Load balancing policies with server-side cancellation of replicas

Rooji Jinan∗ Ajay Badita† Tejas Bodas‡ Parimal Parag†

Abstract

Popular dispatching policies such as the join shortest queue (JSQ), join smallest work (JSW) and their power of two variants are used in load balancing systems where the instantaneous queue length or workload information at all queues or a subset of them can be queried. In situations where the dispatcher has an associated memory, one can minimize this query overhead by maintaining a list of idle servers to which jobs can be dispatched. Recent alternative approaches that do not require querying such information include the cancel on start and cancel on complete based replication policies. The downside of such policies however is that the servers must communicate the start or completion of each service to the dispatcher and must allow cancellation of redundant copies. In this work, we consider a load balancing environment where the dispatcher cannot query load information, does not have a memory, and cannot cancel any replica that it may have created. In such a rigid environment, we allow the dispatcher to possibly append a server side cancellation criteria to each job or its replica. A job or a replica is served only if it satisfies the predefined criteria at the time of service. We focus on a criteria that is based on the waiting time experienced by a job or its replica and analyze several variants of this policy based on the assumption of asymptotic independence of queues. The proposed policies are novel and perform remarkably well in spite of the rigid operating constraints.

Index Terms

Load balancing, Redundant computing, Distributed discard policy

I. INTRODUCTION

Load balancing policies play a vital role in latency reduction in distributed systems such as large data centers and cloud computing. A typical load balancing system comprises of a large number of homogeneous servers and a dispatcher that routes arriving jobs to the queue of these servers. When the instantaneous queue length of the different servers is known, an obvious approach would be to use the join-shortest-queue (JSQ) policy [1]. If instead of the queue length, the workload i.e., the pending amount of work at each server is known, the optimal policy would be the join-smallest-work queue (JSW). Unfortunately, in most practical systems, the number of servers is large and therefore obtaining the instantaneous queue lengths from all servers is difficult.

A popular remedy for this is to consider the power of $d$ choice variant of JSQ and JSW. In a JSQ(d) policy, the dispatcher samples $d$ servers uniformly at random and queries their queue lengths. The job is then routed to a sampled server with the least number of waiting jobs. Implementing such a policy requires only $2d$ messages per job and was shown to have very good performance characteristics [2], [3]. The equivalent workload based policy JSW(d) also has a $2d$ query overhead per job and was analyzed recently [4], [5]. For many systems, a $2d$ query exchange is also a considerable overhead, especially when $d$ is large or when the timescale for message exchange is comparable to the actual service requirement of a job. Recent efforts have therefore been directed towards bringing down the communication overhead using smart feedback techniques [6], [7]. [6] considers a hyper-scalable dispatching scheme where the dispatcher maintains queue length estimates for the different queues and sends an arriving job to the server with the least estimated queue length. Each server occasionally updates the dispatcher about its true queue length and this enables the dispatcher to synchronize its estimates with reality. [7] introduce the join-open-queue scheme where servers send busy alerts to the dispatcher at predetermined times.

Author∗ is with the Robert Bosch centre for cyber-physical systems, and the authors † are with the department of electrical communication engineering, all at Indian Institute of Science, Bangalore, Karnataka 560012, India. Email: {roojijinan, ajaybadita, parimal}@iisc.ac.in.

Author‡ is with the department of electrical engineering, Indian Institute of Technology, Dharwad, Karnataka 580011, India. Email: tejaspbodas@iitdh.ac.in.
When a server is idle, it does not send the alert and thus the dispatcher can infer idle servers without considerable message exchanges.

Join idle queue policy is another load balancing approach which has a low message overhead and a very good performance characteristics [8]. In this policy, idle queues willingly inform the dispatcher about their idleness and the dispatcher notes this in an associated memory. An arriving jobs is sent to an idle queue selected randomly from the list if it is non-empty and therefore this policy has an overhead of at most 1 message per job. Some recent load balancing policies that make use of memory in their dispatching decisions appear in [9], [10].

Redundancy based load balancing policies on the other hand do not require querying instantaneous queue length or workload information. Two popular variants of redundancy-based load balancing are cancel on start (c.o.s.) [11] and cancel on complete (c.o.c) [12]. In these policies, independent replicas of an arriving job are sent to $d$ randomly chosen servers. In c.o.s. (resp. c.o.c), when one of the copy starts receiving service (resp. receives complete service), the $d-1$ replicas are canceled. Such policies also have superior delay performance and are quite amenable to analysis. A detailed product form analysis characterizing the delay performance for both variants is presented in [13]. A major implementation problem with replication based policies is the synchronized cancellation of the redundant replicas. The sophistication required for implementing such an approach in fact may even be nontrivial. Further, depending on the operating scenario, instantaneous cancellation may not always be feasible, thereby adding an overhead on the system in terms of wasted service making the system inefficient [14], [15].

The load balancing policies considered above either involve (a) communication of messages, or (b) require a memory, or (c) require replication with cancellation. Such policies therefore always have an element of feedback from the server to the dispatcher. In this work, we restrict to a working environment where there is no feedback of queue length or workload information from the servers. This renders any memory that the dispatcher may have to be of no use. While the dispatcher can possibly replicate jobs to different servers, the lack of communication prohibits cancellation of redundant copies. In such a rigid load balancing environment, we hope to seek policies that outperform the random routing policy which is an obvious policy in a no feedback regime. Towards this, we offer the dispatcher the ability to append a server-side cancellation criteria to each replica. Before picking any replica for service, each server will check if the appended criteria is satisfied or not. If the criteria is met, then the replica is served or else it is dropped. We consider a criteria that depends on the waiting time of the replica in a queue. For example, one criteria that we consider is to serve the replica only if it has waited in the queue for less than $T$ units of time. Such a criteria is easy for the server to validate, and can be achieved by logging the arrival time information of each job/replica. The key essence of our approach is to exploit possible gains from replication of jobs, but at the same time prevent overloading the system due to extra replicas by preemptively performing server-side cancellation of potentially wasteful replicas.

In a more formal description of our policy, we consider a load balancing system with $N$ queues and where jobs arrive according to a Poisson process with rate $\lambda N$. Jobs have a service requirement that is characterized by a general service time distribution $G(\cdot)$. Servers are identical with service rate $\mu$ and for each arriving job referred to as the primary replica, the dispatcher creates $d-1$ secondary replicas with probability $p$. The servers where the replicas are sent are chosen randomly. Associated with the primary and secondary replicas are discard thresholds $T_1$ and $T_2$. A replica is discarded by the server if the waiting time experienced by the replica is more than its discard threshold. We label our load balancing policy by $\pi(p, T_1, T_2)$ and provide a complete performance characterization.

We observe that when $T_1$ and $T_2$ are both finite, arriving jobs could potentially be lost without service. Keeping this in mind, the two key performance metric that we consider are the conditional mean response time of jobs admitted into the system and the loss probability of an arriving job. To analyze our policy, we make use of the cavity process method of [16], [17] along with a conjecture on the asymptotic independence of the workloads at the different queues as $N \to \infty$. Under this mean field regime, we obtain the moment generating function (MGF) for limiting workload for an arbitrary queue under the policy $\pi(p, T_1, T_2)$. This function can be inverted to obtain the limiting workload distribution, when the service time distribution is exponential. In this case, we derive closed form expressions for marginal workload distribution and conditional mean response time of admitted jobs in terms of the system parameters.

Proposed load balancing policy $\pi(p, T_1, T_2)$ is closely related to replication based policies without cancellation. A setting where cancellation of replicas is expensive or infeasible is considered in [18]. Their setting corresponds to the special case of our policy where the secondary replicas are always selected with probability $p = 1$, and the replicas are always served with thresholds $T_1 = T_2 = \infty$. The idea of replication without cancellation has also been
used in multipath routing in networks [19], [20]. Typically, flows are replicated along multiple paths to extract the diversity in congestion levels across different paths. Proposed probabilistic redundancy policy $\pi(p, T_1, T_2)$ is more generally applicable for such multipath routing in network as well, provided the intermediate nodes/routers have the ability to drop certain flows based on appropriate criteria. We do not proceed with this idea any further in this article.

A. Contribution

We have listed our key contributions below.

1) We propose a load balancing policy with probabilistic redundancy, where secondary replicas are added probabilistically. The policy is distributed since the dispatcher needs no feedback from the servers, and the replicas are discarded at the server if the waiting time exceeds a threshold.

2) Assuming asymptotic independence of workloads in the number of servers, we find an expression for conditional mean response time for a job, given the job is admitted in the system.

3) We obtain closed-form expression for loss probability, limiting marginal workload distribution and conditional mean response time of admitted jobs, when the service time distribution is exponential.

4) We empirically verify that the independence assumption on the marginal workload distribution is a good approximation even for a finite number of servers.

5) We provide design guidelines on choice of number of replicas $d$, and the corresponding cancellation thresholds $T_1, T_2$ for the proposed policy.

B. Organization

We introduce the system model and notations in Section II. This is followed by a discussion on the cavity process method and its application to our problem along with the conjecture on the asymptotic independence of the workloads at different queues. In Section III, we compute the performance metrics for the proposed probabilistic redundancy policy $\pi(p, T_1, T_2)$ with server side cancellation for a general service time distribution, in terms of the limiting marginal workload distribution. In Section IV, we find the closed-form expression for marginal workload distribution when the service time distribution is exponential. We also compute the conditional mean of response time for admitted jobs, for some special cases of $\pi(p, T_1, T_2)$ policy. We conclude with a summary of our work and future directions in Section V.

II. System Model and Preliminaries

We consider a load balancing system with $N$ servers, where jobs arrive according to a Poisson process of rate $\lambda N$. There is a dispatcher associated with this system whose objective is to minimize the response time experienced by each job by suitably balancing the workload across different servers. Owing to the popularity of redundancy based load balancing policies, we assume that the dispatcher has the ability to replicate an arriving job across multiple servers.

Throughout this article, we denote the set of first $n$ consecutive positive integers as $[n] \triangleq \{1, \ldots, n\}$, the set of non-negative integers as $\mathbb{Z}_+$, the set of positive integers as $\mathbb{N}$, the set of non-negative reals as $\mathbb{R}_+$ and the set of positive reals as $\mathbb{R}^+$. We also use the notation $x \land y \triangleq \min\{x, y\}$.

A. Replication

We denote the service time for $n$th arriving job at $i$th server by $X_{n,i} \in \mathbb{R}_+$. We assume that the job service time sequence $(X_{n,i} \in \mathbb{R}_+ : n \in \mathbb{N}, i \in [N])$ is random and independent and identically distributed (i.i.d.) with the common distribution $G : \mathbb{R}_+ \rightarrow [0, 1]$ and the common mean $\frac{1}{\mu}$. That is, we assume that the service time for each replica of the job is i.i.d. according to the same distribution $G$. Even if we consider all servers to be identical in terms of configuration and compute power, there could be some uncertainties in the time taken to service a job at any server due to other background processes [21]. The randomness assumption accommodates these uncertainties. Further, we also assume the service times to be exponentially distributed. Recent studies suggests that the service times in distributed computing systems can be modelled to have two components; a constant startup delay and a random memoryless component [22]–[24]. Although, it is the shifted exponential model that
best fits this profile, whenever the startup time is negligible the service time distribution can be approximated by an exponential distribution. This along with analytical tractability motivates us to assume that the service times follow i.i.d. exponential distribution with rate $\mu$. Also, we denote the tail distribution of the service time or the complementary service time distribution by $G \triangleq 1 - G$. When we focus on a single queue $i$, we will drop the subscript $i$ for brevity.

