
This is an electronic reprint of the original article.
This reprint may differ from the original in pagination and typographic detail.

Doncel, Josu; Aalto, Samuli; Ayesta, Urtzi
Performance Degradation in Parallel-Server Systems

Published in:
IEEE/ACM Transactions on Networking

DOI:
[10.1109/TNET.2019.2902531](https://doi.org/10.1109/TNET.2019.2902531)

Published: 01/04/2019

Document Version
Peer-reviewed accepted author manuscript, also known as Final accepted manuscript or Post-print

Please cite the original version:
Doncel, J., Aalto, S., & Ayesta, U. (2019). Performance Degradation in Parallel-Server Systems. *IEEE/ACM Transactions on Networking*, 27(2), 875 - 888. Article 8672485. <https://doi.org/10.1109/TNET.2019.2902531>

This material is protected by copyright and other intellectual property rights, and duplication or sale of all or part of any of the repository collections is not permitted, except that material may be duplicated by you for your research use or educational purposes in electronic or print form. You must obtain permission for any other use. Electronic or print copies may not be offered, whether for sale or otherwise to anyone who is not an authorised user.

Performance Degradation in Parallel-Server Systems

Josu Doncel, Samuli Aalto and Urtzi Ayesta

Abstract—We consider a parallel-server system with homogeneous servers where incoming tasks, arriving at rate λ , are dispatched by n dispatchers, each of them balancing a fraction $1/n$ of the load to K/n servers. Servers are FCFS queues and dispatchers implement SITA-E, a size-based policy such that the servers are equally loaded. We compare the performance of a system with $n > 1$ dispatchers and of a system with a single dispatcher. We show that the performance of a system with n dispatchers, K servers and arrival rate λ coincides with that of a system with one dispatcher, K/n servers and arrival rate λ/n . We define the degradation factor as the ratio between the performance of a system with K servers and arrival rate λ and the performance of a system with K/n servers and arrival rate λ/n . We establish a partial monotonicity on n for the degradation factor and, therefore, the degradation factor is lower-bounded by one. We then investigate the upper-bound of the degradation factor for particular distributions. We consider two continuous service time distributions: uniform and Bounded Pareto; and a discrete distribution with two values, which is the distribution that maximizes the variance for a given mean. We show that the performance degradation is small for uniformly distributed job sizes, but that for Bounded Pareto and two points distributions it can be unbounded. We have investigated the degradation using the distribution obtained from real traces.

Index Terms—Parallel-server routing, Performance degradation, Economies of Scale.

I. INTRODUCTION

We are interested in measuring the performance of parallel-server systems formed by K homogeneous servers. For these systems, the exact analysis of the mean response time of some routing policies such as Join the Shortest Queue is known to be a difficult task and, as a consequence, in this work we focus on a size-based dispatching policy called Size Interval Task Assignment policy with Equal Load (SITA-E) [15]. In the SITA-E scheduling the service time distribution is divided into intervals, all the jobs whose size fall in a given interval are dispatched to the same server and the servers are equally loaded. It is known that the job sizes in data centers have the heavy-tailed property, i.e., a small fraction of jobs make up the half of the load [31]. This property motivates that a SITA based

dispatching policy can be an appropriate design choice for a data center. Indeed, with a heavy-tailed distribution, separating short jobs from long jobs, makes the waiting time of jobs decrease. Another important property of size-based policies with respect to other popular routing policies in the literature, such as Power of two, is that it does not require signaling between dispatchers and servers. Besides, for SITA-E, none of the parallel servers is heavily loaded.

In this work, we compare the performance of SYS-(K,n,\lambda) , which is formed by $n > 1$ dispatchers, where each of them handles a traffic equal to λ/n and balances it to K/n queues, with the performance of SYS-(K,1,\lambda) . In the SYS-(K,n,\lambda) , we assume that the total traffic is shared uniformly among dispatchers with a random assignment policy. As a metric to measure the difference on the performance of these systems, we define the *degradation factor* as the ratio of the mean waiting time of SYS-(K,n,\lambda) over the mean waiting time of SYS-(K,1,\lambda) .

Given the symmetry of SYS-(K,n,\lambda) , it follows that its performance is equal to the performance of $\text{SYS-(K/n,1,\lambda/n)}$. Thus, the analysis of the degradation factor can be interpreted as the economies of scale of a multiserver system when we scale up the number of servers and the arrival rate proportionately.

This work can have a potential impact in the organization of data centers. For instance, in the context of micro data centers, the performance degradation would correspond to the comparison between having a single centralized data center, and multiple small micro data centers located closer to the end users. Within a single large data center, another potential application could be in the design of a data center, which is typically organized in a tree-based topology, and where the dispatcher is often located in the edge nodes [27]. This architecture corresponds to SYS-(K,n,\lambda) . However, if the routing policies are implemented in the core nodes, data centers consist of SYS-(K,1,\lambda) and the performance difference could be assessed using the results of this article.

We assume that the servers are First-Come-First-Served (FCFS), which is a common model, for example, in supercomputing systems [24]. We denote by γ the ratio between the smallest and the largest job size. The main contributions of this work are presented in Table I, where the degradation factor of a system with K servers and n routers is denoted by $D(K, n)$. We first show that, for an arbitrary continuous distribution, if the ratio n_2/n_1 is integer, $D(K, n_1) \leq D(K, n_2)$. From this result, we conclude that the degradation factor is lower-bounded by one and upper-bounded by $D(K, K)$. Therefore, to analyze the maximum performance degradation, we consider three representative distributions. We first study two job size distributions, uniform and Bounded Pareto.

- **Uniform Distribution.** For uniformly distributed job

Research partially supported by the Academy of Finland through the project FQ4BD (Grant No. 296206), by the French "Agence Nationale de la Recherche (ANR)" through the project ANR-15-CE25-0004 (ANR JCJC RACON), by the Project of the Spanish Ministry of Economy and Competitiveness with reference MTM2016-76329-R, by the Marie Skłodowska-Curie grant agreement No 777778 and by the Consolidated Research Group Grant IT649-13 on "Mathematical Modeling, Simulation, and Industrial Applications (M2SI)"

J. Doncel is with University of the Basque Country UPV/EHU, Spain.

S. Aalto is with Department of Communications and Networking at Aalto University, Finland.

U. Ayesta is with CNRS, IRIT, France, with Université de Toulouse, INP, France, with University of the Basque Country UPV/EHU, Spain and with IKERBASQUE - Basque Foundation for Science, Spain.

Part of this work was carried out while J. Doncel was with Inria Grenoble, France.

| | Degradation Factor | Result |
|--|---|--------------------------------|
| Arbitrary continuous distribution | $1 \leq D(K, n) \leq D(K, K)$ | Corollary 1 |
| Uniform distribution: $K \leq \infty$ | $D(K, K) \leq 4/3$. | Proposition 2 Corollary 2 |
| Bounded Pareto distribution: $\alpha = 1$ | $D(K, K) \rightarrow \infty^1$ | Proposition 3 |
| Bounded Pareto distribution: $\alpha \neq 1$ and $K \rightarrow \infty$ | $D(K, K) \rightarrow \infty^1$ | Proposition 4 |
| Bounded Pareto distribution: $\alpha \in (0, 2) \setminus \{1\}$ and K finite | $D(K, K) \leq K^{\frac{1}{1-\alpha}}$. | Proposition 5 |
| Two Point: $K = 2$, Equally Loaded Jobs | $D(K, n) \geq 1$ and $D(K, K) \rightarrow \infty^1$ | Proposition 6 |
| Two Point: $K = 2$, Unequally Loaded Jobs | $D(K, n) \geq 1$ and $D(K, K) \rightarrow \infty^1$ | Proposition 7 Proposition 8 |

TABLE I: Summary of the main results of this article.

sizes and any finite number of servers, we show that the degradation factor is upper-bounded by $4/3$. As a consequence, this upper bound also holds when the number of servers tends to infinity.

- **Bounded Pareto Distribution.** For Bounded Pareto distributed job sizes with parameter $\alpha = 1$, we show that the degradation factor is unbounded from above. We show that the degradation factor is also unbounded from above for Bounded Pareto distributed job sizes with parameter $\alpha \neq 1$ and $K \rightarrow \infty$. When $\alpha \in (0, 2) \setminus \{1\}$ and finite number of servers, assuming that the degradation factor decreases with γ , we prove that the degradation factor is upper bounded by $K^{\frac{1}{1-\alpha}}$.

According to these results, we conclude that the degradation is small for uniformly distributed job sizes, but for Bounded Pareto, which is a useful model for highly variable workloads, the degradation is extremely high when the variability of jobs increases.

We know that for the distributions with bounded and fixed support (i.e., fixed lower and upper bound), the distribution that maximizes the variance (with a given mean) concentrates on these two extreme points. Therefore, we study the degradation factor for a discrete job size distribution that concentrates on two points, the smallest and the largest job size.

- **Two Point Distribution.** For a discrete job sizes distribution that consists of two points, the smallest and the largest job size, we consider a two-server system and, when the load of both types of jobs is equal or unequal, we show that the degradation factor is lower bounded by one and unbounded from above.

Our results show that the degradation can be non negligible and increases as the variability of the distribution increases. We present simulations where we consider the Degenerate Hyperexponential distribution that confirm that as the variability of the service time increases, so does the degradation. Using numerical experiments, we validate the monotonicity assumptions on the degradation factor. We also investigate the

¹We show that there exist parameters of the system such that the degradation tends to infinity.

performance degradation with real traces of parallel machines and the obtained results also confirm the influence of the variability of jobs in the degradation factor.

Given the complexity of the analysis, our modeling assumptions have various limitations. For instance, we study SITA-E dispatching policy rather than SITA policy where the cutoffs optimize the system performance. Unfortunately, the analytical computation of the optimal cutoffs is known to be intractable even for a system with two servers [17], whereas the SITA-E cutoffs for Bounded Pareto distribution are well-known [15]. Therefore, the analysis of SITA-E seems to be a tractable approach that allows us to get insights in the performance degradation of the systems under study.

The rest of the paper is organized as follows. The related work is presented in Section II. In Section III, we describe the model and give some preliminary results. In Section IV, we explore the monotonicity of the degradation factor for an arbitrary continuous job sizes distribution. We study the degradation factor for uniformly distributed job sizes in Section V and, in Section VI, for Bounded Pareto distributed job sizes. Then, in Section VII we analyze the degradation for a discrete job sizes distribution that concentrates on two points. We present the numerical experiments in Section VIII and the degradation factor analysis with real traces of different data centers in Section IX. Finally, in Section X, the main conclusions of this work are drawn. The proofs that do not appear in the body of the article are available in the Supplementary Material.

An earlier version of this paper appeared in [7].

