Queueing Systems (2022) 100:197–199
https://doi.org/10.1007/s11134-022-09829-9

Correlation in redundancy systems

Kristen Gardner[1]

Received: 6 February 2022 / Accepted: 28 February 2022 / Published online: 5 May 2022

1 Introduction

In the past ten years, there has been considerable interest in the *redundancy* dispatching paradigm. The idea is that, upon a job's arrival to the system, the dispatcher creates multiple copies (also called *replicas*) of the job and sends the replicas to different servers. In the *cancel-on-complete* (c.o.c) version of redundancy, as soon as the first replica completes service on some server, all of the remaining replicas are removed from the system immediately. Empirical work has demonstrated that redundancy can significantly reduce both the mean and the tail of response time [5].

Much of the early literature assumes that a job's replicas either: (1) are independent and identically distributed, or (2) all have identical sizes. Unfortunately, neither model is realistic in practice.

Ideally, theoretical analyses would allow the replicas' service times to be *correlated* across servers. An accurate—and analytically tractable—model is necessary to avoid mistaken conclusions about the costs and benefits of redundancy.

2 Problem statement

The system consists of k homogeneous servers, each of which has its own dedicated queue. Jobs arrive to the system according to a Poisson process with rate $k\lambda$. The system employs the *redundancy-d* dispatching policy: when a job arrives, it is dispatched to d servers chosen uniformly at random, where we assume $d \ll k$ is a constant.

Upon completion of the first replica, all other replicas are cancelled immediately. The service times for a single job's replicas are *correlated* across servers. We identify five key questions, then discuss possible approaches to modeling correlation.

Q1: What is the system's mean response time and/or the response time distribution?
Q2: What value of d is optimal with respect to mean response time?
Q3: What is the stability region (i.e., for what values of d and λ is the system stable)?
Q4: What scheduling policy is optimal with respect to mean response time? How does the optimal scheduling policy depend on the correlation and service time distribution?

✉ Kristen Gardner
kgardner@amherst.edu

[1] Department of Computer Science, Amherst College, Amherst, MA, USA

Q5: Can we design new dispatching and/or scheduling policies that are analytically tractable and that have a provably maximal stability region?

3 Discussion

3.1 Prior work assuming independent or identical replicas

In the independent-replicas model (assuming exponentially distributed service times and first-come-first-served (FCFS) scheduling), exact closed-form expressions for mean response time and the distribution of response time have been derived under both redundancy-d and other job-server compatibility structures [2–4, 6, 8, 9] (**Q1**). The analysis requires the independence and exponentiality assumptions in order to obtain a product-form stationary distribution on the queue state; hence, the approach does not generalize to correlated replicas. Assuming service times follow a new-better-than-used distribution, choosing a higher value of d always improves mean response time [8, 12, 13]; the opposite is true for new-worse-than-used distributions (**Q2**). In line with this result, when service times are exponential redundancy does not change the stability region under FCFS, processor sharing (PS), or random order of service (ROS) scheduling [1, 3, 8] (**Q3**). For new-better-than-used service time distributions, the stability region *increases* with d [12, 15] (**Q3**). However, none of these scheduling policies are optimal with respect to mean response time: the Least Redundant First policy minimizes mean response time, assuming exponential service times, in systems with a nested structure [6]. The proof relies heavily on these assumptions, necessitating a different approach for the correlated-replicas model and redundancy-d dispatching.

In the identical-replicas model, an exact numerical approach has been developed to obtain the stationary workload distribution [11] (**Q1**). This approach was used to show that for bounded Pareto service times, it is optimal to set $1 < d < \infty$ [11] (**Q2**); a similar result is likely to hold in the correlated-replicas model. Stability results in the identical-replicas model are more realistic than those in the independent-replicas model: the stability region is reduced under both FCFS and PS [1], but, surprisingly, is unchanged under ROS (**Q3**). Similar results and proof techniques to those used in [1] may apply to ROS with correlated replicas.

3.2 Approaches to modeling correlation

One existing correlated-replicas model is the $S\&X$ model [7]. Here, $X_j \sim X$ refers to a job j's *inherent size*, and $S_i \sim S$ corresponds to the *slowdown* on server i. The time that a job j spends in service on server i is then $S_i \cdot X_j$, where X_j is drawn independently for each job and S_i is drawn independently for each server and for each job. Within the $S\&X$ model, policies such as redundant-to-idle-queue (RIQ) [7] and delta-probe [14] have been developed. While both policies achieve the maximal stability region (**Q5**), they are both designed such that only one replica for each job is served. This facilitates analysis, but sacrifices potential performance gains.

