Performance Degradation in Parallel-Server Systems

Josu Doncel, Samuli Aalto, Urtzi Ayesta

Performance Degradation in Parallel-Server Systems

Josu Doncel, Samuli Aalto and Urtzi Ayesta

***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. 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 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. We have investigated the degradation using the distribution obtained from real traces.**

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, 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 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) and by the Basque Government through 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.

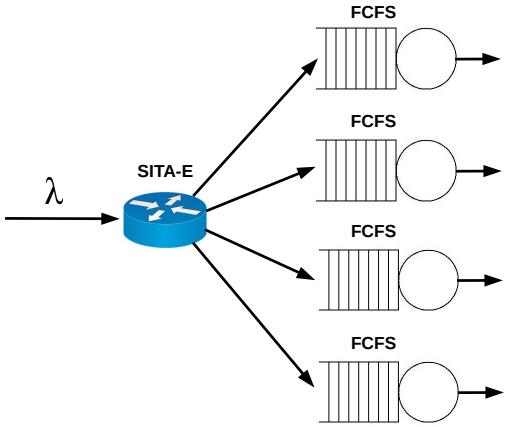

Fig. 1: SYS-(4,1,λ). There is one dispatcher that receives all the traffic and sends it to all the servers.

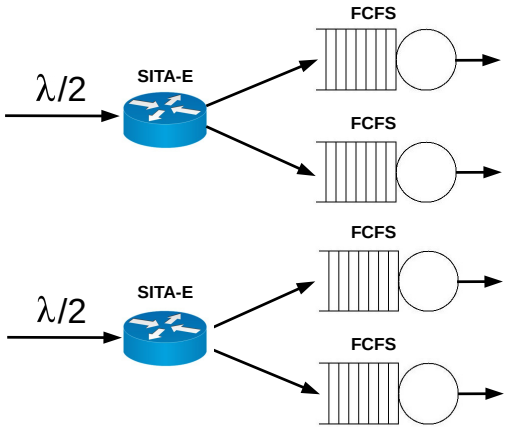

Fig. 2: SYS-(4,2,λ). There are two dispatchers and each of them controls a half of the total incoming traffic and balances the load to two servers.

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).

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) [27]. 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 [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 function, 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, 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 upper bounded by 1.138. For an infinite number of servers, we show that the degradation factor is upper-bounded by $4/3$. For $K > 2$ and finite number of servers, assuming that the degradation factor decreases with γ, we prove that the degradation factor is 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 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 $K \to \infty$. When $\alpha \neq 1$ 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 distributed job sizes 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 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 impossible even for a system with two servers [18]. 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. Finally, we present the numerical experiments in Section VIII.

A conference 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 dispatchers 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. Another important routing policy is the Power of Two [20], [22], 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. When the service demand is known 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], [16] 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

	Degradation Factor	Result
Arbitrary continuous distribution	$1 \leq D(K,n) \leq D(K,K)$	Corollary 1
Uniform distribution: $K = 2$	$D(K,K) \leq 1.138.$	Proposition 2
Uniform distribution: $K \to \infty$	$D(K,K) \leq 4/3.$	Proposition 3
Uniform distribution: $K > 2$ and finite	$D(K,K) \leq 4/3.$	Proposition 4
Bounded Pareto distribution: $\alpha = 1$	$D(K,K) \to \infty$[1]	Proposition 5
Bounded Pareto distribution: $\alpha \neq 1$ and $K \to \infty$	$D(K,K) \to \infty$[1]	Proposition 6
Bounded Pareto distribution: $\alpha \neq 1$ and K finite	$D(K,K) \leq K^{\frac{1}{\lvert 1-\alpha \rvert}}.$	Proposition 7
Two Point: $K = 2$, Equally Loaded Jobs	$D(K,n) \geq 1$ and $D(K,K) \to \infty$[1]	Proposition 8
Two Point: $K = 2$, Unequally Loaded Jobs	$D(K,n) \geq 1$ and $D(K,K) \to \infty$[1]	Proposition 9 Proposition 10

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

analysis for the Bounded Pareto distribution has been done in [4], [26]. The authors in [17] 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 [18], 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 literateture, but in different contexts of our work. The author in [28] analyze the optimal server utilization and he provides a simple approximation of the mean steady state waiting time. In [29] the author considers an M/M/K queue and analyzes the economies of scale for different performace 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], [19], [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. 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, it is important to note that 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 shall use the *degradation factor* to assess the degradation

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

on the performance of parallel-servers systems. We define the degradation factor as follows:

$$D(K,n,x_m,x_M) = \frac{\mathbb{E}(W(K,n,x_m,x_M,\lambda))}{\mathbb{E}(W(K,1,x_m,x_M,\lambda))} = \frac{\mathbb{E}(W(\frac{K}{n},1,x_m,x_M,\frac{\lambda}{n}))}{\mathbb{E}(W(K,1,x_m,x_M,\lambda))}. \quad (2)$$

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. 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, 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.*

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). Therefore, we analyze the performance of a generic SYS-(R,1,$\overline{\lambda}$), which is a system with one dispatcher, R servers and arrival rate $\overline{\lambda}$. Prior to that, we present the SITA-E routing policy for SYS-(R,1,$\overline{\lambda}$).

A. SITA-E Routing

If the sizes of incoming jobs is known, the dispatcher can perform the so-called SITA routing, that rougthly 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 $c_0, c_1, \ldots, c_R$ 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.

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 equation:

$$\int_{x_m}^{c_1} x f(x)dx = \int_{c_1}^{c_2} x f(x)dx = \cdots = \int_{c_{R-1}}^{x_M} x f(x)dx, \quad (3)$$

whereas the computation of the thresholds of the optimal SITA policy is known to be impossible even for a system with two servers [18].

We now observe that if (3) is satisfied, then the load of all the server is equal. In fact, 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, (3) 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,$\overline{\lambda}$)

We study SYS-(R,1,$\overline{\lambda}$), which is a system that consists of R servers, one dispatcher that implements SITA-E load balancing and arrival rate $\overline{\lambda}$. 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

$$\mathbb{E}(W(R,1,x_m,x_M,\overline{\lambda})) = \frac{\overline{\lambda}}{2(1-\overline{\rho})} \sum_{j=1}^{R} (F(c_j) - F(c_{j-1}))^2 \mathbb{E}(X_j^2), \quad (4)$$

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

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

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

Therefore, using (4), (5), we obtain the following expression for the mean waiting time of SYS-(R,1,$\overline{\lambda}$) for continuously distributed job sizes:

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

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,$\overline{\lambda}$) can be obtained from (3) 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 $x_m \leq x \leq x_M$, $f(x) = \frac{\alpha\ x_m^\alpha}{1-(x_m/x_M)^\alpha}\ x^{-\alpha-1}$, and $f(x) = 0$ otherwise, where $\alpha > 0$. 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,$\overline{\lambda}$) is given in [16] 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 γ (see Lemma 5 and Lemma 8), we 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, (3) 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 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:
$$\frac{(1-p_1)px_m}{R-1} = p_1 p x_m + (1-p)x_M.$$
- 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:
$$p x_m + p_2(1-p)x_M = \frac{(1-p_2)(1-p)x_M}{R-1}.$$
- If $1 < l < R-1$, there are $\lfloor l \rfloor$ servers that execute only short jobs and $R - \lceil l \rceil$ [3] 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,
$$\frac{(1-p_1)px_m}{\lfloor l \rfloor} = p_1 p x_m + p_2(1-p)x_M = \frac{(1-p)(1-p_1)x_M}{R - \lceil l \rceil}.$$

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 $\mathbf{x} = (x_m, \ldots, x_M)$ and a probability distribution $\mathbf{p}$.

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 (4), 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 $\overline{\lambda} = \lambda/n$. Besides, for both systems, $\overline{\rho}$ coincides and 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.

