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

Isaac Grosof[a], Yige Hong[a], Mor Harchol-Balter[a], Alan Scheller-Wolf[a]

[a] *Carnegie Mellon University, 5000 Forbes Ave, Pittsburgh, 15213, PA, USA*
{*igrosof, yigeh, harchol, 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.

Keywords: queueing, response time, RESET, MARC, multiserver, MSJ, markovian service rate, heavy traffic

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 [17, 46], 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 [9, 26, 43] and supercomputing [10, 23]. 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 [1, 18, 32, 41, 42]. A key technique in these papers is the *saturated system* approach [2, 12]. The saturated

Figure 1: The structure of our main results: RESET (Theorem 4.2) and MARC (Theorem 4.1).

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 [3, 11]. Mean response time is much better understood under more complex scheduling policies such as ServerFilling and ServerFilling-SRPT [17, 19], 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 is also better understood in MSJ FCFS scaling regimes, where the number of servers and the arrival rate both grow asymptotically [22, 49]. 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 [17], 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 illustrate the structure of our proof in Fig. 1. 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 (see Section 3.3). The At-least-k system is equivalent to a M/M/1 with Markovian service rate (MMSR) (see Section 3.2), 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 Theorem 4.1, 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 4.2, our characterization of mean response time in the MSJ FCFS system, by layering our RESET technique on top of MARC. Theorem 4.2 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. We walk through a specific example of using our result to explicitly characterize mean response time in Appendix C.

Breadth of the RESET technique: Our RESET technique is very broad, and applies to a variety of generalizations of the MSJ model and beyond (See Section 7). For instance, RESET can handle cases

where a job's server need varies throughout its time in service, and where the service rates at the servers can depend on the job. Finally, we can analyze scheduling policies that are close to FCFS but allow limited reordering, such as some backfilling policies.

Breadth of the MARC technique: Our MARC technique is also very broad, and applies to any MMSR system. For example, we can handle systems in which machine breakdowns lead to reduced service rate, or where servers are taken away by higher-priority customers.

This paper is organized as follows:

- Section 2: We discuss prior work on the MSJ model.
- Section 3: We define the MSJ model, the MMSR, the saturated system, relative completions, and related concepts.
- Section 4: We state our main results, and walk through an example of applying our results to a specific MSJ FCFS system.
- Section 5: We characterize mean response time in the MMSR using our MARC technique.
- Section 6: We build upon Section 5 to characterize MSJ FCFS mean response time using our RESET technique.
- Section 7: Our results apply to a very broad class of models which we call "finite skip" models, and which we define in this section.
- Section 8: We empirically validate our theoretical results.

2. Prior work

The bulk of the prior work we discuss is in Section 2.1, which focuses on specific results in the multiserver-job model. In Section 2.2, we briefly discuss prior work on the saturated system, an important tool in our analysis. Finally, in Section 2.3, we discuss prior work on the M/M/1 with Markovian service rate.

2.1. Multiserver-job model

Theoretical results in the multiserver-job model are limited. We first discuss the primary setting of this paper: a fixed number of servers and FCFS service.

2.1.1. Fixed number of servers, FCFS service

In this setting, most results focus on characterizing the stability region. Rumyantsev and Morozov characterize stability for an MSJ system with an arbitrary distribution of server needs, where the duration distribution is exponential and independent of server need [42]. This result can implicitly be seen as solving the saturated system, which has a product-form stationary distribution in this setting. A setting with two job classes, each with distinct server needs and exponential duration distributions has also been considered [16, 40]. In this setting, the saturated system was also proven to have a product-form stationary distribution, which was also used to characterize the stability region.

The only setting in which mean response time $\mathbb{E}[T]$ is known is in the case of $k = 2$ servers and exponential duration independent of server need [3, 11]. In this setting, the exact stationary distribution is known. Mean response time is open in all other settings, including whenever $k > 2$.

2.1.2. Advanced scheduling policies

More advanced scheduling policies for the MSJ system have been investigated, in order to analyze and optimize the stability region and mean response time.

The MaxWeight policy was proven to achieve optimal stability region in the MSJ setting [27]. However, its implementation requires solving an NP-hard optimization problem upon every transition, and it performs frequent preemption. It is also too complex for response time analysis to be tractable. The Randomized Timers policy achieves optimal throughput with no preemption [13, 39], but has very poor empirical mean response time, and no response time analysis.

In some settings, it is possible for a scheduling policy to ensure that all servers are busy whenever there is enough work in the system, which we call "work conservation." Work conservation enables the optimal

stability region to be achieved and mean response time to be characterized. Two examples are ServerFilling and ServerFilling-SRPT scheduling policies [17, 19]. However, the work-conservation-based techniques used in these papers cannot be used to analyze non-work-conserving policies such as FCFS.

2.1.3. Scaling number of servers

The MSJ FCFS model has also been studied in settings where the number of servers, the arrival rate, and the server need distribution all grow in unison to infinity. Analogues of the Halfin-Whitt and non-diminishing-slowdown regimes have been established, proving bounds on the probability of queueing and mean waiting time [22, 49]. These results focus on settings where an *approximate* work conservation property holds, and there is enough excess capacity that this approximate work conservation is sufficient to determine the first-order behavior of the system. These results do not apply to the $\lambda \to \lambda^*$ limit.

2.2. Prior work on the saturated system

The *saturated system* is a queueing system which is used as analysis tool to understand the behavior of an underlying non-saturated queueing system [2, 12]. Baccelli and Foss state that it is a "folk theorem" that the threshold of the stability region of the original open queueing system is equivalent to the completion rate of the saturated system: If the completion rate of the saturated system is μ, then the original system is stable for arrival rate λ if and only if $\lambda < \lambda^* = \mu$ [2]. Baccelli and Foss give sufficient conditions for this folk theorem, known as the "saturation rule," to hold rigorously. These conditions are mild, and are easily shown to hold for the MSJ FCFS system. The strongest stability results in the MSJ FCFS system have either been proven by characterizing the steady state of the saturated system, or are equivalent to such a characterization [16, 18, 42].

Our novel contribution is characterizing the *mean response time* behavior of an original system by reducing its analysis to the analysis of a saturated system. All previous uses of the saturated system focused on characterizing stability. Specifically, our main theorem, Theorem 4.2, characterizes mean response time in terms of $\Delta_{\text{Sat}}(y)$, λ^*, and Y_d^{Sat}. These functions and random variables are specific to the saturated system. They are defined in Section 3, and can be calculated in closed-form by analyzing the saturated system, as we walk through in Appendix C.

2.3. M/M/1 with Markovian Service Rate

The M/M/1 with Markovian service rate (MMSR) has been extensively studied since the 50's, often alongside Markovian arrival rates [5, 6, 20, 24, 29, 33]. A variety of mathematical tools have been applied to the MMSR, including generating function methods, matrix-analytic and matrix-geometric methods, and spectral expansion methods [5, 6, 25, 33]. However, these methods primarily result in *numerical results*, rather than theoretical insights [6, 31].

More is known for special cases of the MMSR system [7, 38]. For instance, the case where arrival rates alternate between a high and low completion rate at some frequency has received specific study. In this case, the generating function can be explicitly solved as the root of a cubic equation [50], but the resulting expression is too complex for analytical insights. In this simplified setting, scaling results [34–36, 47] and monotonicity results [20] have been derived, but those results do not extend to more complex MMSR systems.

By contrast, our MARC technique provides the first explicit characterization of mean response time for the general MMSR system, up to an additive constant.

2.4. Drift method and MARC

The drift method is a popular method for steady-state analysis of queueing models (see, e.g., [8, 22, 28, 49]). In the drift method, one takes a suitable *test function* (also known as a Lyapunov function) of the system state and computes its instantaneous rate of change starting from each state under the transition dynamics, which is called the drift. The drift can be formally calculated using the *instantaneous generator*, defined in Section 3.9. One then utilizes the fact that the drift of any test function has zero steady-state expectation (Lemma 3.2) to characterize system behavior in steady state, through metrics such as mean

Abbreviation	Meaning	Definition
MSJ	Multiserver-job	Section 3.1
FCFS	First-come first-served	Section 3.1
MMSR	M/M/1 with Markovian service rate	Section 3.2
Ak	At-least-k system	Section 3.3
Sat	Saturated system	Section 3.5
SSS	Simplified saturated system	Section 3.11, Appendix B
MARC	Markovian relative completions	Section 5
RESET	Reduction to saturated for expected time	Section 6

Table 1: Table of abbreviations

queue length. Through more specialized choices of test function, stronger results such as State Space Collapse can also be proven.

In prior work which analyzes the mean queue length, the test function is usually a quadratic function of the queue length. For instance, when analyzing the MaxWeight policy in the switch setting, an appropriate test function is $\sum_i q_i^2$, where q_i is the number of jobs present of each class i [44]. For such a test function to provide useful information about the expected queue length, the system must achieve a constant work completion rate whenever there are enough jobs in the system. This constant work completion rate ensures that the test function's drift depends linearly on the queue length, allowing the mean queue length to be characterized. However, in our MSJ system, the work completion rate is variable regardless of the number of jobs in the system, because servers may always be left empty if a job in the queue requires more servers than are available. As a result, the standard test functions for the drift method do not provide useful information about the MSJ system.

Our innovation is to construct a novel test function that combines the queue length q and a new quantity called *relative completions*, defined in Section 3.8. Our use of relative completions allows us to ensure that the test functions f_Δ and f_Δ^{MSJ}, defined in Definitions 5.1 and 6.1, have drift which depend linearly on the queue length. As a result, we can apply the drift method with our novel test functions to characterize mean queue length in the MSJ system, and hence characterize mean response time.

We call this technique the MArkovian Relative Completions (MARC) technique: using relative completions to define a test function for the drift method, to apply the drift method to systems with variable work-completion rate.

3. Model

In this section, we introduce five queueing models: the multiserver-job (MSJ) model, the M/M/1 with Markovian service rate (MMSR), the At-least-k (Ak) model, the saturated system, and the simplified saturated system (SSS). The MSJ is the main focus of this paper. Our RESET technique reduces its analysis to analyzing the Ak system. The Ak system is equivalent to a MMSR system whose completion process is controlled by the saturated system. Our MARC technique allows us to analyze this MMSR system. The SSS is a simpler equivalent of the saturated system. We also introduce the concepts of relative completions and the generator approach, which are key to our analysis.

Table 1 describes each of the abbreviations used in this paper.

3.1. Multiserver-job Model

The MSJ model is a queueing model in which each job requests an integer number of servers, the *server need*, for some duration of time, the *service duration*. Each job requires concurrent service on all of its servers throughout its duration. Let k denote the total number of servers in the system.

We assume that each job's server need and service duration are drawn i.i.d. from some joint distribution. The duration distribution is phase type, and it may depend on the job's server need. This assumption can likely be generalized, which we leave to future work. We assume a Poisson(λ) arrival process.

We focus on the first-come first-served (FCFS) service discipline. Our RESET technique also applies to many other scheduling policies, as we discuss in Section 7. Under FCFS, jobs are placed into service, one by one, in arrival order, as long as the total server need of the jobs in service is at most k. If a job is reached whose server need would push the total over k, that job does not receive service until sufficient completions occur. We consider head-of-the-line blocking, so no subsequent jobs in arrival order receive service. It has been shown that in the MSJ FCFS setting, there exists a threshold λ^*, such that the system is stable if and only if $\lambda < \lambda^*$ [2, 12]. We assume that $\lambda < \lambda^*$.

Note that the only jobs eligible for service are the k oldest jobs in arrival order. We conceptually divide the system into two parts: the *front* and the *back*. When the total number of jobs in the system is at least k, the front consists of the k-oldest jobs in the arrival order; otherwise, the front consists of all jobs in the system. The back consists of all jobs that are not in the front. Note that all of the jobs which are in service must be in the front, because at most k jobs can be in service at a time, and service proceeds in strict FCFS order. The front may also contain some jobs which are not in service, whenever less than k jobs are in service. All of the jobs in the back are not in service.

3.2. M/M/1 with Markovian Service Rate

The MMSR-π system is a queueing system where jobs arrive to the system according to a Poisson process, and complete at a variable rate, where the completions are determined by the transitions of a finite-state Markov chain π. We refer to π as the "service process". When a job arrives, it stays in the queue until it reaches the head of the line, entering service. The job then completes when π next undergoes a transition associated with a completion. Jobs are identical until they reach service. The service process π is unaffected by the number of jobs in the queue.

3.3. At-least-k System

To connect the MSJ FCFS and MMSR systems, we define two systems: the "At-least-k" (Ak) system, and the "saturated system" in Section 3.5. The Ak model mimics the MSJ model, except that the Ak system always has at least k jobs present. Specifically, in addition to the primary Poisson(λ) arrival process, whenever there are exactly k jobs in the system, and a job completes, a new job immediately arrives. The server need and service duration of this job are sampled i.i.d. from the same distribution as the primary arrivals. Due to these extra arrivals, the front of the Ak system always has exactly k jobs present.

Intuitively, the Ak system should have about k more jobs present in steady state than the MSJ system. We thus expect the Ak and MSJ systems to have the same asymptotic mean response time, up to an $O_\lambda(1)$ term. We make this intuition rigorous by using our RESET technique to prove Theorem 4.2.

3.4. Running Example

Throughout this section, we will use a running example to clarify notation and concepts. Consider a MSJ setting with $k = 2$ servers, and two classes of jobs: 2/3 of jobs have server need 1 and duration $Exp(1)$, and the other 1/3 of jobs have server need 2 and duration $Exp(1/2)$.

3.5. Saturated System

The saturated system is a closed multiserver-job system, where completions trigger new arrivals.[1] Jobs are served according to the same FCFS service discipline. There are always exactly k jobs in the system. Whenever a job completes, a new job with i.i.d. server need and service duration is sampled. The state descriptor is just an ordered list of exactly k jobs.

In our running example with $k = 2$ servers, the state space of the saturated system consists of all orderings of 2 jobs:

$$\mathbb{Y}^{\text{Sat}} = \{[1,1],[1,2],[2,1],[2,2]\}.$$

The leftmost entry in each of the lists is the oldest job in FCFS order. In state $[1,2]$, a 1-server job is in service and a 2-server job is not in service, while in state $[2,1]$, a 2-server job is in service and a 1-server job is not in service.

[1] Baccelli and Foss [2] consider a system with infinitely many jobs not in service, which is equivalent to our closed system.

3.6. Equivalence between MMSR-Sat and At-least-k

Now we are ready to connect the MMSR and At-least-k (Ak) systems. Consider the subsystem consisting only of the front of the Ak system, i.e., the k oldest jobs in the Ak system. This subsystem is stochastically identical to the saturated system. Whenever a job completes at the front of the Ak system, a new job enters the front, either from the back (i.e. the jobs not in the front) or from the auxiliary arrival process, if the back is empty. This matches the saturated system's completion-triggered arrival process.

As a result, the Ak system is stochastically equal to an MMSR-π system whose service process π is identical to the saturated system. We refer to this system as the "MMSR-Sat" system. To clarify this equivalency, assume the Ak system starts in a certain front state y with an empty back. Then equivalently the MMSR-Sat system starts empty, with its service process in state y. If a job in the Ak system completes its service, a new job is generated, and the same transition occurs in the service process in the MMSR-Sat system. Similarly, assume a job arrives to the Ak system and enters the back. At the same time, a job arrives in the MMSR-Sat system and enters the queue. Through this mapping, the two systems are sample-path equivalent.

The above arguments are summarized in Lemma 3.1 below.

Lemma 3.1. *There exists a coupling under which the front of the Ak system is identical to the Sat system, and the back of the Ak system is identical to the queue of the MMSR-Sat system.*

3.7. Notation

MSJ system state: A state of the MSJ system consists of a front state, y^{MSJ}, and a number of jobs in the back q^{MSJ}. A job state consists of a server need and a phase of its phase-type duration. The front state y^{MSJ} is a list of up to k job states. If $q^{\text{MSJ}} > 0$, then y^{MSJ} must consist of exactly k job states, while if $q^{\text{MSJ}} = 0$, y^{MSJ} may consist of anywhere from 0 to k job states. Let $\mathbb{Y}^{\text{MSJ}}$ denote the set of all possible front states y^{MSJ} of the MSJ system. For instance, in our running example, $\mathbb{Y}^{\text{MSJ}} = \{[], [1], [2], [1,1], [1,2], [2,1], [2,2]\}$. Note that in the first three states, the back must be empty, so q^{MSJ} must equal 0.

MMSR system state: In the MMSR system, let π denote the Markov chain that modulates the service rate. As a superscript, it signifies "the MMSR system controlled by the Markov chain π." A state of the MMSR-π system consists of a pair (q^π, y^π). The queue length q^π is a nonnegative integer. The state y^π is a state of the service process π, and $\mathbb{Y}^\pi$ is the state space of π.

Because the MMSR-Sat system is stochastically equal to the Ak system, with the MMSR-Sat system's queue length equal to the Ak system's back length, we use the superscripts $^{\text{Sat}}$ and $^{\text{Ak}}$ interchangeably. A state of the Ak system is a pair $(q^{\text{Ak}}, y^{\text{Ak}})$. In contrast to the MSJ system, y^{Ak} always consists of exactly k job states. In particular, $\mathbb{Y}^{\text{Ak}} \subset \mathbb{Y}^{\text{MSJ}}$.

MMSR service process: When the service process π transitions from state y to y', there are two possibilities: Either a completion occurs, which we write as $a = 1$, or no completion occurs, which we write as $a = 0$. We therefore define $\mu^\pi_{y,y',a}$ to denote the system's transition rate from front state y to front state y', accompanied by a completions, where $a \in \{0, 1\}$. For instance, in our running example $\mu^{\text{Sat}}_{[1,1],[1,2],1} = 2/3$. Let the total completion rate from state y be denoted by $\mu^\pi_{y,\cdot,1} = \sum_{y'} \mu^\pi_{y,y',1}$. For instance, in our running example $\mu^{\text{Sat}}_{[1,1],\cdot,1} = 2$.

MSJ service transitions: Let $\mu^{\text{MSJ}}_{y,y',a,b}$ denote a transition rate in the Multiserver-job system, where y, y', and a have the same meaning as in $\mu^{\text{Ak}}_{y,y',a}$. Let $b = \mathbb{1}_{q>0}$ denote whether this transition is associated with an empty back ($b = 0$), or an occupied back ($b = 1$). Note that if $y \notin \mathbb{Y}^{\text{Ak}}$, then $b = 0$ for all nonzero $\mu^{\text{MSJ}}_{y,y',a,b}$, while if $y \in \mathbb{Y}^{\text{Ak}}$, then both values of b are possible. Note that $\forall y \in \mathbb{Y}^{\text{Ak}}, \mu^{\text{MSJ}}_{y,y',a,1} = \mu^{\text{Ak}}_{y,y',a}$.

If a job arrives to the MSJ system and finds that the front state y has fewer than k jobs ($y \notin \mathbb{Y}^{\text{Ak}}$), a fresh job state is sampled and appended to y. Let S be a random variable denoting a fresh job state, let i be a particular fresh job state, let p_i be the probability $\mathbb{P}(S = i)$, and let $y \cdot i$ be the new front state with a job in state i appended. For instance, in the running example, $p_1 = 2/3, p_2 = 1/3$.

Steady-state notation: We will study the time-average steady states of each of these systems, which we write $(Q^{\text{MSJ}}, Y^{\text{MSJ}})$, (Q^π, Y^π), etc. Let Y^π_d denote the departure-average steady state of the MMSR

service process π: the steady-state distribution of the embedded DTMC which samples states after each departure from π.

Let X^π denote the long-term throughput of the service process π. Let λ^*_π denote the threshold of the stability region of the MMSR-π system. The MMSR-π system is stable if and only if $\lambda < \lambda^*$. Note that $X^\pi = \lambda^*_\pi$ by prior results relating the saturated system to the stability region of the original system [2, 12]. In particular, $X^{\text{Sat}} = \lambda^*_{\text{Sat}} = \lambda^*$, where λ^* denotes the threshold of the stability region of the MSJ FCFS system. We will typically write λ^* to avoid confusion between X^{Sat} and a random variable.

A concrete example of this notation is provided in Section 4.1.

