

Achievable Stability in Redundancy Systems

Youri Raaijmakers^{a,*}, Sem Borst^a

^a*Department of Mathematics and Computer Science, Eindhoven University of Technology, 5600 MB Eindhoven, The Netherlands*

Abstract

We consider a system with N parallel servers where incoming jobs are immediately replicated to, say, d servers. Each of the N servers has its own queue and follows a FCFS discipline. As soon as the first job replica is completed, the remaining replicas are abandoned. We investigate the achievable stability region for a quite general workload model with different job types and heterogeneous servers, reflecting job-server affinity relations which may arise from data locality issues and soft compatibility constraints. Under the assumption that job types are known beforehand we show for New-Better-than-Used (NBU) distributed speed variations that no replication ($d = 1$) gives a strictly larger stability region than replication ($d > 1$). Strikingly, this does not depend on the underlying distribution of the intrinsic job sizes, but observing the job types is essential for this statement to hold. In case of non-observable job types we show that for New-Worse-than-Used (NWU) distributed speed variations full replication ($d = N$) gives a larger stability region than no replication ($d = 1$).

Keywords: Parallel-server system, redundancy, stability

1. Introduction

Redundancy scheduling has attracted strong interest as a mechanism to improve the delay performance in parallel-server systems. In redundancy scheduling an incoming job is replicated and dispatched to d different servers and as soon as the first of the d replicas finishes service the remaining replicas are abandoned ('cancel-on-completion' c.o.c.). Adding replicas increases the chance for one of the replicas to find a short queue, thus reducing the latency. On the other hand, adding replicas could cause instability since the same job may be in service at multiple servers, potentially wasting service capacity. Establishing the stability condition is not straightforward since the various replicas may have started service at different times. Among the numerous studies on redundancy scheduling, stability results have remained scarce so far.

Gardner et al. [5] introduce the redundancy- d system and obtain analytical expressions for the expected number of jobs in the system in the scenario with uniform selection of the servers, exponential job sizes, i.i.d. replicas and homogeneous servers, i.e., the server speeds of all servers are equal. From the expressions it is concluded that in this scenario more redundancy is always better for the expected latency. In terms of stability their main result is that the stability condition

*Corresponding author

Email address: y.raaijmakers@tue.nl (Youri Raaijmakers)

Preprint submitted to SIGMETRICS 2021

August 11, 2020

of the c.o.c. version of redundancy scheduling is $\lambda/N\mu < 1$, where λ denotes the arrival rate and the job sizes are exponentially distributed with parameter μ . Note that this stability condition is independent of the number of replicas.

In [15] it is shown that in the same scenario with scaled Bernoulli job sizes the stability condition is asymptotically given by $\lambda/K^{d-1} < 1$ as $K \rightarrow \infty$. Here the job size is either 0 or K with probability $1 - 1/K$ and $1/K$, respectively. Observe that this stability condition is asymptotically independent of the number of servers.

The contrasting results in [5] and [15] indicate that the stability condition is highly sensitive to the job size distribution, and for general job size distributions the stability condition remains unknown. For a discrete-time system with Bernoulli arrivals a lower bound is proved in [12]. While the bound is not always tight, it provides a first result for the necessary stability condition that depends on the number of servers, the number of replicas and the joint distribution of the job sizes.

Anton et al. [1] investigate the stability condition in the scenario of homogeneous servers and exponential job sizes for different service disciplines at each individual server, such as processor sharing, FCFS and random order of service. For FCFS and identical replicas they prove an implicit stability condition. Namely, the system is stable if $\lambda/\bar{l}\mu < 1$ and unstable if $\lambda/\bar{l}\mu > 1$, where \bar{l} is the long-run average number of jobs served in the saturated system, i.e., the system with an infinite backlog of jobs. Finding a closed-form expression for \bar{l} remains an open problem. They also explore the stability condition for heterogeneous server speeds by simulation, showing that heterogeneity in server speed has a profound impact on the stability condition.

Gardner et al. [4] study the same scenario in the S&X model, where the server speeds (slow-down factors) at the various servers are independent and identically distributed. No analytical expression is obtained for either the expected latency or the stability condition. However, simulation shows that for more variable job size distributions, more redundancy at first decreases the expected latency, but then hurts badly. The system can even become unstable if the number of replicas d is too high. A dispatching policy 'Redundancy-to-Idle-Queue' (RIQ) that only replicates the job to idle servers is introduced to overcome this problem. Highly accurate approximations for both the expected latency and the transform of the latency are derived. It is proved that, in contrast to redundancy- d scheduling, the RIQ policy cannot become unstable as the number of replicas increases. Stability aspects of redundancy scheduling in a many-server regime are discussed in [7, 8]. For a recent summary of exact stability condition results we refer to [16, Table 1].

Further work has focused on comparing the stability conditions and showing that either no replication ($d = 1$) or full replication ($d = N$) is optimal in the scenario of i.i.d. replicas and homogeneous server speeds. In [11] it is proved that full replication stochastically maximizes the number of jobs completed jointly across time for NWU job size distributions. No replication is shown to be optimal for two servers and NBU job size distributions, see Definition 1 below for the definition of NBU and NWU distributions. In [10] these results are generalized and it is proved, by a combinatorial argument, that no replication and full replication give the largest stability region for NBU and NWU job sizes, respectively. In [9] these results are extended to log-concave and respectively, log-convex complementary cumulative distribution functions. Note that log-concavity and log-convexity imply NBU and NWU, respectively, but the converse is not true.

In [20] the single fork-join policy is analyzed. This policy launches n tasks and waits until $(1 - p)n$ tasks are finished. For the remaining pn straggling tasks there are two options: either replicate and keep the original task or replicate and kill the original task. Under the assumption

that there is no queueing of the tasks it is proved that for NBU distributions keeping the original task gives lower latency while for NWU distributions killing the original task gives lower latency. The effect of replication in the fork-join model is also analyzed in [13]. Different strategies, such as no replication, full replication or partial replication, are shown to perform better depending on the job size distribution. In [19] a scheduling policy, called fewest unassigned tasks first with low-priority replication, is proposed in case of an NBU distribution, while the earliest due date first with replication policy is proposed for an NWU distribution.

In this paper we investigate the achievable stability region for c.o.c. redundancy systems in a quite general workload model, as considered earlier in [14], with multiple job types and servers that follow a FCFS discipline. Replicas may be assigned to the servers according to static type-dependent probabilities, instead of uniformly at random. Additionally, we deal with the complex dynamics arising from potentially different start times as a result of queueing which may occur when servers are not partitioned in disjoint pools of d servers. Specifically, we allow for generally distributed job sizes and the server speeds (slowdown factors) for a given job type are allowed to be inter-dependent and non-identically distributed, reflecting job-server affinity relations which may arise from data locality issues and soft compatibility constraints that are increasingly prevalent in data center environments. This workload model also subsumes the S&X model introduced in [4].

The general nature of the workload model reveals that the optimal degree of replication is not determined by the distribution of the intrinsic job sizes, but rather by the random variation in service speeds (or slow down factors) for a given job across the various servers. Also, our set-up with different job types and heterogeneous servers separates purely random variation in speeds across servers from systematic differences induced by job-server affinity relations. In particular, our results are the first to demonstrate that when job types are not explicitly observable, this uncertainty plays a similar role as purely random variation in speeds, and creates a potentially strong incentive for replication, even when the speeds for a job of a given type show little or no variation at all. Conversely, if there is little or no random variation in speeds, and the variability primarily arises from fundamental heterogeneity in job characteristics that can be observed beforehand, then replication provides no gains from a stability perspective.

The remainder of the paper is organized as follows. In Section 2 we present a detailed model description and some preliminary results. In Sections 3 and 4 we state and prove the main theorems for NBU and NWU distributed speed variations, respectively. Section 5 contains conclusions and some suggestions for further research.

2. Model description and preliminary results

Consider a system with N parallel servers where jobs arrive as a Poisson process of rate λ . Each of the N servers has its own queue and follows a FCFS discipline. When a job arrives, multiple replicas may be assigned to one or more servers according to static type-dependent probabilities. A special case of such a static probabilistic assignment is the celebrated *power-of- d* policy, where replicas are assigned to d servers selected uniformly at random (without replacement), which is the prevalent case considered in the literature.

In case multiple replicas are assigned, the service speeds R_1, \dots, R_N for that job on the various servers may differ. We allow the service speeds R_1, \dots, R_N of a generic job to be governed by some joint distribution $F(r_1, \dots, r_N)$, reflecting possible server heterogeneity and job-server compatibility relations. For convenience, we consider the case where the joint distribution

$F(r_1, \dots, r_N)$ is discrete, and has mass in a finite number of say M points (r_{1j}, \dots, r_{Nj}) with corresponding probabilities p_j , $j = 1, \dots, M$. This system may equivalently be thought of as having M job types, where r_{ij} is the service speed of type- j jobs at the i -th server. For notational convenience let $\mathcal{N} = \{1, 2, \dots, N\}$ denote the set of servers and $\mathcal{M} = \{1, 2, \dots, M\}$ denote the set of job types.

The intrinsic size of a type- j job is denoted by a generic random variable X_j . Moreover, letting Y_{ij} denote the random speed variation, we assume that Y_{1j}, \dots, Y_{Nj} are i.i.d. copies of some generic random variable S_j . These latter variables can be thought of as job sizes in the standard independent runtime model (taking $X_j = 1$ and $M = 1$ job types) or slowdown factors in the S&X model [4] (taking $R_i = 1$ and $M = 1$ job types). For a particular job on server i , $i = 1, \dots, N$, with intrinsic size x_j , $(x_j Y_{ij})/R_i$ represents the processing time. We distinguish two cases: i) no random speed variation for all job types, i.e., $S_j \equiv c_j$ with $c_j \in \mathbb{R}_+$ for $j = 1, \dots, M$, so-called identical replicas, ii) random speed variation for all job types and servers, so-called i.i.d. replicas.

