

Delay Asymptotics and Bounds for Multi-Task Parallel Jobs

ABSTRACT

We study delay of jobs that consist of multiple parallel tasks, which is a critical performance metric in a wide range of applications such as data file retrieval in coded storage systems and parallel computing. In this problem, each *job* is completed only when *all* of its tasks are completed, so the delay of a job is the maximum of the delays of its tasks. Despite the wide attention this problem has received, tight analysis is still largely unknown since analyzing job delay requires characterizing the complicated correlation among task delays, which is hard to do.

We first consider an asymptotic regime where the number of servers, n , goes to infinity, and the number of tasks in a job, $k^{(n)}$, is allowed to increase with n . We establish the asymptotic independence of any $k^{(n)}$ queues under the condition $k^{(n)} = o(n^{1/4})$. This greatly generalizes the asymptotic-independence type of results in the literature where asymptotic independence is shown only for a fixed constant number of queues. As a consequence of our independence result, the job delay converges to the maximum of independent task delays.

We next consider the non-asymptotic regime. Here we prove that independence yields a stochastic upper bound on job delay for any n and any $k^{(n)}$ with $k^{(n)} \leq n$. The key component of our proof is a new technique we develop, called “Poisson oversampling”. Our approach converts the job delay problem into a corresponding balls-and-bins problem. However, in contrast with typical balls-and-bins problems where there is a negative correlation among bins, we prove that our variant exhibits positive correlation.

ACM Reference Format:

. 2018. Delay Asymptotics and Bounds for Multi-Task Parallel Jobs. In *Proceedings of IFIP WG PERFORMANCE*. ACM, New York, NY, USA, 14 pages. <https://doi.org/10.1145/nnnnnnn.nnnnnnn>

1 INTRODUCTION

The problem

We consider a system with n servers, each with its own queue. Jobs arrive over time according to a Poisson process, and each job consists of some number of tasks, k , where $k \leq n$. Upon arrival, each job chooses k distinct servers uniformly at random and sends one task to each server. Each server serves the tasks in its queue in a First-In, First-Out (FIFO) manner. A job is considered to be completed only when *all* of its tasks are completed. Our goal is to compute the *distribution of job delay*, namely the time from when a job arrives until the whole job completes. If a job’s tasks experienced independent delays, then computing the distribution of job delay

would be easy: take the maximum of the independent task delays. Unfortunately, the task delays are not independent in general.

Our model is a generalization on the classic *fork-join* model, which is identical to our model except that it assumes that $k = n$: every job is forked to all n servers. In contrast, in our model, the fork is *limited* to k servers with $k \leq n$. So we will refer to our model as the *limited fork-join* model. Obtaining tight analytical job delay characterizations for fork-join systems is known to be notoriously difficult, and exact analysis of fork-join remains an open problem except for the two-server case [1, 11]. Many papers have derived bounds and approximations for fork-join systems [2, 3, 22, 23, 28, 30, 35–37, 44, 45, 51]. However, to our knowledge, there are no tight closed-form characterizations.

Motivation

Delay of jobs, rather than delay of individual tasks, is a more critical performance metric in systems with parallelism, yet a fundamental understanding of job delay is still lacking. One example application is data file retrieval in coded storage systems [20, 21, 24, 25, 39, 40]. Here a job is the retrieval of a data file, which is stored as multiple data chunks. The data chunks are in a coded form such that any k -sized subset of them is enough to reconstruct the file. Coded file retrieval can be modeled via the so-called (n, r, k) model [39] where a job can request r data chunks with $r \geq k$ and the job is completed as long as k of them are completed. Existing analysis of the (n, r, k) model is usually not tight except for the light load regime [20, 21, 25]. The special case where $r = d$ and $k = 1$, called the Redundancy- d model, is also highly non-trivial and was solved just last year [15]. Job delay in general (n, r, k) models remains wide open. Within the coded file retrieval setting, our limited fork-join model can be viewed as the (n, k, k) problem.

Another application is parallel computing systems such as the “map” phase of the popular MapReduce framework [8], where a job is divided into tasks that can run in parallel. A few papers have been written to analytically approximate the delay of MapReduce jobs [10, 43, 46]. However, [43] abstracts the map phase as a single-server system, which does not fully capture the parallelism, and [10, 46] provide coarse approximations by making various assumptions on the queueing processes, including assuming (without formal justification) that a job’s tasks have independent delays.

In the above applications, load-balancing policies [24, 25, 29, 40, 41, 49, 52, 55] are usually used for assigning tasks to servers. For scenarios where either low-overhead is desired or information accessibility is constrained (such as in a distributed setting), workload agnostic assignment policies [24, 40, 52] can be preferred. Our limited fork-join model assumes a random task assignment policy, which is suitable for such application scenarios.

Our approach and what makes this problem hard

The root of the hardness of analyzing job delay in our model is the complicated correlation among queues, which leads to the correlation among the delays of a job’s tasks. If the task delays were

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

IFIP WG PERFORMANCE, December 2018, Toulouse, France

© 2018 Copyright held by the owner/author(s).

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM.

<https://doi.org/10.1145/nnnnnnn.nnnnnnn>

independent, then the probability distribution of job delay would have a simple form. In this paper, we are interested in developing conditions and quantifying in what sense the job delay can be approximated by the job delay under the independence assumption.

Asymptotic Regime. We first study a regime where we prove that a job’s tasks can be *viewed* as being independent: We focus on the asymptotic regime where the number of servers, n , goes to infinity. Here we are specifically interested in developing conditions under which the delays of a job’s tasks are *asymptotically independent*, i.e., their joint distribution converges to the product distribution of their marginals.

Asymptotic independence of a number of queues in large systems is often called “chaoticity” and studied under the name “propagation of chaos.” In many papers [13, 14, 47, 48, 53], asymptotic independence is simply assumed to simplify analysis. In some load-balancing settings, asymptotic independence has been proven [4, 16, 55]. One strong restriction of the asymptotic independence proofs is that it can only prove that a *constant* number of queues are asymptotically independent. In contrast, our goal is to establish asymptotic independence for any k queues where k may grow with n ; we write k as $k^{(n)}$ to explicitly indicate its dependence on n . The asymptotic independence of any $k^{(n)}$ queues implies the asymptotic independence of the delays of a job’s tasks since they are sent to $k^{(n)}$ queues. Allowing $k^{(n)}$ to grow with n captures the trends that data files get larger and that jobs are processing larger and larger data sets [5].

When proving asymptotic independence of a constant number of queues in steady state, it is typical to start by showing asymptotic independence over a *constant time interval* $[0, t]$, where t is long enough for these queues to be close to steady state. Unfortunately, since $k^{(n)}$ grows with n in our model, to reach steady state, the system needs a time interval $[0, \tau^{(n)}]$, growing with n . This further complicates the analysis since asymptotic independence then needs to be established over this longer, *non-constant*, time interval.

Non-asymptotic regime. Next, we study the non-asymptotic regime. We show that for *any* n and *any* $k^{(n)} = k$ with $k^{(n)} \leq n$, the distribution of job delay is *stochastically upper bounded* by the distribution given by independent task delays, which we call the *independence upper bound*. Therefore, independence not only characterizes the *limiting* behavior of job delay in the asymptotic regime where $n \rightarrow \infty$, but also yields an upper bound for *any* n . I.e., the independence upper bound is asymptotically *tight*. An illustration of the tightness is provided in Figure 2 generated from simulations. The independence upper bound is also tighter than all the existing upper bounds in prior work [24, 37].

We prove the independence upper bound using the theory of associated random variables [9]. Association (also called positive association) is a form of positive correlation, and it has the property that if a set of random variables are associated, then the maximum of them is stochastically upper bounded by the maximum of independent versions of them. To show the independence upper bound, it thus suffices to show that the delays of a job’s tasks are associated. Such an association result is known for the classical fork-join model with $k^{(n)} = n$, but not for the limited fork-join model when $k^{(n)} < n$. When proving association, a commonly used idea is to observe the system at each job arrival time, and show

that the numbers of tasks sent to different queues are associated [23, 35, 40]. This corresponds to a balls-and-bins problem where $k^{(n)}$ balls are thrown into n bins in the same way that the tasks are sent to the queues. What is needed is that the numbers of balls thrown in different bins are associated, which is obviously true for $k^{(n)} = n$ since they are all equal to one, but not true when $k^{(n)} < n$. In fact, they are actually *negatively associated* by a classical result [19]. However, this does not mean that the *steady-state* queues are negatively associated, leaving the association problem for $k^{(n)} < n$ unsolved in the literature. As pointed out in [24], it was not known if independence yielded a bound, either lower or upper.

We develop a novel technique that we call “Poisson oversampling,” where we *observe* the system not only when jobs arrive but also at the jump times of a Poisson process that is independent of everything else. This oversampling does not change the dynamics in the system since it is only a way to observe the system state. But now at each observation time, there could be one or zero job arrivals. So in the corresponding balls-and-bins problem, there is certain probability that there are no balls at all. By properly choosing the observation rate, this extra randomness surprisingly makes the numbers of balls thrown in any $k^{(n)}$ bins (positively) associated, and further implies that the steady-state queues are associated. With this technique, we are able to prove the independence upper bound for any $k^{(n)} \leq n$ for the first time.

Results

Our goal is to characterize the tail probability of the job delay distribution in steady state, since it is commonly used to quantify the quality of service. We study a system with n servers in which each job consists of $k^{(n)}$ tasks.

Our first result is that under the condition $k^{(n)} = o(n^{1/4})$, the queues at any $k^{(n)}$ servers are *asymptotically independent* in steady state as $n \rightarrow \infty$, and thus the delays of a job’s tasks are also asymptotically independent. It then follows that the job delay converges to the job delay given by the independence assumption. This result is established in Theorem 4.1 for generally distributed service times, and some explicit forms are given in Corollary 4.1 for exponentially distributed service times. One might wonder where the order of $o(n^{1/4})$ comes from or whether it can be increased; we discuss this in Section 5.5. This is the first asymptotically tight characterization of job delay in the limited fork-join model.

Our next result is that for *any* n and *any* $k^{(n)}$ with $k^{(n)} \leq n$, the job delay is stochastically upper bounded by the job delay given by the independence assumption. We refer to this upper bound as the *independence upper bound*. It is a new upper bound on job delay that is tighter than existing upper bounds. The technique we develop for the proof, named “Poisson oversampling”, may be of independent interest for other related problems.

Organization of the paper

The rest of this paper is organized as follows. Section 2 discusses the related work. We introduce our model and notation in Section 3. We summarize our main results in Section 4. In Section 5 we give proofs of the asymptotic independence results and the convergence of job delay. In Section 6 we prove the independence upper bound.

In Section 7 we provide simulation evaluation of our analysis. We conclude our paper in Section 8.

2 RELATED WORK

In this section we discuss prior work on the limited fork-join model and some other related models. Prior work on the limited fork-join model [24, 37] has focused on the non-asymptotic regime and derived bounds on job delay. However, the bounds in [24, 37] do not have tightness guarantees. In particular, the upper bounds there are generally looser than the independence upper bound. Furthermore, none of the prior work has studied the asymptotic regime of the limited fork-join model. Below we give detailed discussions.

Limited fork-join model. Rizk et al. [37] give upper bounds on the tail probabilities of job delay in various settings. For Poisson arrivals and exponentially distributed service times, their upper bound is looser than the independence upper bound. For general service time distributions, their upper bound needs to be computed by numerically solving a non-linear equation. In contrast, we show that the independence upper bound holds and we also further establish asymptotic tightness of the independence upper bound.

Lee et al. [24] give upper and lower bounds on the *mean* job delay, not on the tail probabilities, assuming that service times follow an *exponential* distribution. Their upper bound is in general looser than the expectation of the independence upper bound, although the difference disappears as $n \rightarrow \infty$ when $k^{(n)} = o(n)$. Compared to this, we prove that the independence upper bound is indeed an upper bound for *any* n and $k^{(n)}$ with $k^{(n)} \leq n$. Besides, we prove it for very general service time distributions and in a stochastic dominance sense, which is stronger than the expectation sense. Also, there is a gap between their upper and lower bounds and there is no tightness analysis. Again, we establish asymptotic tightness of the independence upper bound.

There has also been work on variants of the limited fork-join model where each job consists of a random number of tasks. For example, Shah et al. [40] simply *assume* that the number of tasks in each job has a distribution such that the numbers of tasks sent to different queues are associated, thus obtaining the independence upper bound for their model. They further investigate different policies for assigning the tasks of a newly arrived job to servers, and show that the job delay under the two studied policies is shorter (in a proper sense) than the job delay under the random assignment in the limited fork-join model. Nelson et al. [36] consider a model where tasks wait in a central queue until some server becomes available. They show that the mean job delay is given by a set of recurrence equations, but no analytical form is derived. Kumar and Shorey [23] obtain upper and lower bounds on the mean delay when tasks are assigned to servers independently. But still, there are gaps between the upper and lower bounds.

Classic fork-join model. The classic fork-join model, where the number of tasks in a job is *equal* to the number of servers, n , has been widely studied in the literature. Similar to the limited fork-join model, tight characterizations of job delay are generally unknown except when $n = 2$. See [44] for a detailed survey.