3.8. Relative completions

Key to our MARC technique is the novel idea of *relative completions*, which we define for a general MMSR-π system. Let y_1 and y_2 be two states of the service process π. The difference in relative completions between two states y_1 and y_2 is the long-term difference in expected completions between an instance of the service process starting in state y_1 and one starting in y_2. Specifically, let $C_\pi(y, t)$ denote the number of completions up to time t of the service process of π initialized in state y at time $t = 0$. Then let $\Delta_\pi(y_1, y_2)$ denote the relative completions between states y_1 and y_2:

$$\Delta_\pi(y_1, y_2) = \lim_{t\to\infty} \mathbb{E}[C_\pi(y_1, t) - C_\pi(y_2, t)].$$

We prove that $\Delta_\pi(y_1, y_2)$ always exists and is always finite in Lemma A.1. We also allow y_1 and/or y_2 to be distributions over states, rather than single states. Specifically, we will often focus on the case where y_2, rather than being a single state, is the steady state distribution Y^π. In this case, note that $\mathbb{E}[C_\pi(Y^\pi, t)] = X^\pi t = \lambda^*_\pi t$. When it is clear from context, we write $\Delta_\pi(y)$ to denote $\Delta_\pi(y, Y^\pi)$. The relative completions formula for this case simplifies:

$$\Delta_\pi(y) = \Delta_\pi(y, Y^\pi) = \lim_{t\to\infty} \mathbb{E}[C_\pi(y, t)] - \lambda^*_\pi t. \tag{1}$$

The relative completions function $\Delta_\pi(y)$ can be seen as the relative value of a given state y under a Markov reward process whose state is a state of the service process π and whose reward is the instantaneous completion rate in a given state y.

3.9. Generator

We also make use of the *instantaneous generator* of each of our queueing systems, which is the stochastic equivalent of the derivative operator. The instantaneous generator is an operator which takes a function from system states to real values, and returns a function from system states to real values. The latter function is known as the *drift* of the original function.

The generator operator is specific to a given Markov chain. Let η be a Markov chain, and let G^η denote the generator operator for η, which is defined as follows:

For any real-valued function of the state of η, $f(q, y)$,

$$G^\eta \circ f(q, y) := \lim_{t\to 0} \frac{1}{t} \mathbb{E}[f(Q^\eta(t), Y^\eta(t)) - f(q, y) | Q^\eta(0) = q, Y^\eta(0) = y].$$

Importantly, the expected value of the generator in steady state is zero:

Lemma 3.2. *Let f be a real-valued function of the state of a Markov chain η. Assume that the transition rates of the Markov chain η are uniformly bounded, and $\mathbb{E}[f(Q^\eta, Y^\eta)] < \infty$. Then*

$$\mathbb{E}_{(q,y)\sim(Q^\eta, Y^\eta)}[G^\eta \circ f(q, y)] = 0. \tag{2}$$

Proof. Follows from [14, Proposition 3]. Discussion deferred to Appendix A. □

We show in Appendix A that (2) holds for the MSJ, MMSR, At-least-k, and Saturated systems, for any $f(q, y)$ with polynomial dependence on q.

3.10. Asymptotic notation

We use the notation $O_\lambda(f(\lambda))$ to represent a function $g(\lambda)$ such that

$$\exists \text{ a constant } M \text{ such that } |g(\lambda)| \leq M|f(\lambda)| \quad \forall \lambda, 0 < \lambda < \lambda^*.$$

3.11. Simplified saturated system

While the saturated system is a finite-state system, it can have a very large number of possible states. However, many of the states have identical behavior, and can be combined to reduce the state space. For instance, in our running example, the states $[2, 1]$ and $[2, 2]$ are nearly identical: in both states just a 2-server job is in service. We therefore simplify the system by combining the two states into the state $[2]$, and delaying sampling the next job until needed.

We refer to the resulting system as the "simplified saturated system" (SSS), in contrast to the original saturated system, which is the focus of the bulk of this paper. SSS is equivalent to the original saturated system, in the sense of 3.3 stated below.

Lemma 3.3. *There exists a coupling under which the main saturated system and simplified saturated system have identical completions.*

The full definition of the SSS, and the proof of the equivalence of SSS to the original saturated system, are in Appendix B.

The reduction in state space from the SSS can be dramatic. For instance, consider a system where $k = 30$, jobs have server needs 3 or 10, and jobs have exponential duration. The original saturated system has 2^{30} states, while the SSS has just 13 states. We discuss this reduction further in Appendix B.

4. Results

In this paper, we give the first analysis of mean response time in the MSJ FCFS system. To do so, we reduce the problem to the analysis of mean response time in an M/M/1 with Markovian service rate (MMSR) in which the saturated system controls the service process (i.e. the At-least-k system). We call this reduction the RESET technique. Before applying the RESET technique, we start by analyzing the general MMSR-π system.

We prove the first explicit characterization of mean response time in the MMSR. To do so, we use our MARC technique, which is based on the novel concept of *relative completions* (See Section 3.8).

Theorem 4.1 (Mean response time asymptotics of MMSR systems)**.** *In the MMSR-π system, the expected response time in steady state satisfies*

$$\mathbb{E}[T^\pi] = \frac{1}{\lambda^*_\pi} \frac{1 + \Delta_\pi(Y^\pi_d, Y^\pi)}{1 - \lambda/\lambda^*_\pi} + O_\lambda(1), \tag{3}$$

where Δ_π is the relative completions function defined in Section 3.8:

$$\Delta_\pi(Y^\pi_d, Y^\pi) := \lim_{t\to\infty} \mathbb{E}[C_\pi(Y^\pi_d, t)] - \lambda^*_\pi t.$$

To understand (3), first note that the dominant term has order $\Theta(\frac{1}{1-\lambda/\lambda^*_\pi})$. This is the equivalent of the $\Theta(\frac{1}{1-\rho})$ behavior seen in simpler systems such as the M/G/1/FCFS. Next, to understand the numerator, examine the $\Delta_\pi(Y^\pi_d, Y^\pi)$ term. Δ_π, the relative completions function, smooths out the irregularities in completion times, so that the function $q - \Delta_\pi(y)$ has a constant negative drift. Δ_π is the analog of the remaining size of the job in service in the M/G/1. When a generic job arrives, it sees a time-average state of the service process, namely Y^π. When it departs, it leaves behind a departure-average state of the service process, namely Y^π_d. The difference in relative completions between these states captures the asymptotic behavior of mean response time. The overall numerator, $1 + \Delta_\pi(Y^\pi_d, Y^\pi)$, is analogous to the $\mathbb{E}[S_e]$ term in

the M/G/1/FCFS mean response time formula. We walk through calculating $\Delta_\pi(y), \lambda^*_\pi$, and Y^π_d explicitly and in closed-form in Appendix C.

Now that we have characterized the mean response time of the MMSR system, we can use this result to characterize the MSJ FCFS system. With our RESET technique, we show that the MSJ FCFS system has the same mean response time, up to an $O_\lambda(1)$ term, as the MMSR system whose service rate is controlled by the saturated system, or equivalently the At-least-k system.

Theorem 4.2 (Mean response time asymptotics of MSJ systems)**.** *In the multiserver-job system, the expected response time in steady state satisfies*

$$\mathbb{E}[T^{\text{MSJ}}] = \frac{1}{\lambda^*} \frac{1 + \Delta_{\text{Sat}}(Y^{\text{Sat}}_d, Y^{\text{Sat}})}{1 - \lambda/\lambda^*} + O_\lambda(1). \tag{4}$$

Empirically, the $O_\lambda(1)$ term is very small, as seen in Fig. 2a in Section 8. To clarify the meaning of the $O_\lambda(1)$ term in Theorem 4.2, let us restate the theorem explicitly:

Theorem 4.3 (Restatment of Theorem 4.2)**.** *In the multiserver-job system, for any joint duration and server need distribution and for any number of servers k, there exist constants c_ℓ and c_h such that for all arrival rates $\lambda < \lambda^*$,*

$$\frac{1}{\lambda^*} \frac{1 + \Delta_{\text{Sat}}(Y^{\text{Sat}}_d, Y^{\text{Sat}})}{1 - \lambda/\lambda^*} + c_\ell \le \mathbb{E}[T^{\text{MSJ}}] \le \frac{1}{\lambda^*} \frac{1 + \Delta_{\text{Sat}}(Y^{\text{Sat}}_d, Y^{\text{Sat}})}{1 - \lambda/\lambda^*} + c_h.$$

Rather than calculating $\Delta_{\text{Sat}}(Y^{\text{Sat}}_d, Y^{\text{Sat}})$ in Theorem 4.2, we can calculate the equivalent value in the simplified saturated system (SSS) (due to Lemma 3.3). Define $\Delta_{\text{SSS}}, Y^{\text{SSS}}_d$, and Y^{SSS} analogously to the primary saturated system.

Corollary 4.1. *In the MSJ FCFS model,*

$$\mathbb{E}[T^{\text{MSJ}}] = \frac{1}{\lambda^*} \frac{1 + \Delta_{\text{SSS}}(Y^{\text{SSS}}_d, Y^{\text{SSS}})}{1 - \lambda/\lambda^*} + O_\lambda(1).$$

Corollary 4.1 follows from Theorem 4.2 because $\Delta_{\text{Sat}}(y_1, y_2)$ is defined based on the completion times in the primary saturated system, and by Lemma 3.3, the SSS can be coupled to have the same completion times as the primary saturated system.

The quantities $\Delta_{\text{SSS}}(y), \lambda^*$, and Y^{SSS}_d can be calculated explicitly and in closed-form for any given parameterized distribution of server need and job duration, and any number of servers k, giving an explicit closed-form bound on mean response time. We walk through this calculation in Appendix C, and give the explicit closed-form expressions for a 2-server setting in Appendix C.2, to demonstrate the technique.

4.1. Example for demonstration

We now demonstrate applying Theorem 4.2 and Corollary 4.1 to characterize the asymptotic mean response time of our running example from Section 3.4. See Appendix C for a more extensive example, handling a setting with parameterized completion rates and arrival probabilities.

We start with the MSJ system. First, we convert to the Ak system, whose front has state space $\mathbb{Y}^{\text{Ak}} = \{[1,1],[1,2],[2,1],[2,2]\}$. By the RESET technique, this only increases mean response time by $O_\lambda(1)$. By Lemma 3.1, the Ak system is identical to a MMSR-Sat system. By Lemma 3.3, the Sat system is equivalent to Simplified Saturated System (SSS), which has state space $\mathbb{Y}^{\text{SSS}} = \{[1,1],[1,2],[2]\}$.

For the rest of this section, we focus on the SSS, leaving the superscript implicit. Transitions between these states only happen as a result of completions, leading to the following transition rates:

$$\mu_{[1,1],[1,1],1} = 2 \cdot \frac{2}{3} = \frac{4}{3}, \quad \mu_{[1,1],[1,2],1} = 2 \cdot \frac{1}{3} = \frac{2}{3}, \quad \mu_{[1,2],[2],1} = 1,$$
$$\mu_{[2],[1,1],1} = \frac{1}{2}\frac{2}{3}\frac{2}{3} = \frac{2}{9}, \quad \mu_{[2],[1,2],1} = \frac{1}{2}\frac{2}{3}\frac{1}{3} = \frac{1}{9}, \quad \mu_{[2],[2],1} = \frac{1}{2}\frac{1}{3} = \frac{1}{6}.$$

Now, we can calculate the steady states Y^{SSS} and Y_d^{SSS} of the SSS's CTMC and DTMC respectively, and calculate the throughput $X^{\text{SSS}} = X^{\text{Sat}} = \lambda^*$. The vectors are in the order $\{[1,1],[1,2],[2]\}$:

$$Y = \left[\frac{1}{5}, \frac{1}{5}, \frac{3}{5}\right], \quad Y^d = \left[\frac{4}{9}, \frac{2}{9}, \frac{1}{3}\right], \quad X^{\text{SSS}} = X^{\text{Sat}} = \lambda^* = \frac{9}{10}.$$

Now, we can solve for $\Delta(y)$, defined in (1). To do so, we split up the completions $\mathbb{E}[C(y,t)]$ into the time until the first completion, and the time after the first completion. For example, starting in state $y = [1,1]$, the first completion takes an expected $\frac{1}{2}$ second, during which 1 completion occurs, compared to the long-term average rate $\frac{1}{2}\lambda^* = \frac{9}{20}$ completions. The system then transitions to a new state, with corresponding $\Delta(y)$. This gives rise to the following equation:

$$\Delta([1,1]) = 1 - \frac{9}{20} + \frac{2}{3}\Delta([1,1]) + \frac{1}{3}\Delta([1,2]).$$

We use the same process to derive a system of equations that uniquely determines $\Delta(y)$, given in Corollary D.1. We solve for $\Delta(y)$ for each state y:

$$\Delta([1,1]) = 1.38, \quad \Delta([1,2]) = -0.27, \quad \Delta([2]) = -0.37. \tag{5}$$

All decimals are exact. We can then average over the distribution Y^d to find that $\Delta(Y^d) = 0.43$. Recall that $\Delta(Y^d)$ is just shorthand for $\Delta(Y^d, Y)$.

We can therefore apply Theorem 4.2 and Corollary 4.1 to characterize the asymptotic mean response time of the original system:

$$\mathbb{E}[T^{\text{MSJ}}] = \frac{10}{9}\frac{1.43}{1 - \frac{\lambda}{9/10}} + O_\lambda(1).$$

5. MARC Proofs

We start by analyzing the M/M/1 with Markovian service rate π (MMSR-π). Our main result in this section is the proof of Theorem 4.1, a characterization of the asymptotic mean response time of the MMSR-π system.

The main challenge is choosing an appropriate test function $f(q,y)$, to leverage (2), the fact that $\mathbb{E}[G^\pi \circ f(Q^\pi, Y^\pi)] = 0$, to give an expression for $\mathbb{E}[Q^\pi]$. To gain information about $\mathbb{E}[Q^\pi]$ via this approach, it is natural to choose a function f which is quadratic in q, because G^π is effectively a derivative. However, if we choose $f_1(q,y) = \frac{1}{2}q^2$, the expression $G^\pi \circ f_1(q,y)$ will have cross-terms in which both q and y appear, preventing further progress.

Instead, our key idea is to use relative completions Δ_π in our test function:

Definition 5.1. *Let* $f_\Delta^\pi(q,y) = \frac{1}{2}(q - \Delta_\pi(y))^2$.

The $\Delta_\pi(y)$ term smooths out the fluctuations in the system's service rate, so that the quantity $q - \Delta_\pi(y)$ has a constant drift of $-\lambda_\pi^*$ whenever $q > 0$.

This choice of test function ensures that $G^\pi \circ f_\Delta^\pi(q,y)$ separates into a linear term dependent only on q and a term dependent only on y. The separation allows us to characterize $\mathbb{E}[Q^\pi]$, and hence $\mathbb{E}[T^\pi]$, in Theorem 4.1.

Let $u = \mathbb{1}\{q = 0 \wedge a = 1\}$ denote the unused service caused by a given transition. Only completion transitions ($a = 1$) can cause unused service.

We start by decomposing $G^\pi \circ f_\Delta^\pi(q,y)$, into a term linearly dependent on q, and terms dependent only on y, a, and u:

Lemma 5.1. *For any state* (q,y) *of the MMSR-*π *system,*

$$G^\pi \circ f_\Delta^\pi(q,y) = (\lambda - \lambda_\pi^*)q - \lambda\Delta_\pi(y) + \frac{1}{2}\lambda + \sum_{y',a} \mu_{y,y',a}^\pi \left(\frac{1}{2}(-a + u - \Delta_\pi(y'))^2 - \frac{1}{2}\Delta_\pi(y)^2\right). \tag{6}$$

Proof deferred to Appendix D.

We can now characterize the mean response time of the MMSR-π system. We will use the fact that by Lemma 5.1, $G^\pi \circ f^\pi_\Delta(q,y)$ decomposes into a term linearly dependent on the queue length q, and terms that are not dependent on q except through the unused service u. We define $c_0(y,q)$ to comprise the later group of terms. We also define $c_1(y)$ and $c_2(y)$, which are simpler functions that are closely related to $c_0(y,q)$.

Definition 5.2. *Define $c_0(y,q), c_1(y)$, and $c_2(y)$ as follows:*

$$\begin{aligned}
c_0(y,q) &= G^\pi \circ f^\pi_\Delta(q,y) - (\lambda - \lambda^*_\pi)q \\
&= -\lambda\Delta_\pi(y) + \frac{1}{2}\lambda + \sum_{y',a} \mu^\pi_{y,y',a}\left(\frac{1}{2}(-a+u-\Delta_\pi(y'))^2 - \frac{1}{2}\Delta_\pi(y)^2\right), \\
c_1(y) &= -\lambda\Delta_\pi(y) + \frac{1}{2}\lambda + \sum_{y',a} \mu^\pi_{y,y',a}\left(\frac{1}{2}(-a-\Delta_\pi(y'))^2 - \frac{1}{2}\Delta_\pi(y)^2\right), \\
c_2(y) &= c_1(y) - G^\pi \circ h(y), \text{ where } h(y) = \frac{1}{2}\Delta_\pi(y)^2 \\
&= -\lambda\Delta_\pi(y) + \frac{1}{2}\lambda + \sum_{y',a} \mu^\pi_{y,y',a}\left(\frac{1}{2}a^2 + a\Delta_\pi(y')\right).
\end{aligned}$$

We will show that these functions' expected values, $\mathbb{E}[c_0(Y^\pi,Q^\pi)], \mathbb{E}[c_1(Y^\pi)]$, and $\mathbb{E}[c_2(Y^\pi)]$), are all equal up to a $O_\lambda(1-\frac{\lambda}{\lambda^*_\pi})$ error. This fact is crucial to our proof of Theorem 4.1.

Theorem 4.1 (Mean response time asymptotics of MMSR systems)**.** *In the MMSR-π system, the expected response time in steady state satisfies*

$$\mathbb{E}[T^\pi] = \frac{1}{\lambda^*_\pi}\frac{1+\Delta_\pi(Y^\pi_d, Y^\pi)}{1-\lambda/\lambda^*_\pi} + O_\lambda(1). \tag{7}$$

Proof. In this proof we omit π in the subscript of $\Delta_\pi(y)$ and in the superscript of $\mu^\pi_{y,y',a}$. We start from Lemma 5.1, which states that

$$G^\pi \circ f(q,y) = (\lambda - \lambda^*_\pi)q + c_0(y,q).$$

Applying Lemma 3.2, we find that

$$\begin{aligned}
0 = \mathbb{E}[G^\pi \circ f(Q^\pi, Y^\pi)] &= (\lambda - \lambda^*_\pi)\mathbb{E}[Q^\pi] + \mathbb{E}[c_0(Y^\pi, Q^\pi)], \\
\mathbb{E}[Q^\pi] &= \frac{\mathbb{E}[c_0(Y^\pi,Q^\pi)]}{\lambda^*_\pi - \lambda}.
\end{aligned}$$

We therefore focus on $c_0(q,y)$: By characterizing $\mathbb{E}[c_0(Y^\pi,Q^\pi)]$, we will characterize $\mathbb{E}[Q^\pi]$.

