Minimizing the Age of the Information through Queues

Ahmed M. Bedewy[†], Yin Sun[†], and Ness B. Shroff[†‡]

[†]Dept. of ECE, [‡]Dept. of CSE, The Ohio State University, Columbus, OH.
emails: bedewy.2@osu.edu, sunyin02@gmail.com, shroff.11@osu.edu

Abstract

In this paper, we focus on developing simple scheduling policies that can minimize the age of the information sent through multi-server queueing systems. We consider a general packet arrival process, where the generation times and arrival times of the packets are arbitrarily given. Hence, the packets may arrive in an order that is different from the order of their generation times. Further, a packet can be replicated on multiple servers, and one can specify a priori the maximum number of replicas that can be created for a packet. Once a replica is completed, the remaining replicas of this packet are canceled to release the servers. We show that simple variants (e.g., preemptive, non-premptive, replicative, non-replicative) of the Last-Generated, First-Served (LGFS) scheduling policy is age-optimal in a stochastic ordering sense for exponentially distributed packet service times. These policies are optimal for minimizing not only the age process, but also for minimizing any non-decreasing functional of the age process. In addition, we investigate the class of New-Better-than-Used (NBU) service time distributions and develop scheduling policies that are shown to be within a constant gap from the optimum age performance. Somewhat to our surprise, we find that packet replication can reduce the age under some NBU service time distributions, even if it worsens the throughput and delay performance in the meanwhile.

This paper was presented in part at IEEE ISIT 2016 [1].

This work has been supported in part by ONR grants N00014-17-1-2417 and N00014-15-1-2166, Army Research Office grants W911NF-14-1-0368 and MURI W911NF-12-1-0385, National Science Foundation grants CNS-1446582, CNS-1421576, and CNS-1518829, and a grant from the Defense Thrust Reduction Agency HDTRA1-14-1-0058.

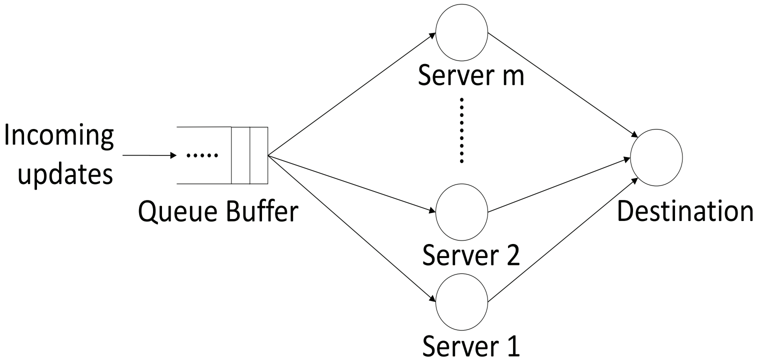


Figure 1: System model.

I. Introduction

The ubiquity of mobile devices and applications has greatly boosted the demand for real-time information updates, such as news, weather reports, email notifications, stock quotes, social updates, mobile ads, etc. Also, timely status updates are crucial in networked monitoring and control systems. These include, but are not limited to, sensor networks used to measure temperature or other physical phenomena, and surround monitoring in autonomous driving.

A common need in these real-time applications is to keep the destination (i.e., information consumer) updated with the freshest information. To identify the timeliness of the updates, a metric called the *age of information*, or simply *age*, was defined in, e.g., [2]–[5]. At time t, if the freshest received update at the destination was generated at time $U(t)$, the age $\Delta(t)$ is defined as

$$\Delta(t) = t - U(t). \tag{1}$$

Hence, age is the time elapsed since the freshest received packet was generated.

In recent years, a variety of approaches have been investigated to reduce the age. In [5]–[7], it was found in First-Come, First-Served (FCFS) queueing systems that the time-average age first decreases with the update frequency and then increases with the update frequency. The optimal update frequency was obtained to minimize the age in FCFS systems. In [8]–[10], it was shown that the age can be further improved by discarding old packets waiting in the queue when a new sample arrives. Characterizing the age in Last-Come, First-Served (LCFS) queueing systems with gamma distributed service times was considered in [11]. However, these studies cannot tell us (i) which queueing discipline can minimize the age and (ii) under what conditions the minimum age is achievable.

In this paper, we answer these two questions for an information-update system illustrated in Fig. 1, where a sequence of update packets arrive at a queue with m servers and a buffer size B. Each server can be used to model a channel in multi-channel communication systems [12], or a computer in parallel computing systems [13]. The service times of the update packets are *i.i.d.* across servers and time. Let s_i be the generation time of packet i at an external source, and a_i be the arrival time of packet i at the queue. Out-of-order packet arrivals are allowed, such that the packets may arrive in an order different from their generation times, i.e., $s_i < s_j$ but $a_i > a_j$. Packet replication [14]–[16] is considered in this study. In particular, multiple replicas of a packet can be assigned to different servers, at possibly different service starting time epochs. The first completed replica is considered as the valid execution of the packet; after that, the remaining replicas of this packet are canceled immediately to release the servers. Suppose that a packet can be replicated on at most r servers ($r \leq m$), where r is called the maximum replication degree. If $r = 1$, this reduces to the case where replication is not allowed at all. The following are the key contributions of this paper:

- If the packet service times are *i.i.d.* exponentially distributed, then for *arbitrary* system parameters (including *arbitrary* packet generation times s_i, packet arrival times a_i, number of servers m, maximum replication degree r, and buffer size B), we prove that the preemptive Last-Generated, First-Served with replication (prmp-LGFS-R) policy minimizes the age process among all causal policies in a stochastic ordering sense (Theorem 1). This further implies that the prmp-LGFS-R policy minimizes any non-decreasing functional of the age process in a stochastic ordering sense. Note that this age penalty model is very general. Many age penalty metrics studied in the literature, such as the time-average age [5], [6], [8]–[11], [17]–[22], average peak age [7]–[9], [11], [21], [23], and time-average age penalty function [24], [25], are special cases of this age penalty model.
- We further investigate a more general class of packet service time distributions called New-Better-than-Used (NBU) distributions. We show that the non-preemptive Last-Generated, First-Served with replication (non-prmp-LGFS-R) policy is within a constant age gap from the optimum average age, and the gap is independent of the system parameters mentioned above (Theorem 4). Note that policy non-prmp-LGFS-R with a maximum replication degree r can be near age-optimal compared with policies with any maximum replication degree. This result is not anticipated: In [16], [26], [27], it was shown that non-replication policies are near delay-optimal and replication policies are far from the optimum delay and throughput performance for NBU service time distributions. From these studies, one would expect that replications may worsen the age performance. To our surprise, we found in this paper that a replicative policy (i.e., non-prmp-LGFS-R) is near-optimal in minimizing the age, even for NBU service time distributions.
- Finally, we investigate the throughput and delay performance of the proposed policies. We show that if the packet service times are i.i.d. exponentially distributed, then the prmp-LGFS-R policy is also throughput and delay optimal among all causal policies (Theorem 5). In addition, if the packet service times are i.i.d. NBU and replications are not allowed, then the non-prmp-LGFS policy is throughput and delay optimal among all non-preemptive causal policies (Theorem 6).

To the best of our knowledge, these are the first optimality results on minimizing the age-of-information in queueing systems for given generation and arrival times of the update packets.

The remainder of this paper is organized as follows. After a brief overview of related work in Section II, we present the model and problem formulation in Section III. The age performance of the proposed policies is analyzed in Section IV, and their throughput and delay performances are investigated in Section V. Finally, the conclusion is drawn in Section VI.

II. Related Work

A series of works studied the age performance of scheduling policies in a single queueing system with Poisson arrival process and exponential service time [5], [6], [8]–[10], [17]–[20]. In [5], [6], the update frequency was optimized to improve data freshness in FCFS information-update systems. The effect of the packet management on the age was considered in [8]–[10]. It was found that a good policy is to discard the old updates waiting in the queue when a new sample arrives, which can greatly reduce the impact of queueing delay on data freshness. In [17], the time-average age was characterized for single source Last-Come, First-Served (LCFS) information-update systems with and without preemption. Expanding the analysis to multiple sources was considered in [18]. In these studies, it was shown that sharing service facility among Poisson sources can improve the total age. Characterizing the time average age for FCFS queueing system with two and infinite number of servers was studied in [19]. The analysis in [19] showed that the model with infinite servers has a lower age in conjunction with more wasting in network resources due to the rise in the obsolete delivered packets. In [20], the average age was characterized in pull model. In this model, a customer sends requests to the servers to retrieve (pull) the interested information, where the servers carry information with different freshness levels. Request replication, where a user sends requests to all servers, was considered in this study to minimize the average age at the user's side.

Characterizing the age for a class of packet service time distributions that are more general than exponential was considered in [7], [11], [23]. In [7], the age was analyzed in multi-class M/G/1 and M/G/1/1 queues. The age performance in the presence of errors when the service times are exponentially distributed was analyzed in [23]. Gamma-distributed service times was considered in [11]. The studies in [11], [23] were carried out for LCFS queueing systems with and without preemption.

In [28], age-optimality was proven to be achievable in multihop networks with arbitrary packet generation times, packet arrival times, and general network topologies. It was shown that the LGFS policy is age-optimal among all causal policies for exponential packet service times. In addition, for arbitrary distributions of packet service times, it was shown that the LGFS policy is age-optimal among all non-preemptive work-conserving policies.

Another line of research is the joint optimization of packet generation and transmissions in [21], [22], [24], [25]. An information update policy was developed in [24], [25], which was proven to minimize a general class of non-negative, non-decreasing age penalty functions among all causally feasible policies. In this setting [22], [24], [25], a counter-intuitive phenomenon was revealed: While a zero-wait or work-conserving policy, that generates and submits a fresh update once the server becomes idle, achieves the maximum throughput and the minimum average delay, surprisingly, this zero-wait policy does not always minimize the age. More recently, a real-time sampling problem of the Wiener process is solved in [29]: If the sampling times are independent of the observed Wiener process, the optimal sampling problem in [29] reduces to an age-of-information optimization problem; otherwise, the optimal sampling policy can use knowledge of the Wiener process to achieve better performance than age-of-information optimization.

III. Model and Formulation

A. Notations and Definitions

For any random variable Z and an event A, let $[Z|A]$ denote a random variable with the conditional distribution of Z for given A, and $\mathbb{E}[Z|A]$ denote the conditional expectation of Z for given A.

Let $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ and $\mathbf{y} = (y_1, y_2, \ldots, y_n)$ be two vectors in $\mathbb{R}^n$, then we denote $\mathbf{x} \leq \mathbf{y}$ if $x_i \leq y_i$ for $i = 1, 2, \ldots, n$. A set $U \subseteq \mathbb{R}^n$ is called upper if $\mathbf{y} \in U$ whenever $\mathbf{y} \geq \mathbf{x}$ and $\mathbf{x} \in U$. We will need the following definitions:

Definition 1. Univariate Stochastic Ordering: [30] Let X and Y be two random variables. Then, X is said to be stochastically smaller than Y (denoted as $X \leq_{\text{st}} Y$), if

$$\mathbb{P}\{X > x\} \leq \mathbb{P}\{Y > x\}, \quad \forall x \in \mathbb{R}.$$

Definition 2. Multivariate Stochastic Ordering: [30] Let $\mathbf{X}$ and $\mathbf{Y}$ be two random vectors. Then, $\mathbf{X}$ is said to be stochastically smaller than $\mathbf{Y}$ (denoted as $\mathbf{X} \leq_{\text{st}} \mathbf{Y}$), if

$$\mathbb{P}\{\mathbf{X} \in U\} \leq \mathbb{P}\{\mathbf{Y} \in U\}, \quad \text{for all upper sets} \quad U \subseteq \mathbb{R}^n.$$

Definition 3. Stochastic Ordering of Stochastic Processes: [30] Let $\{X(t), t \in [0, \infty)\}$ and $\{Y(t), t \in [0, \infty)\}$ be two stochastic processes. Then, $\{X(t), t \in [0, \infty)\}$ is said to be stochastically smaller than $\{Y(t), t \in [0, \infty)\}$ (denoted by $\{X(t), t \in [0, \infty)\} \leq_{\text{st}} \{Y(t), t \in [0, \infty)\}$), if, for all choices of an integer n and $t_1 < t_2 < \ldots < t_n$ in $[0, \infty)$, it holds that

$$(X(t_1), X(t_2), \ldots, X(t_n)) \leq_{\text{st}} (Y(t_1), Y(t_2), \ldots, Y(t_n)), \tag{2}$$

where the multivariate stochastic ordering in (2) was defined in Definition 2.