Later models building upon the $S\&X$ model offer more general correlation structures. In the model studied in [14, 16], the replicas' service times follow an arbitrary joint distribution, allowing for correlated and non-identically distributed server slowdowns. The stability region under PS scheduling has been derived within this model [16] (**Q3**). In an even more general model, the work added to each queue upon a job's arrival depends on the queue's current workload and some additional server-independent random variables including, but not limited to, the job's inherent size [10]. Within this model, the exact numerical approach introduced in [11] was extended to analyze a variety of load balancing policies, assuming FCFS scheduling [10] (**Q1**).

While the above results represent important steps towards understanding redundancy systems with correlated replica sizes, this topic remains a rich area for future study. Open problems include deriving closed-form exact or approximate expressions for mean response time under a variety of scheduling policies (**Q1**); developing analytical approaches for optimizing d (**Q2**); generalizing stability results to scheduling policies beyond PS (**Q3**); and identifying redundancy-based dispatching and scheduling policies that are optimal (**Q4**) or that perform well and are analytically tractable (**Q5**).

References

1. Anton, E., Ayesta, U., Jonckheere, M., Verloop, I.M.: On the stability of redundancy models. Oper. Res. **69**(5), 1540–1565 (2021)
2. Ayesta, U., Bodas, T., Verloop, I.M.: On a unifying product form framework for redundancy models. Perform. Eval. **127**, 93–119 (2018)
3. Bonald, T., Comte, C.: Balanced fair resource sharing in computer clusters. Perform. Eval. **116**, 70–83 (2017)
4. Bonald, T., Comte, C., Mathieu, F.: Performance of balanced fairness in resource pools: a recursive approach. Proc. ACM Meas. Anal. Comput. Syst. **1**(2), 1–25 (2017)
5. Dean, J., Barroso, L.A.: The tail at scale. Commun. ACM **56**(2), 74–80 (2013)
6. Gardner, K., Harchol-Balter, M., Hyytiä, E., Righter, R.: Scheduling for efficiency and fairness in systems with redundancy. Perform. Eval. **116**, 1–25 (2017)
7. Gardner, K., Harchol-Balter, M., Scheller-Wolf, A., Van Houdt, B.: A better model for job redundancy: decoupling server slowdown and job size. IEEE ACM Trans. Netw. **25**(6), 3353–3367 (2017)
8. Gardner, K., Harchol-Balter, M., Scheller-Wolf, A., Velednitsky, M., Zbarsky, S.: Redundancy-d: the power of d choices for redundancy. Oper. Res. **65**(4), 1078–1094 (2017)
9. Gardner, K., Zbarsky, S., Doroudi, S., Harchol-Balter, M., Hyytiä, E.: Reducing latency via redundant requests: exact analysis. ACM SIGMETRICS Perform. Eval. Rev. **43**(1), 347–360 (2015)
10. Hellemans, T., Bodas, T., Van Houdt, B.: Performance analysis of workload dependent load balancing policies. Proc. ACM Meas. Anal. Comput. Syst. **3**(2), 1–35 (2019)
11. Hellemans, T., Van Houdt, B.: Analysis of redundancy (d) with identical replicas. ACM SIGMETRICS Perform. Eval. Rev. **46**(3), 74–79 (2019)
12. Koole, G., Righter, R.: Resource allocation in grid computing. J. Sched. **11**(3), 163–173 (2008)
13. Raaijmakers, Y., Borst, S.: Achievable stability in redundancy systems. Proc. ACM Meas. Anal. Comput. Syst. **4**(3), 1–21 (2020)
14. Raaijmakers, Y., Borst, S., Boxma, O.: Delta probing policies for redundancy. Perform. Eval. **127**, 21–35 (2018)
15. Raaijmakers, Y., Borst, S., Boxma, O.: Redundancy scheduling with scaled Bernoulli service requirements. Queueing Syst. **93**(1), 67–82 (2019)
16. Raaijmakers, Y., Borst, S., Boxma, O.: Stability of redundancy systems with processor sharing. VALUETOOLS '20, New York, NY, USA. Association for Computing Machinery (2020)