Age-Optimal Multi-Flow Status Updating with Errors: A Sample-Path Approach

Yin Sun and Sastry Kompella

Digital Object Identifier: 10.23919/JCN.2023.000041

Abstract—In this paper, we study an age of information minimization problem in *continuous-time* and *discrete-time* status updating systems that involve *multiple packet flows*, *multiple servers*, and *transmission errors*. Four scheduling policies are proposed. We develop a unifying sample-path approach and use it to show that, when the packet generation and arrival times are synchronized across the flows, the proposed policies are (near) optimal for minimizing any *time-dependent*, *symmetric*, and *nondecreasing* penalty function of the ages of the flows over time in a stochastic ordering sense.

Index Terms—Age of information, errors, multiple channels, multiple flows, sample-path approach, status updating.

I. Introduction

In many information-update and networked control systems, such as news updates, stock trading, autonomous driving, remote surgery, robotics control, and real-time surveillance, information usually has the greatest value when it is fresh. A metric for information freshness, called *age of information* or simply *age*, was introduced in [2], [3]. Consider a flow of status update packets that are sent from a source to a destination through a channel. Let $U(t)$ be the time stamp (i.e., generation time) of the newest update that the destination has received by time t. Age of information, as a function of time t, is defined as $\Delta(t) = t - U(t)$, which is the time elapsed since the newest update was generated.

In recent years, there have been a lot of research efforts on (i) Analyzing the distributional quantities of age $\Delta(t)$ for various network models and (ii) Controlling $\Delta(t)$ to keep the destination's information as fresh as possible, e.g., [1]–[44]. If there is a single flow of status update packets, the last generated, first served (LGFS) update transmission policy, in which the last generated packet is served the first, has been shown to be (nearly) optimal for minimizing the age process $\{\Delta(t), t \geq 0\}$ in a stochastic ordering sense for queueing networks with multiple servers or multiple hops [14]–[18]. These results hold for arbitrary packet generation times at the information source (e.g., a sensor) and arbitrary packet arrival times at the transmitter's queueing buffer; they also hold for minimizing any non-decreasing functional $\phi(\{\Delta(t), t \geq 0\})$ of the age process $\{\Delta(t), t \geq 0\}$. If packets arrive at the queue in the order of their generation times, then the LGFS policy reduces to the last come, first served (LCFS) policy, thus demonstrating the (near) age-optimality of the LCFS policy. These studies motivated us to delve deeper into the design of scheduling policies to minimize age of information in more complex networks involving *multiple flows of status update packets* and *transmission errors*, where each flow is from one source node to a destination node. In this scenario, the transmission scheduler must compare not only packets from the same flow, but also packets from different flows. Additionally, the presence of transmission errors adds an additional layer of complexity to the scheduling problem. As a result, addressing these challenges becomes crucial in achieving efficient age minimization in such systems.

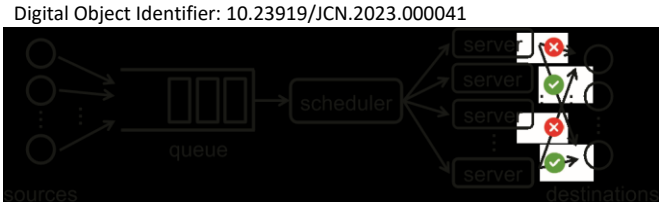


Fig. 1. System model.

In this paper, we investigate age-optimal scheduling in *continuous-time* and *discrete-time* status updating systems that involve *multiple flows*, *multiple servers*, and *transmission errors*, as illustrated in Fig. 1. Each server can transmit packets to their respective destinations, one packet at a time. Different servers are not allowed to simultaneously transmit packets from the same flow. We assume that the packet generation and arrival times are *synchronized* across the flows. In other words, when a packet from flow n arrives at the queue at time A_i, with its generation time denoted as S_i (where $S_i \leq A_i$), one corresponding packet from each flow simultaneously received at time A_i, and all of these packets were generated at the same time S_i. In practice, synchronized packet generations and arrivals occur when there is a single source and multiple destinations (e.g., [22]), or in periodic sampling where multiple sources are synchronized by the same clock, which is common in monitoring and control systems (e.g., [45], [46]). We develop a unifying sample-path approach and use it to show that the proposed scheduling policies can achieve optimal or near-optimal age performance in a quite strong sense (i.e., in terms

Manuscript received July 23, 2023; revised September 3, 2023; approved for publication by Yin Sun, Guest Editor, September 5, 2023.

This paper was presented in part at the IEEE INFOCOM Age of Information (AoI) Workshop in 2018 [1].

Y. Sun's work is supported in part by the NSF grant CNS-2239677 and the ARO grant W911NF-21-1-0244.

Y. Sun is with the Department of ECE, Auburn University, Auburn, AL 36849 USA, email: yzs0078@auburn.edu.

S. Kompella is with Nexcepta Inc., Gaithersburg, MD 20878 USA, e-mail: sk@ieee.org.

Y. Sun is the corresponding author.

1229-2370/23/$10.00 © 2023 KICS

of stochastic ordering of agepenalty stochastic processes). The contributions of this paper are summarized as follows:

- Let $\mathbf{\Delta}(t)$ denote the age vector of multiple flows. We introduce an age penalty function $p_t(\mathbf{\Delta}(t))$ to represent the level of dissatisfaction for having aged information at the destinations at time t, where p_t can be any *timedependent*, *symmetric*, and *non-decreasing* function of the age vector $\mathbf{\Delta}(t)$.
- For continuous-time status updating systems with one or multiple flows, one or multiple servers, and *i.i.d.* exponential transmission times, we propose a *preemptive, maximum age first, last generated first served (P-MAFLGFS) scheduling policy*.[1] If the packet generation and arrival times are synchronized across the flows, then for any age penalty function p_t defined above, any number of flows, any number of servers, any synchronized packet generation and arrival times, and regardless the presence of transmission errors or not, the P-MAF-LGFS policy is proven to minimize the continuous-time age penalty process $\{p_t(\mathbf{\Delta}(t)), t \geq 0\}$ among all causal policies in a stochastic ordering sense (see Theorem 1 and Corollary 1). Theorem 1 is more general than [1, Theorem 1], as the latter was established for the special case of single-server status updating systems without transmission errors. In addition, if packet replication is allowed, we show that a *preemptive, maximum age first, last generated first served scheduling policy with packet replications (PMAF-LGFS-R)* is age-optimal for minimizing the age penalty process $\{p_t(\mathbf{\Delta}(t)), t \geq 0\}$ in terms of stochastic ordering (see Corollary 2).
- For continuous-time status updating systems with one or multiple flows, one or multiple servers, and *i.i.d.* newbetterthan-used (NBU) transmission times (which include exponential transmission times as a special case), ageoptimal multi-flow scheduling is quite difficult to achieve. In this case, we consider an age lower bound called the *age of served information* and propose a *non-preemptive, maximum age of served information first, last generated first served (NP-MASIF-LGFS) scheduling policy*. The NP-MASIF-LGFS policy is shown to be near ageoptimal. Specifically, it is within an additive gap from the optimum for minimizing the expected time-average of the average age of the flows, where the gap is equal to the mean transmission time of one packet (see Theorem 2 and Corollary 3). This additive sub-optimality gap is quite small.
- For discrete-time status updating systems with one or multiple flows and one or multiple servers, we propose a *discrete time, maximum age first, last generated first served (DT-MASIF-LGFS) scheduling policy*. If the packet generation and arrival times are synchronized across the flows, then for any age penalty function p_t, any number of flows, any number of servers, any synchronized packet generation and arrival times, and regardless the presence of transmission errors or not, the DT-MAFLGFS policy is proven to minimize the discrete-time age penalty process $\{p_t(\mathbf{\Delta}(t)), t = 0, T_s, 2T_s, \cdots\}$ among all causal policies in a stochastic ordering sense, where T_s is the fundamental time unit of the discrete-time systems (see Theorem 3).