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) it results:

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

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

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 $y_j = x_{n\cdot j}$.*

Proof. See Appendix A. □

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 $x \in [x_m, x_M]$,*

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

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

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

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 $j = 1, \ldots, k$. Then,*

(i) $p_1 + \cdots + p_k = 1$,
(ii) $p_1 \geq \cdots \geq p_k \geq 0$,
(iii) $s_k \geq \cdots \geq s_2 \geq s_1 \geq 0$.

Proof. See Appendix B. □

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 $s_1, \ldots, s_k$ and $p_1, \ldots, p_k$ of Lemma 3. The key result to prove the monotonicity property is the following:

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

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

Proof. See Appendix C. □

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

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 $i = 1, \ldots, m-n+1$. 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,

$$\sum_{i=n}^{m} \frac{p_i}{\sum_{j=n}^{m} p_j} s_j \leq \frac{1}{m-n+1}\sum_{i=n}^{m} s_i, \tag{9}$$

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 $i = 1, \ldots, k_2$, $\bar{p}_i = \frac{1}{l}\sum_{j=(i-1)l+j}^{il} p_j$. 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 (9). □

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 consider $n_1 = n$ and $n_2 = K$. Indeed, for this case, we derive 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 analyze 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 upper-bound of 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, for a two server system, we show that the degradation factor is upper bounded by 1.139. For a system with $K \to \infty$ servers, we show that the degradation factor is upper bounded by $4/3$. And for a system with $K > 2$ and finite, assuming that $D_U(K, K, \gamma)$ decreases as γ increases, we show that the degradation factor is upper bounded by $4/3$.

It is trivial to check that the scaled thresholds are

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

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*

$$D_U(K, n, \gamma) = \frac{1}{n} \frac{\sum_{j=1}^{K/n} (z_{n\cdot j} - z_{n\cdot(j-1)})(z_{n\cdot j}^3 - z_{n\cdot(j-1)}^3)}{\sum_{j=1}^{K} (z_j - z_{j-1})(z_j^3 - z_{j-1}^3)}. \tag{10}$$

Proof. See Appendix D. □

A. The case $K = 2$

We study the degradation factor for a two-server system and uniformly distributed job sizes. From (10) 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 6. *For $\gamma < 1$, $D_U(2, 2, \gamma)$ is decreasing with γ.*

Proof. See Appendix E. □

We use this result to give a lower bound of $D_U(2, 2, \gamma)$.

Proposition 2. *For two servers and uniformly distributed job sizes, the degradation factor is upper-bounded by 1.139*

Proof. The result follows from Lemma 6 and using that

$$\lim_{\gamma \to 0} D_U(2, 2, \gamma) = \frac{1}{2} \frac{1}{(1 - (\frac{1}{2})^{\frac{1}{2}})(1 - (\frac{1}{2})^{\frac{3}{2}}) + 1/4} \approx 1.138.$$

□

B. The case $K \to \infty$

We now analyze the degradation factor when the number of servers tends to infinity. We denote by $D_U(\infty, \infty, \gamma)$ the degradation factor when $n = K$ and $K \to \infty$. When $K = n$, the numerator of (10) is $(1-\gamma)(1-\gamma^3)$. Besides, when $K \to \infty$, from Riemann's integral, it is satisfied that

$$K \sum_{i=1}^{K} (z_i - z_{i-1})(z_i^3 - z_{i-1}^3) \to \int_0^1 (z_i)' \cdot (z_i^3)' \cdot d\gamma = \frac{3}{4}(1-\gamma^2)^2$$

since $z_i = \sqrt{\gamma^2 + \frac{j}{K}(1-\gamma^2)}$. Therefore, the degradation factor when $n = K$ and $K \to \infty$ is

$$D_U(\infty, \infty, \gamma) = \frac{4}{3} \frac{(1-\gamma)(1-\gamma^3)}{(1-\gamma^2)^2} \tag{11}$$

We show that this expression decreases with γ.

Lemma 7. *$D_U(\infty, \infty, \gamma)$ is a decreasing function of γ.*

Proof. See Appendix F. □

From this result, it follows that $D_U(\infty, \infty, \gamma)$ is upper-bounded by $D_U(\infty, \infty, 0)$, which is $4/3$ from (11).

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

C. The case $2 < K < \infty$

We now study the degradation factor when the number of servers is higher than two and finite. We assume that $D_U(K, K, \gamma)$ decreases with γ when K is finite and higher than two. In the numerical part of this work, we present the experiment that show that this monotonicity property is verified. Unfortunately, given the difficulty of the expressions, we did not succeed in extending the result of Lemma 7 for this case.

Conjecture 1. *$D_U(K, K, \gamma)$ decreases with γ, for any finite K.*

Under this assumption, we can state that the degradation factor is upper-bounded by $\lim_{\gamma \to 0} D_U(K, K, \gamma)$, which is $4/3$.

Proposition 4. *Assume that Conjecture 1 holds. Then, for K higher than two and finite 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 $K \to \infty$, we show that the degradation factor is unbounded from above. For $\alpha \neq 1$ 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:

$$z_j = \begin{cases} \left(\frac{K-j}{K} + \frac{j}{K}\ \gamma^{1-\alpha}\right)^{\frac{1}{1-\alpha}}, & \alpha \neq 1, \\ \gamma^{1-\frac{j}{K}}, & \alpha = 1. \end{cases} \quad (12)$$

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

$$D_{BP(\alpha)}(K,n,\gamma) = \frac{1}{n}\frac{\sum_{j=1}^{K/n}(z_{n\cdot j}^{2-\alpha} - z_{n\cdot(j-1)}^{2-\alpha})(z_{n\cdot(j-1)}^{-\alpha} - z_{n\cdot j}^{-\alpha})}{\sum_{j=1}^{K}(z_j^{2-\alpha} - z_{j-1}^{2-\alpha})(z_{j-1}^{-\alpha} - z_j^{-\alpha})}. \quad (13)$$

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 [18] 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 (12) and simplifying, it results that

$$D_{BP(1)}(K,n,\gamma) = \frac{1}{n^2}\cdot\frac{\gamma^{\frac{-n}{K}}\ (1-\gamma^{\frac{n}{K}})^2}{\gamma^{\frac{-1}{K}}\ (1-\gamma^{\frac{1}{K}})^2}. \quad (14)$$

We show that this expression decreases with γ.

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

Proof. See Appendix G. □

Using the previous result and noting, from (14), that $D_{BP(1)}(K,K,\gamma)$ tends to infinity when $\gamma \to 0$, we give the following result.

Proposition 5. *$D_{BP(1)}(K,K,\gamma)$ 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 fact, this ratio equals infinity when $\gamma \to 0$, 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 consider that the number of servers is infinite and then we analyze the case of a finite number of servers.

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

$$K\sum_{i=1}^{K}(z_i^{-\alpha} - z_{i-1}^{-\alpha})(z_i^{2-\alpha} - z_{i-1}^{2-\alpha}) \to \int_0^1 (z_i^{-\alpha})' \cdot (z_i^{2-\alpha})' \cdot d\gamma = \frac{\alpha\cdot(2-\alpha)}{-(1-\alpha)^2}(1-\gamma^{1-\alpha})^2,$$

since $z_i = (\gamma^{1-\alpha} + \frac{j}{K}(1-\gamma^{1-\alpha}))^{\frac{1}{1-\alpha}}$, where $\alpha > 0$. Therefore, the degradation factor for Bounded Pareto distributed job sizes and $K \to \infty$ and $n = K$ is

$$D_{BP(\alpha)}(\infty,\infty,\gamma) = \frac{-(1-\alpha)^2}{\alpha\cdot(2-\alpha)}\cdot\left(1 - \frac{(1-\gamma)^2}{\gamma^\alpha(1-\gamma^{1-\alpha})^2}\right). \quad (15)$$

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

Lemma 10. *$D_{BP(\alpha)}(\infty,\infty,\gamma)$ is a decreasing function of γ.*

Proof. See Appendix H. □

From (15), it follows that the degradation factor is unbounded from above when $\gamma \to\ 0$. F

Proposition 6. *$D_{BP(\alpha)}(\infty,\infty,\gamma)$ tends to infinity when $\gamma \to 0$.*

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 $\gamma \to 0$, i.e., when the difference between x_m and x_M tends to infinity.