B. Queueing System Model

We consider a queueing system with m servers as shown in Fig. 1. The update packets are generated exogenously to the system and then arrive at the queue. The system starts to operate at time $t = 0$. A sequence of n update packets arrive at the system at time epochs $a_1, \ldots, a_n$, where n can be an arbitrary finite or infinite number and $0 \leq a_1 \leq a_2 \leq \ldots \leq a_n$. The i-th arrived packet, called packet i, is generated at time s_i such that $0 \leq s_i \leq a_i$ for all $i = 1, \ldots, n$. Note that the update packets may arrive at the system *out of the order* of their generation times. For example, in Fig. 2, we have $s_1 > s_2$ but $a_1 < a_2$. Let B denote the buffer size of the queue which can be infinite, finite, or even zero. If B is finite, the queue buffer may overflow and some packets are dropped. The packet service times are *i.i.d.* across time and servers, and are independent of the packet generation and arrival processes. Packet replication is considered in this model, where the maximum replication degree is r ($r \leq m$). In this model, one packet can be replicated to at most r servers and the first completed replica is considered as the valid execution of the packet. After that, the remaining replicas of this packet are cancelled immediately to release the servers.

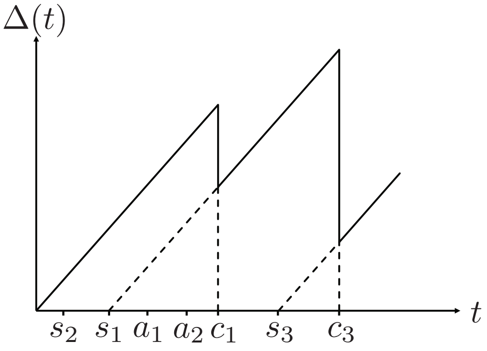


Figure 2: Sample path of the age process $\Delta(t)$.

C. Scheduling Policy

A scheduling policy, denoted by π, determines the packet assignments and replications in the system. Define c_i as the service completion time of packet i (the earliest time that a copy of packet i has left the server), which is a function of the scheduling policy π. The packet generation times $(s_1, s_2, \ldots, s_n)$ and packet arrival times $(a_1, a_2, \ldots, a_n)$ at the system are arbitrary given, which do not change according to the scheduling policy. We also assume that the packet service times are invariant of the scheduling policy.

Define Π_r as the set of all *causal* policies, in which scheduling decisions are made based on the history and current state of the system, when the maximum replication degree is r. Hence, $\Pi_1 \subset \Pi_2 \subset \ldots \subset \Pi_m$. We define several types of policies in Π_r:

A policy is said to be **preemptive**, if a server can switch to send any packet at any time; the preempted packets will be stored back into the queue if there is enough buffer space and sent at a later time when the servers are available again. In contrast, in a **non-preemptive** policy, processing of a packet cannot be interrupted until the packet is completed or cancelled; after completing or cancelling a packet, the server can switch to send another packet. A policy is said to be **work-conserving**, if no server is idle whenever there are packets waiting in the queue.

D. Age Performance Metric

Let $U(t) = \max\{s_i : c_i \leq t\}$ be the generation time of the freshest packet at the destination at time t. The *age-of-information*, or simply the *age*, is defined as [2]–[5]

$$\Delta(t) = t - U(t). \tag{3}$$

The initial state $U(0^-)$ at time $t = 0^-$ is invariant of the policy $\pi \in \Pi_r$, where we assume that $s_0 = U(0^-) = 0$. As shown in Fig. 2, the age increases linearly with t but is reset to a smaller value with the arrival of a fresher packet. The age process is given by

$$\Delta = \{\Delta(t), t \in [0, \infty)\}. \tag{4}$$

In this paper, we introduce a non-decreasing *age penalty functional* $g(\Delta)$ to represent the level of dissatisfaction for data staleness at the receiver or destination.

Definition 4. Age Penalty Functional: Let $\mathbf{V}$ be the set of N-dimensional Lebesgue measurable functions, i.e.,

$$\mathbf{V} = \{f : [0, \infty)^N \mapsto \mathbb{R} \text{ such that } f \text{ is Lebesgue measurable}\}.$$

A functional $g : \mathbf{V} \mapsto \mathbb{R}$ is said to be an *age penalty functional* if g is *non-decreasing* in the following sense:

$$g(\Delta_1) \leq g(\Delta_2), \text{ whenever } \Delta_1(t) \leq \Delta_2(t), \forall t \in [0, \infty). \tag{5}$$

The age penalty functionals used in prior studies include:

- *Time-average age [5], [6], [8]–[11], [17]–[22]:* The time-average age is defined as

$$g_1(\Delta) = \frac{1}{T} \int_0^T \Delta(t) dt, \tag{6}$$

- *Average peak age [7]–[9], [11], [21], [23]:* The average peak is defined as

$$g_2(\Delta) = \frac{1}{K} \sum_{k=1}^{K} A_k, \tag{7}$$

  where A_k denotes the k-th peak value of $\Delta(t)$ since time $t = 0$.
- *Time-average age penalty function [24], [25]:* The average age penalty function is

$$g_3(\Delta) = \frac{1}{T} \int_0^T h(\Delta(t)) dt, \tag{8}$$

  where $h : [0, \infty) \rightarrow [0, \infty)$ can be any non-negative and non-decreasing function. As pointed out in [25], a stair-shape function $h(\Delta) = \lfloor \Delta \rfloor$ can be used to characterize the dissatisfaction of data staleness when the information of interests is checked periodically, and an exponential function $h(\Delta) = e^{\Delta}$ is appropriate for online learning and control applications where the desire for data refreshing grows quickly with respect to the age.

IV. Age-Optimality Results of LGFS Policies

In this section, we provide age-optimality and near age-optimality results for multi-server queueing networks with packet replication. We start by considering the exponential packet service time distribution and show that age-optimality can be achieved. Then, we consider the classes of NBU packet service time distributions and show that there exist simple policies that can come close to age-optimality.

A. Exponential Service Time Distribution

We study the age-optimal packet scheduling when the packet service times are exponentially distributed, and *i.i.d.* across time and servers. We propose a policy called **preemptive Last-Generated, First-Served**

Algorithm 1: Preemptive Last-Generated, First-Served with replication policy.

```
$\alpha := 0$;                                   // $\alpha$ is the smallest generation time of the packets under service
$Q := \emptyset$;                                // $Q$ is the set the packets under service
$k := \lfloor \frac{m}{r} \rfloor$;              // $k$ is number of packets that can be replicated on $r$ servers
while the system is ON do
    if a new packet $p_i$ with generation time $s$ arrives then
        if all servers are busy then
            if $s \leq \alpha$ then
                Store packet $p_i$ in the queue;
            else                                 // Packet $p_i$ carries fresh information.
                Find packet $p_j \in Q$ with generation time $\alpha$;
                Preempt all replicas of packet $p_j$;
                Packet $p_j$ is stored back to the queue;
                $Q := Q \cup \{p_i\} - \{p_j\}$;
            end
        else                                     // At least one of the servers is idle.
            $Q := Q \cup \{p_i\}$;
        end
        $\alpha := \min\{s_i : i \in Q\}$;
        Replicate the $k$ freshest packets in $Q$ such that each of these packets is replicated on $r$ servers;
        Replicate the packet with generation time $\alpha$ on the remaining $(m - kr)$ servers;
    end
    if a packet $p_l$ is delivered then
        Cancel the remaining replicas of packet $p_l$;
        $Q := Q - \{p_l\}$;
        if the queue is not empty then
            Pick the freshest packet in the queue $p_h$;
            $Q := Q \cup \{p_h\}$;
        end
        $\alpha := \min\{s_i : i \in Q\}$;
        Replicate the $k$ freshest packets in $Q$ such that each of these packets is replicated on $r$ servers;
        Replicate the packet with generation time $\alpha$ on the remaining $(m - kr)$ servers;
    end
end
```

with replication (prmp-LGFS-R). This policy follows Last-Generated, First-Served discipline, which is defined as follows.

Definition 5. A scheduling policy is said to follow the **Last-Generated, First-Served (LGFS)** discipline, if the packets under service are generated the latest (i.e., the freshest) among all packets in the queue; after service, the next freshest packet in the queue is assigned to the idle server.

The implementation details of prmp-LGFS-R policy are depicted in Algorithm 1. Define a set of parameters $\mathcal{I} = \{n, (s_i, a_i)_{i=1}^{n}, B, m, r\}$, where n is the total number of packets, s_i is the generation time of packet i, a_i is the arrival time of packet i, B is the queue buffer size, m is the number of servers, and r is the maximum replication degree. Let $\Delta_\pi = \{\Delta_\pi(t), t \in [0, \infty)\}$ be the age processes under policy π. The age performance of the prmp-LGFS-R policy is characterized as follows.

Theorem 1. *Suppose that the packet service times are exponentially distributed and i.i.d. across time and servers, then for all $\mathcal{I}$ and $\pi \in \Pi_r$*

$$[\Delta_{\text{prmp-LGFS-R}} | \mathcal{I}] \leq_{\text{st}} [\Delta_\pi | \mathcal{I}], \tag{9}$$

or equivalently, for all $\mathcal{I}$ and non-decreasing functional g

$$\mathbb{E}[g(\Delta_{\text{prmp-LGFS-R}})|\mathcal{I}] = \min_{\pi\in\Pi_r} \mathbb{E}[g(\Delta_\pi)|\mathcal{I}], \tag{10}$$

provided the expectations in (10) exist.

Proof. See Appendix A. □

Theorem 1 tells us that for arbitrary number n, packet generation times $(s_1, s_2, \ldots, s_n)$, packet arrival times $(a_1, a_2, \ldots, a_n)$, buffer size B, number of servers m, and maximum replication degree r, the prmp-LGFS-R policy achieves optimality of the age process within the policy space Π_r. In addition, (10) tells us that the prmp-LGFS-R policy minimizes any *non-decreasing functional* of the age process, including the time-average age (6), average peak age (7), and time-average age penalty function (8) as special cases.

As a result of Theorem 1, we can deduce the following corollary:

Corollary 2. Suppose that the packet service times are exponentially distributed and *i.i.d.* across time and servers, then for all $\mathcal{I}$, the age performance of the prmp-LGFS-R policy remains the same for any queue size $B \geq 0$.

Proof. From the operation of policy prmp-LGFS-R, its queue is used to store the preempted packets and outdated arrived packets. The age process of the prmp-LGFS-R policy is not affected no matter these packets are delivered or not. Hence, the age performance of the prmp-LGFS-R policy is invariant for any queue size $B \geq 0$. This completes the proof. □

1) Simulation Results: We present some simulation results to compare the age performance of the prmp-LGFS-R policy with other policies. The packet service times are exponentially distributed with mean $1/\mu = 1$. The inter-generation times are *i.i.d.* Erlang-2 distribution with mean $1/\lambda$. The number of servers is m. Hence, the traffic intensity is $\rho = \lambda/m\mu$. [1] The queue size is B, which is a non-negative integer.