II. RELATED WORK

Many researchers in Computer Science have been interested in analyzing how to balance the load in a system with parallel queues optimally, that is, in order to minimize a certain objective function, for example the mean response time of the incoming jobs, see the survey [25] and the book [14]. The typical architecture of the routing policies that are studied in the literature is formed by one dispatcher that receives all the incoming traffic, which distributes the load among the set of servers. In the Join-the-Shortest-Queue [11], [12] the

dispatcher sends the flow to the queue with less customers. This routing policy is very popular since it minimizes the mean response times of jobs when the number of customers in all the servers is known and the service time distribution is exponential or has a non-decreasing hazard rate [28], [32]. Another important routing policy is the Power of Two, where for all incoming jobs, the dispatcher selects two servers independently and uniformly at random and applies the Join-the-Shortest-Queue policy among the chosen servers [20], [22]. Join-Idle-Queue is also a very popular dispatching rule, where processors inform dispatchers when they are idle [19]. When the service demand is known but the queues are non-observable and the servers are FCFS, the SITA policy with optimal thresholds is shown to optimize the performance of the system [10]. In this policy, each host serves jobs whose service demand is in a designated range. The SITA-E policy has been introduced in [15] and, under this routing policy, the cutoffs are chosen to equalize the load in all the servers. This dispatching policy has been also studied by [6], where the authors apply SITA-E to web server farms. In [13] the author introduces the task assignment by guessing size, which is a variant of SITA-E policy where knowledge of the job sizes is not required. Under the SITA routing policy with optimal thresholds, asymptotic analysis for the Bounded Pareto distribution has been done in [4], [26]. The authors in [16] consider a system where the coefficient of variation of incoming tasks is high and they show that the performance of SITA can be much worse than the performance of the Least-Work-Left policy. Another related work is [17], where authors consider a two server system and they give conditions that establish in which direction the load should be unbalanced in order to optimize the performance. Furthermore, for Bounded Pareto distributed job sizes, they show that when (i) $\alpha < 1$, the short job server must be underloaded, (ii) $\alpha = 1$, the load is equally balanced and (iii) $\alpha > 1$, the long job server must be underloaded. The analysis of the economies of scaling the arrival rate and the number of servers proportionally in multiserver systems has been previously done in the literature, but in different contexts of our work. The author in [29] analyzes the optimal server utilization and he provides a simple approximation of the mean steady state waiting time. In [30] the author considers an M/M/K queue and analyzes the economies of scale for different performance measures.

The problem of how to balance the load in a server farm has been extensively studied also in the context of game theory, see [2], [5], [8], [18], [21], [23]. An important assumption in these models is that jobs can decide individually where to get service.

III. MODEL DESCRIPTION

We consider a system with K servers with equal capacity and n dispatchers. We denote by x_m and x_M the minimum and maximum size of the incoming jobs to the system and by λ the total incoming traffic to the system. The traffic that each dispatcher controls arrives to the system according to a Poisson process of rate λ/n and it is balanced to K/n servers. The servers are FCFS queues and the dispatchers

implement the SITA-E routing policy. We assume that service times of incoming jobs form an i.i.d. sequence with a common distribution denoted by X , and let $\mathbb{E}(X)$ and $\mathbb{E}(X^2)$ denote its first and second moment, respectively. Let $F(x) = \mathbb{P}(X \leq x)$ denote the service time distribution. We assume $F(x)$ to be differentiable and we denote $f(x) = \frac{dF(x)}{dx}$. The total load in the system is denoted by $\rho = \lambda \cdot \mathbb{E}(X)/K$. For stability reasons, we assume $\rho < 1$.

We denote by $W(K, n, x_m, x_M, \lambda)$ the random variable corresponding to the steady-state waiting time of jobs in SYS-(K,n, λ). We define the degradation factor as follows:

$$D(K, n, x_m, x_M) = \frac{\mathbb{E}(W(K, n, x_m, x_M, \lambda))}{\mathbb{E}(W(K, 1, x_m, x_M, \lambda))}.$$

In SYS-(K,n, λ) there are n groups and, in each group, there are K/n servers. It is assumed that n is a divisor of K and that the total traffic is shared uniformly among dispatchers with a random assignment policy. Moreover, the traffic that each dispatcher of SYS-(K,n, λ) handles is the same and equal to λ/n and every dispatcher applies SITA-E policy. Thus, SYS-(K,n, λ) and SYS-(K/n,1, λ/n) have the same performance, and the degradation factor can be written as:

$$D(K, n, x_m, x_M) = \frac{\mathbb{E}(W(\frac{K}{n}, 1, x_m, x_M, \frac{\lambda}{n}))}{\mathbb{E}(W(K, 1, x_m, x_M, \lambda))}. \quad (1)$$

From (1), the performance degradation study of this article can be interpreted as a queueing problem that consists of evaluating the economies of scale a parallel-server system when we scale up the arrival rate and the number of servers proportionally. We have not included λ as a parameter of the degradation factor since, as we will see in Section III-E, the degradation factor does not depend on the arrival rate. We define by $\gamma = \frac{x_m}{x_M} \in [0, 1]$ the ratio between the shortest and the largest job size.

Remark 1 (Randomized Load Balancing). *As an example, let us calculate (1) in the case of a load balancing scheme without sized-based information. We consider a system with K homogeneous servers and one dispatcher that operates under Bernoulli routing policy. The probability of a job to be executed in a given server is $1/K$ and, therefore, the arrival rate to that server is λ/K . Thus, we obtain that the mean waiting time of jobs in this system is $\frac{(\lambda/K) \mathbb{E}(X^2)}{2(1-\rho)}$. We now consider a system with K/n homogeneous servers and an incoming traffic λ/n . We observe that the probability of a job to be executed in a given server is n/K and the arrival rate to that server is λ/K . Hence, the mean waiting time in this system is also $\frac{(\lambda/K) \mathbb{E}(X^2)}{2(1-\rho)}$. As a result, the degradation factor for randomized load balancing policies is equal to one.*

From Pollaczek-Khinchine formula, we know that the waiting time of jobs depends on the second moment, which is related to the variability of the service time distribution. With SITA-E, as the number of servers increases, the size variability in each server decreases. Hence, we can expect the performance of SYS-(K,n, λ) to be worse than that of SYS(K,1, λ). Likewise, when x_m and x_M coincide, the jobs arrive to the system following a deterministic distribution. Therefore, size-

based scheduling can not improve the performance and there is no performance degradation in this case.

Lemma 1. *If $x_m = x_M$, then the performance degradation is equal to one.*

From (1), we see that to analyze the degradation factor we need to compare two systems with one dispatcher, where one system has K servers and arrival rate λ , so SYS-(K,1, λ), and the other K/n servers and arrival rate λ/n , so SYS-(K/n,1, λ/n). Therefore, we analyze the performance of a generic SYS-(R,1, $\bar{\lambda}$), which is a system with one dispatcher, R servers and arrival rate $\bar{\lambda}$. Prior to that, we present the SITA-E routing policy for SYS-(R,1, $\bar{\lambda}$).

A. SITA-E Routing

If the size of incoming jobs is known, the dispatcher can perform the so-called SITA routing, that, roughly speaking, separates jobs of different sizes to be executed in different servers. In other words, the SITA routing is 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 server. In a system with R servers, there are $R+1$ thresholds c_0, c_1, \dots, c_R satisfying that $x_m = c_0 < c_1 < \dots < c_{R-1} < c_R = x_M$ and jobs ranging in size from c_{j-1} to c_j are executed in server j .

When the SITA policy is implemented in a multiserver system, its performance is affected by the way we choose the thresholds. One might choose, for example, the thresholds that minimize the response time of jobs. In this work, we assume that the dispatcher carries out the SITA-E routing, that is a particular SITA routing, where the thresholds are chosen so as to Equalize the load of the servers. The main advantage of this routing policy is that the thresholds can be easily obtained using the following expression:

$$\int_{x_m}^{c_1} x f(x) dx = \int_{c_1}^{c_2} x f(x) dx = \dots = \int_{c_{R-1}}^{x_M} x f(x) dx, \quad (2)$$

whereas the computation of the thresholds of the optimal SITA policy is known to be intractable even for a system with two servers [17].

We now observe that if (2) is satisfied, then the load of all the server is equal. In fact, the load in server j is given by

$$\bar{\lambda} \cdot (F(c_j) - F(c_{j-1})) \cdot \int_{c_{j-1}}^{c_j} x \frac{f(x)}{F(c_j) - F(c_{j-1})} dx,$$

and, thus, (2) implies that the load is the same in each server. Throughout this article, we use the notion of scaled thresholds, which are defined as $z_j = c_j/x_M$. We note that, in the particular cases where $j = 0$ and $j = R$, we have respectively that $z_0 = \gamma$ and $z_R = 1$.

B. Waiting Time in SYS-(R,1, $\bar{\lambda}$)

We study SYS-(R,1, $\bar{\lambda}$), which is a system that consists of R servers, one dispatcher that implements SITA-E load balancing and arrival rate $\bar{\lambda}$. As we said before, in this system there are $R+1$ thresholds. We denote the j -th threshold by c_j .

Since the probability of a job to be executed in server j is $F(c_j) - F(c_{j-1})$, the mean waiting time of this system is

$$\mathbb{E}(W(R, 1, x_m, x_M, \bar{\lambda})) = \frac{\bar{\lambda}}{2(1-\bar{\rho})} \sum_{j=1}^R (F(c_j) - F(c_{j-1}))^2 \mathbb{E}(X_j^2), \quad (3)$$

where $\bar{\rho} = \frac{\bar{\lambda} \mathbb{E}(X)}{R}$ and $\mathbb{E}(X_j^2)$ is the second moment of the service time distribution of the tasks executed in server j .

Using conditional probabilities, we obtain that the second moment of the jobs to be executed in server j is

$$\mathbb{E}(X_j^2) = \int_{c_{j-1}}^{c_j} x^2 \frac{f(x)}{F(c_j) - F(c_{j-1})} dx. \quad (4)$$

Therefore, using (3), (4), we obtain the following expression for the mean waiting time of SYS-(R,1, $\bar{\lambda}$) for continuously distributed job sizes:

$$\mathbb{E}(W(R, 1, x_m, x_M, \bar{\lambda})) = \frac{\bar{\lambda}}{2(1-\bar{\rho})} \sum_{j=1}^R (F(c_j) - F(c_{j-1})) \cdot \int_{c_{j-1}}^{c_j} x^2 f(x) dx. \quad (5)$$

C. Continuous Distributions: Uniform and Bounded Pareto

In this paper, we consider two continuous job size distributions: the uniform and the Bounded Pareto. For uniformly distributed job sizes, if $x_m \leq x \leq x_M$, we have that $f(x) = \frac{1}{x_M - x_m}$, and $f(x) = 0$ otherwise. Furthermore, the cumulative distributed function of the job sizes is

$$F(x) = \begin{cases} 0, & x \leq x_m, \\ \frac{x - x_m}{x_M - x_m}, & x_m \leq x \leq x_M, \\ 1, & x \geq x_M. \end{cases}$$

The thresholds of SYS-(R,1, $\bar{\lambda}$) can be obtained from (2) and using that $f(x) = \frac{1}{x_M - x_m}$, for all $x \in [x_m, x_M]$, and are given by $c_j = \sqrt{\frac{(R-j)x_m^2 + jx_M^2}{R}}$, $j = 0, \dots, R$.

For Bounded Pareto distributed job sizes with parameter α , we have that, if $x_m \leq x \leq x_M$, $f(x) = \frac{\alpha x_m^\alpha}{1 - (x_m/x_M)^\alpha} x^{-\alpha-1}$, and $f(x) = 0$ otherwise. The cumulative distributed function of the job sizes is

$$F(x) = \begin{cases} 0, & x \leq x_m, \\ \frac{1 - (x_m/x)^\alpha}{1 - (x_m/x_M)^\alpha}, & x_m \leq x \leq x_M, \\ 1, & x \geq x_M. \end{cases}$$

The value of the thresholds for Bounded Pareto distributed job sizes of SYS-(R,1, $\bar{\lambda}$) is given in [15] and it is $c_j = \left(\frac{R-j}{R} x_m^{1-\alpha} + \frac{j}{R} x_M^{1-\alpha} \right)^{\frac{1}{1-\alpha}}$ if $\alpha \neq 1$ and $c_j = x_m \left(\frac{x_M}{x_m} \right)^{\frac{j}{R}}$ if $\alpha = 1$.