For general n , it has been proven that the mean delay of a job scales as $\Theta(\ln(n))$ as $n \rightarrow \infty$ under proper assumptions [2, 3, 35]. But a tight characterization of the constant in $\Theta(\ln(n))$ is not known.

n	number of servers
superscript (n)	quantities in the n -server system
$k^{(n)}$	number of tasks in a job
$\Lambda^{(n)}$	job arrival rate
λ	task arrival rate to each queue
$1/\mu$	mean of service time
ρ	load at each queue
$W_i^{(n)}(t)$	workload of server i 's queue at time t
$T^{(n)}$	job delay
$\hat{T}^{(n)}$	job delay given by independent task delays
H_m	m -th harmonic number: $H_m = \sum_{j=1}^m \frac{1}{j}$

Table 1: Notation Table

Besides providing bounds for the limited fork-join model, Rizk et al. [37] also derive an upper bound on the tail distribution of the job delay for the classic fork-join model. Again, the tightness of the bound is not addressed. There has also been work on efficient computation of performance bounds (see, e.g., Lui et al. [30]), *closed* fork-join systems (see, e.g., Varki [45]), network settings (see, e.g., Xia et al. [51]), special traffic regimes (see, e.g., Ko and Serfozo [22], Lu and Pang [28]), etc.

MapReduce. Modeling MapReduce systems is challenging since the systems have many complex characteristics such as parallel servers, data locality, communication networks, etc. Most theoretical work on MapReduce does not provide analytical form bounds on the job delay. The papers by Lin et al. [27], Moseley et al. [34], Zheng et al. [56] and Sun et al. [42] design scheduling algorithms such that the job delay is guaranteed to be within a constant factor of the optimal, but do not provide analytical bounds. Tan et al. [43] quantifies the distribution tail of job delay when the map phase is abstracted as a single-server queue. This abstraction pools all the computing resource into a super server, resulting in a system with much higher efficiency, especially when the number of tasks in a job is large. Vianna et al. [46] and Farhat et al. [10] derive approximations on the job delay by assuming that the tasks of a job experience independent delays. But no proof is provided to justify the validity of the approximations.

Asymptotic task delay. One component of the job delay in MapReduce is the task delay. Wang et al. [50] and Xie and Lu [54] bound the mean *task delay*, taking into consideration data locality; however they do not deal with the *job delay*. Bounding *job delay* would require characterizations of the correlation among queues. Ying et al. [55] study the task delay in a model where a load-balancing policy called batch-filling is used. They establish asymptotic independence for a *constant* number of queues, which is insufficient for models with jobs with a growing number of tasks.

3 MODEL AND NOTATION

Basic Notation. The symbols \mathbb{R}_+ and \mathbb{Z}_+ denote the set of non-negative real numbers and nonnegative integers, respectively. We denote random variables by capital letters and vectors by bold letters. When a Markov chain $(X(t), t \geq 0)$ has a unique stationary distribution, we denote by $X(\infty)$ a random element whose distribution is the stationary distribution.

We denote by \Rightarrow convergence in distribution (weak convergence) for random elements. We denote by $d_{TV}(\pi_1, \pi_2)$ the total variation distance between two probability measures π_1 and π_2 on a sigma-algebra σ of some sample space, i.e.,

$$d_{TV}(\pi_1, \pi_2) = \sup_{S \in \sigma} |\pi_1(S) - \pi_2(S)|. \quad (1)$$

Limited fork-join model. Our notation is summarized in Table 1. Recall that we consider a system with n servers, each with its own FIFO queue. We append the superscript (n) to related quantities to indicate that they are for the n -server system. We say that a quantity is a *constant* if it does not scale with n .

Jobs and tasks. Jobs arrive over time according to a Poisson process with rate $\Lambda^{(n)}$, and each job consists of $k^{(n)}$ tasks with $k^{(n)} \leq n$. Upon arrival, each job picks $k^{(n)}$ *distinct* servers uniformly at random from the n servers and sends one task to each server. We assume that $\Lambda^{(n)} = n\lambda/k^{(n)}$ for a *constant* λ , where the constant λ is the task arrival rate to each individual queue. Since different jobs choose servers independently, the task arrival process to each queue is also a Poisson process, and the rate is λ . The service times of tasks are i.i.d. following a cdf G with expectation $1/\mu$ and a finite second moment. We think of the service time of each task as being generated upon arrival: each task brings a required service time with it, but the length of the required service time is revealed to the system only when the task is completed. The load of each queue, $\rho = \lambda/\mu$, is then a constant and we assume that $\rho < 1$.

Queueing dynamics. It is not hard to see that each queue is an M/G/1 queue. But the queues are not independent in general since $k^{(n)}$ tasks arrive to the system at the same time. Let $W_i^{(n)}(t)$ denote the *workload* of server i 's queue at time t , i.e., the total remaining service time of all the tasks in the queue, including the partially served task in service. So the workload of a queue is the waiting time of an incoming task to the queue before the server starts serving it. Let $\mathbf{W}^{(n)}(t) = (W_1^{(n)}(t), W_2^{(n)}(t), \dots, W_n^{(n)}(t))$. Then the workload process, $(\mathbf{W}^{(n)}(t), t \geq 0)$, is Markovian and ergodic. The ergodicity can be proven using the rather standard Foster-Lyapunov criteria [33], so we omit it here. Therefore, the workload process has a unique stationary distribution and $\mathbf{W}^{(n)}(t) \Rightarrow \mathbf{W}^{(n)}(\infty)$ as $t \rightarrow \infty$.

Job delay. We are interested in the distribution of job delay in steady state, i.e., the delay a job would experience if it arrives to the system and finds the system in steady state. Let a random variable $T^{(n)}$ represent this steady-state job delay. Specifically, the distribution of $T^{(n)}$ is determined by the workload $\mathbf{W}^{(n)}(\infty)$ in the following way. When a job comes into the system, its tasks are sent to $k^{(n)}$ queues and experience the delays in these queues. Since the queueing processes are symmetric over the indices of queues, without loss of generality, we can assume that the tasks are sent to the first $k^{(n)}$ queues for the purpose of computing the distribution of $T^{(n)}$. The delay of a task is the sum of its waiting time and service

time. So the task delay in queue i , denoted by $T_i^{(n)}$, can be written as $T_i^{(n)} = W_i^{(n)}(\infty) + X_i$ with X_i being the service time. Recall that the X_i 's are i.i.d. $\sim G$ and independent of everything else. Since the job is completed only when all its tasks are completed,

$$T^{(n)} = \max \{T_1^{(n)}, T_2^{(n)}, \dots, T_{k^{(n)}}^{(n)}\}. \quad (2)$$

We will study the relation between $T^{(n)}$ and $\hat{T}^{(n)}$ with $\hat{T}^{(n)}$ defined as the job delay given by *independent* task delays. Specifically, $\hat{T}^{(n)}$ can be expressed as:

$$\hat{T}^{(n)} = \max \{\hat{T}_1^{(n)}, \hat{T}_2^{(n)}, \dots, \hat{T}_{k^{(n)}}^{(n)}\}, \quad (3)$$

where $\hat{T}_1^{(n)}, \hat{T}_2^{(n)}, \dots, \hat{T}_{k^{(n)}}^{(n)}$ are i.i.d. and each $\hat{T}_i^{(n)}$ has the same distribution as $T_i^{(n)}$. Again, due to symmetry, all the $T_i^{(n)}$'s have the same distribution. Let F denote the cdf of $T_i^{(n)}$, whose form is known from the queueing theory literature. Then, we have the following explicit form for $\hat{T}^{(n)}$:

$$\mathbb{P}(\hat{T}^{(n)} \leq \tau) = (F(\tau))^{k^{(n)}}, \quad \tau \geq 0. \quad (4)$$

4 MAIN RESULTS

In Theorem 4.1, we establish asymptotic independence of any $k^{(n)}$ queues under the condition $k^{(n)} = o(n^{1/4})$ as the number of servers $n \rightarrow \infty$. The asymptotic independence is in the sense that the total variation distance between the distribution of the workloads of these queues and the distribution of $k^{(n)}$ independent queues goes to 0 as $n \rightarrow \infty$. Consequently, the distance between the distribution of job delay, $T^{(n)}$, and the distribution of the job delay given by independent task delays, $\hat{T}^{(n)}$, goes to 0. This result indicates that assuming independence among the delays of a job's tasks gives a good approximation of job delay when the system is large. Again, due to symmetry, we can focus on the first $k^{(n)}$ queues without loss of generality.

THEOREM 4.1. *Consider an n -server system in the limited fork-join model with $k^{(n)} = o(n^{1/4})$. Let $\pi^{(n, k^{(n)})}$ denote the joint distribution of the steady-state workloads $W_1^{(n)}(\infty), W_2^{(n)}(\infty), \dots, W_{k^{(n)}}^{(n)}(\infty)$, and $\hat{\pi}^{(k^{(n)})}$ denote the product distribution of $k^{(n)}$ i.i.d. random variables, each of which follows a distribution that is the same as the distribution of $W_1^{(n)}(\infty)$. Then*

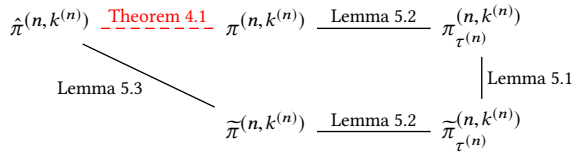
$$\lim_{n \rightarrow \infty} d_{TV}(\pi^{(n, k^{(n)})}, \hat{\pi}^{(k^{(n)})}) = 0. \quad (5)$$

Consequently, the steady-state job delay, $T^{(n)}$, and the job delay given by independent task delays as defined in (3), $\hat{T}^{(n)}$, satisfy

$$\lim_{n \rightarrow \infty} \sup_{\tau \geq 0} |\mathbb{P}(T^{(n)} \leq \tau) - \mathbb{P}(\hat{T}^{(n)} \leq \tau)| = 0. \quad (6)$$

For the special case where the service times are exponentially distributed, the job delay asymptotics have explicit forms presented in Corollary 4.1 below.

COROLLARY 4.1. *Consider an n -server system in the limited fork-join model with $k^{(n)} = o(n^{1/4})$, job arrival rate $\Lambda^{(n)} = n\lambda/k^{(n)}$,*


Figure 1: Distances in the proof of Theorem 4.1

and exponentially distributed service times with mean $1/\mu$. Then the steady-state job delay, $T^{(n)}$, converges as:

$$\lim_{n \rightarrow \infty} \sup_{\tau \geq 0} \left| \mathbb{P}(T^{(n)} \leq \tau) - \left(1 - e^{-(\mu-\lambda)\tau}\right)^{k^{(n)}} \right| = 0, \quad (7)$$

Specifically, if $k^{(n)} \rightarrow \infty$ as $n \rightarrow \infty$, then

$$\frac{T^{(n)}}{H_{k^{(n)}}/(\mu-\lambda)} \Rightarrow 1, \quad \text{as } n \rightarrow \infty, \quad (8)$$

where $H_{k^{(n)}}$ is the $k^{(n)}$ -th harmonic number, and further,

$$\lim_{n \rightarrow \infty} \frac{\mathbb{E}[T^{(n)}]}{H_{k^{(n)}}/(\mu-\lambda)} = 1. \quad (9)$$

The results above characterize job delay in the asymptotic regime where n goes to infinity. In Theorem 4.2 below, we study the non-asymptotic regime for any n and any $k^{(n)}$ with $k^{(n)} = k \leq n$, and we establish the independence upper bound on job delay.

THEOREM 4.2. Consider an n -server system in the limited fork-join model with $k^{(n)} = k \leq n$. Then the steady-state job delay, $T^{(n)}$, is stochastically upper bounded by the job delay given by independent task delays as defined in (3), $\hat{T}^{(n)}$, i.e.,

$$T^{(n)} \leq_{st} \hat{T}^{(n)}, \quad (10)$$

where “ \leq_{st} ” denotes stochastic dominance. Specifically, for any $\tau \geq 0$,

$$\mathbb{P}(T^{(n)} > \tau) \leq \mathbb{P}(\hat{T}^{(n)} > \tau) = 1 - (F(\tau))^{k^{(n)}}. \quad (11)$$

5 PROOFS OF ASYMPTOTIC INDEPENDENCE AND JOB DELAY ASYMPTOTICS

In this section, we prove the asymptotic independence and job delay asymptotics in Theorem 4.1 and Corollary 4.1.

Proof Sketch. To prove Theorem 4.1, we couple each n -server system in the limited fork-join model, which we refer to as system $\mathcal{S}^{(n)}$, with a system $\tilde{\mathcal{S}}^{(n)}$ in which the first $k^{(n)}$ queues are independent. We will specify $\tilde{\mathcal{S}}^{(n)}$ below. Let $\tilde{W}_i^{(n)}(t)$ denote the workload of server i at time t in system $\tilde{\mathcal{S}}^{(n)}$. Let $\tilde{W}^{(n,k^{(n)})}(t) = (\tilde{W}_1^{(n)}(t), \dots, \tilde{W}_{k^{(n)}}^{(n)}(t))$ and $W^{(n,k^{(n)})}(t) = (W_1^{(n)}(t), \dots, W_{k^{(n)}}^{(n)}(t))$. Then the proof will proceed in the following three steps, where we break down the distance $d_{TV}(\pi^{(n,k^{(n)})}, \hat{\pi}^{(k^{(n)})})$ in Theorem 4.1 into three parts, illustrated in Figure 1.