In the remainder of the paper we distinguish between two scenarios referred to as *Known job types* and *Unknown job types*. In both scenarios the design of the assignment policy may involve knowledge of the type probabilities p_j and service speeds r_{ij} . In the Known job types case, the dispatcher can additionally observe the type identity of each job, and thus knows its service speed at each of the servers. In contrast, in the Unknown job types case, the dispatcher cannot identify jobs by type, and thus has no advance knowledge of service speeds of individual jobs.

2.1. Preliminaries

Let \tilde{p}_{ij} denote the proportion of type- j jobs that are assigned to server i . For given \tilde{p}_{ij} , the stability condition for $d = 1$ and known job types, see also [6, 17], is given by $\lambda \sum_{j=1}^M \tilde{p}_{ij} p_j \frac{\mathbb{E}[X_j] \mathbb{E}[S_j]}{r_{ij}} < 1$ for all $i = 1, \dots, N$. Thus, the achievable stability region is

$$\Lambda_K = \left\{ \lambda \geq 0 \mid \exists \tilde{p}_{ij} \geq 0 : \lambda \sum_{j=1}^M \tilde{p}_{ij} p_j \frac{\mathbb{E}[X_j] \mathbb{E}[S_j]}{r_{ij}} < 1 \text{ for all } i \in \mathcal{N}, \sum_{i=1}^N \tilde{p}_{ij} = 1 \text{ for all } j \in \mathcal{M} \right\}, \quad (1)$$

where the subscript K refers to the case of *known* job types. Note that the stability region given by Equation (1) only depends on the distribution of S_j through its mean $\mathbb{E}[S_j]$ since there is no replication.

Now we proceed with the case of *unknown* job types. Let \tilde{p}_i denote the proportion of jobs assigned to server i , which must be common to all job types when these cannot be distinguished. For given \tilde{p}_i , the stability condition for $d = 1$ is then given by $\sum_{j=1}^M \lambda \tilde{p}_i p_j \frac{\mathbb{E}[X_j] \mathbb{E}[S_j]}{r_{ij}} < 1$ for all $i = 1, \dots, N$. Thus, the achievable stability region is

$$\Lambda_U = \left\{ \lambda \geq 0 \mid \exists \tilde{p}_i \geq 0 : \sum_{j=1}^M \lambda \tilde{p}_i p_j \frac{\mathbb{E}[X_j] \mathbb{E}[S_j]}{r_{ij}} < 1 \text{ for all } i \in \mathcal{N}, \tilde{p}_1 + \dots + \tilde{p}_N = 1 \right\}, \quad (2)$$

where the subscript U refers to the case of *unknown* job types.

The stability region for $d = N$ is also known since the system then behaves as an $M/G/1$ system, see for example [1],

$$\Lambda = [0, \lambda^*), \quad (3)$$

with $\lambda^* = \left(\sum_{j=1}^M p_j \mathbb{E}[X_j] \mathbb{E}[\min\{\frac{Y_{1j}}{r_{1j}}, \dots, \frac{Y_{Nj}}{r_{Nj}}\}] \right)^{-1}$. Note that Λ needs no subscript since the stability region is the same in the cases of known and unknown job types.

In the case of generally distributed job sizes, the next example shows that there is a scenario in which the stability region for $d = 1$ is strictly larger than for $d = N = 2$, both for identical and i.i.d. replicas.

Example 1. Consider the scenario with $N = 2$, $M = 2$ and server speeds $(r_{11}, r_{21}) = (1, x)$ and $(r_{12}, r_{22}) = (x, 1)$ with probabilities $p_1 = p_2 = 0.5$, where $x < 1$. In case of $d = 1$ the optimal static probabilistic assignment is $\tilde{p}_{11} = \tilde{p}_{22} = 1$ and $\tilde{p}_{12} = \tilde{p}_{21} = 0$. Thus, the stability conditions are

$$\begin{aligned} \lambda \sum_{j=1}^2 p_j \mathbb{E}[X_j] \mathbb{E}[S_j] &< 1, && \text{for } d = 2 \text{ (identical),} \\ \lambda \sum_{j=1}^2 p_j \mathbb{E}[X_j] \mathbb{E} \left[\min \left\{ Y_{1j}, \frac{Y_{2j}}{x} \right\} \right] &< 1, && \text{for } d = 2 \text{ (i.i.d.),} \\ \lambda \cdot \frac{1}{2} \cdot \mathbb{E}[X_j] \mathbb{E}[S_j] &< 1, \quad j = 1, 2 && \text{for } d = 1, \end{aligned}$$

where Y_{1j} and Y_{2j} are i.i.d. copies of S_j . Moreover observe that

$$\lim_{x \downarrow 0} \mathbb{E} \left[\min \left\{ Y_{1j}, \frac{Y_{2j}}{x} \right\} \right] = \mathbb{E} \left[\lim_{x \downarrow 0} \min \left\{ Y_{1j}, \frac{Y_{2j}}{x} \right\} \right] = \mathbb{E}[S_j],$$

for every distribution of the speed variation. Thus, if $\mathbb{E}[X_j] \mathbb{E}[S_j] = 1$ for $j = 1, 2$, then the stability condition for $d = 1$ is given by $\lambda < 2$ while for $d = 2$ it is given by $\lambda < 1$.

Definition 1. Consider a non-negative random variable S with support denoted by \mathcal{R}_S and cumulative distribution function (cdf) $F_S(x)$. Let $\bar{F}_S(x) = 1 - F_S(x)$ denote the complementary cumulative distribution function (ccdf). Then, S is New-Better-than-Used (NBU) if for all $t_1, t_2 \in \mathcal{R}_S$,

$$\bar{F}_S(t_1 + t_2) \leq \bar{F}_S(t_1) \bar{F}_S(t_2). \quad (4)$$

On the other hand, S is New-Worse-than-Used (NWU) if for all $t_1, t_2 \in \mathcal{R}_S$,

$$\bar{F}_S(t_1 + t_2) \geq \bar{F}_S(t_1) \bar{F}_S(t_2). \quad (5)$$

Moreover, S is strictly NBU or strictly NWU when Equation (4) or (5) holds with strict inequality, respectively, for all values $t_1, t_2 \in \mathcal{R}_S \setminus \{0\}$.

In the case of strictly NWU distributed speed variations, the next example shows that there is a scenario in which the stability region for $d = N = 2$ is strictly larger than for $d = 1$.

Example 2. Consider the scenario with $N = 2$, $M = 1$ and server speeds $(r_{11}, r_{21}) = (1, 1)$ with probability $p_1 = 1$. The stability conditions for $d = 1$ and $d = 2$ are

$$\begin{aligned} \lambda \cdot \frac{1}{2} \cdot \mathbb{E}[X] \mathbb{E}[S] &< 1, && \text{for } d = 1, \\ \lambda \mathbb{E}[X] \mathbb{E}[\min\{Y_1, Y_2\}] &< 1, && \text{for } d = 2 \text{ (i.i.d.),} \end{aligned}$$

where Y_1 and Y_2 are i.i.d. copies of S . Moreover, by definition of strictly NWU, see for example [18, Sec. 1.6],

$$\mathbb{E}[\min\{Y_1, Y_2\}] < \frac{1}{2}\mathbb{E}[S].$$

Thus the stability region for $d = N = 2$ is strictly larger than the stability region for $d = 1$ in this example.

Observe that $G_j(d) = d\mathbb{E}[\min\{Y_{1j}, \dots, Y_{dj}\}]$ is increasing and decreasing in d for NBU and NWU distributions, respectively, see for example [18, Sec. 1.6]. Here, $G_j(d)$ may be interpreted as the aggregate resource usage for d replicas with equal start times on homogeneous servers under redundancy c.o.c., and has emerged as a key metric for stability conditions in scenarios where the servers are partitioned in disjoint pools of d servers, see for instance [9]. We will extend this notion to scenarios with heterogeneous servers and additionally deal with the complex dynamics arising from potentially different start times as a result of queuing which may occur when servers are not partitioned in the above manner.

In the proofs of the main theorems in the next section a property of c.o.c. redundancy systems, viz., Property 1 below, is needed. Note that this property is valid for all scenarios.

Property 1. *The oldest job in the system is served at all servers that it has been replicated to.*

3. No replication is best for NBU speed variations

In this section we prove that no replication maximizes stability when the speed variations are NBU distributed, see Theorem 2. First however we consider the special case where the speed variation of each job type j , S_j , follows a degenerate distribution, see Theorem 1. The proof of this latter theorem is simpler and gives intuition for the general case with NBU distributed speed variations. Both theorems rely on the next proposition.

Proposition 1. *Assume that $r_{ij} > 0$ for all $i = 1, \dots, N$, $j = 1, \dots, M$, and that the system is stable under a given assignment policy with $d > 1$ for some arrival rate $\lambda_0 > 0$. Let τ_{ij} be the long-term fraction of time that server i spends on type- j jobs under this assignment policy with $d > 1$. Suppose that*

$$\sum_{i=1}^N r_{ij}\tau_{ij} \geq \lambda_0 p_j \mathbb{E}[X_j] \mathbb{E}[S_j], \quad (6)$$

for all $j = 1, \dots, M$, and in addition

$$\sum_{j=1}^M \sum_{i=1}^N r_{ij}\tau_{ij} \geq \lambda_0(1 + \epsilon) \sum_{j=1}^M p_j \mathbb{E}[X_j] \mathbb{E}[S_j] \quad (7)$$

for some $\epsilon \geq 0$. Then the system can be stabilized through a suitable probabilistic assignment policy with $d = 1$ for all $\lambda \leq \lambda_0(1 + \kappa\epsilon)$, with $\kappa > 0$ a fixed constant bounded away from zero, independent of λ_0 .

