Economies of Scale in Parallel-Server Systems

J. Doncel[a,b], S. Aalto[c], U. Ayesta[d,e,f,g]

[a] INRIA, 38000 Grenoble, France.
[b] Université Grenoble Alpes, CNRS, LIG, 38000 Grenoble, France
[c] Department of Communications and Networking, Aalto University, Finland
[d] CNRS, IRIT, 2 rue C. Camichel, 31071 Toulouse, France.
[e] Université de Toulouse, INP, 31071 Toulouse, France
[f] IKERBASQUE - Basque Foundation for Science, 48011 Bilbao, Spain
[g] UPV/EHU, Univ. of the Basque Country, 20018 Donostia, Spain

***Abstract*—We consider a parallel-server system with K homogeneous servers where incoming tasks, arriving at rate λ, are dispatched by n dispatchers. Servers are FCFS queues and dispatchers implement 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. Every dispatcher handles a fraction $1/n$ of the incoming traffic and balances the load to K/n servers. 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. Therefore, the performance comparison can be interpreted as the economies of scale in a system with one dispatcher when we scale up the number of servers and the arrival rate proportionately. We consider two continuous service time distributions: uniform and Bounded Pareto that have increasing and decreasing failure rates, respectively; 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.**

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) [1]. 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, when the variability of jobs increases, SITA-E policy improves the performance comparing with other task assignment policies such as Round Robin or Bernoulli. Another important property of SITA-E policy 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.

Research partially supported by the European Commission under FP7 project QUANTICOL (Grant No. 600708), by the French "Agence Nationale de la Recherche (ANR)" through the project ANR-15-CE25-0004 (ANR JCJC RACON) and the Academy of Finland through the project FQ4BD (Grant No. 296206). Part of this work was carried out while U. Ayesta and J. Doncel were affiliated with CNRS-LAAS.

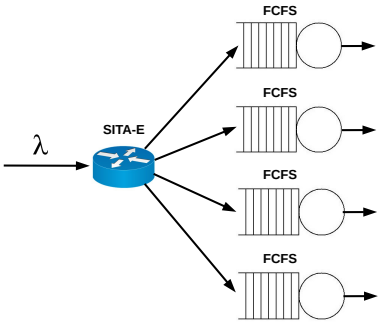

Fig. 1: SYS-(4,1,λ). There is one dispatcher that receives all the traffic and sends it to all the servers.

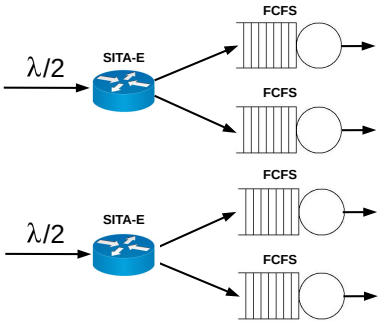

Fig. 2: SYS-(4,2,λ). There are two dispatchers and each of them controls a half of the total incoming traffic and balances the load to two servers.

In this work, we compare the performance of SYS-(K,n,λ), 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,λ). We present in Figure 1 and in Figure 2 an example of the multiserver systems under comparison in this paper. 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,λ) over the mean waiting time of SYS-(K,1,λ).

	Degradation Factor (D)	Result
Uniform Distribution: $K = 2$	$1 \leq \mathrm{D} \leq 1.138$.	Prop 2
Uniform Distribution: $K > 2$ Servers, n Groups.	$1 \leq \mathrm{D} \leq 4/3$.	Prop 3
Bounded Pareto: $\alpha = 1$, K Servers, n Groups.	D≥ 1 and D$\rightarrow \infty$[1]	Prop 4
Bounded Pareto: $\alpha \neq 1$, K Servers, n Groups.	$1 \leq \mathrm{D} \leq n^{\frac{1}{\lvert 1-\alpha \rvert}}$.	Prop 5
Two Point: $K = 2$, Equally Loaded Jobs	D≥ 1 and D$\rightarrow \infty$[1]	Prop 6
Two Point: $K = 2$, Unequally Loaded Jobs	D≥ 1 and D$\rightarrow \infty$[1]	Prop 7 Prop 8

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

We show in Section III that the performance of SYS-(K,n,λ) is equal to the performance of SYS-(K/n,1,λ/n). Thus, the analysis of the degradation factor can be interpreted as the economies of scale a multiserver system when we scale up the number of servers and the arrival rate proportionately.

This work can potentially have an impact in the design of data centers. Indeed, the architecture of modern data centers has a tree-based topology where the knowledge of how to split jobs is given in the edges nodes. This architecture corresponds to SYS-(K,n,λ) [2]. However, if the routing policies are implemented in the core nodes, data centers consist of SYS-(K,1,λ) 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 [3]. 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. We analyze the degradation factor for three representative distributions. We first consider two job size distributions, uniform and Bounded Pareto, whose failure rates are respectively increasing and decreasing.