(i) We carefully choose a finite time $\tau^{(n)}$ and consider systems $\mathcal{S}^{(n)}$ and $\tilde{\mathcal{S}}^{(n)}$ at time $\tau^{(n)}$. We show in Lemma 5.1 that the distribution of $W^{(n,k^{(n)})}(\tau^{(n)})$, denoted by $\pi_{\tau^{(n)}}^{(n,k^{(n)})}$, approaches the distribution of $\tilde{W}^{(n,k^{(n)})}(\tau^{(n)})$, denoted by $\tilde{\pi}_{\tau^{(n)}}^{(n,k^{(n)})}$, as $n \rightarrow \infty$.

(ii) We show in Lemma 5.2 that in both systems $\mathcal{S}^{(n)}$ and $\tilde{\mathcal{S}}^{(n)}$, the finite-time distributions $\pi_{\tau^{(n)}}^{(n,k^{(n)})}$ and $\tilde{\pi}_{\tau^{(n)}}^{(n,k^{(n)})}$ are close to the stationary distributions, $\pi^{(n,k^{(n)})}$ and $\tilde{\pi}^{(n,k^{(n)})}$, respectively.

(iii) We show in Lemma 5.3 that the stationary distribution $\tilde{\pi}^{(n,k^{(n)})}$ in system $\tilde{\mathcal{S}}^{(n)}$ is close to the product distribution $\hat{\pi}^{(k^{(n)})}$ in Theorem 4.1. Note that both $\tilde{\pi}^{(n,k^{(n)})}$ and $\hat{\pi}^{(k^{(n)})}$ are for $k^{(n)}$ independent workloads, but we will see that their loads are different.

Coupling. Now we specify the coupling between $\mathcal{S}^{(n)}$ and $\tilde{\mathcal{S}}^{(n)}$. Both systems have n servers and the queues are all empty at time 0, i.e., $W_i^{(n)}(0) = \tilde{W}_i^{(n)}(0) = 0$ for all $i = 1, \dots, n$. When there is a job arrival to system $\mathcal{S}^{(n)}$, we let a job also arrive to system $\tilde{\mathcal{S}}^{(n)}$. Recall that the job arrival in $\mathcal{S}^{(n)}$ selects $k^{(n)}$ distinct queues uniformly at random and sends one task to each queue. If it selects at most one queue from the set $\{1, 2, \dots, k^{(n)}\}$, then we let the job arrival in $\tilde{\mathcal{S}}^{(n)}$ send its tasks to queues with the same indices as those in $\mathcal{S}^{(n)}$. Otherwise, suppose it selects queues i_1, i_2, \dots, i_m from $\{1, 2, \dots, k^{(n)}\}$ with $2 \leq m \leq k^{(n)}$. Then we let the job arrival in $\tilde{\mathcal{S}}^{(n)}$ send one task to a queue chosen uniformly at random from i_1, i_2, \dots, i_m , kill the other $m-1$ tasks, and send the remaining $k^{(n)}-m$ tasks to queues with the same indices as those in $\mathcal{S}^{(n)}$. For each pair of tasks in $\mathcal{S}^{(n)}$ and $\tilde{\mathcal{S}}^{(n)}$ that are sent to queues with the same indices, we let them have the same service time.

It can be verified that in system $\tilde{\mathcal{S}}^{(n)}$, the queues $1, 2, \dots, k^{(n)}$ are independent M/G/1 queues with arrival rate $\tilde{\lambda}^{(n)}$ and mean service time $1/\mu$, where

$$\tilde{\lambda}^{(n)} = \frac{\Lambda^{(n)}}{k^{(n)}} \left(1 - \frac{\binom{n-k^{(n)}}{n}}{\binom{n}{k^{(n)}}}\right). \quad (12)$$

Let $\tilde{\rho}^{(n)} = \frac{\tilde{\lambda}^{(n)}}{\mu}$ denote the load of each queue. Note that $\tilde{\lambda}^{(n)} < \lambda$ but $\tilde{\lambda}^{(n)} \rightarrow \lambda$ as $n \rightarrow \infty$. Specifically,

$$\lambda - \tilde{\lambda}^{(n)} = O\left(\frac{(k^{(n)})^2}{n}\right).$$

5.1 Lemmas Needed for Theorem 4.1

We first show in Lemma 5.1 that, over a finite time interval with proper length, any $k^{(n)}$ queues in the n -server system $\mathcal{S}^{(n)}$ are asymptotically independent as the number of servers $n \rightarrow \infty$.

LEMMA 5.1. For any time $\tau^{(n)}$ with $\tau^{(n)} = O\left(\frac{n^{1/2}}{k^{(n)}}\right)$,

$$d_{TV}\left(\pi_{\tau^{(n)}}^{(n,k^{(n)})}, \tilde{\pi}_{\tau^{(n)}}^{(n,k^{(n)})}\right) = O\left(\left(\frac{k^{(n)}}{n^{1/4}}\right)^2\right), \quad (13)$$

which goes to 0 as $n \rightarrow \infty$.

Lemma 5.2 states that the time interval $\tau^{(n)}$ in Lemma 5.1 is long enough for the systems $\mathcal{S}^{(n)}$ and $\tilde{\mathcal{S}}^{(n)}$ to be close to steady state.

LEMMA 5.2. For any time $\tau^{(n)}$ with $\tau^{(n)} = O\left(\frac{n^{1/2}}{k^{(n)}}\right)$,

$$d_{TV}\left(\pi_{\tau^{(n)}}^{(n,k^{(n)})}, \pi^{(n,k^{(n)})}\right) = O\left(\left(\frac{k^{(n)}}{n^{1/4}}\right)^2\right), \quad (14)$$

and

$$d_{TV}\left(\tilde{\pi}_{\tau^{(n)}}^{(n,k^{(n)})}, \tilde{\pi}^{(n,k^{(n)})}\right) = O\left(\left(\frac{k^{(n)}}{n^{1/4}}\right)^2\right). \quad (15)$$

The distribution $\tilde{\pi}^{(n,k^{(n)})}$ is the joint distribution of the steady-state workloads of $k^{(n)}$ independent queues, each with arrival rate $\tilde{\lambda}^{(n)}$ and mean service time $1/\mu$. Since $\tilde{\lambda}^{(n)} \rightarrow \lambda$ as $n \rightarrow \infty$, $\tilde{\pi}^{(n,k^{(n)})}$ approaches the product distribution $\hat{\pi}^{(k^{(n)})}$ in Theorem 4.1, which is for $k^{(n)}$ independent queues each with arrival rate λ and mean service time $1/\mu$. This is formally stated in Lemma 5.3.

LEMMA 5.3.

$$d_{TV}\left(\tilde{\pi}^{(n,k^{(n)})}, \hat{\pi}^{(k^{(n)})}\right) = O\left(\left(\frac{k^{(n)}}{n^{1/4}}\right)^2\right). \quad (16)$$

5.2 Proof of Theorem 4.1 Given Lemmas

PROOF. The proof of the asymptotic independence in (5) in Theorem 4.1 is straightforward given the lemmas. Pick any $\tau^{(n)}$ with $\tau^{(n)} = O\left(\frac{n^{1/2}}{k^{(n)}}\right)$. Then

$$\begin{aligned} & d_{TV}\left(\pi^{(n,k^{(n)})}, \hat{\pi}^{(k^{(n)})}\right) \\ & \leq d_{TV}\left(\pi^{(n,k^{(n)})}, \pi_{\tau^{(n)}}^{(n,k^{(n)})}\right) + d_{TV}\left(\pi_{\tau^{(n)}}^{(n,k^{(n)})}, \tilde{\pi}_{\tau^{(n)}}^{(n,k^{(n)})}\right) \\ & \quad + d_{TV}\left(\tilde{\pi}_{\tau^{(n)}}^{(n,k^{(n)})}, \tilde{\pi}^{(n,k^{(n)})}\right) + d_{TV}\left(\tilde{\pi}^{(n,k^{(n)})}, \hat{\pi}^{(k^{(n)})}\right) \\ & = O\left(\left(\frac{k^{(n)}}{n^{1/4}}\right)^2\right). \end{aligned}$$

Therefore, under the condition that $k^{(n)} = o(n^{1/4})$, there holds

$$\lim_{n \rightarrow \infty} d_{TV}\left(\pi^{(n,k^{(n)})}, \hat{\pi}^{(k^{(n)})}\right) = 0.$$

Next we prove the job delay asymptotics in (6) in Theorem 4.1. Recall that $\pi^{(n,k^{(n)})}$ and $\hat{\pi}^{(k^{(n)})}$ are distributions of workloads. Below we compute the distributions of $T^{(n)}$ and $\hat{T}^{(n)}$ using $\pi^{(n,k^{(n)})}$ and $\hat{\pi}^{(k^{(n)})}$, which allows us to bound the distance between the distributions of $T^{(n)}$ and $\hat{T}^{(n)}$ using $d_{TV}\left(\pi^{(n,k^{(n)})}, \hat{\pi}^{(k^{(n)})}\right)$. By the representations of $T^{(n)}$ and $\hat{T}^{(n)}$ in (2) and (3), we have that for any $\tau \geq 0$,

$$\begin{aligned} & \left| \mathbb{P}(T^{(n)} \leq \tau) - \mathbb{P}(\hat{T}^{(n)} \leq \tau) \right| \\ & = \left| \int_{\mathbf{w} \in \mathbb{R}_+^{k^{(n)}}} \left(\prod_{i=1}^{k^{(n)}} \mathbb{P}(w_i + X_i \leq \tau) \right) d\pi^{(n,k^{(n)})}(\mathbf{w}) \right. \\ & \quad \left. - \int_{\mathbf{w} \in \mathbb{R}_+^{k^{(n)}}} \left(\prod_{i=1}^{k^{(n)}} \mathbb{P}(w_i + X_i \leq \tau) \right) d\hat{\pi}^{(k^{(n)})}(\mathbf{w}) \right| \\ & \leq \int_{\mathbf{w} \in \mathbb{R}_+^{k^{(n)}}} \left(\prod_{i=1}^{k^{(n)}} \mathbb{P}(w_i + X_i \leq \tau) \right) \cdot |d\pi^{(n,k^{(n)})}(\mathbf{w}) - d\hat{\pi}^{(k^{(n)})}(\mathbf{w})| \\ & \leq \int_{\mathbf{w} \in \mathbb{R}_+^{k^{(n)}}} |d\pi^{(n,k^{(n)})}(\mathbf{w}) - d\hat{\pi}^{(k^{(n)})}(\mathbf{w})| \\ & = 2d_{TV}\left(\pi^{(n,k^{(n)})}, \hat{\pi}^{(k^{(n)})}\right). \end{aligned}$$

Therefore,

$$\begin{aligned} & \lim_{n \rightarrow \infty} \sup_{\tau \geq 0} \left| \mathbb{P}(T^{(n)} \leq \tau) - \mathbb{P}(\hat{T}^{(n)} \leq \tau) \right| \\ & \leq \lim_{n \rightarrow \infty} 2d_{TV}\left(\pi^{(n,k^{(n)})}, \hat{\pi}^{(k^{(n)})}\right) \\ & = 0. \end{aligned}$$

□

5.3 Proof of Lemmas

Proof of Lemma 5.1

PROOF. In order to bound $d_{TV}\left(\pi_{\tau^{(n)}}^{(n,k^{(n)})}, \tilde{\pi}_{\tau^{(n)}}^{(n,k^{(n)})}\right)$, we first write

$$\begin{aligned} & d_{TV}\left(\pi_{\tau^{(n)}}^{(n,k^{(n)})}, \tilde{\pi}_{\tau^{(n)}}^{(n,k^{(n)})}\right) \\ & \leq \mathbb{P}\left(\mathbf{W}^{(n,k^{(n)})}(\tau^{(n)}) \neq \widetilde{\mathbf{W}}^{(n,k^{(n)})}(\tau^{(n)})\right) \\ & \leq \mathbb{P}\left(\mathbf{W}^{(n,k^{(n)})}(t) \neq \widetilde{\mathbf{W}}^{(n,k^{(n)})}(t) \text{ for some } t \in [0, \tau^{(n)}]\right), \end{aligned}$$

where the first inequality follows from a standard property of total variation distance. By the coupling between $\mathcal{S}^{(n)}$ and $\widetilde{\mathcal{S}}^{(n)}$, $\mathbf{W}^{(n,k^{(n)})}(t)$ and $\widetilde{\mathbf{W}}^{(n,k^{(n)})}(t)$ are different for some time $t \in [0, \tau^{(n)}]$ only when at least one job arrival during $[0, \tau^{(n)}]$ selects more than one queue from $\{1, \dots, k^{(n)}\}$ in system $\mathcal{S}^{(n)}$. We denote this event by \mathcal{E} . Then

$$d_{TV}\left(\pi_{\tau^{(n)}}^{(n,k^{(n)})}, \tilde{\pi}_{\tau^{(n)}}^{(n,k^{(n)})}\right) \leq \mathbb{P}(\mathcal{E}).$$

So it suffices to prove that

$$\mathbb{P}(\mathcal{E}) = O\left(\left(\frac{k^{(n)}}{n^{1/4}}\right)^2\right)$$

for $\tau^{(n)}$ with $\tau^{(n)} = O\left(\frac{n^{1/2}}{k^{(n)}}\right)$. The remainder of this proof is dedicated to bounding $\mathbb{P}(\mathcal{E})$.