Lemma 11. *If $\alpha \neq 1$,*

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

Proof. See Appendix I. □

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. Besides, $n^{\frac{1}{|1-\alpha|}}$ is infinite when $\alpha \to 1$ for any n. Therefore, we conclude from Proposition 5 and Lemma 11 that the limits when γ goes to zero and when α tends to one interchange for Bounded Pareto job sizes distribution, i.e.,

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

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 (13) as well as the scaled thresholds (12), 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 2. *When $\alpha \neq 1$ 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,

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

where the last equality is given by Lemma 11.

Proposition 7. *Assume Conjecture 2 holds. Then, the degradation factor factor for Bounded Pareto distributed job sizes with alpha $\neq 1$ 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 7 and Proposition 6 that the limits when γ goes to zero and K goes to infinity interchanges, that is,

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

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 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 (4), 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 $\gamma \to 0$ 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 8. *$D_{TP(1)}(2,2,\gamma) \geq 1$ 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 (4) 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

$$D_{TP(l)}(2,2,\gamma,p_1) = \frac{1}{2}\frac{p\gamma^2+(1-p)}{p^2(1-p_1)^2\gamma^2+(p_1p+(1-p))(p_1p\gamma^2+(1-p))}. \quad (16)$$

We show that (16) decreases with γ.

Lemma 12. *When $l > 1$, $D_{TP(l)}(2,2,\gamma,p_1)$ is a decreasing function of γ.*

Proof. See Appendix J. □

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 $p_1 \to 0$, (16) 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 $l > 1$,

$$\begin{aligned} D_{TP(l)}(2,2,\gamma,p_1) &\leq \lim_{p_1\to 0} D_{TP(l)}(2,2,\gamma,p_1) \\ &= D_{TP(1)}(2,2,\gamma). \end{aligned}$$

Thus, from Proposition 8, it follows that $D_{TP(1)}(2,2,\gamma)$ is unbounded from above.

Proposition 9. *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

$$\begin{aligned} D_{TP(l)}(2,2,\gamma,p_2) &\leq \lim_{p_2\to 0} D_{TP(l)}(2,2,\gamma,p_2) \\ &= D_{TP(1)}(2,2,\gamma), \end{aligned}$$

and the same techniques as in Lemma 12 show that the degradation factor is decreasing with γ when $l < 1$. As a consequence, we give the following result.

Proposition 10. *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 consider 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 SYS-(K/n1,λ/n) when l is a multiple of n. Thus, for SYS-(K/n1,λ/n), we define

$l^* = \frac{K/n}{1+\frac{(1-p)}{px_m}}$ 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 SYS-(K/n,1,λ/K) coincides with that SYS-(K,1,λ) when l is a multiple of n.

When l is not multiple of n, in SYS-(K/n1,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 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.

A. Monotonicity Assumptions

We aim to check that the conjectures of this work 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 consider uniformly distribution jobs sizes and we represent the evolution of the degradation factor for different values of K (note that the x-axis is in the logarithmic scale) when γ varies from 1 to 10^{-4}. We observe that, in all the instances, the degradation factor decreases with γ, as stated in Conjecture 1. Furthermore, we see that, for $K = 100$ and $K = 1000$, when $\gamma \to 0$, the degradation factor tends to $4/3$, which is the upper-bound given in Proposition 4.

We also investigate the degradation factor for Bounded Pareto distributed job sizes with $\alpha \neq 1$. In Figure 4, we consider $\alpha = 1.5$ 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 2. In addition, we observe that $D_{BP(\alpha)}(K, K, \gamma)$ tends to K^2 when $\gamma \to 0$, which coincides with the value given in Lemma 11.

B. Degradation Factor

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 [16]. 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 [18]. 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 consider that l is multiple of n and we study the degradation factor for this case.

We consider that $K = 1000$ and $n = K$. In Figure 5, 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 8, Proposition 9 and Proposition 10 extend to systems with more than two servers.

3) 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 6, 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

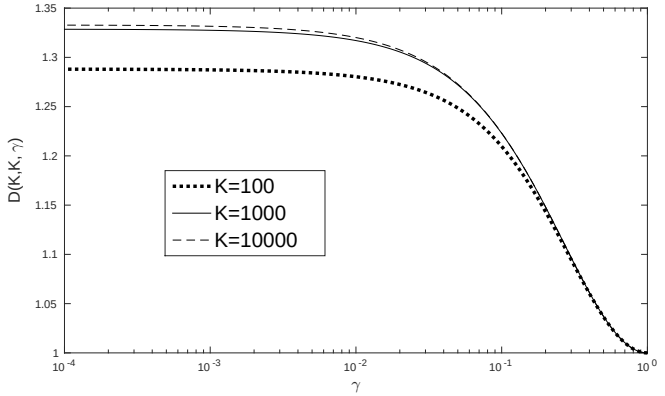

Fig. 3: Evolution over γ of the degradation factor for uniformly distributed job sizes (x-axis in logarithmic scale).

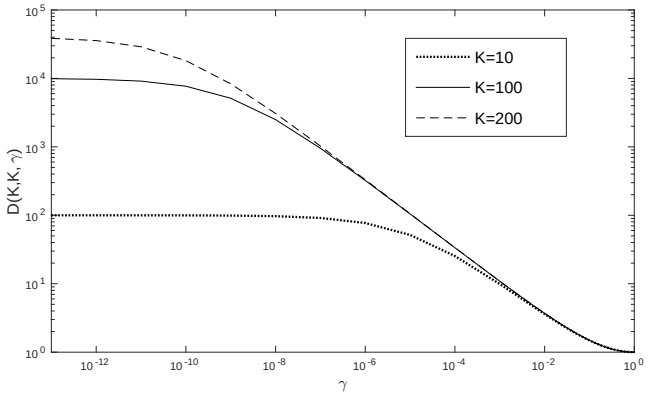

Fig. 4: Evolution over γ of the degradation factor for Bounded Pareto distributed job sizes with parameter $\alpha = 1.5$ (x-axis and y-axis in logarithmic scale).

	$n = 100$		$n = 500$		$n = 1000$	
	$D_{BP(\alpha)}(K,n,\gamma)$	$n^{\frac{1}{\lvert 1-\alpha\rvert}}$	$D_{BP(\alpha)}(K,n,\gamma)$	$n^{\frac{1}{\lvert 1-\alpha\rvert}}$	$D_{BP(\alpha)}(K,n,\gamma)$	$n^{\frac{1}{\lvert 1-\alpha\rvert}}$
$\alpha = 0.25$	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 $n^{\frac{1}{\lvert 1-\alpha\rvert}}$.

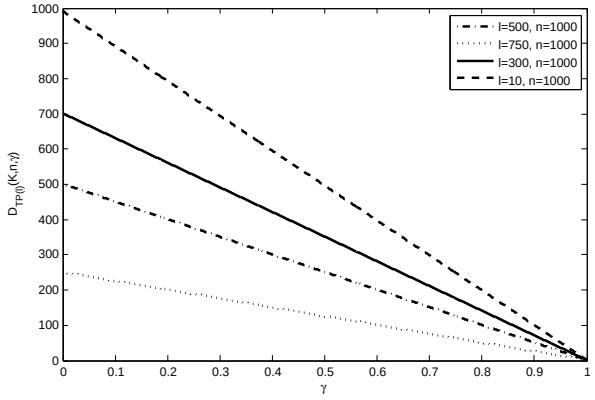

Fig. 5: Evolution over p of the degradation factor for two points distributed job sizes with $K > 2$ and l not multiple of n.

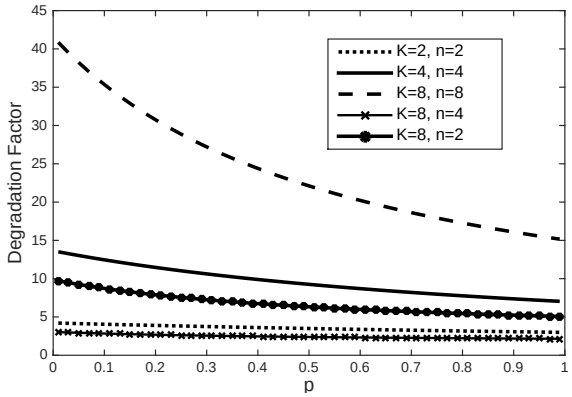

Fig. 6: Evolution over p of the degradation factor for Degenerated Hyperexponential distributed job sizes.