- **Uniform Distribution.** For uniformly distributed job sizes and two servers, we show that the degradation factor is lower bounded by one and upper bounded by 1.138. For more than two servers, assuming that the degradation factor decreases with γ, we prove that this ratio is lower bounded by 1 and upper bounded by $4/3$.
- **Bounded Pareto Distribution.** For Bounded Pareto distributed job sizes with parameter $\alpha = 1$, we show that the degradation factor is lower bounded by one and unbounded from above. When $\alpha \neq 1$, assuming that the degradation factor decreases with γ, we prove that this ratio is lower bounded by 1 and upper bounded by $n^{\frac{1}{\lvert 1-\alpha \rvert}}$.

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.

According to our results, the degradation is small for uniformly distributed job sizes, but for Bounded Pareto and two point distributions 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.

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 impossible even for a system with two servers [4]. 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. We study the degradation factor for uniformly distributed job sizes in Section IV and, in Section V, for Bounded Pareto distributed job sizes. Then, in Section VI we analyze the degradation for a discrete job sizes distribution that concentrates on two points. Finally, we present the numerical experiments in Section VII.

Due to lack of space we have omitted the proofs of our main results and for full details we refer to [5].

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 jobs, see the survey [6] and the book [7]. 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 [8], [9] the dispatcher sends the job to the queue with least number of 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. Another important routing policy is Power of Two [10], [11], 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 two servers. When the service demand is known and the servers are FCFS, the SITA policy with optimal thresholds

[1]We show that the degradation factor is lower bounded by one and that there exist parameters of the system such that $D \rightarrow \infty$.

is shown to optimize the performance of the system [12]. In this policy, each host serves jobs whose service demand is in a designated range. The SITA-E policy has been introduced in [1], [13] and, under this routing policy, the cutoffs are chosen to equalize the load in all the servers. This dispatching policy has been also studied in [14], where the authors apply SITA-E to web server farms. In [15] 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 [16], [17]. The authors in [18] 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 [4], 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 problem of how to balance the load in a server farm has been extensively studied also in the context of game theory, see [19]–[24]. 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. 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 $f(x) = \frac{dF(x)}{dx}$. We denote by x_m and x_M the minimum and maximum size of the incoming jobs to the system, and let $\gamma = \frac{x_m}{x_M} \in [0, 1]$.

We denote 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. Since each server receives traffic from only one dispatcher, a server is said to be of group i if it receives traffic from dispatcher i. We assume that the number of servers in each group is the same and equal to K/n.[2] 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 waiting time of jobs in SYS-(K,n,λ). We observe that when $n = 1$ it is the waiting time of jobs of SYS-(K,1,λ) and when $n = K$ we analyze K independent M/G/1 queues with arrival rate λ/K.

We know that in SYS-(K,n,λ) there are n groups and, in each group, there are K/n servers. 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. Besides, all the n groups are exactly equivalent and, as a result, the mean waiting time in SYS-(K,n,λ) satisfies

$$\begin{aligned}\mathbb{E}(W(K, n, x_m, x_M, \lambda)) &= \sum_{i=1}^{n} \frac{1}{n}\ \mathbb{E}(W(\frac{K}{n}, 1, x_m, x_M, \frac{\lambda}{n})) \\ &= \mathbb{E}(W(\frac{K}{n}, 1, x_m, x_M, \frac{\lambda}{n})). \qquad (1)\end{aligned}$$

This result means that SYS-(K,n,λ) and SYS-(K/n,1,λ/n) have the same performance. Therefore, the performance degradation studied can be interpreted as the economies of scale in a parallel-server system when we scale up the arrival rate and the number of servers proportionally. We shall use the *degradation factor* to assess the degradation on the performance of parallel-servers systems. We define the degradation factor as follows:

$$\begin{aligned}D(K, n, x_m, x_M) &= \frac{\mathbb{E}(W(K, n, x_m, x_M, \lambda))}{\mathbb{E}(W(K, 1, x_m, x_M, \lambda))} \\ &= \frac{\mathbb{E}(W(\frac{K}{n}, 1, x_m, x_M, \frac{\lambda}{n}))}{\mathbb{E}(W(K, 1, x_m, x_M, \lambda))}. \qquad (2)\end{aligned}$$

We have not included λ as a parameter of the degradation factor since, as we will see in Section III-D, the degradation factor does not depend on the arrival rate. When the degradation factor is close to one, we conclude that the performance of both systems is very similar. Besides, when the degradation factor is upper bounded by M, the performance of SYS-(K,n,λ) is, in the worst case, M times the performance of SYS-(K,1,λ).

Remark 1 (Randomized Load Balancing): As an example, let us calculate (2) 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 [25], 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 that 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.

[2]It is implicitly assumed that n is a divisor of K.

From (2), 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). In the next subsection, we consider a generic system with one dispatcher, R servers and arrival rate $\overline{\lambda}$, and we analyze its performance under SITA-E routing policy.

