x_i, x_{i+1}]$, it follows that

$$\begin{aligned} s(1-F(s)) &\leq s(1-F(x_i)) \\ &= (s-x_i)(1-F(x_i)) + x_i(1-F(x_i)) \\ &\leq \frac{r}{n} + x_i(1-F_{Y_n}(x_i)) \\ &\leq \frac{r}{n} + M(Y_n). \end{aligned}$$

As a result, we have that $M(X) \leq \frac{r}{n} + M(Y_n)$. Applying the same argument to Y_n instead of X and noting that $x_i(1 - Y_n(x_i)) = x_i(1-F(x_i)) \leq M(X)$, it follows that $M(Y_n) \leq \frac{r}{n} + M(X)$ and both inequalities together yield the claim for n large enough. □

Lemma 13. *For any continuous job size distribution X with range r, there exists a continuous job size distribution Y of the same range such that*

1) $\rho_{crit}(X) \leq \rho_{crit}(Y)$,
2) *Y is also supported on* $[1, r]$,
3) $M(Y) = 1$.

Proof. Let X be a continuous job size distribution of range r with probability distribution F. We first note that $M(X) \geq 1(1 - X(1)) = 1$. If $M(X) = 1$ then X and Y coincide and the desired result follows. Therefore, we focus on the case where $M(X) > 1$. Let $\tilde{s} > 1$ be the value at which $M(X)$ is achieved, i.e., $\tilde{s}(1 - F(\tilde{s})) = M(X)$. We know that $\tilde{s}$ exists by continuity of X. Consider a job size distribution Y which is supported on the interval $[M(X), r]$ whose probability distribution F_Y defined as follows: for $s \geq \tilde{s}$,

$$F_Y(s) = F(s),$$

whereas for $M(X) \leq s \leq \tilde{s}$,

$$1 - F_Y(s) = M(X)/s.$$

We first note that, by construction, $M(X) = M(Y)$. Besides, from the definition of $M(X)$ and the construction of $\tilde{Y}$, it follows that for all s,

$$1 - F_Y(s) \geq 1 - F(s).$$

As a result,

$$\mathbb{E}[X] = \int_1^r (1 - F(s))\, ds \leq \int_1^p (1 - F_Y(s))\, ds = \mathbb{E}[Y].$$

Therefore, from the above expression and using that $M(X) = M(Y)$, it follows that

$$\rho_{crit}(X) = \frac{\mathbb{E}[X]}{M(X)} \leq \frac{\mathbb{E}[Y]}{M(Y)} = \rho_{crit}(\tilde{Y}).$$

Finally we define $\tilde{Y}$ to be a rescaling of Y whose probability distribution is defined as $F_{\tilde{Y}}(s) = F_Y(M(X)s)$. For this

rescaling, it is easy to see that $\rho_{crit}(\tilde{Y}) = \rho_{crit}(Y)$. Moreover, the support of $\tilde{Y}$ is in $[1, r/M(X)]$, which is contained in $[1, r]$, and $M(\tilde{Y}) = 1$.

□

We now present the proof of the result of Proposition 2.

Proof. We first note that, from the arguments of the above two lemmas, it is sufficient to prove the bound for a continuous distribution which satisfies $M(X) = 1$, in which case we have that $\rho_{crit}(X) = E(X)$. As a result, our goal is to bound $E(X)$. Since $M(X) = 1$ we have for any s,

$$s(1 - F(s)) \leq 1 \iff 1 - F(s) \leq 1/s.$$

As a result,

$$\begin{aligned}\mathbb{E}[X] &= 1 + \int_1^r (1 - F(s))ds \\ &\leq 1 + \int_1^r (1/s)\, ds = 1 + \ln(r).\end{aligned}$$

And the desired result follows. □

C. Proof of Theorem 9

We now from [2] that, when r is large, the normalized mean waiting time of a system that operates under the SITA policy is given by

$$\mathbb{E}[\bar{W}^{SITA}(\mathbf{s})] \approx \sum_{i=1}^{h} f_i^{SITA} s_{i-1}^{-\alpha} s_i^{2-\alpha}, \tag{8}$$

where f_i^{SITA} is given in Lemma 6.1 of [2]. We now provide an analogous result for the normalized mean waiting time of a system that operates under the TAGS policy.