**B. Threshold based cancellation**

We assume that the dispatcher has limited functionality and that it cannot cancel redundant copies when one of the replica has received (or starts receiving) service. Instead, we assume that the dispatcher can append *discard instruction* along with each replica. Before selecting a job/replica for service, each server will read the *discard instruction* and possibly discard the replica based on the instruction. We call this as a redundancy based approach with server side cancellation of replicas. For ease of exposition, we assume that the *instruction* is almost identical for all copies in the system and hence the overhead of implementing this approach is minimal. In this article, we restrict to *instructions* that are characterized by a threshold $T \in [0, \infty)$. To elaborate, we assume that the server serves a replica if it is chosen for service within $T$ units of its arrival or else discards the replica. We call $T$ as the *discard threshold* for brevity.

1) **Primary replica and discard threshold**: We consider the following dispatching policy based on the above idea of a *discard threshold*. When a job arrives, the dispatcher samples a single primary server uniformly at random and sends a primary replica of the job to the server along with the primary discard threshold $T_1$.

2) **Secondary replicas and discard thresholds**: For each job arrival, the dispatcher decides to create secondary replicas independently with replication probability $p$. When dispatcher decides to create secondary replicas, then it samples $d - 1$ other servers uniformly at random and sends i.i.d. replicas of the same job to the sampled $d - 1$ servers after appending each replica with a secondary discard threshold of $T_2$ where $T_2 \ll T_1$.

Since our policy is parametrized by probability of secondary replica $p$, primary discard threshold $T_1$, and secondary discard threshold $T_2$, we shall henceforth denote it by $\pi(p, T_1, T_2)$ for simplicity. The replication probability $p$ controls the redundant load on the system. For example, we do not add any secondary replicas when $p = 0$, and we always add secondary replicas when $p = 1$. We choose the secondary discard threshold to be smaller than the primary discard threshold, since we expect the secondary replicas to be helpful only if the primary is delayed.

Following are some special cases of our *discard threshold* based probabilistic $d$-replication-cancellation policy that we analyze in this article.

1) Selective replication with identical thresholds ($\pi(p, T, T)$): In this policy, each job is replicated $d$ times and assigned to $d$ servers chosen at random with probability $p$. Each job replica will have a threshold of $T$ time units which can possibly result in loss of jobs. When $T = \infty$ and $p = 1$ the policy reduces to that of a simple replication policy without cancellation.

2) Selective replication with no loss ($\pi(p, \infty, T_2)$): This is a selective replication policy, where primary is always served, and the $d - 1$ replicas are created only with probability $p$ reducing the overhead due to large number of replicas. Since $T_1 = \infty$, each primary replica of the job is definitely served. The advantage of this policy is that no jobs are lost.

3) Selective replication on idle servers ($\pi(p, \infty, 0)$): This is special case of selective replication policy with minimal redundancy addition, since secondary replicas only join idle queues.

**C. Server**

We assume that each server has an infinite sized buffer where arriving job replicas can wait for service, on a first come first served (FCFS) basis. We let the random variable $W_{n,i}$ with distribution function $F : \mathbb{R}_+ \rightarrow [0, 1]$, denote the waiting time of the $n$th job at server $i \in [N]$ experienced by an arriving job replica. Due to FCFS service, the random variable $W_{n,i}$ is also the effective workload present at server $i$ that must be served before this replica can receive service. An arriving replica is executed at a server $i$ if its discard threshold $T$ is larger than the observed workload $W_{n,i}$, and is discarded otherwise.

Each arriving job in the system results in a potential arrival at maximum $d$ randomly sampled queues. Depending on the $T$ and $W_{n,i}$, the job either receives service or is discarded. If a replica is served, then it results in an actual arrival at the corresponding server queue.
D. Performance metrics

We consider the following two performance metrics, the mean response time and the loss probability. Since our dispatcher replicates each arriving job to at most $d$ servers, the response time of an arriving job is the minimum of the sojourn times experienced by its different replicas. When both the thresholds $T_1$ and $T_2$ are finite, each replica can be discarded without service, leading to a loss. For lost jobs, the response time metric is meaningless. Hence, we obtain the mean response time of a job, conditioned on the event that it is not discarded. A job is serviced when at least one of its replicas is not discarded at the servers sampled by the dispatcher, i.e. when workload at one of these servers is smaller than or equal to the corresponding discard threshold.

Definition 1. Let $I_1$ be the singleton set of servers where primary replica is dispatched. Let $\zeta$ be the indicator that secondary replicas are created. Let $I_2$ be the candidate set of servers to which the secondary replicas are dispatched. For any server $j$, we define the indicators $\gamma_{1,j} \triangleq \mathbb{1}_{\{j \in I_1\}}$ and $\gamma_{2,j} \triangleq \mathbb{1}_{\{j \in I_2\}}$ which indicates that the queue is selected as a primary or secondary server respectively.

Definition 2. If the replica is dispatched to a server $j \in I_1 \cup I_2$ with workload $W_j$, then we define the indicator that the job is not discarded at this server $j$ as

$$\xi_j \triangleq \mathbb{1}_{\{W_j \leq T_1\}} \gamma_{1,j} + \mathbb{1}_{\{W_j \leq T_2\}} \gamma_{2,j}. \quad (1)$$

We denote the set of servers, where the job replicas are not discarded by $I \triangleq \{j \in I_1 \cup I_2 : \xi_j = 1\}$. A job is not discarded when $I \neq \emptyset$, and we denote this by indicator $\xi \triangleq \mathbb{1}_{\{I \neq \emptyset\}}$. We can write this in terms of the indicator $\zeta$, the set of servers $I_1, I_2$, the indicators $\xi_j$ and $\xi_j \triangleq 1 - \xi_j$ for all $j \in I_1 \cup I_2$,

$$\xi \triangleq 1 - \prod_{j \in I_1} \xi_j \prod_{j \in I_2} (\xi_j \zeta_j + \zeta_j). \quad (2)$$

Definition 3. The loss probability for policy $\pi(p, T_1, T_2)$ is denoted by $P_L \triangleq E[\xi]$.

Definition 4. We denote the response time of any job by $R' \in \mathbb{R}_+ \cup \infty$ and the response time of an undiscarded job by a random variable $R = \xi R' \in \mathbb{R}_+$ following the distribution function $H : \mathbb{R}_+ \rightarrow [0, 1]$, such that the tail distribution $H : \mathbb{R}_+ \rightarrow [0, 1]$ is defined as $H(x) \triangleq 1 - H(x)$ for all $x \in \mathbb{R}_+$. We study the conditional mean response time for a job given that it is not discarded. Specifically, we define the conditional mean response time as

$$\tau \triangleq \frac{\mathbb{E}[R]}{\mathbb{E}[\xi]} = \frac{\int_{x \in \mathbb{R}_+} H(x) dx}{1 - P_L}. \quad (3)$$

In this article, we analyze the performance of the $\pi(p, T_1, T_2)$ load balancing policy for different special cases mentioned in Section II-B, based on the two performance metrics of conditional mean response time and loss probability. Computing the limiting marginal workload distribution at a single queue is straightforward. However, a job response time is the minimum of response time for all possible job replicas, and computation of the conditional mean requires the knowledge of the joint distribution of workloads at all queues with a job replica. As such, we assume that the workloads in different queues are independent of each other and the probability of creating secondary replicas do not depend on the existing workloads, when the number of servers $N$ grows large while keeping the number of replicas $d$ fixed. The cavity process method and the conjecture on the asymptotic independence of the queues is stated in the next subsection.

E. Cavity process method

We first explain the principle of a cavity process method as applied to popular load balancing policies such as least loaded (LL(d)) or join shortest queue (JSQ(d)) and then specialize the discussion to our policy $\pi(p, T_1, T_2)$. See [4], [15]–[17] for more details about this approach. In the LL(d) (resp. JSQ(d)) system with $N$ queues and Poisson arrival rate of $\lambda N$, $d$ queues are sampled for each arriving job. The arriving job is executed on the sampled server with the smallest workload (resp. queue length). Let $\{H(t), t \geq 0\}$ denote the collection of probability measures on $\mathbb{R}_+$. This is called as the environment process. We tag one of the queue in the $N$ queue system as the cavity queue and denote the cavity process by $X^{R(t)}$ which represents the workload process (resp. the queue length process) at the cavity queue under policy LL(d) (resp. JSQ(d)). The potential arrival rate of jobs to the cavity queue under
both policies is $\lambda d$. For a potential arrival at the cavity queue at time $t$, we compare $d - 1$ random variables with law $\mathcal{H}(t)$ with $X^{\mathcal{H}(t-)}$. The potential arrival becomes an actual arrival to the cavity queue if the value of $X^{\mathcal{H}(t-)}$ is lower than the values taken by the $d - 1$ other variables, else the job is discarded. When the job is accepted, we have $X^{\mathcal{H}(t)} = X^{\mathcal{H}(t-)} + 1$ for the JSQ($d$) policy and $X^{\mathcal{H}(t)} = X^{\mathcal{H}(t-)} + x$ for the LL($d$) policy where $x$ is the service requirement of the arriving job. When the job is discarded, we have $X^{\mathcal{H}(t)} = X^{\mathcal{H}(t-)}$. For the LL($d$) policy, the workload $X^{\mathcal{H}(t)}$ at the cavity queue decreases by one unit rate, and for the JSQ($d$) policy, the queue length $X^{\mathcal{H}(t)}$ of the cavity queue decreases by one at a unit rate. The process $\mathcal{H}(\cdot)$ is called as the equilibrium environment process if $X^{\mathcal{H}(\cdot)}(t)$ has distribution $\mathcal{H}(t)$ for all times $t$. If $\mathcal{H}(t) = \mathcal{H}$ for all $t$, then $\mathcal{H}$ is called as equilibrium environment.

The cavity process method was used in [16], [17] to analyze the LL($d$) and the JSQ($d$) policy. A key step in the analysis is to show asymptotic independence between the workloads/queue length random variables at different queues. While the analysis for LL($d$) holds for any service requirement distribution, the proof for JSQ($d$) is only known for the case when the service requirement of a job has decreasing hazard rate distribution. In [4], this approach is used further to obtain the functional differential equation for the workload distribution of the cavity queue. In [15], several workload based load balancing policies based on redundancy were considered and the cavity process method was used to identify the workload distribution for a wide range of load balancing policies. While the asymptotic independence of the queues was only conjectured, this was very recently proved (for most of the policies of [15]) in [25] for a variety of such replication based policies.