Proof. The high-level idea of the proof may be outlined as follows. The inequalities in (6) imply that the weighted fraction of time that the servers collectively spend on type- j jobs under the policy with $d > 1$ is no less than the offered load of type- j jobs, i.e., what this weighted fraction would be without any replication, for each job type $j = 1, \dots, M$. This allows us to distribute the type- j jobs without any replication through suitable assignment probabilities \tilde{p}_{ij} in terms of the r_{ij} and τ_{ij} to sustain the same arrival rate λ_0 without increasing the load of any of the servers, thus ensuring stability. Hence, the statement of the proposition follows when $\epsilon = 0$. When $\epsilon > 0$, the inequality (7) implies that the total weighted amount of time that the servers are collectively occupied under the policy with $d > 1$ is strictly larger than the total offered load. This margin reflects that there is some slack capacity that can be freed up when refraining from replication, and thus be exploited to accommodate a higher arrival rate than λ_0 . While there are several options for dividing the slack capacity, we will simply use assignment probabilities that account for the amount of slack at each server and its speeds for the various job types, but do not depend on the job type. Once again, this will not increase the load of any of the servers, but allow us to support a strictly higher arrival rate.

In order to develop the proof in greater detail, observe that the stability under the given assignment policy with $d > 1$ implies that the long-term fraction of time that each server is busy must be strictly less than unity, i.e., $\sum_{j=1}^M \tau_{ij} < 1$ for all $i = 1, \dots, N$. (For transparency, we tacitly assume here and in the statement of the proposition that these long-term fractions exist, and thus implicitly rule out possibly eccentric (e.g. non-stationary) assignment policies. The proof arguments below could however readily be extended to cover such policies as well, if we stipulate stability to mean that the limsup values of $\sum_{j=1}^M \tau_{ij}$ must be strictly less than unity for all $i = 1, \dots, N$.)

Now consider the system with $d = 1$ and assignment probabilities

$$\tilde{p}_{ij} = \frac{r_{ij}\tau_{ij}}{\sum_{k=1}^N r_{kj}\tau_{kj}}.$$

Then each server behaves as a multi-class $M/G/1$ queue, and for an overall arrival rate $\lambda \leq \lambda_0$ the load on server i is

$$\lambda \sum_{j=1}^M p_j \tilde{p}_{ij} \frac{\mathbb{E}[X_j]\mathbb{E}[S_j]}{r_{ij}} = \lambda \sum_{j=1}^M p_j \tau_{ij} \frac{\mathbb{E}[X_j]\mathbb{E}[S_j]}{\sum_{k=1}^N r_{kj}\tau_{kj}} \leq \sum_{j=1}^M \tau_{ij} < 1, \quad \forall i = 1, \dots, N,$$

where the last-but-one inequality follows from (6) and the fact that $\lambda \leq \lambda_0$, implying that the system is stable. This completes the proof in case $\epsilon = 0$.

In order to prove the statement in case $\epsilon > 0$, let

$$\sigma_j = \frac{\lambda_0 p_j \mathbb{E}[X_j]\mathbb{E}[S_j]}{\sum_{i=1}^N r_{ij}\tau_{ij}} \leq 1,$$

representing the offered load of type- j jobs as fraction of the weighted amount of time spent on these jobs by the servers collectively under the given assignment policy with $d > 1$, and define

$$\begin{aligned} \hat{\tau}_{ij} &= \sigma_j \tau_{ij} \leq \tau_{ij}, \\ \Delta\tau_{ij} &= \tau_{ij} - \hat{\tau}_{ij} = (1 - \sigma_j)\tau_{ij}, \end{aligned}$$

and

$$\Delta\tau_i = \sum_{j=1}^M \tau_{ij} - \sum_{j=1}^M \hat{\tau}_{ij} = \sum_{j=1}^M \Delta\tau_{ij}.$$

The value of $\hat{\tau}_{ij}$ may be interpreted as the fraction of time that server i would need to spend on type- j jobs if the efforts of all servers for type- j jobs are reduced proportionally to match the total offered load. With that interpretation in mind, $\Delta\tau_{ij}$ and $\Delta\tau_i$ may be thought of as measures for the slack capacity.

Further introduce

$$\Delta_j = \sum_{i=1}^N r_{ij}\tau_{ij} - \lambda_0 p_j \mathbb{E}[X_j] \mathbb{E}[S_j] = \sum_{i=1}^N r_{ij}\tau_{ij} - \sum_{i=1}^N r_{ij}\hat{\tau}_{ij} = \sum_{i=1}^N r_{ij}\Delta\tau_{ij}$$

representing the slack between the weighted fraction of time that the servers collectively spend on type- j jobs under the policy with $d > 1$ and the offered load of type- j jobs,

$$r_i = \frac{\sum_{j=1}^M p_j \mathbb{E}[X_j] \mathbb{E}[S_j]}{\sum_{j=1}^M p_j \mathbb{E}[X_j] \mathbb{E}[S_j] / r_{ij}}$$

representing the time-average speed of server i when handling jobs of the various types in the nominal proportions, and

$$\Delta\lambda = \frac{\sum_{k=1}^N r_k \Delta\tau_k}{\sum_{j=1}^M p_j \mathbb{E}[X_j] \mathbb{E}[S_j]}.$$

Now observe that on the one hand

$$\sum_{j=1}^M \Delta_j = \sum_{j=1}^M \sum_{i=1}^N r_{ij}\tau_{ij} - \lambda_0 \sum_{j=1}^M p_j \mathbb{E}[X_j] \mathbb{E}[S_j] \geq \lambda_0 \epsilon \sum_{j=1}^M p_j \mathbb{E}[X_j] \mathbb{E}[S_j],$$

while on the other hand

$$\sum_{j=1}^M \Delta_j = \sum_{j=1}^M \sum_{i=1}^N r_{ij}\Delta\tau_{ij} \leq \sum_{i=1}^N \Delta\tau_i \max_{j \in \mathcal{M}} r_{ij},$$

and hence

$$\Delta\lambda \geq \frac{\sum_{k=1}^N r_k \Delta\tau_k}{\sum_{i=1}^N \Delta\tau_i \max_{j \in \mathcal{M}} r_{ij}} \epsilon \lambda_0.$$

Noting that $r_i > 0$ by virtue of the assumption that $r_{ij} > 0$ for all $i = 1, \dots, N$ and $j = 1, \dots, M$, we obtain that

$$\Delta\lambda \geq \kappa \epsilon \lambda_0,$$

with $\kappa = \frac{\min_{i \in \mathcal{N}} r_i}{\max_{i \in \mathcal{N}, j \in \mathcal{M}} r_{ij}} > 0$.

Now consider the system with $d = 1$ and total arrival rate $\lambda_0 + \Delta\lambda$, and suppose that a fraction $\lambda_0/(\lambda_0 + \Delta\lambda)$ of the jobs are assigned according to the probabilities \tilde{p}_{ij} , while the remaining fraction $\Delta\lambda/(\lambda_0 + \Delta\lambda)$ of the jobs are assigned to server i with probability

$$\hat{p}_i = \frac{r_i \Delta\tau_i}{\sum_{k=1}^N r_k \Delta\tau_k}.$$

Then each server behaves as a multi-class $M/G/1$ queue, and for an overall arrival rate $\lambda \leq \lambda_0 + \Delta\lambda$ the load on server i is

$$\begin{aligned} & \lambda \sum_{j=1}^M p_j \left(\frac{\lambda_0}{\lambda_0 + \Delta\lambda} \tilde{p}_{ij} + \frac{\Delta\lambda}{\lambda_0 + \Delta\lambda} \hat{p}_i \right) \frac{\mathbb{E}[X_j] \mathbb{E}[S_j]}{r_{ij}} \\ &= \frac{\lambda}{\lambda_0 + \Delta\lambda} \sum_{j=1}^M p_j \left(\lambda_0 \frac{r_{ij} \tau_{ij}}{\sum_{k=1}^N r_{kj} \tau_{kj}} + \Delta\lambda \frac{r_i \Delta\tau_i}{\sum_{k=1}^N r_k \Delta\tau_k} \right) \frac{\mathbb{E}[X_j] \mathbb{E}[S_j]}{r_{ij}} \\ &\leq \sum_{j=1}^M \tau_{ij} \frac{\lambda_0 p_j \mathbb{E}[X_j] \mathbb{E}[S_j]}{\sum_{k=1}^N r_{kj} \tau_{kj}} + \frac{\sum_{k=1}^N r_k \Delta\tau_k}{\sum_{j=1}^M p_j \mathbb{E}[X_j] \mathbb{E}[S_j]} \frac{r_i \Delta\tau_i}{\sum_{k=1}^N r_k \Delta\tau_k} \sum_{j=1}^M \frac{p_j \mathbb{E}[X_j] \mathbb{E}[S_j]}{r_{ij}} \\ &= \sum_{j=1}^M \tau_{ij} \sigma_j + r_i \Delta\tau_i \frac{\sum_{j=1}^M p_j \mathbb{E}[X_j] \mathbb{E}[S_j] / r_{ij}}{\sum_{j=1}^M p_j \mathbb{E}[X_j] \mathbb{E}[S_j]} \\ &= \sum_{j=1}^M \hat{\tau}_{ij} + \Delta\tau_i = \sum_{j=1}^M \tau_{ij} < 1, \quad \forall i = 1, \dots, N, \end{aligned}$$

where the inequality in the third line follows from the fact that $\lambda \leq \lambda_0 + \Delta\lambda$.

This yields the statement of the proposition for any $\epsilon \geq 0$. \square

Remark 1. We now present an example illustrating the role of the assumption that $r_{ij} > 0$ for all $i = 1, \dots, N$ and $j = 1, \dots, M$. Consider a system with $N = 3$ servers, $M = 2$ job types, and service speeds $(r_{11}, r_{21}, r_{31}) = (1, 0, 0)$ and $(r_{12}, r_{22}, r_{32}) = (1, 1, 1)$. Assume that $p_j = 1/2$, $E[S_j] = 1$, $E[X_j] = 1$, $j = 1, 2$, $d = 2$, and that all type-1 jobs are assigned to servers 1 and 3 while all type-2 jobs are assigned to servers 2 and 3. We claim that the system is stable for any $\lambda < 2$. In order to see that, observe that the number of type-1 jobs and the number of type-2 jobs are each individually bounded from above by the number of the jobs in an $M/G/1$ queue with load $\lambda/2$. Furthermore, type-1 jobs will never complete on server 3, while in case of identical service times, type-2 jobs will never complete on server 3 before completing on server 2. In other words, all effort of server 3 goes wasted. Nevertheless, a system with $d = 1$ cannot be stabilized for any $\lambda \geq 2$, since type-1 jobs can only be successfully processed by server 1. The wasted effort of server 3 could however be avoided in a system with $d = 1$ to sustain an arrival rate of type-2 jobs that is twice as large.