Our results can be potentially applied to: (i) Cloud-hosted Web services where the servers in Fig. 1 represent a pool of threads (each for a TCP connection) connecting a front-end proxy node to clients [47], (ii) Industrial robotics and factory automation systems where multiple sensor-output flows are sent to a wireless AP and then forwarded to a system monitor and/or controller [48], and (iii) Multi-access edge computing (MEC) that can process fresh data (e.g., data for video analytics, location services, and IoT) locally at the very edge of the mobile network.

II. Related Work

The age of information concept has attracted a significant surge of research interest; see, e.g., [1]–[43] and a recent survey [44]. Initially, research efforts were centered on analyzing and comparing the age performance of different queueing disciplines, such as first-come, first-served (FCFS) [3], [5], [9], [11], preemptive and non-preemptive LCFS [4], [20], and packet management [8], [10]. In [14]–[18], a sample-path approach was developed to prove that LGFS-type policies are optimal or near-optimal for minimizing a broad class of age metrics in multi-server and multi-hop queueing networks with a single packet flow. When packets arrive in the order of their generation times, the LGFS policy becomes the well-known LCFS policy. Hence, the LCFS policy is (near) age-optimal in these queueing networks.

In recent years, researchers have expanded the aforementioned studies to consider age minimization in multi-flow discrete-time status updating systems [22]–[25]. In [22], the authors utilized a sample-path method to establish the optimality of the maximum age first (MAF) policy in minimizing the time-averaged sum age of multiple flows. This investigation focused on discrete-time systems with periodic arrivals and a single broadcast channel, which is susceptible to *i.i.d.* transmission errors. Moreover, in [23], a Markov decision process (MDP) approach was adopted to prove that the MAF policy minimizes the time-averaged sum age of multiple flows in discrete-time systems with Bernoulli arrivals, a single broadcast channel, and no buffer. In this bufferless setup, arriving packets are discarded if they cannot be transmitted immediately in the arriving time slot. In [24], the authors studied discrete-time systems with multiple flows and multiple ON/OFF channels, where the state of each channel (ON/OFF) is known for making scheduling decisions. It was demonstrated that a max-age matching policy is asymptotically optimal for minimizing non-decreasing symmetric functions of the age of the flows as the numbers of flows and channels increase. In [25], it was shown that the MAF policy minimizes the maximum age of multiple flows in discretetime systems with periodic

[1] This new P-MAF-LGFS policy is suitable for both single-server and multiserver systems, whereas the original P-MAF-LGFS policy, as presented in [1], was specifically tailored for single-server scenarios.

arrivals and a single broadcast channel susceptible to *i.i.d.* transmission errors, where the transmission error probability may vary across the flows. In [49], a sample-path method was employed to demonstrate that the round-robin policy minimizes a service regularity metric called *time-since-last-service* in discrete-time systems with multiple flows and transmission errors. In the definition of time-since-last-service, a user can receive service even if its queue is empty. Consequently, time-since-last-service bears similarities to the age of information concept, albeit these two metrics are different. The present paper, alongside its conference version [1], complements the aforementioned studies in several essential ways: (i) It considers general time-dependent, symmetric, and non-decreasing age penalty functions p_t. (ii) Both continuous-time and discrete-time systems with multiple flows, multiple channels (a.k.a. servers), and transmission errors are investigated. (iii) The paper establishes near ageoptimal scheduling results in scenarios where achieving ageoptimality is inherently challenging.

III. System Model

A. Notations and Definitions

We use lower case letters such as x and $\boldsymbol{x}$, respectively, to represent deterministic scalars and vectors. In the vector case, a subscript will index the components of a vector, such as x_i. We use $x_{[i]}$ to denote the ith largest component of vector $\boldsymbol{x}$. Let $\mathbf{0}$ denote a vector with all 0 components. A function $f: \mathrm{R}^n \to \mathrm{R}$ is termed *symmetric* if $f(\boldsymbol{x}) = f(x_{[1]},\cdots,x_{[n]})$ for all $\boldsymbol{x} \in \mathrm{R}^n$. A function $f: \mathrm{R}^n \to \mathrm{R}$ is termed *separable* if there exists functions $f_1,\cdots,f_n$ of one variable such that [redacted] for all $\boldsymbol{x} \in \mathrm{R}^n$. The composition of functions f and g is denoted by $f \circ g(\boldsymbol{x}) = f(g(\boldsymbol{x}))$. For any n-dimensional vectors $\boldsymbol{x}$ and $\boldsymbol{y}$, the elementwise vector ordering $x_i \le y_i$, $i = 1,\cdots,n$, is denoted by $\boldsymbol{x} \le \boldsymbol{y}$. Let A and U denote sets and events. For all random variable X and event A, let $[X|\mathrm{A}]$ denote a random variable with the conditional distribution of X for given A. We will need the following definitions:

Definition 1. *Stochastic ordering of random variables [50]*: A random variable X is said to be *stochastically smaller* than another random variable Y, denoted by $X \le_{\mathrm{st}} Y$, if

$$\Pr(X > t) \le \Pr(Y > t),\ \forall\ t \in \mathrm{R}. \tag{1}$$

Definition 2. *Stochastic ordering of random vectors [50]*: A set $\mathrm{U} \subseteq \mathrm{R}^n$ is called *upper*, if $\boldsymbol{y} \in \mathrm{U}$ whenever $\boldsymbol{y} \ge \boldsymbol{x}$ and $\boldsymbol{x} \in \mathrm{U}$. Let $\boldsymbol{X}$ and $\boldsymbol{Y}$ be two n-dimensional random vectors, $\boldsymbol{X}$ is said to be *stochastically smaller* than $\boldsymbol{Y}$, denoted by $\boldsymbol{X} \le_{\mathrm{st}} \boldsymbol{Y}$, if

$$\Pr(\boldsymbol{X} \in \mathrm{U}) \le \Pr(\boldsymbol{Y} \in \mathrm{U}) \text{ for all upper sets } \mathrm{U} \subseteq \mathrm{R}^n. \tag{2}$$

Definition 3. *Stochastic ordering of stochastic processes [50]*: Let $\{X(t), t \in [0,\infty)\}$ and $\{Y(t), t \in [0,\infty)\}$ be two stochastic processes, $\{X(t), t \in [0,\infty)\}$ is said to be *stochastically smaller* than $\{Y(t), t \in [0,\infty)\}$, denoted by $\{X(t), t \in [0,\infty)\} \le_{\mathrm{st}} \{Y(t), t \in [0,\infty)\}$, if for all integer n and $0 \le t_1 < t_2 < \cdots < t_n$, it holds that

$$(X(t_1), X(t_2), \cdots, X(t_n)) \le_{\mathrm{st}} (Y(t_1), Y(t_2), \cdots, Y(t_n)). \tag{3}$$

A functional is a mapping from functions to real numbers. A functional ϕ is termed *non-decreasing* if $\phi(\{X(t), t \in [0,\infty)\}) \le \phi(\{Y(t), t \in [0,\infty)\})$ whenever $X(t) \le Y(t)$ for $t \in [0,\infty)$. We remark that $\{X(t), t \in [0,\infty)\} \le_{\mathrm{st}} \{Y(t), t \in [0,\infty)\}$ if, and only if, [50]

$$\mathrm{E}[\phi(\{X(t), t \in [0,\infty)\})] \le \mathrm{E}[\phi(\{Y(t), t \in [0,\infty)\})] \tag{4}$$

holds for all non-decreasing functional ϕ, provided that the expectations in (4) exist.

B. Queueing System Model

Consider the status updating system illustrated in Fig. 1, where N flows of status update packets are sent through a queue with an infinite buffer and M servers. Let s_n and d_n denote the source and destination nodes of flow n, respectively. It is possible for multiple flows to share either the same source node or the same destination node.

