The RESET and MARC Techniques, with Application to Multiserver-Job Analysis

Isaac Grosof
Carnegie Mellon University
Computer Science Dept.
Pittsburgh, PA, USA
igrosof@cs.cmu.edu

Yige Hong
Carnegie Mellon University
Computer Science Dept.
Pittsburgh, PA, USA
yigeh@cs.cmu.edu

Mor Harchol-Balter
Carnegie Mellon University
Computer Science Dept.
Pittsburgh, PA, USA
harchol@cs.cmu.edu

Alan Scheller-Wolf
Carnegie Mellon University
Tepper School of Business
Pittsburgh, PA, USA
awolf@andrew.cmu.edu

ABSTRACT

Multiserver-job (MSJ) systems, where jobs need to run concurrently across many servers, are increasingly common in practice. The default service ordering in many settings is First-Come First-Served (FCFS) service. Virtually all theoretical work on MSJ FCFS models focuses on characterizing the stability region, with almost nothing known about mean response time.

We derive the first explicit characterization of mean response time in the MSJ FCFS system. Our formula characterizes mean response time up to an additive constant, which becomes negligible as arrival rate approaches throughput, and allows for general phase-type job durations.

We derive our result by utilizing two key techniques: REduction to Saturated for Expected Time (RESET) and MArkovian Relative Completions (MARC).

Using our novel RESET technique, we reduce the problem of characterizing mean response time in the MSJ FCFS system to an M/M/1 with Markovian service rate (MMSR). The Markov chain controlling the service rate is based on the saturated system, a simpler closed system which is far more analytically tractable.

Unfortunately, the MMSR has no explicit characterization of mean response time. We therefore use our novel MARC technique to give the first explicit characterization of mean response time in the MMSR, again up to constant additive error. We specifically introduce the concept of "relative completions," which is the cornerstone of our MARC technique.

For more detail, see our full paper, [9]

1 Introduction

Multiserver queueing theory predominantly emphasizes models in which each job utilizes only one server (one-server-per-job models), such as the M/G/k. For decades, such models were popular in the study of computing systems, where they provided a faithful reflection of the behavior of such systems while remaining conducive to theoretical analysis. However, one-server-per-job models no longer reflect the behavior of many modern computing systems.

Multiserver jobs: In modern datacenters, such as those of Google, Amazon, and Microsoft, each job now requests many servers (cores, processors, etc.), which the job holds simultaneously. A job's "server need" refers to the number of servers requested by the job. In Google's recently published trace of its "Borg" computation cluster [7, 15], the server needs vary by a factor of 100,000 across jobs. Throughout this paper, we will focus on this "multiserver-job model" (MSJ), in which each job requests some number of servers, and concurrently occupies that many servers throughout its time in service (its "duration").

FCFS service: We specifically study the first-come first-served (FCFS) service ordering for the MSJ model, a natural and practical policy that is the default in both cloud computing [14] and supercomputing [3]. Currently, little is known about FCFS service in MSJ models.

Stability under FCFS: Even the stability region under FCFS scheduling is not generally understood. Some papers characterize the stability region under restrictive assumptions on the job duration distributions [13, 12, 8, 6]. A key technique in these papers is the *saturated system* approach [4, 1]. The saturated system is a closed system in which completions trigger new arrivals, so that the number of jobs in the system is always constant. We are the first to use the saturated system for analysis beyond characterizing the stability region.

Response time for FCFS: Even less is known about mean response time $\mathbb{E}[T]$ in MSJ FCFS systems: The only MSJ FCFS system in which mean response time has been analytically characterized is the simpler case of 2 servers and exponentially distributed durations [2]. Mean response time is much better understood under more complex scheduling policies such as ServerFilling and ServerFilling-SRPT [7, 10], but these policies require assumptions on both preemption and the server need distribution, and do not capture current practices, which emphasize nonpreemptive policies. Mean response time has not been characterized for any nonpreemptive policies [5]. Mean response time is also better understood in MSJ FCFS scaling regimes, where the number of servers and the arrival rate both grow asymptotically [16, 11]. We are the first to analyze MSJ FCFS mean response time under a fixed number of servers.

Why FCFS is hard to analyze: One source of difficulty in studying the FCFS policy is the lack of work conservation. In simpler one-server-per-job models, a work-conservation property holds: If enough jobs are present, no servers will be idle. The same is true under the ServerFilling and ServerFilling-SRPT policies [7], which focus on the power-of-two server-need setting. Each policy selects a subset of the jobs available, and places jobs from that subset into service in largest-server-need-first order. By doing so, and using the power-of-two assumption, these policies always fill all of the servers, whenever sufficiently many jobs are present, thereby achieving work conservation.

Work conservation is key to the mean response time analysis of those systems, as one can often reduce the analysis of response time to the analysis of work. In contrast, the multiserver-job model under FCFS service is not work conserving: a job must wait if it demands more servers than are currently available, leaving those servers idle.