While the assumption that $r_{ij} > 0$ for all $i = 1, \dots, N$ may in general not be strictly necessary, this example demonstrates that it cannot easily be relaxed without creating a need for a tedious case-by-case analysis to determine whether the system with $d = 1$ can be stabilized for a higher overall arrival rate, can only accommodate a larger arrival rate for some of the job types, or cannot support a higher arrival rate for any job type at all.

Theorem 1. *In the case of known job types, the stability region for $d = 1$ is strictly larger than the stability region for $d > 1$ under the c.o.c. redundancy policy with identical replicas and static probabilistic assignment (which may depend on the job type) of the d replicas.*

Proof. Let $\tau_{ij}^{(1)}$ be the fraction of time that server i spends on type- j jobs that it will finish and $\tau_{ij}^{(2)}$ be the fraction of time that server i spends on type- j jobs that it will not finish, with $\tau_{ij}^{(1)} + \tau_{ij}^{(2)} = \tau_{ij}$ under a given assignment policy with $d > 1$ for arrival rate λ .

For the effective component we have

$$\sum_{i=1}^N r_{ij} \tau_{ij}^{(1)} = \lambda p_j \mathbb{E}[X_j] \mathbb{E}[S_j], \quad (8)$$

since

$$\sum_{i=1}^N r_{ij} \mathbb{E}[T_{ij}^{(1)}] = \mathbb{E}[X_j] \mathbb{E}[S_j],$$

where $T_{ij}^{(1)}$ is the amount of time that server i spends on a type- j job that it will finish, with $\tau_{ij}^{(1)} = \lambda p_j \mathbb{E}[T_{ij}^{(1)}]$. This holds because for identical replicas there are no server-dependent slow downs and whether or not a server will finish a particular job is not influenced by the random speed variations.

For the wastage component we have by Property 1 that

$$\sum_{i=1}^N r_{ij} \tau_{ij}^{(2)} \geq \sum_{i=1}^N \min_{k \in \mathcal{N}} r_{kj} \tau_{ij}^{(2)} = \min_{k \in \mathcal{N}} r_{kj} \sum_{i=1}^N \tau_{ij}^{(2)} \geq \min_{k \in \mathcal{N}} r_{kj} (d-1) \bar{\pi}_{0j}, \quad (9)$$

where $\bar{\pi}_{0j}$ is the fraction of time that the system is non-empty in the limit as time goes to infinity and the oldest job is of type j . Letting $\bar{\pi}_0$ be the fraction of time that the system is non-empty with $\sum_{j=1}^M \bar{\pi}_{0j} = \bar{\pi}_0$, it follows that

$$\sum_{j=1}^M \sum_{i=1}^N r_{ij} \tau_{ij}^{(2)} \geq \min_{k \in \mathcal{N}, l \in \mathcal{M}} r_{kl} (d-1) \bar{\pi}_0.$$

We can bound the fraction of time that the system is non-empty as

$$\bar{\pi}_0 \geq \sum_{j=1}^M \frac{\lambda p_j \mathbb{E}[X_j] \mathbb{E}[S_j]}{\sum_{i=1}^N r_{ij}} \geq \frac{\lambda \sum_{j=1}^M p_j \mathbb{E}[X_j] \mathbb{E}[S_j]}{\max_{j \in \mathcal{M}} \sum_{i=1}^N r_{ij}} > 0.$$

Substituting this in Equation (9) gives

$$\sum_{j=1}^M \sum_{i=1}^N r_{ij} \tau_{ij} \geq \lambda \left(1 + (d-1) \frac{\min_{k \in \mathcal{N}, l \in \mathcal{M}} r_{kl}}{\max_{j \in \mathcal{M}} \sum_{i=1}^N r_{ij}} \right) \sum_{j=1}^M p_j \mathbb{E}[X_j] \mathbb{E}[S_j],$$

so that Equation (7) holds with $\epsilon = (d-1) \frac{\min_{k \in \mathcal{N}, l \in \mathcal{M}} r_{kl}}{\max_{j \in \mathcal{M}} \sum_{i=1}^N r_{ij}}$ which is bounded away from zero. Noting that (8) with in addition $\tau_{ij}^{(1)} + \tau_{ij}^{(2)} = \tau_{ij}$ gives (6), the proof then follows from Proposition 1. \square

Remark 2. In Theorem 1 we obtained a lower bound for the wastage component that is strictly increasing in d , see Equation (9). We can also find an upper bound for the wastage component

$$\sum_{j=1}^M \sum_{i=1}^N r_{ij} \tau_{ij}^{(2)} \leq \max_{k \in N, l \in M} r_{kl} \left(N - \left\lfloor \frac{N}{d} \right\rfloor \right) \bar{\pi}_0^* \leq \max_{k \in N, l \in M} r_{kl} \left(N - \left\lfloor \frac{N}{d} \right\rfloor \right) \bar{\pi}_0, \quad (10)$$

where $\bar{\pi}_0^*$ is the fraction of time that all servers are busy and thus $\bar{\pi}_0^* \leq \bar{\pi}_0$. Note that in the special case of homogeneous server speeds and $d = 1$, $d = N - 1$ and $d = N$ the lower- and upper bound for the wastage component coincide. It is therefore natural to conjecture that Theorem 1 extends to the statement that the stability region is strictly decreasing in d .

We proceed with the general case of speed variations that are NBU distributed.

Theorem 2. In the case of known job types, the stability region for $d = 1$ is larger than or equal to (respectively, strictly larger than) the stability region for $d > 1$ with NBU (respectively, strictly NBU) distributed speed variations and static probabilistic assignment (which may depend on the job type) of the d replicas.

Proof. Let T_{ij} be the amount of time that server i spends on an arbitrary type- j job and let τ_{ij} be the fraction of time that server i spends on type- j jobs under a given assignment policy with $d > 1$ for arrival rate λ as introduced before.

Let $T_{\text{awt},j}$ with distribution function $F_{T_{\text{awt},j}}(t)$ (respectively, $T_{\text{awt},j}^I$ with distribution function $F_{T_{\text{awt},j}^I}(t)$) denote the aggregate weighted amount of time, weighted by the server speeds r_{ij} , invested in the service of an arbitrary type- j job divided by the intrinsic job size (respectively, under the assignment I , where I denotes an arbitrary set of d servers). We have that $T_{\text{awt},j}^I$ is equal in distribution to $\sum_{i \in I} \frac{r_{ij} T_{ij}}{X_j}$ (see Figure 1 for a schematic illustration), when server I_i is available after the weighted amount of time b_i of servers I_1, \dots, I_{i-1} , for $i = 2, \dots, d$ and $b_1 = 0$. Thus, a replica of the job is first served on server I_1 and after time b_2 server I_2 becomes available to serve another replica of this job, then after time b_3 the third server I_3 becomes available to serve yet another replica of this job, etc. Note that server I_i may not necessarily serve this job, i.e., the job may already be completed before the server is available. The cdf is

$$\bar{F}_{T_{\text{awt},j}^I}(t) = \begin{cases} \mathbb{P}\left(Y_{I_1,j} > \frac{r_{I_1,j} t}{r_{I_1,j}}\right) & \text{for } 0 < t < b_2, \\ \mathbb{P}\left(Y_{I_1,j} > b_2 + \frac{r_{I_1,j}(t-b_2)}{r_{I_1,j}+r_{I_2,j}}\right) \cdot \mathbb{P}\left(Y_{I_2,j} > \frac{r_{I_2,j}(t-b_2)}{r_{I_1,j}+r_{I_2,j}}\right) & \text{for } b_2 < t < \sum_{l=1}^3 (b_3 - b_l), \\ \vdots & \\ \mathbb{P}\left(Y_{I_1,j} > b_2 + \frac{r_{I_1,j} 2(b_3 - b_2)}{r_{I_1,j} + r_{I_2,j}} + \dots + \frac{r_{I_1,j}(t - \sum_{l=1}^d (b_d - b_l))}{\sum_{i \in I} r_{ij}}\right) \dots & \\ \cdot \mathbb{P}\left(Y_{I_d,j} > \frac{r_{I_d,j}(t - \sum_{l=1}^d (b_d - b_l))}{\sum_{i \in I} r_{ij}}\right) & \text{for } \sum_{l=1}^d (b_d - b_l) < t. \end{cases}$$