In the rest of the article, we denote by $D_U(K, n, x_m, x_M)$ and $D_{BP(\alpha)}(K, n, x_m, x_M)$ the degradation factor when the job sizes are uniformly distributed and Bounded Pareto distributed with parameter α , respectively. Since, in both cases, the degradation factor depends on x_m and x_M only through $\gamma = \frac{x_m}{x_M}$ (see Lemma 5 and Lemma 7), we also use the notation $D_U(K, n, \gamma)$ and $D_{BP(\alpha)}(K, n, \gamma)$.

D. Discrete Distributions

Here we assume the incoming job sizes follow a discrete distribution. We first assume that the job sizes are distributed in two points and hence with probability p an incoming task is of size x_m and with probability $1 - p$ it is of size x_M . The jobs of size x_m (resp. of size x_M) are said to be short jobs (resp. long jobs). Since the distribution under consideration is discrete, (2) does not determine the load balancing for this distribution. Therefore, we define how the load is balanced in SYS-(R,1, $\bar{\lambda}$) when the job sizes are distributed in two points.

Let $l = \frac{R}{1 + \frac{(1-p)x_M}{px_m}}$. If l is integer, we have that the short jobs are executed in l servers and the load is balanced among these servers using the Bernoulli routing policy. On the other hand, the long jobs are executed in $R - l$ servers, where it is also applied the Bernoulli scheduling. Indeed,

$$l = \frac{R}{1 + \frac{(1-p)x_M}{px_m}} \iff \frac{px_m}{l} = \frac{(1-p)x_M}{R-l},$$

and, as a consequence, the load in all the servers is the same. If l is not integer, we have three different possibilities:

- If $l > R - 1$, there is one server that executes all the long jobs and a proportion p_1 of short jobs. In the rest of the servers only short jobs are executed. The value of p_1 is chosen so as to equalize the load of the servers, that is, it is the solution of the following equation:

$$\frac{(1-p_1)px_m}{R-1} = p_1px_m + (1-p)x_M.$$

- If $l < 1$, there is one server that executes all the short jobs and a proportion p_2 of long jobs. In the rest of the servers only long jobs are executed. The value of p_2 is chosen so as to equalize the load of the servers, that is, it is the solution of the following equation:

$$px_m + p_2(1-p)x_M = \frac{(1-p_2)(1-p)x_M}{R-1}.$$

- If $1 < l < R - 1$, there are $[l]$ servers that execute only short jobs and $R - [l]$ servers that execute only long jobs, while in the other server a proportion p_1 of short jobs and a proportion p_2 of long jobs. The values of p_1 and p_2 are chosen in order to equalize the load of the servers, that is,

$$\frac{(1-p_1)px_m}{[l]} = p_1px_m + p_2(1-p)x_M = \frac{(1-p)(1-p_1)x_M}{R-[l]}.$$

We analyze in Section VII the degradation factor when the job sizes is distributed in two points and we denote it by $D_{TP(l)}(K, n, x_m, x_M)$. In Section IX, we analyze the degradation factor with real traces and we consider that the job sizes distribution coincides with the jobs that has been submitted to real data centers. For these cases, the job sizes follow a discrete distribution with more than two points characterized by a vector of job sizes $\mathbf{x} = (x_m, \dots, x_M)$ and a probability distribution \mathbf{p} .

² $[x]$ and $\lceil x \rceil$ denote respectively the floor and the ceil of $x \in \mathbb{R}$.

E. Preliminary Results

We now present that, using the results of Section III-B, we can give the expression for the degradation factor when the job sizes are continuously distributed. We first observe that, from (3), we can obtain the mean waiting time of SYS-(K,1, λ) when $R = K$ and $\bar{\lambda} = \lambda$ and the mean waiting time of SYS-(K/n,1, λ/n) when $R = K/n$ and $\bar{\lambda} = \lambda/n$. Besides, for both systems, \bar{p} coincides and the factor $\frac{\lambda}{2(1-\bar{p})}$ appears in the numerator and denominator of the degradation factor. Hence, we conclude that the degradation factor does not depend on the arrival rate λ .

Let x_0, \dots, x_K denote the thresholds of SYS-(K,1, λ) and $y_0, \dots, y_{K/n}$ denote the thresholds of SYS-(K/n,1, λ/n). Substituting these values in (5) it results:

$$D(K, n, x_m, x_M) = \frac{1 \sum_{j=1}^{K/n} (F(y_j) - F(y_{j-1})) \left(\int_{y_{j-1}}^{y_j} x^2 f(x) dx \right)}{n \sum_{j=1}^K (F(x_j) - F(x_{j-1})) \left(\int_{x_{j-1}}^{x_j} x^2 f(x) dx \right)}. \quad (6)$$

As it can be observed, the degradation factor depends on the thresholds of SYS-(K/n,1, λ/n) and of SYS-(K,1, λ).

We now show that the thresholds of both systems are related for continuously distributed job sizes.

Lemma 2. *If $f(x) > 0$ for all $x \in [x_m, x_M]$, then $y_j = x_{n \cdot j}$.*

From this result and (6), it follows directly the expression for the degradation factor for continuously distributed job sizes.

Proposition 1. *If $f(x) > 0$ for all $x \in [x_m, x_M]$,*

$$D(K, n, x_m, x_M) = \frac{1 \sum_{j=1}^{K/n} (F(x_{n \cdot j}) - F(x_{n \cdot (j-1)})) \left(\int_{x_{n \cdot (j-1)}}^{x_{n \cdot j}} x^2 f(x) dx \right)}{n \sum_{j=1}^K (F(x_j) - F(x_{j-1})) \left(\int_{x_{j-1}}^{x_j} x^2 f(x) dx \right)}, \quad (7)$$

where the thresholds $x_m = x_0, x_1, \dots, x_{K-1}, x_K = x_M$ satisfy

$$\int_{x_m}^{x_1} x f(x) dx = \int_{x_1}^{x_2} x f(x) dx = \dots = \int_{x_{K-1}}^{x_M} x f(x) dx.$$

Let $k = \frac{K}{n}$. In the following result, we show the properties that the probability and the second moment of jobs executed in the servers satisfy.

Lemma 3. *Let $p_j = F(x_{n \cdot j}) - F(x_{n \cdot (j-1)})$ and $s_j = \int_{x_{n \cdot (j-1)}}^{x_{n \cdot j}} x^2 f(x) dx$, for $j = 1, \dots, k$. Then,*

- $p_1 + \dots + p_k = 1$,
- $p_1 \geq \dots \geq p_k \geq 0$,
- $s_k \geq \dots \geq s_2 \geq s_1 \geq 0$.

IV. MONOTONICITY ON n

As we explained in Section III, jobs are split in such a way that the variability of jobs decreases when the number of

servers increases. Hence, one can think that the degradation factor $D(K, n, x_m, x_M)$ increases with n . In this section we deal with this monotonicity property.

We consider the values of s_1, \dots, s_k and p_1, \dots, p_k of Lemma 3. The key result to prove the monotonicity property is the following:

Lemma 4. *Let $k \geq 1$ and s_1, \dots, s_k and p_1, \dots, p_k as defined in Lemma 3. Then*

$$\sum_{i=1}^k p_i s_i \leq \frac{1}{k} \sum_{i=1}^k s_i.$$

This result can be interpreted in the following way: the average of the values s_1, \dots, s_k is always larger than its weighted sum when $p_1 \geq \dots \geq p_k \geq 0$ and $s_k \geq \dots \geq s_1 \geq 0$.

Let n and m such that $1 \leq n \leq m \leq k$. We now write $q_i = \frac{p_{n+i-1}}{\sum_{j=n}^m p_j}$ and $t_i = s_{n+i-1}$, for $i = 1, \dots, m - n + 1$. Hence, from the above result, it follows that

$$\sum_{i=1}^{m-n+1} q_i t_i \leq \frac{1}{m-n+1} \sum_{i=1}^{m-n+1} t_i.$$

Therefore,

$$\sum_{i=n}^m \frac{p_i}{\sum_{j=n}^m p_j} s_j \leq \frac{1}{m-n+1} \sum_{i=n}^m s_i, \quad (8)$$

where the result coincides with that of Lemma 4 when $n = 1$ and $m = k$. In the following result we show that the degradation factor is partially monotone with n .

Theorem 1. *Let $n_1 \leq n_2$ and assume that $K/n_1, K/n_2$ and n_2/n_1 are integers. If $f(x) > 0$ for all $x \in [x_m, x_M]$, then*

$$D(K, n_1, x_m, x_M) \leq D(K, n_2, x_m, x_M).$$

Proof. Let $k_1 = K/n_1, k_2 = K/n_2$ and $l = n_2/n_1$. For any $i = 1, \dots, k_2, \bar{p}_i = \frac{1}{l} \sum_{j=(i-1)l+1}^{il} p_j$. By Proposition 1, we have that $D(K, n_1, x_m, x_M) \leq D(K, n_2, x_m, x_M)$ if and only if

$$\sum_{i=1}^{k_1} p_i s_i \leq \sum_{i=1}^{k_2} \sum_{j=1}^l \bar{p}_i s_{(i-1)l+j}.$$

And the desired result follows from (8). \square

We now present the interesting conclusions that we obtain from this result. First, if we apply this result to $n_1 = 1$ and $n_2 = n$, we conclude the degradation factor is always lower-bounded by one, which means that there exists always degradation when we compare the performance of SYS-(K,n, λ) with that of SYS-(K,1, λ). Another important conclusion of the previous result is obtained if we assume that $n_1 = n$ and $n_2 = K$. Indeed, for this case, we show that the degradation factor is upper-bounded by $D(K, K, x_m, x_M)$. Hence, the following result is a direct consequence of Theorem 1.

Corollary 1. *If $f(x) > 0$ for all $x \in [x_m, x_M]$, then*

$$1 \leq D(K, n, x_m, x_M) \leq D(K, K, x_m, x_M).$$

The following sections are devoted to analyzing the upper-bound $D(K, K, x_m, x_M)$ for specific distributions. First, we consider uniformly distributed job sizes and, then, Bounded Pareto distributed job sizes. Finally, we explore the degradation factor for two points distributed job sizes, i.e., when there are only two types of jobs: small ones and large ones. In Section IX, we consider discrete distributions with more than two points and we present instances where the degradation factor is not monotone on n when the ratio n_2/n_1 is not an integer. However, Corollary 1 still holds for these cases.

V. UNIFORMLY DISTRIBUTED JOB SIZES

We focus on the degradation factor when the job sizes are uniformly distributed. We observe that this distribution satisfies that $f(x) > 0$ for all $x \in [x_m, x_M]$. Therefore, we can use the result of Corollary 1 to state that the degradation factor for uniformly distributed job sizes is lower-bounded by one and upper-bounded by $D_U(K, K, \gamma)$. In this section, we show that, for any value of K , the degradation factor is upper bounded by $4/3$.

It is trivial to check that the scaled thresholds are

$$z_j = \sqrt{\frac{(K-j)\gamma^2 + j}{K}}, \quad j = 0, \dots, K.$$

In the following result, we give an expression of the degradation factor for uniformly distributed job sizes, which, as expected, depends on x_m and x_M only through γ .