Figure 3 illustrates the time-average age versus ρ for an information-update system with $m = 1$ server. The time difference $(a_i - s_i)$ between packet generation and arrival is zero, i.e., the update packets arrive in the same order of their generation times. We can observe that the prmp-LGFS-R policy achieves a smaller age than the FCFS policy analyzed in [5], and the non-preemptive LCFS policy with queue size $B = 1$ [17] which was also named "M/M/1/2*" in [8], [9]. Note that in these prior studies, the age was characterized only for the special case of Poisson arrival process. Moreover, with ordered arrived packets at the server, the LGFS policy and LCFS policy have the same age performance.

Figure 4 plots the average peak age versus ρ for an information-update system with $m = 4$ servers. The time difference between packet generation and arrival, i.e., $a_i - s_i$, is modeled to be either 1 or 100, with equal probability. The maximum replication degree r is either 1, 2, or 4. For each r, we found that the prmp-LGFS-R policy achieves better age performance than other policies that belong to the policy

[1] Throughout this paper, the traffic intensity ρ is computed without considering replications (i.e., ρ is calculated when $r = 1$).

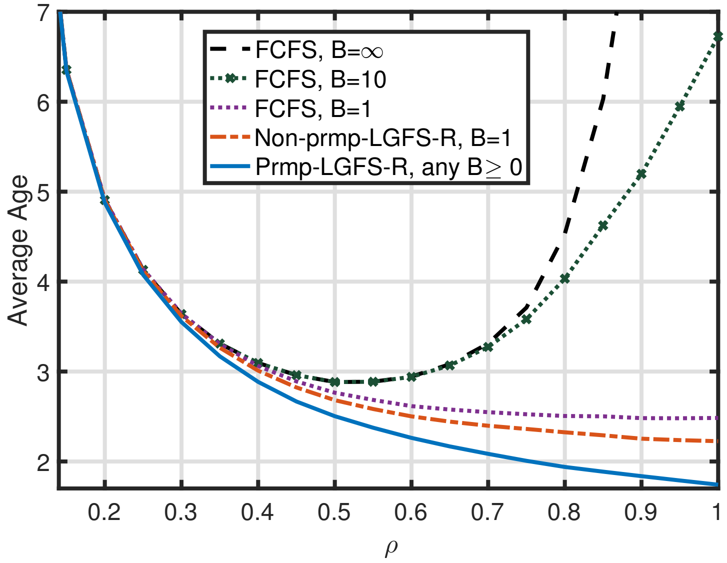


Figure 3: Average age versus traffic intensity ρ for an update system with $m = 1$ server, queue size B, and *i.i.d.* exponential service times.

space Π_r. For example, the age performance of the prmp-LGFS-R policy when $r = 2$ is better than the age performance of the other policies that are plotted when r equal to 1 and 2. Note that the age performance of the prmp-LGFS-R policy remains the same for any queue size $B \geq 0$. However, the age performance of the non-prmp-LGFS-R policy and FCFS policy varies with the queue size B. We also observe that the average peak age in case of FCFS policy with $B = \infty$ blows up when the traffic intensity is high. This is due to the increased congestion in the network which leads to a delivery of stale packets. Moreover, in case of FCFS policy with $B = 10$, the average peak age is high but bounded at high traffic intensity, since the fresh packet has a better opportunity to be delivered in a relatively short period compared with FCFS policy with $B = \infty$. These numerical result agrees with Theorem 1.

B. NBU Service Time Distributions

The next question we proceed to answer is whether for an important class of distributions that are more general than exponential, age-optimality or near age-optimality can be achieved. We consider the class of NBU packet service time distributions, which are defined as follows.

Definition 6. New-Better-than-Used distributions: Consider a non-negative random variable Z with complementary cumulative distribution function (CCDF) $\bar{F}(z) = \mathbb{P}[Z > z]$. Then, Z is **New-Better-than-Used (NBU)** if for all $t, \tau \geq 0$

$$\bar{F}(\tau + t) \leq \bar{F}(\tau)\bar{F}(t). \tag{11}$$

Examples of NBU distributions include constant service time, (shifted) exponential distribution, geometrical distribution, Erlang distribution, negative binomial distribution, etc.

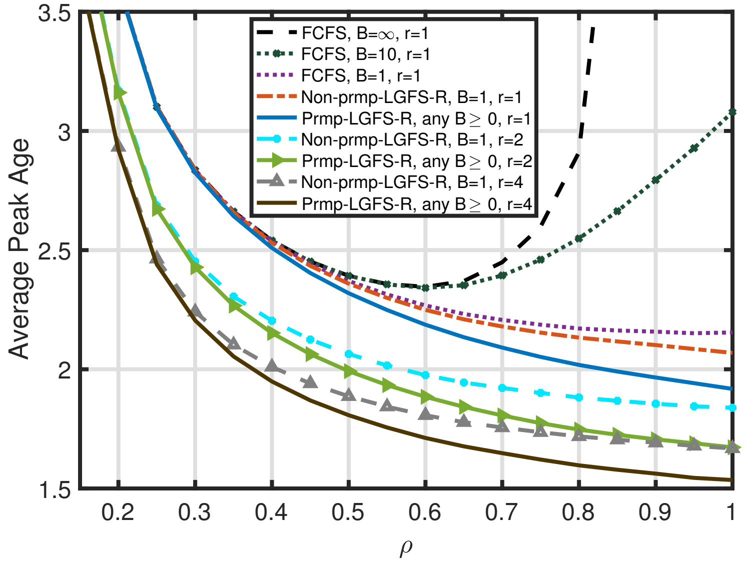


Figure 4: Average peak age versus traffic intensity ρ for an update system with $m = 4$ servers, queue size B, maximum replication degree r, and *i.i.d.* exponential service times.

Algorithm 2: Non-preemptive Last-Generated, First-Served with replication policy.

```
$k := \lfloor \frac{m}{r} \rfloor$;        // $k$ is number of packets that can be replicated on $r$ servers
while the system is ON do
    if a new packet $p_i$ arrives then
        if all servers are busy then
            Store packet $p_i$ in the queue;
        else                                // At least one of the servers is idle.
            Packet $p_i$ is replicated on at most $r$ servers;
        end
    end
    if a packet $p_l$ is delivered then
        Cancel the remaining replicas of packet $p_l$;
        Find packet $p_j$ that is replicated on $(m - kr)$ servers;
        if the queue is empty then
            Packet $p_j$ is replicated on extra $((k+1)r - m)$ servers;
        else
            Pick the freshest packet in the queue $p_h$;
            if packet $p_j$ exists and generation time of packet $p_j$ > generation time of packet $p_h$ then
                Packet $p_j$ is replicated on extra $((k+1)r - m)$ servers;
                Packet $p_h$ is replicated on $(m - kr)$ servers;
            else // Packet $p_j$ is staler than packet $p_h$ or there is no packet that is replicated on $(m - kr)$ servers.
                Packet $p_h$ is replicated on at most $r$ servers;
            end
        end
    end
end
```

Next, we show that near age-optimality can be achieved when the service times are NBU. We propose a policy called non-preemptive LGFS with replication (non-prmp-LGFS-R). The description of the non-prmp-LGFS-R policy is depicted in Algorithm 2. It is important to note that under non-prmp-LGFS-R policy, the fresh packet replaces the oldest packet in the queue when it has a finite buffer size and full. To

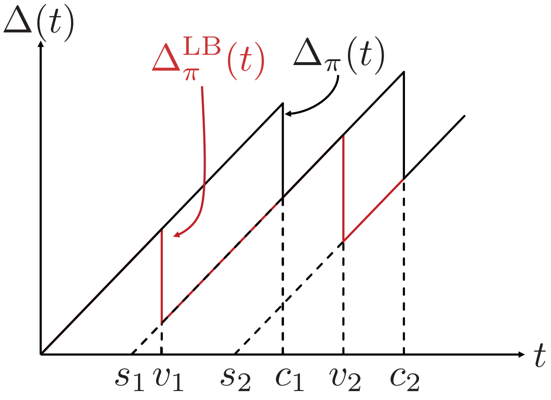


Figure 5: The evolution of Δ_π^{LB} and Δ_π in a single server queue.

show that policy non-prmp-LGFS-R can come close to age-optimal, we need to construct an age lower bound as follows:

Let v_i denote the earliest time that packet i has started service (the earliest assignment time of packet i to a server), which is a function of the scheduling policy π. Define a function $\Delta_\pi^{\text{LB}}(t)$ as

$$\Delta_\pi^{\text{LB}}(t) = t - \max\{s_i : v_i(\pi) \leq t\}. \tag{12}$$

The process of $\Delta_\pi^{\text{LB}}(t)$ is given by $\Delta_\pi^{\text{LB}} = \{\Delta_\pi^{\text{LB}}(t), t \in [0, \infty)\}$. The definition of the process $\Delta_\pi^{\text{LB}}(t)$ is similar to that of the age process of policy π except that the packets completion times are replaced by their assignment times to the servers. In this case, the process $\Delta_\pi^{\text{LB}}(t)$ increases linearly with t but is reset to a smaller value with the assignment of a fresher packet to a server under policy π, as shown in Fig. 5. The process $\Delta_{\text{non-prmp-LGFS-R}}^{\text{LB}}$ is a lower bound of all policies in Π_m in the following sense.

Lemma 3. Suppose that the packet service times are NBU and *i.i.d.* across time and servers, then for all $\mathcal{I}$ satisfying $B \geq 1$, and $\pi \in \Pi_m$

$$[\Delta_{\text{non-prmp-LGFS-R}}^{\text{LB}} | \mathcal{I}] \leq_{\text{st}} [\Delta_\pi | \mathcal{I}]. \tag{13}$$

Proof. See Appendix B. □

We can now proceed to characterize the age performance of policy non-prmp-LGFS-R. Let $X_1, \ldots, X_m$ denote the *i.i.d.* packet service times of the m servers, with mean $E[X_l] = E[X] < \infty$. We use Lemma 3 to prove the following theorem.

Theorem 4. Suppose that the packet service times are NBU and *i.i.d.* across time and servers, then for all $\mathcal{I}$ satisfying $B \geq 1$

(a)
$$\min_{\pi \in \Pi_m} [\bar{\Delta}_\pi | \mathcal{I}] \leq [\bar{\Delta}_{\text{non-prmp-LGFS-R}} | \mathcal{I}] \leq \min_{\pi \in \Pi_m} [\bar{\Delta}_\pi | \mathcal{I}] + \mathbb{E}[X]. \tag{14}$$

(b) If m is a multiple of r such that $m \bmod r = 0$

$$\min_{\pi \in \Pi_m} [\bar{\Delta}_\pi | \mathcal{I}] \leq [\bar{\Delta}_{\text{non-prmp-LGFS-R}} | \mathcal{I}] \leq \min_{\pi \in \Pi_m} [\bar{\Delta}_\pi | \mathcal{I}] + \mathbb{E}[\min_{l=1,\ldots,r} X_l], \tag{15}$$

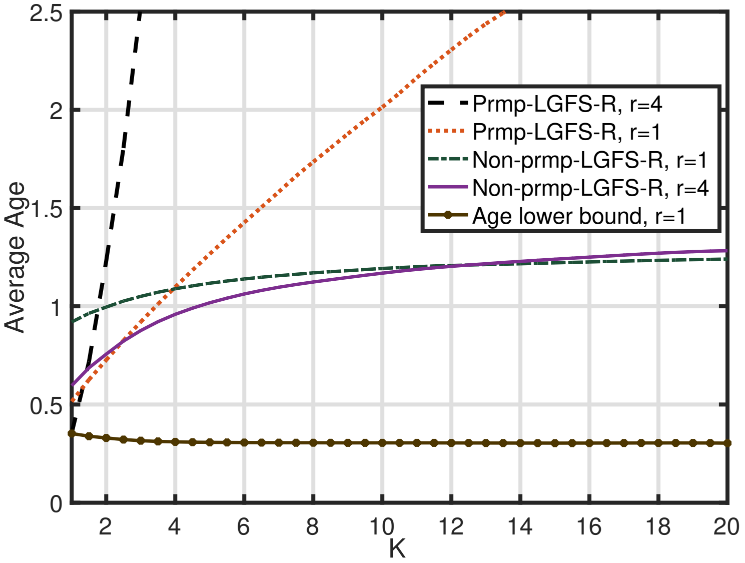


Figure 6: Average age under gamma service time distributions with different shape parameter K, where $m = 4$ servers, queue size $B = \infty$, and maximum replication degree r.

where $\bar{\Delta}_\pi = \limsup_{T\to\infty} \frac{\mathbb{E}[\int_0^T \Delta_\pi(t)dt]}{T}$ is the average age under policy π.