Hence

$$\bar{F}_{T_{\text{awt},j}^I}(t) = \begin{cases} \bar{F}_{Y_{I_1,j}}(t) & \text{for } 0 < t < b_2, \\ \bar{F}_{Y_{I_1,j}}\left(b_2 + \frac{r_{I_1,j}(t-b_2)}{r_{I_1,j}+r_{I_2,j}}\right) \cdot \bar{F}_{Y_{I_2,j}}\left(\frac{r_{I_2,j}(t-b_2)}{r_{I_1,j}+r_{I_2,j}}\right) & \text{for } b_2 < t < \sum_{l=1}^3 (b_3 - b_l), \\ \vdots & \\ \bar{F}_{Y_{I_1,j}}\left(b_2 + \frac{r_{I_1,j} 2(b_3 - b_2)}{r_{I_1,j} + r_{I_2,j}} + \dots + \frac{r_{I_1,j}(t - \sum_{l=1}^d (b_d - b_l))}{\sum_{i \in I} r_{ij}}\right) \dots & \\ \cdot \bar{F}_{Y_{I_d,j}}\left(\frac{r_{I_d,j}(t - \sum_{l=1}^d (b_d - b_l))}{\sum_{i \in I} r_{ij}}\right) & \text{for } \sum_{l=1}^d (b_d - b_l) < t, \end{cases}$$

and by definition of NBU distributions we get

$$\bar{F}_{T_{\text{awt},j}^l}(t) \geq \begin{cases} \bar{F}_{S_j}(t) & \text{for } 0 < t < b_2, \\ \bar{F}_{S_j}\left(b_2 + \frac{r_{1j}(t-b_2)}{r_{1j}+r_{2j}} + \frac{r_{2j}(t-b_2)}{r_{1j}+r_{2j}}\right) = \bar{F}_{S_j}(t) & \text{for } b_2 < t < \sum_{l=1}^3(b_3 - b_l), \\ \vdots \\ \bar{F}_{S_j}\left(b_2 + \frac{r_{1j}2(b_3-b_2)}{r_{1j}+r_{2j}} + \dots + \frac{r_{1j}(t-\sum_{l=1}^d(b_d-b_l))}{\sum_{i \in l} r_{ij}} + \dots + \frac{r_{dj}(t-\sum_{l=1}^d(b_d-b_l))}{\sum_{i \in l} r_{ij}}\right) = \bar{F}_{S_j}(t) & \text{for } \sum_{l=1}^d(b_d - b_l) < t. \end{cases} \quad (11)$$

It then follows that the expected aggregate weighted amount of time invested in the service of a job is larger than or equal to the mean size of a single job instance, i.e.,

$$\sum_{i=1}^N r_{ij} \mathbb{E}[T_{ij}] = \mathbb{E}[X_j] \int_{t=0}^{\infty} \bar{F}_{T_{\text{awt},j}}(t) dt \geq \mathbb{E}[X_j] \int_{t=0}^{\infty} \bar{F}_{S_j}(t) dt = \mathbb{E}[X_j] \mathbb{E}[S_j], \quad (12)$$

and substituting $\tau_{ij} = \lambda p_j \mathbb{E}[T_{ij}]$ yields Equation (6)

$$\sum_{i=1}^N r_{ij} \tau_{ij} \geq \lambda p_j \mathbb{E}[X_j] \mathbb{E}[S_j].$$

Summing over all the job types gives Equation (7)

$$\sum_{j=1}^M \sum_{i=1}^N r_{ij} \tau_{ij} \geq \lambda \sum_{j=1}^M p_j \mathbb{E}[X_j] \mathbb{E}[S_j]. \quad (13)$$

Note that at this point, by Proposition 1 with $\epsilon = 0$, it follows that the stability region for $d = 1$ is larger than or equal to the stability region for $d > 1$ in the case of NBU distributed speed variations.

In the case of a strictly NBU distribution Equation (11) is a strict inequality if two or more servers are serving this particular job, i.e., for $t > b_2$. We proceed by proving that Equation (13) holds with strict inequality.

Note that we can write Equation (12) as

$$\sum_{i=1}^N r_{ij} \mathbb{E}[T_{ij}] = \mathbb{E}[X_j] \int_{t=0}^{\infty} \bar{F}_{T_{\text{awt},j}}(t) dt \geq \mathbb{E}[X_j] \left(\int_{t=0}^{\infty} \bar{F}_{S_j}(t) dt + \mathbb{E}[L(d, S_j, B)] \right), \quad (14)$$

where the latter expectation is with respect to S and where

$$L(d, S_j, \mathbf{b}) = \int_{t=\sum_{l=1}^d(b_d-b_l)}^{\infty} (\bar{F}_{T_{\text{awt},j}}(t) - \bar{F}_{S_j}(t)) dt$$

denotes the difference between, starting from the time a job is in service at d servers, of the aggregate weighted amount of time invested in the service of an arbitrary type- j job and the job size under the distributions of X and B , where $B = (B_2, \dots, B_d)$ is the random variable that

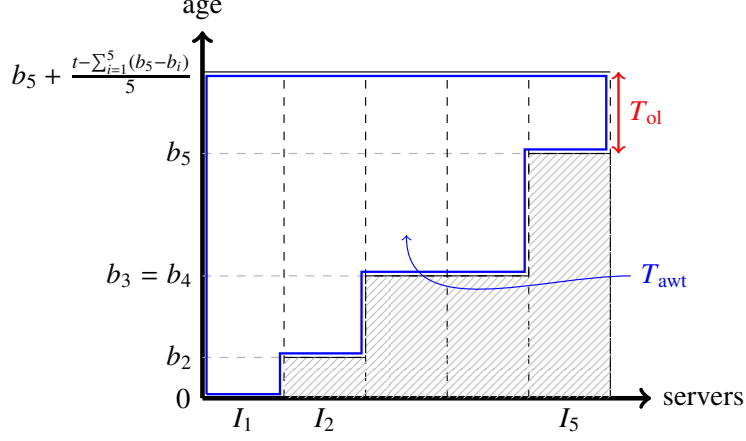


Figure 1: Illustration of the definition of T_{await} and T_{ol} in case of homogeneous server speeds.

denotes the weighted amount of time after which the server is available, with $0 \leq B_2 \leq \dots \leq B_d$ and joint probability density function $f_B(b_2, \dots, b_d)$. Although all jobs that are in service at two or more servers contribute to the strict inequality of Equation (13), we only consider the job that is in service at all the d servers. Moreover, by Equation (11) we know that $\bar{F}_{T_{\text{await},j}}(t) > \bar{F}_{S_j}(t)$ for all $t \geq \sum_{l=1}^d (b_d - b_l)$. Note that by Property 1, if the system is non-empty, there is always a job that is served at all the servers that it has been replicated to. Substituting $\tau_{ij} = \lambda p_j \mathbb{E}[T_{ij}]$ in Equation (14) and summing over all the job types gives

$$\begin{aligned} \sum_{j=1}^M \sum_{i=1}^N r_{ij} \tau_{ij} &\geq \sum_{j=1}^M \left(\lambda p_j \mathbb{E}[X_j] \mathbb{E}[S_j] + \min_{k \in N, l \in M} r_{kl} \lambda \mathbb{E}[L(d, S_j, B)] \right) \\ &\geq \lambda \sum_{j=1}^M p_j \mathbb{E}[X_j] \mathbb{E}[S_j] \left(1 + \frac{\min_{k \in N, l \in M} r_{kl} \mathbb{E}[L(d, S_j, B)]}{p_j \mathbb{E}[X_j] \mathbb{E}[S_j]} \right). \end{aligned} \quad (15)$$

To prove the strict inequality in Equation (13) we have to show that $\mathbb{E}[L(d, S_j, B)] > 0$.

Let $T_{\text{ol}}(d, S_j, \mathbf{b})$ denote the overlap in the service of an arbitrary type- j job, see Figure 1 for a visual interpretation, then

$$\mathbb{E}[T_{\text{ol}}(d, S_j, B)] = \int_{b_2=0}^{\infty} \dots \int_{b_d=0}^{\infty} T_{\text{ol}}(d, S_j, \mathbf{b}) f_B(b_2, \dots, b_d) db_2 \dots db_d, \quad (16)$$

where

$$T_{\text{ol}}(d, S_j, \mathbf{b}) = \int_{t=\sum_{l=1}^d (b_d - b_l)}^{\infty} \frac{1}{d} \cdot \bar{F}_{T_{\text{await},j}}(t) dt.$$

Since $\lambda \mathbb{E}[T_{\text{ol}}(d, S_j, B)] \geq \bar{\pi}_0$, we can get a lower bound for the expected overlap

$$\mathbb{E}[T_{\text{ol}}(d, S_j, B)] \geq \frac{\bar{\pi}_0}{\lambda} \geq \frac{1}{\lambda} \sum_{j=1}^M \frac{\lambda p_j \mathbb{E}[\min\{Y_{1j}, \dots, Y_{dj}\}]}{\sum_{i=1}^N r_{ij}} \geq \frac{\mathbb{E}[\min\{Y_{1j}, \dots, Y_{dj}\}]}{\max_{j \in M} \sum_{i=1}^N r_{ij}}.$$

Observe that from this lower bound and Equation (16) it follows that there exists $\tau(\delta) < \max\{\mathcal{R}_{S_j}\}$ such that $\mathbb{P}(B_d < \tau(\delta)) \geq \delta$, otherwise $\mathbb{E}[T_{ol}(d, S_j, B)]$ is too small. Using this we can write

$$\begin{aligned}\mathbb{E}[L(d, S_j, B)] &\geq \mathbb{P}(B_d < \tau(\delta)) \cdot \mathbb{E}[L(d, S_j, B)|B_d < \tau(\delta)] \\ &\geq \delta \mathbb{E}[L(d, S_j, B)|B_d < \tau(\delta)] \\ &\geq \delta \int_{t=(d-1)\tau(\delta)}^{\infty} (\bar{F}_{T_{awt,j}}(t) - \bar{F}_{S_j}(t)) dt =: \delta I(\delta) > 0.\end{aligned}$$

Hence,

$$\sum_{j=1}^M \sum_{i=1}^N r_{ij} \tau_{ij} \geq \lambda \sum_{j=1}^M p_j \mathbb{E}[X_j] \mathbb{E}[S_j] \left(1 + \frac{\min_{k \in \mathcal{N}, l \in \mathcal{M}} r_{kl} \delta I(\delta)}{p_j \mathbb{E}[X_j] \mathbb{E}[S_j]}\right).$$

Now the proof follows by Proposition 1, with $\epsilon = \min_{j \in \mathcal{M}} \frac{\min_{k \in \mathcal{N}, l \in \mathcal{M}} r_{kl} \delta I(\delta)}{p_j \mathbb{E}[X_j] \mathbb{E}[S_j]}$ which is bounded away from zero. \square

Remark 3. We obtained a lower bound for $\mathbb{E}[L(d, S_j, B)]$ which is strictly increasing in d , see Theorem 2, but have no meaningful upper bound for this expression. Nonetheless, it would be natural to conjecture that Theorem 2 extends to the statement that the stability region is strictly decreasing in d .