Lemma 5. *The degradation factor for uniformly distributed job sizes only depends on K, n and γ and it is given by*

$$D_U(K, n, \gamma) = \frac{1}{n} \frac{\sum_{j=1}^{K/n} (z_{n \cdot j} - z_{n \cdot (j-1)}) (z_{n \cdot j}^3 - z_{n \cdot (j-1)}^3)}{\sum_{j=1}^K (z_j - z_{j-1}) (z_j^3 - z_{j-1}^3)}. \quad (9)$$

We first study the degradation factor when the number of servers is finite. Our goal is to give an upper bound on $D_U(K, K, \gamma)$ for a finite K . First, we present the following lemma.

Lemma 6. *For any $x \geq 0$ and $h \geq 0$,*

$$\left((x+h)^{\frac{1}{2}} - x^{\frac{1}{2}} \right) \left((x+h)^{\frac{3}{2}} - x^{\frac{3}{2}} \right) \geq \frac{3}{4} h^2.$$

Since $z_i^2 - z_{i-1}^2 = \frac{1-\gamma^2}{K}$, it follows from Lemma 6 that each term of the denominator of $D_U(K, K, \gamma)$ is lower bounded by $\frac{3}{4} \frac{(1-\gamma^2)^2}{K^2}$ and, as a result,

$$D_U(K, K, \gamma) \leq \frac{(1-\gamma)(1-\gamma^3)}{K \sum_{i=1}^K \frac{3}{4} \frac{(1-\gamma^2)^2}{K^2}} = \frac{4}{3} \frac{(1-\gamma)(1-\gamma^3)}{(1-\gamma^2)^2}.$$

We now note that

$$\frac{(1-\gamma)(1-\gamma^3)}{(1-\gamma^2)^2} = \frac{(1+\gamma+\gamma^2)}{(1+\gamma)^2} = 1 - \frac{\gamma}{(1+\gamma)^2} \leq 1,$$

which implies that $D_U(K, K, \gamma) \leq \frac{4}{3}$.

Proposition 2. For $K < \infty$ and uniformly distributed job sizes, the degradation factor is upper bounded by $4/3$.

A direct consequence of this result is that the degradation factor is also upper bounded by $4/3$ when $K \rightarrow \infty$.

Corollary 2. For $K \rightarrow \infty$ and uniformly distributed job sizes, the degradation factor is upper bounded by $4/3$.

VI. BOUNDED PARETO DISTRIBUTED JOB SIZES

In this section, we concentrate on the degradation factor for Bounded Pareto distributed job sizes. We observe that this distribution satisfies that $f(x) > 0$ for all $x \in [x_m, x_M]$. Therefore, we can use the results of Corollary 1 to state that the degradation factor for uniformly distributed job sizes is lower-bounded by one and upper-bounded by $D_{BP(\alpha)}(K, K, \gamma)$. For $\alpha = 1$ and for $\alpha \neq 1$ and $K \rightarrow \infty$, we show that the degradation factor is unbounded from above. For $\alpha \in (0, 2) \setminus \{1\}$ and finite K , assuming that $D_{BP(\alpha)}(K, K, \gamma)$ decreases as γ increases, we show that the degradation factor is upper bounded by $K^{\frac{1}{1-\alpha}}$.

We now present the values of the scaled thresholds for Bounded Pareto distributed job sizes:

$$z_j = \begin{cases} \left(\frac{K-j}{K} + \frac{j}{K} \gamma^{1-\alpha} \right)^{\frac{1}{1-\alpha}}, & \alpha \neq 1, \\ \gamma^{1-\frac{j}{K}}, & \alpha = 1. \end{cases} \quad (10)$$

In the following result, we give an expression of the degradation factor for Bounded Pareto distributed job sizes, which also depends on x_m and x_M only through γ .

Lemma 7. The degradation factor for Bounded Pareto distributed job sizes only depends on K , n , α and γ and it is given by

$$D_{BP(\alpha)}(K, n, \gamma) = \frac{1}{n} \frac{\sum_{j=1}^{K/n} (z_{n \cdot j}^{2-\alpha} - z_{n \cdot (j-1)}^{2-\alpha})(z_{n \cdot (j-1)}^{-\alpha} - z_{n \cdot j}^{-\alpha})}{\sum_{j=1}^K (z_j^{2-\alpha} - z_{j-1}^{2-\alpha})(z_{j-1}^{-\alpha} - z_j^{-\alpha})}. \quad (11)$$

A. The case $\alpha = 1$

We first analyze the degradation factor for Bounded Pareto distributed job sizes with $\alpha = 1$. As we said in Section II, the authors in [17] show that SITA-E optimizes the performance of a system with two servers and Bounded Pareto distributed jobs sizes with $\alpha = 1$. From Lemma 7 and (10) and simplifying, it results that

$$D_{BP(1)}(K, n, \gamma) = \frac{1}{n^2} \cdot \frac{\gamma^{\frac{-n}{K}} (1 - \gamma^{\frac{n}{K}})^2}{\gamma^{\frac{-1}{K}} (1 - \gamma^{\frac{1}{K}})^2}. \quad (12)$$

We show that this expression decreases with γ .

Lemma 8. $D_{BP(1)}(K, n, \gamma)$ is a decreasing function of γ .

Using the previous result and noting, from (12), that $D_{BP(1)}(K, K, \gamma)$ tends to infinity when $\gamma \rightarrow 0$, we give the following result.

Proposition 3. $D_{BP(1)}(K, K, \gamma)$ tends to infinity when $\gamma \rightarrow 0$.

From this result, we state that the performance of $\text{SYS}(K/n, 1, \lambda/n)$ is, in the worst case, infinite times worse than the performance of $\text{SYS}(K, 1, \lambda)$. In fact, this ratio equals infinity when $\gamma \rightarrow 0$, in which case the Bounded Pareto distribution is very skewed and the variance goes to infinity.

B. The case $\alpha \neq 1$

We now study the degradation factor for Bounded Pareto distributed job sizes with $\alpha \neq 1$. We first assume that the number of servers is infinite and then we analyze the case of a finite number of servers.

1) *Degradation Factor when $K \rightarrow \infty$ and $n = K$:* We first focus on the case $n = K$ and $K \rightarrow \infty$. The scaled thresholds satisfy that, when $K \rightarrow \infty$,

$$\sum_{i=1}^K (z_i^{-\alpha} - z_{i-1}^{-\alpha})(z_i^{2-\alpha} - z_{i-1}^{2-\alpha}) \rightarrow \int_{\gamma^{1-\alpha}}^1 (y^{\frac{-\alpha}{1-\alpha}})' (y^{\frac{2-\alpha}{1-\alpha}})' dy (1-\gamma^{1-\alpha}) = \frac{\alpha \cdot (2-\alpha)}{-(1-\alpha)^2} (1-\gamma^{1-\alpha})^2,$$

since $z_i = (\gamma^{1-\alpha} + \frac{j}{K}(1-\gamma^{1-\alpha}))^{\frac{1}{1-\alpha}}$. Therefore, the degradation factor for Bounded Pareto distributed job sizes and $K \rightarrow \infty$ and $n = K$ is

$$D_{BP(\alpha)}(\infty, \infty, \gamma) = \frac{-(1-\alpha)^2}{\alpha \cdot (2-\alpha)} \cdot \left(1 - \frac{(1-\gamma)^2}{\gamma^\alpha (1-\gamma^{1-\alpha})^2} \right). \quad (13)$$

We now show that the above expression is decreasing with γ .

Lemma 9. $D_{BP(\alpha)}(\infty, \infty, \gamma)$ is a decreasing function of γ .

From (13), it follows that the degradation factor is unbounded from above when $\gamma \rightarrow 0$.

Proposition 4. $D_{BP(\alpha)}(\infty, \infty, \gamma)$ tends to infinity when $\gamma \rightarrow 0$.

2) *Degradation Factor when K is finite:* We now analyze the degradation factor when K is finite and we first give the value of $D_{BP(\alpha)}(K, n, \gamma)$ when $\gamma \rightarrow 0$, i.e., when the difference between x_m and x_M tends to infinity.

Lemma 10. If $\alpha \in (0, 2) \setminus \{1\}$,

$$\lim_{\gamma \rightarrow 0} D_{BP(\alpha)}(K, n, \gamma) = n^{\frac{1}{1-\alpha}}.$$

It is important to note that, when $\gamma \rightarrow 0$, the degradation factor for Bounded Pareto distributed job sizes with $\alpha \neq 1$ does not depend on K . Besides, $n^{\frac{1}{1-\alpha}}$ is infinite when $\alpha \rightarrow 1$ for any n . Therefore, we conclude from Proposition 3 and Lemma 10 that the limits when γ goes to zero and when α tends to one interchange for Bounded Pareto job sizes distribution, i.e.,

$$\lim_{\gamma \rightarrow 0} \lim_{\alpha \rightarrow 1} D_{BP(\alpha)}(K, n, \gamma) = \lim_{\alpha \rightarrow 1} \lim_{\gamma \rightarrow 0} D_{BP(\alpha)}(K, n, \gamma).$$

We assume that $D_{BP(\alpha)}(K, K, \gamma)$ with $\alpha \neq 1$ and for any K finite decreases with γ . Given the difficulty of the expression (11) as well as the scaled thresholds (10), we have not

succeeded in showing this monotonicity property. We have performed many numerical experiments to conjecture that the degradation factor decreases with γ when $\alpha \neq 1$ and $n = K$.

Conjecture 1. *When $\alpha \in (0, 2) \setminus \{1\}$, and for any finite K , $D_{BP(\alpha)}(K, K, \gamma)$ is a decreasing function of γ .*

Under this assumption, we conclude that, for this case,

$$D(K, K, \gamma) \leq \lim_{\gamma \rightarrow 0} D(K, K, \gamma) = K^{\frac{1}{|1-\alpha|}},$$

where the last equality is given by Lemma 10.

Proposition 5. *Assume Conjecture 1 holds. Then, the degradation factor for Bounded Pareto distributed job sizes with $\alpha \in (0, 2) \setminus \{1\}$, and K finite is upper-bounded by $K^{\frac{1}{|1-\alpha|}}$.*

We observe that $K^{\frac{1}{|1-\alpha|}}$ is infinite when K tends to infinity. Therefore, we conclude from Proposition 5 and Proposition 4 that the limits when γ goes to zero and K goes to infinity interchange, that is,

$$\lim_{\gamma \rightarrow 0} \lim_{K \rightarrow \infty} D_{BP(\alpha)}(K, K, \gamma) = \lim_{K \rightarrow \infty} \lim_{\gamma \rightarrow 0} D_{BP(\alpha)}(K, K, \gamma).$$

VII. TWO POINT DISTRIBUTED JOB SIZES

In this section, we assume that the job sizes are distributed in two points with parameter p , i.e., $p = \mathbb{P}(X = x_m)$ and $\mathbb{P}(X = x_M) = 1 - p$. We recall that the load balancing of SYS-(K,1, λ) for this distribution depends on $l = \frac{K}{1 + \frac{(1-p)x_M}{px_m}}$.

A. The Case $K = 2$

We first study the degradation factor for this distribution in a two-server system. Hence, we aim to compare the performance of SYS-(2,1, λ) with the performance of SYS-(1,1, $\lambda/2$). SYS-(1,1, $\lambda/2$) is an M/G/1 queue with arrival rate $\lambda/2$ and, according to the Pollaczek-Khinchine formula, its expected waiting time is $\frac{\frac{\lambda}{2} \mathbb{E}(X^2)}{2(1-\rho)}$. We now analyze the degradation factor for different values of l .