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

From the Pollaczek-Khinchine formula, it follows that in a system formed by R servers, arrival rate $\overline{\lambda}$ and one dispatcher that implements SITA-E policy, the mean waiting time is

$$\mathbb{E}(W(R,1,x_m,x_M,\overline{\lambda})) = \frac{\overline{\lambda}}{2(1-\overline{\rho})}\sum_{j=1}^{R} q_j^2\mathbb{E}(X_j^2), \qquad (3)$$

where $\overline{\rho} = \frac{\overline{\lambda}\mathbb{E}(X)}{R}$, q_j is the probability of a job to be executed in server j and $\mathbb{E}(X_j^2)$ is the second moment of the service time distribution of the tasks executed in server j.

When the job sizes distribution is continuous, there are $R+1$ thresholds $c_0, c_1, \ldots, c_R$ satisfying that $x_m = c_0 < c_1 < \cdots < c_{R-1} < c_R = x_M$ and jobs ranging in size from c_{j-1} to c_j are executed in server j. Furthermore, it is required that

$$\int_{x_m}^{c_1} xf(x)dx = \int_{c_1}^{c_2} xf(x)dx = \cdots = \int_{c_{R-1}}^{x_M} xf(x)dx. \qquad (4)$$

We note that the load in server j is given by

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

and, thus, (4) implies that the load is the same in each server. Let $z_j = c_j/x_M$ denote the scaled thresholds. In the particular cases where $j=0$ and $j=R$, we have respectively that $z_0=\gamma$ and $z_R=1$.

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

$$\mathbb{E}(X_j^2) = \int_{c_{j-1}}^{c_j} x^2\frac{f(x)}{F(c_j)-F(c_{j-1})}dx. \qquad (5)$$

Therefore, using (3), (5) and also that $q_j = F(c_j)-F(c_{j-1})$, we obtain the following expression for the mean waiting time of SYS-(R,1,$\overline{\lambda}$) for continuously distributed job sizes:

$$\mathbb{E}(W(R,1,x_m,x_M,\overline{\lambda})) = \frac{\overline{\lambda}}{2\ (1-\overline{\rho})}\sum_{j=1}^{R}(F(c_j)-F(c_{j-1}))\cdot\int_{c_{j-1}}^{c_j} x^2\ f(x)dx. \qquad (6)$$

B. Continuous Distributions: Uniform and Bounded Pareto

For uniformly distributed job sizes, if $x_m \le x \le x_M$, 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

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

The thresholds of SYS-(R,1,$\overline{\lambda}$) can be obtained from (4) and using that $f(x) = \frac{1}{x_M-x_m}$, for all $x\in[x_m,x_M]$, and are given by $c_j = \sqrt{\frac{(R-j)\ x_m^2 + j\ x_M^2}{R}},\ j=0,\ldots,R$.

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

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

The value of the thresholds for Bounded Pareto distributed job sizes of SYS-(R,1,$\overline{\lambda}$) is given in [13] and it is

$$c_j = \begin{cases} \left(\frac{R-j}{R}\ x_m^{1-\alpha} + \frac{j}{R}\ x_M^{1-\alpha}\right)^{\frac{1}{1-\alpha}}, & \text{if } \alpha \ne 1, \\ x_m\left(\frac{x_M}{x_m}\right)^{\frac{j}{R}}, & \text{if } \alpha = 1. \end{cases}$$

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 γ (see Lemma 3 and Lemma 8), we use the notation $D_U(K,n,\gamma)$ and $D_{BP(\alpha)}(K,n,\gamma)$.

C. Discrete Distribution: Two Point Distribution

Here we 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, (4) does not determine the load balancing for this distribution. Therefore, we define how the load is balanced in SYS-(R,1,$\overline{\lambda}$) when the job sizes are distributed in two points.

Let $l = \frac{R}{1+\frac{(1-p)x_M}{px_m}}$. If l is an integer, 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 an 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 $\lfloor l \rfloor$ servers that execute only short jobs and $R-\lceil l \rceil$ servers[3] 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, $\frac{(1-p_1)px_m}{\lfloor l \rfloor} = p_1px_m+p_2(1-p)x_M = \frac{(1-p)(1-p_2)x_M}{R-\lceil l \rceil}$.

We analyze in Section VI the degradation factor when the job sizes are distributed in two points and we denote it by $D_{TP(l)}(K,n,x_m,x_M)$.

D. Preliminary Results

We present how the results obtained in Section III-A can be used to give the expression for the degradation factor. We first observe that, from (3), we can obtain the mean waiting time of SYS-(K,1,λ) when $R=K$ and $\overline{\lambda}=\lambda$ and the mean waiting time of SYS-(K/n,1,λ/n) when $R=K/n$ and $\overline{\lambda}=\lambda/n$. We observe that, for both systems, $\overline{\rho}$ coincides and that the factor $\frac{\lambda}{2(1-\overline{\rho})}$ 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.