Proof. See Appendix C. □

Theorem 4 tells us that for arbitrary number n, packet generation times $(s_1, s_2, \ldots, s_n)$, arrival times $(a_1, a_2, \ldots, a_n)$, number of servers m, maximum replication degree r, and buffer size $B \geq 1$, the non-prmp-LGFS-R policy is within a constant age gap from the optimum average age among policies in Π_m. It is important to emphasis that policy non-prmp-LGFS-R with a maximum replication degree r can be near age-optimal compared with policies with any maximum replication degree.

1) Simulation Results: We now provide a simulation result to illustrate the age performance of different policies when the service times are NBU. Figure 6 plots the average age under gamma service time distributions with different shape parameter K, where $m = 4$, $B = \infty$, and the traffic intensity $\rho = \lambda/m\mu = 1.8$. The mean of the gamma service time distributions are normalized to $1/\mu = 1$. The inter-generation times are *i.i.d.* Erlang-2 distribution with mean $1/\lambda$. The time difference $(a_i - s_i)$ between packet generation and arrival is zero. The maximum replication degree r is either 1 or 4. Note that the average age of the FCFS policy in this case is extremely high and hence is not plotted in this figure. The "Age lower bound" curve is generated by using $\frac{\int_0^T \Delta^{\text{LB}}_{\text{non-prmp-LGFS-R}}(t)dt}{T}$ when $B = \infty$ and $r = 1$ which, according to Lemma 3, is a lower bound of the optimum average age. We can observe that the gap between the "Age lower bound" curve and the average age of the non-prmp-LGFS-R policy when $r = 1$ is no larger than $E[X] = 1/\mu = 1$, which agrees with Theorem 4. In addition, one can notice that packet replication can improve the age performance. In particular, we found that the non-prmp-LGFS-R policy when $r = 4$ has the best age performance among all plotted policies when the shape parameter K is between 2.5 and 12.5. This is a surprising result since it was shown in [16], [26], [27] that replication policies are far from the optimum delay and throughput performance for NBU service time distributions.

In the limit scenario $K = \infty$, the gamma distributed service time is $X = 1$ with probability one. In this case, replication is not helpful, but the non-prmp-LGFS-R policy is still within a small gap from age optimality. Furthermore, we can observe that the average age of the non-prmp-LGFS-R policy increases as the shape parameter K increases. An explanation of this observation is that as the shape parameter K increases, the variance (variability) of the normalized gamma distribution decreases. This, in turn, reduces the benefit gained from the diversity provided by four servers and hence increases the average age. Finally, we can observe that prmp-LGFS-R policy achieves the best age performance among all plotted policies when $K = 1$. This is because a gamma distribution with shape parameter $K = 1$ is an exponential distribution. Thus, age-optimality can be achieved in this case by policy prmp-LGFS-R as stated in Theorem 1. However, as can be seen in the figure, the average age of the prmp-LGFS-R policy blows up as the shape parameter K increases. The reason of this phenomenon is as follows: As K increases, the variance (variability) of normalized gamma distribution decreases. Hence, when a packet is preempted, the service time of a new packet is probably longer than the remaining service time of the preempted packet. Because the arrival rate is high, packet preemption happen frequently, which leads to infrequent packet delivery and increase the age, as observed in [8].

C. Discussion

In this subsection, we discuss our results and compare it with prior works.

1) Preemption vs. Non-Preemption: The effect of the preemption on the age performance depends basically on the distribution of the packet service time. More specifically, when the packet service times are exponentially distributed, preemptive policies (i.e., prmp-LGFS-R) can achieve age-optimality (Theorem 1). This is because the remaining service time of a preempted packet has the same distribution with the service time of a new packet. Thus, we suggest using preemption when the packet service times are exponentially distributed. However, when the packet service times are NBU, we suggest to not use preemption. This is because the service times are no longer memoryless. Hence, when a packet is preempted, the service time of a new packet is probably longer than the remaining service time of the preempted packet. As shown in Fig. 6, the age of the preemptive LCFS policy grows to infinity at high traffic intensity for gamma distributed service times. Thus, we suggest using non-preemptive policies (i.e., non-prmp-LGFS-R) instead when the packet service times are NBU.

Similar observations have been made in previous studies [11], [18]. For exponential service time distribution, Yates and Kaul showed in Theorem 3(a) of [18] that the average age of the preemptive LCFS policy is a decreasing function of the traffic intensity ρ in M/M/1 queues as ρ grows to infinite. This agrees with our study, in which we proved that the preemptive LCFS policy is age-optimal for exponential service times and general system parameters. For NBU service time distributions, our study agrees with [11]. In particular, in [11, Numerical Results], the authors showed that the non-preemptive LCFS policy can achieve better average age than the preemptive LCFS policy. In this paper, we further show that the non-prmp-LGFS-R policy is within a small constant gap from the optimum average age for all NBU service time distributions, which include gamma distribution as one example.

In general, our study was carried out for system settings that are more general than [18] and [11].

2) Replication vs. Non-Replication: The replication technique has gained significant attention in recent years to reduce the delay in queueing systems [14]–[16]. However, it was shown in [16], [26], [27] that replication policies are far from the optimum delay and throughput performance for NBU service time distributions. A simple explanation of this result is as follows: Let $X_1, \ldots, X_m$ be i.i.d. NBU random variables with mean $\mathbb{E}[X_l] = \mathbb{E}[X] < \infty$. From the properties of the NBU distributions, we can obtain [30]

$$\mathbb{E}[\min_{l=1,\ldots,m} X_l] \geq \frac{\mathbb{E}[X]}{m}. \tag{16}$$

Now, if X_l represents the packet service time of server l, then the left-hand side of (16) represents the mean service time (the mean time spent by each server per packet) when each packet is replicated to all servers; and the right-hand side of (16) represents the mean service time when there is no replication. This gives insight why packet replication can worsen the delay and throughput performance when the service times are NBU.

Somewhat to our surprise, we found that the non-prmp-LGFS-R policy is near-optimal in minimizing the age, even for NBU service time distributions. The intuition behind this result is that the age is affected by only the freshest packet, instead of all the packets in the queue. In other words, to reduce the age, we need to deliver the freshest packet as soon as possible. Obviously, we have

$$\mathbb{E}[\min_{l=1,\ldots,m} X_l] \leq \mathbb{E}[X]. \tag{17}$$

Thus, packet replication can help to reduce the age by exploiting the diversity provided by multiple servers. As shown in Fig. 6, we can observe that packet replication can improve the age performance. In particular, the age performance of the non-prmp-LGFS-R policy with $r = 4$ is better than that of the non-prmp-LGFS-R policy with $r = 1$ when $K \leq 12.5$.

V. Throughput-Delay Analysis

In this section, we investigate the throughput and delay performances of the proposed policies. We first consider the exponential service time distribution. Then, we generalize the service time distribution to the NBU distributions. We need the following definitions:

Definition 7. Throughput-optimality: A policy is said to be **throughput-optimal**, if it maximizes the expected number of delivered packets among all causal policies.

The average delay under policy π is defined as

$$D_{\text{avg}}(\pi) = \frac{1}{n}\sum_{i=1}^{n}[c_i(\pi) - a_i], \tag{18}$$

where the delay of packet i under policy π is $c_i(\pi) - a_i$.[2]

Definition 8. Delay-optimality: A policy is said to be **delay-optimal**, if it minimizes the expected average delay among all causal policies.

[2] The $\limsup$ operator is enforced on the right hand side of (18) if $n \to \infty$.

A. Exponential Service Time Distribution

We study the throughput and delay performance of the prmp-LGFS-R policy when the service times are exponentially distributed and *i.i.d.* across time and servers. The delay and throughput performances of the prmp-LGFS-R policy are characterized as follows:

Theorem 5. Suppose that the packet service times are exponentially distributed and *i.i.d.* across time and servers, then for all $\mathcal{I}$ such that $B = \infty$, the prmp-LGFS-R policy is throughput-optimal and delay-optimal among all policies in Π_m.

Proof. In particular, any work-conserving policy is throughput-optimal and delay-optimal. The proof details are provided in Appendix D. □

B. NBU Service Time Distributions

Now, we consider class of NBU service time distributions. We study the throughput and delay performance of the non-prmp-LGFS-R policy when the maximum replication degree is 1 (i.e., there is no replication). Moreover, we assume that all packets have identical size. The delay and throughput performances of the non-prmp-LGFS-R policy are characterized as follows:

Theorem 6. Suppose that the packet service times are NBU and *i.i.d.* across time and servers, then for all $\mathcal{I}$ such that $B = \infty$ and $r = 1$, the non-prmp-LGFS-R policy is throughput-optimal and delay-optimal among all non-preemptive policies in Π_1.

Indeed, when the packet service times are NBU and preemption is not allowed, any non-preemptive work-conserving policy in Π_1 is throughput and delay optimal. The proof is omitted because it is similar to that of Theorem 5. The only difference is that using the property of NBU distributions, we can show that service idling in any non-work-conserving policy leads to delayed packet departure instants, which in turn worsens its throughput and delay performance.

VI. Conclusions

In this paper, we studied the age-of-information optimization in multi-server queues. Packet replication was considered in this model, where the maximum replication degree is constrained. We considered general system settings including arbitrary arrival processes where the incoming update packets may arrive *out of order* of their generation times. We developed scheduling policies that can achieve age-optimality for any maximum replication degree when the packet service times are exponentially distributed. This optimality result not only holds for the age process, but also for any *non-decreasing functional* of the age process. Interestingly, the proposed policies can also achieve throughput and delay optimality. In addition, we investigated the class of NBU packet service time distributions and showed that LGFS policies with replication are near age-optimal for any maximum replication degree.

References

[1] A. M. Bedewy, Y. Sun, and N. B. Shroff, "Optimizing data freshness, throughput, and delay in multi-server information-update systems," in *Proc. IEEE ISIT*, July 2016, pp. 2569–2573.

[2] B. Adelberg, H. Garcia-Molina, and B. Kao, "Applying update streams in a soft real-time database system," in *ACM SIGMOD Record*, 1995, vol. 24, pp. 245–256.

[3] J. Cho and H. Garcia-Molina, "Synchronizing a database to improve freshness," in *ACM SIGMOD Record*, 2000, vol. 29, pp. 117–128.

[4] L. Golab, T. Johnson, and V. Shkapenyuk, "Scheduling updates in a real-time stream warehouse," in *Proc. IEEE 25th International Conference on Data Engineering*, March 2009, pp. 1207–1210.

[5] S. Kaul, R. D. Yates, and M. Gruteser, "Real-time status: How often should one update?," in *Proc. IEEE INFOCOM*, 2012, pp. 2731–2735.

[6] R. D. Yates and S. Kaul, "Real-time status updating: Multiple sources," in *IEEE International Symposium on Information Theory (ISIT)*, July 2012, pp. 2666–2670.

[7] L. Huang and E. Modiano, "Optimizing age-of-information in a multi-class queueing system," in *Proc. IEEE ISIT*, June 2015, pp. 1681–1685.

[8] M. Costa, M. Codreanu, and A. Ephremides, "On the age of information in status update systems with packet management," *IEEE Transactions on Information Theory*, vol. 62, no. 4, pp. 1897–1910, April 2016.

[9] M. Costa, M. Codreanu, and A. Ephremides, "Age of information with packet management," in *Proc. IEEE ISIT*, June 2014, pp. 1583–1587.