Let $p^{(n)}$ denote the probability for a job arrival to select less than or equal to 1 queue from queues $1, 2, \dots, k^{(n)}$ in system $\mathcal{S}^{(n)}$. Then

$$p^{(n)} = \frac{\binom{n-k^{(n)}}{k^{(n)}}}{\binom{n}{k^{(n)}}} + \frac{k^{(n)} \binom{n-k^{(n)}}{k^{(n)}-1}}{\binom{n}{k^{(n)}}}.$$

Let A be the number of job arrivals during $[0, \tau^{(n)}]$. Then

$$\begin{aligned} \mathbb{P}(\mathcal{E}) & = \sum_{j=0}^{\infty} \mathbb{P}(A = j) \mathbb{P}(\mathcal{E} | A = j) \\ & = \sum_{j=0}^{\infty} \frac{(\Lambda^{(n)} \tau^{(n)})^j e^{-\Lambda^{(n)} \tau^{(n)}}}{j!} (1 - (p^{(n)})^j) \\ & = 1 - e^{-\Lambda^{(n)} \tau^{(n)} (1 - p^{(n)})}, \end{aligned} \quad (17)$$

where (17) follows from the definition of the Poisson generating function. We calculate $p^{(n)}$ as follows:

$$\begin{aligned} p^{(n)} & = \frac{\binom{n-k^{(n)}}{k^{(n)}}}{\binom{n}{k^{(n)}}} + \frac{k^{(n)} \binom{n-k^{(n)}}{k^{(n)}-1}}{\binom{n}{k^{(n)}}} \\ & = \frac{(n-k^{(n)})!}{(n-2k^{(n)})!} \frac{(n-k^{(n)})!}{n!} \left(1 + \frac{(k^{(n)})^2}{n-2k^{(n)}+1} \right) \end{aligned}$$

$$\begin{aligned}
 &= \left(1 - \frac{k^{(n)}}{n}\right) \left(1 - \frac{k^{(n)}}{n-1}\right) \cdots \left(1 - \frac{k^{(n)}}{n-k^{(n)}+1}\right) \\
 &\quad \cdot \left(1 + \frac{(k^{(n)})^2}{n-2k^{(n)}+1}\right) \\
 &\geq \left(1 - \frac{k^{(n)}}{n-k^{(n)}+1}\right)^{k^{(n)}} \left(1 + \frac{(k^{(n)})^2}{n-k^{(n)}+1}\right).
 \end{aligned}$$

Since

$$\begin{aligned}
 &\left(1 - \frac{k^{(n)}}{n-k^{(n)}+1}\right)^{k^{(n)}} \\
 &= e^{k^{(n)} \ln\left(1 - \frac{k^{(n)}}{n-k^{(n)}+1}\right)} \\
 &= e^{-\frac{(k^{(n)})^2}{n-k^{(n)}+1} + O\left(\frac{(k^{(n)})^3}{(n-k^{(n)}+1)^2}\right)} \\
 &= 1 - \frac{(k^{(n)})^2}{n-k^{(n)}+1} + O\left(\frac{(k^{(n)})^4}{(n-k^{(n)}+1)^2}\right),
 \end{aligned}$$

we have

$$\begin{aligned}
 1 - p^{(n)} &\leq 1 - \left(1 - \frac{k^{(n)}}{n-k^{(n)}+1}\right)^{k^{(n)}} \left(1 + \frac{(k^{(n)})^2}{n-k^{(n)}+1}\right) \\
 &= O\left(\frac{(k^{(n)})^4}{(n-k^{(n)}+1)^2}\right).
 \end{aligned}$$

Recall that

$$\Lambda^{(n)} k^{(n)} = n\lambda, \quad \tau^{(n)} = O\left(\frac{n^{1/2}}{k^{(n)}}\right).$$

Thus,

$$\Lambda^{(n)} \tau^{(n)} (1 - p^{(n)}) = O\left(\left(\frac{k^{(n)}}{n^{1/4}}\right)^2\right).$$

Consequently, inserting this to (17) yields

$$\mathbb{P}(\mathcal{E}) = O\left(\left(\frac{k^{(n)}}{n^{1/4}}\right)^2\right),$$

which completes the proof of Lemma 5.1. \square

Proof of Lemma 5.2

PROOF. We first prove (14). For system $\mathcal{S}^{(n)}$, we consider the following coupling between two copies of the workload process. In one copy, the system starts from empty queues, i.e., this is the workload process $(\mathbf{W}^{(n)}(t), t \geq 0)$ we have introduced. The other copy, which we denote by $(\overline{\mathbf{W}}^{(n)}(t), t \geq 0)$, starts from its stationary distribution, i.e., the distribution of $\overline{\mathbf{W}}^{(n)}(0)$ is $\pi^{(n)}$. Then the distribution of $\overline{\mathbf{W}}^{(n)}(t)$ is $\pi^{(n)}$ for any t . We let these two copies have the same arrival processes, and each arriving task has the same service time under both queueing processes.

By this coupling, $W_i^{(n)}(t) \leq \overline{W}_i^{(n)}(t)$ for any time t and any i . For each $i = 1, 2, \dots, k^{(n)}$, let $\tau_i^{(n)}$ be the earliest time that the workload $\overline{W}_i^{(n)}$ is 0, i.e.,

$$\tau_i^{(n)} = \min\{t : \overline{W}_i^{(n)}(u) = 0 \text{ for some } u \in [0, t]\}.$$

Let

$$\tau_0^{(n)} = \max\{\tau_1^{(n)}, \dots, \tau_{k^{(n)}}^{(n)}\}.$$

Then $W_i^{(n)}(t) = \overline{W}_i^{(n)}(t)$ for any $t \geq \tau_0^{(n)}$ and any $i = 1, 2, \dots, k^{(n)}$. To show (14) in Lemma 5.2, which we restate here for reference

$$d_{TV}\left(\pi_{\tau^{(n)}}^{(n, k^{(n)})}, \pi^{(n, k^{(n)})}\right) = O\left(\left(\frac{k^{(n)}}{n^{1/4}}\right)^2\right), \quad (14) \text{ (Restated)}$$

it suffices to prove that

$$\mathbb{P}(\tau_0^{(n)} > \tau^{(n)}) = O\left(\left(\frac{k^{(n)}}{n^{1/4}}\right)^2\right). \quad (18)$$

To see that this is sufficient, we first note that

$$\begin{aligned}
 &d_{TV}\left(\pi_{\tau^{(n)}}^{(n, k^{(n)})}, \pi^{(n, k^{(n)})}\right) \\
 &\leq \mathbb{P}\left(\mathbf{W}^{(n, k^{(n)})}(\tau^{(n)}) \neq \overline{\mathbf{W}}^{(n, k^{(n)})}(\tau^{(n)})\right). \quad (19)
 \end{aligned}$$

By the definition of $\tau_0^{(n)}$, $\mathbf{W}^{(n, k^{(n)})}(\tau^{(n)}) \neq \overline{\mathbf{W}}^{(n, k^{(n)})}(\tau^{(n)})$ if and only if $\tau_0^{(n)} > \tau^{(n)}$. So (19) further implies

$$d_{TV}\left(\pi_{\tau^{(n)}}^{(n, k^{(n)})}, \pi^{(n, k^{(n)})}\right) \leq \mathbb{P}(\tau_0^{(n)} > \tau^{(n)}), \quad (20)$$

and thus (18) implies (14).

Now we prove (18). Note that the distribution of $\tau_i^{(n)}$ does not depend on n since each individual $(\overline{W}_i^{(n)}(t), t \geq 0)$ evolves as an M/G/1 queue with arrival rate λ and service time distribution G , and $\tau_i^{(n)}$ is a busy period started by the amount of work in steady state. Thus, by standard results on busy periods (see, e.g., [17]),

$$\mathbb{E}[\tau_i^{(n)}] = \frac{\lambda g_2}{2(1-\rho)^2}, \quad (21)$$

where g_2 is the second moment of G , which is a constant. By Markov's inequality,

$$\mathbb{P}(\tau_i^{(n)} > \tau^{(n)}) \leq \frac{\mathbb{E}[\tau_i^{(n)}]}{\tau^{(n)}}. \quad (22)$$

Since $\tau_0^{(n)} = \max\{\tau_1^{(n)}, \tau_2^{(n)}, \dots, \tau_{k^{(n)}}^{(n)}\}$, by the union bound we have

$$\begin{aligned}
 \mathbb{P}(\tau_0^{(n)} > \tau^{(n)}) &\leq \sum_{i=1}^{k^{(n)}} \mathbb{P}(\tau_i^{(n)} > \tau^{(n)}) \\
 &\leq \frac{k^{(n)}}{\tau^{(n)}} \frac{\lambda g_2}{2(1-\rho)^2} \\
 &= O\left(\left(\frac{k^{(n)}}{n^{1/4}}\right)^2\right).
 \end{aligned}$$

This is (18) and thus it completes the proof of (14).

The proof of (15) in Lemma 5.2 is very much similar to the proof of (14). We obtain (15) by noting that each $(\overline{W}_i^{(n)}(t), t \geq 0)$ with $i = 1, 2, \dots, k^{(n)}$ is an M/G/1 queue with arrival rate $\tilde{\lambda}^{(n)} < \lambda$ and following arguments similar to those in the proof of (14). \square

Proof of Lemma 5.3

PROOF. This proof has a similar flavor to the proofs of Lemmas 5.1 and 5.2. Recall that $(\overline{\mathbf{W}}^{(n, k^{(n)})}(t), t \geq 0)$, the workload processes of the first $k^{(n)}$ queues in system $\tilde{\mathcal{S}}^{(n)}$, are $k^{(n)}$ independent M/G/1 queues each with arrival rate $\tilde{\lambda}^{(n)}$ and service

time distribution G . We couple this with $(\hat{W}^{(k^{(n)})}(t), t \geq 0)$, where $\hat{W}^{(k^{(n)})}(t) = (\hat{W}_1(t), \dots, \hat{W}_{k^{(n)}}(t))$ is the workload vector of $k^{(n)}$ independent M/G/1 queues each with arrival rate λ and service time distribution G . Then $\hat{\pi}^{(k^{(n)})}$ is its stationary distribution. We will prove the bound on $d_{TV}(\tilde{\pi}^{(n, k^{(n)})}, \hat{\pi}^{(k^{(n)})})$ by showing that $(\tilde{W}^{(n, k^{(n)})}(t), t \geq 0)$ and $(\hat{W}^{(k^{(n)})}(t), t \geq 0)$ are close.

Now we specify the coupling. All the queues start from empty, i.e., $\hat{W}_i(0) = \tilde{W}_i^{(n)}(0) = 0$ for all $i = 1, 2, \dots, k^{(n)}$. When there is a task arrival to some queue of $\hat{W}^{(k^{(n)})}$, we let a task arrive to the corresponding queue of $\tilde{W}^{(n, k^{(n)})}$ with probability $\frac{\tilde{\lambda}^{(n)}}{\lambda}$, and let these two tasks require the same service time. So with probability $1 - \frac{\tilde{\lambda}^{(n)}}{\lambda}$ there is no task arrival to $\tilde{W}^{(n, k^{(n)})}$.