We now concentrate on continuously distributed job sizes. Let $x_0,\ldots,x_K$ denote the thresholds of SYS-(K,1,λ) and $y_0,\ldots,y_{\frac{K}{n}}$ denote the thresholds of SYS-(K/n,1,λ/n). Substituting these values in (6) gives:

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

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.

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 (7), it follows directly the expression for the degradation factor for continuously distributed job sizes.

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

$$D(K,n,x_m,x_M) = \frac{1}{n}\frac{\sum_{j=1}^{K/n}\left(F(x_{n\cdot j})-F(x_{n\cdot(j-1)})\right)\left(\int_{x_{n\cdot(j-1)}}^{x_{n\cdot j}}x^2f(x)dx\right)}{\sum_{j=1}^{K}\left(F(x_j)-F(x_{j-1})\right)\left(\int_{x_{j-1}}^{x_j}x^2f(x)dx\right)}, \quad (8)$$

where the thresholds $x_m = x_0, x_1, \ldots, x_{K-1}, x_K = x_M$ satisfy $\int_{x_m}^{x_1} xf(x)dx = \int_{x_1}^{x_2} xf(x)dx = \cdots = \int_{x_{K-1}}^{x_M} xf(x)dx$.

IV. Uniformly Distributed Job Sizes

In this section, we focus on the degradation factor when the job sizes are uniformly distributed. It is trivial to check that the scaled thresholds are $z_j = \sqrt{\frac{(K-j)\gamma^2+j}{K}}$, $j=0,\ldots,K$.

We now observe that this distribution satisfies that $f(x)>0$ for all $x\in[x_m,x_M]$. Therefore, we can use the result of Proposition 1 to compute the degradation factor when the job sizes are uniformly distributed. 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 3: The degradation factor for uniformly distributed job sizes only depends on K, n and γ and it is given by

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

A. The case $K=2$

We study the degradation factor for a two-server system and uniformly distributed job sizes. From (9) and noting that $z_1 = \sqrt{\frac{\gamma^2+1}{2}}$, we obtain that the degradation factor for uniformly distributed job sizes in a system with two servers is

$$D_U(2,2,\gamma) = \frac{1}{2}\frac{(1-\gamma)(1-\gamma^3)}{(1-z_1)(1-z_1^3)+(z_1-\gamma)(z_1^3-\gamma^3)}.$$

We now show that $D_U(2,2,\gamma)$ decreases with γ.

Lemma 4: For $\gamma<1$, $D_U(2,2,\gamma)$ is decreasing with γ.

We use this result and Lemma 1 to give a lower bound and an upper bound of $D_U(2,2,\gamma)$.

Proposition 2: $1 \le D_U(2,2,\gamma) \le \lim_{\gamma\to0} D_U(2,2,\gamma) = \frac{1}{2}\frac{1}{(1-(\frac{1}{2})^{\frac{1}{2}})(1-(\frac{1}{2})^{\frac{3}{2}})+1/4} \approx 1.138$.

B. The case $K>2$

In this section, we study the degradation factor for arbitrary K. When $\gamma\to0$, we have $z_j \to \sqrt{\frac{j}{K}}$. Therefore, it follows from (9) that $\lim_{\gamma\to0} D_U(K,n,\gamma) = \frac{n}{s(K)}\frac{s(K/n)}{}$,where, for all integers m, $s(m) = \sum_{j=1}^{m}(j^{3/2}-(j-1)^{3/2})(j^{1/2}-(j-1)^{1/2})$.

1) Fixed Number of Servers in Each Group: We study the value of $\lim_{\gamma\to0} D_U(K,n,\gamma)$ when the number of servers in each group $p=\frac{K}{n}$ is fixed. We first give the following lemma.

Lemma 5:

(a) $s(m)-s(m-1)$ is a decreasing function of m.
(b) $\frac{s(m)}{m}$ is decreasing with m.
(c) Fix p, $\lim_{\gamma\to0} D_U(K,n,\gamma)$ increases with K.
(d) $\lim_{K\to\infty}\frac{s(K)}{K} = \frac{3}{4}$.
(e) Fix p, then $\lim_{K\to\infty}\lim_{\gamma\to0} D_U(K,n,\gamma) = \frac{4}{3}\cdot\frac{s(p)}{p}$.
(f) For all integers m, $s(m+1)-s(m) \ge 3/4$.

Using Lemma 5(c) and Lemma 5(e), we give an upper bound of $\lim_{\gamma\to0} D_U(K,n,\gamma)$ when the number of servers in each group is fixed.

Lemma 6: Fix p. Then, $\lim_{\gamma\to0} D_U(K,n,\gamma) \le \frac{4}{3}\cdot\frac{s(p)}{p}$.

2) Fixed Number of Groups: We analyze the behavior of $\lim_{\gamma\to0} D_U(K,n,\gamma)$ when n is fixed. Using Lemma 5, we show that, when we fix n, the maximum of $\lim_{\gamma\to0} D_U(K,n,\gamma)$ is achieved when $K=n$.