For our $\pi(p, T_1, T_2)$ policy, we use this cavity process method along with the conjecture that the workload distribution across any finite subset of queues is asymptotically independent. For our policy, note that the potential arrival rate to the cavity queue is $\dot{\lambda} \triangleq \lambda(1 - p) + p\lambda(d - 1)$. If the copy at the cavity queue is a primary replica, then $X^{\mathcal{H}(t)} = X^{\mathcal{H}(t-)} + 1$ if $X^{\mathcal{H}(t-)} \leq T_1$ else the copy is discarded. Similarly if the replica at the cavity queue is a secondary one, then the replica is served if $X^{\mathcal{H}(t-)} \leq T_2$. Clearly, the potential arrival at the cavity queue becomes an actual arrival based on the workload level at the queue. Remarkably, for our policy there is no influence on the cavity queue of the $d - 1$ random variables with law $\mathcal{H}(\cdot)$. With the following conjecture on the asymptotic independence of the workload at the queue, using the cavity process approach, we can view the cavity queue as an $M/G/1$ queue with workload dependent arrival rates. The workload distribution of the cavity queue is in fact the equilibrium environment $\mathcal{H}$ for our system. See [26] for one possible approach to obtain the workload distribution for an $M/G/1$ queue with workload dependent arrival rates. In the following we use a different approach based on the Lindley type recursion and moment generating function (MGF) to obtain the workload distribution for the queue at cavity. We believe that this approach is novel and can be applied to more general load balancing policies beyond this work.

**Conjecture 5.** Consider the load balancing policy $\pi(p, T_1, T_2)$ and assume this system is stable for the chosen parameter values of $p, T_1$ and $T_2$ for any $N$. Then as $N \to \infty$, the system has a unique equilibrium workload distribution under which any finite number of queues are independent. Furthermore, this distribution is same as the equilibrium distribution of the cavity process.

**Remark 1.** Based on this conjecture, we first obtain the MGF for the workload at the cavity queue. We then use this to obtain the conditional mean response time for the different policies. We illustrate the accuracy of our expressions in Appendix A by comparing them with simulation experiments for different values of $N$. As a validation of the conjecture, we see that as $N$ increases, the mean response time from simulations approach the analytical values.

### III. Performance Analysis

We note that it is difficult to analyze the conditional mean response time and loss probability under general conditions. However, based on the cavity process method and the assumption of asymptotic independence, we will obtain expression for both the performance metrics. This computation is an approximation for finite number of servers. However, we empirically verify that this approximation is quite accurate even for a small number of servers.

---

1The system is trivially stable for all arrival rates when $T_1 < \infty$, since no arrivals are accepted at queue with workloads higher than the threshold $T_1$. When $T_2 < T_1 = \infty$, then the stability condition is $\lambda < \mu$, and when $T_1 = T_2 = \infty$, the stability condition is $\lambda < \mu$. 

A. Asymptotic independence of queues

For the proposed discard threshold based dispatching policy, due to the assumption on the asymptotic independence of the queues, each queue in the system can be modeled as an M/G/1 queue with workload dependent arrival. When replica of nth job with discard threshold $T$ is added to a queue $i$ with workload $w$, it is served at this queue if $T \geq w$. Further, the workload in the queue after this actual arrival will be incremented by the random service time $X_{n,i}$ of the arriving nth replica. Note that every job arrival is replicated at maximum of $d$ servers at the same time, and therefore the arrivals to different queues are correlated. We ignore this fact in our limiting analysis using the asymptotic independence and hence our analysis is an approximate one.

B. Loss probability

When both primary and secondary thresholds are finite, some jobs can be discarded from the system. Under the asymptotic independence assumption, we compute the limiting loss probability of a job being discarded in the following Lemma.

Lemma 6. The limiting loss probability of a job under discard threshold based dispatching policy $\pi(p, T_1, T_2)$ with equilibrium workload distribution $F$ and tail distribution of service time $G$ is given by

$$P_L = \bar{F}(T_1)(p\bar{F}(T_2)^d - 1 + (1 - p)).$$

Proof: From (2) in Definition 3, we obtain $P_L = \mathbb{E}\left[\prod_{j \in I_1} \tilde{\xi}_j \prod_{j \in I_2} (\xi_j + \bar{\xi}) \right]$. The result follows from the independence of the indicators $\zeta$ and $(\tilde{\xi}_j : j \in I_1 \cup I_2)$, under the assumption of asymptotic independence across the servers $j \in I_1 \cup I_2$, and the fact that the mean of indicators $\mathbb{E}\zeta = p, \mathbb{E}\xi_j = \bar{F}(T_1)\gamma_{1,j} + \bar{F}(T_2)\gamma_{2,j}$ for all $j \in I_1 \cup I_2$.

C. Conditional mean response time

Next, we characterize the mean response time for a job under the dispatching policy $\pi(p, T_1, T_2)$. Note that when the discard thresholds $T_1, T_2$ are finite, then all jobs that arrive at a server with workload $w > T_1$ will be lost. For lost jobs, the response time metric is meaningless. Hence, we obtain the conditional mean response time given that the job is not discarded. A job is serviced when at least one of its replicas is not discarded at the servers sampled by the dispatcher, i.e. when workload at one of these servers is smaller than or equal to the corresponding discard threshold.

Theorem 7. The conditional mean response time of an undiscarded job under $\pi(p, T_1, T_2)$ policy with equilibrium workload distribution $F$ and tail distribution of service time $G$ is given by

$$\tau = \frac{1}{1 - P_L} \int_x \mathbb{E}\left[\left(F(T_1) + k(x, T_1)(F(T_2) + k(x, T_2))^{d-1} - \bar{F}(T_1)\bar{F}(T_2)^{d-1}\right) + (1 - p)k(x, T_1)\right]dx,$$

where $k(x, T) \triangleq \mathbb{E}\left[G(x - W)1_{\{W \leq T\}}\right]$.

Proof: The tail distribution of an undiscarded job in the system is denoted by $\bar{H}$ and defined in Definition 4. Therefore, the mean response time for an undiscarded job can be written as $\mathbb{E}[R] = \int_x \bar{H}(x)dx$. Next, we derive an expression for the tail distribution $\bar{H}$ of the response time for each undiscarded job under $\pi(p, T_1, T_2)$ policy. Recall that $\zeta$ is the indicator that secondary replicas are created, and $I_1, I_2$ denote the disjoint random sets of servers where primary and secondary replicas are dispatched. Then, the indicator that the job replica at server $j \in I_1 \cup I_2$ with workload $W_j$ is not discarded is defined in (1). Recall that the set of servers, where the job replicas are not discarded is denoted by $I = \{j \in I_1 : \xi_j = 1\} \cup \{j \in I_2 : \zeta = 1, \xi_j = 1\}$, and the indicator of an undiscarded job is $\xi = 1_{\{I \neq \emptyset\}}$. Therefore, we can write the indicator of response time of an undiscarded job being larger than a threshold $x \in R_+$ as

$$1_{\{R > x\}} = \xi \prod_{j \in I} 1_{\{W_j + X_j > x\}} = \xi \prod_{j \in I_1} (1_{\{W_j + X_j > x\}} + \bar{\xi}_j) \prod_{j \in I_2} (\xi \prod_{j \in I_1} 1_{\{W_j + X_j > x\}} + \bar{\xi}_j) + \bar{\zeta}.$$
Substituting (2) for the indicator $\xi$ in the above equation, using the fact that $\xi_j \xi_j = \xi = 0$, and re-arranging the terms, we can write
\[
\mathbb{1}_{\{R > x\}} = \zeta \left( \prod_{j \in I_1 \cup I_2} (\xi_j \mathbb{1}_{\{W_j + X_j > x\}} + \bar{\xi}_j) - \prod_{j \in I_1} \bar{\xi}_j \right) + \bar{\zeta} \left( \prod_{j \in I_1} (\xi_j \mathbb{1}_{\{W_j + X_j > x\}} + \bar{\xi}_j) - \prod_{j \in I_1} \bar{\xi}_j \right).
\]
Taking expectation on both sides of the above equations, using the independence of indicators $\zeta$ and $\xi_j : j \in I_1 \cup I_2$ with the respective means $\mathbb{E}[\xi] = p$ and $\mathbb{E}[\xi_j|I_1, I_2] = F(T_1) \gamma_{1,j} + F(T_2) \gamma_{2,j}$, and the definition of $k(x, T) = \mathbb{E} \left[ \mathbb{1}_{\{W_j \leq T\}} \mathbb{1}_{\{X_j + W_j > x\}} \right]$, we obtain the tail distribution of the response time for an undiscarded job as
\[
\mathbb{E}[\mathbb{1}_{\{R > x\}}|I_1, I_2] = (1 - p)k(x, T_1) + p\left( (k(x, T_1) + F(T_1))(k(x, T_2) + F(T_2))^d - F(T_1)F(T_2)^d - 1 \right).
\]
Since the right hand side of the above equation doesn’t depend on $I_1, I_2$, and hence we have $\bar{H}(x) = \mathbb{E}[\mathbb{1}_{\{R > x\}}]|I_1, I_2]$. The result follows from equation (3) for conditional mean of response time.

**Remark 2.** From the non-negativity of distribution functions, we can exchange two integrals from Monotone convergence theorem. Therefore, we have
\[
\int_{x \in \mathbb{R}_+} k(x, T)dx = \frac{F(T)}{\mu} + \mathbb{E}\left[ W \mathbb{1}_{\{W \leq T\}} \right].
\]
Defining $k(x) \triangleq \lim_{T \to \infty} k(x, T)$, we observe that $\int_{x \in \mathbb{R}_+} k(x)dx = \mathbb{E}W + \frac{1}{\mu}$.

**Remark 3.** When the replication probability $p$ equals 1, the tail distribution of response time simplifies to
\[
\bar{H}(x) = \frac{(F(T_1) + k(x, T_1))(F(T_2) + k(x, T_2))^d - 1}{F(T_1)F(T_2)^d - 1}.
\]

**Remark 4.** When the thresholds $T_1$ and $T_2$ are infinity, we see the tail workload distributions $F(T_1) = F(T_2) = 0$ and we have $k(x) = k(x, \infty) = \mathbb{E}G(x - W)$. It follows that the tail distribution of response time is $\bar{H}(x) = pk(x)^d + (1 - p)k(x)$.

**IV. Workload Distribution and Conditional Mean Response Time Under Exponential Service Times**

In this section, we evaluate the workload distribution $F$ in the cavity queue under various load balancing polices discussed in section II-B when the service times of each job is independent and follows an identical exponential distribution with rate $\mu$. We choose the service times to be exponentially distributed as they are amenable to analytical computations, due to their memoryless property. Let us first introduce some preliminary definitions prior to introducing the results.