Lemma 14. *When r is large,*

$$\mathbb{E}[W^*] \approx \sum_{i=1}^{h} f_i s_{i-1}^{-\alpha} s_i^{2-\alpha}, \tag{9}$$

where for $i < h$

$$f_i = \frac{2}{\alpha} f_i^{SITA} \tag{10}$$

and

$$f_h = f_h^{SITA}. \tag{11}$$

Proof. We first show different properties that the SITA system and the TAGS system verify when they have the same cutoffs and when r is large:

- We compute the portion of jobs executed in server i in a SITA system, i.e., the probability of a job ranging in size between s_{i-1} and s_i, that for the Bounded Pareto distribution results

$$p_i^{SITA} = \frac{1}{1 - (\frac{1}{p})^\alpha}(s_{i-1}^\alpha - s_i^\alpha). \tag{12}$$

We also compute the portion of jobs which pass through server i in a TAGS system and it results that $\bar{p}_i = \frac{1}{1-(\frac{1}{p})^\alpha}(s_{i-1}^\alpha - p^\alpha)$.

$$\bar{p}_i = \frac{1}{1 - (\frac{1}{p})^\alpha}(s_{i-1}^\alpha - p^\alpha) \tag{13}$$

By (12-13) and since $s_i/s_{i-1} \to \infty$ when $r \to \infty$, we get $p_i^{SITA} \sim \bar{p}_i$ and $p_h^{SITA} = \bar{p}_h$. From the above expressions, it follows that, when r tends to infinity,

$$1 - \frac{p_i^{sita}}{\bar{p}_i} \to s_{i-1}^\alpha s_i^{-\alpha}.$$

- For the arrival rate, it follows from the above reasoning that λ_i^{SITA}, which is the arrival rate to server i of the SITA system, and λ_i, which is the arrival rate of server i of the TAGS system, satisfy the following property: $\lambda_i^{SITA} \approx \lambda_i$.
- The j-th moment of the distribution of the service time of jobs in server i of the TAGS system, that is X_i^j, satisfies that

$$\mathbb{E}[X_i^j] = \frac{p_i^{SITA}}{\bar{p}_i}\mathbb{E}[X_{i,SITA}^j] + (1 - \frac{p_i^{SITA}}{\bar{p}_i})s_i^j, \tag{14}$$

where $\mathbb{E}[X_{i,SITA}^j]$ is the j-th moment of the service time of jobs in server i of the SITA system. The reason for this is that, in the TAGS system, the jobs which pass through server i consist of those which do not pass onto server $i+1$ (since the job size is less than s_i) and those who do (and in this case the service time in server i is s_i). Besides, since the distribution of jobs size is Bounded Pareto, it follows that, if $\alpha \neq 1$ and $j \neq 1$,

$$\mathbb{E}[X_{i,SITA}^j] = \frac{\alpha s_{i-1}^\alpha}{1 - (\frac{s_{i-1}}{s_i})^\alpha} \frac{s_{i-1}^{j-\alpha} - s_i^{j-\alpha}}{\alpha - j} \tag{15}$$

and if $j = 1$ and $\alpha = 1$, it follows that

$$\mathbb{E}[X_{i,SITA}^j] = \frac{s_{i-1}}{1 - (\frac{s_{i-1}}{s_i})} \ln \frac{s_i}{s_{i-1}} \tag{16}$$

- We now shown that the load of the servers for SITA and TAGS coincides in the asympototic regime. From (12),(13), (14) and (15) and the above formula, it follows that the mean service time of jobs in server i satisfies that $\mathbb{E}[X_i^1] \approx \mathbb{E}[X_{i,SITA}^1]$ for $\alpha < 1$ and $i = h$ or for $\alpha > 1$ and any i, whereas for $\alpha < 1$ and $i < h$, $\mathbb{E}[X_i^1] \approx \frac{1}{\alpha}\mathbb{E}[X_{i,SITA}^1]$. Besides, using that (12),(13), (14) and (16), it follows that $\mathbb{E}[X_i^1] \approx \mathbb{E}[X_{i,SITA}^1]$ for $\alpha = 1$. Therefore, since $\lambda_i^{SITA} \approx \lambda_i$, the load of server i of the SITA system, that is ρ_i^{SITA}, and the load of server i of the TAGS system, ρ_i, satisfy that $\rho_i \approx \rho_i^{SITA}$ in the following instances: (i) $\alpha < 1$ and $i = h$, (ii) $\alpha > 1$ and any i and (iii) $\alpha = 1$. On the other hand, for $i < h$ and $\alpha < 1$, we we have that $\rho_i \approx \frac{1}{\alpha}\rho_i^{SITA}$. We know from (59) of [2] that ρ_i^{SITA} tends to zero when $\alpha > 1$ for all $i > 1$ and, by the duality result of Lema 4.1 of [2], it follows that ρ_i^{SITA} tends to zero when $\alpha < 1$ for all $i < h$. Thus, since $\rho_i \approx \frac{1}{\alpha}\rho_i^{SITA}$, ρ_i also tends to zero