Remark 4. In Theorems 1 and 2 we restricted ourselves to static probabilistic assignment of the d replicas. This restriction could probably be relaxed to dynamic assignments policies. Think for example of an assignment policy that replicates the job to, say, \tilde{d} servers, where \tilde{d} is a realization from some underlying distribution which may depend on the job type.

In the next subsection we show, by providing counterexamples, that the assumptions in Theorems 1 and 2, i.e., known job types and static probabilistic type-dependent assignment, are in fact *necessary*.

3.1. Necessary assumptions

In this section we analyze the stability region in cases where the assumptions in Theorems 1 and 2 do not all hold.

Example 3. Consider the scenario of Example 1, i.e., $N = 2$, $M = 2$ and server speeds $(r_{11}, r_{21}) = (1, x)$ and $(r_{12}, r_{22}) = (x, 1)$ with probabilities $p_1 = p_2 = 0.5$, where $x < 1$. However, in this scenario the job types are unknown. In case of $d = 1$, unknown job types implies that both servers are equivalent, thus the optimal static probabilistic assignment is in that case $\tilde{p}_{1j} = \tilde{p}_{2j} = 0.5$ for $j = 1, 2$. The stability conditions, see also Equation (2), are

$$\begin{aligned}\lambda \sum_{j=1}^2 p_j \mathbb{E}[X_j] \mathbb{E}[S_j] &< 1, && \text{for } d = 2 \text{ (identical),} \\ \lambda \sum_{j=1}^2 p_j \mathbb{E}[X_j] \mathbb{E} \left[\min \left\{ Y_{1j}, \frac{Y_{2j}}{x} \right\} \right] &< 1, && \text{for } d = 2 \text{ (i.i.d.),} \\ \lambda \sum_{j=1}^2 p_j \mathbb{E}[X_j] \mathbb{E}[S_j] \left(0.5 + \frac{0.5}{x} \right) &< 1, && \text{for } d = 1,\end{aligned}$$

where Y_{1j} and Y_{2j} are i.i.d. copies of S . Note that $\mathbb{E}\left[\min\left\{Y_{1j}, \frac{Y_{2j}}{x}\right\}\right] \leq \mathbb{E}[S_j]$, and that $0.5 + \frac{0.5}{x} > 1$ for $x < 1$.

The above example shows that Theorems 1 and 2 do not hold when job types cannot be distinguished.

Random job assignment:

To achieve the largest stability region with no replication we allowed for static probabilistic assignment of jobs. Example 3 illustrates that Theorems 1 and 2 do not hold when we restrict the assignment probabilities of jobs in case of no replication to be uniform.

3.2. No replication may be best for NWU speed variations

Example 1 already showed that even for NWU speed variations, in this specific scenario, no replication gives a larger stability region than full replication. However, Example 2 showed that in the scenario with homogeneous server speeds full replication gives a larger stability region. From both examples we conclude that in the case of known job types and NWU distributed speed variations the number of replicas that achieves the largest stability region heavily depends on the server speeds. Loosely speaking, full replication or no replication gives the largest stability region if the server speeds within a job type are balanced and unbalanced, respectively.

4. Full replication is best for NWU speed variations

In this section we prove that full replication gives a larger stability region than no replication when the speed variations are NWU distributed and job types cannot be observed, see Theorem 3. We also discuss the possible extensions of this statement, replacing no replication by an arbitrary number of replicas, in Conjectures 1 and 2.

We first introduce some useful notation. Consider $K = \binom{N}{d}$ probabilities, where each probability corresponds to assigning a job to one of the $\binom{N}{d}$ possible combinations of d servers. Let $s^i \subset \{1, \dots, N\}$ denote the set of servers corresponding to the i -th probability. Without loss of generality, we suppose that \tilde{p}_1 corresponds to the set of servers $s^1 = \{1, \dots, d\}$, \tilde{p}_2 corresponds to the set of servers $s^2 = \{1, \dots, d-1, d+1\}$ and finally \tilde{p}_K corresponds to the set of servers $s^K = \{N-d+1, \dots, N\}$, with $\sum_{i=1}^K \tilde{p}_i = 1$.

For brevity, we further define $\gamma_i = \sum_{j=1}^M p_j \mathbb{E}[X_j] \gamma_{ij}$, with

$$\gamma_{ij} = \sum_{h:i \in s^h} \frac{\tilde{p}_h}{\sum_{h^*:i \in s^{h^*}} \tilde{p}_{h^*}} \theta_{ijh},$$

and

$$\theta_{ijh} = \mathbb{E}\left[\min\left\{\frac{Y_{1j}}{r_{s^h_j}}, \dots, \frac{Y_{dj}}{r_{s^h_j}}\right\}\right],$$

representing the expected execution time per unit size for a type- j job assigned to the set of servers $s^h \ni i$ if all d replicas were to start at the same time. Thus, γ_i may be interpreted as a

proxy for the load associated with an arbitrary job assigned to server i . In case the random speed variation S_j is exponentially distributed, the expression for θ_{ijh} reduces to

$$\hat{\theta}_{ijh} = \frac{\mathbb{E}[S_j]}{\sum_{l=1}^d r_{s_{lj}}^h},$$

and we will add a hat to the coefficients γ_i in that case accordingly and informally refer to these as the *exponential* load values. For $d = 1$, the expression for γ_{ij} simplifies to $\mathbb{E}[S_j]/r_{ij}$, yielding

$$\tilde{\gamma}_i = \sum_{j=1}^M p_j \mathbb{E}[X_j] \frac{\mathbb{E}[S_j]}{r_{ij}},$$

which is in fact the *exact* load in that case. For $d = N$, the values of γ_{ij} are all equal to

$$\theta_j = \mathbb{E}[\min \left\{ \frac{Y_{1j}}{r_{1j}}, \dots, \frac{Y_{Nj}}{r_{Nj}} \right\}],$$

and hence the values of γ_i are all equal to

$$\gamma_0 = \sum_{j=1}^M p_j \mathbb{E}[X_j] \theta_j,$$

which is also the *exact* load since all the N replicas are guaranteed to start at the same time. Finally note that in case the random speed variation S_j is exponentially distributed, the expression for θ_j simplifies to

$$\hat{\theta}_j = \frac{\mathbb{E}[S_j]}{\sum_{i=1}^N r_{ij}},$$

yielding

$$\hat{\gamma}_0 = \sum_{j=1}^M p_j \mathbb{E}[X_j] \frac{\mathbb{E}[S_j]}{\sum_{i=1}^N r_{ij}}.$$

In the next theorem we prove that full replication gives a (strictly) larger stability region than no replication when the speed variations are (strictly) NWU distributed.

Theorem 3. *In the case of unknown job types, the stability region for $d = N$ is larger than or equal to (respectively, strictly larger than) the stability region for $d = 1$ with NWU (respectively, strictly NWU) distributed speed variations and static probabilistic assignment (which cannot depend on the job type) of the d replicas.*

Proof. For $d = 1$, the stability condition is $\max_{i \in N} \tilde{p}_i \tilde{\gamma}_i < 1$ for some probabilities \tilde{p}_i , see (2). For $d = N$, the stability condition is $\gamma_0 < 1$, see (3). For all NWU distributed speed variations, see for example [18, Sec. 1.6], we have

$$\theta_j = \mathbb{E} \left[\min \left\{ \frac{Y_{1j}}{r_{1j}}, \dots, \frac{Y_{Nj}}{r_{Nj}} \right\} \right] \leq \frac{\mathbb{E}[S_j]}{\sum_{i=1}^N r_{ij}} = \hat{\theta}_j,$$

and hence $\gamma_0 \leq \hat{\gamma}_0$, which is a strict inequality in the case of a strictly NWU distribution. The remainder of the proof follows as a special case of Lemma 1 stated below, noting that $\hat{\gamma}_i \sum_{h: i \in s^h} \tilde{p}_h = \tilde{p}_i \tilde{\gamma}_i$ when $d = 1$. \square

Lemma 1 establishes a fundamental algebraic inequality for the exponential load values which will be of key importance throughout the remainder of this section as well.

Lemma 1. *For all choices of the probabilities \tilde{p}_k , $k = 1, \dots, K$, we have*

$$\hat{\gamma}_0 \leq \max_{i \in N} \hat{\gamma}_i \sum_{h: i \in s^h} \tilde{p}_h. \quad (17)$$

Proof. If we minimize the right-hand side in Equation (17) by setting

$$\hat{\gamma}_1 \sum_{h: 1 \in s^h} \tilde{p}_h = \dots = \hat{\gamma}_N \sum_{h: N \in s^h} \tilde{p}_h,$$

then it follows that this term is equal to

$$\frac{d}{\frac{1}{\hat{\gamma}_1} + \dots + \frac{1}{\hat{\gamma}_N}} = \frac{d \prod_{k=1}^N \hat{\gamma}_k}{\sum_{l=1}^N \prod_{k \neq l} \hat{\gamma}_k} = \hat{\gamma}_i \sum_{h: i \in s^h} \tilde{p}_h.$$

Note that for $d = 1$ we can get an explicit expression for the probabilities since we have a system of N equations with N unknowns, i.e., $\tilde{p}_i = \frac{\prod_{k=1, k \neq i}^N \hat{\gamma}_k}{\sum_{l=1}^N \prod_{k \neq l} \hat{\gamma}_k}$. For $d > 1$ we have K unknowns which makes the system of equations underdetermined. Thus, Equation (17) is equivalent to

$$\hat{\gamma}_0 \leq \frac{d}{\frac{1}{\hat{\gamma}_1} + \dots + \frac{1}{\hat{\gamma}_N}} \Leftrightarrow \frac{d}{\hat{\gamma}_0} \geq \frac{1}{\hat{\gamma}_1} + \dots + \frac{1}{\hat{\gamma}_N}. \quad (18)$$