A scheduler assigns packets from the transmitter's queue to servers over time. The queue contains packets from different flows, and each packet can be assigned to any available server. Each server is capable of transmitting only one packet at a time. Different servers are not allowed to simultaneously transmit packets from the same flow. The packet transmission times are independent and identically distributed (*i.i.d.*) across both servers and packets, with a finite mean $1/\mu$. The packet transmissions are susceptible to *i.i.d.* errors with an error probability $q \in [0,1)$, occurring at the end of the packet transmission time intervals. The scheduler is made aware of transmission errors once they occur. In the event of such a error, the packet is promptly returned to the queue, where it awaits the next transmission opportunity. if $q = 0$, then there is no transmission errors.

The system starts to operate at time $t = 0$. The ith packet of flow n is generated at the source node s_n at time $S_{n,i}$, arrives at the queue at time $A_{n,i}$, and is delivered to the destination d_n at time $D_{n,i}$ such that $0 \le S_{n,1} \le S_{n,2} \le \cdots$ and $S_{n,i} \le A_{n,i} \le D_{n,i}$.[2] We consider the following class of *synchronized* packet generation and arrival processes:

Definition 4. *Synchronized packet generations and arrivals:* The packet generation and arrival processes are said to be *synchronized* across the N flows, if there exist two sequences $\{S_1,S_2,\cdots\}$ and $\{A_1,A_2,\cdots\}$ such that for all $i = 1,2,\cdots$, and $n = 1,\cdots,N$

$$S_{n,i} = S_i,\ A_{n,i} = A_i. \tag{5}$$

We note that the sequences $\{S_1,S_2,\cdots\}$ and $\{A_1,A_2,\cdots\}$ in (5) are *arbitrary*. Hence, *out-of-order arrivals*, e.g., $S_i < S_{i+1}$ but $A_i > A_{i+1}$, are allowed. In the special case that the system has a single flow ($N = 1$), the packet generation times $S_{n,1}$ and arrival times $A_{n,1}$ of this flow are arbitrarily given without any

[2] This paper allows $S_{n,i} \le A_{n,i}$, which is more general than the conventional assumption $S_{n,i} = A_{n,i}$ adopted in related literature.

constraint. Age-optimal scheduling in this special case has been previously studied in [14]–[17].

Let π represent a scheduling policy that determines how to assign packets from the queue to servers over time. Let Π denote the set of all *causal* scheduling policies in which the scheduling decisions are made based on the history and current states of the system. A scheduling policy is said to be *preemptive* if a busy server can stop the transmission of the current packet and start sending another packet at any time; the preempted packet is stored back to the queue, waiting to be sent at a later time. A scheduling policy is said to be *non-preemptive* if each server must complete the transmission of the current packet before initiating the service of another packet. A scheduling policy is said to be *work-conserving* if all servers remain busy whenever the queue contains packets waiting to be processed. We use Π_{np} to denote the set of nonpreemptive and causal scheduling policies, where $\Pi_{np} \subset \Pi$. Let

$$\mathrm{I} = \{S_i, A_i,\ i = 1,2,\cdots\}, \tag{6}$$

denote the synchronized packet generation and arrival times of the flows. We assume that the packet generation/arrival times I, the packet transmission times, and the transmission errors are governed by three *mutually independent* stochastic processes, none of which are influenced by the scheduling policy.

C. Age Metrics

Among the packets that have been delivered to the destination d_n of flow n by time t, the freshest packet was generated at time

$$U_n(t) = \max\{S_{n,i} : D_{n,i} \le t\}. \tag{7}$$

Age of information, or simply *age*, for flow n is defined as [2], [3]

$$\Delta_n(t) = t - U_n(t) = t - \max\{S_{n,i} : D_{n,i} \le t\}, \tag{8}$$

which is the time difference between the current time t and the generation time $U_n(t)$ of the freshest packet currently available at destination d_n. Because $S_{n,i} \le D_{n,i}$, one can get $\Delta_n(t) \ge 0$ for all flow n and time t. Let $\mathbf{\Delta}(t) = (\Delta_1(t),\cdots,\Delta_N(t)) \in [0,\infty)^N$ be the age vector of the N flows at time t.

We introduce an *age penalty function* $p(\mathbf{\Delta}) = p \circ \mathbf{\Delta}$ to represent the level of dissatisfaction for having aged information at the N destinations, where $p : [0,\infty)^N \to \mathrm{R}$ can be any *non-decreasing* function of the N-dimensional age vector $\mathbf{\Delta}$. Some examples of the age penalty function are:

1. The *average age* of the N flows is

[redacted] (9)

2. The *maximum age* of the N flows is

$$p_{\max}(\mathbf{\Delta}) = \max_{n=1,\cdots,N} \Delta_n. \tag{10}$$

3. The *mean square age* of the N flows is

[redacted] (11)

4. The *l-norm of the age vector* of the N flows is

[redacted] (12)

5. The *sum of per-flow age penalty functions* is

$$p_{\text{sum-penalty}}(\mathbf{\Delta}) = \sum_{n=1}^{N} g(\Delta_n), \tag{13}$$

where $g : [0,\infty) \to \mathrm{R}$ is a *non-decreasing* function. Practical applications of non-decreasing age functions can be found in [32], [33], [34], [36], [44].

In this paper, we consider a class of *symmetric* and *nondecreasing* age penalty functions, i.e.,

$$\mathrm{P}_{\text{sym}} = \{p : [0,\infty)^N \to \mathrm{R} \text{ is symmetric and non-decreasing}\}.$$

This is a fairly large class of age penalty functions, where the function p can be discontinuous, non-convex, or non-separable. It is easy to see

$$\{p_{\text{avg}}, p_{\max}, p_{\text{ms}}, p_{l\text{-norm}}, p_{\text{sum-penalty}}\} \subset \mathrm{P}_{\text{sym}}. \tag{14}$$

In this paper, we consider both continuous-time and discrete-time status updating systems. In the continuous-time setting, time $t \in [0,\infty)$ can take any positive value and the packet transmission times are *i.i.d.* continuous random variables. On the other hand, in the discrete-time setting, time is quantized into multiples of a fundamental time unit T_s, i.e., $t \in \{0, T_s, 2T_s, \cdots\}$, and each packet's transmission time is fixed and equal to T_s. Consequently, the variables $S_{n,i}, A_{n,i}, D_{n,i}, t, U_n(t), \Delta_n(t)$ are all multiples of T_s. In realistic discrete-time systems, service preemption is not allowed.

Let $\Delta_{n,\pi}(t)$ denote the age of flow n achieved by scheduling policy π and $\mathbf{\Delta}_\pi(t) = (\Delta_{1,\pi}(t),\cdots,\Delta_{N,\pi}(t))$. In the continuous-time case, we assume that the initial age $\mathbf{\Delta}_\pi(0^-)$ at time $t = 0^-$ remains the same for all scheduling policies $\pi \in \Pi$, where $t = 0^-$ is the moment right before $t = 0$. In the discrete-time case, we assume that the initial age $\mathbf{\Delta}_\pi(0)$ at time $t = 0$ remains the same for all scheduling policies $\pi \in \Pi$.

The results in this paper remain true even if the age penalty function p_t varies over time t. For example, it is allowed that $p_t = p_{\text{avg}}$ for $0 \le t \le 100$ and $p_t = p_{\max}$ for $100 < t \le 200$. In the continuous-time case, we use $\{p_t \circ \mathbf{\Delta}_\pi(t), t \in [0,\infty)\}$ to represent the age-penalty stochastic process formed by the *time-dependent* penalty function p_t of the age vector $\mathbf{\Delta}_\pi(t)$ under scheduling policy π. In the discrete-time case, the agepenalty stochastic process is denoted by $\{p_t \circ \mathbf{\Delta}_\pi(t), t = 0, T_s, 2T_s, \cdots\}$.