when $\alpha < 1$ for all $i < h$. And this implies that ρ_i and ρ_i^{SITA} coincide when r tends to infinity.

- For the second moment of the service time of jobs in server i, we have that $\mathbb{E}[X_i^2] \approx \frac{2}{\alpha}\mathbb{E}[X_{i,SITA}^2]$ for $i < h$ and $\mathbb{E}[X_i^2] \approx \mathbb{E}[X_{i,SITA}^2]$.
- Since the arrivals in both systems are Poisson, we have compute the mean waiting time of jobs in server i using the Pollaczek-Kinchine formula. Let $\mathbb{W}_i^{SITA}$ be the mean waiting time of jobs in the SITA system. Using the above formulas, it follows that, for $i < h$,

$$\mathbb{E}[W_i] \approx \frac{2}{\alpha}\mathbb{E}[W_i^{SITA}],$$

and for $i = h$,

$$\mathbb{E}[W_i] \approx \mathbb{E}[W_i^{SITA}].$$

We recall that, in a TAGS system, a job that finishes service at server i spends an additional time of

$$T_i(\mathbf{s}) = \sum_{j=1}^{i-1} s_j \leq (h-1)s_{i-1},$$

being serviced at servers $1, 2, \ldots, i-1$ and that the average excess service time satisfies that

$$\mathbb{E}[T(\mathbf{s})] = \sum_{i=1}^{h} p_i T_i(\mathbf{s}) \leq (h-1)\mathbb{E}[X]$$

or equivalently

$$\mathbb{E}[T(\mathbf{s})]/\mathbb{E}[X] \leq h-1 \tag{17}$$

Let $W^{SITA}(\mathbf{s})$ be the mean waiting time of jobs in a SITA system. For any vector of cutoffs $\mathbf{s}$, we know that

$$\mathbb{E}[\bar{W}(\mathbf{s})] \geq \mathbb{E}[\bar{W}^*] \geq \min_{\mathbf{s}} E[\bar{W}^{SITA}(\mathbf{s})].$$

By the asymptotic result in [17], it follows that the last term of the above inequality tends to infinity when r is large. This implies that $\mathbb{E}[\bar{W}(\mathbf{s})]$ tends to infinity when r is large, i.e., $\mathbb{E}[W(\mathbf{s})]/\mathbb{E}[X]$ is unbounded when r is large. Hence, in the asymptotic regime $\mathbb{E}[W(\mathbf{s})]/\mathbb{E}[X]$ is unbounded and by (17), we know that $\mathbb{E}[T(\mathbf{s})]/\mathbb{E}[X]$ is bounded. As a result, it follows that $\mathbb{E}[T(\mathbf{s})]$ is asymptotically negligible. Therefore, the mean waiting time of jobs for the TAGS system satisfies that

$$\mathbb{E}[W] \sim \sum_{i=1}^{h} \bar{p}_i \mathbb{E}[W_i], \tag{18}$$

whereas for the SITA system

$$\mathbb{E}[W^{SITA}] \sim \sum_{i=1}^{h} p_i^{SITA}\mathbb{E}[W_i^{SITA}]. \tag{19}$$

Finally, given the asymptotic relation given above between p_i^{SITA} and $\bar{p}_i$ and between $\mathbb{E}[W_i]$ and $\mathbb{E}[W_i^{SITA}]$, using (8) as well as (18) and (19), the desired result follows. □

We now provide the proof of Theorem 9.