First response time analysis: We derive the first characterization of mean response time in the MSJ FCFS system. We allow any phase-type duration distribution, and any correlated distribution of server need and duration. Our result holds at all loads up to an additive error, which becomes negligible as the arrival rate λ approaches λ^*, the threshold of stability.

Proof structure: We first use our RESET technique (REduction to Saturated for Expected Time) to reduce from the MSJ FCFS system to the At-least-k system. The At-least-k system is equivalent to a M/M/1 with Markovian service rate (MMSR), where the service rate is based on the saturated system. By "Markovian service rate", we refer to a system in which the completion rate fluctuates over time, driven by an external finite-state Markov chain. We next use our MARC technique (MArkovian Relative Completions) to prove the first characterization of mean response time in the MMSR.

Both steps are novel, hard, and of independent interest. We prove our MARC result first because it is a standalone result, characterizing mean response time for any MMSR system up to an additive constant. We then prove Theorem 2.1, our characterization of mean response time in the MSJ FCFS system, by layering our RESET technique on top of MARC. Theorem 2.1 characterizes mean response time in terms of several quantities that can be characterized explicitly and in closed form via a straightforward analysis of the saturated system.

2 Main Result

Theorem 2.1. *In the multiserver-job system, the expected response time in steady state satisfies*

$$\mathbb{E}[T^{\mathrm{MSJ}}] = \frac{1}{\lambda^*} \frac{1 + \Delta_{\mathrm{Sat}}(Y_d^{\mathrm{Sat}}, Y^{\mathrm{Sat}})}{1 - \lambda/\lambda^*} + O_\lambda(1). \qquad (1)$$

3 References

[1] F. Baccelli and S. Foss. On the saturation rule for the stability of queues. *Journal of Applied Probability*, 32(2):494–507, 1995.

[2] P. H. Brill and L. Green. Queues in which customers receive simultaneous service from a random number of servers: A system point approach. *Management Science*, 30(1):51–68, 1984.

[3] D. G. Feitelson, L. Rudolph, and U. Schwiegelshohn. Parallel job scheduling—a status report. In *Workshop on Job Scheduling Strategies for Parallel Processing*, pages 1–16, New York, NY, USA, 2004. Springer.

[4] S. Foss and T. Konstantopoulos. An overview of some stochastic stability methods. *Journal of the Operations Research Society of Japan*, 47(4):275–303, 2004.

[5] I. Grosof and M. Harchol-Balter. Invited paper: ServerFilling: A better approach to packing multiserver jobs. In *Proceedings of the 5th Workshop on Advanced Tools, Programming Languages, and PLatforms for Implementing and Evaluating Algorithms for Distributed Systems*, ApPLIED 2023, New York, NY, USA, 2023. Association for Computing Machinery.

[6] I. Grosof, M. Harchol-Balter, and A. Scheller-Wolf. Stability for two-class multiserver-job systems. *arXiv preprint arXiv:2010.00631*, 2020.

[7] I. Grosof, M. Harchol-Balter, and A. Scheller-Wolf. WCFS: A new framework for analyzing multiserver systems. *Queueing Systems*, 2022.

[8] I. Grosof, M. Harchol-Balter, and A. Scheller-Wolf. New stability results for multiserver-job models via product-form saturated systems. *MAthematical performance Modeling and Analysis (MAMA)*, 4(6):1, 2023.

[9] I. Grosof, Y. Hong, M. Harchol-Balter, and A. Scheller-Wolf. The reset and marc techniques, with application to multiserver-job analysis. *Performance Evaluation*, 2023.

[10] I. Grosof, Z. Scully, M. Harchol-Balter, and A. Scheller-Wolf. Optimal scheduling in the multiserver-job model under heavy traffic. *Proc. ACM Meas. Anal. Comput. Syst.*, 6(3), dec 2022.

[11] Y. Hong. Sharp zero-queueing bounds for multi-server jobs. *SIGMETRICS Perform. Eval. Rev.*, 49(2):66–68, jan 2022.

[12] A. Rumyantsev, R. Basmadjian, S. Astafiev, and A. Golovin. Three-level modeling of a speed-scaling supercomputer. *Annals of Operations Research*, pages 1–29, 2022.

[13] A. Rumyantsev and E. Morozov. Stability criterion of a multiserver model with simultaneous service. *Annals of Operations Research*, 252(1):29–39, 2017.

[14] L. Sliwko. A taxonomy of schedulers–operating systems, clusters and big data frameworks. *Global Journal of Computer Science and Technology*, 2019.

[15] M. Tirmazi, A. Barker, N. Deng, M. E. Haque, Z. G. Qin, S. Hand, M. Harchol-Balter, and J. Wilkes. Borg: The next generation. In *Proceedings of the Fifteenth European Conference on Computer Systems*, EuroSys '20, New York, NY, USA, 2020. Association for Computing Machinery.

[16] W. Wang, Q. Xie, and M. Harchol-Balter. Zero queueing for multi-server jobs. In *Abstract Proceedings of the 2021 ACM SIGMETRICS / International Conference on Measurement and Modeling of Computer Systems*, SIGMETRICS '21, page 13–14, New York, NY, USA, 2021. Association for Computing Machinery.