[10] N. Pappas, J. Gunnarsson, L. Kratz, M. Kountouris, and V. Angelakis, "Age of information of multiple sources with queue management," in *Proc. IEEE ICC*, June 2015, pp. 5935–5940.

[11] E. Najm and R. Nasser, "Age of information: The gamma awakening," in *Proc. IEEE ISIT*, July 2016, pp. 2574–2578.

[12] B. Ji, G. R. Gupta, X. Lin, and N. B. Shroff, "Performance of low-complexity greedy scheduling policies in multi-channel wireless networks: Optimal throughput and near-optimal delay," in *2013 Proceedings IEEE INFOCOM*, April 2013, pp. 2589–2597.

[13] V. Kumar, A. Grama, A. Gupta, and G. Karypis, *Introduction to parallel computing: design and analysis of algorithms*, vol. 400, Benjamin/Cummings Redwood City, CA, 1994.

[14] S. Chen, Y. Sun, U. C. Kozat, L. Huang, P. Sinha, G. Liang, X. Liu, and N. B. Shroff, "When queueing meets coding: Optimal-latency data retrieving scheme in storage clouds," in *IEEE INFOCOM 2014 - IEEE Conference on Computer Communications*, April 2014, pp. 1042–1050.

[15] Y. Sun, Z. Zheng, C. E. Koksal, K. H. Kim, and N. B. Shroff, "Provably delay efficient data retrieving in storage clouds," in *2015 IEEE Conference on Computer Communications (INFOCOM)*, April 2015, pp. 585–593.

[16] Y. Sun, C. E. Koksal, and N. B. Shroff, "On delay-optimal scheduling in queueing systems with replications," CoRR, abs/1603.07322, Jan. 2016.

[17] S. Kaul, R. D. Yates, and M. Gruteser, "Status updates through queues," in *Conf. on Info. Sciences and Systems*, Mar. 2012.

[18] R. D. Yates and S. K. Kaul, "The age of information: Real-time status updating by multiple sources," submitted to *IEEE Trans. Inf. Theory*, 2016, https://arxiv.org/abs/1608.08622.

[19] C. Kam, S. Kompella, G. D. Nguyen, and A. Ephremides, "Effect of message transmission path diversity on status age," *IEEE Trans. Inf. Theory*, vol. 62, no. 3, pp. 1360–1374, March 2016.

[20] Y. Sang, B. Li, and B. Ji, "The power of waiting for more than one response in minimizing the age-of-information," 2017, https://arxiv.org/abs/1704.04848.

[21] T. Bacinoglu, E. T. Ceran, and E. Uysal-Biyikoglu, "Age of information under energy replenishment constraints," in *Proc. Info. Theory and Appl. Workshop*, Feb. 2015.

[22] R. D. Yates, "Lazy is timely: Status updates by an energy harvesting source," in *Proc. IEEE Int. Symp. Inform. Theory*, 2015.

[23] K. Chen and L. Huang, "Age-of-information in the presence of error," in *Proc. IEEE ISIT*, July 2016, pp. 2579–2583.

[24] Y. Sun, E. Uysal-Biyikoglu, R. D. Yates, C. E. Koksal, and N. B. Shroff, "Update or wait: How to keep your data fresh," in *Proc. IEEE INFOCOM*, April 2016.

[25] Y. Sun, E. Uysal-Biyikoglu, R. D. Yates, C. E. Koksal, and N. B. Shroff, "Update or wait: How to keep your data fresh," *IEEE Trans. Inf. Theory*, in press, 2017.

[26] N. B. Shah, K. Lee, and K. Ramchandran, "When do redundant requests reduce latency?," *IEEE Transactions on Communications*, vol. 64, no. 2, pp. 715–722, 2016.

[27] G. Joshi, E. Soljanin, and G. Wornell, "Efficient redundancy techniques for latency reduction in cloud systems," *ACM Transactions on Modeling and Performance Evaluation of Computing Systems (TOMPECS)*, vol. 2, no. 2, pp. 12, 2017.

[28] A. M. Bedewy, Y. Sun, and N. B. Shroff, "Age-optimal information updates in multihop networks," in *Proc. IEEE ISIT*, 2017.

[29] Y. Sun, Y. Polyanskiy, and E. Uysal-Biyikoglu, "Remote estimation of the Wiener process over a channel with random delay," in *Proc. IEEE ISIT*, 2017.

[30] M. Shaked and J. G. Shanthikumar, *Stochastic orders*, Springer Science & Business Media, 2007.

[31] R. Durrett, *Probability: theory and examples*, Cambridge university press, 2010.

APPENDIX A
PROOF OF THEOREM 1

We need to define the system state of any policy π:

Definition 9. At any time t, the system state of policy π is specified by $\mathbf{V}_\pi(t) = (U_\pi(t), \alpha_{1,\pi}(t), \ldots, \alpha_{m,\pi}(t))$, where $U_\pi(t)$ is the generation time of the freshest packet that have already been delivered to the destination. Define $\alpha_{i,\pi}(t)$ as the i-th largest generation time of the packets being processed by the servers. Without loss of generality, if h servers are sending stale packets (i.e., $\alpha_{m,\pi}(t) \leq \alpha_{(m-1),\pi}(t) \ldots \leq \alpha_{(m-h+1),\pi}(t) \leq U_\pi(t)$) or h servers are idle, then we set $\alpha_{m,\pi}(t) = \ldots = \alpha_{(m-h+1),\pi}(t) = U_\pi(t)$. Hence,

$$U_\pi(t) \leq \alpha_{m,\pi}(t) \leq \ldots \leq \alpha_{1,\pi}(t). \tag{19}$$

Let $\{\mathbf{V}_\pi(t), t \in [0, \infty)\}$ be the state process of policy π, which is assumed to be right-continuous. For notational simplicity, let policy P represent the prmp-LGFS-R policy.

The key step in the proof of Theorem 1 is the following lemma, where we compare policy P with any work-conserving policy π.

Lemma 7. Suppose that $\mathbf{V}_P(0^-) = \mathbf{V}_\pi(0^-)$ for all work conserving policies π, then for all $\mathcal{I}$

$$[\{\mathbf{V}_P(t), t \in [0, \infty)\}|\mathcal{I}] \geq_{\text{st}} [\{\mathbf{V}_\pi(t), t \in [0, \infty)\}|\mathcal{I}]. \tag{20}$$

We use coupling and forward induction to prove Lemma 7. For any work-conserving policy π, suppose that stochastic processes $\widetilde{\mathbf{V}}_P(t)$ and $\widetilde{\mathbf{V}}_\pi(t)$ have the same stochastic laws as $\mathbf{V}_P(t)$ and $\mathbf{V}_\pi(t)$. The state processes $\widetilde{\mathbf{V}}_P(t)$ and $\widetilde{\mathbf{V}}_\pi(t)$ are coupled in the following manner: If the packet with generation time $\widetilde{\alpha}_{i,P}(t)$ is delivered at time t as $\widetilde{\mathbf{V}}_P(t)$ evolves, then the packet with generation time $\widetilde{\alpha}_{i,\pi}(t)$ is delivered at time t as $\widetilde{\mathbf{V}}_\pi(t)$ evolves. Such a coupling is valid because the service times are exponentially distributed and thus memoryless. Moreover, policy P and policy π have identical packet generation times $(s_1, s_2, \ldots, s_n)$ and packet arrival times $(a_1, a_2, \ldots, a_n)$. According to Theorem 6.B.30 in [30], if we can show

$$\mathbb{P}[\widetilde{\mathbf{V}}_P(t) \geq \widetilde{\mathbf{V}}_\pi(t), t \in [0, \infty)|\mathcal{I}] = 1, \tag{21}$$

then (20) is proven. To ease the notational burden, we will omit the tildes on the coupled versions in this proof and just use $\mathbf{V}_P(t)$ and $\mathbf{V}_\pi(t)$. Next, we use the following lemmas to prove (21):

Lemma 8. Suppose that the system state of policy P is $\{U_P, \alpha_{1,P}, \ldots, \alpha_{m,P}\}$, and meanwhile the system state of policy π is $\{U_\pi, \alpha_{1,\pi}, \ldots, \alpha_{m,\pi}\}$. If

$$U_P \geq U_\pi, \tag{22}$$

then,

$$\alpha_{i,P} \geq \alpha_{i,\pi}, \quad \forall i = 1, \ldots, m. \tag{23}$$

Proof. Let S denote the set of packets that have arrived to the system at the considered time epoch. It is important to note that the set S is invariant of the scheduling policy. We use $s_{[i]}$ to denote the i-th largest generation time of the packets in S. Define $k = \lfloor \frac{m}{r} \rfloor$. From the definition of the system state and policy P, we have

$$\begin{aligned} &\alpha_{i,P} = \max\{s_{[j]}, U_P\},\ \forall i = (j-1)r+1, \ldots, jr,\ \forall j = 1, \ldots, k, \\ &\alpha_{i,P} = \max\{s_{[k+1]}, U_P\},\ \forall i = kr+1, \ldots, m. \end{aligned} \tag{24}$$

Since policy π is an arbitrary policy, the servers under policy π may not process the freshest packets in the set S or policy π may replicate older packets more than the fresher ones in the set S. Hence, we have

$$\begin{aligned} &\alpha_{i,\pi} \leq \max\{s_{[j]}, U_\pi\},\ \forall i = (j-1)r+1, \ldots, jr,\ \forall j = 1, \ldots, k, \\ &\alpha_{i,\pi} \leq \max\{s_{[k+1]}, U_\pi\},\ \forall i = kr+1, \ldots, m. \end{aligned} \tag{25}$$

where the maximization here follows from the definition of the system state. Since the set S is invariant of the scheduling policy and $U_P \geq U_\pi$, this with (24) and (25) imply

$$\alpha_{i,P} \geq \alpha_{i,\pi}, \quad \forall i = 1, \ldots, m, \tag{26}$$

which completes the proof. □

Lemma 9. Suppose that under policy P, $\{U'_P, \alpha'_{1,P}, \ldots, \alpha'_{m,P}\}$ is obtained by delivering a packet with generation time $\alpha_{l,P}$ to the destination in the system whose state is $\{U_P, \alpha_{1,P}, \ldots, \alpha_{m,P}\}$. Further, suppose that under policy π, $\{U'_\pi, \alpha'_{1,\pi}, \ldots, \alpha'_{m,\pi}\}$ is obtained by delivering a packet with generation time $\alpha_{l,\pi}$ to the destination in the system whose state is $\{U_\pi, \alpha_{1,\pi}, \ldots, \alpha_{m,\pi}\}$. If

$$\alpha_{i,P} \geq \alpha_{i,\pi}, \quad \forall i = 1, \ldots, m, \tag{27}$$

then,

$$U'_P \geq U'_\pi, \alpha'_{i,P} \geq \alpha'_{i,\pi}, \quad \forall i = 1, \ldots, m. \tag{28}$$

Proof. Since the packet with generation time $\alpha_{l,P}$ is delivered under policy P, the packet with generation time $\alpha_{l,\pi}$ is delivered under policy π, and $\alpha_{l,P} \geq \alpha_{l,\pi}$, we get

$$U'_P = \alpha_{l,P} \geq \alpha_{l,\pi} = U'_\pi. \tag{29}$$

This, together with Lemma 8, implies

$$\alpha'_{i,P} \geq \alpha'_{i,\pi}, \quad i = 1, \ldots, m. \tag{30}$$

Hence, (28) holds for any queue size $B \geq 0$, which completes the proof. □

Lemma 10. Suppose that under policy P, $\{U'_P, \alpha'_{1,P}, \ldots, \alpha'_{m,P}\}$ is obtained by adding a packet to the system whose state is $\{U_P, \alpha_{1,P}, \ldots, \alpha_{m,P}\}$. Further, suppose that under policy π, $\{U'_\pi, \alpha'_{1,\pi}, \ldots, \alpha'_{m,\pi}\}$ is obtained by adding a packet to the system whose state is $\{U_\pi, \alpha_{1,\pi}, \ldots, \alpha_{m,\pi}\}$. If

$$U_P \geq U_\pi, \tag{31}$$

then

$$U'_P \geq U'_\pi, \alpha'_{i,P} \geq \alpha'_{i,\pi}, \quad \forall i = 1, \ldots, m. \tag{32}$$