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. 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 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 [18], 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 cases where $\alpha \neq 1$.

Our objective is to know whether the optimal SITA degradation is high or not. We use the numerical results of [18], where they consider a two-server system and they obtain numerically the ratio of the performance of SITA-E policy

	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.

over the performance of the optimal SITA policy. In our simulations, 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 (13). 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 observed, the degradation factor has a symmetry since we have obtained the same results for α and $2-\alpha$. The reason of this is that the symmetry sends the cutoffs of the optimal SITA to the cutoffs of the optimal SITA of the dual distribution. We refer to [3] for full details on this symmetry property. Furthermore, we observe in Table III that 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 $2.2476 \cdot 10^4$.

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 $\mathbf{x} = (x_m, \ldots, x_M)$ and a probability distribution of these jobs sizes $\mathbf{p}$.

One might think that the degradation factor always increases with n. In Appendix K, 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 probalility 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 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 $c_0, c_1, \ldots, c_R, c_{R+1}$ 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.

[3]We show that there exist parameters of the system such that $D(K, n) \to \infty$.

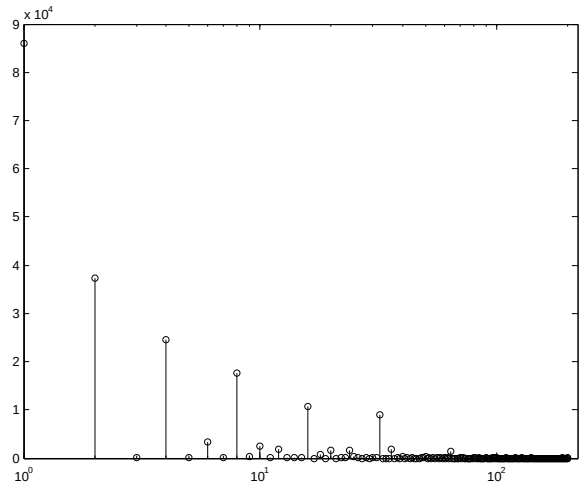

Fig. 7: HPC2N system. In the x-axis we represent the size of the submitted jobs (in bytes) and in the y-axis the number of jobs of each size.

$K = 10000$	Degradation Factor
$n = 1$	1
$n = 10$	1.0001
$n = 50$	1.0012
$n = 100$	1.0034
$n = 200$	1.0067
$n = 250$	1.0060
$n = 500$	1.0219
$n = 1000$	1.0476
$n = 10000$	4.0365

TABLE IV: Degradation factor of HPC2N system for $K = 10000$ when n varies from 1 to 10000.

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 7, 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 7. We consider that the number of servers is 10,000 and we present in Table IV 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 that that of $n = 250$, which confirms the observation of Appendix K. In the rest of the values of n, we see that there is a partial monotonicity on n for the

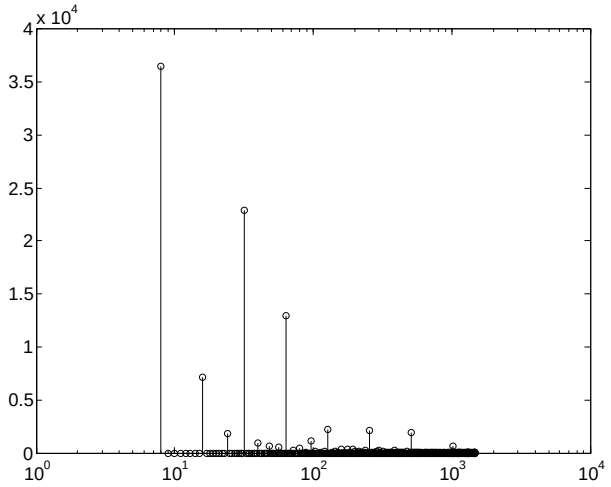

Fig. 8: SDSC Datastar. In the x-axis we represent the size of the submitted jobs (in bytes) and in the y-axis the number of jobs of each size.

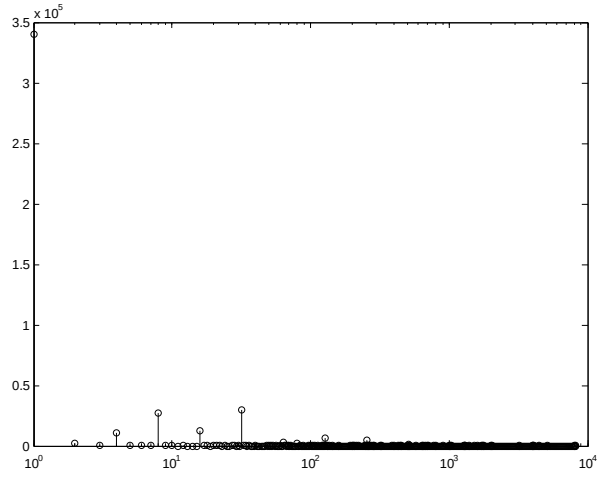

Fig. 9: RICC system. In the x-axis we represent the size of the submitted jobs (in bytes) and in the y-axis the number of jobs of each size.

$K = 10000$	Degradation Factor
$n = 1$	1
$n = 10$	1.0001
$n = 50$	1.0010
$n = 100$	1.0019
$n = 200$	1.0063
$n = 250$	1.0051
$n = 500$	1.0163
$n = 1000$	1.0658
$n = 10000$	5.7973

TABLE V: Degradation factor of SDSC Datastar system for $K = 10000$ when n varies from 1 to 10000.

$K = 10000$	Degradation Factor
$n = 1$	1
$n = 10$	1.0007
$n = 50$	1.0031
$n = 100$	1.0048
$n = 200$	1.0137
$n = 250$	1.0184
$n = 500$	1.0487
$n = 1000$	1.1238
$n = 10000$	7.5531

TABLE VI: Degradation factor of RICC system for $K = 10000$ when n varies from 1 to 10000.

degradation factor. Besides, as it can be observed in Table IV, 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.

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 8, 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 8. We consider that the number of servers is $K = 10000$ and the number of groups n varies from 1 to 10000. We present in Table V 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 of the largest of of 8192 bytes, therefore the difference on the size between the smallest and the largest is higher than in SDSC system. We plot in Figure 9 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 9. We consider that the number of servers is $K = 10000$ and also the values of n varying from 1 to 10000. In Table VI 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 multiservers systems where dispatchers operate under SITA-E policy, i.e., a size-based routing policy that equalizes the load of all the servers. We compare the performance of SYS-(K,n,λ), which is a system formed by K servers, $n > 1$ dispatcher and arrival rate λ with the performance of SYS-(K,1,λ). We first show that the performance of SYS-(K,n,λ) is the same as the performance of SYS-(K/n,1,λ/n). As a consequence, the performance analysis we perform in this article can be seen as the economies when, in a system with a single dispatcher, we scale up the number of servers and the arrival rate proportionally. We define the degradation factor as the ratio between the performance of SYS-(K,1,λ) and the performance of SYS-(K/n,1,λ/n). We define as γ the ratio between the size of the smallest and the largest job.