Let us separate out the terms where u appears in $c_0(y,q)$ from the terms without u:

$$c_0(y,q) - c_1(y) = \sum_{y',a} \mu_{y,y',a} u\left(\frac{1}{2}u - a - \Delta(y')\right). \tag{8}$$

Note that in the time-average steady state Y^π, the fraction of service-process completions that occur while the queue is empty (i.e. where $u=1$) is $1-\frac{\lambda}{\lambda^*_\pi}$, because λ jobs arrive per second, and λ^* service-process completions occur per second. As a result,

$$E_{y\sim Y^\pi}\left[\sum_{y',a}\mu_{y,y',a}u\right] = 1 - \frac{\lambda}{\lambda^*_\pi}.$$

Note that $a \leq 1$ and $u \leq 1$, because at most 1 job completes at a time. Note that $\Delta(y')$ is bounded by a constant over all y', because $y' \in \mathbb{Y}^\pi$, which is a finite state space. Thus, the $u/2 - a - \Delta(y')$ term in (8) is bounded by a constant. As a result, (8) contributes $O_\lambda(1 - \frac{\lambda}{\lambda^*_\pi})$ to $\mathbb{E}[c_0(Y^\pi, Q^\pi)]$:

$$\mathbb{E}[c_1(Y^\pi) - c_0(Y^\pi, Q^\pi)] = O_\lambda(1 - \lambda/\lambda^*_\pi).$$

Next, recall that $c_2(y) := c_1(y) - G^\pi \circ h(y)$. By Lemma 3.2, $\mathbb{E}[G^\pi \circ h(Y^\pi)] = 0$, so $\mathbb{E}[c_2(Y^\pi)] = \mathbb{E}[c_1(Y^\pi)]$. Let us now simplify $c_2(y)$, using the fact that $a = 0$ or 1:

$$\begin{aligned} c_2(y) &= -\lambda\Delta(y) + \frac{1}{2}\lambda + \sum_{y',a} \mu^\pi_{y,y',a}\left(\frac{1}{2}a^2 + a\Delta_\pi(y')\right) \\ &= -\lambda\Delta(y) + \frac{1}{2}\lambda + \frac{1}{2}\mu_{y,\cdot,1} + \sum_{y'} \mu_{y,y',1}\Delta(y'). \end{aligned}$$

We now apply Lemma D.3 to simplify the summation term of $c_2(y)$. Lemma D.3 states that

$$\frac{1}{\lambda^*_\pi}\mathbb{E}_{y\sim Y^\pi}[\mu^\pi_{y,y',1}] = \mathbb{P}(Y^\pi_d = y').$$

Thus, taking the expectation of the summation term of $c_2(y)$ over $y \sim Y^\pi$, we find that

$$\begin{aligned} \mathbb{E}_{y\sim Y^\pi}[\sum_{y'} \mu_{y,y',1}\Delta(y')] &= \lambda^*_\pi \sum_{y'} \mathbb{P}(Y^\pi_d = y')\Delta(y') = \lambda^*_\pi \Delta(Y^\pi_d), \\ \mathbb{E}[c_2(Y^\pi)] &= \mathbb{E}[-\lambda\Delta(Y^\pi) + \frac{1}{2}(\mu_{Y^\pi,\cdot,1} + \lambda) + \lambda^*_\pi\Delta(Y^\pi_d)]. \end{aligned}$$

Now note that $\mathbb{E}[\Delta(Y^\pi)] = 0$, $\mathbb{E}[\mu_{Y^\pi,\cdot,1}] = \lambda^*_\pi$, and $\lambda = \lambda^*_\pi + O_\lambda(1 - \frac{\lambda}{\lambda^*_\pi})$:

$$\begin{aligned} \mathbb{E}[c_1(Y^\pi)] = \mathbb{E}[c_2(Y^\pi)] &= \lambda^*_\pi + \lambda^*_\pi\Delta(Y^\pi_d) + O_\lambda(1 - \frac{\lambda}{\lambda^*_\pi}). \qquad (9) \\ \mathbb{E}[c_0(Y^\pi, Q^\pi)] = \mathbb{E}[c_1(Y^\pi)] + O_\lambda(1 - \frac{\lambda}{\lambda^*_\pi}) &= \mathbb{E}[c_2(Y^\pi)] + O_\lambda(1 - \frac{\lambda}{\lambda^*_\pi}). \\ \mathbb{E}[Q^\pi] = \frac{\mathbb{E}[c_0(Y^\pi, Q^\pi)]}{\lambda^*_\pi - \lambda^*} &= \frac{\lambda^*_\pi + \lambda^*_\pi\Delta(Y^\pi_d)}{\lambda^*_\pi - \lambda} + O_\lambda(1) = \frac{\Delta(Y^\pi_d) + 1}{1 - \lambda/\lambda^*_\pi} + O_\lambda(1). \end{aligned}$$

Now, we apply Little's Law, which states that $\mathbb{E}[T^\pi] = \frac{1}{\lambda}\mathbb{E}[Q^\pi]$:

$$\mathbb{E}[T^\pi] = \frac{1}{\lambda}\frac{1 + \Delta(Y^\pi_d)}{1 - \lambda/\lambda^*_\pi} + O_\lambda\left(\frac{1}{\lambda}\right).$$

Note that for any x, $\frac{1}{\lambda}\frac{x}{1-\lambda/\lambda^*} = \frac{1}{\lambda^*}\frac{x}{1-\lambda/\lambda^*} + \frac{x}{\lambda}$, so

$$\mathbb{E}[T^\pi] = \frac{1}{\lambda^*_\pi}\frac{1 + \Delta(Y^{\text{Sat}}_d)}{1 - \lambda/\lambda^*_\pi} + O_\lambda\left(\frac{1}{\lambda}\right). \qquad (10)$$

Note that in the $\lambda \to \lambda^*_\pi$ limit, $O_\lambda(\frac{1}{\lambda}) = O_\lambda(1)$. Consider the $\lambda \to 0$ limit: $\mathbb{E}[T^\pi]$ is bounded for small λ. Likewise, $\frac{1}{\lambda^*_\pi}\frac{1+\Delta(Y^\pi_d)}{1-\lambda/\lambda^*_\pi}$ is bounded for small λ. As a result, the two differ by $O_\lambda(1)$:

$$\mathbb{E}[T^{\text{MSJ}}] = \frac{1}{\lambda^*_\pi}\frac{1 + \Delta(Y^\pi_d)}{1 - \lambda/\lambda^*_\pi} + O_\lambda(1). \quad \square$$

6. RESET Proofs

To characterize the asymptotic behavior of mean response time of the MSJ system, we use the At-least-k (Ak) system, which is stochastically equal to the MMSR-Sat system. The MARC results from Section 5 allow us to characterize the MMSR-Sat system. To prove that the MSJ FCFS and Ak systems have the same asymptotic mean response time behavior, our key idea is to show that Y^{MSJ} and Y^{Ak}, the steady states of their fronts, are "almost identical."

To formalize and prove the relationship between Y^{MSJ} and Y^{Ak}, we design a coupling in Section 6.1 between the MSJ system and the Ak system. We use a renewal-reward argument based on busy periods to prove Lemma 6.2, which states that under the coupling, $\mathbb{P}(Y^{\text{MSJ}} \neq Y^{\text{Ak}}) = O_\lambda(1 - \frac{\lambda}{\lambda^*})$.

Then, in Section 6.2, we combine Theorem 4.1 and Lemma 6.2 to prove Theorem 4.2, our main result, in which we give the first analysis of the asymptotic mean response time in the MSJ system, by reduction to the saturated system. Theorem 4.2 parallels the proof steps that Theorem 4.1 uses to characterize the MMSR system, using Lemma 6.2 to prove that the equivalent proof steps hold for the MSJ system.

We will make use of a test function $f_\Delta^{\text{MSJ}}(q, y)$ for the multiserver-job system which is similar to $f_\Delta^{\pi}(q, y)$, which was defined in Definition 5.1.

Definition 6.1. *For states $y \in \mathbb{Y}^{\text{Ak}}$,*

$$f_\Delta^{\text{MSJ}}(q, y) := f_\Delta^{\text{Ak}}(q, y) = f_\Delta^{\text{Sat}}(q, y).$$

Otherwise,

$$f_\Delta^{\text{MSJ}}(q, y) := 0.$$

Importantly, $G^{\text{MSJ}} \circ f_\Delta^{\text{MSJ}}(q, y)$ is similar to $G^{\text{Ak}} \circ f_\Delta^{\text{Ak}}(q, y)$:

Lemma 6.1.

$$G^{\text{MSJ}} \circ f_\Delta^{\text{MSJ}}(q, y) = \mathbb{1}_{q>0} G^{\text{Ak}} \circ f_\Delta^{\text{Ak}}(q, y) + \mathbb{1}_{q=0} O_\lambda(1).$$

Proof deferred to Appendix E.

6.1. Coupling between At-least-k and MSJ

To show that the Ak system and the MSJ system have identical asymptotic mean response time, we define the following coupling of the two systems. We let the arrivals of the two systems happen at the same time. We couple the transitions of their front states based on their joint state $(q^{\text{MSJ}}, y^{\text{MSJ}}, q^{\text{Ak}}, y^{\text{Ak}})$. If $y^{\text{MSJ}} = y^{\text{Ak}}$, $q^{\text{MSJ}} > 0$, and $q^{\text{Ak}} > 0$, the completions happen at the same time in both systems, the same jobs complete, the same job phase transitions occur, and the jobs entering the front are the same. We call the two systems "merged" during such a time period. Note that under this coupling, if the two systems become merged, they will stay merged until $q^{\text{MSJ}} = 0$ or $q^{\text{Ak}} = 0$. If the systems are not merged, the two systems have independent completions and phase transitions, and independently sampled jobs.

The two systems transition according to synchronized Poisson timers whenever they are merged, and independent Poisson timers otherwise. Because all transitions are exponentially distributed, this poses no obstacle to the coupling.

We want to show that under this coupling, the two systems spend almost all of their time merged, in the limit as $\lambda \to \lambda^*$. Specifically, we will show that the fraction of time in which the two systems are *unmerged* is $O_\lambda(1 - \frac{\lambda}{\lambda^*})$. This implies Lemma 6.2, which is the key lemma we need for our main RESET result, Theorem 4.2.

Lemma 6.2 (Tight coupling)**.** *In the MSJ system, for any $\lambda < \lambda^*$, we have the following two properties:*

1. *Property 1: $P(Q^{\text{MSJ}} = 0) = O_\lambda(1 - \frac{\lambda}{\lambda^*})$.*
2. *Property 2: $P(Y^{\text{MSJ}} \neq Y^{\text{Ak}}) = O_\lambda(1 - \frac{\lambda}{\lambda^*})$.*

where property 2 holds under the coupling in Section 6.1.

To prove Lemma 6.2, we prove two key lemmas:

- Lemma 6.3: Whenever the two systems are unmerged, the expected time until the systems become merged is $O_\lambda(1)$.
- Lemma 6.4: Whenever the two systems are merged, the expected time for which they stay merged is $\Omega_\lambda(\frac{1}{1-\lambda/\lambda^*})$.

We then use a renewal-reward approach to prove Lemma 6.2.

Lemma 6.3 (Quick merge)**.** *From any joint MSJ, Ak state, for any $\epsilon > 0$, under the coupling above, the expected time until $y^{\mathrm{MSJ}} = y^{\mathrm{Ak}}$, $q^{\mathrm{MSJ}} \geq k+1$, and $q^{\mathrm{Ak}} \geq k+1$ is at most $m_1(\epsilon)$ for some $m_1(\epsilon)$ independent of the arrival rate λ and initial joint states, given that $\lambda \in [\epsilon, \lambda^*)$.*

Lemma 6.4 (Long merged period)**.** *From any joint MSJ, Ak state such that $y^{\mathrm{MSJ}} = y^{\mathrm{Ak}}$, $q^{\mathrm{MSJ}} \geq k+1$, and $q^{\mathrm{Ak}} \geq k+1$, the expected time until $q^{\mathrm{MSJ}} = 0$, $q^{\mathrm{Ak}} = 0$, or $y^{\mathrm{MSJ}} \neq y^{\mathrm{Ak}}$, is at least $\frac{m_2}{1-\lambda/\lambda^*}$ for some m_2 independent of the arrival rate λ and initial joint states, given that $\lambda < \lambda^*$.*

Proofs deferred to Appendix G.

Using Lemmas 6.3 and 6.4, we can prove Lemma 6.2:

Proof. Let $\epsilon = \frac{\lambda^*}{2}$. Note that if $\lambda < \epsilon$, the properties are trivial: $O_\lambda(1 - \frac{\lambda}{\lambda^*}) \equiv O_\lambda(1)$, and probabilities are bounded. Therefore, we will focus on the case where $\lambda \geq \epsilon$, where we can apply Lemmas 6.3 and 6.4.

Let us define a *good period* to begin when $Y^{\mathrm{MSJ}}(t) = Y^{\mathrm{Ak}}(t)$, $Q^{\mathrm{MSJ}}(t) \geq k+1$ and $Q^{\mathrm{Ak}}(t) \geq k+1$, and end when $Q^{\mathrm{MSJ}}(t) = 0$ or $Q^{\mathrm{Ak}}(t) = 0$. Let a *bad period* be the time between two good periods. Note that throughout a good period, the front states are merged ($Y^{\mathrm{MSJ}}(t) = Y^{\mathrm{Ak}}(t)$) and both queues are nonempty.

To bound the fraction of time that the joint system is in a good period, we introduce the concept of a "y^*-cycle." Let y^* be an arbitrary state in $\mathbb{Y}^{\mathrm{Ak}}$. Let a y^*-cycle be a renewal cycle whose renewal points are moments when a bad period begins, and $Y^{\mathrm{MSJ}}(t) = Y^{\mathrm{Ak}}(t) = y^*$, and $Q^{\mathrm{MSJ}}(t) = Q^{\mathrm{Ak}}(t) = 0$, for some designated state y^*. We will show that a y^*-cycle has finite mean time. Given that fact, we can apply renewal reward to derive the equations below:

$$P(Q^{\mathrm{MSJ}} = 0) = \frac{\mathbb{E}[Q^{\mathrm{MSJ}}(t) = 0 \text{ time per } y^*\text{-cycle}]}{\mathbb{E}[\text{total time per } y^*\text{-cycle}]}, \tag{11}$$

$$P(Y^{\mathrm{MSJ}} \neq Y^{\mathrm{Ak}}) = \frac{\mathbb{E}[Y^{\mathrm{MSJ}}(t) \neq Y^{\mathrm{Ak}}(t) \text{ time per } y^*\text{-cycle}]}{\mathbb{E}[\text{total time per } y^*\text{-cycle}]}. \tag{12}$$

Note that $Q^{\mathrm{MSJ}}(t) = 0$ or $Y^{\mathrm{MSJ}}(t) \neq Y^{\mathrm{Ak}}(t)$ only during a bad period, so the two probabilities in (11) and (12) are both bounded by the fraction of time spent in bad periods. By Lemma 6.3 and Lemma 6.4, the expected length of a bad period is at most m_1 and the expected length of a good period is at least $\frac{m_2}{1-\lambda/\lambda^*}$, conditioned on any initial joint state. Let Z be a random variable denoting the number of good periods in a y^* cycle. Note that good and bad periods alternate.

$$\mathbb{E}[\text{total time per } y^*\text{-cycle}] \geq \frac{m_2}{1-\lambda/\lambda^*}\mathbb{E}[Z],$$
$$\mathbb{E}[\text{bad period time per } y^*\text{-cycle}] \leq m_1\mathbb{E}[Z].$$

If a y^*-cycle has finite mean time, then we also have $\mathbb{E}[Z] < \infty$ because each good period and bad period take a positive time. Plugging the above inequalities into (11) and (12), we derive Properties 1 and 2:

$$P(Q^{\mathrm{MSJ}} = 0) \leq \frac{m_1}{m_2}\left(1 - \frac{\lambda}{\lambda^*}\right), \qquad P(Y^{\mathrm{MSJ}} \neq Y^{\mathrm{Ak}}) \leq \frac{m_1}{m_2}\left(1 - \frac{\lambda}{\lambda^*}\right).$$

It remains to show that a y^*-cycle has finite mean time. We first use a Lyapunov argument to show that the joint states of the two systems return to a bounded set in a finite mean time. Consider the Lyapunov function $f_\Delta^{\mathrm{MSJ}}(q^{\mathrm{MSJ}}, y^{\mathrm{MSJ}}) + f_\Delta^{\mathrm{Ak}}(q^{\mathrm{Ak}}, y^{\mathrm{Ak}})$. Its drift is:

$$G^{\mathrm{MSJ,Ak}} \circ \left(f_\Delta^{\mathrm{MSJ}}(q^{\mathrm{MSJ}}, y^{\mathrm{MSJ}}) + f_\Delta^{\mathrm{Ak}}(q^{\mathrm{Ak}}, y^{\mathrm{Ak}})\right) = G^{\mathrm{MSJ}} \circ f_\Delta^{\mathrm{MSJ}}(q^{\mathrm{MSJ}}, y^{\mathrm{MSJ}}) + G^{\mathrm{Ak}} \circ f_\Delta^{\mathrm{Ak}}(q^{\mathrm{Ak}}, y^{\mathrm{Ak}}).$$

Applying Lemma 5.1 to the Ak system,

$$G^{\mathrm{Ak}} \circ f_{\Delta}^{\mathrm{Ak}}(q^{\mathrm{Ak}}, y^{\mathrm{Ak}}) = (\lambda - \lambda^*)q^{\mathrm{Ak}} + c_0(y^{\mathrm{Ak}}, q^{\mathrm{Ak}}),$$

where $c_0(y, q)$ is defined in Definition 5.2. Note that $c_0(y, q)$ is a bounded function because $\Delta(y)$ is bounded, by Lemma A.1. Let $c_{\max}^{\mathrm{Ak}}$ be the maximum of $c_0(y, q)$. For all $y^{\mathrm{Ak}}, q^{\mathrm{Ak}}$,

$$G^{\mathrm{Ak}} \circ f_{\Delta}^{\mathrm{Ak}}(q^{\mathrm{Ak}}, y^{\mathrm{Ak}}) \le (\lambda - \lambda^*)q^{\mathrm{Ak}} + c_{\max}^{\mathrm{Ak}}.$$

By similar reasoning, applying Lemma 6.1, there exists a $c_{\max}^{\mathrm{MSJ}}$ such that

$$G^{\mathrm{MSJ}} \circ f_{\Delta}^{\mathrm{MSJ}}(q^{\mathrm{MSJ}}, y^{\mathrm{MSJ}}) \le (\lambda - \lambda^*)q^{\mathrm{Ak}} + c_{\max}^{\mathrm{MSJ}}$$

Let $c_{\max} = \max(c_{\max}^{\mathrm{Ak}}, c_{\max}^{\mathrm{MSJ}})$. Consider any $q^{\mathrm{Ak}} \ge \frac{2c_{\max}+1}{\lambda^* - \lambda}$. Then for any y^{Ak},

$$G^{\mathrm{Ak}} \circ f_{\Delta}^{\mathrm{Ak}}(q^{\mathrm{Ak}}, y^{\mathrm{Ak}}) \le -c_{\max} - 1.$$

Similarly, for any $q^{\mathrm{MSJ}} \ge \frac{2c_{\max}+1}{\lambda^* - \lambda}$ and any y^{MSJ},

$$G^{\mathrm{MSJ}} \circ f_{\Delta}^{\mathrm{MSJ}}(q^{\mathrm{MSJ}}, y^{\mathrm{MSJ}}) \le -c_{\max} - 1.$$

Let $c_{\mathrm{cap}} = \max\{\frac{2c_{\max}+1}{\lambda^* - \lambda}, k+1\}$. We define the bounded set $\mathbb{S}$ as

$$\mathbb{S} = \left\{(q^{\mathrm{MSJ}}, q^{\mathrm{Ak}}, y^{\mathrm{MSJ}}, y^{\mathrm{Ak}}) \colon q^{\mathrm{MSJ}} \le c_{\mathrm{cap}}, q^{\mathrm{Ak}} \le c_{\mathrm{cap}}\right\}.$$

By the calculation above, outside $\mathbb{S}$,

$$G^{\mathrm{MSJ}} f_{\Delta}^{\mathrm{MSJ}}(q^{\mathrm{MSJ}}, y^{\mathrm{MSJ}}) + G^{\mathrm{Ak}} f_{\Delta}^{\mathrm{Ak}}(q^{\mathrm{Ak}}, y^{\mathrm{Ak}}) \le -1.$$

In particular, outside of $\mathbb{S}$, either $q^{\mathrm{MSJ}} > c_{\max}$ or $q^{\mathrm{Ak}} > c_{\max}$, yielding a drift term $\le -c_{\max} - 1$, outweighing the term where q is small. Thus, by the Foster-Lyapunov theorem [30, Theorem A.4.1], the system returns to $\mathbb{S}$ in finite mean time.

We call a period of time inside the bounded set $\mathbb{S}$ an $\mathbb{S}$-*visit*. Each $\mathbb{S}$-visit has a finite mean time because there is a positive probability of having a lot of arrivals in the next second and leaving $\mathbb{S}$. Moreover, as proved above using the Lyapunov argument, the time between two $\mathbb{S}$-visits has finite mean.

Each $\mathbb{S}$-visit has a positive probability of ending the y^*-cycle. To prove this, we construct a positive probability sample path of beginning a good period with $q^{\mathrm{MSJ}} = q^{\mathrm{Ak}}$ and ending the good period in $(0, 0, y^*, y^*)$, while remaining in $\mathbb{S}$.