IV. Multi-flow Status Update Scheduling: The Continuous-time Case

In this section, we investigate multi-flow scheduling in continuous-time status updating systems. We first consider a

system setting with multiple servers and exponential transmission times, where an age-optimal scheduling result is established. Next, we study a more general system setting with multiple servers and NBU transmission times. In the second setting, age optimality is inherently difficult to achieve and we present a near age-optimal scheduling result.

A. Multiple Flows, Multiple Servers, Exponential Service Times

To address the multi-flow scheduling problem, we consider a flow selection discipline called *MAF* [6], [22], [23], in which *the flow with the maximum age is served first, with ties broken arbitrarily*.

For multi-flow single-server systems, a scheduling policy is defined by combining the Preemptive, MAF, and LGFS service disciplines as follows:

Definition 5. *P-MAF-LGFS policy:* This is a work-conserving scheduling policy for multiple-server, continuous-time systems with synchronized packet generations and arrivals. It operates as follows:

1. If the queue is not empty, a server is assigned to process the most recently generated packet from the flow with the maximum age, with ties broken arbitrarily.
2. The next server is assigned to process the most recently generated packet from the flow with the second maximum age, with ties broken arbitrarily.
3. This process continues until either (i) The most recently generated packet of every flow is under service or has been delivered, or (ii) All servers are busy.
4. If the most recently generated packet of every flow is under service or has been delivered, the remaining servers can be arbitrarily assigned to send the remaining packets in the queue, until the queue becomes empty.
5. When fresher packets arrive, the scheduler can preempt the packets that are currently under service and assign the new packets to servers following Steps 1-4 above. The preempted packets are then returned to the queue, where they await their turn to be transmitted at a later time.

The following observation provides useful insights into the operations of the P-MAF-LGFS policy: Due to synchronized packet generations and arrivals, when the most recently generated packet of flow n is successfully delivered in the P-MAFLGFS policy, flow n must have the *minimum* age among the N flows. Conversely, if flow n does not have the *minimum* age among all the flows, its most recently generated packet must be undelivered. Hence, in the P-MAF-LGFS policy, the most recently generated packet from a flow that does not have the *minimum* age is always available to be scheduled.

The above P-MAF-LGFS policy is suitable for use in both single-server and multiple-server systems. It extends the original single-server P-MAF-LGFS policy introduced in [1] to encompass the more general multi-server scenario.

The age optimality of the P-MAF-LGFS policy is established in Theorem 1 and Corollary 1 below.

Theorem 1. (Continuous-time, multiple flows, multiple servers, exponential transmission times with transmission errors) In continuous-time status updating systems, if (i) The transmission errors are *i.i.d.* with an error probability $q \in [0,1)$, (ii) The packet generation and arrival times are synchronized across the N flows, and (iii) The packet transmission times are exponentially distributed and *i.i.d.* across packets, then it holds that for all I, all $p_t \in \mathrm{P}_{\mathrm{sym}}$, and all $\pi \in \Pi$

$$[\{p_t \circ \mathbf{\Delta}_{\text{P-MAF-LGFS}}(t), t \in [0,\infty)\}|\mathrm{I}] \leq_{\mathrm{st}} [\{p_t \circ \mathbf{\Delta}_{\pi}(t), t \in [0,\infty)\}|\mathrm{I}], \quad (15)$$

or equivalently, for all I, all $p_t \in \mathrm{P}_{\mathrm{sym}}$, and all non-decreasing functional ϕ

$$\mathrm{E}[\phi(\{p_t \circ \mathbf{\Delta}_{\text{P-MAF-LGFS}}(t), t \in [0,\infty)\})|\mathrm{I}] = \min_{\pi\in\Pi} \mathrm{E}[\phi(\{p_t \circ \mathbf{\Delta}_{\pi}(t), t \in [0,\infty)\})|\mathrm{I}], \quad (16)$$

provided that the expectations in (16) exist.

Proof. See Appendix A. □

According to Theorem 1, for any age penalty function in $\mathrm{P}_{\mathrm{sym}}$, any number of flows N, any number of servers M, any synchronized packet generation and arrival times in I, and regardless the presence of *i.i.d.* transmission errors or not, the P-MAF-LGFS policy minimizes the stochastic process $[\{p_t \circ \mathbf{\Delta}_{\pi}(t), t \in [0,\infty)\}|\mathrm{I}]$ among all causal policies in terms of stochastic ordering. Theorem 1 is more general than [1, Theorem 1], as the latter was established for the special case of single-server systems without transmission errors.

By considering a mixture over the different realizations of I, it can be readily deduced from Theorem 1 that

Corollary 1. Under the conditions of Theorem 1, it holds that for all $p_t \in \mathrm{P}_{\mathrm{sym}}$ and all $\pi \in \Pi$

$$\{p_t \circ \mathbf{\Delta}_{\text{P-MAF-LGFS}}(t), t \in [0,\infty)\} \leq_{\mathrm{st}} \{p_t \circ \mathbf{\Delta}_{\pi}(t), t \in [0,\infty)\}, \quad (17)$$

or equivalently, for all $p_t \in \mathrm{P}_{\mathrm{sym}}$ and all non-decreasing functional ϕ

$$\mathrm{E}[\phi(\{p_t \circ \mathbf{\Delta}_{\text{P-MAF-LGFS}}(t), t \in [0,\infty)\})] = \min_{\pi\in\Pi} \mathrm{E}[\phi(\{p_t \circ \mathbf{\Delta}_{\pi}(t), t \in [0,\infty)\})], \quad (18)$$

provided that the expectations in (18) exist.

Corollary 1 states that the P-MAF-LGFS policy minimizes the stochastic process $\{p_t \circ \mathbf{\Delta}_{\pi}(t), t \in [0,\infty)\}$ in a stochastic ordering sense, outperforming all other causal policies.

1) Status update scheduling with packet replications: As discussed in Section III-B, our study has been centered on a scenario where different servers are not allowed to simultaneously transmit packets from the same flow. In this context, we have demonstrated the age-optimality of the P-MAFLGFS policy in Theorem 1. However, in situations where multiple servers can transmit packets from the same flow, and packet replication is permitted, it becomes possible to create multiple copies of the same packet and transmit these copies concurrently across multiple servers. The packet is considered delivered once any one of these copies is successfully delivered; at that point, the other copies are canceled to release the servers. If the packet service times follow an *i.i.d.*

exponential distribution with a service rate of μ, the N servers can be equivalently viewed as a single, faster server with exponential service times and a higher service rate of $N\mu$. Additionally, this fast server exhibits *i.i.d.* transmission errors with an error probability q. Our study also addresses this scenario.

Definition 6. *P-MAF-LGFS-R:* In this policy, the last generated packet from the flow with the maximum age is served the first among all packets of all flows, with ties broken arbitrarily. This packet is replicated into N copies, which are transmitted concurrently over the N servers. The packet is considered delivered once any one of these N copies is successfully delivered; at that point, the other copies are canceled to release the servers.

By applying Theorem 1 to this particular scenario with a single, faster server, we derive the following corollary.

Corollary 2. Under the conditions of Theorem 1, if packet replication is allowed, then it holds that for all I, all $p_t \in \mathrm{P}_{\mathrm{sym}}$, and all $\pi \in \Pi$

$$[\{p_t \circ \mathbf{\Delta}_{\text{P-MAF-LGFS-R}}(t), t \in [0,\infty)\}|\mathrm{I}] \leq_{\mathrm{st}} [\{p_t \circ \mathbf{\Delta}_{\pi}(t), t \in [0,\infty)\}|\mathrm{I}], \quad (19)$$

or equivalently, for all I, all $p_t \in \mathrm{P}_{\mathrm{sym}}$, and all non-decreasing functional ϕ

$$\mathrm{E}[\phi(\{p_t \circ \mathbf{\Delta}_{\text{P-MAF-LGFS-R}}(t), t \in [0,\infty)\})|\mathrm{I}] = \min_{\pi\in\Pi} \mathrm{E}[\phi(\{p_t \circ \mathbf{\Delta}_{\pi}(t), t \in [0,\infty)\})|\mathrm{I}], \quad (20)$$

provided that the expectations in (20) exist.