We can rewrite the right-hand side of the expression to

$$\frac{1}{\hat{\gamma}_1} + \dots + \frac{1}{\hat{\gamma}_N} = \frac{1}{\sum_{h: 1 \in s^h} \frac{\tilde{p}_h}{\sum_{h^*: 1 \in s^{h^*}} \tilde{p}_{h^*}} \left(\frac{\hat{\gamma}_{01}}{x_{h1}} + \dots + \frac{\hat{\gamma}_{0M}}{x_{hM}} \right)} + \dots + \frac{1}{\sum_{h: N \in s^h} \frac{\tilde{p}_h}{\sum_{h^*: N \in s^{h^*}} \tilde{p}_{h^*}} \left(\frac{\hat{\gamma}_{01}}{x_{h1}} + \dots + \frac{\hat{\gamma}_{0M}}{x_{hM}} \right)},$$

where $\hat{\gamma}_{0j} = p_j \mathbb{E}[X_j] \frac{\mathbb{E}[S_j]}{\sum_{i=1}^N r_{ij}}$ and $x_{lj} = \frac{\sum_{i=1}^d r_{ij}}{\sum_{i=1}^N r_{ij}}$ for $l = 1, \dots, K$ and $j = 1, \dots, M$. The above expression is concave in x_{ij} when fixing the values $\sum_{i=1}^N r_{ij}$, $j = 1, \dots, M$, and is therefore maximized for $x_{l1} = \dots = x_{lM}$ for all $l = 1, \dots, K$, for which the expression is equal to $\frac{d}{\hat{\gamma}_0}$. \square

Extending Theorem 3 to all values of $1 \leq d \leq N$ is challenging. One of the key difficulties is that the various replicas do not necessarily start at the same time as a result of queuing, making it impossible to determine the exact load values when d is strictly between 1 and N . Establishing suitable lower bounds for the load values would provide a potential way to circumvent that issue. The next lemma presents a possible path in that direction by showing that the minimum expected aggregate weighted load is achieved when all replicas start at exactly the same time.

Lemma 2. *For any number of replicas and NWU distributed job sizes the expected aggregate weighted amount of time invested in the service of a job is minimized when all the replicas start at exactly the same time. Specifically, for each job type j ,*

$$\sum_{i \in s^h} r_{ij} \mathbb{E}[T_{ij}] \geq \sum_{i \in s^h} r_{ij} \theta_{ijh}.$$

Proof. Observe that for NWU distributions, Equation (11) changes to

$$\bar{F}_{T_{\text{awt},j}^I}(t) = \begin{cases} \bar{F}_{Y_{1j}}(t) \geq \bar{F}_{Y_{1j}}\left(\frac{r_{1j}t}{\sum_{i \in I} r_{ij}}\right) \cdots \bar{F}_{Y_{dj}}\left(\frac{r_{dj}t}{\sum_{i \in I} r_{ij}}\right) & \text{for } 0 < t < b_2, \\ \bar{F}_{Y_{1j}}\left(b_2 + \frac{r_{1j}(t-b_2)}{r_{1j}+r_{2j}}\right) \cdot \bar{F}_{Y_{2j}}\left(\frac{r_{2j}(t-b_2)}{r_{1j}+r_{2j}}\right) \\ \geq \bar{F}_{Y_{1j}}\left(\frac{r_{1j}t}{\sum_{i \in I} r_{ij}}\right) \cdots \bar{F}_{Y_{dj}}\left(\frac{r_{dj}t}{\sum_{i \in I} r_{ij}}\right) & \text{for } b_2 < t < \sum_{l=1}^2 (b_3 - b_l), \\ \vdots \\ \bar{F}_{Y_{1j}}\left(b_2 + \frac{r_{1j}2(b_3-b_2)}{r_{1j}+r_{2j}} + \cdots + \frac{r_{1j}(t-\sum_{l=1}^d (b_d-b_l))}{\sum_{i \in I} r_{ij}}\right) \\ \cdots \bar{F}_{Y_{dj}}\left(\frac{r_{dj}(t-\sum_{l=1}^d (b_d-b_l))}{\sum_{i \in I} r_{ij}}\right) \\ \geq \bar{F}_{Y_{1j}}\left(\frac{r_{1j}t}{\sum_{i \in I} r_{ij}}\right) \cdots \bar{F}_{Y_{dj}}\left(\frac{r_{dj}t}{\sum_{i \in I} r_{ij}}\right) & \text{for } \sum_{l=1}^d (b_d - b_l) < t. \end{cases}$$

By definition of NWU distributions,

$$\begin{aligned} \sum_{i \in I} r_{ij} \mathbb{E}[T_{ij}] &= \int_{t=0}^{\infty} \bar{F}_{T_{\text{awt},j}^I}(t) dt \geq \int_{t=0}^{\infty} \bar{F}_{Y_{1j}}\left(\frac{r_{1j}t}{\sum_{i \in I} r_{ij}}\right) \cdots \bar{F}_{Y_{dj}}\left(\frac{r_{dj}t}{\sum_{i \in I} r_{ij}}\right) dt \\ &= \sum_{i \in I} r_{ij} \mathbb{E}\left[\min\left\{\frac{Y_{1j}}{r_{1j}}, \dots, \frac{Y_{dj}}{r_{dj}}\right\}\right] = \sum_{i \in I} r_{ij} \theta_{ijh}, \end{aligned} \quad (19)$$

with $s^h = I$. This implies that the minimum expected aggregate weighted amount of time invested in the service of a job is achieved when all replicas start at exactly the same time. \square

Now observe that if $\mathbb{E}[T_{ij}] \geq \theta_{ijh}$ for all $i \in I = s^h$, then similar arguments as in Theorem 3 and Lemma 1 would yield that the stability region for $d = N$ is larger than for any $1 < d < N$ as well. Unfortunately, these detailed inequalities cannot be deduced from the aggregate weighted inequalities in (19) without further conditions. This leads to the next conjecture, which is also illustrated in Figure 2 for Weibull($\lambda_w = 1.128, k = 2$) (NBU), exponential and Weibull($\lambda_w = 0.5, k = 0.5$) (NWU) distributed speed variations.

Conjecture 1. For (strictly) NWU distributed speed variations and unknown job types, the load at server i of the system with $1 < d < N$ replicas, denoted by $\tilde{\gamma}_i$, is bounded by $\tilde{\gamma}_i \geq (>) \gamma_i$, for all $i = 1, \dots, N$.

In Figure 2 it can be seen that Conjecture 1 cannot be extended to NBU distributed speed variations, i.e., for the NBU Weibull distribution the stability condition of the original system (the expected latency is depicted with a solid lime green line) seems tighter than the stability condition in the system where all the d replicas were to start at the same time (dashed lime green line).

Conjecture 2. In the case of unknown job types, the stability region for $d = N$ when the job types are unknown is larger than or equal to (respectively, strictly larger than) the stability region for $1 \leq d < N$ with identical servers and NWU (respectively, strictly NWU) distributed speed variations where replicas are assigned to d servers selected uniformly at random (without replacement).

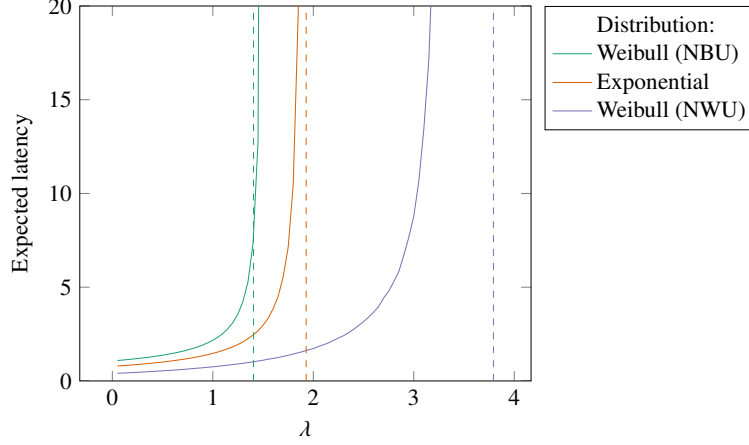


Figure 2: Expected latency for the scenario of Example 3 with $N = 3$ servers, $d = 2$ replicas and $r_{\text{slow}} = 0.5$ for various distributions for the speed variation with $\mathbb{E}[S_j] = 1$ and $\mathbb{E}[X_j] = 1$ for $j = 1, 2$. Assignment $\tilde{p}_k = \frac{1}{3}$, for $k = 1, 2, 3$, which is optimal. The dashed lines represent the stability condition for the load at server i equal to γ_i , for $i = 1, 2, 3$.

Conjecture 2 is supported by the observation that Conjecture 1 implies

$$\max_{i \in \mathcal{N}} \tilde{\gamma}_i \sum_{h: i \in s^h} \tilde{p}_h \geq \max_{i \in \mathcal{N}} \gamma_i \sum_{h: i \in s^h} \tilde{p}_h,$$

while Lemma 1 gives $\hat{\gamma}_0 \leq \max_{i \in \mathcal{N}} \hat{\gamma}_i \sum_{h: i \in s^h} \tilde{p}_h$. If Conjecture 1 is true, it would thus suffice to establish the equivalence relation

$$\frac{1}{\gamma_1} + \dots + \frac{1}{\gamma_N} \leq \frac{d}{\gamma_0} \Leftrightarrow \frac{1}{\hat{\gamma}_1} + \dots + \frac{1}{\hat{\gamma}_N} \leq \frac{d}{\hat{\gamma}_0}. \quad (20)$$

For identical servers with uniform selection of the servers we have that

$$\begin{aligned} \gamma &= \gamma_1 = \dots = \gamma_N = \sum_{j=1}^M p_j \mathbb{E}[X_j] \sum_{h: i \in s^h} \frac{\theta_{ijh}}{d}, \\ \hat{\gamma} &= \hat{\gamma}_1 = \dots = \hat{\gamma}_N = \sum_{j=1}^M p_j \mathbb{E}[X_j] \sum_{h: i \in s^h} \frac{\hat{\theta}_{ijh}}{d}. \end{aligned}$$

Substituting these in Equation (20) gives

$$\frac{N}{\gamma} \leq \frac{d}{\gamma_0} \Leftrightarrow \frac{N}{\hat{\gamma}} \leq \frac{d}{\hat{\gamma}_0},$$

or equivalently

$$\frac{\gamma}{N} \geq \frac{\gamma_0}{d} \Leftrightarrow \frac{\hat{\gamma}}{N} \geq \frac{\hat{\gamma}_0}{d}.$$

If we look at the difference, we get

$$\left(\frac{\hat{\gamma}}{N} - \frac{\gamma}{N} \right) - \left(\frac{\hat{\gamma}_0}{d} - \frac{\gamma_0}{d} \right). \quad (21)$$

Now observe that $\frac{\hat{\gamma}}{N} = \frac{\hat{\gamma}_0}{d}$ and therefore the term in Equation (21) simplifies to

$$\frac{\gamma_0}{d} - \frac{\gamma}{N}.$$

The last expression is negative since for NWU distributions, see for example [18, Sec. 1.6], we have

$$N\theta_j \leq d \sum_{h:i \in S^h} \frac{\theta_{ijh}}{d},$$

for all $j = 1, 2, \dots, M$.