- First, we have a lot of completions in the two systems, completely emptying both. $q^{\mathrm{MSJ}} = q^{\mathrm{Ak}} = |y^{\mathrm{MSJ}}| = 0$. Next, k jobs arrive. Now $q^{\mathrm{Ak}} = k$ and $q^{\mathrm{MSJ}} = 0$. During this time $y^{\mathrm{MSJ}} \ne y^{\mathrm{Ak}}$.
- Then k jobs complete in the Ak system, no jobs complete in the MSJ system, and the newly generated Ak jobs are sampled such that $y^{\mathrm{MSJ}} = y^{\mathrm{Ak}}$, while $q^{\mathrm{MSJ}} = q^{\mathrm{Ak}} = 0$.
- Next, $k+1$ jobs arrive, and a good period begins.
- Finally, $k+1$ jobs complete in both systems, ending with $y^{\mathrm{MSJ}} = y^{\mathrm{Ak}} = y^*$, and $q^{\mathrm{MSJ}} = q^{\mathrm{Ak}} = 0$. Now a y^*-cycle ends, and the next begins.

All of these events have strictly positive probability and are independent of each other, so their joint occurrence has strictly positive probability as well. Thus, the length of a y^*-cycle is bounded by a geometric number of $\mathbb{S}$-visits, each of which has finite mean time, completing the proof. □

6.2. Proof of Theorem 4.2

We now are ready to prove our main theorem, Theorem 4.2, progressing along similar lines as Theorem 4.1 and making use of Lemmas 5.1 and 6.2. First, we restate several definitions from Definition 5.2, specialized to Ak system:

Definition 6.2. *Recall the definitions of $c_0(y,q)$ and $c_1(y)$ from Definition 5.2:*

$$\begin{aligned} c_0(y,q) &= G^{\text{Ak}} \circ f_\Delta^{\text{Ak}}(q,y) - (\lambda - \lambda^*)q \\ &= -\lambda\Delta(y) + \frac{1}{2}\lambda + \sum_{y',a} \mu_{y,y',a} \left(\frac{1}{2}(-a + u - \Delta(y'))^2 - \frac{1}{2}\Delta(y)^2 \right), \\ c_1(y) &= -\lambda\Delta(y) + \frac{1}{2}\lambda + \sum_{y',a} \mu_{y,y',a} \left(\frac{1}{2}(-a - \Delta(y'))^2 - \frac{1}{2}\Delta(y)^2 \right), \end{aligned}$$

where $u = \mathbb{1}\{q = 0 \wedge a = 1\}$.

We also make use of a key fact about $c_1(y)$, from (9):

$$\mathbb{E}[c_1(Y^{\text{Ak}})] = \lambda^* + \lambda^* \Delta(Y_d^{\text{Sat}}) + O_\lambda\left(1 - \frac{\lambda}{\lambda^*}\right).$$

Throughout this section, whenever we make use of results from Section 5, we set $\pi = \text{Sat}$. In particular, we make use of $c_0(y,q)$ and $c_1(y)$, from Definition 5.2.

Theorem 4.2. *In the multiserver-job system, the expected response time in steady state satisfies*

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

Proof. We will show that the MSJ model has the same asymptotic mean response time as the Ak system. We will make use of the test function $f_\Delta^{\text{MSJ}}(q,y)$, from Definition 6.1. Recall from Lemma 6.1 that

$$G^{\text{MSJ}} \circ f_\Delta^{\text{MSJ}}(q,y) = G^{\text{Ak}} \circ f_\Delta^{\text{Ak}}(q,y)\mathbb{1}_{q>0} + \mathbb{1}_{q=0} O_\lambda(1).$$

We will next use (2), the fact that the expected value of a generator function in steady state is zero, which implies that

$$0 = \mathbb{E}[G^{\text{Ak}} \circ f_\Delta^{\text{Ak}}(Q^{\text{MSJ}}, Y^{\text{MSJ}})\mathbb{1}\{Q^{\text{MSJ}} > 0\}] + \mathbb{P}(Q^{\text{MSJ}} = 0) O_\lambda(1). \tag{13}$$

By Lemma 6.2, $\mathbb{P}(Q^{\text{MSJ}} = 0) = O_\lambda(1 - \frac{\lambda}{\lambda^*})$. Next, we apply Lemma 5.1 to the Ak system, finding that

$$G^{\text{Ak}} \circ f_\Delta^{\text{Ak}}(q,y) = (\lambda - \lambda^*)q + c_0(y,q).$$

From Definition 6.2, we can see that $c_0(y,q)\mathbb{1}_{q>0} = c_1(y)\mathbb{1}_{q>0}$. Combining with (13) and invoking Lemmas 5.1 and 6.2 and the fact that $c_1(y)$ is bounded, we have

$$\begin{aligned} (\lambda - \lambda^*)\mathbb{E}[Q^{\text{MSJ}}] + \mathbb{E}[c_0(Y^{\text{MSJ}}, Q^{\text{MSJ}})\mathbb{1}\{Q^{\text{MSJ}} > 0\}] &= O_\lambda(1 - \lambda/\lambda^*), \\ (\lambda - \lambda^*)\mathbb{E}[Q^{\text{MSJ}}] + \mathbb{E}[c_1(Y^{\text{MSJ}})] &= O_\lambda(1 - \lambda/\lambda^*), \\ \mathbb{E}[Q^{\text{MSJ}}] &= \frac{\mathbb{E}[c_1(Y^{\text{MSJ}}])}{\lambda^* - \lambda} + O_\lambda(1). \end{aligned} \tag{14}$$

Next, specializing (9) in the proof of Theorem 4.1 to the Ak system, we know that

$$\mathbb{E}[c_1(Y^{\text{Ak}})] = \lambda^* + \lambda^* \Delta(Y_d^{\text{Sat}}) + O_\lambda\left(1 - \frac{\lambda}{\lambda^*}\right).$$

By Lemma 6.2, we know that $\mathbb{P}(Y^{\text{Ak}} \neq Y^{\text{MSJ}}) = O_\lambda(1 - \frac{\lambda}{\lambda^*})$. Again because $c_1(y)$ is bounded,

$$\mathbb{E}[c_1(Y^{\text{MSJ}})] = \mathbb{E}[c_1(Y^{\text{Ak}})] + O_\lambda\left(1 - \frac{\lambda}{\lambda^*}\right) = \lambda^* + \lambda^* \Delta(Y_d^{\text{Sat}}) + O_\lambda\left(1 - \frac{\lambda}{\lambda^*}\right).$$

Therefore, applying (14), we find that

$$\mathbb{E}[Q^{\text{MSJ}}] = \frac{1 + \Delta(Y_d^{\text{Sat}})}{1 - \lambda/\lambda^*} + O_\lambda(1).$$

Now, we apply Little's Law, which states that $\mathbb{E}[T^{\text{Ak}}] = \frac{1}{\lambda}\mathbb{E}[N^{\text{Ak}}]$. Note that Q^{Ak} and N^{Ak} differ by the number of jobs in the front, which is $O_\lambda(1)$:

$$\mathbb{E}[T^{\text{MSJ}}] = \frac{1}{\lambda}\frac{1 + \Delta(Y_d^{\text{Sat}})}{1 - \lambda/\lambda^*} + O_\lambda\left(\frac{1}{\lambda}\right) = \frac{1}{\lambda^*}\frac{1 + \Delta(Y_d^{\text{Sat}})}{1 - \lambda/\lambda^*} + O_\lambda\left(\frac{1}{\lambda}\right).$$

For the second equality, note that for any x, $\frac{1}{\lambda}\frac{x}{1-\lambda/\lambda^*} = \frac{1}{\lambda^*}\frac{x}{1-\lambda/\lambda^*} + \frac{x}{\lambda}$. Here x is a constant, so the extra term is absorbed by the $O_\lambda(1/\lambda)$.

By the same bounding argument as used for (10) in the $\lambda \to 0$ limit,

$$\mathbb{E}[T^{\text{MSJ}}] = \frac{1}{\lambda^*}\frac{1 + \Delta(Y_d^{\text{Sat}})}{1 - \lambda/\lambda^*} + O_\lambda(1). \quad \square$$

7. Extensions of RESET: Finite skip models

While our main MSJ result, Theorem 4.2, was stated for the MSJ FCFS model, our techniques do not depend on the details of that model. Our RESET technique can handle a wide variety of models, which we call "finite skip" models:

Definition 7.1. *A* finite skip *queueing model is one in which jobs are served in near-FCFS order. Only jobs among the n oldest jobs in arrival order are eligible for service, for some constant n. Service is only dependent on the states of the n oldest jobs in arrival order, plus an optional environmental state from a finite-state Markov chain. Furthermore, jobs must have finite state spaces, and arrivals must be Poisson with i.i.d. initial job states.*

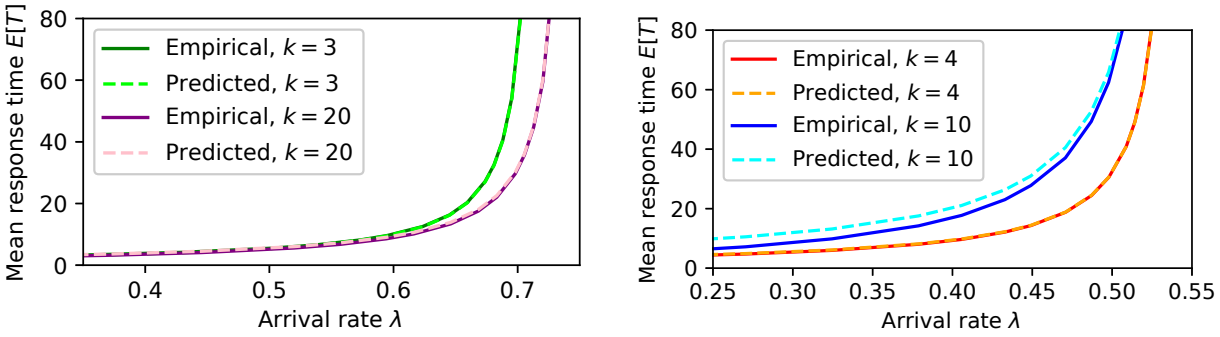
Definition 7.1 generalizes the work-conserving finite-skip (WCFS) class [17]. The MARC and RESET techniques can characterize the asymptotic mean response time of *any* finite skip model, via the procedure in Fig. 1. Additional finite skip MSJ models include nontrivial scheduling policies, including some backfilling policies; changing server need during service; multidimensional resource constraints; heterogeneous servers; turning off idle servers; and preemption overheads. For discussion of each of these variants, see Appendix H.

8. Empirical Validation

We have characterized the asymptotic mean response time behavior of the FCFS multiserver-job system. To illustrate and empirically validate our theoretical results, we simulate the mean response time of the MSJ model to compare it to our predictions. Recall (4) from Theorem 4.2, in which we proved mean response time can be characterized as a dominant term plus a $O_\lambda(1)$ term:

$$\mathbb{E}[T^{\text{MSJ}}] = \frac{1}{\lambda^*}\frac{1 + \Delta(Y_d^{\text{Sat}}, Y^{\text{Sat}})}{1 - \lambda/\lambda^*} + O_\lambda(1). \tag{15}$$

In this section, we simulate mean response time $\mathbb{E}[T^{\text{MSJ}}]$, and compare it against the dominant term of (15), which we compute explicitly.



(a) (1) $k = 3$, server need sampled uniformly from $\{1, 2, 3\}$, durations $Exp(1/3)$, $Exp(2/3)$, and $Exp(1)$, respectively. (2) $k = 20$, server need sampled uniformly from $\{1, 20\}$, durations $Exp(1)$ and $Exp(1/2)$, respectively.

(b) (1) $k = 4$, two classes of jobs: Server need 1, duration $Exp(1/4)$ w.p. 42%. Server need 4, duration $Exp(1)$ w.p. 58%. (2) $k = 10$, two classes of jobs: Server need 1, duration $Exp(1/10)$ w.p. 10%. Server need 10, duration $Exp(1)$ w.p. 90%.

Figure 2: Empirical and predicted mean response time $\mathbb{E}[T]$ for two MSJ settings in each of figures (a) and (b). Simulated 10^8 arrivals at arrival rates ranging over $\lambda/\lambda^* \in [0.5, 0.99]$.

8.1. Accuracy of formula

In Fig. 2a, we show that our predictions are an excellent match for the empirical behavior of the MSJ system in two different settings. In the first, there are $k = 3$ servers and jobs have server needs of 1, 2, and 3. In the second, there are $k = 20$ servers, and jobs have server needs 1 and 20. We thereby cover a spectrum from few-server-systems to many-server-systems, demonstrating extremely high accuracy in both regimes. The $O_\lambda(1)$ term in (15) is negligible in both of these examples.

In Fig. 2b, we compare mean response time in two settings with the same size distribution and stability region, but which have very different Δ. We discuss these settings further in Section 8.2.

The first setting has $k = 4$, and 42% of jobs have server need 1, while 58% of jobs have server need 4. The second setting has $k = 10$, and 10% of jobs have server need 1, while 90% of jobs have server need 10. The settings' stability regions are near-identical, with thresholds $\lambda_4^* \approx 0.5413, \lambda_{10}^* \approx 0.5411$, and their *size* distributions, defined as duration times server need over k, are both $Exp(1)$. However, our predictions for mean response time are very different in the two settings: $\Delta(Y_d^{\text{Sat}})_4 \approx 0.3271, \Delta(Y_d^{\text{Sat}})_{10} \approx 1.850$. The $k = 10$ setting considered here, with its relatively large value of $\Delta(Y_d^{\text{Sat}})$, is an especially difficult test-case. Nonetheless, our predictions are validated by the simulation results in Fig. 2b.

In Fig. 3, we illustrate the relative error between our predicted mean response time and the simulated mean response time for the four settings depicted in Fig. 2. In all four settings, as the arrival rate λ approaches λ^*, the threshold of the stability region, the relative error converges to 0.

Note that the convergence rate is slowest in the $k = 10$ setting, which also has the largest $\Delta(Y_d^{\text{Sat}})$ value. We further explore the relationship between $\Delta(Y_d^{\text{Sat}})$ values and convergence rates in Appendix I. We find that such a correlation exists in some settings, but it is not robust or reliable.

8.2. Understanding the importance of Δ

Our results show that the relative completions function Δ is key to understanding the response time behavior of non-work-conserving systems such as the MSJ FCFS system. This is in contrast to work-conserving systems, in which response time is determined by the size distribution and load [17]. This contrast is illustrated by Fig. 2b, in which we compare mean response time in two settings with the same size distribution and stability region, but which have very different Δ.

The differing mean response time behavior in these two settings is caused by the difference in *waste correlation*. In the $k = 10$ case, wasteful states persist for long periods of time: If a 1-server job is the only job in service, it takes more time for it to complete than in the $k = 4$ system. Thus, in the $k = 4$ case, wasteful states are more short-lasting. This difference in waste correlation produces the differences in $\Delta(Y_d^{\text{Sat}})$ and in mean response time.

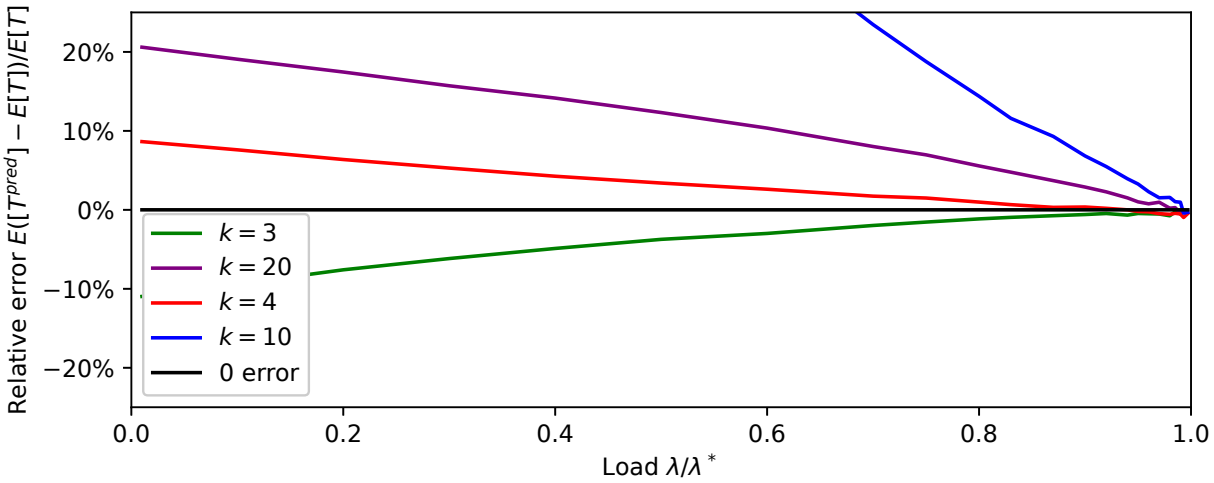


Figure 3: Relative error between empirical and predicted mean response time $E[T]$ for the four MSJ setting described in Fig. 2. Simulated 10^8 arrivals at arrival rates ranging over $\lambda/\lambda^* \in [0, 0.997]$.

This example highlights a crucial feature of MSJ FCFS: The failure of work conservation injects idiosyncratic idleness patterns in to the system. To characterize $\mathbb{E}[T]$, we need to characterize these patterns, which the RESET and MARC techniques enable us to do for the first time.

9. Conclusion

We introduce the RESET and MARC techniques. The RESET technique allows us to reduce the problem of characterizing mean response time in the MSJ FCFS system, up to an additive constant, to the problem of characterizing the M/M/1 with Markovian service rate (MMSR), where the service process is controlled by the saturated system. The MARC technique gives the first explicit characterization of mean response time in the MMSR, up to an additive constant. Together, our techniques reduce $\mathbb{E}[T^{\text{MSJ}}]$ to two properties of the saturated system: the departure-average steady state Y_d^{Sat}, and the relative completions function $\Delta(y_1, y_2)$. Our RESET and MARC techniques apply to any finite skip model, including many MSJ generalizations.

We also introduce the simplified saturated system, a yet-simper variant of the saturated system with identical behavior. We empirically validate our theoretical result, showing that it closely tracks simulation at all arrival rates λ.

An important direction for future work is to analytically characterize the relative completions $\Delta(y_1, y_2)$ for specific MSJ FCFS settings, such as settings where Y_d^{Sat} is known to have a product-form distribution [16, 42].

10. Acknowledgements

Isaac Grosof and Mor Harchol-Balter were supported by the National Science Foundation under grant number CMMI-2307008. Yige Hong was supported by the National Science Foundation under grant number ECCS-2145713. We thank the shepherd and the anonymous reviewers for their helpful comments.

References

[1] Larisa Afanaseva, Elena Bashtova, and Svetlana Grishunina. 2019. Stability Analysis of a Multi-server Model with Simultaneous Service and a Regenerative Input Flow. *Methodology and Computing in Applied Probability* (2019), 1–17.

[2] François Baccelli and Serguei Foss. 1995. On the saturation rule for the stability of queues. *Journal of Applied Probability* 32, 2 (1995), 494–507. https://doi.org/10.2307/3215303

[3] Percy H. Brill and Linda Green. 1984. Queues in Which Customers Receive Simultaneous Service from a Random Number of Servers: A System Point Approach. *Management Science* 30, 1 (1984), 51–68.

[4] Danilo Carastan-Santos, Raphael Y. De Camargo, Denis Trystram, and Salah Zrigui. 2019. One Can Only Gain by Replacing EASY Backfilling: A Simple Scheduling Policies Case Study. In *2019 19th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID)*. 1–10.

[5] A Bruce Clarke. 1956. A waiting line process of Markov type. *The Annals of Mathematical Statistics* (1956), 452–459.

[6] Mohammad Delasay, Armann Ingolfsson, and Bora Kolfal. 2016. Modeling Load and Overwork Effects in Queueing Systems with Adaptive Service Rates. *Operations Research* 64, 4 (2016), 867–885.

[7] Sherwin Doroudi. 2016. Stochastic analysis of maintenance and routing policies in queueing systems. (2016).

[8] Atilla Eryilmaz and R. Srikant. 2012. Asymptotically Tight Steady-State Queue Length Bounds Implied by Drift Conditions. *Queueing Syst. Theory Appl.* 72, 3–4 (dec 2012), 311–359. https://doi.org/10.1007/s11134-012-9305-y

[9] Yoav Etsion and Dan Tsafrir. 2005. A short survey of commercial cluster batch schedulers. *School of Computer Science and Engineering, The Hebrew University of Jerusalem* 44221 (2005), 2005–13.