B. Multiple Flows, Multiple Servers, NBU Service Times

Next, we consider a more general system setting with multiple servers and a class of NBU transmission time distributions that include exponential distribution as a special case.

Definition 7. *NBU distributions:* Consider a non-negative random variable X with complementary cumulative distribution function (CCDF) $\bar{F}(x) = \Pr[X > x]$. Then, X is said to be *NBU* if for all $t,\tau \geq 0$

$$\bar{F}(\tau + t) \leq \bar{F}(\tau)\bar{F}(t). \quad (21)$$

Examples of NBU distributions include deterministic distribution, exponential distribution, shifted exponential distribution, geometric distribution, gamma distribution, and negative binomial distribution.

In the scheduling literature, optimal scheduling results were successfully established for delay minimization in singleserver queueing systems, e.g., [51], [52], but it can become inherently difficult in the multi-server cases. In particular, minimizing the average delay in deterministic scheduling problems with more than one servers is NP-hard [53]. Similarly, delay-optimal stochastic scheduling in multi-class, multiserver queueing systems is deemed to be quite difficult [54]–[56]. The key challenge in multi-class, multi-server scheduling is that *one cannot combine the capacities of all the servers to jointly process the most important packet*. Due to the same reason, age-optimal scheduling in multi-flow, multiserver systems is quite challenging. In the sequel, we consider a relaxed goal to seek for *near* age-optimal scheduling of

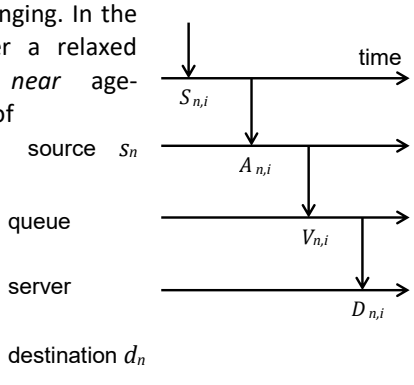


Fig. 2. An illustration of $S_{n,i}$, $A_{n,i}$, $V_{n,i}$, and $D_{n,i}$.

multiple information flows, where our proposed scheduling policy is shown to be within a small additive gap from the optimum age performance.

To establish near age optimality, we introduce another age metric named *age of served information*, denoted as $\Xi_n(t)$, which is a lower bound for age of information $\Delta_n(t)$:

Let $V_{n,i}$ be the time that the ith packet of flow n starts its service by a server, i.e., the service starting time of the ith packet of flow n. It holds that $S_{n,i} \leq A_{n,i} \leq V_{n,i} \leq D_{n,i}$, as illustrated in Fig. 2. *Age of served information* for flow n is defined as

$$\Xi_n(t) = t - \max_i\{S_{n,i} : V_{n,i} \leq t\}, \quad (22)$$

which is the time difference between the current time t and the generation time of the freshest packet that has started service by time t. Let $\mathbf{\Xi}(t) = (\Xi_1(t),\cdots,\Xi_N(t))$ be the age of served information vector at time t. Age of served information $\Xi_n(t)$ reflects the staleness of the packets that has begun service, whereas $\Delta_n(t)$ represents the staleness of the packets that has been successfully delivered to their destination. As depicted in Fig. 3, it is evident that $\Xi_n(t) \leq \Delta_n(t)$. In non-preemptive policies, the discrepancy between $\Xi_n(t)$ and $\Delta_n(t)$ solely arises from the *i.i.d.* packet transmission times. Consequently, the age of served information $\Xi_n(t)$ closely approximates the age $\Delta_n(t)$.

We propose a new flow selection discipline called *maximum age of served information first (MASIF)*, in which *the flow with the maximum age of served information is served first, with ties broken arbitrarily*. Using this discipline, we define another scheduling policy:

Definition 8. *NP-MASIF-LGFS policy:* This is a nonpreemptive, work-conserving scheduling policy for multiserver systems. It operates as follows:

1. When the queue is not empty and there are idle servers, an idle server is assigned to process the most recently generated packet from the flow with the maximum age of served information, with ties broken arbitrarily.
2. After a packet from flow n is assigned to an idle server, the server transitions into a busy state and will complete the transmission of the current packet from flow n before serving any other packet. The age of served information $\Xi_n(t)$ of flow n decreases. As a result, flow n may no longer

retain the maximum age of served information, allowing the remaining idle servers to be allocated to process other flows. The next idle server is assigned to

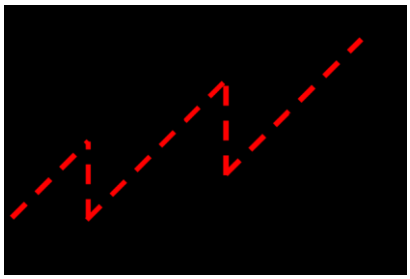


Fig. 3. The age of served information $\Xi_n(t)$ as a lower bound of age $\Delta_n(t)$.

process the most recently generated packet from the flow with the maximum age of served information, with ties broken arbitrarily.

3. This procedure continues until either all servers are busy or the queue becomes empty.

Next, we will establish the near-age optimality of the NPMASIF-LGFS policy. The following theorem shows that the age of served information obtained by the NP-MASIF-LGFS policy serves as a lower bound (in terms of stochastic ordering) for the age of all other non-preemptive and causal policies.

Theorem 2. (Continuous-time, multiple flows, multiple servers, NBU transmission times with no errors) In continuous-time status updating systems, if (i) There is no transmission errors (i.e., $q = 0$), (ii) The packet generation and arrival times are synchronized across the N flows, and (iii) The packet transmission times are NBU and *i.i.d.* across both servers and packets, then it holds that for all I, all $p_t \in \mathrm{P_{sym}}$, and all $\pi \in \Pi_{np}$[3]

$$[\{p_t \circ \Xi_{\text{NP-MASIF-LGFS}}(t), t \in [0,\infty)\}|\mathrm{I}] \leq_{\text{st}} [\{p_t \circ \Delta_\pi(t), t \in [0,\infty)\}|\mathrm{I}], \quad (23)$$

or equivalently, for all I, all $p_t \in \mathrm{P_{sym}}$, and all non-decreasing functional ϕ

$$\mathrm{E}[\phi(\{p_t \circ \Xi_{\text{NP-MASIF-LGFS}}(t), t \in [0,\infty)\})|\mathrm{I}] \leq \min_{\pi \in \Pi_{np}} \mathrm{E}[\phi(\{p_t \circ \Delta_\pi(t), t \in [0,\infty)\})|\mathrm{I}] \leq \mathrm{E}[\phi(\{p_t \circ \Delta_{\text{NP-MASIF-LGFS}}(t), t \in [0,\infty)\})|\mathrm{I}], \quad (24)$$

provided that the expectations in (24) exist.

Proof idea. In the NP-MASIF-LGFS policy, if a packet from flow n^* begins service, it implies that flow n^* possesses the *maximum* age of served information before the service starts. If the packet generation and arrival times are synchronized across the flows, flow n^* also exhibits the *minimum* age of served information after the service starts. The proof of Theorem 2 relies on this property and a sample-path argument that is developed for NBU service time distributions. See Appendix B for the details. □

[3] Recall that Π_{np} is the set of non-preemptive and causal scheduling policies.