Proof. Since there is no packet delivery, we have

$$U'_P = U_P \geq U_\pi = U'_\pi. \tag{33}$$

This, together with Lemma 8, implies

$$\alpha'_{i,P} \geq \alpha'_{i,\pi}, \quad i = 1, \ldots, m. \tag{34}$$

Hence, (32) holds for any queue size $B \geq 0$, which completes the proof. □

Proof of Lemma 7. For any sample path, we have that $U_P(0^-) = U_\pi(0^-)$ and $\alpha_{i,P}(0^-) = \alpha_{i,\pi}(0^-)$ for $i = 1, \ldots, m$. According to the coupling between the system state processes $\{\mathbf{V}_P(t), t \in [0, \infty)\}$ and $\{\mathbf{V}_\pi(t), t \in [0, \infty)\}$, as well as Lemma 9 and 10, we get

$$[U_P(t)|\mathcal{I}] \geq [U_\pi(t)|\mathcal{I}], [\alpha_{i,P}(t)|\mathcal{I}] \geq [\alpha_{i,\pi}(t)|\mathcal{I}],$$

holds for all $t \in [0, \infty)$ and $i = 1, \ldots, m$. Hence, (21) follows which implies (20) by Theorem 6.B.30 in [30]. This completes the proof. □

Proof of Theorem 1. As a result of Lemma 7, we have

$$[\{U_P(t), t \in [0, \infty)\}|\mathcal{I}] \geq_{\text{st}} [\{U_\pi(t), t \in [0, \infty)\}|\mathcal{I}],$$

holds for all work-conserving policies π, which implies

$$[\{\Delta_P(t), t \in [0, \infty)\}|\mathcal{I}] \leq_{\text{st}} [\{\Delta_\pi(t), t \in [0, \infty)\}|\mathcal{I}], \tag{35}$$

holds for all work-conserving policies π.

For non-work-conserving policies, since the service times are exponentially distributed and *i.i.d.* across time and servers, service idling only increases the waiting time of the packet in the system. Therefore,

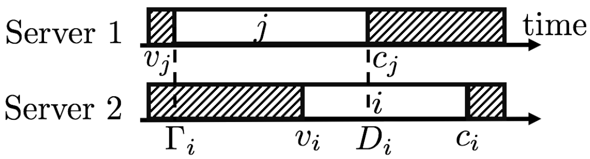


Figure 7: An illustration of v_i, c_i, Γ_i, and D_i. There are 2 servers, and $s_j > s_i$. The packet j is assigned to Server 1, and packet i is assigned to Server 2. The service starting time and completion time of packet j are earlier than those of packet i. Thus, we have $\Gamma_i = v_j$ and $D_i = c_j$.

the age under non-work-conserving policies will be greater. As a result, we have

$$[\{\Delta_P(t), t \in [0,\infty)\}|\mathcal{I}] \leq_{\text{st}} [\{\Delta_\pi(t), t \in [0,\infty)\}|\mathcal{I}],\ \forall \pi \in \Pi_r.$$

Finally, (10) follows directly from (9) using the properties of stochastic ordering [30]. This completes the proof. □

Appendix B
Proof of Lemma 3

The proof of Lemma 3 is motivated by [16]. For notation simplicity, let policy P represent the non-prmp-LGFS-R policy. We need to define the following parameters:

Define Γ_i and D_i as

$$\Gamma_i = \min\{v_j : s_j \geq s_i\}, \tag{36}$$

$$D_i = \min\{c_j : s_j \geq s_i\}. \tag{37}$$

where Γ_i and D_i are the smallest assignment time and completion time, respectively, of all packets that are fresher than the packet i. An illustration of these parameters is provided in Fig. 7. Define the vectors $\mathbf{\Gamma} = (\Gamma_1, \ldots, \Gamma_n)$, and $\mathbf{D} = (D_1, \ldots, D_n)$. All these quantities are functions of the scheduling policy π.

To prove (13), we need to show that

$$[\mathbf{\Gamma}(P)|\mathcal{I}] \leq_{\text{st}} [\mathbf{D}(\pi)|\mathcal{I}], \tag{38}$$

holds for all $\pi \in \Pi_m$. We pick an arbitrary policy $\pi \in \Pi_m$ and prove (38) using Theorem 6.B.3 of [30] into two steps.

Step 1: Consider packet 1 that arrived at time a_1, where $a_1 \leq a_2 \leq ....$ Since all servers are idle by time a_1 and policy P is work-conserving policy, packet 1 will be assigned to a server under policy P once it arrives. Thus, from (36), we obtain

$$[\Gamma_1(P)|\mathcal{I}] = [v_1(P)|\mathcal{I}] = a_1. \tag{39}$$

Under policy π, the completion times of all packets must be no smaller than a_1. Hence, we have

$$[c_i(\pi)|\mathcal{I}] \geq a_1,\ \forall i \geq 1. \tag{40}$$

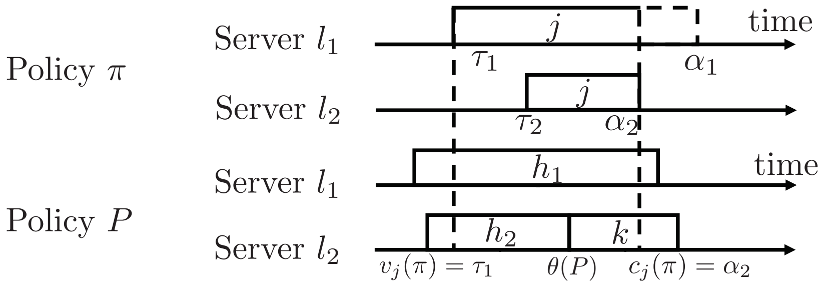


Figure 8: Illustration of packet assignments under policy π and policy P, respectively. In policy π, two copies of packet j are replicated on the server l_1 and server l_2 at time τ_1 and τ_2, where $v_j(\pi) = \min\{\tau_1, \tau_2\} = \tau_1$. Server l_2 completes one copy of packet j at time $c_j(\pi) = \alpha_2$, server l_1 cancels its redundant copy of packet j at time $c_j(\pi)$. Hence, the service duration of packet j is $[v_j(\pi), c_j(\pi)]$ in policy π. In policy P, at least one of the servers l_1 and l_2 becomes idle before time $c_j(\pi)$. In this example, server l_2 becomes idle at time $c_{h_2}(P) < c_j(\pi)$ and a fresh packet k with $s_k \geq s_j$ starts its service on server l_2 at time $c_{h_2}(P)$.

This with (37) imply

$$[D_1(\pi)|\mathcal{I}] \geq a_1. \tag{41}$$

Combining (39) and (41), we get

$$[\Gamma_1(P)|\mathcal{I}] \leq [D_1(\pi)|\mathcal{I}]. \tag{42}$$

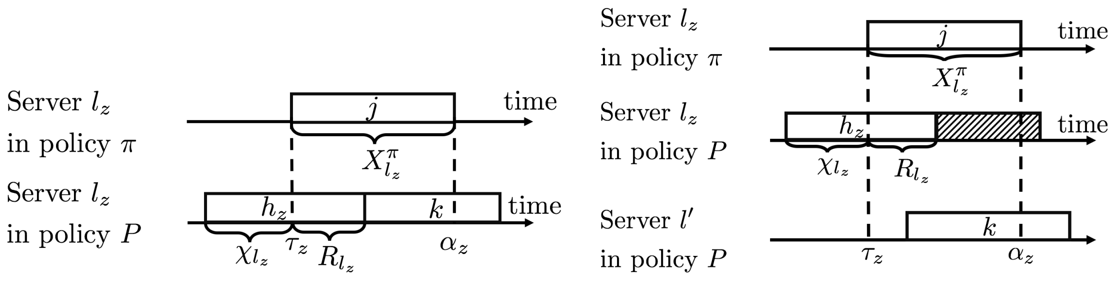
Step 2: Consider a packet j, where $2 \leq j \leq n$. We suppose that, both in math description and in words, there is no packet with generation time greater than s_j has been delivered before packet j under policy π; otherwise, if there is a fresher packet y with $s_y > s_j$ and $c_y(\pi) < c_j(\pi)$, then we replace packet j by packet y in the following argument. We need to prove that

$$\begin{aligned} &[\Gamma_j(P)|\mathcal{I}, \Gamma_1(P) = \gamma_1, \ldots, \Gamma_{j-1}(P) = \gamma_{j-1}] \\ \leq_{\text{st}} &[D_j(\pi)|\mathcal{I}, D_1(\pi) = d_1, \ldots, D_{j-1}(\pi) = d_{j-1}] \\ &\text{whenever} \quad \gamma_i \leq d_i, i = 1, 2, \ldots, j-1. \end{aligned} \tag{43}$$

For notational simplicity, define $\Gamma^{j-1} \triangleq \{\Gamma_1(P) = \gamma_1, \ldots, \Gamma_{j-1}(P) = \gamma_{j-1}\}$ and $D^{j-1} \triangleq \{D_1(\pi) = d_1, \ldots, D_{j-1}(\pi) = d_{j-1}\}$.

As illustrated in Fig. 8, suppose that u copies of packet j are replicated on the servers $l_1, l_2, \ldots, l_u$ at the time epochs $\tau_1, \tau_2, \ldots, \tau_u$ in policy π, where $v_j(\pi) = \min_{w=1,\ldots,u} \tau_w$.[3] In addition, suppose that server l_w will complete serving its copy of packet j at time α_w if there is no cancellation. Then, one of these u servers will complete one copy of packet j at time $c_j(\pi) = \min_{w=1,\ldots,u} \alpha_w$, which is the earliest among these u servers. Hence, packet j starts service at time $v_j(\pi)$ and completes service at time $c_j(\pi)$ in policy π. In policy P, let h_w represent the index of the last packet that has been assigned to server

[3] If $u = 1$, then either there is no replication or policy π decides not to replicate packet j.



(a) Case 1: Packet k is assigned to server l_z after the completion of packet h_z.

(b) Case 2: Packet k is assigned to server l' before the completion of packet h_z.