Proof. Using (8) and (9), the ratio between the mean waiting time of a system that operates under the TAGS policy and of a system that operates under the SITA policy is given by

$$\frac{\min_{\mathbf{s}} \mathbb{E}[\bar{W}^{SITA}(\mathbf{s})]}{\mathbb{E}[\bar{W}^*]}, \tag{20}$$

when r is large. A simple scaling argument shows that the ratio is independent of r and depends only on the ratios f_i^{TAGS}/f_i^{SITA}, which as we have shown in the previous lemma it is $2/\alpha$ for $i < h$ and one for $i = h$. Therefore, we apply the result of Lemma 5.3 of [2] with $c_i = f_i^{TAGS}/f_i^{SITA}$ and it results that (20) is equal to $\left(\frac{2}{\alpha}\right)^{\mu}$, where $\mu = \frac{(q^{h-1}-1)q}{q^h-1}$ and $q = \frac{\alpha}{2-\alpha}$.

We still need to show that $\left(\frac{2}{\alpha}\right)^{\mu} \leq 2$. If $\alpha \geq 1$, then $\mu \leq 1$ and, therefore, it is clear that $\left(\frac{2}{\alpha}\right)^{\mu} \leq 2$. If $\alpha < 1$, then we have that $q < 1$. This implies that

$$\mu = \frac{(q^{h-1}-1)q}{q^h-1} < q = \frac{\alpha}{2-\alpha}.$$

Differentiating $(\frac{2}{\alpha})^{\frac{\alpha}{2-\alpha}}$ it is easy to verify that it is an increasing function in the interval $(0, 1]$ with value 2 at $\alpha = 1$ and the desired result follows. □

D. Proof of Proposition 11

We first study the case $\alpha > 1$. For this instance, we denote by i the minimal number of servers needed for a system operating under the TAGS policy to be stable. Thus, there exist i cutoffs $\tilde{s}_0, \tilde{s}_1, \ldots, \tilde{s}_{i-1}, \tilde{s}_i$ such that the load of the first i-1 servers is equal to one. The load of jobs whose size is larger than $\tilde{s}_{i-1}$ is less than one. Let $s_{i-1} < \tilde{s}_{i-1}$ such the load of jobs whose size is larger than s_{i-1} is also less than one. We choose $s_1, s_2, \ldots, s_{i-2}$ such that the load of the first $i-1$ servers is the same. Since $s_{i-1} < \tilde{s}_{i-1}$, the first $i-1$ servers are stable. The remaining load is handled by $\tilde{h} = h - i + 1$ servers and, since it is less than 1 and up to scaling the job size distribution is Bounded Pareto, from the result of Theorem 11, we have that the normalized mean waiting time of jobs whose size is larger than s_{i-1}, in the asymptotic regime where $r \to \infty$, is of the same order of magnitude of that of a system operating under the SITA policy, which, according to (67) of [2], is given by (6).

We now focus on the order of magnitud of the first $i-1$ servers for $\alpha > 1$. We observe that $\tilde{s}_{i-1}$ is increasing with r and, since the value of $\tilde{s}_{i-1}$ is bounded by the value $\tilde{s}_{i-1}$ for the (unbounded) Pareto distribution, which is finite. Therefore, the order of magnitude of the first $i-1$ servers is negligible in the asymptotic regime.

For $\alpha < 1$, we use a similar strategy, but we start from the last server. Hence, we define $\tilde{s}_{h-1}$ to be such that the load on the last server is precisely 1. Inductively, given $\tilde{s}_{h-j}$ we define $\tilde{s}_{h-j-1}$ to be such that the load on the $h-j$ server is equal to 1. We can proceed this way to define $\tilde{s}_{\tilde{h}}, \tilde{s}_{\tilde{h}+1}, ..., \tilde{s}_{h-1}$ such that the load of jobs whose size is in the interval $[1, \tilde{s}_{\tilde{h}}]$ is less than one. And using the same arguments as in the case $\alpha > 1$, the desired result follows.

REFERENCES

[1] E. Bachmat and A. Natazon. Analysis of sita queues with many servers and spacetime geometry. *SIGMETRICS Performance Evaluation Review*, 40(3):92–94, 2012.

[2] E. Bachmat and H. Sarfati. Analysis of sita policies. *Performance Evaluation*, 67(2):102 – 120, 2010.