For an arbitrary distribution, we have established monotonicity on the number of dispatchers. From this result, we conclude that the degradation factor is lower-bounded by one and upper-bounded by the ratio between the performance of SYS-(K,1,λ) and of SYS-(1,1,λ/K). We investigate the maximum performance degradation for particular distributions. First, we consider that job sizes are uniformly distributed. For two servers, we show that the degradation factor is upper-bounded by 1.139. For an infinite number of servers, we show that the performance degradation for uniformly distributed job sizes is upper-bounded by $4/3$. For more than two and finite number of servers, assuming that the degradation factor decreases with γ, we show that the degradation is upper-bounded by $4/3$. We then consider that job sizes are Bounded Pareto distributed. We show that the degradation factor is unbounded from above for the following cases: (i) $\alpha = 0$ and (ii) $\alpha \neq$ and $K \to \infty$. For $\alpha \neq$ and finite K, assuming that the degradation factor decreases with γ, we show that the degradation factor is upper-bounded by $K^{\frac{1}{|1-\alpha|}}$. From this results, we conclude that the degradation is high when the variability of job increases. We also consider that job sizes are distributed in two points, since it is the distribution that maximizes the variance for a given mean. For this distribution, we show that, for two servers, the degradation factor is unbounded from above when load of both type of jobs is equal and when it is unequal.

As a future work, we think that an interesing extension of the degradation factor analysis performed in this work would be to consider other scheduling policies that are known to achieve an optimal performance in the literature such as Power of two, Join the Shortest Queue or the SITA policy with optimal thresholds. Another possible future research for this work 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 degration of multiservers systems considering a model with less restrictions, for instance, when the servers are hetereogenous 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. Equiload: 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. Tsafrir, 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/-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. Crovella, and C. Murta. Task assignment in a distributed system: Improving performance by load unbalancing. In *Proceedings of SIGMETRICS*, 1998.

[16] 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.

[17] 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.

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

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

[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. *IEEE/ACM Transactions on Networking*, 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. Advanced Topics in Cloud Computing.

[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] W. Whitt. Understanding the efficiency of multi-server service systems. *Management Science*, 38(5):708–723, 1992.

[29] W. Whitt. How multiserver queues scale with growing congestion-dependent demand. *Operations Research*, 51(4):531–542, 2003.

APPENDIX

A. Proof of Lemma 2

Let $H(x) = \int_{x_m}^{x} zf(z)dz$. Thus, we have that $H(x_m) = 0$ and $H(x_M) = \mathbb{E}(X)$. For SYS-(K,1,λ), it follows from (3) that

$$H(x_j) = \frac{\mathbb{E}(X) \cdot j}{K},\ j = 0, \ldots, K.$$

Similarly, for SYS-(K/n,1,λ/n), we have

$$H(y_j) = \frac{\mathbb{E}(X) \cdot j}{K/n},\ j = 0, \ldots, K/n.$$

From these results, we conclude that for $j = 0, \ldots, \frac{K}{n}$

$$H(y_j) = \frac{\mathbb{E}(X) \cdot j}{K/n} = \frac{\mathbb{E}(X) \cdot n \cdot j}{K} = H(x_{n \cdot j}). \tag{17}$$

Since the thresholds of SITA-E policy are unique if $f(x) > 0$ for all $x \in [x_m, x_M]$, then (17) implies that $y_j = x_{n \cdot j}$, for all $j = 0, \ldots, K/n$.

B. Proof of Lemma 3

First, (i) follows since $F(x_0) = 0$ and $F(x_K) = 1$. We now show (ii). Under the routing policy under consideration, the load of all the servers is the same, i.e., for $j = 2, \ldots, k$, we have that $p_{j-1}\mathbb{E}[X_{j-1}] = p_j\mathbb{E}[X_j]$. Now, since all the jobs executed in server j are smaller that the jobs executed in server $j+1$, it follows that the mean service time of server j is less that that of server $j+1$.

Finally, to prove (iii), we observe that for all j, $\int_{x_{n\cdot j}}^{x_{n\cdot(j+1)}} x^2 f(x)dx = \int_{x_{n\cdot(j-1)}}^{x_{n\cdot j}} x^2 f(x)dx$. Therefore

$$\begin{aligned} s_{j+1} &= \int_{x_{n\cdot j}}^{x_{n\cdot(j+1)}} x^2 f(x)dx \\ &\geq x_{n\cdot j} \int_{x_{n\cdot j}}^{x_{n\cdot(j+1)}} xf(x)dx \\ &= x_{n\cdot j} \int_{x_{n\cdot(j-1)}}^{x_{n\cdot j}} xf(x)dx \\ &\geq \int_{x_{n\cdot(j-1)}}^{x_{n\cdot j}} x^2 f(x)dx \\ &= s_j, \end{aligned}$$

where the inequalities are given since x^2 is an increasing function. And the desired result is proven.

C. Proof of Lemma 4

We first show that, for all $1 \leq i \leq k-1$,

$$\frac{p_i}{1 - \sum_{j=1}^{i-1} p_j} \geq \frac{1}{k-i+1}. \tag{18}$$

This is equivalent to show that

$$1 - \sum_{j=1}^{i} p_j \leq k - i + 1,$$

which holds since the LHS is less than one and the RHS is higher or equal than one $1 \leq i \leq k-1$. Furthermore, we show that

$$s_i \leq \frac{1}{k-i} \sum_{j=i+1}^{k} s_j,$$

since, from Lemma 3, $s_k \geq \cdots \geq s_1$ and therefore $(k-i)s_i \leq \sum_{j=i+1}^{k} s_j$ for all $i = 1, \ldots, k-1$. We now multiply (18) by $s_i - \frac{1}{k-i}\sum_{j=i+1}^{k} s_j$ and it results that

$$\begin{aligned} \frac{p_i}{1 - \sum_{j=1}^{i-1} p_j} s_i - \frac{p_i}{1 - \sum_{j=1}^{i-1} p_j} \frac{1}{k-i} \sum_{j=i+1}^{k} s_j \leq \\ \frac{1}{k-i+1} s_i - \frac{1}{k-i+1} \frac{1}{k-i} \sum_{j=i+1}^{k} s_j, \end{aligned}$$

where the sense of the sign changes since $s_i - \frac{1}{k-i}\sum_{j=i+1}^{k} s_j$ is negative. This expression is equivalent to the following one:

$$\begin{aligned} \frac{p_i}{1 - \sum_{j=1}^{i-1} p_j} s_i + \left(1 - \frac{p_i}{1 - \sum_{j=1}^{i-1} p_j}\right) \frac{1}{k-i} \sum_{j=i+1}^{k} s_j \leq \\ \frac{1}{k-i+1} s_i + \left(1 - \frac{1}{k-i+1}\right) \frac{1}{k-i} \sum_{j=i+1}^{k} s_j, \end{aligned}$$

which after small manipulations we obtain:

$$\frac{1 - \sum_{j=1}^{i-1} p_j}{k-i+1} \sum_{j=i}^{k} s_j \geq p_i s_i + \frac{1 - \sum_{j=1}^{i} p_j}{k-i} \sum_{i=i+1}^{k} s_j, \tag{19}$$

for $i = 1, \ldots, k-1$. Hence, for i=1, we have that

$$\frac{1}{k} \sum_{j=1}^{k} s_j \geq p_1 s_1 + \frac{1-p_1}{k-1} \sum_{i=2}^{k} s_j.$$

We now use that (19) for $i = 2$ to see that

$$p_1 s_1 + \frac{1-p_1}{k-1} \sum_{i=2}^{k} s_j \geq p_1 s_1 + p_2 s_2 + \frac{1 - \sum_{j=1}^{2} p_j}{k-2} \sum_{i=3}^{k} s_j.$$

Again, we apply (19) for $i = 3$ and we get that

$$\begin{aligned} p_1 s_1 + p_2 s_2 + \frac{1 - \sum_{j=1}^{2} p_j}{k-2} \sum_{i=3}^{k} s_j \geq \\ p_1 s_1 + p_2 s_2 + p_3 s_3 + \frac{1 - \sum_{j=1}^{3} p_j}{k-3} \sum_{i=4}^{k} s_j. \end{aligned}$$

We continue with the same reasoning until $i = k-1$ and we derive that

$$\frac{1}{k} \sum_{j=1}^{k} s_j \geq \sum_{j=1}^{k-1} p_j s_j + (1 - \sum_{j=1}^{k-1} p_j) \cdot s_k.$$

And the desired result follows since $p_k = 1 - \sum_{j=1}^{k-1} p_j$.