In the next subsection we will show that even for NBU distributed speed variations full replication may give the largest stability region when job types cannot be observed. This demonstrates that unpredictability in speeds induced by uncertainty in job types can create a strong rationale for replication, even when the random speed variations do not. More specifically, we give examples illustrating that the number of replicas that yields the largest stability region depends on the server speeds.

4.1. Full replication may be best for NBU speed variations

In Section 3 we proved that the stability region is largest for $d = 1$ when the speed variations are NBU and job types can be distinguished. We now show that the complete opposite may be true when job types cannot be observed. More specifically, we will prove that even with NBU random speed variations in some scenarios full replication gives the largest stability region when the uncertainty in the systematic speed variations is sufficiently significant in some suitable sense.

Consider the scenario where job type j , for $j = 1, \dots, N$, is fast on server j , i.e., server speed r_{fast} , and slow on the other servers, i.e., server speed r_{slow} (see Example 3 with $N = 2$ servers, $r_{\text{fast}} = 1$ and $r_{\text{slow}} = x$). We refer to this scenario as the *FS* (Fast-Slow) scenario.

Theorem 4. *In the case of unknown job types, the stability region for $d = N$ is larger than the stability region for $d < N$ in the FS scenario with NBU distributed speed variations and static probabilistic assignment (which cannot depend on the job type) of the d replicas, when the ratio $\frac{r_{\text{slow}}}{r_{\text{fast}}} = x \downarrow 0$.*

Proof. Note that the stability region for $d = N$, given by Equation (3), does not depend on the value of x . Now, for $d < N$, the probability of assigning all replicas of a type- j job to slow servers is strictly larger than 0. For the expected service requirement of this job, denoted by $\mathbb{E}[B_j^*]$, it follows that $\mathbb{E}[B_j^*] \geq \frac{\mathbb{E}[X_j] \mathbb{E}[\min\{Y_{1j}, \dots, Y_{dj}\}]}{x}$. \square

In Figure 3 (right) it can be seen that for r_{slow} sufficiently large $d = 1$ gives the largest stability region in the special case of Weibull($\lambda_w = 1.128, k = 2$) distributed speed variations, which belongs to the class of NBU distributions. As stated in Theorem 4 for r_{slow} sufficiently small we observe that $d = N = 3$ gives the largest stability region.

In [2] a similar result for the processor-sharing discipline is proved. For this discipline it is shown that redundancy can improve the stability of the system with identical replicas if the servers are sufficiently heterogeneous when the assignment probabilities are restricted to be uniform.

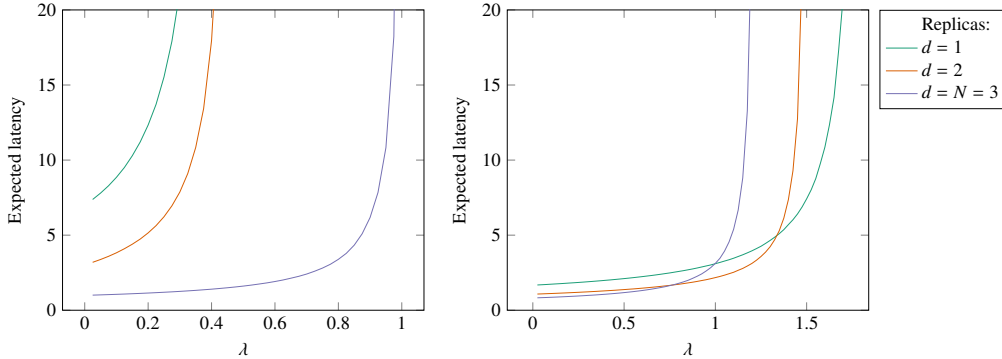


Figure 3: Expected latency for the scenario of Example 3 with $N = 3$ servers and $r_{\text{rslow}} = 0.1$ (left) and $r_{\text{rslow}} = 0.5$ (right), where the speed variations are Weibull($\lambda_w = 1.128, k = 2$) (NBU) distributed and $\mathbb{E}[X] = 1$.

5. Conclusion and suggestions for further research

We have proven that for c.o.c. redundancy scheduling with identical replicas, general job size distributions and suitable type-dependent assignment probabilities the stability region for $d = 1$ is strictly larger than the stability region for $d > 1$. Moreover, we established that the same statement holds in case of i.i.d. replicas and NBU distributed speed variations. For both identical and i.i.d. replicas a critical assumption is that the job types can be observed. In case of non-observable job types the stability region for $d = N$ is larger than or equal to the stability region for $d = 1$ when the speed variations are NWU distributed. Under the conjecture that the stability region increases in the latter case when all replicas start at the same time, we extended the above-mentioned statement, i.e., we showed that for identical servers the stability region for $d = N$ is larger than or equal to the stability region for all $d < N$.

In case the type identities of jobs are unknown, it may be possible to learn them, and for further research we intend to analyze the stability region when we are able to learn the job types; cf. [3] where a learning framework is proposed to answer these questions for a different model. Ultimately, we hope to quantify the performance loss in terms of the stability region when the job types are unknown beforehand and explore how decreasing the uncertainty about the job types can increase the stability region.

Acknowledgments

The work in this paper is supported by the Netherlands Organisation for Scientific Research (NWO) through Gravitation grant NETWORKS 024.002.003. The authors gratefully acknowledge several helpful discussions with Onno Boxma.

References

- [1] E. Anton, U. Ayesta, M. Jonckheere, and I.M. Verloop. On the stability of redundancy models. *ArXiv 1903.04414*, 2019.
- [2] E. Anton, U. Ayesta, M. Jonckheere, and I.M. Verloop. Improving the performance of heterogeneous data centers through redundancy. *ArXiv 2003.01394*, 2020.

- [3] K. Bimpikis and M.G. Markakis. Learning and hierarchies in service systems. *Management Science*, 65(3):1–18, 2018.
- [4] K. Gardner, M. Harchol-Balter, A. Scheller-Wolf, and B. Van Houdt. A better model for job redundancy: Decoupling server slowdown and job size. *IEEE ACM Transactions on Networking*, 25(6):3353–3367, 2017.
- [5] K. Gardner, M. Harchol-Balter, A. Scheller-Wolf, M. Velednitsky, and S. Zbarsky. Redundancy- d : The power of d choices for redundancy. *Operations Research*, 65(4):1078–1094, 2017.
- [6] J.M. Harrison and M.J. López. Heavy traffic resource pooling in parallel-server systems. *Queueing Systems*, 33(4):339–368, 1999.
- [7] T. Hellemans, T. Bodas, and B. Van Houdt. Performance analysis of workload dependent load balancing policies. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 3(2):1–35, 2019.
- [8] T. Hellemans and B. Van Houdt. Performance of redundancy(d) with identical/independent replicas. *ACM Transactions on Modeling and Performance Evaluation of Computing Systems*, 4(2):1–28, 2019.
- [9] G. Joshi. *Efficient Redundancy Techniques to Reduce Delay in Cloud Systems*. PhD thesis, Massachusetts Institute of Technology, 2016.
- [10] Y. Kim, R. Righter, and R. Wolff. Job replication on multiserver systems. *Advances in Applied Probability*, 41(2):546–575, 2009.
- [11] G. Koole and R. Righter. Resource allocation in grid computing. *Journal of Scheduling*, 11:163–173, 2008.
- [12] G. Mendelson. A lower bound on the stability region for redundancy- d with fifo service discipline. *ArXiv 2004.14793*, 2020.
- [13] F. Poloczek and F. Ciucu. Contrasting effects of replication in parallel systems: From overload to underload and back. *ACM SIGMETRICS Performance Evaluation Review*, 44(1):375–376, 2016.
- [14] Y. Raaijmakers, S.C. Borst, and O.J. Boxma. Delta probing policies for redundancy. *Performance Evaluation*, 127-128:21–35, 2018.
- [15] Y. Raaijmakers, S.C. Borst, and O.J. Boxma. Redundancy scheduling with scaled Bernoulli service requirements. *Queueing Systems*, 93(1-2):67–82, 2019.
- [16] Y. Raaijmakers, S.C. Borst, and O.J. Boxma. Stability of redundancy systems with processor sharing. *ArXiv 1912.00681*, 2019.
- [17] A.L. Stolyar. Optimal routing in output-queued flexible server systems. *Probability in the Engineering and Informational Sciences*, 19(2):141–189, 2005.
- [18] D. Stoyan. *Comparison Methods for Queues and Other Stochastic Models*. Chichester, Wiley, 1983. (edited with revisions by D.J. Daley).
- [19] Y. Sun, C.E. Koksal, and N.B. Shroff. On delay-optimal scheduling in queueing systems with replications computing. *ArXiv 1603.07322v8*, 2017.
- [20] D. Wang, G. Joshi, and G.W. Wornell. Efficient straggler replication in large-scale parallel computing. *ACM Transactions on Modeling and Performance Evaluation of Computing Systems*, 4(2):1–23, 2019.