Lemma 7: Fix n. Then $\lim_{\gamma\to0} D_U(K,n,\gamma) \le \lim_{\gamma\to0} D_U(n,n,\gamma) = \frac{n}{s(n)}$.

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

3) Degradation Factor: We now present how, using the results of Section IV-B1 and Section IV-B2, we can study the performance degradation for arbitrary K and uniformly distributed job sizes.

We assume that $D_U(K,n,\gamma)$ decreases with γ when $K>2$ and any n. Unfortunately, given the difficulty of the expression (9), we have not succeeded to generalize the result of Lemma 4 to a system with more than two servers. From extensive numerical experiments, we conjecture that the degradation factor decreases with γ for $K>2$ and any n.

Conjecture 1: $D_U(K,n,\gamma)$ decreases with γ, for $K>2$ and any n.

In the following result, we give a lower bound and an upper bound for $D_U(K,n,\gamma)$, where $K>2$ under this conjecture.

Proposition 3: Assume Conjecture 1 holds. Then, $1 \leq D_U(K,n,\gamma) \leq 4/3$.

Using Lemma $5(b)$ and Lemma $5(d)$, we show that the upper bound is tight when $K=n$ and $K\to\infty$.

Corollary 1: Assume Conjecture 1 holds. When $K=n$ and $K\to\infty$, the degradation factor for uniformly distributed job sizes equals $4/3$.

V. Bounded Pareto Distributed Job Sizes

In this section, we concentrate on the degradation factor for Bounded Pareto distributed job sizes.

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

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

Since $f(x)>0$ for all $x\in[x_m,x_M]$, the result of Proposition 1 can be used to obtain the expression of the degradation factor for Bounded Pareto distributed job sizes. Moreover, the proof of Lemma 3 applies *mutatis mutandis* to show that the degradation factor for Bounded Pareto distributed job sizes depends on x_m and on x_M only through γ.

Lemma 8: The degradation factor for Bounded Pareto distributed job sizes only depends on K, n, α and γ and it is given by $D_{BP(\alpha)}(K,n,\gamma) = \frac{1}{n}\frac{\sum_{j=1}^{K/n}(z_{n\cdot j}^{2-\alpha}-z_{n\cdot(j-1)}^{2-\alpha})(z_{n\cdot(j-1)}^{-\alpha}-z_{n\cdot j}^{-\alpha})}{\sum_{j=1}^{K}(z_j^{2-\alpha}-z_{j-1}^{2-\alpha})(z_{j-1}^{-\alpha}-z_j^{-\alpha})}$.

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 [4] show that SITA-E optimizes the performance of a system with two servers and Bounded Pareto distributed jobs sizes with $\alpha=1$. From Lemma 8 and (10), it results that

$$D_{BP(1)}(K,n,\gamma) = \frac{1}{n^2}\cdot\frac{\gamma^{\frac{-n}{K}}\ (1-\gamma^{\frac{n}{K}})^2}{\gamma^{\frac{-1}{K}}\ (1-\gamma^{\frac{1}{K}})^2}. \qquad (11)$$

We show that this expression decreases with γ.

Lemma 9: $D_{BP(1)}(K,n,\gamma)$ is a decreasing function of γ.

Using this result and Lemma 1 and noting, from (11), that $D_{BP(1)}(K,n,\gamma)$ tends to infinity when $\gamma\to 0$, we give the following result.

Proposition 4: $D_{BP(1)}(K,n,\gamma)\geq 1$ and it tends to infinity when $\gamma\to 0$.

From this result, we state that the performance of SYS-(K/n,1,λ/n) is, in the worst case, infinite times worse that the performance of SYS-(K,1,λ). In particular, this ratio equals infinity when $x_M\to\infty$, in which case we know that 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 give the value of $D_{BP(\alpha)}(K,n,\gamma)$ when $\gamma\to 0$, i.e., when the ratio between x_m and x_M tends to zero.

Lemma 10: If $\alpha\neq 1$, $\lim_{\gamma\to 0} D_{BP(\alpha)}(K,n,\gamma) = n^{\frac{1}{|1-\alpha|}}$.

It is important to note that, when $\gamma\to 0$, the degradation factor for Bounded Pareto distributed job sizes with $\alpha\neq 1$ does not depend on K.

We observe, see numerical section, that the degradation factor for Bounded Pareto distributed job sizes with $\alpha\neq 1$ decreases with γ. Given the difficulty of the expression (8) as well as the scaled thresholds (10), we have not succeeded in showing this monotonicity property when $\alpha\neq 1$. We have performed many numerical experiments to conjecture that the degradation factor decreases with γ when $\alpha\neq 1$.

Conjecture 2: When $\alpha\neq 1$, $D_{BP(\alpha)}(K,n,\gamma)$ is a decreasing function of γ, for all K and n.

Under this assumption, the minimum of the degradation factor is achieved when $\gamma\to 1$ and the maximum when $\gamma\to 0$. From the results of Lemma 1 and Lemma 10, we give a lower bound and an upper bound of $D_{BP(\alpha)}(K,n,\gamma)$ when $\alpha\neq 1$.