Figure 9: The possible cases to occur after the completion of packet h_z.

l_w before time τ_w. Suppose that under policy P, server l_w has spent χ_{l_w} $(\chi_{l_w} \geq 0)$ seconds on serving packet h_w before time τ_w. Let R_{l_w} denote the remaining service time of server l_w for serving packet h_w after time τ_w in policy P. Let $X_{l_w}^{\pi} = \alpha_w - \tau_w$ denote the service time of one copy of packet j in server l_w under policy π and $X_{l_w}^{P} = \chi_{l_w} + R_{l_w}$ denote the service time of packet h_w in server l_w under policy P. The CCDF of R_{l_w} is given by

$$\mathbb{P}[R_{l_w} > s] = \mathbb{P}[X_{l_w}^{P} - \chi_{l_w} > s | X_{l_w}^{P} > \chi_{l_w}]. \tag{44}$$

Because the packet service times are NBU, we can obtain that for all $s, \chi_{l_w} \geq 0$

$$\mathbb{P}[X_{l_w}^{P} - \chi_{l_w} > s | X_{l_w}^{P} > \chi_{l_w}] = \mathbb{P}[X_{l_w}^{\pi} - \chi_{l_w} > s | X_{l_w}^{\pi} > \chi_{l_w}] \leq \mathbb{P}[X_{l_w}^{\pi} > s]. \tag{45}$$

By combining (44) and (45), we obtain

$$R_{l_w} \leq_{\text{st}} X_{l_w}^{\pi}. \tag{46}$$

Because the packet service times are independent across the servers, by Lemma 13 of [16], $R_{l_1}, \ldots, R_{l_u}$ are mutually independent. By Theorem 6.B.16.(b) of [30] and (46), we can obtain

$$\min_{w=1,\ldots,u} \tau_w + R_{l_w} \leq_{\text{st}} \min_{w=1,\ldots,u} \tau_w + X_{l_w}^{\pi} = \min_{w=1,\ldots,u} \alpha_w. \tag{47}$$

From (47) we can deduce that at least one of the servers $l_1, \ldots, l_u$, say server l_z, becomes available to serve a new packet under policy P at a time that is stochastically smaller than the time $c_j(\pi) = \min_{w=1,\ldots,u} \alpha_w$. Let $\theta(P)$ denote the time that server l_z becomes available to serve a new packet in policy P. According to (47), we have

$$\begin{aligned} &[\theta(P) | \mathcal{I}, \Gamma^{j-1}] \leq_{\text{st}} [c_j(\pi) | \mathcal{I}, D^{j-1}] \\ &\text{whenever} \quad \gamma_i \leq d_i, i = 1, 2, \ldots, j-1. \end{aligned} \tag{48}$$

At time $\theta(P)$, we have two possible cases under policy P:

Case 1: A fresh packet k is assigned at time $\theta(P)$ to server l_z under policy P such that $s_k \geq s_j$, as shown in Fig. 9(a). Hence, we obtain

$$\begin{aligned}[v_k(P)|\mathcal{I}, \Gamma^{j-1}] = [\theta(P)|\mathcal{I}, \Gamma^{j-1}] \leq_{\text{st}} [c_j(\pi)|\mathcal{I}, D^{j-1}] \\ \text{whenever} \quad \gamma_i \leq d_i, i = 1, 2, \ldots, j-1.\end{aligned} \tag{49}$$

Since $s_k \geq s_j$, (36) implies

$$[\Gamma_j(P)|\mathcal{I}, \Gamma^{j-1}] \leq [v_k(P)|\mathcal{I}, \Gamma^{j-1}] \tag{50}$$

Since there is no packet with generation time greater than s_j has been delivered before packet j under policy π, (37) implies

$$[D_j(\pi)|\mathcal{I}, D^{j-1}] = [c_j(\pi)|\mathcal{I}, D^{j-1}] \tag{51}$$

By combining (49), (50), and (51), (43) follows.

Case 2: A stale packet (with generation time smaller than s_j) is assigned to server l_z or there is no packet assignment to server l_z at time $\theta(P)$ under policy P. Since policy P is a work-conserving policy, policy P serves the freshest packet first, and the packet generation times $(s_1, \ldots, s_n)$ and arrival times $(a_1, \ldots, a_n)$ are invariant of the scheduling policy, a packet k with $s_k \geq s_j$ must have been assigned to another server, call it server l', before time $\theta(P)$, as shown in Fig. 9(b). Hence, we obtain

$$\begin{aligned}[v_k(P)|\mathcal{I}, \Gamma^{j-1}] \leq [\theta(P)|\mathcal{I}, \Gamma^{j-1}] \leq_{\text{st}} [c_j(\pi)|\mathcal{I}, D^{j-1}] \\ \text{whenever} \quad \gamma_i \leq d_i, i = 1, 2, \ldots, j-1.\end{aligned} \tag{52}$$

Similar to Case 1, we can use (36), (37), and (52) to show that (43) follows in this case.

As we mentioned before, if there is a packet y with $s_y > s_j$ and $c_y(\pi) < c_j(\pi)$, then we replace packet j by packet y in the previous argument to obtain

$$\begin{aligned}[\Gamma_y(P)|\mathcal{I}, \Gamma^{j-1}] \leq_{\text{st}} [D_y(\pi)|\mathcal{I}, D^{j-1}] \\ \text{whenever} \quad \gamma_i \leq d_i, i = 1, 2, \ldots, j-1.\end{aligned} \tag{53}$$

Observing that $s_y > s_j$, (36) implies

$$[\Gamma_j(P)|\mathcal{I}, \Gamma^{j-1}] \leq [\Gamma_y(P)|\mathcal{I}, \Gamma^{j-1}]. \tag{54}$$

Since $c_y(\pi) < c_j(\pi)$ and $s_y > s_j$, (37) implies

$$[D_j(\pi)|\mathcal{I}, D^{j-1}] = [D_y(\pi)|\mathcal{I}, D^{j-1}]. \tag{55}$$

By combining (53), (54), and (55), we can prove (43) in this case too. Now, substituting (42) and (43) into Theorem 6.B.3 of [30], (38) is proven.

Finally, we can deduce from (3) that the age process $\{\Delta_\pi(t), t \in [0, \infty)\}$ under any policy π is an increasing function of $\mathbf{D}(\pi)$. Moreover, we can deduce from (12) that the process $\{\Delta_P^{\text{LB}}(t), t \in [0, \infty)\}$ is an increasing function of $\mathbf{\Gamma}(P)$. By using Theorem 6.B.16.(a) of [30], (13) follows directly from (38). This completes the proof.

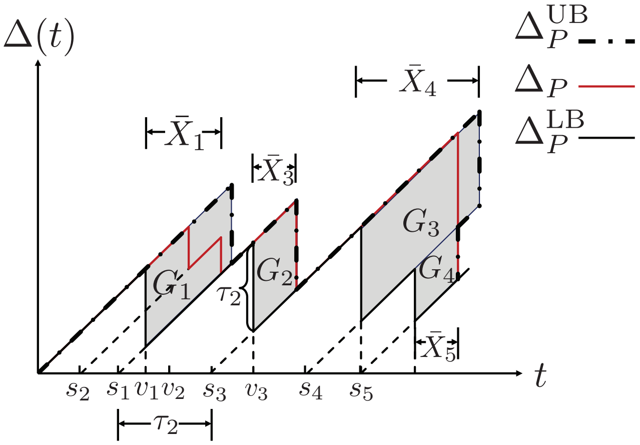


Figure 10: The evolution of Δ_P^{LB}, Δ_P, and Δ_P^{UB} in a queue with 4 servers and $r = 3$.

Appendix C
Proof of Theorem 4

For notation simplicity, let policy P represent the non-prmp-LGFS-R policy.

Proof of Theorem 4.(a). We prove Theorem 4.(a) into two steps:

Step 1: Construct an upper bound of the age process under policy P, denoted by Δ_P^{UB}. A packet i is said to be an informative assigned packet under policy P if every packet starting service before packet i are staler than packet i, i.e., $s_j \leq s_i$ for all packet j satisfying $v_j(P) \leq v_i(P)$. The informative assigned packets under policy P are those packets affecting the process Δ_P^{LB} defined in (12). We use i_k to represent the index of the k-th informative assigned packet under policy P, where we assume that $s_{i_0} = s_0 = 0$. For example, in Fig. 10, the second informative assigned packet under policy P is packet 3, so $i_2 = 3$.

We construct the upper bound Δ_P^{UB} of the age process $\Delta_P(t)$ from its lower bound $\Delta_P^{\text{LB}}(t)$. We know that the k-th informative assigned packet (i.e., packet i_k) starts service at time $v_{i_k}(P)$ under policy P. Suppose that packet i_k is replicated to several servers, let $\bar{X}_{i_k}$ denote the service time of a randomly chosen replica of packet i_k that started service at time $v_{i_k}(P)$, where $\mathbb{E}[\bar{X}_{i_k}] = \mathbb{E}[X]$ for all i_k.[4] Because of replications, packet i_k completes service under policy P as soon as one of its replica is completes service. Hence, packet i_k is delivered no later than $v_{i_k}(P) + \bar{X}_{i_k}$ under policy P. Thus, the upper bound Δ_P^{UB} is constructed by postpone each drop in the graph of $\Delta_P^{\text{LB}}(t)$ resulting from assignment of informative packet i_k by the time $\bar{X}_{i_k}$, as shown in Fig. 10.

Step 2: Identify the average gap between Δ_P^{LB} and Δ_P^{UB}. We use $\{G(t), t \in [0, \infty)\}$ to denote the gap process between Δ_P^{LB} and Δ_P^{UB}. The average gap is given by

$$[\bar{G}|\mathcal{I}] = \limsup_{T\to\infty} \frac{\mathbb{E}[\int_0^T G(t)dt]}{T}. \tag{56}$$

[4] The service time of a packet in a server means here that the amount of time that this packet spends in this server as if there is no cancellation.

Let τ_k denote the inter-generation time between packet i_k and packet i_{k-1} (i.e., $\tau_k = s_{i_k} - s_{i_{k-1}}$), where $\tau = \{\tau_k, k \geq 1\}$. Define $N(T) = \max\{k : s_{i_k} \leq T\}$ as the number of informative assigned packets under policy P by time T. Note that $[0, s_{i_{N(T)}}] \subseteq [0, T]$, where the length of the interval $[0, s_{i_{N(T)}}]$ is $\sum_{k=1}^{N(T)} \tau_k$. Thus, we have

$$\sum_{k=1}^{N(T)} \tau_k \leq T. \tag{57}$$

The area defined by the integral in (56) can be decomposed into a sum of disjoint geometric parts. Observing Fig. 10, the area can be approximated to the concatenation of the parallelograms $G_1, G_2, \dots$ (G_k's are highlighted in Fig. 10). Note that the parallelogram G_k results from the assignment of the informative packet i_k. Since each parallelogram G_k comes after the time s_{i_k} and the observing time T is chosen arbitrary, we have

$$\sum_{k=1}^{N(T)} G_k \geq \int_0^T G(t)dt. \tag{58}$$

Combining (57) and (58), we get

$$\frac{\int_0^T G(t)dt}{T} \leq \frac{\sum_{k=1}^{N(T)} G_k}{\sum_{k=1}^{N(T)} \tau_k}. \tag{59}$$

Then, take conditional expectation given τ and $N(T)$ on both sides of (59), we obtain

$$\frac{\mathbb{E}[\int_0^T G(t)dt|\tau, N(T)]}{T} \leq \frac{\mathbb{E}[\sum_{k=1}^{N(T)} G_k|\tau, N(T)]}{\sum_{k=1}^{N(T)} \tau_k} = \frac{\sum_{k=1}^{N(T)} \mathbb{E}[G_k|\tau, N(T)]}{\sum_{k=1}^{N(T)} \tau_k}, \tag{60}$$

where the second equality follows from the linearity of the expectation. From Fig. 10, G_k can be calculated as

$$G_k = \tau_k \bar{X}_{i_k}. \tag{61}$$

Substituting by (61) into (60), yields

$$\frac{\mathbb{E}[\int_0^T G(t)dt|\tau, N(T)]}{T} \leq \frac{\sum_{k=1}^{N(T)} \mathbb{E}[\tau_k \bar{X}_{i_k}|\tau, N(T)]}{\sum_{k=1}^{N(T)} \tau_k} = \frac{\sum_{k=1}^{N(T)} \tau_k \mathbb{E}[\bar{X}_{i_k}|\tau, N(T)]}{\sum_{k=1}^{N(T)} \tau_k}. \tag{62}$$

Note that the packet service times are independent of the packet generation process. Thus, we have $\mathbb{E}[\bar{X}_{i_k}|\tau, N(T)] = \mathbb{E}[\bar{X}_{i_k}] = \mathbb{E}[X]$ for all i_k. Substituting this into (62), yields

$$\frac{\mathbb{E}[\int_0^T G(t)dt|\tau, N(T)]}{T} \leq \frac{\sum_{k=1}^{N(T)} \tau_k \mathbb{E}[X]}{\sum_{k=1}^{N(T)} \tau_k} = \mathbb{E}[X], \tag{63}$$

by the law of iterated expectations, we have

$$\frac{\mathbb{E}[\int_0^T G(t)dt]}{T} \leq \mathbb{E}[X]. \tag{64}$$

By taking the $\limsup$ of both side of (64) when $T \to \infty$, we obtain

$$\limsup_{T\to\infty} \frac{\mathbb{E}[\int_0^T G(t)dt]}{T} \leq \mathbb{E}[X]. \tag{65}$$