We pick a time $\tau^{(n)} = O\left(\frac{n^{1/2}}{k^{(n)}}\right)$. Let $\hat{\pi}_{\tau^{(n)}}^{(k^{(n)})}$ denote the distribution of $\hat{W}^{(k^{(n)})}(\tau^{(n)})$. Then

$$d_{TV}\left(\tilde{\pi}^{(n, k^{(n)})}, \hat{\pi}^{(k^{(n)})}\right) \leq d_{TV}\left(\tilde{\pi}_{\tau^{(n)}}^{(n, k^{(n)})}, \hat{\pi}_{\tau^{(n)}}^{(k^{(n)})}\right) + d_{TV}\left(\tilde{\pi}_{\tau^{(n)}}^{(n, k^{(n)})}, \tilde{\pi}^{(n, k^{(n)})}\right) + d_{TV}\left(\hat{\pi}_{\tau^{(n)}}^{(k^{(n)})}, \hat{\pi}^{(k^{(n)})}\right).$$

Noting Lemma 5.2, we have

$$d_{TV}\left(\tilde{\pi}_{\tau^{(n)}}^{(n, k^{(n)})}, \tilde{\pi}^{(n, k^{(n)})}\right) = O\left(\left(\frac{k^{(n)}}{n^{1/4}}\right)^2\right), \quad (23)$$

$$d_{TV}\left(\hat{\pi}_{\tau^{(n)}}^{(k^{(n)})}, \hat{\pi}^{(k^{(n)})}\right) = O\left(\left(\frac{k^{(n)}}{n^{1/4}}\right)^2\right). \quad (24)$$

Next we bound $d_{TV}\left(\tilde{\pi}_{\tau^{(n)}}^{(n, k^{(n)})}, \hat{\pi}_{\tau^{(n)}}^{(k^{(n)})}\right)$ using arguments similar to those in the proof of Lemma 5.1. By the coupling, $\hat{W}^{(k^{(n)})}(t)$ and $\tilde{W}^{(n, k^{(n)})}(t)$ are different for some $t \in [0, \tau^{(n)}]$ only when some task arrives to $\hat{W}^{(k^{(n)})}$ but not to $\tilde{W}^{(n, k^{(n)})}$. We denote this event by \mathcal{E} . Then

$$d_{TV}\left(\tilde{\pi}_{\tau^{(n)}}^{(n, k^{(n)})}, \hat{\pi}_{\tau^{(n)}}^{(k^{(n)})}\right) \leq \mathbb{P}(\mathcal{E}).$$

So the remainder of this proof is dedicated to bounding $\mathbb{P}(\mathcal{E})$.

Consider the time interval $[0, \tau^{(n)}]$. Let A be the number of task arrivals to $\hat{W}^{(k^{(n)})}$ during this time interval. Then

$$\begin{aligned} \mathbb{P}(\mathcal{E}) &= \sum_{j=0}^{\infty} \mathbb{P}(A = j) \mathbb{P}(\mathcal{E} \mid A = j) \\ &\leq \sum_{j=0}^{\infty} \frac{(\lambda k^{(n)} \tau^{(n)})^j e^{-\lambda k^{(n)} \tau^{(n)}}}{j!} j \left(1 - \frac{\tilde{\lambda}^{(n)}}{\lambda}\right) \\ &= k^{(n)} \tau^{(n)} (\lambda - \tilde{\lambda}^{(n)}), \end{aligned} \quad (25)$$

where we have used a union bound for (25). By definition,

$$\begin{aligned} \tilde{\lambda}^{(n)} &= \frac{\Lambda^{(n)}}{k^{(n)}} \left(1 - \frac{\binom{n-k^{(n)}}{k^{(n)}}}{\binom{n}{k^{(n)}}}\right) \\ &\geq \frac{\Lambda^{(n)}}{k^{(n)}} \left(1 - \left(1 - \frac{k^{(n)}}{n}\right)^{k^{(n)}}\right) \\ &= \frac{\Lambda^{(n)}}{k^{(n)}} \left(\frac{(k^{(n)})^2}{n} + O\left(\frac{(k^{(n)})^4}{n^2}\right)\right) \end{aligned}$$

$$= \lambda + O\left(\frac{(k^{(n)})^2}{n}\right).$$

Therefore,

$$\mathbb{P}(\mathcal{E}) = O\left(\left(\frac{k^{(n)}}{n^{1/4}}\right)^2\right),$$

which completes the proof. \square

5.4 Proof of Corollary 4.1

COROLLARY 4.1 (RESTATEd). *Consider an n -server system in the limited fork-join model with $k^{(n)} = o(n^{1/4})$, job arrival rate $\Lambda^{(n)} = n\lambda/k^{(n)}$, and exponentially distributed service times with mean $1/\mu$. Then the steady-state job delay, $T^{(n)}$, converges as:*

$$\lim_{n \rightarrow \infty} \sup_{\tau \geq 0} \left| \mathbb{P}(T^{(n)} \leq \tau) - \left(1 - e^{-(\mu-\lambda)\tau}\right)^{k^{(n)}} \right| = 0, \quad (7) \text{ (Restated)}$$

Specifically, if $k^{(n)} \rightarrow \infty$ as $n \rightarrow \infty$, then

$$\frac{T^{(n)}}{H_{k^{(n)}}/(\mu-\lambda)} \Rightarrow 1, \quad \text{as } n \rightarrow \infty, \quad (8) \text{ (Restated)}$$

where $H_{k^{(n)}}$ is the $k^{(n)}$ -th harmonic number, and further,

$$\lim_{n \rightarrow \infty} \frac{\mathbb{E}[T^{(n)}]}{H_{k^{(n)}}/(\mu-\lambda)} = 1. \quad (9) \text{ (Restated)}$$

PROOF. When the service times are exponentially distributed, each queue is an M/M/1 queue and thus the cdf of the task delay at each queue, F , is given by

$$F(\tau) = 1 - e^{-(\mu-\lambda)\tau}.$$

Then the convergence in (7) directly follows from Theorem 4.1.

To prove the weak convergence of $\frac{T^{(n)}}{H_{k^{(n)}}/(\mu-\lambda)}$ in (8), we first note that

$$\frac{\hat{T}^{(n)}}{H_{k^{(n)}}/(\mu-\lambda)} \Rightarrow 1, \quad \text{as } n \rightarrow \infty,$$

which is a direct implication of the standard result in the asymptotic theory of extremes (see, e.g., Theorem 8.12 in [7]). Combining this with (7) yields (8).

To prove the convergence of the expectation in (9), we actually need the stochastic dominance shown in Theorem 4.2. The expectation in (9) can be written as

$$\frac{\mathbb{E}[T^{(n)}]}{H_{k^{(n)}}/(\mu-\lambda)} = \int_0^{\infty} \mathbb{P}\left(\frac{T^{(n)}}{H_{k^{(n)}}/(\mu-\lambda)} > \tau\right) d\tau.$$

By Theorem 4.2, for any $\tau \geq 0$,

$$\mathbb{P}\left(\frac{T^{(n)}}{H_{k^{(n)}}/(\mu-\lambda)} > \tau\right) \leq \mathbb{P}\left(\frac{\hat{T}^{(n)}}{H_{k^{(n)}}/(\mu-\lambda)} > \tau\right).$$

Since

$$\frac{\hat{T}^{(n)}}{H_{k^{(n)}}/(\mu-\lambda)} \Rightarrow 1, \quad \text{as } n \rightarrow \infty,$$

and

$$\frac{\mathbb{E}[\hat{T}^{(n)}]}{H_{k^{(n)}}/(\mu-\lambda)} = \int_0^{\infty} \mathbb{P}\left(\frac{\hat{T}^{(n)}}{H_{k^{(n)}}/(\mu-\lambda)} > \tau\right) d\tau = 1,$$

by the General Lebesgue Dominated Convergence Theorem (see, e.g., Theorem 19 in [38]), we can take the limit inside the integral and using (8), get

$$\begin{aligned} \lim_{n \rightarrow \infty} \frac{\mathbb{E}[T^{(n)}]}{H_{k^{(n)}}/(\mu - \lambda)} &= \int_0^\infty \lim_{n \rightarrow \infty} \mathbb{P}\left(\frac{T^{(n)}}{H_{k^{(n)}}/(\mu - \lambda)} > \tau\right) d\tau \\ &= \int_0^1 1 d\tau \\ &= 1, \end{aligned}$$

which completes the proof. \square

5.5 Explaining $o(n^{1/4})$ and Possible Extensions

In this section we first explain in a bit more detail where the condition $k^{(n)} = o(n^{1/4})$ comes from. Recall that in our proof of Theorem 4.1, we choose a finite time instance $\tau^{(n)}$ and decompose the distance $d_{TV}\left(\pi_{\tau^{(n)}}^{(n, k^{(n)})}, \hat{\pi}^{(k^{(n)})}\right)$ in Theorem 4.1 into the four distances in Lemmas 5.1–5.3 accordingly. To understand the result, it helps to return to these lemmas. Instead of directly choosing $\tau^{(n)}$ as $O\left(\frac{n^{1/2}}{k^{(n)}}\right)$, now we keep $\tau^{(n)}$ as a variable.

- The distance in (13) of Lemma 5.1 becomes

$$d_{TV}\left(\pi_{\tau^{(n)}}^{(n, k^{(n)})}, \tilde{\pi}_{\tau^{(n)}}^{(n, k^{(n)})}\right) = \tau^{(n)} O\left(\frac{(k^{(n)})^3}{n}\right). \quad (26)$$

This is the distance between the limited fork-join system, $\mathcal{S}^{(n)}$, and the system whose first $k^{(n)}$ queues are independent, $\tilde{\mathcal{S}}^{(n)}$, at time $\tau^{(n)}$. Intuitively, the longer $\tau^{(n)}$ is, the more jobs are expected to arrive during $[0, \tau^{(n)}]$, and thus the more likely it is that the first $k^{(n)}$ queues in $\mathcal{S}^{(n)}$ deviate from the independent queues in $\tilde{\mathcal{S}}^{(n)}$. Careful calculation yields that the distance between $\mathcal{S}^{(n)}$ and $\tilde{\mathcal{S}}^{(n)}$ at time $\tau^{(n)}$ increases linearly with $\tau^{(n)}$ as shown in (26).

- The distances in (14) and (15) in Lemma 5.2 become

$$d_{TV}\left(\pi_{\tau^{(n)}}^{(n, k^{(n)})}, \pi_{\tau^{(n)}}^{(n, k^{(n)})}\right) = \frac{1}{\tau^{(n)}} O(k^{(n)}), \quad (27)$$

and

$$d_{TV}\left(\tilde{\pi}_{\tau^{(n)}}^{(n, k^{(n)})}, \tilde{\pi}_{\tau^{(n)}}^{(n, k^{(n)})}\right) = \frac{1}{\tau^{(n)}} O(k^{(n)}). \quad (28)$$

These are the distances between the system state at time $\tau^{(n)}$ and the steady state for systems $\mathcal{S}^{(n)}$ and $\tilde{\mathcal{S}}^{(n)}$, respectively. Intuitively, a long $\tau^{(n)}$ brings the system close to steady state. So as shown in (27) and (28), these distances decrease with $\tau^{(n)}$.

- The distance in (16) of Lemma 5.3 becomes

$$d_{TV}\left(\tilde{\pi}_{\tau^{(n)}}^{(n, k^{(n)})}, \hat{\pi}^{(k^{(n)})}\right) = \tau^{(n)} O\left(\frac{(k^{(n)})^3}{n}\right) + \frac{1}{\tau^{(n)}} O(k^{(n)}). \quad (29)$$

This distance has this sum form since it is bounded in a similar way to the distances in Lemma 5.1 and 5.2.

To make the sum of the distances in (26)–(29) as small as possible, we should choose $\tau^{(n)}$ such that these distances are equal, which leads to the choice

$$\tau^{(n)} = O\left(\frac{n^{1/2}}{k^{(n)}}\right)$$

and a total distance of

$$O\left(\left(\frac{k^{(n)}}{n^{1/4}}\right)^2\right).$$

Therefore, for this distance to converge to zero, we need the condition that $k^{(n)} = o(n^{1/4})$.

We acknowledge that $k^{(n)} = o(n^{1/4})$ is not the optimal threshold for asymptotic independence to hold. In fact, one can improve the result to $k^{(n)} = o\left(\frac{n^{1/3}}{\log^m n}\right)$ for some m using more delicate bounding techniques in Lemma 5.2. However, our main contribution is the generalization from the asymptotic independence of a *constant* number of queues to that of a *growing* number of queues. So we choose to present the case $k^{(n)} = o(n^{1/4})$ to not obscure the main idea with technical details.

As an interesting complement to the asymptotic independence result, we also show that when $k^{(n)} = \Theta(n)$ and the service times are exponentially distributed, any number of queues are *not* asymptotically independent. The formal statement and its proof are given in Theorem A.1 in Appendix A. It then remains an open problem whether there exists a critical value for $k^{(n)}$, where smaller $k^{(n)}$ yields asymptotic independence and larger $k^{(n)}$ does not.

6 NON-ASYMPTOTIC REGIME: PROOF OF INDEPENDENCE UPPER BOUND

THEOREM 4.2 (RESTATED). *Consider an n -server system in the limited fork-join model with $k^{(n)} \leq n$. Then the steady-state job delay, $T^{(n)}$, is stochastically upper bounded by the job delay given by independent task delays as defined in (3), $\hat{T}^{(n)}$, i.e.,*

$$T^{(n)} \leq_{st} \hat{T}^{(n)}, \quad (10) \text{ (Restated)}$$

where “ \leq_{st} ” denotes stochastic dominance. Specifically, for any $\tau \geq 0$,

$$\mathbb{P}(T^{(n)} > \tau) \leq \mathbb{P}(\hat{T}^{(n)} > \tau) = 1 - (F(\tau))^{k^{(n)}}. \quad (11) \text{ (Restated)}$$

The main tool we will use is the theory of associated random variables. For convenience of reference, we give the formal definition of association and some properties that we will use in Appendix B. We refer interested readers to [9] for further details. Intuitively, association is a form of positive correlation among random variables. If a set of random variables are associated, then the maximum of them is stochastically upper bounded by the maximum of independent versions of them. Since the delay of a job is the maximum of its task delays, to show Theorem 4.2, it thus suffices to show association of the task delays. This further boils down to showing association among the workloads of any $k^{(n)}$ queues in steady state since each task delay is the workload of the queue that the task is sent to plus the service time of the task.