We denote the indicator that the $j$th server is selected by $n$th job as a primary or secondary server by $\gamma_{1,j}^n$ and $\gamma_{2,j}^n$ respectively. Recall that the workload seen by the $n$th job arrival at server $j$ is $W_{n,j}$ and the service time for n$th$ job if it joins server $j$ is given by $X_{n,j}$. Since we are interested in a single cavity queue $j$, we drop the subscript $j$ in the following. Using Lindley’s recursion for single queue workload sequence ($W_n : n \in \mathbb{N}$) in terms of random service time sequence ($X_n : n \in \mathbb{N}$), inter-arrival time sequence ($T_n : n \in \mathbb{N}$), and indicator $\zeta^n$ denoting whether secondary replicas are created or not, we get
\[
W_{n+1} = (W_n + X_n((\gamma_1^n + \zeta^n \gamma_2^n)\mathbb{1}_{\{W_n \in [0, T_2]\}} + \gamma_1^n \mathbb{1}_{\{W_n \in (T_2, T_1]\}})) - T_{n+1}+, \quad n \in \mathbb{Z}_+.
\]
That is, we have
\[
W_{n+1} = \begin{cases} 
(W_n - T_{n+1})+, & W_n \in (T_1, \infty), \\
(1 - \gamma_1^n)(W_n - T_{n+1}) + \gamma_1^n(W_n + X_n - T_{n+1})+, & W_n \in (T_2, T_1], \\
(1 - \gamma_1^n - \zeta^n \gamma_2^n)(W_n - T_{n+1}) + \gamma_1^n \zeta_2^n(W_n + X_n - T_{n+1})+, & W_n \in [0, T_2]. 
\end{cases}
\]
In order to derive the workload distribution in the cavity queue, we make use of the moment generating function of the workload.
Definition 8. The moment generating function of the limiting workload $W$ in a single queue, restricted to different workload regimes is defined as

$$
\Phi_W(\theta) \triangleq \mathbb{E}\left[ e^{-\theta W} \right], \quad \Phi_2(\theta) \triangleq \mathbb{E}\left[ e^{-\theta W} 1_{\{W>T_2\}} \right], \quad \Phi_1(\theta) \triangleq \mathbb{E}\left[ e^{-\theta W} 1_{\{W>T_1\}} \right].
$$

Theorem 9. For an $N$ server system with i.i.d. exponential service times of rate $\mu$ and Poisson arrivals of rate $N\lambda$, the moment generating function $\Phi_W(\theta)$ for the waiting time of admitted jobs at any queue under $\pi(p,T_1,T_2)$ policy is given by

$$
F(0)(1 + \frac{\tilde{\lambda}}{\theta + \mu - \lambda}) + ((\mu - \lambda)\bar{F}(T_2) + \lambda\bar{F}(T_1))\left[ \frac{e^{-\theta T_2}}{\theta + \mu - \lambda} - \frac{e^{-\theta T_2}}{\theta + \mu - \lambda} \right] - \mu\bar{F}(T_1)\left[ \frac{e^{-\theta T_1}}{\theta + \mu - \lambda} - \frac{e^{-\theta T_1}}{\theta + \mu} \right],
$$

where $F(0) = 1 - \frac{\lambda}{\mu} + \left[ \frac{\lambda - \lambda}{\mu} \bar{F}(T_2) + \frac{\lambda}{\mu} \bar{F}(T_1) \right]$.

Proof: The detailed proof is in Appendix B.

Corollary 10. For an $N$ server system with i.i.d. exponential service times of rate $\mu$ and Poisson arrivals of rate $N\lambda$, the single queue workload distribution under $\pi(p,T_1,T_2)$ policy is given by

$$
F(w) = F(0)\left(1 + \frac{\tilde{\lambda}(1-e^{-(\mu-\lambda)w})}{\mu - \lambda}\right) - \mu\bar{F}(T_1)\left[ \frac{1-e^{-(\mu-\lambda)(w-T_1)_+}}{\mu - \lambda} - \frac{1-e^{-(\mu-\lambda)(w-T_1)_+}}{\mu} \right]
+ ((\mu - \lambda)\bar{F}(T_2) + \lambda\bar{F}(T_1))\left[ \frac{1-e^{-(\mu-\lambda)(w-T_2)_+}}{\mu - \lambda} - \frac{1-e^{-(\mu-\lambda)(w-T_2)_+}}{\mu} \right].
$$

Next, we study some special cases of the $\pi(p,T_1,T_2)$ policy listed in Section II-B.

A. Selective replication with identical thresholds

First, we study the system under the selective replication with identical thresholds policy, $\pi(p,T,T)$. Next result follows from Corollary 10 by substituting $T_1 = T_2$.

Corollary 11. For an $N$ server system with i.i.d. exponential service times of rate $\mu$ and Poisson arrivals of rate $N\lambda$, the workload distribution at the cavity queue at stationarity under $\pi(p,T,T)$ policy, is given by

$$
F(w) = \begin{cases} 
F(0)\left(\frac{\mu}{\mu-\lambda} - \frac{\lambda}{\mu-\lambda} e^{-(\mu-\lambda)w}\right), & 0 < w \leq T \\
F(T) + \frac{\mu}{\lambda} e^{\lambda T} F(0) (e^{-\mu T} - e^{-\mu w}), & w > T 
\end{cases}
$$

where $F(0) = \left[ \frac{1-e^{-(\mu-\lambda)w}}{1-e^{-(\mu-\lambda)w+\lambda^2 T_2}} \right] 1_{\{\mu \neq \lambda\}} + \frac{\lambda T + 1}{\lambda T + 2} 1_{\{\mu = \lambda\}}$ and $F(T) = \frac{\mu}{\lambda} (1 - F(0))$.

Using the above corollary, we now compute the loss probability and conditional mean response time using Theorem 7.

Corollary 12. The loss probability of a job under discard threshold based dispatching policy $\pi(p,T,T)$ with equilibrium workload distribution $F$ and tail distribution of service time $G$, is given by

$$
P_L = p\bar{F}(T)^d + (1-p)\bar{F}(T),
$$

where $\bar{F}(T) = 1 - \frac{\mu}{\lambda} (1 - F(0))$ and probability of zero workload $F(0)$ is given in Corollary 11.

From Theorem 7, we know the conditional mean response time under $\pi(p,T,T)$ policy is

$$
\tau = \frac{1}{1-P_L} \int_x \left( p \left[ (\bar{F}(T) + k(x,T))^d - \bar{F}(T)^d \right] + (1-p)k(x,T) \right) dx.
$$

Thus, we see that computing the term $k(x,T)$ shall allow us to evaluate the mean response time of the $N$ server system under the policy $\pi(p,T,T)$. The next lemma provides us this result.
Random routing illustrate the advantages of our policy over random routing. For the policy $\pi(p, T, T)$ with fixed number of servers $N = 20$, arrival rate $\lambda = 0.3$, probability $p = 1$, service rate $\mu = 1$, conditional mean response time $\tau$ as a function of threshold $T$ is plotted in Fig. 1a, loss probability $P_L$ as a function of threshold $T$ is in Fig. 1b, tradeoff between conditional mean response time $\tau$ and loss probability $P_L$ is in Fig. 1c.

**Lemma 13.** For an $N$ server system with i.i.d. exponential service times of rate $\mu$ and Poisson arrivals of rate $N\lambda$, we can find the following constants under $\pi(p, T, T)$ policy,

\[
F(T) = 1 - F(0) \left[ \frac{\mu}{\mu - \lambda} - \frac{\lambda}{\mu - \lambda} e^{-(\mu - \lambda)T} \right],
\]

\[
F(0) = \frac{(1 - \frac{\bar{\lambda}}{\mu})}{1 - (\frac{\bar{\lambda}}{\mu})^2 e^{-(\mu - \lambda)T}}.
\]

The function $k(x, T)$ is given by

\[
k(x, T) = \begin{cases} 
F(0) e^{-\mu x} e^{\bar{\lambda}T}, & x \geq T \\
F(0) \left( \frac{\mu}{\mu - \lambda} e^{-(\mu - \lambda)x} - \frac{\bar{\lambda}}{\mu - \lambda} e^{-(\mu - \lambda)T} \right), & x < T.
\end{cases}
\]

**Proof:** We know that the service time are exponential and hence the tail service time distribution is $\bar{G}(x) = e^{-\mu(x)_+}$, where $(x)_+ = \max \{x, 0\}$. Therefore, we can write

\[
k(x, T) = \mathbb{E}[\bar{G}(x - W) \mathbb{1}_{\{W \leq T\}}] = F(T) - F(T \wedge x) + e^{-\mu x} \int_0^{T \wedge x} e^{\mu w} dF(w).
\]

Considering the two cases when $x \geq T$ and $x < T$, we get $k(x, T)$ as

\[
k(x, T) = \begin{cases} 
e^{-\mu x} \int_0^{T \wedge x} e^{\mu w} dF(w), & x \geq T \\
F(T) - F(x) + e^{-\mu x} \int_0^x e^{\mu w} dF(w), & x < T.
\end{cases}
\]

The result follows from the workload distribution $F$ given in Corollary 11.

In Fig. 1, we consider a numerical plot for the policy $\pi(p, T, T)$ for the number of servers $N = 20$, the normalized arrival rate $\lambda = 0.3$, and probability of replication $p = 1$. We compare the conditional mean response time and the loss probabilities for different values of discard threshold $T$ and for different choices of number of replicas $d$. From Fig. 1a, we see that the conditional mean response time increases in threshold $T$. At the same time, we observe that the loss probability decreases in threshold $T$ in Fig. 1b. For values of discard threshold $T$ between 0 and 1, there seems to be significant gain from using our policy as compared to random routing. Further, we note that the tradeoff in terms of the loss probability also seems marginal, since the maximum loss probability is observed to be around 0.095. The tradeoff between the conditional mean response time of admitted jobs and loss probability $P_L$ for different numbers of replicas $d$, is illustrated in Fig. 1c. We observe that for the number of replicas $d > 1$, the proposed policy can offer a significantly lower conditional mean response time compared to random routing policy by allowing a small loss probability.

In Fig. 2, we study the behaviour of conditional mean response time and the loss probability for $\pi(p, T, T)$ as the normalized arrival rate $\lambda$ increases. We choose the number of servers $N = 20$, discard threshold $T = 1.5$, and probability of replication $p = 1$. Fig. 2a and Fig. 2b illustrate the advantages of our policy over random routing. For lower values of normalized arrival rates $\lambda$, our policy for $d > 1$ outperforms random routing. Even for higher values
Random routing illustrates the tradeoff between the two performance parameters of arrival rates, the proposed policy has a lower mean response time as compared to random routing. Moreover, as the discard thresholds are finite, the system stays stable for any arrival rate unlike a random routing policy. However, this comes at the cost of loss probabilities and we see that the loss probability rises up to 0.35 in the provided plots. Note that the case of single primary replica \( d = 1 \) is different from random routing. When \( d = 1 \) in \( \pi(p, T, T) \) policy, it implies that only one primary replica is executed if the workload at the randomly selected primary server is smaller than the discard threshold \( T \). Fig 2c illustrates the tradeoff between the two performance metrics for this policy. This study shows that the right value for the number of replicas \( d \) must be chosen for a given admissible loss probability in order to minimize the conditional mean response time.

**B. Selective replication with no loss**

We next study the \( N \) server system under the selective replication with no loss policy. Specifically, we assume that the primary discard threshold \( T_1 = \infty \), and the secondary discard threshold \( T_2 < T_1 \) is finite. In this case, the system is stable only if \( \lambda < \mu \). First, we obtain the following result from Corollary 10 by substituting \( T_1 = \infty \).