[10] Dror G. Feitelson, Larry Rudolph, and Uwe Schwiegelshohn. 2004. Parallel job scheduling—a status report. In *Workshop on Job Scheduling Strategies for Parallel Processing*. Springer, New York, NY, USA, 1–16.

[11] Dimitrios Filippopoulos and Helen Karatza. 2007. An M/M/2 parallel system model with pure space sharing among rigid jobs. *Mathematical and Computer Modelling* 45, 5 (2007), 491 – 530.

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

[13] Javad Ghaderi. 2016. Randomized algorithms for scheduling VMs in the cloud. In *IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications*. 1–9.

[14] Peter W Glynn, Assaf Zeevi, et al. 2008. Bounding stationary expectations of Markov processes. *Markov processes and related topics: a Festschrift for Thomas G. Kurtz* 4 (2008), 195–214.

[15] Isaac Grosof and Mor Harchol-Balter. 2023. 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* (Orlando, FL, USA) *(ApPLIED 2023)*. Association for Computing Machinery, New York, NY, USA, Article 7, 5 pages. https://doi.org/10.1145/3584684.3597264

[16] Isaac Grosof, Mor Harchol-Balter, and Alan Scheller-Wolf. 2020. Stability for two-class multiserver-job systems. *arXiv preprint arXiv:2010.00631* (2020).

[17] Isaac Grosof, Mor Harchol-Balter, and Alan Scheller-Wolf. 2022. WCFS: A new framework for analyzing multiserver systems. *Queueing Systems* (2022).

[18] Isaac Grosof, Mor Harchol-Balter, and Alan Scheller-Wolf. 2023. New stability results for multiserver-job models via product-form saturated systems. *MAthematical performance Modeling and Analysis (MAMA)* 4, 6 (2023), 1.

[19] Isaac Grosof, Ziv Scully, Mor Harchol-Balter, and Alan Scheller-Wolf. 2022. Optimal Scheduling in the Multiserver-Job Model under Heavy Traffic. *Proc. ACM Meas. Anal. Comput. Syst.* 6, 3, Article 51 (dec 2022), 32 pages. https://doi.org/10.1145/3570612

[20] Varun Gupta, Mor Harchol-Balter, Alan Scheller Wolf, and Uri Yechiali. 2006. Fundamental characteristics of queues with fluctuating load. In *Proceedings of the joint international conference on Measurement and modeling of computer systems*. 203–215.

[21] Bruce Hajek. 1982. Hitting-time and occupation-time bounds implied by drift analysis with applications. *Advances in Applied Probability* 14, 3 (1982), 502–525. https://doi.org/10.2307/1426671

[22] Yige Hong. 2022. Sharp Zero-Queueing Bounds for Multi-Server Jobs. *SIGMETRICS Perform. Eval. Rev.* 49, 2 (jan 2022), 66–68.

[23] James Patton Jones and Bill Nitzberg. 1999. Scheduling for Parallel Supercomputing: A Historical Perspective of Achievable Utilization. In *Job Scheduling Strategies for Parallel Processing*, Dror G. Feitelson and Larry Rudolph (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 1–16.

[24] Charles Knessl and Yongzhi Peter Yang. 2002. An exact solution for an M (t)/M (t)/1 queue with time-dependent arrivals and service. *Queueing systems* 40 (2002), 233–245.

[25] David M Lucantoni and Marcel F Neuts. 1994. Some steady-state distributions for the MAP/SM/1 queue. *Stochastic Models* 10, 3 (1994), 575–598.

[26] Syed Hamid Hussain Madni, Muhammad Shafie Abd Latiff, Mohammed Abdullahi, Shafi'i Muhammad Abdulhamid, and Mohammed Joda Usman. 2017. Performance comparison of heuristic algorithms for task scheduling in IaaS cloud computing environment. *PLOS ONE* 12, 5 (05 2017), 1–26. https://doi.org/10.1371/journal.pone.0176321

[27] S. T. Maguluri and R. Srikant. 2014. Scheduling Jobs With Unknown Duration in Clouds. *IEEE/ACM Transactions on Networking* 22, 6 (2014), 1938–1951.

[28] Siva Theja Maguluri and R. Srikant. 2016. Heavy traffic queue length behavior in a switch under the MaxWeight algorithm. 6, 1 (2016), 211–250.

[29] William A Massey. 1985. Asymptotic analysis of the time dependent M/M/1 queue. *Mathematics of Operations Research* 10, 2 (1985), 305–327.

[30] Sean Meyn. 2008. *Control techniques for complex networks*. Cambridge University Press.

[31] Isi Mitrani and Ram Chakka. 1995. Spectral expansion solution for a class of Markov models: Application and comparison with the matrix-geometric method. *Performance Evaluation* 23, 3 (1995), 241–260.

[32] Evsey Morozov and Alexander Rumyantsev. 2016. Stability Analysis of a MAP/M/s Cluster Model by Matrix-Analytic Method. In *Computer Performance Engineering*, Dieter Fiems, Marco Paolieri, and Agapios N. Platis (Eds.). Springer International Publishing, Cham, 63–76.

[33] Marcel F Neuts. 1966. The single server queue with Poisson input and semi-Markov service times. *Journal of Applied Probability* 3, 1 (1966), 202–230.
[34] GF Newell. 1968. Queues with time-dependent arrival rates. III—A mild rush hour. *Journal of Applied Probability* 5, 3 (1968), 591–606.
[35] GF Newell. 1968. Queues with time-dependent arrival rates. II—The maximum queue and the return to equilibrium. *Journal of Applied Probability* 5, 3 (1968), 579–590.
[36] Gordon Frank Newell. 1968. Queues with time-dependent arrival rates I—the transition through saturation. *Journal of Applied Probability* 5, 2 (1968), 436–451.
[37] Edwin Peng. 2022. Exact Response Time Analysis of Preemptive Priority Scheduling with Switching Overhead. *ACM SIGMETRICS Performance Evaluation Review* 49, 2 (2022), 72–74.
[38] Efrat Perel and Uri Yechiali. 2008. Queues where customers of one queue act as servers of the other queue. *Queueing Systems* 60 (2008), 271–288.
[39] Konstantinos Psychas and Javad Ghaderi. 2018. Randomized Algorithms for Scheduling Multi-Resource Jobs in the Cloud. *IEEE/ACM Transactions on Networking* 26, 5 (2018), 2202–2215.
[40] Alexander Rumyantsev. 2020. Stability of multiclass multiserver models with automata-type phase transitions. In *Proceedings of the second international workshop on stochastic modeling and applied research of technology (SMARTY 2020)*, Vol. 2792. 213–225.
[41] Alexander Rumyantsev, Robert Basmadjian, Sergey Astafiev, and Alexander Golovin. 2022. Three-level modeling of a speed-scaling supercomputer. *Annals of Operations Research* (2022), 1–29.
[42] Alexander Rumyantsev and Evsey Morozov. 2017. Stability criterion of a multiserver model with simultaneous service. *Annals of Operations Research* 252, 1 (2017), 29–39.
[43] Leszek Sliwko. 2019. A Taxonomy of Schedulers–Operating Systems, Clusters and Big Data Frameworks. *Global Journal of Computer Science and Technology* (2019).
[44] Rayadurgam Srikant and Lei Ying. 2013. *Communication networks: an optimization, control, and stochastic networks perspective*. Cambridge University Press.
[45] Srividya Srinivasan, Rajkumar Kettimuthu, Vijay Subramani, and Ponnuswamy Sadayappan. 2002. Characterization of backfilling strategies for parallel job scheduling. In *Proceedings. International Conference on Parallel Processing Workshop*. 514–519.
[46] Muhammad Tirmazi, Adam Barker, Nan Deng, Md E. Haque, Zhijing Gene Qin, Steven Hand, Mor Harchol-Balter, and John Wilkes. 2020. Borg: The next Generation. In *Proceedings of the Fifteenth European Conference on Computer Systems* (Heraklion, Greece) *(EuroSys '20)*. Association for Computing Machinery, New York, NY, USA, Article 30, 14 pages.
[47] Rein Vesilo, Mor Harchol-Balter, and Alan Scheller-Wolf. 2022. Scaling properties of queues with time-varying load processes: extensions and applications. *Probability in the Engineering and Informational Sciences* 36, 3 (2022), 690–731.
[48] Juan Wang and Wenming Guo. 2009. The Application of Backfilling in Cluster Systems. In *2009 WRI International Conference on Communications and Mobile Computing*, Vol. 3. 55–59.
[49] Weina Wang, Qiaomin Xie, and Mor Harchol-Balter. 2021. Zero Queueing for Multi-Server Jobs. In *Abstract Proceedings of the 2021 ACM SIGMETRICS / International Conference on Measurement and Modeling of Computer Systems* (Virtual Event, China) *(SIGMETRICS '21)*. Association for Computing Machinery, New York, NY, USA, 13–14.
[50] Ury Yechiali and Pinhas Naor. 1971. Queuing problems with heterogeneous arrivals and service. *Operations Research* 19, 3 (1971), 722–734.

Appendix A. Finiteness of Δ, and the conditions for drift lemma

Lemma A.1. *The relative completion function*

$$\Delta_\pi(y_1, y_2) = \lim_{t\to\infty} \mathbb{E}[C_\pi(y_1, t) - C_\pi(y_2, t)]$$

is well-defined and finite for any pair of states y_1 and y_2 of the service process π.

Proof. Throughout this proof, we leave the subscript π implicit.

To characterize $\mathbb{E}[C(y_1, t) - C(y_2, t)]$, we construct a coupling between the two instances of the service process π, starting with initial states y_1 and y_2. We let the two chains transition independently when their states are different, and let them transition identically once their states become the same. Let τ be the time that the states of the two systems become the same. Because the two systems remain identical after τ, for any $t \geq 0$,

$$C(y_1, t) - C(y_2, t) = C(y_1, \min(t, \tau)) - C(y_2, \min(t, \tau)).$$

We assume that the system π is irreducible. Because each system is irreducible, the joint Markov chain of two systems is also irreducible and $\tau < \infty$ almost surely. Therefore,

$$\lim_{t\to\infty} \mathbb{E}[C(y_1, t) - C(y_2, t)] = \mathbb{E}[C(y_1, \tau) - C(y_2, \tau)].$$

The RHS of the above equality is clearly finite. □

Now we show that for any Markov chain η,

$$\mathbb{E}[G^\eta \circ f(Q^\eta, Y^\eta)] = 0. \tag{2}$$

The lemma below is implied by [14, Proposition 3]:

Lemma 3.2. *Let f be a real-valued function of the state of a Markov chain η. Assume that the transition rates of the Markov chain η are uniformly bounded, and $\mathbb{E}[f(Q^\eta, Y^\eta)] < \infty$. Then*

$$\mathbb{E}_{(q,y)\sim(Q^\eta,Y^\eta)}[G^\eta \circ f(q,y)] = 0. \tag{2}$$

To check that the conditions of Lemma 3.2 hold for the At-least-k and MSJ systems, first notice that their transitions rates are both uniformly bounded. In particular, the transition rates of the At-least-k system are uniformly bounded by $\lambda + \max_y \sum_{y',a} \mu^{\mathrm{Ak}}_{y,y',a}$, and the transition rates of the MSJ system are uniformly bounded by $\lambda + \max_{y,b} \sum_{y',a} \mu^{\mathrm{MSJ}}_{y,y',a,b}$. Therefore we only need to check that each f used in the paper has finite steady-state expectations in At-least-k and MSJ systems, i.e.

$$\mathbb{E}[f(Q^{\mathrm{Ak}}, Y^{\mathrm{Ak}})] < \infty,$$
$$\mathbb{E}[f(Q^{\mathrm{MSJ}}, Y^{\mathrm{MSJ}})] < \infty.$$

The following lemma shows that a function f has finite expectations in the At-least-k and MSJ system as long as it grows at a polynomial rate in q, which is true for all f which we will apply Lemma 3.2 to.

Lemma A.2. *Consider the MMSR system controlled by the Markov chain π and the MSJ system. For any positive integer m,*

$$\mathbb{E}[(Q^\pi)^m] < \infty,$$
$$\mathbb{E}[(Q^{\mathrm{MSJ}})^m] < \infty.$$

To prove the lemma, we need [21, Theorem 2.3], restated as below:

Lemma A.3. *Consider a Markov chain η with uniformly bounded total transition rates, and a Lyapunov function V that satisfy the conditions below: $V(q,y) \geq 0$; there exists a constant $b, \gamma > 0$ such that whenever $V(q,y) \geq b$,*

$$G^\eta \circ V(q,y) \leq -\gamma; \tag{A.1}$$

there exists $d > 0$ such that

$$\max_{\text{next state } (q',y')} |V(q',y') - V(q,y)| \leq d. \tag{A.2}$$

Then there exists $\theta > 0$ such that

$$\mathbb{E}[e^{\theta V(Q^\eta, Y^\eta)}] < \infty. \tag{A.3}$$

Now we prove Lemma A.2.

Proof of Lemma A.2. We first prove the lemma for the MMSR system controlled by the Markov chain π.

Let $\Delta_{\max}$ be the maximal absolute value of $\Delta_\pi(y)$ for any y in the state spaces of $\mathbb{Y}^\pi$, which must be finite due to Lemma A.1 and the fact that there are only finitely many possible y.

We construct the Lyapunov function $V(q,y) = (q - \Delta(y))^+$. We first check the conditions of Lemma A.3 for the MMSR system controlled by π. To check (A.1), we let $b = 1 + 2\Delta_{\max}$ and $\gamma = \lambda^* - \lambda$. If $V(q,y) \geq b$, we must have $q \geq 1 + \Delta_{\max}$; for any state (q',y') that the system can jump to after one transition, $V(q',y') \geq q' - \Delta(y') \geq q - 1 - \Delta_{max} \geq 0$, so $V(q',y') = q' - \Delta(y')$. Therefore,

$$G^\pi \circ V(q,y) = G^\pi \circ (q - \Delta(y)) = \lambda - \lambda^* = -\gamma.$$

It is also easy to see that (A.2) holds with $d = 1 + 2\Delta_{\max}$. Therefore, by Lemma A.3, there exists $\theta > 0$ such that

$$\mathbb{E}[e^{\theta V(Q^\pi, Y^\pi)}] < \infty.$$

Observe that $e^{\theta V(q,y)}$ grows with q exponentially fast. Therefore, for any positive integer m,

$$q^m = O(e^{\theta V(q,y)}),$$
$$\mathbb{E}[(Q^\pi)^m] < \infty.$$

The analysis of the MSJ system is similar to the analysis of the At-least-k system, which is a special case of the MMSR system with $\pi = \text{Sat}$. We consider the Lyapunov function

$$V(q, y) = \begin{cases} \text{if } q > 1 & (q - \Delta_{\text{Sat}}(y))^+ \\ \text{otherwise} & 0, \end{cases}$$

and check the conditions of Lemma A.3. Notice that $G^{\text{MSJ}} \circ V(q, y) = G^{\text{Ak}} \circ V(q, y)$ for any $q \geq 1$, so the rest of the argument is verbatim. □

Appendix B. Simplified Saturated System

For clarity, we refer to the previously-defined saturated system, defined in Section 3.5, as the "main saturated system."

While the main saturated system is a finite-state system, it can have a very large number of possible states. We therefore introduce the *simplified saturated system* (SSS), a new closed system with identical behavior, but smaller state space. The SSS can be more amenable to theoretical analysis, such as in the case of the product-form result in [16].

The simplified saturated system is a closed system which always contains jobs with total server need $\geq k$, and contains the minimal number of jobs to reach that threshold. Whenever a job completes, the system admits new jobs until the total server need is $\geq k$. Jobs are served in FCFS order. Note that at most one job in the system is not in service.

In particular, a state of the SSS consists of a multiset of job states for the jobs in service, plus the server need of the job not in service, if any. The total server need of these jobs is just enough to be $\geq k$.

For instance, consider a system with $k = 30$ jobs, and server needs either 3 or 10, and exponential durations. The main saturated system has state space $\mathbb{Y}^{\text{Sat}} = \{3, 10\}^{30}$, with over a billion states. In contrast, the simplified saturated system has 13 states. We will write each state as a triple, consisting of the server need of the job not in service, and the number of 3-server and 10-server jobs in service. Then the state space of the SSS is:

$$\mathbb{Y}^{\text{SSS}} = \{[\emptyset, 0, 3], [10, 1, 2], [10, 2, 2], [3, 3, 2], [10, 3, 2], [10, 4, 1], [10, 5, 1],$$
$$[3, 6, 1], [10, 6, 1], [10, 7, 0], [10, 8, 0], [10, 9, 0], [\emptyset, 10, 0]\}.$$

Despite its much smaller state space, the SSS has essentially identical behavior to the main saturated system:

Lemma 3.3. *There exists a coupling under which the main saturated system and simplified saturated system have identical completions.*

Proof. To form the coupling, let us sample in advance the entire arrival sequence: For each arrival, we pre-sample which initial state it will arrive in.

Next, we initialize both systems based on this arrival sequence: For the main saturated system, the first k jobs are initially present, while for the simplified saturated system, a subset of those jobs are initially present. Note that the set of jobs in service in the main saturated system is identical to the set of jobs in service in the simplified saturated system, because the total server need of jobs in service is at most k.

Note that the ordering of the jobs in service does not affect any transitions, so the fact that SSS does not track this information poses no obstacle. We will ensure that the set of jobs in service in the two systems is identical throughout time.

Next, we couple the two systems' completions and job state transitions to be identical. Jobs' states can only change while those jobs are in service, so this coupling is valid as long as the set of jobs in service is identical in both systems. Finally, whenever a pair of jobs completes, new jobs are generated according to the shared global arrival sequence. This ensures that the jobs that enter service are identical in the two systems.

By construction, the set of jobs in service is always identical in the two systems. Under this coupling, the completion moments are also identical in the two systems. □

Appendix C. Closed-form formulas for $\lambda^*, Y_d, \Delta(y)$

Our result on the M/M/1 with Markovian Service Rate (MMSR), Theorem 4.1, characterizes mean response time in the MMSR-π system in terms of the following quantities:

- λ^*_π, the threshold of the stability region of the MMSR-π system,
- Y^π_d, the departure-average steady state of the service process π, and
- $\Delta_\pi(y)$, the relative completions function of the service process π.

Similarly, our result on the MSJ system, Theorem 4.2, characterizes mean response time in the MSJ system in terms of $\lambda^*, Y^{\text{Sat}}_d$, and $\Delta_{\text{Sat}}(y)$, or equivalently in terms of $\lambda^*, Y^{\text{SSS}}_d$, and $\Delta_{\text{SSS}}(y)$, by Corollary 4.1.

In this section, we demonstrate how to explicitly calculate λ^*_π, Y^π_d, and $\Delta_\pi(y)$ by solving a system of linear equations, and walk through this exercise for a specific parameterized setting, giving an explicit, closed-form expression for mean response time within the given parameterized setting.

Appendix C.1. Solving for λ^*_π, Y^π_d, and $\Delta_\pi(y)$

First, we solve the continuous-time balance equations for the service process π to determine the time-average steady state Y^π:

$$\forall y \in \mathbb{Y}^\pi, \quad \mathbb{P}(Y^\pi = y)\mu_{y,\cdot,\cdot} = \sum_{y' \in \mathbb{Y}^\pi} \mathbb{P}(Y^\pi = y')\mu_{y',y,\cdot}, \tag{C.1}$$

$$\sum_{y \in \mathbb{Y}^\pi} \mathbb{P}(Y^\pi = y) = 1.$$

Next, we calculate the throughput X^π of the service process π, which by prior results [2] is equal to the threshold of the MMSR-π stability region, λ^*_π:

$$\lambda^*_\pi = X^\pi = \mathbb{E}_{y \sim Y^\pi}[\mu_{y,\cdot,1}] = \sum_{y \in \mathbb{Y}^{\text{Ak}}} \mu_{y,\cdot,1}\mathbb{P}(Y^\pi = y). \tag{C.2}$$

Next, we calculate the departure-average steady state Y^π_d. Recall that Y^π_d is the steady-state distribution of the embedded DTMC which samples states just after each departure from π. To calculate $\mathbb{P}(Y^\pi_d = y)$ from $\mathbb{P}(Y^\pi = d)$, we divide by the expected time spent in state y per visit, $\frac{1}{\mu_{y,\cdot,\cdot}}$, and multiply by the probability that the transition into state y was a completion:

$$\mathbb{P}(Y^\pi_d = y) = Z^\pi \mathbb{P}(Y^\pi = y)\frac{\mu_{\cdot,y,1}}{\mu_{\cdot,y,\cdot}}\mu_{y,\cdot,\cdot}, \tag{C.3}$$

where Z^π is a normalization constant.