Such an association result has been proven for the classical fork-join model [35] where $k^{(n)} = n$, but the approach of the proof breaks down once we have $k^{(n)} < n$. The proof idea there is to observe the system at each job arrival time, and show that the numbers of tasks sent to different queues are associated. This proof idea is widely used in the literature to establish association (see, e.g., [23, 40]), but it does not work in the limited fork-join model when $k^{(n)} < n$. We can think of the process of assigning a job’s tasks to queues as a balls-and-bins problem, where the $k^{(n)}$ tasks correspond to $k^{(n)}$

balls, the queues are the bins, and the number of balls thrown in each bin is the number of tasks sent to each queue. When $k^{(n)} = n$, it is obvious that the numbers of balls in the bins are associated since they are all exactly equal to one. But when $k^{(n)} < n$, the numbers of balls in the bins are actually *negatively associated* by a classical result [19]! However, one should not be discouraged since this does not mean that the *steady-state* workloads are negatively associated.

In our proof, we develop a novel technique that we call ‘‘Poisson oversampling’’, where we observe the system not only when jobs arrive but also at the jump times of a Poisson process that is independent of everything else. In the existing approach where the system is observed only at job arrival times, there is always one job arrival at each observation time. But with oversampling, there could be one or zero job arrivals at each observation time. Recall that jobs arrive with rate $\Lambda^{(n)}$. Let the additional Poisson process have rate $\beta^{(n)}$. Then in the corresponding balls-and-bins problem, with probability $\Lambda^{(n)}/(\Lambda^{(n)} + \beta^{(n)})$, $k^{(n)}$ balls are thrown into $k^{(n)}$ distinct bins chosen uniformly at random, and with probability $\beta^{(n)}/(\Lambda^{(n)} + \beta^{(n)})$, there are no balls at all. We will see in the proof that, surprisingly, now the numbers of balls thrown into any $k^{(n)}$ bins become associated with properly chosen $\beta^{(n)}$. This enables us to show association of steady-state workloads.

REMARK. Before we present the proof, we remark that it may be possible to explore the monotonicity of the workload process to establish association [6, 18, 26].¹ However, the results in [6, 18, 26] assume either a finite or a compact state space. It may be possible to generalize the results from a finite state space [6, 18] to a countable state space for some Markov chains, which then can be applied to our problem for certain phase-type service time distributions. To further deal with more general service time distributions, we may be able to utilize the existing results for a compact state space [26]. But there we need to compactify the state space and verify a condition on the generator of the workload process. We do not pursue such an approach here.

PROOF. Recall that

$$T^{(n)} = \max\{T_1^{(n)}, T_2^{(n)}, \dots, T_{k^{(n)}}^{(n)}\},$$

where $T_i^{(n)}$ denotes the steady-state task delay at queue i . Then to prove the stochastic dominance, it suffices to prove that $T_1^{(n)}, T_2^{(n)}, \dots, T_{k^{(n)}}^{(n)}$ are associated [9, Theorem 5.1].

We start by noting that it is sufficient to prove that the steady-state workloads, $W_1^{(n)}(\infty), W_2^{(n)}(\infty), \dots, W_{k^{(n)}}^{(n)}(\infty)$, are associated. The sufficiency follows from the fact that each $T_i^{(n)}$ can be expressed in the following form:

$$T_i^{(n)} = W_i^{(n)}(\infty) + X_i,$$

where $X_1, X_2, \dots, X_{k^{(n)}}$ represent the service times of tasks so they are i.i.d. $\sim G$ and independent of everything else. Then $T_1^{(n)}, T_2^{(n)}, \dots, T_{k^{(n)}}^{(n)}$ are nondecreasing functions of $W_1^{(n)}(\infty), W_2^{(n)}(\infty), \dots, W_{k^{(n)}}^{(n)}(\infty), X_1, X_2, \dots, X_{k^{(n)}}$. So by Properties (P1)

and (P2) in Lemma B.2, $T_1^{(n)}, T_2^{(n)}, \dots, T_{k^{(n)}}^{(n)}$ are associated when $W_1^{(n)}(\infty), W_2^{(n)}(\infty), \dots, W_{k^{(n)}}^{(n)}(\infty)$ are associated.

All that remains is prove the claim that $W_1^{(n)}(\infty), W_2^{(n)}(\infty), \dots, W_{k^{(n)}}^{(n)}(\infty)$ are associated. We will work with a discrete-time Markov chain constructed from the continuous-time workload process $(\mathbf{W}^{(n)}(t), t \geq 0)$. Specifically, we consider a Poisson process, denoted by $(B(t), t \geq 0)$, that is independent of everything else. Let the rate of this Poisson process be $\beta^{(n)}$, which will be specified later in (32). Then we sample the workload process $(\mathbf{W}^{(n)}(t), t \geq 0)$ at time instances right before either a job arrival or an event of the Poisson process $(B(t), t \geq 0)$. Let such time instances be denoted by $\{U_s, s = 0, 1, \dots\}$ with $U_0 = 0$. This gives us a discrete-time Markov chain, which we denote by $(\Phi^{(n)}(s), s = 0, 1, \dots)$, i.e., $\Phi_i^{(n)}(s) = W_i^{(n)}(U_s^-)$, where $W_i^{(n)}(U_s^-)$ is the workload of queue i right before time U_s . Since $(\Phi^{(n)}(s), s = 0, 1, \dots)$ is constructed by sampling the workload process more often than the job arrival process, we call this technique ‘‘Poisson oversampling’’.

We first claim that $(\Phi^{(n)}(s), s = 0, 1, \dots)$ converges to a well-defined steady state $\Phi^{(n)}(\infty)$ and that $\Phi^{(n)}(\infty)$ and $W^{(n)}(\infty)$ are identically distributed. This claim can be proven by showing that $(\Phi^{(n)}(s), s = 0, 1, \dots)$ is aperiodic and positive Harris recurrent and then appealing to the PASTA property [31]. We omit the proof of this claim since the aperiodicity is straightforward to check and the positive Harris recurrence follows from the rather standard Foster-Lyapunov criteria using the quadratic Lyapunov function [32]. With this claim, it then suffices to prove that $\Phi_1^{(n)}(\infty), \Phi_2^{(n)}(\infty), \dots, \Phi_{k^{(n)}}^{(n)}(\infty)$ are associated.

We assume that $\Phi_i^{(n)}(0) = 0$ for every $i = 1, 2, \dots, n$. We will prove that $\Phi_1^{(n)}(s), \Phi_2^{(n)}(s), \dots, \Phi_{k^{(n)}}^{(n)}(s)$ are associated for any $s \geq 0$ by induction on s . Then $\Phi_1^{(n)}(\infty), \Phi_2^{(n)}(\infty), \dots, \Phi_{k^{(n)}}^{(n)}(\infty)$ are associated since $\Phi^{(n)}(s) \Rightarrow \Phi^{(n)}(\infty)$ as $s \rightarrow \infty$ [9].

Base Step: $\Phi_1^{(n)}(0), \Phi_2^{(n)}(0), \dots, \Phi_{k^{(n)}}^{(n)}(0)$ are associated since they are all zero.

Inductive Step: Assuming that $\Phi_1^{(n)}(s), \Phi_2^{(n)}(s), \dots, \Phi_{k^{(n)}}^{(n)}(s)$ are associated for some $s \geq 0$, we will show that $\Phi_1^{(n)}(s+1), \Phi_2^{(n)}(s+1), \dots, \Phi_{k^{(n)}}^{(n)}(s+1)$ are associated. By Lindley equation,

$$\Phi_i^{(n)}(s+1) = \left(\Phi_i^{(n)}(s) + Y_i(s) - \Delta U(s) \right)^+, \quad (30)$$

where $Y_i(s)$ is the service time needed by the task that arrives to queue i at time U_s , and $\Delta U(s) = U_{s+1} - U_s$. Note that at time U_s , there may be no task arrival to queue i , either because there is no job arrival or because there is a job arrival but it does not send any tasks to queue i . So we can write $Y_i(s)$ as

$$Y_i(s) = A_i(s) \cdot X_i(s),$$

where $A_i(s)$ equals to either 1 or 0, representing the number of task arrivals to queue i at time U_s , and $X_i(s)$ is a r.v. with distribution G and is independent of everything else, representing the service time. Then $\Phi_i^{(n)}(s+1), i = 1, 2, \dots, k^{(n)}$ are nondecreasing functions of the $\Phi_i^{(n)}(s)$'s, $A_i(s)$'s, $X_i(s)$'s and $-\Delta U(s)$ with $i = 1, 2, \dots, k^{(n)}$. We can see that each of the following four sets

¹We thank Prof. XYZ for suggesting this possible approach.

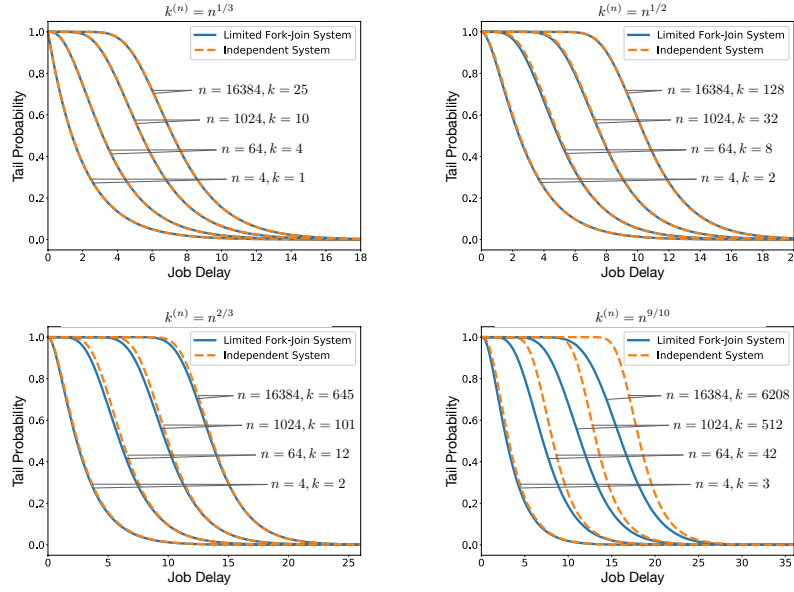


Figure 2: Tail distributions of job delays in the limited fork-join systems and the independence upper bounds.

of r.v.'s, $\{\Phi_1^{(n)}(s), \Phi_2^{(n)}(s), \dots, \Phi_{k^{(n)}}^{(n)}(s)\}$, $\{A_1(s), A_2(s), \dots, A_{k^{(n)}}(s)\}$, $\{X_1(s), X_2(s), \dots, X_{k^{(n)}}(s)\}$, and $\{-\Delta U(s)\}$, is independent of the union of others. So to show that $\Phi_i^{(n)}(s+1)$, $i = 1, 2, \dots, k^{(n)}$ are associated, it suffices to show that each of these sets is a set of associated r.v.'s.

(i) The $\Phi_i^{(n)}(s)$, $i = 1, 2, \dots, k^{(n)}$ are associated by assumption.

(ii) The $X_i(s)$, $i = 1, 2, \dots, k^{(n)}$ are associated since they are independent.

(iii) The r.v. $-\Delta U(s)$ is associated since a single r.v. is associated.

(iv) We now prove that $A_i(s)$, $i = 1, 2, \dots, k^{(n)}$ are associated.