**Corollary 14.** For an \( N \) server system with i.i.d. exponential service times of rate \( \mu \) and Poisson arrivals of rate \( N\lambda \), the stationary workload distribution at the cavity queue under \( \pi(p, \infty, T_2) \) policy exists only for \( \lambda < \mu \), and is given by

\[
F(w) = \begin{cases} 
F(0)(\frac{\mu}{\mu-\lambda} - \frac{\lambda}{\mu-\lambda}e^{-(\mu-\lambda)w}), & w \leq T_2 \\
F(T_2) + \frac{\lambda}{\mu-\lambda}F(0)(e^{-(\mu-\lambda)T_2}(e^{-(\mu-\lambda)T_2} - e^{-(\mu-\lambda)w})), & w > T_2,
\end{cases}
\]

where the probability mass at 0 is \( F(0) = \frac{(1-\frac{\lambda}{\mu})(1-\frac{\lambda}{\mu})}{(1-\frac{\lambda}{\mu})(1-\frac{\lambda}{\mu})e^{-(\mu-\lambda)T_2}} \).

**Remark 5.** Note that the loss probability is 0 under this policy. Then, from Theorem 7, we get the conditional mean response time

\[
\tau = \int p[k(x, \infty)(\bar{F}(T_2) + k(x, T_2))^{d-1}] + (1-p)k(x, \infty)dx.
\]  

From the definition of \( k(x, T) = \mathbb{E}[G(x-W)1_{W\leq T}] \leq F(T) \), it follows that \( (\bar{F}(T_2) + k(x, T_2)) \leq 1 \), and hence the conditional mean response time for this special case is minimized for \( p = 1 \).

The next lemma provides us with the terms \( k(x, T_2), k(x, \infty) \) and \( \bar{F}(T_2) \) that enable us to compute the conditional mean response time \( \tau \) under the scheduling policy \( \pi(1, \infty, T_2) \). Note that, we provide the results only for the regime of arrival rates where the system is stable, that is, when \( \lambda < \mu \).

**Lemma 15.** For a stable \( N \) server system with i.i.d. exponential service times of rate \( \mu \) and Poisson arrivals of rate \( N\lambda \), the function \( k(x, T_2) \) under the \( \pi(p, \infty, T_2) \) policy is

\[
k(x, T_2) = \begin{cases} 
k(0)e^{-\mu x}e^{\mu T_2}, & x \geq T_2 \\
k(0)[\frac{\mu}{\mu-\lambda}e^{-(\mu-\lambda)x} - \frac{\lambda}{\mu-\lambda}e^{-(\mu-\lambda)T_2}], & x < T_2,
\end{cases}
\]
We can also find the function \( k(x, \infty) \) as

\[
k(x, \infty) = k(x, T_2) + \begin{cases} F(0)\lambda e^{(\lambda-\mu)T_2} e^{-\mu x} \left[ \frac{e^{\lambda x} - e^{\lambda T_2}}{\lambda} + \frac{e^{\lambda x}}{\mu - \lambda} \right], & x \geq T_2 \\
\frac{\lambda}{\mu - \lambda} F(0) e^{(\mu-\lambda)T_2}, & x < T_2.\end{cases}
\]

The probability mass at 0 is given by \( F(0) = \left[ \frac{1}{\lambda} \left( \frac{1 - e^{-(\mu-\lambda)T_2}}{\mu - \lambda} + \frac{e^{-(\mu-\lambda)T_2}}{\mu - \lambda} \right) + 1 \right]^{-1} \).

Proof: Since the service time is exponentially distributed with rate \( \mu \), we get \( G(x) = e^{-\mu x} \). Therefore, we can write the function

\[
k(x, T) = \mathbb{E} \left[ G(x - W) \mathbb{1}_{\{W \leq T\}} \right] = F(T) - F(T \wedge x) + e^{-\mu x} \int_0^{T \wedge x} e^{\mu w} dF(w).
\]

Setting \( T = \infty \) in the above equation, we get

\[
k(x, \infty) = e^{-\mu x} \int_0^\infty e^{\mu w} dF(w).
\]

Substituting the workload distribution \( F \) from Corollary 14, we get the result.

We compare the conditional mean response time for jobs under policy \( \pi(1, \infty, T_2) \) as a function of normalized arrival rate \( \lambda \) for different number of replications \( d \), the number of servers \( N = 20 \), and the exponential service rates of jobs to be \( \mu = 1 \) in Figure 3. We choose secondary discard threshold \( T_2 = 2 \) which is twice the mean service time of a job. As anticipated, lower values of replications \( d \) are preferable with increase in the normalized arrival rate \( \lambda \). That is, when the arrival rates are high, the creation of redundant replicas causes increase in load in the system which adversely affects the performance. Recall that for \( d = 1 \), this policy is same as the random routing policy, and observed to be the preferred policy in a high load regime. Also evident from the figure is the fact that the stability condition is \( \lambda < \mu \), independent of number of replicas \( d \).

![Figure 3](image-url)

**Fig. 3:** For the policy \( \pi(p, \infty, T_2) \) with threshold \( T_2 = 2 \), number of servers \( N = 20 \), probability \( p = 1 \), service rate \( \mu = 1 \), conditional mean response time \( \tau \) as a function of arrival rate \( \lambda \) for different values of replicas \( d \in \{1, 3, 6, 9, 12\} \).

We compare the mean response time under policy \( \pi(1, \infty, T_2) \) as a function of secondary discard threshold \( T_2 \) for different number of replications \( d \), in Fig. 4. We choose the normalized arrival rate \( \lambda = 0.3 \) and the number of servers \( N = 20 \). We observe that the choice of replication factor \( d \) affects the conditional mean response time, and the conditional mean response time minimizing replication factor \( d \) depends on the discard threshold \( T_2 \). Alternatively, it implies that if the number of secondary replicas \( d \) is chosen *apriori*, then the secondary discard threshold \( T_2 \) should be chosen carefully to minimize the conditional mean response.

Remark 6. Let us consider the \( \pi(p, \infty, \infty) \) policy, which is a special case of \( \pi(p, T, T) \) for \( T = \infty \) as well as of \( \pi(p, \infty, T_2) \) for \( T_2 = \infty \). It is to be noted that there is no jobs lost in such a system and therefore, the loss probability is zero. Further, as the arrival rate to any queue is \( \lambda \), the system is stable only if \( \lambda < \mu \). For this regime, using Lemma 13 it can be shown that \( k(x, \infty) = e^{-\mu (x-\lambda)} \). From this, the conditional mean response time for exponential service time distribution can be found to be

\[
\tau = \frac{p}{(\mu - \lambda)d} + \frac{1 - p}{\mu - \lambda}
\]
In Fig. 5, we plot this conditional mean response time for policy \( \pi(p, \infty, \infty) \) as a function of \( \lambda \) for \( N = 20 \) servers, \( p = 1 \), and for different values of \( d \). The figure is indicative of the stability condition \( \lambda < \frac{1}{d} \) for this policy. The performance gain from using larger values of \( d \) is also evident, but this comes at a cost of requiring a stricter stability condition. Of course, the clear advantage of this policy over random routing (\( d = 1 \)) is limited to lower arrival rates. At higher values of \( \lambda \), the fact that the copies cannot be cancelled impacts the performance of the system severely. For better clarity, we also provide the percentage improvement of conditional mean response time of the policy \( \pi(p, \infty, \infty) \) over random routing policy across stable regions in Table I.

![Fig. 4](threshold_response_rate.png)

**Fig. 4:** For the policy \( \pi(p, \infty, T_2) \) with number of servers \( N = 20 \), arrival rate \( \lambda = 0.3 \), service rate \( \mu = 1 \), conditional mean response time \( \tau \) as a function of threshold \( T_2 \) for different values of replicas \( d \in \{1, 4, 6, 9, 12\} \).

![Fig. 5](replica_response_rate.png)

**Fig. 5:** For the policy \( \pi(p, \infty, \infty) \) with fixed number of servers \( N = 20 \), probability \( p = 1 \), service rate \( \mu = 1 \), conditional mean response time \( \tau \) as a function of arrival rate \( \lambda \) for different values of replicas \( d \in \{1, 2, 3, 4, 6, 9\} \).

| Replicas | \( \lambda = 0.1 \) | \( \lambda = 0.15 \) | \( \lambda = 0.2 \) | \( \lambda = 0.25 \) |
|----------|----------------|----------------|----------------|----------------|
| \( d=2 \) | 43.6\% | 39.18\% | 33.19\% | 24.79\% |
| \( d=3 \) | 57\% | 48.26\% | 32.91\% | -1\% |
| \( d=4 \) | 62.29\% | 46.4\% | -1.91\% | NA |

**TABLE I:** Percentage improvement of conditional mean response time of the policy \( \pi(p, \infty, \infty) \) against random routing with fixed servers \( N = 20 \), probability \( p = 1 \) [See Fig. 5].

From the above studies, we observe that introduction of secondary replicas add to the load in the system and deteriorates the system performance when arrival rates are high. Therefore, in the following section, we study a variant of selective replication policy, where we replicate only on idle server.
C. Selective replication on idle servers

As mentioned above, we next study the special case of $\pi(1, \infty, T_2)$ policy where the discard threshold $T_2 = 0$. We note that this implies that primary replica is chosen uniformly at random from $N$ servers, and $(d-1)$ secondary replicas are chosen uniformly at random from remaining $N-1$ servers. The secondary replicas are added only if the sampled secondary servers are idle. Since this policy is a special case of selective replication with no loss policy, we can obtain the mean response time directly.

Lemma 16. The mean response time of any job under the dispatching policy $\pi(1, \infty, 0)$ when service times of each job is i.i.d. exponential with rate $\mu$ and arrivals are Poisson with rate $N\lambda$, is given by

$$\tau = \sum_{n=0}^{d-1} \binom{d-1}{n} \bar{F}(0)^{d-n} F(0)^n \left[ \frac{d\mu\lambda}{(\mu-\lambda)(\mu(n+1)-\lambda)} - \frac{\lambda(d-1)}{\lambda\mu(n+1)} \right],$$

for $\lambda < \mu$ and $F(0) = \frac{\mu-\lambda}{\mu+\lambda(d-1)}$ and $\bar{F}(0) = 1 - F(0)$.

Proof: Substituting $T_2 = 0$ in Lemma 15 and substituting the terms in (7) gives the result.

$\blacksquare$

| Replicas $d$ | $\lambda = 0.2$ | $\lambda = 0.4$ | $\lambda = 0.6$ | $\lambda = 0.8$ |
|-------------|----------------|----------------|----------------|----------------|
| $d=3$       | 43.14%         | 22.02%         | 8.43%          | 1.74%          |
| $d=6$       | 57.23%         | 29.30%         | 11.01%         | 2.22%          |
| $d=9$       | 62.33%         | 31.97%         | 11.93%         | 2.39%          |
| $d=12$      | 64.96%         | 33.55%         | 1.40%          | 2.47%          |

TABLE II: Percentage improvement of conditional mean response time of the policy $\pi(p, \infty, 0)$ over random routing with fixed servers $N = 20$, probability $p = 1$ [See Fig. 6].