Proposition 5: Assume Conjecture 2 holds. Then, when $\alpha\neq 1$, $1\leq D_{BP(\alpha)}(K,n,x_m,x_M)\leq n^{\frac{1}{|1-\alpha|}}$.

We observe that $n^{\frac{1}{|1-\alpha|}}$ is infinite when $\alpha\to 1$ for any n. Therefore, we conclude from Proposition 4 and Proposition 5 that the limits when γ goes to zero and when α tends to one interchange for Bounded Pareto job sizes distribution, i.e., $\lim_{\gamma\to 0}\lim_{\alpha\to 1} D_{BP(\alpha)}(K,n,\gamma) = \lim_{\alpha\to 1}\lim_{\gamma\to 0} D_{BP(\alpha)}(K,n,\gamma)$.

VI. 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 a 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: $D_{TP(1)}(2,2,\gamma) = \frac{(1+\gamma)^2}{4\gamma}$. It is easy to see that this expression is decreasing with γ for all $\gamma \in [0,1]$ and, as a result, an upper bound and a lower bound are given when $\gamma \to 0$ and $\gamma \to 1$, respectively. From Lemma 1 and since the degradation factor tends to infinity when $\gamma \to 0$, it implies the following result:

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

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 that 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 gives

$$D_{TP(l)}(2,2,\gamma,p_1) = \frac{1}{2}\frac{p\gamma^2+(1-p)}{p^2(1-p_1)^2\gamma^2+(p_1p+(1-p))(p_1p\gamma^2+(1-p))}. \quad (12)$$

We show that (12) 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 $p_1 \to 0$, (12) 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 $l > 1$, $D_{TP(l)}(2,2,\gamma,p_1) \leq \lim_{p_1\to 0} D_{TP(l)}(2,2,\gamma,p_1) = D_{TP(1)}(2,2,\gamma)$. 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 $\gamma \to 0$ and $p_1 \to 0$.

When $l < 1$, the situation is very similar to that of $l > 1$. In this case, we have that $D_{TP(l)}(2,2,\gamma,p_2) \leq \lim_{p_2\to 0} D_{TP(l)}(2,2,\gamma,p_2) = D_{TP(1)}(2,2,\gamma)$ 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 $\gamma \to 0$ and $p_2 \to 0$.

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 an 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 SYS-(K/n,1,λ/n) when l is a multiple of n. Thus, for SYS-(K/n,1,λ/n), we define $l^* = \frac{K/n}{1+\frac{(1-p)}{px_m}}$ and, if l^* is an 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 an integer since l is a 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 SYS-(K/n,1,λ/K) coincides with that SYS-(K,1,λ) when l is a multiple of n.

When l is not a multiple of n, in SYS-(K/n,1,λ/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 exists 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.

VII. Numerical Computations

A. Monotonicity Assumptions

We aim to check that the properties of Conjecture 1 and Conjecture 2 hold. 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 a few results that are illustrative of the general pattern.

In Figure 3, we represent the evolution of $D_U(K,n,\gamma)$ over γ when $K = 1000$ and different values of n (note that the x-axis is in the logarithmic scale). We observe that, in all the instances, the degradation factor decreases with γ. Furthermore, we see that, when $n = 1000$ and $\gamma \to 0$, the degradation factor tends to $4/3$, which is the upper-bound given in Proposition 3. However, in the other cases, the degradation factor is strictly less than $4/3$, when $\gamma \to 0$.

We also investigate the degradation factor for Bounded Pareto distributed job sizes with $\alpha \neq 1$. In Figure 4, we consider a system with 1000 servers and $\alpha = 1.5$ and we plot the evolution of the degradation factor with respect to γ for different values of n. We observe that the degradation factor in all the instances is always decreasing with γ, as stated in Conjecture 2. In addition, we observe that $D_{BP(\alpha)}(K,n,\gamma)$ tends to n^2 when $\gamma \to 0$, which coincides with the value given in Lemma 10.

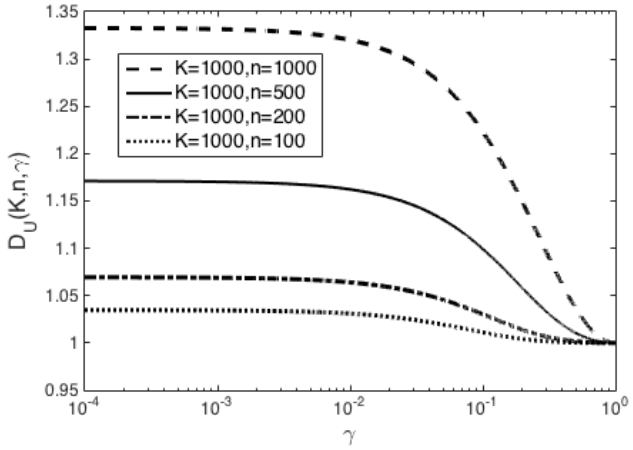

Fig. 3: Evolution of the degradation factor for uniformly distributed job sizes with respect to γ(x-axis in logarithmic scale).