Considering the close approximation between the age of served information $\Xi_{\text{NP-MASIF-LGFS}}(t)$ and the age of information $\Delta_{\text{NP-MASIF-LGFS}}(t)$ in (24), we can conclude that the NPMASIF-LGFS policy is near age-optimal. Furthermore, in the case of the average age metric as defined in (9) (i.e., $p_t = p_{\text{avg}}$ for all t), we can derive the following corollary:

Corollary 3. Under the conditions of Theorem 2, it holds that for all I $\min_{\pi \in \Pi_{np}} [\Delta^-_\pi|\mathrm{I}] \leq [\Delta^-_{\text{NP-MASIF-LGFS}}|\mathrm{I}] \leq$ [redacted]

where [redacted]

is the expected time-average of the average age of the N flows, and $1/\mu$ is the mean packet transmission time.

Proof. The proof of Corollary 3 is the same as that of Theorem 12 in [15] and hence is omitted here. □

By Corollary 3, the average age of the NP-MASIF-LGFS policy is within an additive gap from the optimum, where the gap $1/\mu$ is invariant of the packet arrival and generation times I, the number of flows N, and the number of servers M.

Similar to Corollary 1, by taking a mixture over the different realizations of I, one can remove the condition I from (23), (24), (25), and (26).

The sampling-path argument utilized in the proof of Theorem 2 can effectively handle multiple flows, multiple servers, and *i.i.d.* NBU transmission time distributions. This is achieved by establishing a coupling between the start time of packet transmissions in the NP-MASIF-LGFS policy and the completion time of packet transmissions in any other work-conserving policy from Π_{np}. However, extending this sampling-path argument to encompass the scenario of *i.i.d.* transmission errors poses additional challenges that are currently difficult to overcome.

V. Multi-flow Status Update Scheduling: The Discrete-time Case

In this section, we investigate age-optimal scheduling in discrete-time status updating systems, where the variables $S_{n,i}, A_{n,i}, D_{n,i}, t, U_n(t), \Delta_n(t)$ are all multiples of the period T_s, the transmission time of each packet is fixed as T_s, and the packet submissions are subject to *i.i.d.* errors with an error probability $q \in [0,1)$. Service preemption is not allowed in discrete-time systems.

For multiple-server, discrete-time systems, a scheduling policy is defined by combining the MAF and LGFS service disciplines as follows:

Definition 9. *DT-MAF-LGFS policy:* This is a workconserving scheduling policy for multiple-server, discrete-time systems with synchronized packet generations and arrivals. It operates as follows:

1. When time t is a multiple of period T_s, if the queue is not empty, an idle server is assigned to process the most recently generated packet from the flow with the maximum age, with ties broken arbitrarily.

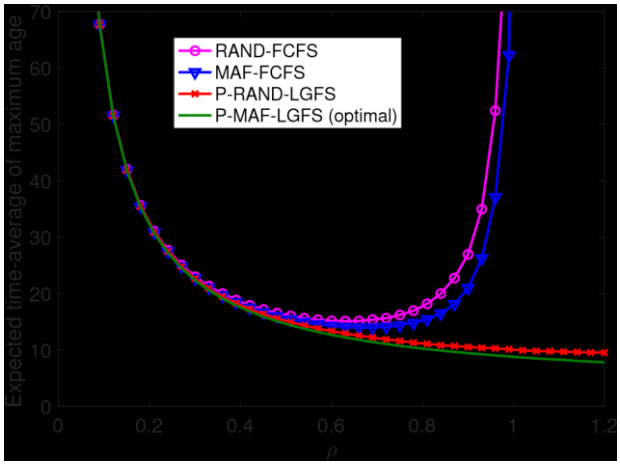


Fig. 4. Expected time-average of the maximum age of 3 flows in a system with a single server and *i.i.d.* exponential transmission times.

2. The next idle server is assigned to process the most recently generated packet from the flow with the second maximum age, with ties broken arbitrarily.
3. This process continues until either (i) The most recently generated packet of each flow is under service or has been delivered, or (ii) All servers are busy.
4. If the most recently generated packet of each flow is under service or has been delivered, and there are additional idle servers, then these servers can be arbitrarily assigned to send the remaining packets in the queue, until the queue becomes empty.

One can observe that the DT-MAF-LGFS policy for discrete-time systems is similar to the P-MAF-LGFS policy designed for continuous-time systems.

The age optimality of the DT-MAF-LGFS policy is obtained in the following theorem.

Theorem 3. (Discrete-time, multiple flows, multiple servers, constant transmission times with transmission errors) In discrete-time status updating systems, if (i) The transmission errors are *i.i.d.* with an error probability $q \in [0,1)$, (ii) The packet generation and arrival times are synchronized across the N flows, and (iii) The packet transmission times are fixed as T_s, then it holds that for all I, all $p_t \in \mathrm{P_{sym}}$, and all $\pi \in \Pi_{np}$

$$[\{p_t \circ \mathbf{\Delta}_{\text{DT-MAF-LGFS}}(t), t = 0, T_s, 2T_s, \cdots\}|\mathrm{I}] \leq_{\text{st}} [\{p_t \circ \mathbf{\Delta}_{\pi}(t), t = 0, T_s, 2T_s, \cdots\}|\mathrm{I}], \quad (27)$$

or equivalently, for all I, all $p_t \in \mathrm{P_{sym}}$, and all non-decreasing functional ϕ

$$\mathrm{E}[\phi(\{p_t \circ \mathbf{\Delta}_{\text{DT-MAF-LGFS}}(t), t = 0, T_s, 2T_s, \cdots\})|\mathrm{I}]$$

provided that the expectations in (28) exist.

Proof. See Appendix C. □

According to Theorem 3, the DT-MAF-LGFS policy minimizes the stochastic process $[\{p_t \circ \mathbf{\Delta}_{\pi}(t), t = 0, T_s, 2T_s, \cdots)\}|\mathrm{I}]$ in terms of stochastic ordering within discrete-time status updating systems. This optimality result

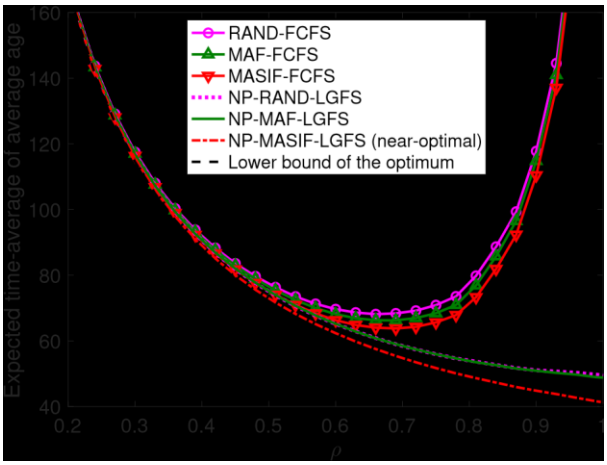


Fig. 5. Expected time-average of the average age of 50 flows in a system with 3 servers and *i.i.d.* NBU service times.

holds for any age penalty function in $\mathrm{P_{sym}}$, any number of flows N, any number of servers M, any synchronized packet generation and arrival times in I, and regardless the existence of *i.i.d.* transmission errors.

Theorem 3 generalizes [22, Theorem 1], by allowing for multiple servers and a broader range of age penalty functions. Similar to Corollary 1, one can remove the condition I from (27) and (28).

VI. Numerical Results

In this section, we evaluate the age performance of several multi-flow scheduling policies. These scheduling policies are defined by combining the flow selection disciplines {MAF, MASIF, RAND} and the packet selection disciplines {FCFS, LGFS}, where RAND represents randomly choosing a flow among the flows with un-served packets. The packet generation times S_i follow a Poisson process with rate λ, and the time difference $(A_i - S_i)$ between packet generation and arrival is equal to either 0 or $4/\lambda$ with equal probability. The mean transmission time of each server is set as $\mathrm{E}[X] = 1/\mu = 1$. Hence, the traffic intensity is $\rho = \lambda N/M$, where N is the number of flows and M is the number of servers.