Finally, we calculate the relative completions function $\Delta_\pi(y)$. To do so, we use the system of equations given in Corollary D.1:

$$\Delta_\pi(y) = \frac{\mu_{y,\cdot,1} - \lambda^*_\pi}{\mu_{y,\cdot,\cdot}} + \sum_{y'} \frac{\mu_{y,y',\cdot}}{\mu_{y,\cdot,\cdot}}\Delta_\pi(y'). \tag{C.4}$$

This system of equations characterizes $\Delta_\pi(y)$ up to an additive offset. To uniquely determine $\Delta_\pi(y)$, we use the fact that $\Delta(Y^\pi) = 0$:

$$0 = \Delta(Y^\pi) = \sum_y \mathbb{P}(Y^\pi = y)\Delta(y). \tag{C.5}$$

Appendix C.2. Specific parameterized example: Two servers, arbitrary service rates

We now walk through the process of determining $\lambda^*, Y_d^{\text{SSS}}$, and $\Delta_{\text{SSS}}(y)$, in a parameterized MSJ setting, allowing us to use Theorem 4.2 to explicitly characterize mean response time $\mathbb{E}[T^{\text{MSJ}}]$.

Consider an MSJ system with $k = 2$ servers, and jobs with server need either 1 or 2. p_1 fraction of jobs have server need 1, and $p_2 = 1 - p_1$ have server need 2. Let server need 1 jobs have service duration $Exp(\mu_1)$, and server need 2 jobs have service duration $Exp(\mu_2)$.

Note that this setting is a generalized, parameterized version of the setting discussed in Section 8. The same methods can handle any MSJ setting with phase-type service durations - we merely choose this one as a clean example.

The Simplified Saturated System (SSS) for this setting has three states: $[1,1]$, $[1,2]$, and $[2]$. Between these states, we have the following transition rates:

$$\mu_{[1,1],[1,1],1} = 2\mu_1 p_1, \quad \mu_{[1,1],[1,2],1} = 2\mu_1 p_2, \quad \mu_{[1,2],[2],1} = \mu_1,$$
$$\mu_{[2],[1,1],1} = \mu_2 p_1^2, \quad \mu_{[2],[1,2],1} = \mu_2 p_1 p_2, \quad \mu_{[2],[2],1]} = \mu_2 p_2.$$

Note that all transitions in this setting involve a completion. This is due to the fact that the service durations are exponential. For more complex service duration distributions, there would also be non-completion transitions.

Now, we calculate the time-average steady state Y, using (C.1):

$$\begin{aligned}
2\mu_1\mathbb{P}(Y = [1,1]) &= 2\mu_1 p_1 \mathbb{P}(Y = [1,1]) + \mu_2 p_1^2 \mathbb{P}(Y = [2]),\\
\mu_1\mathbb{P}(Y = [1,2]) &= 2\mu_1 p_2 \mathbb{P}(Y = [1,1]) + \mu_2 p_1 p_2 \mathbb{P}(Y = [2]),\\
\mu_2\mathbb{P}(Y = [2]) &= \mu_1\mathbb{P}(Y = [1,2]) + \mu_2 p_2 \mathbb{P}(Y = [2]),\\
\mathbb{P}(Y = [1,1]) + \mathbb{P}(Y = [1,2]) &+ \mathbb{P}(Y = [2]) = 1.
\end{aligned}$$

Solving, we find that

$$\begin{aligned}
\mathbb{P}(Y = [1,1]) &= \frac{\mu_2 p_1^2}{\mu_2 p_1^2 + 2\mu_2 p_1 p_2 + 2\mu_1 p_2},\\
\mathbb{P}(Y = [1,2]) &= \frac{2\mu_2 p_1 p_2}{\mu_2 p_1^2 + 2\mu_2 p_1 p_2 + 2\mu_1 p_2},\\
\mathbb{P}(Y = [2]) &= \frac{2\mu_1 p_2}{\mu_2 p_1^2 + 2\mu_2 p_1 p_2 + 2\mu_1 p_2}.
\end{aligned}$$

Next, we use (C.2) to calculate λ^*, the threshold of the stability region:

$$\lambda^* = 2\mu_1\mathbb{P}(Y = [1,1]) + \mu_1\mathbb{P}(Y = [1,2]) + \mu_2\mathbb{P}(Y = [2]) = \frac{2\mu_1\mu_2}{\mu_2 p_1^2 + 2\mu_2 p_1 p_2 + 2\mu_1 p_2}.$$

Next, we use (C.3) to calculate Y_d, the departure-average steady-state. Note that all transitions are completions, so (C.3) simplifies to the following:

$$\begin{aligned}
\mathbb{P}(Y_d = y) &= \frac{1}{\lambda^*}\mathbb{P}(Y = y)\mu_{y,\cdot,\cdot},\\
\mathbb{P}(Y_d = [1,1]) &= p_1^2,\\
\mathbb{P}(Y_d = [1,2]) &= p_1 p_2,\\
\mathbb{P}(Y_d = [2]) &= p_2.
\end{aligned} \tag{C.6}$$

Note that this is a product-form distribution. This is a special case of the product-form behavior that was established for the general 2-class exponential MSJ setting [16, 18].

Finally, we use (C.4) and (C.5) to derive $\Delta(y)$ for each state y. Because all transitions are completions, (C.4) simplifies to the following:

$$\Delta(y) = 1 - \frac{\lambda^*}{\mu_{y,\cdot,\cdot}} + \sum_{y'} \frac{\mu_{y,y',\cdot}}{\mu_{y,\cdot,\cdot}} \Delta(y').$$

Now, let's substitute in our expressions for λ^* and the transition rates and simplify. First we characterize $\Delta([1,1])$:

$$\begin{aligned}
\Delta([1,1]) &= 1 - \frac{1}{2\mu_1} \frac{2\mu_1\mu_2}{\mu_2 p_1^2 + 2\mu_2 p_1 p_2 + 2\mu_1 p_2} + p_1\Delta([1,1]) + p_2\Delta([1,2]), \\
p_2\Delta([1,1]) &= 1 - \frac{\mu_2}{\mu_2 p_1^2 + 2\mu_2 p_1 p_2 + 2\mu_1 p_2} + p_2\Delta([1,2]), \\
\Delta([1,1]) &= \frac{1}{p_2} \frac{\mu_2 p_1^2 + 2\mu_2 p_1 p_2 + 2\mu_1 p_2 - \mu_2}{\mu_2 p_1^2 + 2\mu_2 p_1 p_2 + 2\mu_1 p_2} + \Delta([1,2]) \\
&= \frac{1}{p_2} \frac{\mu_2 p_1 p_2 + (2\mu_1 - \mu_2) p_2}{\mu_2 p_1^2 + 2\mu_2 p_1 p_2 + 2\mu_1 p_2} + \Delta([1,2]) \\
&= \frac{\mu_2 p_1 + 2\mu_1 - \mu_2}{\mu_2 p_1^2 + 2\mu_2 p_1 p_2 + 2\mu_1 p_2} + \Delta([1,2]) \\
&= \frac{2\mu_1 - \mu_2 p_2}{\mu_2 p_1^2 + 2\mu_2 p_1 p_2 + 2\mu_1 p_2} + \Delta([1,2]).
\end{aligned} \tag{C.7}$$

Next $\Delta([1,2])$:

$$\begin{aligned}
\Delta([1,2]) &= 1 - \frac{1}{\mu_1} \frac{2\mu_1\mu_2}{\mu_2 p_1^2 + 2\mu_2 p_1 p_2 + 2\mu_1 p_2} + \Delta([2]) \\
&= 1 - \frac{2\mu_2}{\mu_2 p_1^2 + 2\mu_2 p_1 p_2 + 2\mu_1 p_2} + \Delta([2]) \\
&= \frac{\mu_2 p_1^2 + 2\mu_2 p_1 p_2 + 2\mu_1 p_2 - 2\mu_2}{\mu_2 p_1^2 + 2\mu_2 p_1 p_2 + 2\mu_1 p_2} + \Delta([2]) \\
&= \frac{-\mu_2 p_1^2 + 2(\mu_1 - \mu_2) p_2}{\mu_2 p_1^2 + 2\mu_2 p_1 p_2 + 2\mu_1 p_2} + \Delta([2])
\end{aligned} \tag{C.8}$$

Finally, we characterize $\Delta([2])$:

$$\begin{aligned}
\Delta([2]) &= 1 - \frac{1}{\mu_2} \frac{2\mu_1\mu_2}{\mu_2 p_1^2 + 2\mu_2 p_1 p_2 + 2\mu_1 p_2} + p_1^2\Delta([1,1]) + p_1 p_2\Delta([1,2]) + p_2\Delta([2]), \\
p_1\Delta([2]) &= 1 - \frac{2\mu_1}{\mu_2 p_1^2 + 2\mu_2 p_1 p_2 + 2\mu_1 p_2} + p_1^2\Delta([1,1]) + p_1 p_2\Delta([1,2]), \\
\Delta([2]) &= \frac{1}{p_1} \frac{\mu_2 p_1^2 + 2\mu_2 p_1 p_2 + 2\mu_1 p_2 - 2\mu_1}{\mu_2 p_1^2 + 2\mu_2 p_1 p_2 + 2\mu_1 p_2} + p_1\Delta([1,1]) + p_2\Delta([1,2]) \\
&= \frac{1}{p_1} \frac{(\mu_2 - 2\mu_1) p_1^2 + 2(\mu_2 - \mu_1) p_1 p_2}{\mu_2 p_1^2 + 2\mu_2 p_1 p_2 + 2\mu_1 p_2} + p_1\Delta([1,1]) + p_2\Delta([1,2]) \\
&= \frac{(\mu_2 - 2\mu_1) p_1 + 2(\mu_2 - \mu_1) p_2}{\mu_2 p_1^2 + 2\mu_2 p_1 p_2 + 2\mu_1 p_2} + p_1\Delta([1,1]) + p_2\Delta([1,2]).
\end{aligned} \tag{C.9}$$

Note that our final equations, (C.7), (C.8), and (C.9), are redundant: We can omit any one and still

calculate $\Delta(y)$, up to an additive constant. From these equations, we find that

$$\begin{aligned}\Delta([1,1]) &= \frac{2\mu_1 - \mu_2 p_2}{\mu_2 p_1^2 + 2\mu_2 p_1 p_2 + 2\mu_1 p_2} + C,\\ \Delta([1,2]) &= C,\\ \Delta([2]) &= \frac{\mu_2 p_1^2 + 2(\mu_2 - \mu_1)p_2}{\mu_2 p_1^2 + 2\mu_2 p_1 p_2 + 2\mu_1 p_2} + C,\end{aligned}$$

where C is an additive constant to be determined. To find C, we use (C.5). Substituting our known values, we find that

$$\begin{aligned}0 &= \frac{\mu_2 p_1^2(2\mu_1 - \mu_2 p_2)}{(\mu_2 p_1^2 + 2\mu_2 p_1 p_2 + 2\mu_1 p_2)^2} + \frac{2\mu_1 p_2(\mu_2 p_1^2 + 2(\mu_2 - \mu_1)p_2)}{(\mu_2 p_1^2 + 2\mu_2 p_1 p_2 + 2\mu_1 p_2)^2} + C,\\ -C(\mu_2 p_1^2 + 2\mu_2 p_1 p_2 + 2\mu_1 p_2)^2 &= \mu_2 p_1^2(2\mu_1 - \mu_2 p_2) + 2\mu_1 p_2(\mu_2 p_1^2 + 2(\mu_2 - \mu_1)p_2)\\ &= -4\mu_1^2 p_2^2 + 2\mu_1\mu_2(p_1^2 + p_1^2 p_2 + 2p_2^2) - \mu_2^2 p_1^2 p_2,\\ C &= \frac{4\mu_1^2 p_2^2 - 2\mu_1\mu_2(p_1^2 + p_1^2 p_2 + 2p_2^2) + \mu_2^2 p_1^2 p_2}{(\mu_2 p_1^2 + 2\mu_2 p_1 p_2 + 2\mu_1 p_2)^2}.\end{aligned}$$

We can therefore derive expressions for $\Delta(y)$:

$$\begin{aligned}\Delta([1,1]) &= \frac{2p_2(2\mu_1^2(1+p_2) - \mu_1\mu_2(-2p_1 + p_1^2 + 3p_2) - \mu_2^2 p_1 p_2)}{(\mu_2 p_1^2 + 2\mu_2 p_1 p_2 + 2\mu_1 p_2)^2},\\ \Delta([1,2]) &= \frac{4\mu_1^2 p_2^2 - 2\mu_1\mu_2(p_1^2 + p_1^2 p_2 + 2p_2^2) + \mu_2^2 p_1^2 p_2}{(\mu_2 p_1^2 + 2\mu_2 p_1 p_2 + 2\mu_1 p_2)^2},\\ \Delta([2]) &= \frac{\mu_2 p_1(-2\mu_1(1+p_2^2) + \mu_2(p_1^2 p_1^2 p_2 + 3p_2 + p_2^2))}{(\mu_2 p_1^2 + 2\mu_2 p_1 p_2 + 2\mu_1 p_2)^2}.\end{aligned}$$

Finally, we can apply our expressions for Y_d, (C.6), to calculate $\Delta(Y_d)$:

$$\begin{aligned}\Delta(Y_d) &= \mathbb{E}_{y\sim Y_d}[\Delta(y)]\\ &= \Delta([1,1])\mathbb{P}(Y_d = [1,1]) + \Delta([1,2])\mathbb{P}(Y_d = [1,2]) + \Delta([2])\mathbb{P}(Y_d = [2])\\ &= p_1^2\Delta([1,1]) + p_1 p_2\Delta([1,2]) + p_2\Delta([2])\\ &= \frac{p_1 p_2(4\mu_1^2 - 2\mu_1\mu_2(1+3p_2) + \mu_2^2(1 + p_2 + 2p_2^2))}{(\mu_2 p_1^2 + 2\mu_2 p_1 p_2 + 2\mu_1 p_2)^2}.\end{aligned}$$

Having explicitly characterized λ^* and $\Delta(Y_d)$, our main result, Theorem 4.2, gives an explicit, closed-form expression for mean response time:

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

Appendix D. MMSR Lemmas

Lemma D.1. *Consider the MMSR system controlled by the Markov chain π. For any state $y \in \mathbb{Y}^\pi$,*

$$G^\pi \circ \Delta_\pi(y, Y^\pi) = \lambda_\pi^* - \mu^\pi_{y,\cdot,1}. \tag{D.1}$$

Proof. Recall that by the definition of the generator, $G^\pi \circ \Delta_\pi(y, Y^\pi)$ is given by

$$G^\pi \circ \Delta_\pi(y, Y^\pi) = \lim_{t\to 0}\frac{1}{t}\mathbb{E}[\Delta_\pi(Y^\pi(t), Y^\pi) - \Delta_\pi(y, Y^\pi)|Y^\pi(0) = y]. \tag{D.2}$$

To figure out $\mathbb{E}[\Delta_\pi(Y^\pi(t), Y^\pi) - \Delta_\pi(y, Y^\pi)|Y^\pi(0) = y]$, recall the definition that

$$\Delta_\pi(y, Y^\pi) = \lim_{t'\to\infty} \mathbb{E}[C_\pi(y, t') - \lambda^*_\pi t'],$$

where recall that $C_\pi(y, t')$ is the expected number of completion up to time t' of the MMSR system whose service process is controlled by the Markov chain π initializing in state y. Therefore, if we replace y by $Y^\pi(t)$ on the LHS of the above definition and take the expectation, we have

$$\begin{aligned}
&\mathbb{E}[\Delta_\pi(Y^\pi(t), Y^\pi)|Y^\pi(0) = y] \\
&= \lim_{t'\to\infty} \mathbb{E}[C_\pi(Y^\pi(t), t') - \lambda^*_\pi t'|Y^\pi(0) = y] \\
&= \lim_{t'\to\infty} \mathbb{E}[C_\pi(y, t + t') - C_\pi(y, t) - \lambda^*_\pi t'|Y^\pi(0) = y],
\end{aligned}$$

where in the second equality we have used the fact that

$$\begin{aligned}
\mathbb{E}[C_\pi(y, t + t')] &= \mathbb{E}[C_\pi(y, t) + C_\pi([Y_\pi(t) \mid Y_\pi(0) = y], t')] \\
\mathbb{E}[C_\pi(Y_\pi(t), t') \mid Y_\pi(0) = y] &= \mathbb{E}[C_\pi(y, t + t')] - \mathbb{E}[C_\pi(y, t)].
\end{aligned} \tag{D.3}$$

(D.3) simply splits up the completions from time 0 to $t + t'$ into the completions from time 0 to t, and the completions from time t to $t + t'$.

Therefore,

$$\begin{aligned}
&\mathbb{E}[\Delta_\pi(Y^\pi(t), Y^\pi)) - \Delta_\pi(y, Y^\pi)|Y^\pi(0) = y] \\
&= \lim_{t'\to\infty} \mathbb{E}[C_\pi(y, t + t') - C_\pi(y, t) - \lambda^*_\pi t'] - \lim_{t'\to\infty} \mathbb{E}[C_\pi(y, t') - \lambda^*_\pi t'] \\
&= \lim_{t'\to\infty} \mathbb{E}[C_\pi(y, t + t') - C_\pi(y, t) - \lambda^*_\pi t'] - \lim_{t'\to\infty} \mathbb{E}[C_\pi(y, t + t') - \lambda^*_\pi t - \lambda^*_\pi t'] \\
&= \mathbb{E}[-C_\pi(y, t) + \lambda^*_\pi t],
\end{aligned}$$

where in the second inequality we replace t' with $t + t'$ in the second term, which will not change the limit because t' and $t + t'$ are both going to infinity. Plugging the above calculations into (D.2), we get

$$G^\pi \circ \Delta_\pi(y, Y^\pi) = \lim_{t\to 0} \frac{1}{t}\mathbb{E}[-C_\pi(y, t) + \lambda^*_\pi t] = -\mu^\pi_{y,\cdot,1} + \lambda^*_\pi,$$

where in the last inequality we use the fact that $\lim_{t\to 0} \frac{1}{t}\mathbb{E}[C_\pi(y, t)] = \mu^\pi_{y,\cdot,1}$ (the instantaneous completion rate at state y). □

Lemma D.2. *For any $f(q, y)$ which is a real-valued function of the state of the MMSR-π system,*

$$G^\pi \circ f(q, y) = \lambda \left(f(q + 1, y) - f(q, y)\right) + \sum_{\substack{y'\in\mathbb{Y}^\pi, \\ a\in\{0,1\}}} \mu^\pi_{y,y',a} \left(f((q - a)^+, y') - f(q, y)\right).$$

Proof. In this proof we omit π in the subscript of $\Delta_\pi(y)$ and in the superscript of $\mu^\pi_{y,y',a}$ for readability.

Recall the definition of the generator

$$G^\pi \circ f(q, y) = \lim_{t\to 0} \frac{1}{t}\mathbb{E}[f(Q^\pi(t), Y^\pi(t)) - f(q, y)|Q^\pi(0) = q, Y^\pi(0) = y],$$

which can be interpreted as the instantaneous rate of change of the function $f(Q^\pi(t), Y^\pi(t))$ when $(Q^\pi(t), Y^\pi(t))$ is initialized in (q, y). Note that $(Q^\pi(t), Y^\pi(t))$ can change either due to an arrival event, or a transition event of the Markov chain π. An arrival event happens with rate λ, and causes $Q^\pi(t)$ to change from q to $q + 1$, so arrival events contribute

$$\lambda \left(f(q + 1, y) - f(q, y)\right)$$

to $G^\pi \circ f(q,y)$. A transition event of the Markov chain π from y to $y' \in \mathbb{Y}^\pi$ accompanied by $a \in \{0,1\}$ completions happens with rate $\mu_{y,y',a}$. Such a event causes $(Q^\pi(t), Y^\pi(t))$ to change from (q,y) to $((q-a)^+, y')$, so it contributes

$$\mu_{y,y',a}\left(f((q-a)^+,y') - f(q,y)\right)$$

to $G^\pi \circ f(q,y)$, for each $y' \in \mathbb{Y}^{\mathrm{Ak}}$ and $a \in \{0,1\}$. This proves the expression in the lemma statement. □

As a corollary of Lemma D.1 and Lemma D.2, we can derive a forward recurrence for $\Delta_\pi(y) := \Delta_\pi(y, Y^\pi)$. Solving the resulting system of equations, together with the fact that $\Delta_\pi(Y^\pi) = 0$, gives the value of $\Delta_\pi(y)$.