1) *Equally Loaded Jobs ($l = 1$):* We assume that $l = 1$, which occurs when $px_m = (1-p)x_M$, i.e., the load of short jobs and of long jobs is equal. We note that for any $\gamma \in [0, 1]$, there exists a value $p \in [0.5, 1]$ such that $p\gamma = (1-p)$ holds.

When $l = 1$, in SYS-(2,1, λ), the short and long jobs are executed in different servers. From (3), it follows that the expected waiting time of SYS-(2,1, λ) when $l = 1$ is given by $\frac{\lambda}{2(1-\rho)}(p^2x_m^2 + (1-p)^2x_M^2)$. Using that $px_m = (1-p)x_M$ and also that $\mathbb{E}(X^2) = px_m^2 + (1-p)x_M^2$, we obtain the following expression for the degradation factor:

$$D_{TP(1)}(2, 2, \gamma) = \frac{1 + \gamma^2}{4\gamma p}.$$

It is easy to see that this expression is decreasing with γ for all $\gamma \in [0, 1]$ and p and, as a result, an upper bound and a lower bound are given when $\gamma \rightarrow 0$ and $\gamma \rightarrow 1$, respectively. From Lemma 1 and since the degradation factor tends to infinity when $\gamma \rightarrow 0$, it implies the following result:

Proposition 6. *$D_{TP(1)}(2, 2, \gamma) \geq 1$ and it tends to infinity when $\gamma \rightarrow 0$.*

2) *Unequally Loaded Jobs ($l \neq 1$):* We assume that $l > 1$. For this case, in SYS-(2,1, λ), we have that $px_m > (1-p)x_M$, i.e., the load of small jobs is higher than the load of large jobs, and also that there exists a proportion p_1 such that

$$(1 - p_1)px_m = p_1px_m + (1 - p)x_M,$$

holds. This means that there is one server that executes all the large jobs and a proportion p_1 of small jobs, while in the other server only small jobs are executed. From (3) and using conditional probability properties, we have that the expected waiting time of SYS-(2,1, λ) is

$$\frac{\lambda}{2(1-\rho)}((1-p_1)^2p^2x_m^2 + (p_1p + (1-p))(p_1px_m^2 + (1-p)x_M^2)),$$

which results

$$D_{TP(l)}(2, 2, \gamma, p_1) = \frac{1}{2} \frac{p\gamma^2 + (1-p)}{p^2(1-p_1)^2\gamma^2 + (p_1p + (1-p))(p_1p\gamma^2 + (1-p))}. \quad (14)$$

We show that (14) decreases with γ .

Lemma 11. *When $l > 1$, $D_{TP(l)}(2, 2, \gamma, p_1)$ is a decreasing function of γ .*

From this result and Lemma 1, we conclude that $D_{TP(l)}(2, 2, \gamma, p_1)$ is lower bounded by one when $l > 1$. We now observe that when $p_1 \rightarrow 0$, (14) coincides with $D_{TP(1)}(2, 2, \gamma)$. Besides, executing long jobs and short jobs in different servers leads to a performance improvement in SYS-(2,1, λ) with respect to the case $l > 1$. As a consequence, since SYS-(1,1, $\lambda/2$) does not vary with l , we have that when $l > 1$,

$$D_{TP(l)}(2, 2, \gamma, p_1) \leq \lim_{p_1 \rightarrow 0} D_{TP(l)}(2, 2, \gamma, p_1) = D_{TP(1)}(2, 2, \gamma).$$

Thus, from Proposition 6, it follows that $D_{TP(1)}(2, 2, \gamma)$ is unbounded from above.

Proposition 7. *When $l > 1$, $D_{TP(l)}(2, 2, \gamma, p_1) \geq 1$ and it tends to infinity when $\gamma \rightarrow 0$ and $p_1 \rightarrow 0$.*

When $l < 1$, the situation is very similar to that of $l > 1$. In this case, we have that

$$D_{TP(l)}(2, 2, \gamma, p_2) \leq \lim_{p_2 \rightarrow 0} D_{TP(l)}(2, 2, \gamma, p_2) = D_{TP(1)}(2, 2, \gamma),$$

and the same techniques as in Lemma 11 show that the degradation factor is decreasing with γ when $l < 1$. As a consequence, we give the following result.

Proposition 8. *When $l < 1$, $D_{TP(l)}(2, 2, \gamma, p_2) \geq 1$ and it tends to infinity when $\gamma \rightarrow 0$ and $p_2 \rightarrow 0$.*

B. The case $K > 2$

We show that there are instances where there is no performance degradation for arbitrary K . We assume that l is integer. Hence, we know that in SYS-(K,1, λ) the short jobs are executed in l servers using Bernoulli policy, while the long

jobs are executed in $K - l$ servers, also applying Bernoulli policy. Therefore, the arrival rate to a server that executes short jobs is $\lambda \frac{p}{l}$ and the arrival rate to a server that executes long jobs is $\lambda \frac{1-p}{K-l}$.

We now analyze the performance of $\text{SYS}(K/n, 1, \lambda/n)$ when l is a multiple of n . Thus, for $\text{SYS}(K/n, 1, \lambda/n)$, we define $l^* = \frac{K/n}{1 + \frac{(1-p)}{p \cdot m}}$ and, if l^* is integer, the short jobs are executed in l^* servers and the long jobs in $K/n - l^*$. Note that $l^* = l/n$ and therefore l^* is integer since l is multiple of n . Hence, the arrival rate to a server that executes short jobs is $\frac{\lambda}{n} \frac{p}{l^*} = \lambda \frac{p}{l}$ and the arrival rate to a server that executes long jobs is $\frac{\lambda}{n} \frac{1-p}{K/n - l^*} = \lambda \frac{1-p}{K-l}$. It follows directly thus that the performance of $\text{SYS}(K/n, 1, \lambda/n)$ coincides with that $\text{SYS}(K, 1, \lambda)$ when l is a multiple of n .

When l is not multiple of n , in $\text{SYS}(K/n, 1, \lambda/n)$ there is one server where jobs of both types are executed. Therefore, the performance of both systems do not coincide for this instance and we can claim that there exist a performance degradation. Given the difficulty of the expressions of the degradation factor for arbitrary K and when l is not a multiple of n , we did not succeed in performing the analytical study of the performance degradation.

VIII. NUMERICAL COMPUTATIONS

In this section, we present the numerical experiments of this work. First, we provide numerical evidence to validate the assumptions required for some of the results. Then, we focus on the degradation factor for Bounded Pareto distributed job sizes, for two point distributed job sizes with more than two servers and l not multiple of n and for Degenerate Hyper-exponential distributed job sizes. Finally, we analyze the degradation factor in a system with two servers under SITA policy with optimal cutoffs (SITA-Opt), in M/M/k and M/G/k systems and in Join the Shortest Queue (JSQ) policy.

A. Monotonicity Assumptions

We aim to check that the conjecture of this work holds. We have performed a large number of simulations modifying the parameters of the system. In all the cases, we have observed that the monotonicity property is satisfied. We now present results that are illustrative of the general pattern.

We investigate the degradation factor for Bounded Pareto distributed job sizes with $\alpha \neq 1$. In Figure 1, we assume that $\alpha = 1.5$ and we plot the evolution of $D_{BP(\alpha)}(K, K, \gamma)$ with respect to γ for different values of K . We observe that the degradation factor in all the instances is always decreasing with γ , as stated in Conjecture 1. In addition, we observe that $D_{BP(\alpha)}(K, K, \gamma)$ tends to K^2 when $\gamma \rightarrow 0$, which coincides with the value given in Lemma 10.

We now aim to study if the degradation factor decreases with γ for other dispatching policies than SITA-E. We consider the following dispatching policies: JSQ, M/G/k systems and SITA-Opt. For M/G/10 and M/G/100 systems and different values of n and for SITA-Opt policy with two servers, we have studied numerically the degradation factor when γ varies and the job size distribution is Bounded Pareto and we have

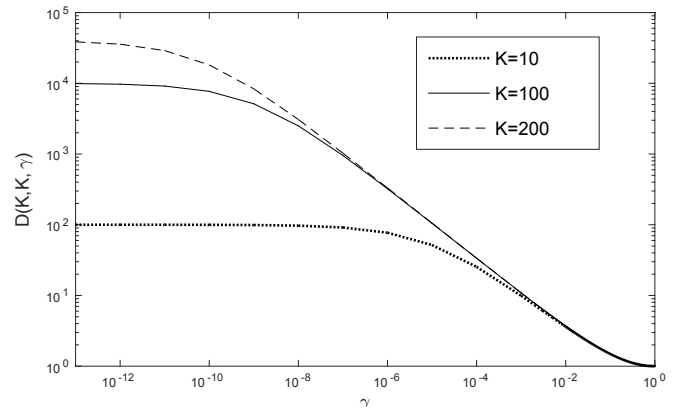


Fig. 1: Evolution over γ of the degradation factor for Bounded Pareto distributed job sizes with parameter $\alpha = 1.5$ (x-axis and y-axis in logarithmic scale).

observed the degradation factor decreases with γ for these routing policies. Due to space constraints, we do not report these results. The simulations results for JSQ are reported in Table II. We consider Bounded Pareto distributed job sizes with $\alpha = 1.25$. We set the smallest job size to one, i.e. $x_m = 1$, and, for different values of K and n , we vary the value of the longest job size from 2 to 1000. The arrival rate is set to ensure that the load of the system is 0.8. For each value of K , n and x_M we have performed five independent runs each with 10^6 arriving jobs and we write as $D(K, n)$ the sample mean of the degradation factor and the numbers in brackets denote the corresponding 95% confidence interval. As it can be seen in Table II, the degradation factor increases with x_M (i.e., decreases with γ) (i) when $K = 10$ and any value of n and (ii) when $K = 100$ and n is 20, 50 and 100. Hence, when $K = 100$ and n is small, the degradation factor decreases with x_M and, therefore, the property of the conjecture does not hold for the JSQ in these cases.

B. Degradation Factor

Here we study the degradation factor for the original SITA-E dispatching policy.

1) *Bounded Pareto*: We now study the degradation factor for Bounded Pareto distributed job sizes. We know from the results of Section VI that the degradation factor can be very large, for example when γ is zero and α is close to one. We consider a system with $K = 1000$ servers and we set γ to $9/10^{14}$, which is the value used by [15]. As we saw in Lemma 7, the performance degradation does not depend on the arrival rate of the system. Hence, we do not specify the value of this parameter in these experiments.