Fig. 4 illustrates the expected time-average of the maximum age $p_{\max}(\mathbf{\Delta}(t))$ of 3 flows in a system with a single server and *i.i.d.* exponential transmission times. One can see that the P-MAF-LGFS policy has the best age performance and its age is quite small even for $\rho > 1$, in which case the queue is actually unstable. On the other hand, both the RAND and FCFS disciplines have much higher age. Note that there is no need for preemptions under the FCFS discipline. Fig. 5 plots the expected time-average of the average age $p_{\text{avg}}(\mathbf{\Delta}(t))$ of 50 flows in a system with 3 servers and *i.i.d.* NBU transmission

times. In particular, the transmission time X follows the following shifted exponential distribution:

$$\Pr[X > x] = \exp[1, \ -^{3}(x - {}^{1})], \quad \text{if } xx <\geq 3\underline{1}_{1}; \tag{29}$$

One can observe that the NP-MASIF-LGFS policy is better than the other policies, and is quite close to the age lower bound where the gap from the lower bound is no more than the mean transmission time $\mathrm{E}[X] = 1$. One interesting observation is that the NP-MASIF-LGFS policy is better than the NPMAF-LGFS policy for NBU transmission times. The reason behind this is as follows: When multiple servers are idle, the NP-MAF-LGFS policy will assign these servers to process multiple packets from the flow with the maximum age (say flow n). This reduces the age of flow n, but at a cost of postponing the service of the flows with the second and third maximum ages. On the other hand, in the NP-MASIF-LGFS policy, once a packet from the flow with the maximum age of served information (say flow m) is assigned to a server, the age of served information of flow m drops greatly, and the next server will be assigned to process the flow with the second maximum age of served information. As shown in [57], [58], using multiple parallel servers to process different flows is beneficial for NBU service times.

VII. Conclusion

We have proposed causal scheduling policies and developed a unifying sample-path approach to prove that these scheduling policies are (near) optimal for minimizing age of information in continuous-time and discrete-time status updating systems with multiple flows, multiple servers, and transmission errors.

Acknowledgement

We appreciate Elif Uysal's support throughout this endeavor. Additionally, we thank the anonymous reviewers for their valuable comments.

References

[1] Y. Sun, E. Uysal-Biyikoglu, and S. Kompella, "Age-optimal updates of multiple information flows," in *Proc. IEEE INFOCOM WKSHPS*, Apr. 2018.

[2] X. Song and J. W. S. Liu, "Performance of multiversion concurrency control algorithms in maintaining temporal consistency," in *Proc. CMPSAC*, Oct. 1990.

[3] S. K. Kaul, R. D. Yates, and M. Gruteser, "Real-time status: How often should one update?" in *Proc. IEEE INFOCOM*, Mar. 2012.

[4] S. K. Kaul, R. D. Yates, and M. Gruteser, "Status updates through queues," in *Proc. CISS*, Mar. 2012.

[5] R. D. Yates and S. K. Kaul, "Real-time status updating: Multiple sources," in *Proc. IEEE ISIT*, Jul. 2012.

[6] B. Li, A. Eryilmaz, and R. Srikant, "On the universality of age-based scheduling in wireless networks," in *Proc. IEEE INFOCOM*, Apr. 2015.

[7] M. Costa, M. Codreanu, and A. Ephremides, "Age of information with packet management," in *Proc. IEEE ISIT*, Jun. 2014.

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

[9] 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, Mar. 2016.

[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*, Jun. 2015.

[11] L. Huang and E. Modiano, "Optimizing age-of-information in a multiclass queueing system," in *Proc. IEEE ISIT*, Jun. 2015.

[12] 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*, Apr. 2016.

[13] 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*, vol. 63, no. 11, pp. 7492–7508, Nov. 2017.

[14] 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*, Jul. 2016.

[15] A. M. Bedewy, Y. Sun, and N. B. Shroff, "Minimizing the age of information through queues," *IEEE Trans. Inf. Theory*, vol. 65, no. 8, pp. 5215–5232, Aug. 2019.

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

[17] A. M. Bedewy, Y. Sun, and N. B. Shroff, "The age of information in multihop networks," *IEEE/ACM Trans. Netw.*, vol. 27, no. 3, pp. 1248–1257, Jun. 2019.

[18] Y. Sun, I. Kadota, R. Talak, and E. Modiano, *Age of information: A new metric for information freshness*. Morgan & Claypool, 2019.

[19] A. Kosta, N. Pappas, and V. Angelakis, "Age of information: Metric of timeliness," *Foundations and Trends in Networking*, vol. 12, no. 3, pp. 162–259, 2017.

[20] R. D. Yates and S. K. Kaul, "The age of information: Real-time status updating by multiple sources," *IEEE Trans. Inf. Theory*, vol. 65, no. 3, pp. 1807–1827, Mar. 2019.

[21] A. Maatouk, Y. Sun, A. Ephremides, and M. Assaad, "Timely updates with priorities: Lexicographic age optimality," *IEEE Trans. Commun.*, vol. 70, no. 5, pp. 3020–3033, May 2022.

[22] I. Kadota, E. Uysal-Biyikoglu, R. Singh, and E. Modiano, "Minimizing age of information in broadcast wireless networks," in *Proc. Allerton*, Sep. 2016.

[23] Y.-P. Hsu, E. Modiano, and L. Duan, "Scheduling algorithms for minimizing age of information in wireless broadcast networks with random arrivals," *IEEE Trans. Mob. Comput.*, vol. 19, no. 12, pp. 2903–2915, Dec. 2020.

[24] V. Tripathi and S. Moharir, "Age of information in multi-source systems," in *Proc. IEEE GLOBECOM*, Dec. 2017.

[25] A. Srivastava, A. Sinha, and K. Jagannathan, "On minimizing the maximum age-of-information for wireless erasure channels," in *Proc. IEEE/IFIP WiOPT*, Jun. 2019.

[26] Q. He, D. Yuan, and A. Ephremides, "Optimal link scheduling for age minimization in wireless systems," *IEEE Trans. Inf. Theory*, vol. 64, no. 7, pp. 5381–5394, Jul. 2018.

[27] A. Arafa, J. Yang, S. Ulukus, and H. V. Poor, "Age-minimal transmission for energy harvesting sensors with finite batteries: Online policies," *IEEE Trans. Inf. Theory*, vol. 66, no. 1, pp. 534–556, Jan. 2020.

[28] Y. Sun, Y. Polyanskiy, and E. Uysal, "Sampling of the Wiener process for remote estimation over a channel with random delay," *IEEE Trans. Inf. Theory*, vol. 66, no. 2, pp. 1118–1135, Feb. 2020.

[29] A. Maatouk, S. Kriouile, M. Assaad, and A. Ephremides, "The age of incorrect information: A new performance metric for status updates," *IEEE/ACM Trans. Netw.*, vol. 28, no. 5, pp. 2215–2228, Oct. 2020.

[30] C. Kam, S. Kompella, G. D. Nguyen, J. E. Wieselthier, and A. Ephremides, "Towards an effective age of information: Remote estimation of a Markov source," in *Proc. IEEE INFOCOM WKSHPS*, Apr. 2018.

[31] Y. Sun and B. Cyr, "Information aging through queues: A mutual information perspective," in *IEEE SPAWC*, Jun. 2018.

[32] Y. Sun and B. Cyr, "Sampling for data freshness optimization: Nonlinear age functions," *J. Commun. Netw.*, vol. 21, no. 3, pp. 204–219, Jun. 2019.

[33] M. K. C. Shisher, H. Qin, L. Yang, F. Yan, and Y. Sun, "The age of correlated features in supervised learning based forecasting," in *Proc. IEEE INFOCOM WKSHPS*, May 2021.