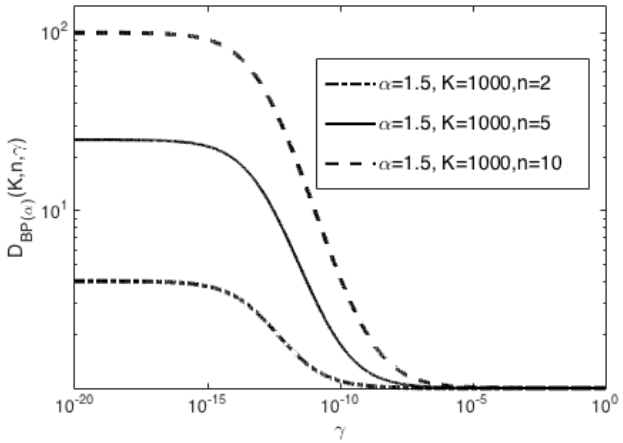

Fig. 4: Evolution of the degradation factor for Bounded Pareto distributed job sizes with parameter $\alpha = 1.5$ with respect to γ (x-axis and y-axis in logarithmic scale).

B. Degradation Factor

1) Bounded Pareto: We now study the degradation factor for Bounded Pareto. We know from the results of Section V 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 [13]. As we saw in Lemma 8, 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 II, we show the degradation factor when $n = 100$, $n = 500$ and $n = 1000$ for different values of α. We also present in Table II 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 computer and networking systems [4]. 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 $n = 1000$.

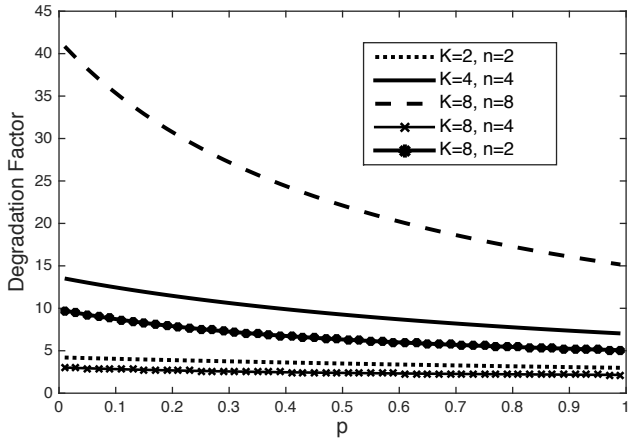

Fig. 5: Evolution over p of the degradation factor for Degenerated Hyperexponential distributed job sizes.

2) Degenerate Hyper-exponential: We consider a system with SITA-E policy and Degenerate Hyper-exponential distributed 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 (8) for this distribution when p varies.

In Figure 5, 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 significant, and that the degradation increases as the variability of jobs increases.

C. Optimal SITA Degradation Factor

We now consider a system with two servers and we compare SYS-(2,2,λ) and SYS-(2,1,λ) for Bounded Pareto distributed

	$n = 100$		$n = 500$		$n = 1000$	
	$D_{BP(\alpha)}(K, n, \gamma)$	$n^{\frac{1}{\lvert 1-\alpha \rvert}}$	$D_{BP(\alpha)}(K, n, \gamma)$	$n^{\frac{1}{\lvert 1-\alpha \rvert}}$	$D_{BP(\alpha)}(K, n, \gamma)$	$n^{\frac{1}{\lvert 1-\alpha \rvert}}$
$\alpha = 0.25$	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 II: Degradation factor for Bounded Pareto distributed job sizes when $K = 1000$ and $\gamma = \frac{9}{10^{14}}$ compared with $n^{\frac{1}{\lvert 1-\alpha \rvert}}$.

	Optimal SITA Degradation Factor		
	$\rho = 0.005$	$\rho = 0.5$	$\rho = 0.8$
$\alpha = 0.25$	333.74	87.77	8.6594
$\alpha = 0.5$	$2.2476 \cdot 10^4$	4219.9	18.7679
$\alpha = 0.75$	$3.3604 \cdot 10^5$	$1.3187 \cdot 10^5$	133.8889
$\alpha = 1.25$	$3.3604 \cdot 10^5$	$1.3187 \cdot 10^5$	133.8889
$\alpha = 1.5$	$2.2476 \cdot 10^4$	4219.9	18.7679
$\alpha = 1.75$	333.74	87.77	8.6594

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

job sizes when the SITA thresholds are chosen to optimize the performance. In this case, the ratio of performances is said to be the optimal SITA degradation factor. According to the result of [4], in a two server system, the degradation factor coincides with the optimal SITA degradation factor 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 the case where $\alpha \neq 1$.