D. Proof of Lemma 5

First, we substitute the values of x_j, $f(x)$ and $F(x)$ in (8) and thus

$$D_U(K,n,x_m,x_M) = \frac{1}{n}\frac{\sum_{j=1}^{K/n}(x_{n\cdot j}-x_{n\cdot(j-1)})(x_{n\cdot j}^3-x_{n\cdot(j-1)}^3)}{\sum_{j=1}^{K}(x_j-x_{j-1})(x_j^3-x_{j-1}^3)}.$$

Using that $x_j = x_M\, z_j$, (10) follows by dividing the numerator and denominator by x_M^4 and, since z_j depends on x_m and x_M only through γ, the desired result follows.

E. Proof of Lemma 6

We aim to prove that $D(2,2,\gamma)$ is decreasing with γ. Before presenting this result, we give the following lemma.

Lemma 13. *For all $\gamma < 1$,*

(a) $\frac{1-z_1}{1-\gamma} \leq \frac{1-z_1^3}{1-\gamma^3}$.
(b) $\frac{d}{d\gamma}\left(\frac{1-z_1^3}{1-\gamma^3}\right) < 0$.
(c) $\frac{d}{d\gamma}\left(\frac{1-z_1}{1-\gamma}\right) + \frac{d}{d\gamma}\left(\frac{1-z_1^3}{1-\gamma^3}\right) > 0$.

Proof. (a) We observe that

$$\frac{1-z_1^3}{1-\gamma^3} = \frac{1-z_1}{1-\gamma}\cdot\left(1+\frac{z_1^2-\gamma^2+z_1-\gamma}{\gamma^2+\gamma+1}\right).$$

Since $0 \leq \gamma \leq z_1$, we conclude that $\frac{z_1^2-\gamma^2+z_1-\gamma}{\gamma^2+\gamma+1}$ is positive, which gives the desired result.

(b) We have that

$$\frac{d}{d\gamma}\left(\frac{1-z_1^3}{1-\gamma^3}\right) < 0 \iff 3\gamma^2(1-z_1^3) - 3z_1^2\frac{d}{d\gamma}z_1(1-\gamma^3) < 0.$$

Using that $z_1 = \sqrt{\frac{1+\gamma^2}{2}}$ and that $\frac{d}{d\gamma}z_1 = \frac{1}{\sqrt{2}}\frac{\gamma}{\sqrt{1+\gamma^2}}$ and simplifying, it results

$$2\gamma\sqrt{2} < (1+\gamma)\sqrt{1+\gamma^2}.$$

Squaring both sides of the inequality and simplifying, we obtain that

$$(1-\gamma)^4 + 6\gamma\,(1-\gamma)^2 > 0,$$

which always holds when $\gamma < 1$. Therefore, the desired result is proven.

(c) Since $\frac{dz_1}{d\gamma} = \frac{\gamma}{2z_1}$ and $\frac{1-z_1^3}{1-\gamma^3} = \frac{1-z_1}{1-\gamma}\cdot\frac{z_1^2+z_1+1}{\gamma^2+\gamma+1}$, we derive that

$$\frac{d}{d\gamma}\left(\frac{1-z_1}{1-\gamma}\right) + \frac{d}{d\gamma}\left(\frac{1-z_1^3}{1-\gamma^3}\right) > 0 \iff$$

$$2z_1(\gamma^4+2\gamma^3+6\gamma^2+2\gamma+1) > (\gamma+1)(\gamma^4+\frac{7}{2}\gamma^3+3\gamma^2+\frac{7}{2}\gamma+1).$$

Squaring both sides of this inequality and simplifying, we obtain that

$$(\gamma-1)^2(4\gamma^6+12\gamma^5+27\gamma^4+22\gamma^3+27\gamma^2+\gamma+4) > 0,$$

which is always positive when $\gamma \neq 1$.

□

We now give the proof of Lemma 6

Proof. We observe that the derivative of $D(2,2,\gamma)$ is negative if and only if the derivative of

$$\frac{(1-z_1^3)(1-z_1)}{(1-\gamma^3)(1-\gamma)} + \frac{(z_1^3-\gamma^3)(z_1-\gamma)}{(1-\gamma^3)(1-\gamma)}$$

is positive. We now notice that this expression can be written as follows:

$$\frac{(1-z_1^3)(1-x)}{(1-\gamma^3)(1-\gamma)} + \left(1-\frac{(1-z_1^3)}{(1-\gamma^3)}\right)\left(1-\frac{(1-z_1)}{(1-\gamma)}\right). \quad (20)$$

Hence, we aim to show then that the derivative of (20) is positive. We write $h = \frac{1-z_1}{1-\gamma}$ and $g = \frac{1-z_1^3}{1-\gamma^3}$. Using this notation, the derivative of (20) is positive if and only if

$$h'(2g-1) + g'(2h-1) > 0,$$

where h' and g' denote the derivate with respect to γ. From Lemma 13(c), we have that $h' > -g'$ if $\gamma < 1$ and we thus derive that $h'(2g-1)+g'(2h-1) > -g'(2g-1)+g'(2h-1) = 2g'(h-g)$.

Using Lemma 13(b) we have that g' is negative for all $\gamma < 1$ and from Lemma 13(a) it follows that $g \geq h$. Thus, we have that $g'(h-g)$ is positive, which implies that the derivative of $D(2,2,\gamma)$ with respect to γ is negative, and the result follows. □

F. Proof of Lemma 7

The derivative of $D_U(\infty,\infty,\gamma)$ with respect to γ is negative if and only if

$$(-(1-\gamma^3)-3\gamma^2(1-\gamma))(1-\gamma^2)^2+2\gamma(1-\gamma^2)(1-\gamma)(1-\gamma^3) < 0.$$

After small manipulations of the previous expression, we obtain the following one:

$$(1-\gamma)(1-\gamma^2)(-1+\gamma-3\gamma^2+\gamma^3+2\gamma^4) < 0.$$

And the desired result follows since $-1+\gamma-3\gamma^2+\gamma^3+2\gamma^4$ is negative for all $\gamma \in (0,1)$.

G. Proof of Lemma 9

Prior to present the proof of Lemma 9, we give the following result.

Lemma 14. *The function $\frac{n\,(1+\gamma^{\frac{n}{K}})}{(1-\gamma^{\frac{n}{K}})}$ is an increasing function of n.*

Proof. The derivative of $\frac{n\,(1+\gamma^{\frac{n}{K}})}{(1-\gamma^{\frac{n}{K}})}$ with respect to n is positive if and only if

$$\left(1+\gamma^{\frac{n}{K}}+n\,\frac{n}{K}\,\gamma^{\frac{n}{K}}\,\log\gamma\right)(1-\gamma^{\frac{n}{K}}) + n\,(1+\gamma^{\frac{n}{K}})\,\frac{n}{K}\,\gamma^{\frac{n}{K}}\,\log\gamma > 0.$$

After simplification, we obtain that

$$1-\gamma^{\frac{2n}{K}} + 2\frac{n}{K}\gamma^{\frac{n}{K}}\log\gamma > 0.$$

Let $L(\gamma,K,n) = 1-\gamma^{\frac{2n}{K}} + 2\frac{n}{K}\gamma^{\frac{n}{K}}\log\gamma$. We now observe that $\lim_{\gamma\to 1} L(\gamma,K,n) = 0$. Therefore, since $\gamma \in [0,1]$, to

show the desired result, it is enough to prove that $L(\gamma, K, n)$ is decreasing with γ. Thus,

$$\frac{d}{d\gamma}L(\gamma,K,n) < 0 \iff \frac{2n^2}{K^2}\frac{\gamma^{\frac{2n}{K}}}{\gamma} > 0.$$

And we observe that $\frac{2n^2}{K^2}\frac{\gamma^{\frac{2n}{K}}}{\gamma}$ is always positive and the proof is completed. □

We are now showing Lemma 9.