In Fig. 6, we compare the mean sojourn time under dispatch policy $\pi(1, \infty, 0)$ as a function of normalized arrival rate $\lambda$ for different numbers of replica $d$. We have selected the total number of servers as $N = 20$, and exponential service rate $\mu = 1$. It follows from the figure that for lower arrival rates, a higher number of replicas $d$ is preferred. Moreover, the performance of the policies with secondary replicas is never worse than the random routing policy. That is, as opposed to other policies seen earlier, the additional replicas are executed only if the server is idle in this policy. Therefore, a higher choice of replication factor $d$ does not increase the system load significantly. For moderate to higher values of arrival rates, all the different choices of number of replicas have a similar performance with the stability condition being $\lambda < \mu$, independent of number of replicas $d$. We also provide the percentage improvement of conditional mean response time of the policy $\pi(p, \infty, 0)$ over random routing policy for various values of normalized arrival rate in Table II.

Fig. 6: For the policy $\pi(p, \infty, 0)$ with fixed number of servers $N = 20$, service rate $\mu = 1$, probability 1, conditional mean response time $\tau$ as a function of arrival rate $\lambda$ for different values of replicas $d \in \{1, 3, 6, 9, 12, 15\}$.
In this work, we consider load balancing policies that are suitable for rigid working environments with no feedback, no memory, and no synchronized replica cancellations. In such settings, random routing policy is the default choice for load balancing. Equipped with the ability to create replicas and pass on cancellation instructions to servers, we have introduced a new class of policy namely $\pi(p,T_1,T_2)$. An attractive feature of this policies is the server-side cancellation of replica based on discard thresholds $T_1$ and $T_2$. In this work we have shown that this policy (and several of its special cases) not only offer a marked improvement over the random routing policy (for suitable choice of parameters $\lambda, d$) but do so without using any communication from the servers. We analyze this policy using the cavity queue approach and the conjecture on asymptotic independence of queues. Using the MGF approach, we characterize the mean conditional sojourn time of a job and the loss probability for the policy as part of our key result.

A key assumption in our analysis has been the exponential service requirements for jobs, and that the copies of jobs require independent and identically distributed service time. We believe that relaxing these assumptions and analyzing the proposed $\pi(p,T_1,T_2)$ policy for more general service time distributions and for the case of identical replicas is an interesting open direction. One can think of more nuanced policies without feedback such as replicating only short jobs (if the service requirement of a job is known at arrival) or replicating only if the identical replicas is an interesting open direction. One can think of more nuanced policies without feedback such as replicating only short jobs (if the service requirement of a job is known at arrival) or replicating only if the primary copy is discarded. Analyzing such policies is also part of our agenda. Finally, while the performance of $\pi(p,T_1,T_2)$ seems to be good (compared to random routing) for lower values of normalized arrival rates $\lambda$, it would be interesting to investigate if there exist such no feedback policies that are better than random routing even for higher values of normalized arrival rates $\lambda$.

**REFERENCES**

[1] W. Winston, “Optimality of the shortest line discipline,” *Journal of Applied Probability*, pp. 181–189, Mar. 1977.
[2] M. Mitzenmacher, “The power of two choices in randomized load balancing,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 12, no. 10, pp. 1094–1104, Oct. 2001.
[3] N. D. Vvedenskaya, R. L. Dobrushin, and F. I. Karpelevich, “Queueing system with selection of the shortest of two queues: An asymptotic approach,” *Problemy Peredachi Informatsii*, vol. 32, no. 1, pp. 20–34, 1996.
[4] T. Hellemans and B. V. Houdt, “On the power-of-d-choices with least loaded server selection,” *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, vol. 2, no. 2, p. 27, Jun. 2018.
[5] U. Ayesta, T. Bodas, and I. M. Verloop, “On redundancy-d with cancel-on-start aka join-shortest-work (d),” *ACM SIGMETRICS Performance Evaluation Review*, vol. 46, no. 2, pp. 24–26, Jan. 2019.
[6] M. van der Boor, S. Borst, and I. van Leeuwaarden, “Hyper-scalable jq with sparse feedback,” *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, vol. 3, no. 1, pp. 1–37, Mar. 2019.
[7] M. van der Boor, M. Zabeldia, and S. Borst, “Zero-wait load balancing with sparse messaging,” *Operations Research Letters*, Apr. 2020.
[8] Y. Lu, Q. Xie, G. Kliot, A. Geller, J. R. Larus, and A. Greenberg, “Join-idle-queue: A novel load balancing algorithm for dynamically scalable web services,” *Performance Evaluation*, vol. 68, no. 11, pp. 1056–1071, Nov. 2011.
[9] D. Gamarnik, J. N. Tsitsiklis, and M. Zubeldia, “Delay, memory, and messaging tradeoffs in distributed service systems,” *Stochastic Systems*, vol. 8, no. 1, pp. 45–74, Jun. 2018.
[10] T. Hellemans and B. V. Houdt, “Performance analysis of load balancing policies with memory,” in *Proceedings of the 13th EAI International Conference on Performance Evaluation Methodologies and Tools*, May 2020, pp. 27–34.
[11] U. Ayesta, T. Bodas, and I. M. Verloop, “On a unifying product form framework for redundancy models,” *Performance Evaluation*, vol. 127, pp. 93–119, Nov. 2018.
[12] K. Gardner, M. Harchol-Balter, A. Scheller-Wolf, M. Velednitsky, and S. Zbarsky, “Redundancy-d: The power of d choices for redundancy,” *Operations Research*, Aug. 2017.
[13] U. Ayesta, T. Bodas, J. Dorosman, and I. Verloop, “A token-based central queue with order-independent service rates,” *arXiv preprint arXiv:1902.02137*, 2019.
[14] K. Lee, R. Pedarsani, and K. Ramchandran, “On scheduling redundant requests with cancellation overheads,” *IEEE/ACM Transactions on Networking (TON)*, vol. 25, no. 2, pp. 1279–1290, 2017.
[15] T. Hellemans, T. Bodas, and B. V. Houdt, “Performance analysis of workload dependent load balancing policies,” *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, vol. 3, no. 2, pp. 1–35, Jun. 2019.
[16] M. Bramson, Y. Lu, and B. Prabhakar, “Randomized load balancing with general service time distributions,” in *ACM SIGMETRICS Performance Evaluation Review*, vol. 38, no. 1, Jun. 2010, pp. 275–286.
[17] ——, “Asymptotic independence of queues under randomized load balancing,” *Queueing Systems*, vol. 71, no. 3, pp. 247–292, Jul. 2012.
[18] Z. Qiu, J. F. Pérez, R. Birke, L. Chen, and P. G. Harrison, “Cutting latency tail: Analyzing and validating replication without canceling,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 28, no. 11, pp. 3128–3141, May 2017.
[19] S. Liu, H. Xu, L. Liu, W. Bai, K. Chen, and Z. Cai, “Repnet: Cutting latency with flow replication in data center networks,” *IEEE Transactions on Services Computing*, Jan. 2018.
[20] N. F. Maxemchuk, “Dispersity routing,” in Proceedings of ICC, vol. 75, Jun. 1975.
[21] D. Cheng, J. Rao, Y. Guo, and X. Zhou, “Improving mapreduce performance in heterogeneous environments with adaptive task tuning,” in Int. Middle. Conf. (Middleware). Bordeaux, France: ACM, 2014, pp. 97–108.
[22] K. Lee, M. Lam, R. Pedarsani, D. Papailiopoulos, and K. Ramchandran, “Speeding up distributed machine learning using codes,” IEEE Trans. Inf. Theory, vol. 64, no. 3, pp. 1514–1529, March 2018.
[23] R. Bitar, P. Parag, and S. El Rouayheb, “Minimizing latency for secure distributed computing,” in IEEE Inter. Symp. Info. Theory (ISIT), Aachen, Germany, 2017, pp. 2900–2904.
[24] A. O. Al-Abbasi and V. Aggarwal, “Video streaming in distributed erasure-coded storage systems: Stall duration analysis,” IEEE/ACM Trans. Netw., vol. 26, no. 4, pp. 1921–1932, 2018.
[25] S. Shneer and A. Stolyar, “Large-scale parallel server system with multi-component jobs,” arXiv preprint arXiv:2006.11256, 2020.
[26] R. Bekker, S. C. Borst, O. J. Boxma, and O. Kella, “Queues with workload-dependent arrival and service rates,” QS, vol. 46, no. 3-4, pp. 537–556, Mar. 2004.
APPENDIX A
MODEL VALIDATION

In this section, we discuss the accuracy of our theoretical results and compare them with simulation experiments. We obtained the conditional mean sojourn time $\tau$ for undiscarded jobs and the probability of discard $P_L$ under proposed probabilistic redundancy policy $\pi(p, T_1, T_2)$, based on the conjecture of the asymptotic independence of the queues. The workload distribution for the cavity queue under policy $\pi(p, T_1, T_2)$ has a closed form expression for exponentially distributed service time, and is provided in Corollary 10. The expression for the conditional mean sojourn time under policy $\pi(p, T_1, T_2)$ is complex, and hence we have omitted it. Instead, we restrict our validation results for three special cases: (a) deterministic $d$ replicas with identical finite discard threshold $\pi(1, T, T)$, (b) deterministic $d$ replicas with no discard $\pi(1, \infty, \infty)$, and (c) deterministic $d$ replicas with secondary replicas only at idle servers $\pi(1, \infty, 0)$.

Findings of the simulation experiments under the policy $\pi(1, T, T)$ are reported in Fig. 7. We note that this is a lossy system, where some jobs can be discarded if none of the sampled servers have workload smaller than the threshold $T$. We plot the conditional response time for $\pi(1, T, T)$ as a function of normalized arrival rate $\lambda$, when the jobs have i.i.d. exponential service times with unit mean. The identical discard threshold for primary and secondary replicas is taken as $T_1 = T_2 = 5$ and total number of replicas is selected as $d = 3$. Each experiment is run over $10^5$ iterations and we repeat this experiment for increasing number of servers $N$. We empirically compute the average response time of undiscarded jobs, as a function of normalized arrival rate $\lambda$. We observe that the empirical curve approaches our analytical computation under asymptotic independence conjecture, as the number of servers $N$ increases. This provides an empirical validation of the asymptotic independence conjecture, and hence our theoretical results. In particular, it indicates that even for the most general of our policies, the asymptotic independence of queues is indeed true.

![Fig. 7: For the policy $\pi(p, T, T)$ with fixed thresholds $T_1 = T_2 = 5$, number of replicas $d = 3$, probability $p = 1$, service rate $\mu = 1$, conditional mean response time $\tau$ as function of arrival rate $\lambda$ for different values of servers $N \in \{3, 5, 8, 10\}$.](image)

When the primary discard threshold is infinite, then all jobs get served. We illustrate a similar validation for two special cases where the primary replica is never discarded. The results for deterministic $d$ replicas with no discard ($\pi(1, \infty, \infty)$ policy) is presented in Fig. 8, and for deterministic $d$ replicas with secondary on idle servers ($\pi(1, \infty, 0)$ policy) in Fig. 9. The closed form theoretical expressions of the conditional mean response time of these policies are provided in Remark 6 and Lemma 16 respectively. As in the case of $\pi(p, T, T)$, we see that the empirically computed mean response time of undiscarded job converge to the corresponding theoretical expression with increase in number of servers $N$. This indicates that as the number of servers $N$ increases the workload across queues tend to be independent, validating our conjecture on the asymptotic independence of queues. It is remarkable to note that the theoretical values and those obtained empirically from the simulation, coincide even when the number of servers $N$ is as low as 10.