Our objective is to know whether the optimal SITA degradation is high or not. We use the numerical results of [4], where they consider a two-server system and they obtain numerically the ratio of the performance of SITA-E policy over the performance of the optimal SITA policy. In our computations, we compute the optimal SITA degradation factor for the same parameters as theirs. To do so, we multiply the performance ratio they obtain in their simulations with the degradation factor obtained in (8). Hence, in Table III, we represent the optimal SITA degradation factor for low load ($\rho = 0.005$), medium load ($\rho = 0.5$) and high load ($\rho = 0.8$) and for different values of α. As it can be seen in Table III, the optimal SITA degradation factor is very high in some instances. For example, if $\rho = 0.005$, when $\alpha = 1.25$ and when $\alpha = 1.5$, the optimal SITA degradation factor is, respectively, $3.3604 \cdot 10^5$ and $2.2476 \cdot 10^4$.

REFERENCES

[1] M. Harchol-Balter, M. Crovella, and C. Murta, "Task assignment in a distributed system: Improving performance by load unbalancing," in *Proceedings of SIGMETRICS*, 1998.

[2] T. Wang, Z. Su, Y. Xia, and M. Hamdi, "Rethinking the data center networking: Architecture, network protocols, and resource sharing," *IEEE Access*, vol. 2, pp. 1481–1496, 2014.

[3] B. Schroeder and M. Harchol-Balter, "Evaluation of task assignment policies for supercomputing servers: The case for load unbalancing and fairness," *Cluster Computing*, vol. 7, no. 2, pp. 151–161, 2004.

[4] M. Harchol-Balter and R. Vesilo, "To balance or unbalance load in size-interval task allocation," *Probability in the Engineering and Informational Sciences*, vol. 24, no. 2, pp. 219–244, Apr. 2010.

[5] J. Doncel, S. Aalto, and U. Ayesta, "Economies of scale in parallel-server systems," *HAL Technical Report*, 2017.

[6] F. Semchedine, L. Bouallouche-Medjkoune, and D. Aissani, "Task assignment policies in distributed server systems: A survey," *Journal of Network and Computer Applications*, vol. 34, no. 4, pp. 1123 – 1130, 2011.

[7] M. Harchol-Balter, *Performance Modeling and Design of Computer Systems: Queueing Theory in Action*. Cambridge Univ. Press, 2013.

[8] R. D. Foley and D. R. McDonald, "Join the shortest queue: Stability and exact asymptotics," *Annals of Applied Probab.*, vol. 11, no. 3, 2001.

[9] V. Gupta, M. Harchol-Balter, K. Sigman, and W. Whitt, "Analysis of join-the-shortest-queue routing for web server farms," *Performance Evaluation*, vol. 64, no. 9, pp. 1062–1081, 2007.

[10] M. Mitzenmacher, "The power of two choices in randomized load balancing," *IEEE Transaction on Parallel and Distributed Systems*, vol. 12, no. 10, 2001.

[11] A. W. Richa, M. Mitzenmacher, and R. Sitaraman, "The power of two random choices: A survey of techniques and results," *Handbook of Randomized Computing*, vol. 1, 2001.

[12] H. Feng, V. Misra, and D. Rubenstein, "Optimal state-free, size-aware dispatching for heterogeneous M/G/-type systems," *Performance Evaluation*, vol. 62, no. 1-4, pp. 475–492, Oct. 2005.

[13] 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*, vol. 59, no. 2, pp. 204 – 228, 1999.

[14] G. Ciardo, A. Riska, and E. Smirni, "Equiload: a load balancing policy for clustered web servers," *Performance Evaluation*, vol. 46, no. 2, 2001.

[15] M. Harchol-Balter, "Task assignment with unknown duration," in *International Conference on Distributed Computing Systems*, 2000.

[16] E. Bachmat and H. Sarfati, "Analysis of SITA policies," *Performance Evaluation*, vol. 67, no. 2, pp. 102–120, 2010.

[17] 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.

[18] 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.

[19] C. H. Bell and S. Stidham, "Individual versus social optimization in the allocation of customers to alternative servers," *Management Science*, vol. 29, pp. 831–839, 1983.

[20] A. Orda, R. Rom, and N. Shimkin, "Competitive routing in multiuser communication networks," *IEEE/ACM Transactions on Networking*, vol. 1, no. 5, pp. 510–521, 1993.

[21] T. Roughgarden and É. Tardos, "How bad is selfish routing?" *Journal of the ACM*, vol. 49, no. 2, pp. 236–259, 2002.

[22] M. Haviv and T. Roughgarden, "The price of anarchy in an exponential multi-server," *Operations Research Letters*, vol. 35, pp. 421–426, 2007.

[23] E. Altman, U. Ayesta, and B. J. Prabhu, "Load balancing in processor sharing systems," *Telecommunication Systems*, vol. 47, no. 1, 2011.

[24] 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)*, vol. 14, no. 2-3, p. 18, 2014.

[25] R. B. Cooper, "Introduction to queueing theory. 1981," *Edward Arnold, London*, 2004.