We note that here they do not satisfy the lattice condition in the celebrated FKG inequality [12]. For conciseness of notation, let $\mathbf{A} = (A_1(s), A_2(s), \dots, A_{k^{(n)}}(s))$. To show association, it suffices to prove that for all *binary-valued*, (*entrywisely*) *nondecreasing* functions f and g [9],

$$\mathbb{E}[f(\mathbf{A})g(\mathbf{A})] \geq \mathbb{E}[f(\mathbf{A})]\mathbb{E}[g(\mathbf{A})]. \quad (31)$$

By construction, it is clear that $\mathbf{A} \in \{0, 1\}^{k^{(n)}}$. If either f or g always has constant value 0 or 1, then (31) trivially holds. So we can focus on the case that neither f nor g is a constant function. In this case, by the monotonicity of f and g , we have $f((0, \dots, 0)) = g((0, \dots, 0)) = 0$ and $f((1, \dots, 1)) = g((1, \dots, 1)) = 1$. Note that at each sample time U_s , the probability that there is a job arrival is $\Lambda^{(n)}/(\Lambda^{(n)} + \beta^{(n)})$. Then

$$\begin{aligned} \mathbb{E}[f(\mathbf{A})g(\mathbf{A})] &\geq \mathbb{P}(\mathbf{A} = (1, \dots, 1)) \cdot f((1, \dots, 1))g((1, \dots, 1)) \\ &= \frac{\Lambda^{(n)}}{\Lambda^{(n)} + \beta^{(n)}} \frac{1}{\binom{n}{k^{(n)}}}. \end{aligned}$$

Since $f(\mathbf{a}) \leq 1, g(\mathbf{a}) \leq 1$ for any $\mathbf{a} \in \{0, 1\}^{k^{(n)}}$,

$$\begin{aligned} \mathbb{E}[f(\mathbf{A})] &= \sum_{\substack{\mathbf{a} \in \{0, 1\}^{k^{(n)}} \\ \mathbf{a} \neq (0, \dots, 0)}} \mathbb{P}(\mathbf{A} = \mathbf{a})f(\mathbf{a}) \\ &\leq \mathbb{P}(\mathbf{A} \neq (0, \dots, 0)) \\ &= \frac{\Lambda^{(n)}}{\Lambda^{(n)} + \beta^{(n)}} p, \end{aligned}$$

where p is the probability that a job arrival does not sent tasks to queues $1, 2, \dots, k^{(n)}$, so p does not depend on $\beta^{(n)}$ and

$$p = \begin{cases} 1 & \text{if } k^{(n)} > n/2, \\ \frac{1}{\binom{n}{k^{(n)}}} \left(\binom{n}{k^{(n)}} - \binom{n}{k^{(n)}} \right) & \text{if } k^{(n)} \leq n/2. \end{cases}$$

Similarly,

$$\mathbb{E}[g(\mathbf{A})] \leq \frac{\Lambda^{(n)}}{\Lambda^{(n)} + \beta^{(n)}} p.$$

We choose any $\beta^{(n)}$ such that

$$\frac{\Lambda^{(n)}}{\Lambda^{(n)} + \beta^{(n)}} \frac{1}{\binom{n}{k^{(n)}}} \geq \left(\frac{\Lambda^{(n)}}{\Lambda^{(n)} + \beta^{(n)}} \right)^2 p^2,$$

i.e., any $\beta^{(n)}$ such that

$$\beta^{(n)} \geq \Lambda^{(n)} \left(\left(\frac{n}{k^{(n)}} \right) p^2 - 1 \right). \quad (32)$$

Then

$$\mathbb{E}[f(\mathbf{A})g(\mathbf{A})] \geq \mathbb{E}[f(\mathbf{A})]\mathbb{E}[g(\mathbf{A})],$$

which completes the induction, and thus completes the proof. \square

7 EVALUATION VIA SIMULATIONS

In this section we use simulation to explore the regimes of $k^{(n)}$ that are not covered by our theoretical analysis. Specifically, our theoretical analysis has established that when $k^{(n)} = o(n^{1/4})$, any $k^{(n)}$ queues are asymptotically independent and the job delay converges to the independence upper bound; when $k^{(n)} = \Theta(n)$, any number of multiple queues are bounded away from being independent. We therefore simulate the limited fork-join systems for the following four settings between $o(n^{1/4})$ and $\Theta(n)$: $k^{(n)} = n^{1/3}$, $k^{(n)} = n^{1/2}$, $k^{(n)} = n^{2/3}$ and $k^{(n)} = n^{9/10}$. We simulate the n -server system for $n = 4, 64, 1024$ and 16384 under each setting.

We compare the tail distribution (complementary cumulative distribution function) of the job delay in each limited fork-join system with the independence upper bound. Figure 2 shows the results for systems with exponentially distributed service times and load $\rho = 2/3$ on each individual queue. We see that for $k^{(n)} = n^{1/3}$, the independence upper bound is strikingly accurate. For $k^{(n)} = n^{1/2}$, the gap between the job delay and the independence upper bound seems to be diminishing when n is large enough. But for $k^{(n)} = n^{2/3}$, it is rather unclear if the job delay will converge to the independence upper bound or not. Finally, when $k^{(n)} = n^{9/10}$, the job delay evidently *diverges* from the independence upper bound. We have also simulated systems for different loads ($\rho = 1/3, 0.9$) and different service time distributions (deterministic, truncated Pareto, hyperexponential), and similar phenomena are observed.

8 CONCLUSIONS

We study the limited fork-join model where there are n servers in the system and each job consists of $k^{(n)} \leq n$ tasks that are sent to $k^{(n)}$ distinct servers chosen uniformly at random. A job is considered complete only when all its tasks complete processing. We characterize the delay of jobs both in an asymptotic regime where $n \rightarrow \infty$ and in the non-asymptotic regime for any n and any $k^{(n)} = k$.

For the asymptotic regime, we show that under the condition $k^{(n)} = o(n^{1/4})$, the workloads of any $k^{(n)}$ queues in the n -server system are asymptotically independent, and the delay of a job therefore converges to the maximum of independent task delays. For the non-asymptotic regime, we show that the steady-state workloads of any $k^{(n)}$ queues are associated, and therefore assuming independent task delays yields an upper bound on the job delay. Our results provide the first tight characterization of job delay in the limited fork-join model, and the upper bound is tighter than other existing upper bounds.

From a technical perspective, we make the following two contributions: (1) Our asymptotic results open up new regimes for asymptotic independence: $k^{(n)}$ queues are shown to be asymptotically independent, where $k^{(n)}$ is allowed to grow with n instead of being a constant, as was previously studied. (2) We develop new proof techniques to establish association in steady state. We believe that the results and techniques in this paper will shed light on related problems such as order statistics in coded data storage systems, job redundancy, load-balancing algorithms.

REFERENCES

- [1] François Baccelli. 1985. *Two parallel queues created by arrivals with two demands: the M/G/2 symmetrical case*. Technical Report RR-0426. INRIA.
- [2] François Baccelli and Armand M. Makowski. 1985. *Simple computable bounds for the fork-join queue*. Technical Report RR-0394. INRIA.
- [3] François Baccelli, Armand M. Makowski, and Adam Shwartz. 1989. The fork-join queue and related systems with synchronization constraints: stochastic ordering and computable bounds. *Adv. Appl. Probab.* 21 (1989), 629–660.
- [4] Maury Bramson, Yi Lu, and Balaji Prabhakar. 2012. Asymptotic independence of queues under randomized load balancing. *Queueing Syst.* 71, 3 (01 July 2012), 247–292.
- [5] Yanpei Chen, Sara Alspaugh, and Randy Katz. 2012. Interactive Analytical Processing in Big Data Systems: A Cross-industry Study of MapReduce Workloads. *Proc. VLDB Endow.* 5, 12 (Aug. 2012), 1802–1813.
- [6] J. Theodore Cox. 1984. An Alternate Proof of a Correlation Inequality of Harris. *Ann. Probab.* 12, 1 (02 1984), 272–273.
- [7] Anirban DasGupta. 2008. *Asymptotic theory of statistics and probability*. Springer Science & Business Media.
- [8] Jeffrey Dean and Sanjay Ghemawat. 2004. MapReduce: Simplified Data Processing on Large Clusters. In *Proc. USENIX Conf. Operating Systems Design and Implementation (OSDI)*. San Francisco, CA, 10–10.
- [9] J. D. Esary, F. Proschan, and D. W. Walkup. 1967. Association of Random Variables, with Applications. *Ann. Math. Statist.* 38, 5 (10 1967), 1466–1474.
- [10] Farshid Farhat, Diman Tootaghaj, Yuxiong He, Anand Sivasubramaniam, Mahmut Kandemir, and Chita Das. 2016. Stochastic Modeling and Optimization of Stragglers. *IEEE Trans. Cloud Comput.* (2016). to be published.
- [11] L. Flatto and S. Hahn. 1984. Two parallel queues created by arrivals with two demands I. *SIAM J. Appl. Math.* 44, 5 (1984), 1041–1053.
- [12] C. M. Fortuin, P. W. Kasteleyn, and J. Ginibre. 1971. Correlation inequalities on some partially ordered sets. *Comm. Math. Phys.* 22, 2 (1971), 89–103.
- [13] Kristen Gardner, Mor Harchol-Balter, and Alan Scheller-Wolf. 2016. A Better Model for Job Redundancy: Decoupling Server Slowdown and Job Size. In *IEEE Int. Symp. Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS)*. London, United Kingdom, 1–10.
- [14] Kristen Gardner, Mor Harchol-Balter, Alan Scheller-Wolf, and Benny Van Houdt. 2017. A Better Model for Job Redundancy: Decoupling Server Slowdown and Job Size. *IEEE/ACM Trans. Netw.* 25, 6 (Dec. 2017), 3353–3367.
- [15] Kristen Gardner, Mor Harchol-Balter, Alan Scheller-Wolf, Mark Velednitsky, and Samuel Zbarsky. 2017. Redundancy-d: The Power of d Choices for Redundancy. *Oper. Res.* 65, 4 (2017), 1078–1094.
- [16] Carl Graham. 2000. Chaoticity on path space for a queueing network with selection of the shortest queue among several. *J. Appl. Probab.* 37, 1 (March 2000), 198–211.
- [17] Mor Harchol-Balter. 2013. *Performance Modeling and Design of Computer Systems: Queueing Theory in Action* (1st ed.). Cambridge University Press, New York, NY.
- [18] T. E. Harris. 1977. A Correlation Inequality for Markov Processes in Partially Ordered State Spaces. *Ann. Probab.* 5, 3 (06 1977), 451–454.
- [19] Kumar Joag-Dev and Frank Proschan. 1983. Negative Association of Random Variables with Applications. *Ann. Statist.* 11, 1 (March 1983), 286–295.
- [20] Gauri Joshi, Yanpei Liu, and Emina Soljanin. 2012. Coding for fast content download. In *Proc. Ann. Allerton Conf. Communication, Control and Computing*. Monticello, IL, 326–333.
- [21] Gauri Joshi, Yanpei Liu, and Emina Soljanin. 2014. On the Delay-Storage Trade-Off in Content Download from Coded Distributed Storage Systems. *IEEE J. Sel. Areas Commun.* 32, 5 (May 2014), 989–997.
- [22] Sung-Seok Ko and Richard F. Serfozo. 2008. Sojourn times in G/M/1 fork-join networks. *Nav. Res. Log.* 55, 5 (2008), 432–443.
- [23] Anurag Kumar and Rajeev Shorey. 1993. Performance analysis and scheduling of stochastic fork-join jobs in a multicomputer system. *IEEE Trans. Parallel Distrib. Syst.* 4, 10 (Oct. 1993), 1147–1164.
- [24] Kangwook Lee, Nihar B. Shah, Longbo Huang, and Kannan Ramchandran. 2017. The MDS Queue: Analysing the Latency Performance of Erasure Codes. *IEEE Trans. Inf. Theory* 63, 5 (May 2017), 2822–2842.
- [25] Bin Li, Aditya Ramamoorthy, and R. Srikant. 2016. Mean-field-analysis of coding versus replication in cloud storage systems. In *Proc. IEEE Int. Conf. Computer Communications (INFOCOM)*. San Francisco, CA, 1–9.
- [26] Thomas M. Liggett. 2005. *Interacting Particle Systems*. Springer-Verlag Berlin Heidelberg.
- [27] Minghong Lin, Li Zhang, Adam Wierman, and Jian Tan. 2013. Joint Optimization of Overlapping Phases in MapReduce. *Perform. Eval.* 70, 10 (Oct. 2013), 720–735.
- [28] Hongyuan Lu and Guodong Pang. 2017. Heavy-traffic limits for an infinite-server fork-join queueing system with dependent and disruptive services. *Queueing Syst.* 85, 1 (Feb. 2017), 67–115.
- [29] Yi Lu, Qiaomin Xie, Gabriel Kliot, Alan Geller, James R. Larus, and Albert Greenberg. 2011. Join-Idle-Queue: A Novel Load Balancing Algorithm for Dynamically Scalable Web Services. *Perform. Eval.* 68, 11 (Nov. 2011), 1056–1071.