Even though, we have performed extensive validations for different values of $p, T_1$ and $T_2$ (for which closed form results are available) and have observed a similar behavior with increase in number of servers $N$, we have presented only a select few of the plots validating our models.
Fig. 8: For the policy \( \pi(p, \infty, \infty) \) with fixed probability \( p = 1 \), replicas \( d = 3 \), service rate \( \mu = 1 \), conditional mean response time \( \tau \) as function of arrival rate \( \lambda \) for different values of servers \( N \in \{3, 5, 8, 10\} \).

Fig. 9: For the policy \( \pi(p, \infty, 0) \) with fixed probability \( p = 1 \), replicas \( d = 3 \), service rate \( \mu = 1 \), conditional mean response time \( \tau \) as function of arrival rate \( \lambda \) for different values of servers \( N \in \{3, 5, 8, 10\} \).

**APPENDIX B**

**PROOF OF THEOREM 9**

This section provides the moment generating function based approach for deriving the stationary workload distribution in a single queue in an \( N \) server system with i.i.d. service times and Poisson arrivals with threshold based dispatching policy, \( \pi(p, T_1, T_2) \). Although, the proof is provided only for the case where the service times are exponentially distributed with rate \( \mu \), the same approach can be used when the service times follow a shifted exponential distribution. We omit the details due to space constraints. Let us now begin the proof by providing two simple results.

**Lemma 17.** For the interarrival time sequence \( (T_n : n \in \mathbb{N}) \), we have

\[
\mathbb{E} \left[ e^{\theta T_{n+1}} \mathbb{1}_{\{W_n + X_n > T_{n+1}\}} | W_n, X_n \right] = \frac{N \lambda}{N \lambda - \theta} \left(1 - e^{-(N \lambda - \theta)(W_n + X_n)}\right).
\]  

(9)

**Proof:** Recall that interarrival times \( (T_n : n \in \mathbb{N}) \) are i.i.d. exponential with rate \( N \lambda \), and duration \( T_{n+1} \) is independent of past workloads \( (W_1, \ldots, W_n) \) and past and present service times \( (X_1, \ldots, X_n) \) for all \( n \in \mathbb{Z}_+ \). Hence, the result follows.

**Lemma 18.** For i.i.d. exponential service time sequence \( (X_n : n \in \mathbb{N}) \) with rate \( \mu \), we have

\[
\mathbb{E} \left[ e^{-\theta X_n} \mathbb{1}_{\{X_n < T - W_n\}} | W_n \right] = \Phi_X(\theta)(1 - e^{-(\mu + \theta)(T - W_n)_+}).
\]  

(10)
In addition, we have the following identity

$$\frac{\Phi_X(\theta) - 1}{\theta} = -\frac{1}{\mu}\Phi_X(\theta).$$  \hspace{1cm} (11)

**Proof:** The $n$th service time $X_n$ is independent of workloads $(W_1, \ldots, W_n)$ seen by first $n$ incoming arrivals. The first equality follows from this observation. The second equality follows from the fact that $\Phi_X(\theta) = \frac{\mu}{\mu+\theta}$.

**Proposition 19.** For an $N$ server system with i.i.d. exponential service times of rate $\mu$, Poisson arrivals of rate $N\lambda$ under $\pi(p, T_1, T_2)$ policy and the moment generating functions of the limiting workload $W$ in a single queue defined in definition 8,

$$\Phi_W(\theta) = F(0)(1 + \frac{\lambda}{\theta + \mu - \lambda}) + ((\mu - \lambda)\tilde{F}(T_2) + \lambda\tilde{F}(T_1))e^{-\theta T_2}\left[\frac{1}{\theta + \mu - \lambda} - \frac{1}{\theta + \mu - \lambda}\right]$$

$$= -\tilde{F}(T_1)\left[\frac{1}{\theta + \mu - \lambda} - \frac{1}{\theta + \mu - \lambda}\right].$$

**Proof:** From (4), we can write the restricted moment generating function for $W_{n+1}$ in terms of $W_n$. We assume that there exists a limiting workload distribution $\lim_{n \to \infty} \mathbb{P}(W_n \leq w)$ seen by an arriving customer, which equals the limiting distribution of workload in the system by the PASTA property. At stationarity, we will take the distribution of both $W_{n+1}$ and $W_n$ as the limiting distribution $F$.

Now let us compute $\Phi_W(\theta)$. From the definition of moment generating function for workload at $(n+1)$th arrival is given by

$$\Phi_W(\theta) = \mathbb{E}\left[e^{-\theta W_{n+1} \mathbb{1}_{\{W_{n+1} > T_1\}}} + \mathbb{E}\left[e^{-\theta W_{n+1} \mathbb{1}_{\{W_n < T_1\}}} + \mathbb{E}\left[e^{-\theta W_{n+1} \mathbb{1}_{\{W_n > T_1\}}}\right]\right] + \mathbb{E}\left[e^{-\theta W_{n+1} \mathbb{1}_{\{W_n > T_2\}}}\right].$$

We will derive the three terms separately. We first observe that in the region $W_n > T_1$, we have $W_{n+1} = (W_n - T_{n+1}) \mathbb{1}_{\{W_n > T_{n+1}\}}$ from (5). Using the identity in (9), we can write

$$\mathbb{E}\left[e^{-\theta W_{n+1} \mathbb{1}_{\{W_n > T_1\}}}\right] = \mathbb{E}\left[\mathbb{1}_{\{W_n > T_1\}}e^{-N\lambda W_n}\right] + \mathbb{E}\left[\frac{N\lambda}{\theta + N\lambda - \theta} \mathbb{1}_{\{W_n > T_1\}}(e^{-\theta W_n} - e^{-N\lambda W_n})\right] = \frac{N\lambda \Phi(\theta) - \Phi(\theta) \Phi(1) (N\lambda)}{N\lambda - \theta}.$$

We next observe that in the region $W_n \in (T_2, T_1]$, we have $W_{n+1} = (W_n - T_{n+1}) \mathbb{1}_{\{W_n > T_{n+1}\}}$ with probability $1 - \frac{\lambda}{N\lambda}$, and $W_{n+1} = (W_n + X_n - T_{n+1}) \mathbb{1}_{\{W_n + X_n > T_{n+1}\}}$ with probability $\frac{\lambda}{N\lambda}$. We can write

$$\mathbb{E}\left[e^{-\theta (X_n + W_n - T_{n+1})} \mathbb{1}_{\{W_n \in (T_2, T_1]\}}\right] = \frac{1}{N\lambda - \theta} \left(\frac{\lambda}{N\lambda} (\Phi(\theta) - \Phi_1(\theta)) \Phi(\theta) - \theta \Phi_2(N\lambda) \Phi(X(N\lambda))\right).$$

For $X_n = 0$ the moment generating function $\Phi_X(\theta) = 1$, and hence combining the above two equations, we get

$$\mathbb{E}\left[e^{-\theta W_{n+1} \mathbb{1}_{\{W_n > T_2\}}}\right] = \left[\mathbb{E}\left[\mathbb{1}_{\{W_n > T_2\}}e^{-N\lambda W_n}\right] + \mathbb{E}\left[\frac{N\lambda}{\theta + N\lambda - \theta} \mathbb{1}_{\{W_n > T_2\}}(e^{-\theta W_n} - e^{-N\lambda W_n})\right] = \frac{N\lambda \Phi(\theta) - \Phi(\theta) \Phi(1) (N\lambda)}{N\lambda - \theta}.$$

We next observe that in the region $W_n \in (T_2, T_1]$, we have $W_{n+1} = (W_n - T_{n+1}) \mathbb{1}_{\{W_n > T_{n+1}\}}$ with probability $1 - \frac{\lambda}{N\lambda}$, and $W_{n+1} = (W_n + X_n - T_{n+1}) \mathbb{1}_{\{W_n + X_n > T_{n+1}\}}$ with probability $\frac{\lambda}{N\lambda}$. Repeating the steps followed above for the region $W_n > T_1$ and rearranging, we get

$$\Phi_W(N\lambda) + \left[\lambda (\Phi_2(N\lambda) - \Phi_1(N\lambda)) + \tilde{\lambda} (\Phi_W(N\lambda) - \Phi_2(N\lambda))\right] \frac{\Phi(X(N\lambda)) - 1}{N\lambda}$$

$$= \Phi_W(\theta) + \left[\lambda (\Phi_2(\theta) - \Phi_1(\theta)) + \tilde{\lambda} (\Phi_W(\theta) - \Phi_2(\theta))\right] \frac{\Phi(X(\theta)) - 1}{\theta}.$$

We observe that LHS and RHS have the form $f(\theta) = f(N\lambda)$ for an arbitrary function $f$ and variables $\theta$ and $\lambda$. Therefore, we conclude that $f(\theta) = f(0)$. Further, note that $\Phi_i(0) = \tilde{F}_i$ for $i \in [2]$. Then, using equation (11), we can write for exponential service times,

$$\Phi_W(\theta) \left[1 - \tilde{\lambda} \Phi_X(\theta)\right] + \left[\tilde{\lambda} - \lambda \Phi_2(\theta) + \frac{\lambda}{\mu} \Phi_1(\theta)\right] \Phi_X(\theta) = 1 - \frac{\lambda}{\mu} + \left[\frac{\lambda}{\mu} - \tilde{\lambda}\right] \tilde{F}(T_2) + \frac{\lambda}{\mu} \tilde{F}(T_1).$$
Now, we substitute $\Phi_1(\theta)$ and $\Phi_2(\theta)$ from equations (13) and (16) respectively in the above equation. Further incorporating equations (14) and (17) and rearranging the terms will yield equation (12).

**Remark 7.** Upon inverting the moment generating function in equation (12), we see that the complementary workload distribution function for $w \geq 0$ is given by

$$
\tilde{F}(w) = 1 - F(0) \left( 1 + \frac{\lambda(1 - e^{-(\mu - \lambda)w})}{\mu - \lambda} \right) - \mu \tilde{F}(T_1) \left( \frac{1 - e^{-(\mu - \lambda)(w - T_1)_+}}{\mu - \lambda} - \frac{1 - e^{-(\mu - \lambda)(w - T_1)_+}}{\mu} \right)
$$

In addition, we can find the constant, $F(0) = 1 - \frac{\lambda}{\mu} + \left[ \frac{\lambda - \lambda}{\mu} \tilde{F}(T_2) + \frac{\lambda}{\mu} \tilde{F}(T_1) \right]$. 