[3] A. Bestavros. Load profiling: a methodology for scheduling real-time tasks in a distributed system. In *Proceedings of 17th International Conference on Distributed Computing Systems*, pages 449–456, May 1997.

[4] J. Broberg, Z. Tari, and P. Zeephongsekul. Task assignment with work-conserving migration. *Parallel Computing*, 32(11-12):808–830, 2006.

[5] M. E. Crovella, M. Harchol-Balter, and C. D. Murta. Task assignment in a distributed system (extended abstract): Improving performance by unbalancing load. In *Proceedings of the 1998 ACM SIGMETRICS Joint International Conference on Measurement and Modeling of Computer Systems*, SIGMETRICS '98/PERFORMANCE '98, pages 268–269, New York, NY, USA, 1998. ACM.

[6] J. Doncel, S. Aalto, and U. Ayesta. Performance degradation in parallel-server systems. *IEEE/ACM Transactions on Networking*, 27(2):875–888, April 2019.

[7] D. Down and S. Meyn. Stability of acyclic multiclass queueing networks. *IEEE transactions on automatic control*, 40(5):916–919, 1995.

[8] M. El-Taha and B. Maddah. Allocation of service time in a multiserver system. *Management Science*, 52(4):623–637, 2006.

[9] D. G. Feitelson, L. Rudolph, U. Schwiegelshohn, K. C. Sevcik, and P. Wong. Theory and practice in parallel job scheduling. In *Workshop on Job Scheduling Strategies for Parallel Processing*, pages 1–34. Springer, 1997.

[10] H. Feng, V. Misra, and D. Rubenstein. Optimal state-free, size-aware dispatching for heterogeneous m/g/-type systems. *Performance Evaluation*, 62(1):475 – 492, 2005. Performance 2005.

[11] R. D. Foley, D. R. McDonald, et al. Join the shortest queue: stability and exact asymptotics. *The Annals of Applied Probability*, 11(3):569–607, 2001.

[12] V. Gupta, M. H. Balter, K. Sigman, and W. Whitt. Analysis of join-the-shortest-queue routing for web server farms. *Performance Evaluation*, 64(9-12):1062–1081, 2007.

[13] M. Harchol-Balter. Task assignment with unknown duration. In *Proceedings 20th IEEE International Conference on Distributed Computing Systems*, pages 214–224. IEEE, 2000.

[14] M. Harchol-Balter. *Performance modeling and design of computer systems: queueing theory in action*. Cambridge University Press, 2013.

[15] M. Harchol-Balter, M. E. Crovella, and C. D. Murta. On choosing a task assignment policy for a distributed server system. *Journal of Parallel and Distributed Computing*, 59(2):204–228, 1999.

[16] M. Harchol-Balter and A. B. Downey. Exploiting process lifetime distributions for dynamic load balancing. *ACM Trans. Comput. Syst.*, 15(3):253–285, Aug. 1997.

[17] M. Harchol-Balter, A. Scheller-Wolf, and A. R. Young. Surprising results on task assignment in server farms with high-variability workloads. *ACM SIGMETRICS Performance Evaluation Review*, 37(1):287–298, 2009.

[18] M. Harchol-Balter and R. Vesilo. To balance or unbalance load in size-interval task allocation. *Probability in the Engineering and Informational Sciences*, 24(2):219–244, 2010.

[19] A. W. Richa, M. Mitzenmacher, and R. Sitaraman. The power of two random choices: A survey of techniques and results. *Combinatorial Optimization*, 9:255–304, 2001.

[20] B. Schroeder and M. Harchol-Balter. Evaluation of task assignment policies for supercomputing servers: The case for load unbalancing and fairness. *Cluster Computing*, 7(2):151–161, 2004.

[21] R. R. Weber. On the optimal assignment of customers to parallel servers. *Journal of Applied Probability*, 15(2):406–413, 1978.

[22] W. Whitt. Deciding which queue to join: Some counterexamples. *Operations Research*, 34(1).

[23] A. Williams, M. Arlitt, C. Williamson, and K. Barker. Web workload characterization: Ten years later. In *Web content delivery*, pages 3–21. Springer, 2005.

[24] W. Winston. Optimality of the shortest line discipline. *Journal of Applied Probability*, 14(1):181–189, 1977.