- [30] John C.S. Lui, Richard R. Muntz, and Don Towsley. 1998. Computing performance bounds of fork-join parallel programs under a multiprocessing environment. *IEEE Trans. Parallel Distrib. Syst.* 9, 3 (March 1998), 295–311.
- [31] Benjamin Melamed and Ward Whitt. 1990. On Arrivals That See Time Averages. *Oper. Res.* 38, 1 (1990), 156–172.
- [32] Sean P. Meyn and R. L. Tweedie. 1992. Stability of Markovian Processes I: Criteria for Discrete-Time Chains. *Adv. Appl. Probab.* 24, 3 (1992), 542–574.
- [33] Sean P. Meyn and R. L. Tweedie. 1993. Stability of Markovian Processes III: Foster-Lyapunov Criteria for Continuous-Time Processes. *Adv. Appl. Probab.* 25, 3 (1993), 518–548.
- [34] Benjamin Moseley, Anirban Dasgupta, Ravi Kumar, and Tamás Sárlos. 2011. On Scheduling in Map-Reduce and Flow-shops. In *Proc. Ann. ACM Symp. Parallelism in Algorithms and Architectures (SPAA)*. San Jose, CA, 289–298.
- [35] Randolph Nelson and Asser N. Tantawi. 1988. Approximate Analysis of Fork/Join Synchronization in Parallel Queues. *IEEE Trans. Comput.* 37, 6 (June 1988), 739–743.
- [36] Randolph Nelson, Don Towsley, and Asser N. Tantawi. 1988. Performance Analysis of Parallel Processing Systems. *IEEE Trans. Softw. Eng.* 14, 4 (April 1988), 532–540.
- [37] Amr Rizk, Felix Poloczek, and Florin Ciucu. 2016. Stochastic bounds in Fork-Join queueing systems under full and partial mapping. *Queueing Syst.* 83, 3 (Aug. 2016), 261–291.
- [38] Halsey L. Royden and Patrick M. Fitzpatrick. 2010. *Real Analysis* (4th ed.). Pearson.
- [39] Nihar B. Shah, Kangwook Lee, and Kannan Ramchandran. 2013. When do redundant requests reduce latency?. In *Proc. Ann. Allerton Conf. Communication, Control and Computing*. Monticello, IL, 731–738.
- [40] Virag Shah, Anne Bouillard, and François Baccelli. 2017. Delay comparison of delivery and coding policies in data clusters. In *Proc. Ann. Allerton Conf. Communication, Control and Computing*. Monticello, IL, 397–404.
- [41] Alexander L. Stolyar. 2015. Pull-based load distribution in large-scale heterogeneous service systems. *Queueing Syst.* 80, 4 (Aug. 2015), 341–361.
- [42] Yin Sun, C. Emre Koksal, and Ness B. Shroff. 2017. *Near Delay-Optimal Scheduling of Batch Jobs in Multi-Server Systems*. Technical Report. The Ohio State University.
- [43] Jian Tan, Xiaoqiao Meng, and Li Zhang. 2012. Delay Tails in MapReduce Scheduling. In *Proc. ACM SIGMETRICS/PERFORMANCE Jt. Int. Conf. Measurement and Modeling of Computer Systems*. London, United Kingdom, 5–16.
- [44] Alexander Thomasian. 2014. Analysis of Fork/Join and Related Queueing Systems. *ACM Comput. Surv.* 47, 2, Article 17 (Aug. 2014), 71 pages.
- [45] E. Varki. 2001. Response time analysis of parallel computer and storage systems. *IEEE Trans. Parallel Distrib. Syst.* 12, 11 (Nov. 2001), 1146–1161.
- [46] Emanuel Vianna, Giovanni Comarella, Tatiana Pontes, Jussara Almeida, Vigilio Almeida, Kevin Wilkinson, Harumi Kuno, and Umeshwar Dayal. 2013. Analytical Performance Models for MapReduce Workloads. *Int. J. Parallel Prog.* 41, 4 (Aug. 2013), 495–525.
- [47] Ashish Vulimiri, Philip Brighten Godfrey, Radhika Mittal, Justine Sherry, Sylvia Ratnasamy, and Scott Shenker. 2013. Low Latency via Redundancy. In *Proc. Conf. Emerging Networking Experiments and Technologies (CoNEXT)*. Santa Barbara, CA, 283–294.
- [48] Ashish Vulimiri, Oliver Michel, P. Brighten Godfrey, and Scott Shenker. 2012. More is Less: Reducing Latency via Redundancy. In *Proc. ACM Workshop Hot Topics in Networks (HotNets)*. Redmond, WA, 13–18.
- [49] N. D. Vvedenskaya, R. L. Dobrushin, and F. I. Karpelevich. 1996. Queueing System with Selection of the Shortest of Two Queues: An Asymptotic Approach. *Probl. Inf. Transm.* 32, 1 (1996), 15–27.
- [50] Weina Wang, Kai Zhu, Lei Ying, Jian Tan, and Li Zhang. 2016. MapTask Scheduling in MapReduce With Data Locality: Throughput and Heavy-Traffic Optimality. *IEEE/ACM Trans. Netw.* 24 (Feb. 2016), 190–203.
- [51] Cathy H. Xia, Zhen Liu, Don Towsley, and Marc Lelarge. 2007. Scalability of Fork/Join Queueing Networks with Blocking. In *Proc. ACM SIGMETRICS Int. Conf. Measurement and Modeling of Computer Systems*. San Diego, CA, 133–144.
- [52] Yu Xiang, Tian Lan, Vaneet Aggarwal, and Yih-Farn R. Chen. 2016. Joint Latency and Cost Optimization for Erasure-Coded Data Center Storage. *IEEE/ACM Trans. Netw.* 24, 4 (Aug. 2016), 2443–2457.
- [53] Qiaomin Xie, Xiaobo Dong, Yi Lu, and R. Srikant. 2015. Power of d Choices for Large-Scale Bin Packing: A Loss Model. In *Proc. ACM SIGMETRICS Int. Conf. Measurement and Modeling of Computer Systems*. Portland, OR, 321–334.
- [54] Qiaomin Xie and Yi Lu. 2015. Priority algorithm for near-data scheduling: Throughput and heavy-traffic optimality. In *Proc. IEEE Int. Conf. Computer Communications (INFOCOM)*. Hong Kong, China, 963–972.
- [55] Lei Ying, R. Srikant, and Xiaohan Kang. 2015. The power of slightly more than one sample in randomized load balancing. In *Proc. IEEE Int. Conf. Computer Communications (INFOCOM)*. Kowloon, Hong Kong, 1131–1139.
- [56] Yousi Zheng, Ness B. Shroff, and Prasan Sinha. 2013. A new analytical technique for designing provably efficient MapReduce schedulers. In *Proc. IEEE Int. Conf. Computer Communications (INFOCOM)*. Turin, Italy, 1600–1608.

A NON-INDEPENDENCE RESULT FOR

$$k^{(n)} = \Theta(n)$$

THEOREM A.1. *Consider an n -server system in the limited fork-join model with $k^{(n)} = \Theta(n)$, job arrival rate $\Lambda^{(n)} = n\lambda/k^{(n)}$, and exponentially distributed service times with rate μ . Let $\pi^{(n,2)}$ denote the joint distribution of the steady-state queue lengths for any two queues in the n -server system. Let $\hat{\pi}^{(2)}$ denote the joint distribution of the steady-state queue lengths of two independent $M/M/1$ queues, each with load ρ . Then there exists an $\epsilon > 0$ and $n_0 > 0$, such that for any $n > n_0$, $d_{TV}(\pi^{(n,2)}, \hat{\pi}^{(2)}) > \epsilon$.*

PROOF. We assume that $k^{(n)} = pn$ for a constant p with $0 < p \leq 1$. Then the job arrival rate is given by $\Lambda^{(n)} = \lambda/p$, which is a constant. So we rewrite $\Lambda^{(n)}$ as Λ for conciseness.

Let $\epsilon = \frac{p\lambda(1-p)^2}{2(11\Lambda+8\mu)}$. We will specify n_0 later. Suppose by contradiction that $d_{TV}(\pi^{(n,2)}, \hat{\pi}^{(2)}) \leq \epsilon$ for all $n > n_0$. We will show that this assumption contradicts with the balance equations of the first two queues in the limited fork-join system with n servers.

We first write out the balance equations for the Markov chain formed by the queue lengths of the first two queues. Consider a job arrival to this n -server system. Let $p_0^{(n)}$ be the probability that no task arrives to the first two queues, and $p_1^{(n)}$ be the probability that exactly one task arrives to the first two queues. Let $p_2^{(n)} = 1 - p_0^{(n)} - p_1^{(n)}$ be the probability that two tasks arrive to the first two queues. We can compute these probabilities as follows:

$$\begin{aligned} p_0^{(n)} &= \binom{n-2}{k} / \binom{n}{k} \rightarrow p_0 := (1-p)^2 \quad \text{as } n \rightarrow \infty, \\ p_1^{(n)} &= \frac{2\binom{n-2}{k-1}}{\binom{n}{k}} \rightarrow p_1 := 2p(1-p) \quad \text{as } n \rightarrow \infty, \\ p_2^{(n)} &= \frac{\binom{n-2}{k-2}}{\binom{n}{k}} \rightarrow p_2 := p^2 \quad \text{as } n \rightarrow \infty. \end{aligned}$$

Recall that the joint distribution of the steady-state queue lengths of the first two queues is $\pi^{(n,2)}$. Then the balance equation of the first two queues for the state $(1, 1)$ can be written as

$$\begin{aligned} 0 &= \pi^{(n,2)}(1, 1) \cdot (p_1^{(n)}\Lambda + p_2^{(n)}\Lambda + 2\mu) \\ &\quad - \left(\frac{1}{2}\pi^{(n,2)}(0, 1)p_1^{(n)}\Lambda + \frac{1}{2}\pi^{(n,2)}(1, 0)p_1^{(n)}\Lambda \right. \\ &\quad \left. + \pi^{(n,2)}(0, 0)p_2^{(n)}\Lambda + \pi^{(n,2)}(1, 2)\mu + \pi^{(n,2)}(2, 1)\mu \right). \end{aligned} \quad (33)$$

Let the right-hand-side of (33) be denoted by $\mathcal{R}(\pi^{(n,2)})$. Let

$$\begin{aligned} a_1 &= (p_1^{(n)} - p_1)\Lambda \left(\pi^{(n,2)}(1, 1) - \frac{1}{2}\pi^{(n,2)}(0, 1) - \frac{1}{2}\pi^{(n,2)}(1, 0) \right) \\ &\quad + (p_2^{(n)} - p_2)\Lambda \left(\pi^{(n,2)}(1, 1) - \frac{1}{2}\pi^{(n,2)}(0, 1) \right. \\ &\quad \left. - \frac{1}{2}\pi^{(n,2)}(1, 0) - \pi^{(n,2)}(0, 0) \right), \\ a_2 &= (\pi^{(n,2)}(1, 1) - \hat{\pi}^{(2)}(1, 1))(p_1\Lambda + p_2\Lambda + 2\mu) \\ &\quad - \frac{1}{2}(\pi^{(n,2)}(0, 1) - \hat{\pi}^{(2)}(0, 1) + \pi^{(n,2)}(1, 0) - \hat{\pi}^{(2)}(1, 0))p_1\Lambda \end{aligned}$$

$$\begin{aligned}
 & - (\pi^{(n,2)}(0,0) - \hat{\pi}^{(2)}(0,0))p_2\Lambda \\
 & - (\pi^{(n,2)}(1,2) - \hat{\pi}^{(2)}(1,2) + \pi^{(n,2)}(2,1) - \hat{\pi}^{(2)}(2,1))\mu.
 \end{aligned}$$

Then since $\hat{\pi}^{(2)}(q_1, q_2) = (1 - \rho)^2 \rho^{q_1 + q_2}$ for any $(q_1, q_2) \in \mathbb{Z}_+^2$,

$$\begin{aligned}
 \mathcal{R}(\pi^{(n,2)}) &= a_1 + a_2 + \hat{\pi}^{(2)}(1,1) \cdot (p_1\Lambda + p_2\Lambda + 2\mu) \\
 & - \left(\frac{1}{2} \hat{\pi}^{(2)}(0,1)p_1\Lambda + \frac{1}{2} \hat{\pi}^{(2)}(1,0)p_1\Lambda \right. \\
 & \quad \left. + \hat{\pi}^{(2)}(0,0)p_2\Lambda + \hat{\pi}^{(2)}(1,2)\mu + \hat{\pi}^{(2)}(2,1)\mu \right) \\
 &= a_1 + a_2 - p\lambda(1 - \rho)^4. \tag{34}
 \end{aligned}$$

We choose n_0 such that for any $n > n_0$, $|p_1^{(n)} - p_1| \leq \epsilon$ and $|p_2^{(n)} - p_2| \leq \epsilon$. Then it is not hard to see that $|a_1| \leq 3\Lambda\epsilon$. By the assumption that $d_{TV}(\pi^{(n,2)}, \hat{\pi}^{(2)}) \leq \epsilon$, we have that $|a_2| \leq 8(\Lambda + \mu)\epsilon$. By the choice of ϵ , $|a_1 + a_2| \leq (11\Lambda + 8\mu)\epsilon = \frac{1}{2}p\lambda(1 - \rho)^2$. Therefore, $\mathcal{R}(\pi^{(n,2)}) < 0$ by (34), which contradicts with the balance equation (33). This completes the proof of Theorem A.1. \square

B DEFINITION AND SOME PROPERTIES OF ASSOCIATION

Definition B.1 (Association [9]). We say random variables X_1, X_2, \dots, X_m are associated if for all (entrywisely) nondecreasing functions f and g ,

$$\begin{aligned}
 & \mathbb{E}[f(X_1, X_2, \dots, X_m)g(X_1, X_2, \dots, X_m)] \\
 & \geq \mathbb{E}[f(X_1, X_2, \dots, X_m)]\mathbb{E}[g(X_1, X_2, \dots, X_m)]. \tag{35}
 \end{aligned}$$

LEMMA B.2 ([9]). *Associated random variables have the following properties:*

- (P1) *Nondecreasing functions of associated random variables are associated.*
- (P2) *If two sets of associated random variables are independent of one another, then their union is a set of associated random variables.*
- (P3) *If a sequence of random vectors $X(u) \Rightarrow X$ as $u \rightarrow \infty$ and for each u , the entries of $X(u)$ are associated, then the entries of X are associated.*