Equation (65) tells us that the average gap between Δ_P^{LB} and Δ_P^{UB} is no larger than $\mathbb{E}[X]$. Since, Δ_P^{LB} and Δ_P^{UB} are lower and upper bounds, respectively, of the age process of policy P, we obtain

$$[\bar{\Delta}_P^{\text{LB}}|\mathcal{I}] \leq [\bar{\Delta}_P|\mathcal{I}] \leq [\bar{\Delta}_P^{\text{LB}}|\mathcal{I}] + \mathbb{E}[X], \tag{66}$$

where $\bar{\Delta}_P^{\text{LB}} = \limsup_{T\rightarrow\infty} \frac{\mathbb{E}[\int_0^T \Delta_P^{\text{LB}}(t)dt]}{T}$. From Lemma 3, we have for all $\mathcal{I}$ satisfying $B \geq 1$, and $\pi \in \Pi_m$

$$[\{\Delta_P^{\text{LB}}(t), t \in [0,\infty)\}|\mathcal{I}] \leq_{\text{st}} [\{\Delta_\pi(t), t \in [0,\infty)\}|\mathcal{I}], \tag{67}$$

which implies that

$$[\bar{\Delta}_P^{\text{LB}}|\mathcal{I}] \leq [\bar{\Delta}_\pi|\mathcal{I}], \tag{68}$$

holds for all $\pi \in \Pi_m$. As a result, we get

$$[\bar{\Delta}_P^{\text{LB}}|\mathcal{I}] \leq \min_{\pi\in\Pi_m} [\bar{\Delta}_\pi|\mathcal{I}]. \tag{69}$$

Since policy non-prmp-LGFS-R is a feasible policy, we get

$$\min_{\pi\in\Pi_m} [\bar{\Delta}_\pi|\mathcal{I}] \leq [\bar{\Delta}_P|\mathcal{I}]. \tag{70}$$

Combining (66), (69), and (70), we get

$$\min_{\pi\in\Pi_m} [\bar{\Delta}_\pi|\mathcal{I}] \leq [\bar{\Delta}_P|\mathcal{I}] \leq \min_{\pi\in\Pi_m} [\bar{\Delta}_\pi|\mathcal{I}] + \mathbb{E}[X], \tag{71}$$

which complete the proof. □

Proof of Theorem 4.(b). The proof of part (b) is similar to that of part (a). We use i_k to represent the index of the k-th informative assigned packet under policy P. Note that when m is a multiple of r such that $m \bmod r = 0$, the k-th informative assigned packet (i.e., packet i_k), that starts service at time $v_{i_k}(P)$, is processed by r servers in policy P. Let $\mathcal{S}_{i_k} \subseteq \{1, \ldots, m\}$ be the set of servers that process packet i_k, which satisfies $|\mathcal{S}_{i_k}| = r$. Because of replications, packet i_k completes service under policy P as soon as one of its replica is completes service. Hence, packet i_k is delivered at time $v_{i_k}(P) + \min_{l\in\mathcal{S}_{i_k}} X_l$ under policy P. Thus, the upper bound Δ_P^{UB} is constructed by postpone each drop in the graph of $\Delta_P^{\text{LB}}(t)$ resulting from assignment of informative packet i_k by the time $\min_{l\in\mathcal{S}_{i_k}} X_l$, as shown in Fig. 11.

Similarly, we use $\{G(t), t \in [0,\infty)\}$ to denote the gap process between Δ_P^{LB} and Δ_P^{UB}. The average gap is given by

$$[\bar{G}|\mathcal{I}] = \limsup_{T\rightarrow\infty} \frac{\mathbb{E}[\int_0^T G(t)dt]}{T}. \tag{72}$$

Following the same steps as in the proof of part (a), we can show that

$$\limsup_{T\rightarrow\infty} \frac{\mathbb{E}[\int_0^T G(t)dt]}{T} \leq \mathbb{E}[\min_{l=1,\ldots,r} X_l]. \tag{73}$$

Equation (73) tells us that the average gap between Δ_P^{LB} and Δ_P^{UB} is no larger than $\mathbb{E}[\min_{l=1,\ldots,r} X_l]$.

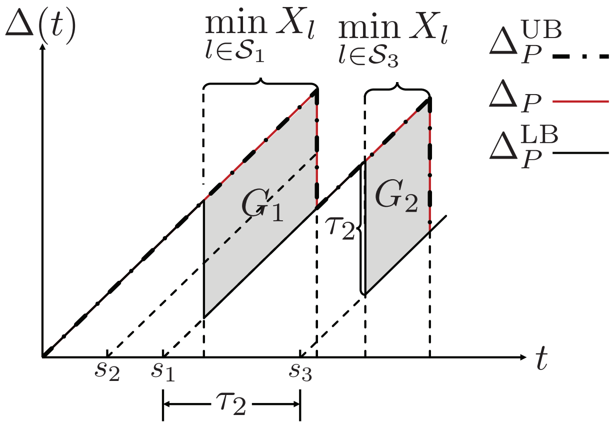


Figure 11: The evolution of $\Delta_P^{\rm LB}$, Δ_P, and $\Delta_P^{\rm UB}$ in a queue with 2 servers and $r = 2$.

Since, $\Delta_P^{\rm LB}$ and $\Delta_P^{\rm UB}$ are lower and upper bounds, respectively, of the age process of policy P, we obtain

$$[\bar{\Delta}_P^{\rm LB}|\mathcal{I}] \leq [\bar{\Delta}_P|\mathcal{I}] \leq [\bar{\Delta}_P^{\rm LB}|\mathcal{I}] + \mathbb{E}[\min_{l=1,\ldots,r} X_l]. \tag{74}$$

Similar to part (a), we can use (74) with Lemma 3 to show that

$$\min_{\pi\in\Pi_m} [\bar{\Delta}_\pi|\mathcal{I}] \leq [\bar{\Delta}_P|\mathcal{I}] \leq \min_{\pi\in\Pi_m} [\bar{\Delta}_\pi|\mathcal{I}] + \mathbb{E}[\min_{l=1,\ldots,r} X_l], \tag{75}$$

which complete the proof. □

Appendix D
Proof of Theorem 5

We follow the same proof technique of Theorem 1. We start by comparing policy P (prmp-LGFS-R policy) with an arbitrary work-conserving policy π. For this, we need to define the system state of any policy π:

Definition 10. At any time t, the system state of policy π is specified by $H_\pi(t) = (N_\pi(t), \gamma_\pi(t))$, where $N_\pi(t)$ is the total number of packets in the system at time t. Define $\gamma_\pi(t)$ as the total number of packets that are delivered to the destination at time t. Let $\{H_\pi(t), t \in [0, \infty)\}$ be the state process of policy π, which is assumed to be right-continuous.

To prove Theorem 5, we will need the following lemma.

Lemma 11. For any work-conserving policy π, if $H_P(0^-) = H_\pi(0^-)$ and $B = \infty$, then $[\{H_P(t), t \in [0, \infty)\}|\mathcal{I}]$ and $[\{H_\pi(t), t \in [0, \infty)\}|\mathcal{I}]$ are of the same distribution.

Suppose that $\{\widetilde{H}_P(t), t \in [0, \infty)\}$ and $\{\widetilde{H}_\pi(t), t \in [0, \infty)\}$ are stochastic processes having the same stochastic laws as $\{H_P(t), t \in [0, \infty)\}$ and $\{H_\pi(t), t \in [0, \infty)\}$. Now, we couple the packet delivery times during the evolution of $\widetilde{H}_P(t)$ to be identical with the packet delivery times during the evolution of $\widetilde{H}_\pi(t)$.

To ease the notational burden, we will omit the tildes henceforth on the coupled versions and just use $\{H_P(t)\}$ and $\{H_\pi(t)\}$. The following two lemmas are needed to prove Lemma 11:

Lemma 12. Suppose that under policy P, $\{N'_P, \gamma'_P\}$ is obtained by delivering a packet to the destination in the system whose state is $\{N_P, \gamma_P\}$. Further, suppose that under policy π, $\{N'_\pi, \gamma'_\pi\}$ is obtained by delivering a packet to the destination in the system whose state is $\{N_\pi, \gamma_\pi\}$. If

$$N_P = N_\pi, \gamma_P = \gamma_\pi,$$

then

$$N'_P = N'_\pi, \gamma'_P = \gamma'_\pi. \tag{76}$$

Proof. Because the packet service times are *i.i.d.* across time and servers and the CCDF $\bar{F}$ is continuous, the probability for any two servers to complete their packets at the same time is zero. Therefore, in policy P, if one copy of a replicated packet is completed on a server, the remaining replicated copies of this packet are still being processed on the other servers; these replicated packet copies are cancelled immediately and a new packet is replicated on these servers. Since there is a packet delivery, we have

$$N'_P = N_P - 1 = N_\pi - 1 = N'_\pi,$$
$$\gamma'_P = \gamma_P + 1 = \gamma_\pi + 1 = \gamma'_\pi.$$

Hence, (76) holds, which complete the proof. □

Lemma 13. Suppose that under policy P, $\{N'_P, \gamma'_P\}$ is obtained by adding a new packet to the system whose state is $\{N_P, \gamma_P\}$. Further, suppose that under policy π, $\{N'_\pi, \gamma'_\pi\}$ is obtained by adding a new packet to the system whose state is $\{N_\pi, \gamma_\pi\}$. If

$$N_P = N_\pi, \gamma_P = \gamma_\pi,$$

then

$$N'_P = N'_\pi, \gamma'_P = \gamma'_\pi. \tag{77}$$

Proof. Because $B = \infty$, no packet is dropped in policy P and policy π. Since there is a new added packet to the system, we have

$$N'_P = N_P + 1 = N_\pi + 1 = N'_\pi.$$

Also, there is no packet delivery, hence

$$\gamma'_P = \gamma_P = \gamma_\pi = \gamma'_\pi.$$

Thus, (77) holds, which complete the proof. □

Proof of Lemma 11. For any sample path, we have that $N_P(0^-) = N_\pi(0^-)$ and $\gamma_P(0^-) = \gamma_\pi(0^-)$. According to the coupling between the system state processes $\{H_P(t), t \in [0, \infty)\}$ and $\{H_\pi(t), t \in$

$[0,\infty)\}$, as well as Lemma 12 and 13, we get

$$[N_P(t)|\mathcal{I}] = [N_\pi(t)|\mathcal{I}], [\gamma_P(t)|\mathcal{I}] = [\gamma_\pi(t)|\mathcal{I}],$$

holds for all $t \in [0,\infty)$. This implies that $[\{H_P(t), t \in [0,\infty)\}|\mathcal{I}]$ and $[\{H_\pi(t), t \in [0,\infty)\}|\mathcal{I}]$ are of the same distribution, which completes the proof. □

Proof of Theorem 5. As a result of Lemma 11, $[\{\gamma_P(t), t \in [0,\infty)\}|\mathcal{I}]$ and $[\{\gamma_\pi(t), t \in [0,\infty)\}|\mathcal{I}]$ are of the same distribution for any work-conserving policy π. This implies that all work-conserving policies have the same throughput performance. Also, from Lemma 11, we have that $[\{N_P(t), t \in [0,\infty)\}|\mathcal{I}]$ and $[\{N_\pi(t), t \in [0,\infty)\}|\mathcal{I}]$ are of the same distribution for all work conserving policies. Hence, all work conserving policies have the same delay performance.

Finally, since the service times are *i.i.d.* across time and servers, service idling only increases the waiting time of the packet in the system. Therefore, the throughput and delay performance under non-work-conserving policies will be worse. As a result, the prmp-LGFS-R policy is throughput-optimal and delay-optimal among all policies in Π_m (indeed, any work-conserving policy with infinite queue size $B = \infty$ is throughput-optimal and delay-optimal). □