Corollary D.1. *For any MMSR-π system and any state $y \in \mathbb{Y}^\pi$,*

$$\Delta_\pi(y) = \frac{\mu_{y,\cdot,1} - \lambda^*_\pi}{\mu_{y,\cdot,\cdot}} + \sum_{y'} \frac{\mu_{y,y',\cdot}}{\mu_{y,\cdot,\cdot}} \Delta(y'),$$

where $\mu_{y,\cdot,\cdot}$ is the total transition rate out of state y.

Moreover, if all transitions in π are associated with completions (if a always equals 1), then the recurrence simplifies:

$$\Delta_\pi(y) = 1 - \frac{\lambda^*_\pi}{\mu_{y,\cdot,1}} + \sum_{y'} \frac{\mu_{y,y',1}}{\mu_{y,\cdot,1}} \Delta(y').$$

Proof. Start with Lemma D.1:

$$G^\pi \circ \Delta_\pi(y) = \lambda^*_\pi - \mu^\pi_{y,\cdot,1}. \tag{D.4}$$

Here we write $\Delta_\pi(y)$ as a shorthand for $\Delta_\pi(y, Y^\pi)$.

Expand the left-hand side of (D.4) using Lemma D.2:

$$G^\pi \circ \Delta_\pi(y) = \sum_{y',a} \mu^\pi_{y,y',a}(\Delta_\pi(y') - \Delta_\pi(y)).$$

Note that Lemma D.2 simplifies because $\Delta_\pi(y)$ does not depend on q.

Now we can perform algebraic manipulation to complete the proof:

$$\begin{aligned}
\lambda^*_\pi - \mu^\pi_{y,\cdot,1} &= \sum_{y',a} \mu^\pi_{y,y',a}(\Delta_\pi(y') - \Delta_\pi(y)) \\
&= -\mu^\pi_{y,\cdot,\cdot}\Delta_\pi(y) + \sum_{y',a} \mu^\pi_{y,y',a}\Delta_\pi(y'), \\
\mu^\pi_{y,\cdot,\cdot}\Delta_\pi(y) &= \mu^\pi_{y,\cdot,1} - \lambda^*_\pi + \sum_{y',a} \mu^\pi_{y,y',a}\Delta_\pi(y'), \\
\Delta_\pi(y) &= \frac{\mu^\pi_{y,\cdot,1} - \lambda^*_\pi}{\mu^\pi_{y,\cdot,\cdot}} + \sum_{y',a} \frac{\mu^\pi_{y,y',a}}{\mu^\pi_{y,\cdot,\cdot}}\Delta_\pi(y').
\end{aligned}$$

Note that if all transitions are associated with completions, e.g. if $a = 1$, then $\mu^\pi_{y,\cdot,1} = \mu^\pi_{y,\cdot,\cdot}$ □

Lemma 5.1. *For any state (q,y) of the MMSR-π system,*

$$G^\pi \circ f^\pi_\Delta(q,y) = (\lambda - \lambda^*_\pi)q - \lambda\Delta_\pi(y) + \frac{1}{2}\lambda + \sum_{y',a} \mu^\pi_{y,y',a}\left(\frac{1}{2}(-a + u - \Delta_\pi(y'))^2 - \frac{1}{2}\Delta_\pi(y)^2\right). \tag{D.5}$$

Proof. In this proof we omit π in the subscript of $\Delta_\pi(y)$ and in the superscript of $\mu^\pi_{y,y',a}$ for readability.

To calculate $G^\pi \circ f^\pi_\Delta(q,y)$, we begin by applying Lemma D.2:

$$G^\pi \circ f^\pi_\Delta(q,y) = \lambda(q - \Delta(y) + \frac{1}{2}) \tag{D.6}$$

$$+ \sum_{y',a} \mu_{y,y',a} \left(\frac{1}{2} \left((q-a)^+ - \Delta(y') \right)^2 - \frac{1}{2} \left(q - \Delta(y) \right)^2 \right). \tag{D.7}$$

Recall that the unused service $u = \mathbb{1}\{q = 0 \wedge a = 1\}$, so $(q-a)^+ = q - a + u$. We can decompose (D.7) into two terms, with and without q:

$$(\text{D.7}) = q \sum_{y',a} \mu_{y,y',a} \left(-a + u - \Delta(y') + \Delta(y) \right) \tag{D.8}$$

$$+ \sum_{y',a} \mu_{y,y',a} \left(\frac{1}{2}(-a + u - \Delta(y'))^2 - \frac{1}{2}\Delta(y)^2 \right).$$

The coefficient of q in (D.8) can be simplified considerably using Lemma D.1.

$$\begin{aligned}
&\sum_{y',a} \mu_{y,y',a} \left(-a + u - \Delta(y') + \Delta(y) \right) \\
&= \sum_{y',a} \mu_{y,y',a}(-a) + \sum_{y',a} \mu_{y,y',a} u - \sum_{y',a} \mu_{y,y',a}(\Delta(y') - \Delta(y)) \\
&= -\mu_{y,\cdot,1} - G^{\text{Ak}} \circ \Delta(y) + \sum_{y',a} \mu_{y,y',a} u \\
&= -\mu_{y,\cdot,1} - (\lambda^*_\pi - \mu^{\text{Ak}}_{y,\cdot,1}) + \sum_{y',a} \mu_{y,y',a} u \\
&= -\lambda^*_\pi + \sum_{y',a} \mu_{y,y',a} u.
\end{aligned}$$

Note that either $u = 0$ or $q = 0$, because new jobs are only generated if the queue is empty. As a result, $qu = 0$. We can therefore further simplify the q-term in (D.8):

$$q(\sum_{y',a} \mu_{y,y',a} u - \lambda^*_\pi) = -q\lambda^*_\pi \tag{D.9}$$

Substituting (D.9) into (D.8), (D.8) into (D.7), and performing some rearrangement, we find that

$$G^\pi \circ f^\pi_\Delta(q,y) = (\lambda - \lambda^*_\pi)q - \lambda\Delta(y) + \frac{1}{2}\lambda + \sum_{y',a} \mu_{y,y',a} \left(\frac{1}{2}(-a + u - \Delta(y'))^2 - \frac{1}{2}\Delta(y)^2 \right). \qquad \square$$

Lemma D.3. *In the MMSR-π system, the departure average distribution Y^π_d is given by*

$$\frac{1}{\lambda^*_\pi} \mathbb{E}_{y \sim Y^\pi}[\mu^\pi_{y,y',1}] = \mathbb{P}(Y^\pi_d = y'). \tag{D.10}$$

Proof. We will show that

$$\mathbb{P}(Y^\pi_d = y') = \frac{1}{\lambda^*_\pi} \sum_y \mathbb{P}(Y^\pi = y) \mu_{y,y',1}.$$

As an intermediate step, let Y^π_{DTMC} be the transition-average steady state of the Markov chain π. $\mathbb{P}(Y^\pi_{DTMC} = y)$ is the fraction of state-visits that are visits to y, in the embedded DTMC of π.

Let $\mu_{y,\cdot,\cdot}$ be the total transition rate out of state y:

$$\mu_{y,\cdot,\cdot} = \sum_{y',a} \mu_{y,y',a}.$$

Note that the CTMC that controls Y^π and the DTMC that controls Y^π_{DTMC} visit the same states in the same order, but that Y^π stays in state y for $Exp(\mu_{y,\cdot,\cdot})$ time for each visit. As a result,

$$\mathbb{P}(Y^\pi = y) = a^\pi \mathbb{P}(Y^\pi_{DTMC} = y) \frac{1}{\mu_{y,\cdot,\cdot}}.$$

where a^π is a normalization constant. Specifically, a^π is the long-term transition rate, which can be calculated as the reciprocal of the average time per visit to a state:

$$a^\pi = \left(\sum_y \mathbb{P}(Y^\pi_{DTMC} = y) \frac{1}{\mu_{y,\cdot,\cdot}} \right)^{-1}.$$

From Y^π_{DTMC}, we can calculate the fraction of transitions that move from a generic state y to another generic state y' via a completion. Call this fraction $p_{y \to y',1}$:

$$p_{y \to y',1} = \mathbb{P}(Y^\pi_{DTMC} = y) \frac{\mu_{y,y',1}}{\mu_{y,\cdot,\cdot}}.$$

Summing over all initial states y, we can find the fraction of transitions that are completions which result in the state y':

$$p_{\cdot \to y',1} = \sum_y \mathbb{P}(Y^\pi_{DTMC} = y) \frac{\mu_{y,y',1}}{\mu_{y,\cdot,\cdot}}.$$

Let b^π be the overall fraction of transitions that are completions. Conditioning on the transition into state y' being a completion, we find that the probability that a generic completion results in state y' is

$$\mathbb{P}(Y^\pi_d = y') = \frac{p_{\cdot \to y',1}}{b^\pi}.$$

Combining all of the above equations, we find that

$$\begin{aligned}
\mathbb{P}(Y^\pi_d = y') &= \frac{1}{b^\pi} \sum_y \mathbb{P}(Y^\pi_{DTMC} = y) \frac{\mu_{y,y',1}}{\mu_{y,\cdot,\cdot}} \\
&= \frac{1}{b^\pi} \sum_y \frac{1}{a^\pi} \mathbb{P}(Y^\pi = y) \mu_{y,\cdot,\cdot} \frac{\mu_{y,y',1}}{\mu_{y,\cdot,\cdot}} \\
&= \frac{1}{a^\pi b^\pi} \sum_y \mathbb{P}(Y^\pi = y) \mu_{y,y',1}.
\end{aligned}$$

Recall that a^π is the long-term transition rate, and that b^π is the fraction of transitions that are completions. Thus, $a^\pi b^\pi$ is the long-term completion rate $X^\pi = \lambda^*_\pi$. □

Appendix E. Lemmas about G^{MSJ}

Lemma E.1. *For any $f(q,y)$ which is a real-valued function of the state of the MSJ system,*

$$G^{\mathrm{MSJ}} \circ f(q,y) = \lambda \left(f(q+1,y) - f(q,y)\right) \mathbb{1}_{\{y \in \mathbb{Y}^{\mathrm{Ak}}\}} \tag{E.1}$$

$$+ \mathbb{1}_{q=0, y \notin \mathbb{Y}^{\mathrm{Ak}}} \lambda \sum_{i \in S} p_i (f(0, y \cdot i) - f(0,y)) \tag{E.2}$$

$$+ \mathbb{1}_{q>0} \sum_{\substack{y' \in \mathbb{Y}^{\mathrm{Ak}}, \\ a \in \{0,1\}}} \mu^{\mathrm{Ak}}_{y,y',a} \left(f((q-a)^+, y') - f(q,y)\right) \tag{E.3}$$

$$+ \mathbb{1}_{q=0} \sum_{\substack{y' \in \mathbb{Y}^{\mathrm{MSJ}}, \\ a \in \{0,1\}}} \mu^{\mathrm{MSJ}}_{y,y',a,0} \left(f((q-a)^+, y') - f(q,y)\right). \tag{E.4}$$

Proof. Recall the definition of the generator

$$G^{\mathrm{MSJ}} \circ f(q,y) = \lim_{t \to 0} \frac{1}{t} \mathbb{E}[f(Q^{\mathrm{MSJ}}(t), Y^{\mathrm{MSJ}}(t)) - f(q,y) | Q^{\mathrm{MSJ}}(0) = q, Y^{\mathrm{MSJ}}(0) = y],$$

which can be interpreted as the instantaneous rate of change of the function $f(Q^{\mathrm{MSJ}}(t), Y^{\mathrm{MSJ}}(t))$ when $(Q^{\mathrm{MSJ}}(t), Y^{\mathrm{MSJ}}(t))$ is initialized in (q,y). Note that $(Q^{\mathrm{MSJ}}(t), Y^{\mathrm{MSJ}}(t))$ can change either due to an arrival event, or a transition event of the front state. An arrival event happens with rate λ, and its effect depends on whether $y \in \mathbb{Y}^{\mathrm{Ak}}$: if $y \in \mathbb{Y}^{\mathrm{Ak}}$, there are k jobs in the front, so $Q^{\mathrm{MSJ}}(t)$ changes from q to $q+1$, $Y^{\mathrm{MSJ}}(t)$ remains unchanged; if $y \notin \mathbb{Y}^{\mathrm{Ak}}$, there are strictly fewer than k jobs in the front, so $Q^{\mathrm{MSJ}}(t)$ remains zero after the arrival, and $Y^{\mathrm{MSJ}}(t)$ changes from y to $y \cdot i$ with probability p_i (append a fresh job in state i to the front state with probability p_i). Therefore, arrival events contribute

$$\lambda \left(f(q+1,y) - f(q,y)\right) \mathbb{1}_{\{y \in \mathbb{Y}^{\mathrm{Ak}}\}} + \mathbb{1}_{q=0, y \notin \mathbb{Y}^{\mathrm{Ak}}} \lambda \sum_{i \in S} p_i (f(0, y \cdot i) - f(0,y))$$

to $G^{\mathrm{MSJ}} \circ f(q,y)$, which are the terms in (E.1) and (E.2) in the lemma statement. As for the transition events of the front, a transition from state y to state y' accompanied by a completions causes $(Q^{\mathrm{MSJ}}(t), Y^{\mathrm{MSJ}}(t))$ to change from (q,y) to $((q-a)^+, y')$. Such a transition happens with the rate $\mu^{\mathrm{MSJ}}_{y,y',a,1} = \mu^{\mathrm{Ak}}_{y,y',a}$ if $q > 0$ and $y' \in \mathbb{Y}^{\mathrm{Ak}}$, and happens with rate $\mu^{\mathrm{MSJ}}_{y,y',a,0}$ if $q = 0$. Therefore, the transition events of the front contribute

$$\mathbb{1}_{q>0} \sum_{\substack{y' \in \mathbb{Y}^{\mathrm{Ak}}, \\ a \in \{0,1\}}} \mu^{\mathrm{Ak}}_{y,y',a} \left(f((q-a)^+, y') - f(q,y)\right) + \mathbb{1}_{q=0} \sum_{\substack{y' \in \mathbb{Y}^{\mathrm{MSJ}}, \\ a \in \{0,1\}}} \mu^{\mathrm{MSJ}}_{y,y',a,0} \left(f((q-a)^+, y') - f(q,y)\right)$$

to $G^{\mathrm{MSJ}} \circ f(q,y)$, which are the terms in (E.3) and (E.4) in the lemma statement. □

Lemma 6.1.

$$G^{\mathrm{MSJ}} \circ f^{\mathrm{MSJ}}_{\Delta}(q,y) = \mathbb{1}_{q>0} G^{\mathrm{Ak}} \circ f^{\mathrm{Ak}}_{\Delta}(q,y) + \mathbb{1}_{q=0} O_{\lambda}(1) \tag{E.5}$$

Proof. Let us begin by using Lemmas D.2 and E.1 to give expressions for $G^{\mathrm{MSJ}} \circ f^{\mathrm{MSJ}}_{\Delta}(q,y)$ and $G^{\mathrm{Ak}} \circ f^{\mathrm{Ak}}_{\Delta}(q,y)$.

Note that whenever $q > 0$, $G^{\mathrm{MSJ}} \circ f^{\mathrm{MSJ}}_{\Delta}(q,y)$ is identical to $G^{\mathrm{Ak}} \circ f^{\mathrm{Ak}}_{\Delta}(q,y)$, because the two systems have the same transitions and because $f^{\mathrm{MSJ}}_{\Delta}(q,y)$ and $f^{\mathrm{Ak}}_{\Delta}(q,y)$ are identical.

Note also that whenever $q = 0$, both $G^{\text{MSJ}} f_{\Delta}^{\text{MSJ}}(q, y)$ and $G^{\text{Ak}} \circ f_{\Delta}^{\text{Ak}}(q, y)$ are $O_\lambda(1)$, because $\Delta(y)$ is bounded by a constant for all y, because $\mathbb{Y}^{\text{MSJ}}$ is finite.

As a result,

$$G^{\text{MSJ}} \circ f_{\Delta}^{\text{MSJ}}(q, y) = \mathbb{1}_{q>0} G^{\text{Ak}} \circ f_{\Delta}^{\text{Ak}}(q, y) + \mathbb{1}_{q=0} O_\lambda(1). \quad \square$$

Appendix F. At-least-k busy period

We also prove a lemma about busy periods in the At-least-k system. Define a busy period to begin when the back length q^{Ak} in the At-least-k system transitions from 0 to 1, and to end when the back length next returns to 0. Let B^{Ak} be a random variable representing the length of a busy period in the At-least-k system in stationarity.

Lemma F.1. *In the At-least-k system, for all $\lambda < \lambda^*$*

$$\mathbb{E}[B^{\text{Ak}}] = \Omega_\lambda \left(\frac{1}{1 - \lambda/\lambda^*} \right). \tag{F.1}$$

Proof. In this proof we omit Ak in the subscript of $\mu^{\text{Ak}}_{y,y',a}$ for readability.

To prove Lemma F.1, it suffices to show that $\mathbb{P}(Q^{\text{Ak}} = 0) = O_\lambda(1 - \frac{\lambda}{\lambda^*})$, and that the non-busy periods (periods when $Q^{\text{Ak}} = 0$) have expected duration $\Omega_\lambda(1)$. The latter follows from the fact that all transitions have expected duration $\Omega_\lambda(1)$.

To prove the former, let $u(q^{\text{Ak}}, y^{\text{Ak}})$ be the rate at which new jobs are generated due to completions in a particular state $(q^{\text{Ak}}, y^{\text{Ak}})$ of the At-least-k system. Note that $u(q^{\text{Ak}}, y^{\text{Ak}})$ is positive only if $q^{\text{Ak}} = 0$. The time-average value of $u(q^{\text{Ak}}, y^{\text{Ak}})$ is the difference between the completion rate of the system and the Poisson arrival rate, because in steady state the total completion rate and total arrival rate must match. Thus,

$$\mathbb{E}[u(Q^{\text{Ak}}, Y^{\text{Ak}})] = \lambda^* - \lambda. \tag{F.2}$$

Note that $u(q, y) = \mu_{y,\cdot,1} \mathbb{1}_{\{q=0\}}$, so

$$\mathbb{E}[\mu_{Y^{\text{Ak}},\cdot,1} \mathbb{1}_{\{Q^{\text{Ak}}=0\}}] = \lambda^* - \lambda.$$

Note that

$$\mathbb{P}(Q^{\text{Ak}} = 0) = \frac{\mathbb{E}[\mu_{Y^{\text{Ak}},\cdot,1} \mathbb{1}_{\{Q^{\text{Ak}}=0\}}]}{\mathbb{E}[\mu_{Y^{\text{Ak}},\cdot,1} | Q^{\text{Ak}} = 0]} = \frac{\lambda^* - \lambda}{\mathbb{E}[\mu_{Y^{\text{Ak}},\cdot,1} | Q^{\text{Ak}} = 0]}. \tag{F.3}$$

It therefore suffices to show that there exists a constant $c > 0$ not dependent on λ such that $\mathbb{E}[\mu_{Y^{\text{Ak}},\cdot,1} | Q^{\text{Ak}} = 0] \geq c$.