Proof. The derivative of $D_{BP(1)}(K,n,\gamma)$ with respect to γ is negative if and only if:

$$-n(1-\gamma^{\frac{2n}{K}})(1-\gamma^{\frac{1}{K}})^2 + (1-\gamma^{\frac{2}{K}})(1-\gamma^{\frac{n}{K}})^2 < 0.$$

Dividing this expression by $(1-\gamma^{\frac{n}{K}})(1-\gamma^{\frac{1}{K}})$ and rearranging both sides of the expression, we obtain that

$$\frac{n\ (1-\gamma^{\frac{2n}{K}})}{(1-\gamma^{\frac{n}{K}})^2} > \frac{(1-\gamma^{\frac{2}{K}})}{(1-\gamma^{\frac{1}{K}})^2}.$$

Lemma 14 says that $\frac{n\ (1+\gamma^{\frac{n}{K}})}{(1-\gamma^{\frac{n}{K}})}$ is increasing with n and this means that the previous inequality is always true. Hence, the desired result follows. □

H. Proof of Lemma 10

We aim to prove that (15) decreases with γ. Prior to that, we give the following result:

Lemma 15. *The equation* $\frac{1-x}{(1-x^{1-\alpha})^2 x^{\alpha/2}}$ *is*

- *decreasing with* x *if* $\alpha < 1$,
- *increasing with* x *if* $\alpha > 1$.

Proof. The sign of the derivative of $\frac{1-x}{(1-x)^2x^{\alpha/2}}$ is the same as the sign of

$$x^{\frac{\alpha}{2}}\left(-(1-x^{1-\alpha}) - (1-x)\left(\frac{\alpha}{2}\frac{1}{x}(1-x^{1-\alpha}) - (1-\alpha)x^{-\alpha}\right)\right).$$

The sign of the above expression coincides with that of

$$-(1-x^{1-\alpha}) - (1-x)\left(\frac{\alpha}{2}\frac{1}{x}(1-x^{1-\alpha}) - (1-\alpha)x^{-\alpha}\right),$$

which after small manipulations, gives

$$\frac{\alpha}{2}(1-x^{-\alpha}+x^{1-\alpha}) - 1 + x^{-\alpha} - \frac{\alpha}{2}\frac{1}{x}.$$

This expression tends to zero when $x \to 1$, therefore it is positive (resp. negative) if it is decreasing with x (resp. increasing with x). Therefore, we study the sign of its derivative. Hence, the derivative of the above expression is

$$(\frac{\alpha^2}{2}-1)x^{-1-\alpha} + \frac{\alpha}{2}(1-\alpha)x^{-\alpha} + \frac{1}{2x^2},$$

whose sign coincides with that of

$$(\frac{\alpha^2}{2}-1) + \frac{1}{2}(1-\alpha)x + \frac{x^{\alpha-1}}{2}.$$

This expression equals zero when $x = 0$. Hence, its sign is the oposite of that of its derivative:

$$\frac{1}{2}(1-\alpha)(1-x^{\alpha-2}),$$

which is negative if $\alpha < 1$ and positive if $\alpha > 1$ since $(1-x^{\alpha-2} > 0$. Therefore, we have shown that the equation decreases with x if $\alpha < 1$ and increases with x if $\alpha > 1$. □

We now prove the result of Lemma 10.

Proof. We first observe that $D_{BP(\alpha)}(\infty,\infty,\gamma)$ can be written in the following way:

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

where

$$B(\alpha,\gamma) = \frac{1-\gamma}{\gamma^{\alpha/2}(1-\gamma^{1-\alpha})}.$$

Now, we note that the sign of $B(\alpha,\gamma)$ is given by the sign of $1-\gamma^{1-\alpha}$, which is positive if $\alpha > 1$ and negative if $\alpha < 1$. Finally, from the result of Lemma 15, it follows that the derivative of $D_{BP(\alpha)}(\infty,\infty,\gamma)$ is always negative. □

I. Proof of Lemma 11

First, we divide the numerator and denominator of (13) by $\gamma^{-\alpha}-1$ and we write the degradation factor as follows: $D_{BP(\alpha)}(K,n,\gamma) = \frac{1}{n}\frac{\sum_{j=1}^{K/n} m_j}{\sum_{j=1}^{K} l_j}$, where

$$m_j = \frac{(z_{nj}^{2-\alpha} - z_{n(j-1)}^{2-\alpha})(z_{n(j-1)}^{-\alpha} - z_{nj}^{-\alpha})}{\gamma^{-\alpha}-1},$$

for $j = 1,\dots,K/n$ and for $j = 1,\dots,K$

$$l_j = \frac{(z_j^{2-\alpha} - z_{j-1}^{2-\alpha})(z_{j-1}^{-\alpha} - z_j^{-\alpha})}{\gamma^{-\alpha}-1}.$$

We now summarize how we prove the desired result. If $0 < \alpha < 1$, we show that, when $\gamma \to 0$, then $l_K = (1/K)^{\frac{2-\alpha}{1-\alpha}}$ and $m_{K/n} = (n/K)^{\frac{2-\alpha}{1-\alpha}}$, while the rest of the terms are zero, that is, $l_j = 0$, for all $j = 1,\dots,K-1$, and $m_j = 0$, for all $j = 1,\dots,\frac{K}{n}-1$. Thus,

$$\lim_{\gamma\to 0} D_{BP(\alpha)}(K,n,\gamma) = \lim_{\gamma\to 0}\frac{1}{n}\cdot\frac{m_{\frac{K}{n}}}{l_K} = \frac{1}{n}\cdot\frac{\left(\frac{n}{K}\right)^{\frac{2-\alpha}{1-\alpha}}}{\left(\frac{1}{K}\right)^{\frac{2-\alpha}{1-\alpha}}} = n^{\frac{1}{1-\alpha}}.$$

On the other hand, if ,$1 < \alpha < 2$, we show that, when $\gamma \to 0$, then $l_1 = (1/K)^{\frac{-\alpha}{1-\alpha}}$ and $m_1 = (n/K)^{\frac{-\alpha}{1-\alpha}}$, while the rest of the terms are zero, i.e., $l_j = 0$, for all $j = 2,\dots,K$, and $m_j = 0$, for all $j = 2,\dots,K/n$. Hence,

$$\lim_{\gamma\to 0} D_{BP(\alpha)}(K,n,\gamma) = \frac{1}{n}\cdot\frac{m_1}{l_1} = \frac{1}{n}\cdot\frac{\left(\frac{n}{K}\right)^{\frac{-\alpha}{1-\alpha}}}{\left(\frac{1}{K}\right)^{\frac{-\alpha}{1-\alpha}}} = \frac{1}{n}\cdot n^{\frac{-\alpha}{1-\alpha}} = n^{\frac{-1}{1-\alpha}}.$$

- **The case** $0 < \alpha < 1$**:** The denominator of l_j tends to infinity when $\gamma \to 0$ for all $j = 1,\dots,K$. Furthermore, for all $j = 1,\dots,K-1$, the numerator of l_j when $\gamma \to 0$, tends to a finite value. Therefore, $\lim_{\gamma\to 0} l_j = 0$, for $j = 1,\dots,K-1$.
  Besides, the denominator of m_j tends to infinity when $\gamma \to 0$ for all $j = 1,\dots,K/n$. And, for all $j = 1,\dots,\frac{K}{n}-1$, the numerator of m_j when $\gamma \to 0$, tends to a finite value. Therefore, $\lim_{\gamma\to 0} m_j = 0$, for $j = 1,\dots,\frac{K}{n}-1$.