In Table III, we show the degradation factor when $n = 100$, $n = 500$ and $n = 1000$ for different values of α . We also present in Table III the evolution over α of the value $n^{\frac{1}{1-\alpha}}$, which is the degradation factor when γ is zero. We observe that the degradation factor is always far from the value of the upper bound achieved when γ is zero. However, there are some values of α where the degradation factor is high. An example is $\alpha = 1.25$, which is a typical value found in

| K | n | x_M | $D(K, n)$ | 95% CI |
|-----|-----|-------|-----------|-------------------|
| 10 | 2 | 2 | 2.0026 | [1.9847, 2.0205] |
| | | 10 | 2.0554 | [2.0249, 2.0858] |
| | | 100 | 2.2786 | [2.2501, 2.3070] |
| | | 1000 | 2.7236 | [2.4292, 3.0181] |
| | 5 | 2 | 5.0528 | [5.0208, 5.0847] |
| | | 10 | 5.4128 | [5.3488, 5.4769] |
| | | 100 | 6.5810 | [6.4216, 6.7403] |
| | | 1000 | 9.6124 | [9.0108, 10.2140] |
| | 10 | 2 | 10.601 | [10.472, 10.730] |
| | | 10 | 11.211 | [10.907, 11.516] |
| | | 100 | 14.851 | [14.411, 15.291] |
| | | 1000 | 24.151 | [23.467, 24.835] |
| 100 | 2 | 2 | 5.9275 | [5.5965, 6.2585] |
| | | 10 | 5.9187 | [5.4806, 6.3569] |
| | | 100 | 5.3444 | [4.4209, 6.2679] |
| | | 1000 | 4.4991 | [3.7681, 5.2302] |
| | 5 | 2 | 29.175 | [27.404, 30.947] |
| | | 10 | 28.194 | [26.686, 29.701] |
| | | 100 | 24.103 | [21.012, 27.194] |
| | | 1000 | 18.992 | [15.619, 22.364] |
| | 10 | 2 | 67.422 | [63.454, 71.391] |
| | | 10 | 64.850 | [61.334, 68.365] |
| | | 100 | 57.803 | [51.491, 64.115] |
| | | 1000 | 56.800 | [49.168, 64.431] |
| | 20 | 2 | 134.53 | [126.47, 142.59] |
| | | 10 | 134.52 | [126.92, 142.11] |
| | | 100 | 134.91 | [121.78, 148.04] |
| | | 1000 | 155.36 | [138.27, 172.45] |
| | 50 | 2 | 343.58 | [322.95, 364.22] |
| | | 10 | 351.04 | [326.80, 375.28] |
| | | 100 | 394.06 | [347.62, 440.49] |
| | | 1000 | 561.26 | [499.83, 622.70] |
| | 100 | 2 | 720.00 | [677.49, 762.50] |
| | | 10 | 731.46 | [679.92, 783.00] |
| | | 100 | 858.45 | [743.30, 973.60] |
| | | 1000 | 1339.2 | [1171.5, 1506.9] |

TABLE II: Sample mean and 95% confidence interval of the degradation factor of JSQ for different values of K , n and x_M .

computer and networking systems [17]. As it can be seen, for this instance, the degradation factor is equal to 4149 for $n = 100$, to $2.537 \cdot 10^7$ for $n = 500$ and to $4.0481 \cdot 10^8$ for $n = 1000$. We also observe that the upper bound gets tighter when $n = 1000$. Besides, when $\alpha = 1$, the upper bound is infinity and the degradation factor is $1.2311 \cdot 10^{10}$ for $n = 1000$.

2) Two Points when $K > 2$ and l is not multiple of n :

In Section VII-B, we show that the degradation factor equals one for $K > 2$ when l is multiple of n . We now assume that l is not multiple of n and we study the degradation factor for this case.

We assume that $K = 1000$ and $n = K$. In Figure 2, we analyze the evolution of the degradation factor when we vary the value of γ from one to 10^{-3} and we consider four different values of l : 10 (dashed line), 300 (solid line), 500 (dotted and dashed line) and 750 (dotted line). We note that, for all the cases, the value of l is not a multiple of n . As it can be observed, the value of the degradation factor grows unboundedly for all the instances under consideration. Therefore, we conclude that the results of Proposition 6, Proposition 7 and Proposition 8 extend to systems with more than two servers.

3) *Degenerate Hyper-exponential*: We consider a system with SITA-E policy and Degenerate Hyper-exponential dis-

tributed job sizes. This distribution with probability p is an exponential of rate μp and with probability $1 - p$ it is an exponential with rate infinity. Interestingly, the mean of the Degenerate Hyperexponential distribution is $1/\mu$, which does not depend on p and the second moment is $\frac{1}{p\mu^2}$. The coefficient of variation is $C = 2/p - 1$ and it belongs to $[1, \infty)$ as p varies. Therefore, we study the degradation factor (7) for this distribution when p varies.

In Figure 3, we consider $\lambda = \mu = 1$ and we depict the evolution of the degradation factor when p varies from 0.01 to 0.99 in a system with: (i) two servers and two groups; (ii) four servers and four groups, (iii) eight servers and eight groups; (iv) eight servers and four groups and (v) eight servers and two groups. We observe that the degradation factor decreases with p in all the cases. In fact, when p decreases, the variability of jobs increases and this implies that the difference in the performance of both systems increases. We also see that, as expected, the degradation factor is always higher than one, which means that the performance of both systems never coincides. Furthermore, when $p = 0.01$, the coefficient of variation is 199 and the degradation obtained in a system with eight servers and eight groups for this case is 41.4. We have done more experiments changing the value of the system parameters, for example μ , and the obtained results confirm that the performance degradation is also significant and that the degradation increases as the variability of jobs increases.

C. SITA-Opt Degradation Factor

We now study the degradation factor of SITA-Opt, which is a SITA policy where the thresholds are chosen to optimize the performance of the system. We consider a system with two servers and we compare SYS-(2,2, λ) and SYS-(2,1, λ) for Bounded Pareto distributed job size. In this case, the ratio of performances is said to be the degradation factor of SITA-Opt. According to the result of [17], in a two server system, the degradation factor we study in this paper coincides with the degradation factor of SITA-Opt when $\alpha = 1$. Besides, the analytical computation of the optimal thresholds seems to be intractable even in a system with two servers. Therefore, we explore here numerically the degradation when $\alpha \neq 1$.

Our objective is to assess the performance degradation of SITA-Opt and to compare it with the degradation factor obtained with (11). The authors in [17] consider a two-server system and they obtain numerically the ratio of the performance of SITA-E policy over the performance of SITA-Opt policy. We obtain the degradation factor of SITA-Opt for the same parameters as theirs by multiplying the performance ratio they obtain with the degradation factor obtained in (11). Hence, in Table IV, we represent the degradation factor of SITA-Opt for low load ($\rho = 0.005$), medium load ($\rho = 0.5$) and high load ($\rho = 0.8$) and for different values of α . We also show in this table the degradation obtained using (11) for different values of α . As it can be observed, the degradation factor has a symmetry since we have obtained the same results for α and $2-\alpha$. For SITA-Opt, we know that that the symmetry sends the cutoffs of the SITA-Opt to the cutoffs of the SITA-Opt of the dual distribution, see [3] for full details on this

| | $n = 100$ | | $n = 500$ | | $n = 1000$ | |
|-----------------|--------------------------------|--------------------------|--------------------------------|--------------------------|--------------------------------|--------------------------|
| | $D_{BP(\alpha)}(K, n, \gamma)$ | $n^{\frac{1}{1-\alpha}}$ | $D_{BP(\alpha)}(K, n, \gamma)$ | $n^{\frac{1}{1-\alpha}}$ | $D_{BP(\alpha)}(K, n, \gamma)$ | $n^{\frac{1}{1-\alpha}}$ |
| $\alpha = 0.25$ | 89.0317 | 464.15 | 755.68 | 3968.5 | $1.9033 \cdot 10^3$ | $9.999 \cdot 10^3$ |
| $\alpha = 0.5$ | 5263.6 | 10^4 | $1.3158 \cdot 10^5$ | $25 \cdot 10^4$ | $5.2631 \cdot 10^5$ | 10^6 |
| $\alpha = 0.75$ | 4149 | 10^8 | $2.537 \cdot 10^7$ | $6.25 \cdot 10^{10}$ | $4.0481 \cdot 10^8$ | 10^{12} |
| $\alpha = 1$ | 2.0183 | ∞ | $1.4775 \cdot 10^4$ | ∞ | $1.2311 \cdot 10^{10}$ | ∞ |
| $\alpha = 1.25$ | 4149 | 10^8 | $2.537 \cdot 10^7$ | $6.25 \cdot 10^{10}$ | $4.0481 \cdot 10^8$ | 10^{12} |
| $\alpha = 1.5$ | 5263.6 | 10^4 | $1.3158 \cdot 10^5$ | $25 \cdot 10^4$ | $5.2631 \cdot 10^5$ | 10^6 |
| $\alpha = 1.75$ | 89.0317 | 464.15 | 755.68 | 3968.5 | $1.9033 \cdot 10^3$ | $9.999 \cdot 10^3$ |

TABLE III: Degradation factor for Bounded Pareto distributed job sizes when $K = 1000$ and $\gamma = \frac{9}{10^{14}}$ compared with $n^{\frac{1}{1-\alpha}}$.

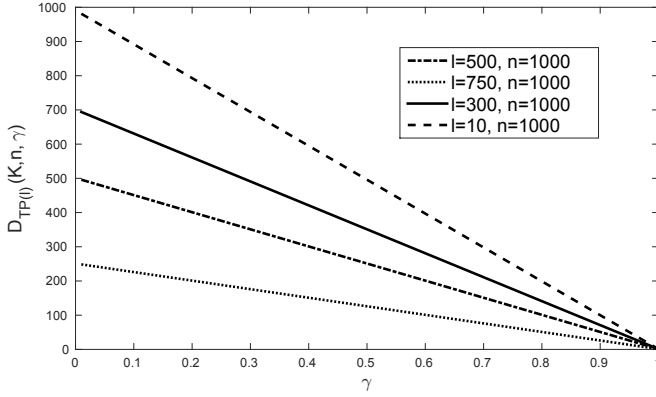


Fig. 2: Degradation factor for two points distributed job sizes with $K > 2$ and l not multiple of n when p varies.

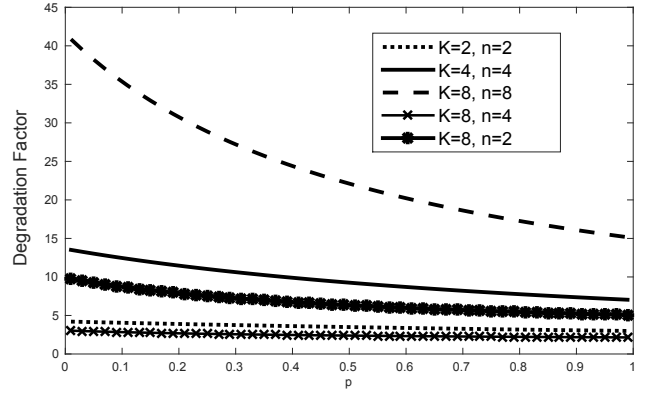


Fig. 3: Degradation factor for Degenerated Hyperexponential distributed job sizes when p varies.

| | Degradation Factor | | | |
|-----------------|--------------------|-------------------|--------------|--------|
| | SITA-Opt | | | SITA-E |
| | $\rho = 0.005$ | $\rho = 0.5$ | $\rho = 0.8$ | |
| $\alpha = 0.25$ | 333.74 | 87.77 | 8.6594 | 2.52 |
| $\alpha = 0.5$ | $2.24 \cdot 10^4$ | 4219.9 | 18.7679 | 4 |
| $\alpha = 0.75$ | $3.36 \cdot 10^5$ | $1.31 \cdot 10^5$ | 133.88 | 15.96 |
| $\alpha = 1.25$ | $3.36 \cdot 10^5$ | $1.31 \cdot 10^5$ | 133.88 | 15.96 |
| $\alpha = 1.5$ | $2.24 \cdot 10^4$ | 4219.9 | 18.7679 | 4 |
| $\alpha = 1.75$ | 333.74 | 87.77 | 8.6594 | 2.52 |

TABLE IV: Degradation factor in a system with two servers.

symmetry property. Furthermore, we observe in Table IV that the degradation factor of SITA-Opt is very high in some instances, whereas for SITA-E it is not that high. For example, when $\alpha = 1.25$ and when $\alpha = 1.5$, the degradation factor of SITA-Opt for $\rho = 0.005$ is, respectively, $3.3604 \cdot 10^5$ and $2.2476 \cdot 10^4$ and for the degradation factor considering SITA-E is 15.96. The reason for this big difference is that, for the SITA-Opt threshold, the load is unbalanced very heavily toward one of the servers.