From an arbitrary state y^{Ak} with $q^{\text{Ak}} = 0$, the distribution of time until a completion next occurs does not depend on λ. Consider the probability of a completion happening in the next second, with no arrivals happening before that completion. This probability is nonzero, and only dependent only λ via the arrival process. The probability can be lower bounded away from zero by substituting a Poisson(λ^*) process instead. We can thus lower bound the completion rate over the next second with an empty back away from 0. This therefore provides a lower bound on the completion rate conditional on the back being empty, $\mathbb{E}[\mu_{Y^{\text{Ak}},\cdot,1} | Q^{\text{Ak}} = 0]$, as desired. Calling that lower bound c, we have:

$$\mathbb{P}(Q^{\text{Ak}} = 0) = \frac{\mathbb{E}[\mu_{Y^{\text{Ak}},\cdot,1} \mathbb{1}_{\{Q^{\text{Ak}}=0\}}]}{\mathbb{E}[\mu_{Y^{\text{Ak}},\cdot,1} | Q^{\text{Ak}} = 0]} \leq \frac{\lambda^* - \lambda}{c} = O_\lambda \left(1 - \frac{\lambda}{\lambda^*} \right). \tag{F.4}$$

This completes the proof. $\square$

Appendix G. Coupling Lemmas

Let us restate the coupling between the At-least-k and MSJ systems. We let the arrivals of the two systems happen at the same time. We couple the transitions of their front states based on their joint state $(q^{\text{MSJ}}, y^{\text{MSJ}}, q^{\text{Ak}}, y^{\text{Ak}})$. If $y^{\text{MSJ}} = y^{\text{Ak}}$, $q^{\text{MSJ}} > 0$, and $q^{\text{Ak}} > 0$, the completions happen at the same time in both systems, the same jobs complete, the same job phase transitions occur, and the jobs entering the front are the same. We call the two systems "merged" during such a time period. Note that under this coupling, if the two systems become merged, they will stay merged until $q^{\text{MSJ}} = 0$ or $q^{\text{Ak}} = 0$. If the systems are not merged, the two systems have independent completions and phase transitions, and independently sampled jobs.

The two systems transition according to synchronized Poisson timers whenever they are merged, and independent Poisson timers otherwise. Because all transitions are exponentially distributed, this poses no obstacle to the coupling.

Lemma 6.3 (Quick merge)**.** *From any joint MSJ, At-least-k state, for any $\epsilon > 0$, under the coupling above, the expected time until $y^{\text{MSJ}} = y^{\text{Ak}}$, $q^{\text{MSJ}} \geq k+1$, and $q^{\text{Ak}} \geq k+1$ is at most $m_1(\epsilon)$ for some $m_1(\epsilon)$ independent of the arrival rate λ and initial joint states, given that $\lambda \in [\epsilon, \lambda^*)$.*

Proof. We call the period of time until $y^{\text{MSJ}} = y^{\text{Ak}}$, $q^{\text{MSJ}} \geq k+1$, and $q^{\text{Ak}} \geq k+1$ the "bad period." We wish to show that the expected length of the bad period is upper bounded by some constant $m_1(\epsilon)$ for all λ such that $\lambda \in [\epsilon, \lambda^*)$.

Consider the possibility that the following sequence of events occurs: over a period of $1/2$ second, at least $2k+1$ jobs arrive. Then, over another $1/2$ second, k completions occur in each of the MSJ and At-least-k systems, which is sufficient to clear out every job initially present in the fronts and replace them with freshly sampled jobs. Finally, the sampled jobs in the fronts of the two systems are the same, in the same order. After this sequence of events, $y^{\text{MSJ}} = y^{\text{Ak}}$, $q^{\text{MSJ}} \geq k+1$, and $q^{\text{Ak}} \geq k+1$, which ends the bad period.

Recall that as long as the front states of the two systems are distinct, their completions are independent. As a result, the probability of this sequence of events is positive, for any $\lambda > 0$ and for any initial states $y^{\text{MSJ}}, y^{\text{Ak}}$. We call the probability of this sequence of events $\text{pGood}(\lambda, y^{\text{MSJ}}, y^{\text{Ak}})$.

Moreover, $\text{pGood}(\lambda, y^{\text{MSJ}}, y^{\text{Ak}})$ is monotonically increasing in λ, as λ only affects the probability that at least $2k+1$ jobs arrive in the first half second.

Therefore, the least value of $\text{pGood}(\lambda, y^{\text{MSJ}}, y^{\text{Ak}})$ is achieved when $\lambda = \epsilon$. Because there are only finitely many possible front states $y^{\text{MSJ}} \in \mathbb{Y}^{\text{MSJ}}, y^{\text{Ak}} \in \mathbb{Y}^{\text{Ak}}$, there must be some lowest value of $\text{pGood}(\epsilon, y^{\text{MSJ}}, y^{\text{Ak}})$. We call this value $\text{pGood}^*(\epsilon)$. Note that for all $\lambda \geq \epsilon$ and for all $y^{\text{MSJ}} \in \mathbb{Y}^{\text{MSJ}}, y^{\text{Ak}} \in \mathbb{Y}^{\text{Ak}}$,

$$\text{pGood}(\lambda, y^{\text{MSJ}}, y^{\text{Ak}}) \geq \text{pGood}^*(\epsilon) > 0. \tag{G.1}$$

In the first second, there is at least a $\text{pGood}^*(\epsilon)$ chance of the desired sequence of events happening and the bad period completing. In the next second, the same is true. In general, the time until the bad period completes is upper bounded by a geometric distribution with completion probability $\text{pGood}^*(\epsilon)$. Taking $m_1(\epsilon) = 1/\text{pGood}^*(\epsilon)$, the mean time until the bad period completes is upper bounded by $m_1(\epsilon)$, which is independent of the arrival rate λ and initial joint states, as desired. □

Lemma 6.4 (Long merged period)**.** *From any joint MSJ, At-least-k state such that $y^{\text{MSJ}} = y^{\text{Ak}}$, $q^{\text{MSJ}} \geq k+1$, and $q^{\text{Ak}} \geq k+1$, the expected time until $q^{\text{MSJ}} = 0$, $q^{\text{Ak}} = 0$, or $y^{\text{MSJ}} \neq y^{\text{Ak}}$, is at least $\frac{m_2}{1-\lambda/\lambda^*}$ for some m_2 independent of the arrival rate λ and initial joint states, given that $\lambda < \lambda^*$.*

Note that the time until $q^{\text{MSJ}} = 0$ or $q^{\text{Ak}} = 0$ is a lower bound on the time until $y^{\text{MSJ}} \neq y^{\text{Ak}}$.

Proof. In this proof we omit Ak in the subscript of $\mu^{\text{Ak}}_{y,y',a}$ for readability.

Let's call the period of interest the "good period." Note that throughout the good period, $y^{\text{MSJ}} = y^{\text{Ak}}$. Let us introduce a new lower-bounding MSJ system, M', beginning in a general state $y^{M'} = y^{\text{MSJ}} = y^{\text{Ak}}$ and beginning with $q^{M'} = k+1$. Let us define a coupling between M' and the original MSJ and At-least-k systems in the same synchronized/independent fashion defined at the start of Section 6.1. As a result, for

all time until $q^{M'} = 0$, $y^{M'} = y^{\text{MSJ}} = y^{\text{Ak}}$, and $q^{M'} \leq q^{\text{MSJ}}$, and $q^{M'} \leq q^{\text{Ak}}$. In particular, the duration until $q^{M'} = 0$ is a lower bound on the length of the good period.

Let us set up a new coupled system, M''. The M'' system is an At-least-k system initialized in a specific front state distribution to be specified later and with $q^{M''} = 1$. Let us define a coupling between the M' and M'' systems in the same synchronized/independent fashion defined at the start of Section 6.1. Note however that M'' is a new system, distinct from all of the previous systems.

Let $B^{M'}$ be the length of the first busy period of the M' system, which is the time in M' until $q^{M'} = 0$; similarly, let $B^{M''}$ be the length of the first busy period of the M'' system. We want to show that

$$\mathbb{E}[B^{M'}] \geq m_3 \mathbb{E}[B^{M''}], \tag{G.2}$$

$$\mathbb{E}[B^{M''}] \geq \frac{m_4}{1 - \lambda/\lambda^*}. \tag{G.3}$$

for some positive numbers m_3 and m_4 independent of the arrival rate λ and the initial front state of the M' system $y^{M'}$.

We will choose the front state distribution of the M'' system in order to guarantee that (G.3) holds. To do so, we will make use of Lemma F.1, which states that the At-least-k system has long busy periods:

$$\mathbb{E}[B^{\text{Ak}}] = \Omega_\lambda \left(\frac{1}{1 - \lambda/\lambda^*} \right). \tag{G.4}$$

Let $Y^{\text{Ak}-BP}$ denote the long-term-average distribution of the front state in the At-least-k system at the start of a busy period. We let the initial state distribution of the M'' system be $y^{M''} \sim Y^{\text{Ak}-BP}$ and $q^{M''} = 1$. As a result, $\mathbb{E}[B^{M''}] = \mathbb{E}[B^{\text{Ak}}]$, the expected busy period length of the At-least-k system. By Lemma F.1, we have $\mathbb{E}[B^{M''}] \geq \frac{m_4}{1-\lambda/\lambda^*}$ for some positive number m_4 independent of λ and $y^{M'}$.

Now, we wish to show (G.2): that the length of the first busy period in M', initialized in an arbitrary initial front state $y^{M'}$ and $q^{M'} = k + 1$, is also long in expectation.

To prove this, let us introduce a very fast Poisson process with a rate μ^* given by

$$\mu^* = \lambda^* + \max_{y \in \mathbb{Y}^{\text{Ak}}} \sum_{y',a} \mu_{y,y',a}.$$

Note that μ^* is at least as fast as the transition rate of M'' in any state, and μ^* is independent of λ. Let us define a coupling between the Poisson(μ^*) process and the M'' system. Transitions in the M'' system only occur when the Poisson(μ^*) increment occurs, where with some probability sampled on each Poisson increment a transition happens, and otherwise no transition occurs. In state y, a transition happens with probability

$$\frac{\lambda + \sum_{y',a} \mu_{y,y',a}}{\mu^*}.$$

Note that this probability is always less than 1, by the definition of μ^*.

To lower bound $\mathbb{E}[B^{M'}]$, the expected busy period length in the M' system, let us consider $\mathbb{E}[B^{M'} \mathbb{1}_{\{A_1 \wedge A_2\}}]$, where A_1 and A_2 are the following two events:

1. Event A_1: the first increment of the Poisson(μ^*) process takes at least 1 second.
2. Event A_2: during the first second M' has exactly k completions, after each of which the job entering the front of the M' system is sampled to have the same server need as the corresponding job of the M'' system, and then all the jobs transition to the same phase as in the M'' system. At the end of the first second, M' and M'' have identical front states y and back lengths $q = 1$ after exactly k completions in the M' system.

First, note that

$$\mathbb{E}[B^{M'}] \geq \mathbb{E}[B^{M'} \mathbb{1}_{\{A_1 \wedge A_2\}}] = \mathbb{E}[B^{M'} | A_1 \wedge A_2] \mathbb{P}(A_1 \wedge A_2). \tag{G.5}$$

Note that $\mathbb{P}(A_1 \wedge A_2)$ is lower bounded by a positive constant for every λ such that $\epsilon \leq \lambda \leq \lambda^*$, so we can focus on $\mathbb{E}[B^{M'}|A_1 \wedge A_2]$.

Note that if events A_1 and A_2 occur, the M' and M'' systems have the same busy period length, because after 1 second, the two systems have identical states. Specifically, both systems become empty at the same time, which is the first time after each is initialized when each becomes empty.

As a result,

$$\mathbb{E}[B^{M'}|A_1 \wedge A_2] = \mathbb{E}[B^{M''}|A_1 \wedge A_2]. \tag{G.6}$$

Note that Event A_2 is conditionally independent of the behavior of the M'' system, given that Event A_1 occurs. As a result,

$$\mathbb{E}[B^{M''}|A_1 \wedge A_2] = \mathbb{E}[B^{M''}|A_1]. \tag{G.7}$$

Notice that event A_1 is independent of the state of M''. Conditioning on the event A_1 merely increases the time of the first transition in M'', without altering the future updates of M'' at all. As a result,

$$\mathbb{E}[B^{M''}|A_1] \geq \mathbb{E}[B^{M''}]. \tag{G.8}$$

Thus, $\mathbb{E}[B^{M'}]$ is lower bounded by a constant multiple of $\mathbb{E}[B^{M''}]$. Recall that by construction, $\mathbb{E}[B^{M''}] = \Omega_\lambda(\frac{1}{1-\lambda/\lambda^*})$. Combining (G.2) and (G.3) and letting $m_2 = m_3 m_4$, we get the desired lower bound on the expected length of the good period. □

Appendix H. Extensions of the Multiserver-job model

Appendix H.1. Nontrival scheduling policies: Backfilling

A common family of MSJ scheduling policies in practice are *backfilling* policies [4, 45, 48]. Under a backfilling policy, the scheduler begins by placing jobs into service in arrival order, as in the FCFS policy. However, once a job is encountered which does not fit in the available servers, additional jobs are considered for service. By doing so, the stability region and mean response time are improved relative to FCFS, though it is unclear whether the full stability region can be achieved [15]. Some backfilling policies give rise to finite skip models, and can thus be handled by the RESET technique.

As an example, consider the "First Fit-k" policy: The scheduler iterates through the first k jobs in arrival order, checking for each job whether it can be served in the available servers. Each job that fits is served. This policy only serves jobs among the k oldest in arrival order, so it can be handled by the RESET technique.

Beyond backfilling policies, more advanced packing policies can also be considered. For instance, for small k the scheduler could simply search over all subsets of the k oldest jobs and serve the subset with maximal total server need $\leq k$. This policy is also finite skip, and the RESET technique also applies.

Appendix H.2. Changing server need during service

The standard MSJ model assumes that jobs require a fixed service need throughout their time in service. However, in some settings, jobs may require a varying number of servers. For example, consider a fork-join model with simultaneous start. Suppose that each job is made of some number of tasks, each with independent duration, and each requiring 1 server. As the tasks complete, the server need of the job as a whole diminishes, freeing up space for other jobs to run. This setting still gives rise to a finite-skip model, and poses no difficulty to our RESET technique.

Another natural setting in which server needs change over time is the directed acyclic graph (DAG) setting, in which jobs are broken up into small segments of work, with potentially complex dependencies between segments. The DAG scheduling literature often focuses on scheduling the segments of an individual DAG job. It is natural to consider a scheduling setting where many DAG jobs arrive over time. Holding the DAG scheduling policy constant, this model effectively gives rise to a MSJ model where server needs can vary over time, and potentially vary dynamically in response to the service conditions. As long as the high-level scheduling policy deciding which DAG jobs to run is finite-skip, the model as a whole is finite-skip, and our RESET technique can characterize its asymptotic mean response time.

Appendix H.3. Multidimensional resource constraints

The standard MSJ model considers a single constrained resource. However, computing jobs are often constrained by a variety of resources, such as CPU, GPU, other accelerators, memory bandwidth, cache capacity, network bandwidth, etc. Such multidimensional resource constraints are often considered in the VM scheduling literature. In that literature, only stability results are known. Our RESET technique thus gives the first characterization of asymptotic mean response time in that setting.

Appendix H.4. Heterogeneous servers

In the standard MSJ model, all servers are identical. However, it is also important to consider settings where different kinds of servers are available, which can provide different amounts of resources. One can also consider jobs that need to be served at a particular server or set of servers, such as a job that processes data stored at that server. In a multidimensional resource setting, some servers may also provide different resources, such as a GPU-heavy or CPU-heavy server. All of these extensions are compatible with the RESET technique.

Appendix H.5. Turning off idle servers

To improve energy efficiency, it may be preferable to turn off idle servers. Idle servers consume nearly as much energy as active servers. However, turned-off servers take some time to restart. It is important to characterize the impact of this start-up delay on mean response time to understand the tradeoff inherent in turning off idle servers. The process of turning off and on servers can be incorporated into a finite-skip model, because there are a finite number of possible states that the servers can be in. As a result, our RESET technique can provide a characterization of mean response time.

Appendix H.6. Preemption overheads

The FCFS policy never preempts any jobs. Prior work has studied settings with unlimited preemption. However, practical settings often allow only a limited subset of jobs to be preempted, and jobs may incur an overhead when preemption occurs. This overhead corresponds to the time necessary to snapshot a job in service, and for the new job to be transferred onto the freed servers. Models with preemption overheads have only recently begun to be analyzed in the M/G/1 setting [37], with no mean response time analysis known in the one-server-per-job multiserver model, much less the multiserver-job model. Preemption overheads can be modeled with a finite-number of additional states, marking the corresponding servers as undergoing preemption. As a result, our RESET technique can provide a characterization of mean response time.

Appendix I. Empirical correlation between $\Delta(Y_d^{\text{Sat}})$ and convergence rate

As discussed in Section 8, we have empirically noticed a correlation between large $\Delta(Y_d^{\text{Sat}})$ values and slower convergence rates of our predicted value of $E[T]$ to the exact value, as $\lambda \to \lambda^*$. Our predicted value of mean response time is:

$$E[T^{pred}] = \frac{1}{\lambda^*} \frac{\Delta(Y_d^{\text{Sat}}) + 1}{1 - \lambda/\lambda^*}.$$

We prove in Theorem 4.2 that $E[T^{pred}] - E[T] = O_\lambda(1)$, ensuring that the two values reliably converge in the $\lambda \to \lambda^*$ limit in all settings, as illustrated in Section 8.

In this section, we further investigate this correlation by comparing $\Delta(Y_d^{\text{Sat}})$ with the relative error $\frac{E[T^{pred}]-E[T]}{E[T]}$ for a pair of parameterized sequences of workload settings. The setting we investigate has $k = 5$ servers, with jobs having server need either 1 or 5. We set the arrival rate $\lambda = 0.8\lambda^*$, using 80% of the stability region.

We separately parameterize the fraction of 1-server jobs present, as well as the duration of 1-server jobs. First, we vary p_1, the fraction of 1-server jobs from 1% to 99%, in 1% increments, while setting 1-server jobs to have duration $Exp(1/5)$. Second, we set the duration of 1-server jobs to be $Exp(\mu_1)$, with μ_1 ranging

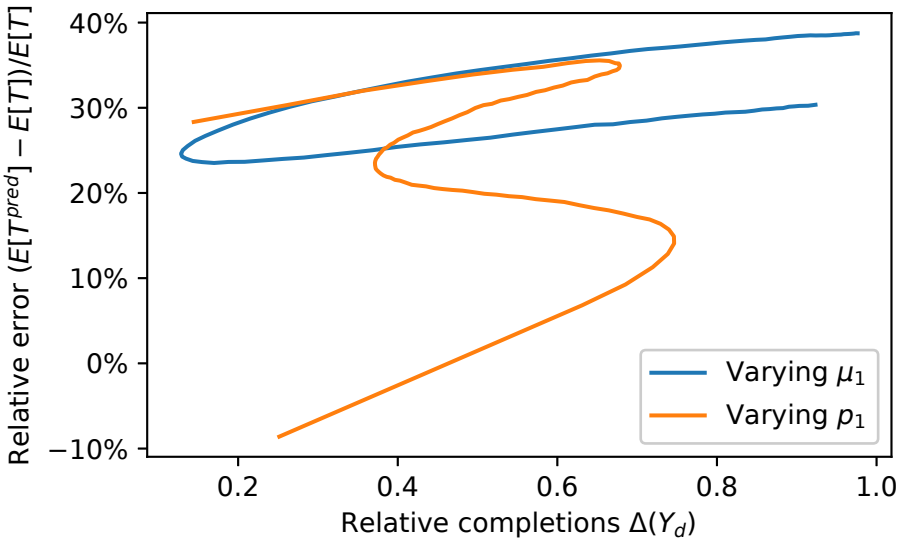


Figure I.4: Empirical relationship between $\Delta(Y_d^{\text{Sat}})$ and the relative error between our prediction of mean response time and the true value, in two parameterized workload settings, both with $k = 5$ servers and server needs either 1 or 5. We alternately parameterize μ_1, the completion rate of 1-server jobs, and p_1, the fraction of 1-server jobs. Load $\lambda/\lambda^* = 0.8$. Simulated 10^8 arrivals.

from 0.01 to 100 in 100 evenly multiplicatively-spaced increments, while setting 50% of jobs to have each server need. In both cases, we set the 5-server jobs to have duration $Exp(1)$.

We plot the behavior of these two settings in Fig. I.4, comparing the $\Delta(Y_d^{\text{Sat}})$, the relative completion in the departure-average state of the saturated system, against the relative error $\frac{E[T^{pred}]-E[T]}{E[T]}$. The empirical results show a significant correlation between $\Delta(Y_d^{\text{Sat}})$ and the relative error $\frac{E[T^{pred}]-E[T]}{E[T]}$ in the case of parameterized μ_1 ($R^2 = 0.526$), but no significant correlation in the case of parameterized p_1 ($R^2 = 0.005$). This indicates that the correlation observed in Section 8 may exist in some settings, but is not robust or reliable. Further investigation will be needed to better understand this correlation.