We now observe that $z_K = \gamma$ and $z_{K-1}^{-\alpha}$ tend to a finite value, when $\gamma \to 0$. Thus,

$$\lim_{\gamma\to 0} \frac{z_K^{-\alpha} - z_{K-1}^{-\alpha}}{\gamma^{-\alpha} - 1} = \lim_{\gamma\to 0} \frac{\gamma^{-\alpha} - z_{K-1}^{-\alpha}}{\gamma^{-\alpha} - 1} = 1.$$

We now observe that,

$$\lim_{\gamma\to 0} z_{K-1}^{2-\alpha} - z_K^{2-\alpha} = (1/K)^{\frac{2-\alpha}{1-\alpha}}.$$

As a result,

$$\lim_{\gamma\to 0} l_K = \left(\frac{1}{K}\right)^{\frac{2-\alpha}{1-\alpha}}.$$

For $m_{\frac{K}{n}}$, since $z_{K-n}^{-\alpha}$ is finite when $\gamma \to 0$, we have that

$$\lim_{\gamma\to 0} \frac{z_K^{-\alpha} - z_{K-n}^{-\alpha}}{\gamma^{-\alpha} - 1} = \lim_{\gamma\to 0} \frac{\gamma^{-\alpha} - z_{K-n}^{-\alpha}}{\gamma^{-\alpha} - 1} = 1,$$

and also that, $\lim_{\gamma\to 0} z_{K-n}^{2-\alpha} - z_K^{2-\alpha} = (\frac{n}{K})^{\frac{2-\alpha}{1-\alpha}}$. Therefore, $\lim_{\gamma\to 0} m_{\frac{K}{n}} = \left(\frac{n}{K}\right)^{\frac{2-\alpha}{1-\alpha}}$.

- **The case** $1 < \alpha < 2$**:** Using that $\frac{2-\alpha}{1-\alpha} < 0$ if $1 < \alpha < 2$ and that $\gamma^{1-\alpha} \to \infty$ when $\gamma \to 0$, it follows that

$$\lim_{\gamma\to 0} z_j^{2-\alpha} \to 0,$$

for all $j = 1, \ldots, K$. We recall that $z_0 = 1$. Moreover,

$$\frac{z_{j-1}^{-\alpha} - z_j^{-\alpha}}{\gamma^{1-\alpha} - 1}$$

tends to a finite value when $\gamma \to 0$. Hence, for all $j = 2, \ldots, K$, $\lim_{\gamma\to 0} l_j$ is equal to a finite value times zero, i.e., it is zero.
We observe that $\frac{z_{j-n}^{-\alpha} - z_j^{-\alpha}}{\gamma^{1-\alpha}-1}$ also tends to a finite value for all $j = 2, \ldots, K/n$. Therefore, for all $j = 2, \ldots, K/n$, $\lim_{\gamma\to 0} m_j$ is equal to a finite value times zero which means that it tends to zero.
We now focus on l_1. We first give the value of $z_1^{-\alpha} - z_0^{-\alpha}$ when $\gamma \to 0$:

$$\lim_{\gamma\to 0} \frac{\left(\left(\frac{n}{K}\ \gamma^{1-\alpha} + \frac{K-n}{K}\right)^{\frac{-\alpha}{1-\alpha}} - 1\right)}{\gamma^{-\alpha} - 1} = \left(\frac{n}{K}\right)^{\frac{-\alpha}{1-\alpha}},$$

which holds since $\gamma^{-\alpha} \to \infty$ when $\gamma \to 0$ and $\frac{-\alpha}{1-\alpha} > 0$. Using that $z_0^{2-\alpha} - z_1^{2-\alpha}$ tends to 1 when $\gamma \to 0$, we conclude that

$$\lim_{\gamma\to 0} m_K = \left(\frac{n}{K}\right)^{\frac{-\alpha}{1-\alpha}}.$$

We now concentrate on m_1 and, using the same reasoning as before, we have that

$$\lim_{\gamma\to 0} \frac{\left(\left(\frac{1}{K}\ \gamma^{1-\alpha} + \frac{K-1}{K}\right)^{\frac{-\alpha}{1-\alpha}} - 1\right)}{\gamma^{-\alpha} - 1} = \left(\frac{1}{K}\right)^{\frac{-\alpha}{1-\alpha}}.$$

We also observe that $z_0^{2-\alpha} - z_n^{2-\alpha}$ tends to 1 when $\gamma \to 0$ and, thus,

$$\lim_{\gamma\to 0} l_K = \left(\frac{1}{K}\right)^{\frac{-\alpha}{1-\alpha}}.$$

J. Proof of Lemma 12

We first substitute $p_1 = \frac{p\gamma+(1-p)}{2p\gamma}$ in (16) and simplifying we obtain the following equivalent expression:

$$\frac{\gamma(p\gamma^2 + (1-p))}{\gamma(p\gamma^2 + (1-p)) - (1-p)^2(1-\gamma)^2}.$$

The first derivative of this expression with respect to γ is negative if and only if

$$(3p\gamma^2 + (1-p))(\gamma(p\gamma^2 + (1-p)) - (1-p)^2(1-\gamma)^2) - (3p\gamma^2 + (1-p) + 2(1-\gamma)(1-p)^2)(\gamma(p\gamma^2 + (1-p))) < 0.$$

Simplifying this expression, we obtain the following equivalent one:

$$-(1-p)^2(1-\gamma)(p\gamma^2(3-\gamma) + (1-p)(1-\gamma)) < 0,$$

which is always negative for all $\gamma < 1$ and the desired result follows.

K. Monotonicity counterexample

Consider the following discrete job size distribution: $\mathbf{x} = [1,\ 10,\ 100]$ and $\mathbf{p} = [0.1,\ 0.1,\ 0.8]$ and also that $K = 10000$, $n_1 = 100$ and $n_2 = 125$. We aim to show that $D(K, n_1, \mathbf{x}, \mathbf{p}) > D(K, n_2, \mathbf{x}, \mathbf{p})$. From (8), it follows that the denominator of $D(K, n_1, \mathbf{x}, \mathbf{p})$ and of $D(K, n_2, \mathbf{x}, \mathbf{p})$ are the same and, therefore, it is enough check that

$$\frac{1}{n_1} \sum_{j=1}^{K/n_1} \mathbb{E}(X_j) \sum_{i=1}^{3} q_{ij}^{n_1} p_i > \frac{1}{n_1} \sum_{j=1}^{K/n_2} \mathbb{E}(X_j) \sum_{i=1}^{3} q_{ij}^{n_2} p_i \quad (21)$$

where $q_{ij}^{n_1}$ (resp. $q_{ij}^{n_2}$) is the proportion of jobs of size x_i, $i = 1, 2, 3$, to be executed in server j when the system is formed by n_1 servers (resp. n_2 servers) and p_i is the i-th component of vector $\mathbf{p}$. We show the desired result below:

- The LHS of (21) can be seen as the performance of a system with 100 servers and a single dispatcher. For this system, we have that: (i) one server executes all the smallest jobs and a proportion of 0.711 of the medium size jobs, (ii) another server executes a proportion of 0.289 of the medium size jobs and a proportion of 0.006525 of the largest jobs, and (iii) all the remaining 98 servers execute only the largest jobs. Therefore, it results that

$$\begin{aligned}(1/100) * ((1 - 0.006525)^2 * 0.8^2 * 100^2/98 \\ +(0.289 * 0.1 + 0.006525 * 0.8)* \\ (0.289 * 0.1 * 10^2 + 0.006525 * 0.8 * 100^2) \\ +(0.1 + (1 - 0.289) * 0.1)* \\ (0.1 * 1^2 + (1 - 0.289) * 0.1 * 10^2)),\end{aligned}$$

  which equals 0.6757.
- The RHS of (21) can be seen as the performance of a system with 80 servers and a single dispatcher. For this system, we have that: (i) one server executes all the smallest jobs and a proportion of 0.9137 of the medium size jobs, (ii) another server executes a proportion of 0.0863 of the medium size jobs and a proportion of

0.0116 of the largest jobs, and (iii) all the remaining 78 servers execute only the largest jobs. Therefore, it results that

$$\begin{aligned}(1/125) * ((1-0.0116)^2 * 0.8^2 * 100^2/78 \\ +(0.0863 * 0.1 + 0.0116 * 0.8)* \\ (0.0863 * 0.1 * 10^2 + 0.0116 * 0.8 * 100^2) \\ +(0.1 + (1-0.0863) * 0.1)* \\ (0.1 * 1^2 + (1-0.0863) * 0.1 * 10^2)),\end{aligned}$$

which equals 0.6688.

Hence, we have seen that the degradation factor does not always increase with n for discrete job size distributions.