D. M/M/K and M/G/K Systems Degradation Factor

We now analyze the performance degradation in an M/M/K system. In this system, there is a central queue that dispatches the incoming jobs to the servers that are idle. We consider an M/M/10 system and an M/M/100 system with $\mu = 1$ and $\rho = 0.8$ and we present in Table V the degradation of these systems for different values of n . We observe that the degradation factor is higher than one for all the considered cases and that the degradation increases with n . Besides, for $K = 10$

| | Degradation Factor | |
|----------|--------------------|------------------|
| | $K = 10$ | $K = 100$ |
| | $n = 2$ | 2.7084 |
| $n = 5$ | 8.6895 | 65.172 |
| $n = 10$ | 19.5513 | 208.27 |
| | | $n = 20$ 564.08 |
| | | $n = 50$ 1809.8 |
| | | $n = 100$ 4072.0 |

TABLE V: Degradation factor for M/M/K systems over n .

servers, the degradation is 19.5513 when $n = 10$, which is not very large. However, for 100 servers, the degradation factor is very large, specially when $n = 50$ and $n = 100$. In fact, when K is large, the mean waiting time of an M/M/K system is very small since, for all the incoming jobs, the probability of having an idle queue is very close to one; this explains the large values obtained for the degradation factor when $K = 100$.

We now focus on M/G/K systems with Bounded Pareto distributed job sizes and $\alpha = 1.25$. We remark that the degradation of M/G/k systems coincides with that of Join the Shortest Workload policy. We assume that $x_m = 1$, $x_M = 10$ and $\rho = 0.8$. For $K = 10$ and $K = 100$ and different values of n , we have performed for five independent runs each with 10^6 arriving jobs and, in Table VI, we write as $D(K, n)$ the sample mean of the degradation factor and the numbers in brackets denote the corresponding 95% confidence interval. As it can be observed in Table VI, the degradation factor for $K = 10$ is, respectively, 2.7239 and 8.835 when $n = 2$ and $n = 5$, whereas for $K = 100$ the degradation is high (for instance, when $n = 100$ it is 3856.3).

| Degradation Factor | | | | | |
|--------------------|----------|-----------------|-----------|----------|-----------------|
| $K = 10$ | | | $K = 100$ | | |
| n | $D(K,n)$ | 95% CI | n | $D(K,n)$ | 95% CI |
| 2 | 2.7239 | [2.6769,2.7708] | 2 | 8.8020 | [7.1825,10.421] |
| 5 | 8.8350 | [8.6935,8.9765] | 5 | 63.457 | [56.626,70.288] |
| 10 | 19.607 | [19.039,20.175] | 10 | 197.36 | [174.38,220.34] |
| | | | 20 | 539.51 | [472.31,606.71] |
| | | | 50 | 1713.2 | [1514.0,1912.4] |
| | | | 100 | 3856.3 | [3389.0,4323.7] |

TABLE VI: Degradation factor for M/G/K systems over n .

| Degradation Factor | | | | | |
|--------------------|----------|---------------|-----------|----------|---------------|
| $K = 10$ | | | $K = 100$ | | |
| n | $D(K,n)$ | 95% CI | n | $D(K,n)$ | 95% CI |
| 2 | 2.13 | [2.08,2.17] | 2 | 6.02 | [5.22,6.82] |
| 5 | 5.52 | [5.33,5.72] | 5 | 27.64 | [25.43,29.85] |
| 10 | 11.11 | [10.87,11.35] | 10 | 65.08 | [59.58,70.57] |
| | | | 20 | 138.8 | [126.8,150.9] |
| | | | 50 | 357.8 | [327.3,388.2] |
| | | | 100 | 736.7 | [678.9,794.5] |

TABLE VII: Degradation factor for JSQ policy over n .

E. JSQ Degradation Factor

We now analyze the degradation factor for the JSQ dispatching policy. We have performed simulations of parallel server system with 10 and 100 servers and different values of n in each case. We assume that the service time distribution is exponential with load $\rho = 0.8$. For each value of K and n we have performed five independent runs each with 10^6 arriving jobs. The results presented in Table VII are based on 5 independent runs each with 10^6 arriving jobs. We observe that the degradation factor is 2.13 for $K = 10$ and $n = 2$, but for $K = 100$ the value of the degradation factor is much higher, specially when n is high. For example, when $n = 100$, the degradation factor for Join the Shortest Queue dispatching policy is 736.7. We also observe that the degradation factor for JSQ and exponentially distributed job sizes is smaller than the degradation for M/M/k systems.

IX. DEGRADATION WITH REAL TRACES

In this section, we study the degradation when the job sizes follow the distribution of jobs that have been submitted to real data centers. The distribution of the jobs sizes are discrete and does not satisfy the conditions of the distributions we have considered in the previous sections. The distributions under consideration here are characterized by a vector of job sizes $\mathbf{x} = (x_m, \dots, x_M)$ and a probability distribution of these jobs sizes \mathbf{p} .

One might think that the degradation factor always increases with n . In the Supplementary Material, we present an example that shows that, for discrete job sizes distributions, this is not always true.

We aim to analyze the degradation factor when the job sizes follow the distribution of the jobs submitted to real parallel machines. Hence, we obtain the values of \mathbf{x} and \mathbf{p} of several data centers, that is, the vector of the job sizes and a probability distribution of the job sizes, from the repository [1]. The details about the archive and the handling of the data provided in this repository are available in [9].

The parallel machines we consider here are the High Performance Computing Center North (HPC2N), the San Diego

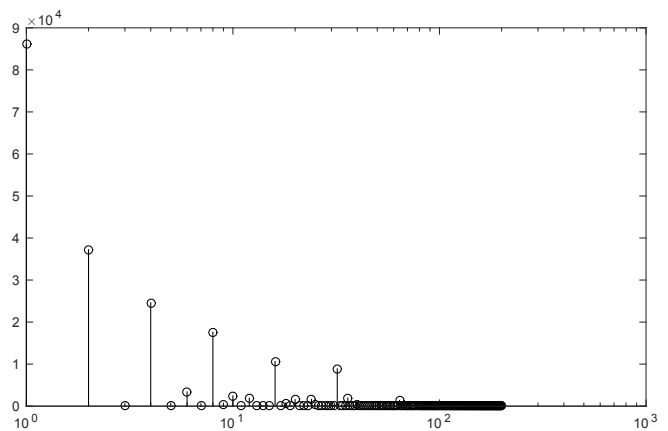


Fig. 4: HPC2N system. In the x-axis we represent the size of the submitted jobs (in bytes) and in the y-axis the number of jobs of each size.

Supercomputing Center (SDSC) Datastar and the RIKEN Integrated Cluster of Clusters (RICC). In each case, we consider a system with K servers and K/n servers and, using the values of \mathbf{x} and \mathbf{p} of each parallel machines system, we first compute numerically the proportions of jobs to be executed in each server to ensure that the load of all the servers is equalized and also that the thresholds $c_0, c_1, \dots, c_R, c_{R+1}$ must constitute a nondecreasing sequence, that is, if jobs of size x_{i-1} and of size x_{i+1} are executed in the same server, all the jobs of size x_i are executed in that server. Then, using these proportions, we compute the performance of each of the systems to get insights of the degradation factor when the job sizes follow the distribution of the jobs submitted to real parallel machines.

A. HPC2N Data Center

The High Performance Computing Center North is a parallel machine system that is in Sweden. The traces we analyze here consist of more than 200,000 jobs submitted between July 2002 until January 2006. In Figure 4, we represent the number of jobs submitted of each size. As it can be observed, job sizes vary from 1 byte to 200 bytes. Besides, the majority of jobs are of small size, for example, the number of jobs of size 1 byte is almost 90,000 and of size 2 bytes is 40,000.

We analyze the degradation factor when the job sizes follow the distribution of Figure 4. The number of servers is 10,000 and we present in Table VIII the values of the degradation factor for several values of n between 1 and 10,000. We observe that the degradation factor when $n = 200$ is higher than that of $n = 250$, which confirms that the degradation factor for discrete job sizes distributions does not always increase with n . In the rest of the values of n , we see that there is a partial monotonicity on n for the degradation factor. Besides, as it can be observed in Table VIII, the degradation factor is lower bounded by 1 and upper bounded by the $D(K, K, x_m, x_M)$, which coincides with the result of Corollary 1.

| HPC2N | | SDSC | | RICC | |
|-------------|--------------------|-------------|--------------------|-------------|--------------------|
| $K = 10000$ | Degradation Factor | $K = 10000$ | Degradation Factor | $K = 10000$ | Degradation Factor |
| $n = 1$ | 1 | $n = 1$ | 1.0001 | $n = 10$ | 1.0001 |
| $n = 10$ | 1.0001 | $n = 10$ | 1.0001 | $n = 10$ | 1.0007 |
| $n = 50$ | 1.0012 | $n = 50$ | 1.0010 | $n = 10$ | 1.0031 |
| $n = 100$ | 1.0034 | $n = 100$ | 1.0019 | $n = 100$ | 1.0048 |
| $n = 200$ | 1.0067 | $n = 200$ | 1.0063 | $n = 200$ | 1.0137 |
| $n = 250$ | 1.0060 | $n = 250$ | 1.0051 | $n = 250$ | 1.0184 |
| $n = 500$ | 1.0219 | $n = 500$ | 1.0163 | $n = 500$ | 1.0487 |
| $n = 1000$ | 1.0476 | $n = 1000$ | 1.0658 | $n = 1000$ | 1.1238 |
| $n = 10000$ | 4.0365 | $n = 10000$ | 5.7973 | $n = 10000$ | 7.5531 |

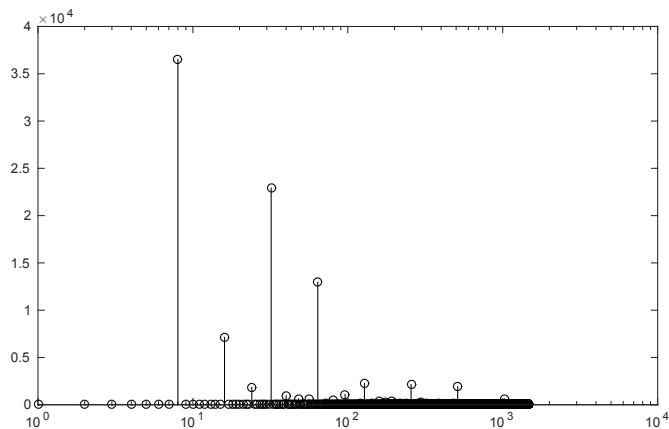
TABLE VIII: Degradation factor of HPC2N, SDSC and RICC systems for $K = 10000$ when n varies from 1 to 10000.

Fig. 5: SDSC Datastar. In the x-axis we represent the size of the submitted jobs (in bytes) and in the y-axis the number of jobs of each size.