**Proposition 20.** For an $N$ server system with i.i.d. exponential service times of rate $\mu$, Poisson arrivals of rate $N\lambda$ under $\pi(p, T_1, T_2)$ policy and the moment generating functions of the limiting workload $W$ in a single queue defined in definition 8,

$$
\Phi_1(\theta) = e^{-\mu T_1} \left[ \frac{\lambda}{\mu} (\Phi_2(-\mu) - \Phi_1(-\mu)) + \frac{\lambda}{\mu} (\Phi(-\mu) - \Phi_2(-\mu)) \right] e^{-\theta T_1} \Phi_X(\theta). \quad (13)
$$

This implies that for $w > T_1$, $\tilde{F}(w) = \tilde{F}(T_1) e^{-\mu(w-T_1)_+}$, where

$$
\tilde{F}(T_1) = e^{-\mu T_1} \left[ \frac{\lambda}{\mu} (\Phi_2(-\mu) - \Phi_1(-\mu)) + \frac{\lambda}{\mu} (\Phi(-\mu) - \Phi_2(-\mu)) \right]. \quad (14)
$$

**Proof:** The computation remains similar to the previous step, with an additional restriction of $W_{n+1} > T_1$. Therefore, we can write

$$
\Phi_1(\theta) = \mathbb{E}[e^{-\theta W_{n+1} \mathbb{1}_{\{W_{n+1} > T_1\}}} (\mathbb{1}_{\{W_n > T_1\}} + \mathbb{1}_{\{T_2 < W_n \leq T_1\}} + \mathbb{1}_{\{W_n \leq T_2\}})]. \quad (15)
$$

We sequentially compute the first term, the summation of first two terms, and the summation of all three terms as before. In the region $W_n > T_1$, we have $e^{-\theta W_{n+1} \mathbb{1}_{\{W_{n+1} > T_1\}}} \mathbb{1}_{\{W_n > T_1\}} = e^{-\theta (W_n - T_{n+1})} \mathbb{1}_{\{T_{n+1} < W_n < T_1\}} \mathbb{1}_{\{W_n > T_1\}}$. Then, it follows that

$$
\mathbb{E} \left[ e^{-\theta W_{n+1} \mathbb{1}_{\{W_{n+1} > T_1\}}} \mathbb{1}_{\{W_n > T_1\}} \right] = \frac{N\lambda}{N\lambda - \theta} \mathbb{E} \left[ e^{-\theta W_n} \mathbb{1}_{\{W_n > T_1\}} (1 - e^{-(N\lambda - \theta)(W_n - T_1)_+}) \right]
$$

$$
= \frac{N\lambda}{N\lambda - \theta} \left[ \Phi_1(\theta) - e^{(N\lambda - \theta)T_1} \Phi_1(N\lambda) \right].
$$

Note that, in the region $W_n \leq T_1$, it is not possible for $W_{n+1} > T_1$, unless the $n$th arrival with service time $X_n$ is admitted at the cavity queue. This occurs with probability $\frac{1}{N\lambda}$ in region $T_2 < W_n \leq T_1$, and with probability $\frac{\lambda}{N\lambda}$ in region $W_n \leq T_2$. Therefore, for the region $T_2 < W_n \leq T_1$, we can write

$$
\mathbb{E} \left[ e^{-\theta W_{n+1} \mathbb{1}_{\{W_{n+1} > T_1\}}} \mathbb{1}_{\{T_2 < W_n \leq T_1\}} \right] = \frac{\lambda e^{-(\mu + \theta)T_1}}{N\lambda - \theta} \left( \Phi_2(-\mu) - \Phi_1(-\mu) \right) (\Phi_X(\theta) - \Phi_X(N\lambda)).
$$

Similarly, for the region $W_n \leq T_2$, we can write

$$
\mathbb{E} \left[ e^{-\theta W_{n+1} \mathbb{1}_{\{W_{n+1} > T_1\}}} \mathbb{1}_{\{W_n < T_2\}} \right] = \frac{\lambda e^{-(\mu + \theta)T_1}}{N\lambda - \theta} \left( \Phi(-\mu) - \Phi_2(-\mu) \right) (\Phi_X(\theta) - \Phi_X(N\lambda)).
$$

Substituting the above three equations in equation (15) and simplifying as in the previous proof, we get

$$
\Phi_1(\theta) = \left[ \lambda (\Phi_2(-\mu) - \Phi_1(-\mu)) + \frac{\lambda}{\mu} (\Phi(-\mu) - \Phi_2(-\mu)) \right] e^{-(\mu + \theta)T_1} \Phi_X(\theta). \quad (16)
$$

The result follows by inverting the moment generating function and noting that $\Phi_1(0) = \tilde{F}(T_1)$. 

**Proposition 21.** For an $N$ server system with i.i.d. exponential service times of rate $\mu$, Poisson arrivals of rate $N\lambda$ under $\pi(p, T_1, T_2)$ policy and the moment generating functions of the limiting workload $W$ in a single queue defined in definition 8,

$$
\Phi_2(\theta) = \frac{\lambda}{\mu} (\Phi_2(\theta) - \Phi_1(\theta)) \Phi_X(\theta) + \frac{\lambda}{\mu} e^{-\mu T_2} (\Phi(-\mu) - \Phi_2(-\mu)) e^{-\theta T_2} \Phi_X(\theta). \quad (16)
$$
This implies that for \( w > T_2 \),

\[
\hat{F}(w) = \hat{F}(T_1) \left( e^{-\mu(w-T_1)^+} - \frac{\mu}{\mu - \lambda} e^{-(\mu - \lambda)(w-T_1)^+} + \left[ e^{-\mu T_2} \Phi(-\mu) - \Phi_2(-\mu) \right] \frac{\lambda}{\mu - \lambda} e^{-(\mu - \lambda)(w-T_2)^+} \right).
\]

In addition,

\[
\hat{F}(T_2) = \frac{\lambda}{\mu - \lambda} e^{-\mu T_2} \Phi(-\mu) - \Phi_2(-\mu) - \frac{\lambda}{\mu - \lambda} \hat{F}(T_1).
\] (17)

**Proof:** The computation remains similar to the previous case but here we have the restriction of \( W_{n+1} > T_2 \). Then, we can write

\[
\Phi_2(\theta) = \mathbb{E}[e^{-\theta W_{n+1}} \mathbb{1}_{\{W_{n+1} > T_2\}} \left( \mathbb{1}_{\{W_n > T_2\}} + \mathbb{1}_{\{T_2 < W_n \leq T_1\}} + \mathbb{1}_{\{W_n \leq T_2\}} \right)].
\] (18)

We sequentially compute the first term, the summation of first two terms, and the summation of all three terms as before. The indicator \( W_{n+1} > T_2 \) implies that \( W_{n+1} \) can’t be zero. In the region \( W_n > T_1 \), we have

\[
e^{-\theta W_{n+1}} \mathbb{1}_{\{W_{n+1} > T_2\}} \mathbb{1}_{\{W_n > T_1\}} = e^{-\theta(W_n - T_n+1)} \mathbb{1}_{\{T_n+1 < W_n \leq T_2\}} \mathbb{1}_{\{W_n > T_1\}}.
\]

Therefore, it follows that

\[
\mathbb{E}\left[ e^{-\theta W_{n+1}} \mathbb{1}_{\{W_{n+1} > T_2\}} \mathbb{1}_{\{W_n > T_1\}} \right] = \frac{N\lambda(\Phi_1(\theta) - e^{(N\lambda - \theta)T_2}\Phi_1(N\lambda))}{N\lambda - \theta}.
\]

Similarly, for the region \( T_2 < W_n \leq T_1 \), an external arrival is admitted with probability \( \frac{1}{N} \). When there is no arrival \( W_{n+1} = W_n - T_{n+1} \), and we have

\[
\mathbb{E}\left[ e^{-\theta(W_n - T_{n+1})} \mathbb{1}_{\{W_n - X_n > T_2\}} \mathbb{1}_{\{T_2 < W_n \leq T_1\}} \right] = \frac{N\lambda(\Phi_2(\theta) - \Phi_1(\theta) - e^{(N\lambda - \theta)T_2}(\Phi_2(N\lambda) - \Phi_1(N\lambda))}{N\lambda - \theta}.
\]

In the region \( T_2 < W_n \leq T_1 \), the \( n \)th arrival with service time \( X_n \) is admitted at the cavity queue with probability \( \frac{1}{N} \). In this case, \( W_{n+1} = W_n + X_n - T_{n+1} \), and we can write

\[
\mathbb{E}\left[ e^{-\theta(W_n + X_n - T_{n+1})} \mathbb{1}_{\{W_n + X_n - T_{n+1} > T_2\}} \mathbb{1}_{\{T_2 < W_n \leq T_1\}} \right] = \frac{N\lambda(\Phi_2(\theta) - \Phi_1(\theta))\Phi_X(\theta) - e^{(N\lambda - \theta)T_2}(\Phi_2(N\lambda) - \Phi_1(N\lambda))\Phi_X(N\lambda)}{N\lambda - \theta}.
\]

Combining these results in the region \( W_n > T_2 \), we can write

\[
\mathbb{E}\left[ e^{-\theta W_{n+1}} \mathbb{1}_{\{W_{n+1} > T_2\}} \mathbb{1}_{\{W_n > T_2\}} \right] = \frac{N\lambda(\Phi_2(\theta) - e^{(N\lambda - \theta)T_2}\Phi_2(N\lambda))}{N\lambda - \theta} + \frac{\lambda}{N\lambda - \theta} \left[ (\Phi_2(\theta) - \Phi_1(\theta))(\Phi_X(\theta) - 1) - e^{(N\lambda - \theta)T_2}(\Phi_2(N\lambda) - \Phi_1(N\lambda))(\Phi_X(N\lambda) - 1) \right].
\]

In the region \( W_n \leq T_2 \), it’s not possible for \( W_{n+1} > T_2 \), unless the \( n \) arrival with service time \( X_n \) is admitted at the cavity queue. This occurs with probability \( \frac{1}{N} \), and we can write

\[
\mathbb{E}\left[ e^{-\theta W_{n+1}} \mathbb{1}_{\{W_{n+1} > T_1\}} \mathbb{1}_{\{W_n \leq T_2\}} \right] = \frac{\lambda e^{-(\mu+\theta)T_2}}{N\lambda - \theta} (\Phi(-\mu) - \Phi_2(-\mu))(\Phi_X(\theta) - \Phi_X(N\lambda)).
\]

Combining the above equations and simplifying as in the previous proof, we obtain

\[
\Phi_2(\theta) = \frac{\lambda}{\mu} (\Phi_2(\theta) - \Phi_1(\theta))\Phi_X(\theta) + \frac{\lambda}{\mu} e^{-\mu T_2} (\Phi(-\mu) - \Phi_2(-\mu)) e^{-\theta T_2} \Phi_X(\theta).
\] (19)

To prove the second statement, note that \( \Phi_1(\theta) = \hat{F}(T_1)e^{-\theta T_1}\Phi_X(\theta) \) from equation (13). Substitution and simplification tells us that

\[
\Phi_2(\theta) = \left( \frac{1}{\mu + \theta} - \frac{1}{\mu - \lambda + \theta} \right) \mu F(T_1)e^{-\theta T_1} + \frac{\lambda e^{-\theta T_2}}{\mu - \lambda + \theta} e^{-\mu T_2} (\Phi(-\mu) - \Phi_2(-\mu))
\]

when service times are exponentially distributed with rate \( \mu \). The result follows by inverting this moment generating function and the fact that \( \Phi_2(0) = \hat{F}(T_2) \).