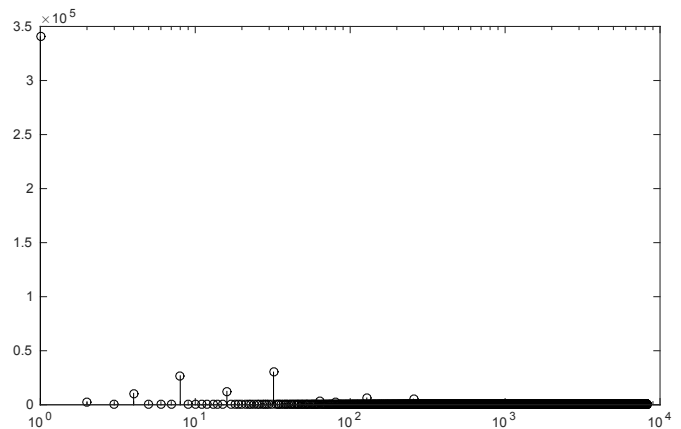


Fig. 6: RICC system. In the x-axis we represent the size of the submitted jobs (in bytes) and in the y-axis the number of jobs of each size.

B. SDSC Datastar

The San Diego Supercomputing Center is a high performance computing system devoted to scientific research. We investigate the traces of this system that consist of almost 100,000 jobs submitted from March 2004 to March 2005. In Figure 5, we illustrate the number of jobs of each size that has been submitted to this system. We see that the difference between the smallest job size and the largest is higher than in the HPC2N system since the smallest job is of size 8 bytes and the largest of 1480 bytes. We also observe that the majority of jobs are of size 8 bytes, 32 bytes and 64 bytes.

Our goal is to analyze the degradation factor when the job sizes follow the distribution represented in Figure 5. We assume that the number of servers is $K = 10,000$ and the number of groups n varies from 1 to 10000. We present in Table VIII the values of the degradation factor for the considered cases. We observe that the degradation factor is the degradation factor is partially monotone on n and when $n = 200$ the degradation is higher than that of $n = 250$. Furthermore, the result of Corollary 1 also holds for this case, since the degradation factor is lower bounded by one and upper bounded by $D(K, K, x_m, x_M)$.

C. RICC System

The RIKEN Integrated Cluster of Clusters is a system of RIKEN, which is a research institution of the Japanese

government. The traces we analyze in this part of the work consist of almost 500,000 jobs that have been submitted to RICC system from May 2010 to September 2010. The size of the smallest job is 1 byte and the largest 8192 bytes, therefore the difference on the size between the smallest and the largest is higher than in SDSC system. We plot in Figure 6 the number of jobs submitted of each size. We observe that almost 350,000 jobs are of size 1 byte.

We are interested in studying the degradation factor when the sizes of job follow the distribution shown in Figure 6. We assume that the number of servers is $K = 10000$ and also the values of n varying from 1 to 10000. In Table VIII we present the values of the degradation factor for the instances under consideration. We observe that the degradation factor is lower bounded by 1 and upper bounded by 7.5531, which is the degradation obtained when $n = 10000$. Besides, we observe that for this case, the degradation factor increases with n .

X. CONCLUSIONS

In this work, we have analyzed the performance of multi-servers systems where dispatchers operate under SITA-E policy. We compare the performance of $\text{SYS}(K, n, \lambda)$, which is a system formed by K servers, $n > 1$ dispatcher and arrival rate λ with the performance of $\text{SYS}(K, 1, \lambda)$. We first show that the performance of $\text{SYS}(K, n, \lambda)$ is the same as the performance of $\text{SYS}(K/n, 1, \lambda/n)$. As a consequence, the performance analysis we perform in this article can be seen as the economies of scaling up the number of servers and the arrival rate

proportionally in a system with a single dispatcher. We define the degradation factor as the ratio between the performance of $\text{SYS}-(K, 1, \lambda)$ and the performance of $\text{SYS}-(K/n, 1, \lambda/n)$. For an arbitrary distribution, we show that the degradation factor is lower-bounded by one and upper-bounded by the ratio between the performance of $\text{SYS}-(K, 1, \lambda)$ and of $\text{SYS}-(1, 1, \lambda/K)$. For uniformly distributed job sizes, we show that the degradation factor is upper bounded by $4/3$, whereas for Bounded Pareto distributed job sizes we show that it is unbounded from above for the following cases: (i) $\alpha = 0$ and (ii) $\alpha \neq 1$ and $K \rightarrow \infty$. For $\alpha \in (0, 1) \cup (1, 2)$ and finite K , assuming that the degradation factor decreases with γ , we show that the degradation factor is upper-bounded by $K^{\frac{1}{1-\alpha}}$. We also consider the two point job size distribution and show that, for two servers, the degradation factor is unbounded from above.

For future work, an interesting extension of the degradation factor analysis performed in this work would be to consider other popular load balancing policies such as Power of two, Join the Shortest Queue or the SITA policy with optimal thresholds. Another possible future research is considering other metrics for the performance of these systems, such as tail-probabilities or second moment of the waiting time. Finally, we would like to analyze the performance degradation of multiservers systems considering a model with less restrictions, for instance, when the servers are heterogeneous and not necessarily FCFS.

REFERENCES

- [1] <http://www.cs.huji.ac.il/labs/parallel/workload/index.html>.
- [2] E. Altman, U. Ayesta, and B. J. Prabhu. Load balancing in processor sharing systems. *Telecommunication Systems*, 47(1), 2011.
- [3] E. Bachmat. *Mathematical adventures in performance analysis*. Springer-Birkhauser, 2014.
- [4] E. Bachmat and H. Sarfati. Analysis of SITA policies. *Performance Evaluation*, 67(2):102–120, 2010.
- [5] C. H. Bell and S. Stidham. Individual versus social optimization in the allocation of customers to alternative servers. *Management Science*, 29:831–839, 1983.
- [6] G. Ciardo, A. Riska, and E. Smirni. Equiloat: a load balancing policy for clustered web servers. *Performance Evaluation*, 46(2), 2001.
- [7] J. Doncel, S. Aalto, and U. Ayesta. Economies of scale in parallel-server systems. In *IEEE Infocom*, 2017.
- [8] J. Doncel, U. Ayesta, O. Brun, and B. Prabhu. Is the price of anarchy the right measure for load-balancing games? *ACM Transactions on Internet Technology (TOIT)*, 14(2-3):18, 2014.
- [9] D. G. Feitelson, D. Tsafir, and D. Krakov. Experience with using the parallel workloads archive. *Journal of Parallel and Distributed Computing*, 74(10):2967 – 2982, 2014.
- [10] H. Feng, V. Misra, and D. Rubenstein. Optimal state-free, size-aware dispatching for heterogeneous M/G/1-type systems. *Performance Evaluation*, 62(1-4):475–492, Oct. 2005.
- [11] R. D. Foley and D. R. McDonald. Join the shortest queue: Stability and exact asymptotics. *Annals of Applied Probab.*, 11(3), 2001.
- [12] V. Gupta, M. Harchol-Balter, K. Sigman, and W. Whitt. Analysis of join-the-shortest-queue routing for web server farms. *Performance Evaluation*, 64(9):1062–1081, 2007.
- [13] M. Harchol-Balter. Task assignment with unknown duration. In *International Conference on Distributed Computing Systems*, 2000.
- [14] M. Harchol-Balter. *Performance Modeling and Design of Computer Systems: Queueing Theory in Action*. Cambridge Univ. 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, A. Scheller-Wolf, and A. R. Young. Surprising results on task assignment in server farms with high-variability workloads. In *Proceedings of SIGMETRICS*, 2009.
- [17] M. Harchol-Balter and R. Vesilo. To balance or unbalance load in size-interval task allocation. *Probability in the Engineering and Information Sciences*, 24(2):219–244, Apr. 2010.
- [18] M. Haviv and T. Roughgarden. The price of anarchy in an exponential multi-server. *Operations Research Letters*, 35:421–426, 2007.
- [19] Y. Lu, Q. Xie, G. Kliot, A. Geller, J. R. Larus, and A. Greenberg. Join-idle-queue: A novel load balancing algorithm for dynamically scalable web services. *Performance Evaluation*, 68(11):1056–1071, 2011.
- [20] M. Mitzenmacher. The power of two choices in randomized load balancing. *IEEE Trans. on Parallel and Distr. Systems*, 12(10), 2001.
- [21] A. Orda, R. Rom, and N. Shimkin. Competitive routing in multiuser communication networks. 1(5):510–521, 1993.
- [22] A. W. Richa, M. Mitzenmacher, and R. Sitaraman. The power of two random choices: A survey of techniques and results. *Handbook of Randomized Computing*, 1, 2001.
- [23] T. Roughgarden and É. Tardos. How bad is selfish routing? *Journal of the ACM*, 49(2):236–259, 2002.
- [24] 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.
- [25] F. Semchedine, L. Bouallouche-Medjkoune, and D. Aissani. Task assignment policies in distributed server systems: A survey. *Journal of Network and Computer Applications*, 34(4):1123 – 1130, 2011.
- [26] R. Vesilo. Asymptotic analysis of load distribution for size-interval task allocation with bounded pareto job sizes. In *IEEE International Conference on Parallel and Distributed Systems*, 2008.
- [27] T. Wang, Z. Su, Y. Xia, and M. Hamdi. Rethinking the data center networking: Architecture, network protocols, and resource sharing. *IEEE Access*, 2:1481–1496, 2014.
- [28] R. R. Weber. On the optimal assignment of customers to parallel servers. *Journal of Applied Probability*, 15(2):406–413, 1978.
- [29] W. Whitt. Understanding the efficiency of multi-server service systems. *Management Science*, 38(5):708–723, 1992.
- [30] W. Whitt. How multiserver queues scale with growing congestion-dependent demand. *Operations Research*, 51(4):531–542, 2003.
- [31] A. Williams, M. Arlitt, C. Williamson, and K. Barker. *Web Workload Characterization: Ten Years Later*, pages 3–21. Springer US, Boston, MA, 2005.
- [32] W. Winston. Optimality of the shortest line discipline. *Journal of Applied Probability*, 14(1):181–189, 1977.

Josú Doncel obtained from the University of the Basque Country (UPV/EHU) the Industrial Engineering degree in 2007, the Mathematics degree in 2010 and, in 2011, the Master degree in Applied Mathematics and Statistics. He received in 2015 the PhD degree in Computer Science from Université de Toulouse (France). He is currently an assistant professor at the UPV/EHU. He has previously held research positions at LAAS-CNRS (France), Inria Grenoble (France) and BCAM-Basque Center for Applied Mathematics (Spain), teaching positions at ENSIMAG (France), INSA-Toulouse (France) and IUT-Blagnac (France) and invited professor positions at David laboratory (France) and Inria Paris (France).

Samuli Aalto received his M.Sc. and Ph.D. degrees in Mathematics from the University of Helsinki in 1984 and 1998, respectively. From 1984 to 1997, Dr. Aalto worked as a Research Scientist at VTT Technical Research Centre of Finland. Since 1997, he has been with TKK Helsinki University of Technology, which is now part of Aalto University. Currently he acts as Senior University Lecturer leading the Performance Analysis Group in the Department of Communications and Networking. Dr. Aalto's research interests include queueing theory, teletraffic theory, and performance analysis of modern communications systems and networks.

Urtzi Ayesta received a PhD degree from Université de Nice-Sophia Antipolis (France), a MS degree in Electrical Engineering from Columbia University (US) and a BS/MS degree in Telecommunication Engineering from Nafarroako Unibertsitate Publikoa-Universidad Publica de Navarra (Spain). His PhD research was carried out at the research laboratories of INRIA Sophia-Antipolis and France Telecom R&D. He is currently a CNRS researcher at IRIT, Toulouse, and he also holds an adjunct lecturer position (part-time appointment funded by Ikerbasque Foundation) in the Computer Science Faculty at the University of the Basque Country.