[34] M. K. C. Shisher and Y. Sun, "How does data freshness affect real-time supervised learning?" in *Proc. ACM MobiHoc*, Oct. 2022.

[35] M. K. C. Shisher, Y. Sun, and I.-H. Hou, "Timely communications for remote inference," 2023, in preparation.
[36] M. K. C. Shisher, B. Ji, I.-H. Hou, and Y. Sun, "Learning and communications co-design for remote inference systems: Feature length selection and transmission scheduling," 2023, arXiv:2308.10094.
[37] J. Pan, A. M. Bedewy, Y. Sun, and N. B. Shroff, "Age-optimal scheduling over hybrid channels," *IEEE Trans. Mob. Comput.*, 2022, in press.
[38] J. Pan, A. M. Bedewy, Y. Sun, and N. B. Shroff, "Optimal sampling for data freshness: Unreliable transmissions with random two-way delay," *IEEE/ACM Trans. Netw.*, vol. 31, no. 1, pp. 408–420, 2022.
[39] A. M. Bedewy, Y. Sun, R. Singh, and N. B. Shroff, "Low-power status updates via sleep-wake scheduling," *IEEE/ACM Trans. Netw.*, vol. 29, no. 5, pp. 2129–2141, Oct. 2021.
[40] T. Z. Ornee and Y. Sun, "Sampling and remote estimation for the Ornstein-Uhlenbeck process through queues: Age of information and beyond," *IEEE/ACM Trans. Netw.*, vol. 29, no. 5, pp. 1962–1975, Oct. 2021.
[41] A. M. Bedewy, Y. Sun, S. Kompella, and N. B. Shroff, "Optimal sampling and scheduling for timely status updates in multi-source networks," *IEEE Trans. Inf. Theory*, vol. 67, no. 6, pp. 4019–4034, Jun. 2021.
[42] H. Tang, Y. Sun, and L. Tassiulas, "Sampling of the Wiener process for remote estimation over a channel with unknown delay statistics," in *Proc. ACM MobiHoc*, Oct. 2022.
[43] T. Z. Ornee and Y. Sun, "A Whittle index policy for the remote estimation of multiple continuous Gauss-Markov processes over parallel channels," in *Proc. ACM MobiHoc*, Oct. 2023.
[44] R. D. Yates, *et al.*, "Age of information: An introduction and survey," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 5, pp. 1183–1210, May 2021.
[45] A. G. Phadke, B. Pickett, M. Adamiak, and et. al., "Synchronized sampling and phasor measurements for relaying and control," *IEEE Trans. Power Delivery*, vol. 9, no. 1, pp. 442–452, Jan. 1994.
[46] F. Sivrikaya and B. Yener, "Time synchronization in sensor networks: A survey," *IEEE Netw.*, vol. 18, no. 4, pp. 45–50, Jul. 2004.
[47] A. Fox, S. D. Gribble, Y. Chawathe, E. A. Brewer, and P. Gauthier, "Cluster-based scalable network services," *SIGOPS Oper. Syst. Rev.*, vol. 31, no. 5, pp. 78–91, Oct. 1997.
[48] V. C. Gungor and G. P. Hancke, "Industrial wireless sensor networks: Challenges, design principles, and technical approaches," *IEEE Trans. Industrial Electronics*, vol. 56, no. 10, pp. 4258–4265, Oct. 2009.
[49] R. Li, A. Eryilmaz, and B. Li, "Throughput-optimal wireless scheduling with regulated inter-service times," in *Proc. IEEE INFOCOM*, Jul. 2013. [50] M. Shaked and J. G. Shanthikumar, *Stochastic orders*. Springer, 2007.
[51] L. Schrage, "A proof of the optimality of the shortest remaining processing time discipline," *Operations Research*, vol. 16, pp. 687–690, Jun. 1968.
[52] J. R. Jackson, "Scheduling a production line to minimize maximum tardiness," management Science Research Report, University of California, Los Angeles, CA, 1955.
[53] S. Leonardi and D. Raz, "Approximating total flow time on parallel machines," in *ACM STOC*, May 1997.
[54] G. Weiss, "Turnpike optimality of Smith's rule in parallel machines stochastic scheduling," *Math. Oper. Res.*, vol. 17, no. 2, pp. 255–270, May 1992.
[55] G. Weiss, "On almost optimal priority rules for preemptive scheduling of stochastic jobs on parallel machines," *Advances in Applied Probability*, vol. 27, no. 3, pp. 821–839, Jul. 1995.
[56] M. Dacre, K. Glazebrook, and J. Nino Mora, "The achievable region ap-˜ proach to the optimal control of stochastic systems," *J. Royal Statistical Society: Series B (Statistical Methodology)*, vol. 61, no. 4, pp. 747–791, 1999.
[57] Y. Sun, C. E. Koksal, and N. B. Shroff, "On delay-optimal scheduling in queueing systems with replications," 2016, arXiv:1603.07322.
[58] Y. Sun, C. E. Koksal, and N. B. Shroff, "Near delay-optimal scheduling of batch jobs in multi-server systems," 2017,[Online]. Available: http://webhome.auburn.edu/%7eyzs0078/parallel-servers.pdf
[59] L. Kleinrock, *Queueing systems*. John Wiley and Sons, 1975, vol. 1: Theory.
[60] J. Nino-Mora, "Conservation laws and related applications," in *Wiley Encyclopedia of Operations Research and Management Science*. John Wiley & Sons, Inc., 2010.
[61] J. C. Gittins, K. Glazebrook, and R. Weber, *Multi-armed bandit allocation indices*, 2nd ed. Wiley, Chichester, NY, 2011.

Appendix A Proof of Theorem 1

Let the age vector $\mathbf{\Delta}_\pi(t)$ represent the *system state* of policy π at time t and $\{\mathbf{\Delta}_\pi(t), t \in [0,\infty)\}$ be the *state process* of policy π. For notational simplicity, let policy P represent the P-MAF-LGFS policy, which is a work-conserving policy. We first establish two lemmas that are useful to prove Theorem 1. Using the memoryless property of exponential distribution, we can obtain the following coupling lemma:

Lemma 1. *(Coupling Lemma)* In continuous-time status updating systems, consider policy P and any work-conserving policy $\pi \in \Pi$. For any given $\mathcal{I}$, if (i) The transmission errors are *i.i.d.* with an error probability $q \in [0,1)$ and (ii) The packet transmission times are exponentially distributed and *i.i.d.* across packets, then there exist policy P_1 and policy π_1 in the same probability space which satisfy the same scheduling disciplines with policy P and policy π, respectively, such that

1. the state process $\{\mathbf{\Delta}_{P_1}(t), t \in [0,\infty)\}$ of policy P_1 has the same distribution as the state process $\{\mathbf{\Delta}_P(t), t \in [0,\infty)\}$ of policy P,
2. the state process $\{\mathbf{\Delta}_{\pi_1}(t), t \in [0,\infty)\}$ of policy π_1 has the same distribution as the state process $\{\mathbf{\Delta}_\pi(t), t \in [0,\infty)\}$ of policy π,
3. if a packet from the flow with age $\Delta_{[i],P_1}(t)$ is successfully delivered at time t in policy P_1, then almost surely, a packet from the flow with age $\Delta_{[i],\pi_1}(t)$ is successfully delivered at time t in policy π_1; and vice versa.

Proof. Notice that (i) All policies have identical packet generation and arrival times $\mathcal{I}$, (ii) The packet transmission times are *i.i.d.* memoryless, and (iii) Policy P and policy π are both work-conserving. In addition, the packet generation/arrival times $\mathcal{I}$, the packet transmission times, and the transmission failures are governed by three mutually independent stochastic processes, none of which are influenced by the scheduling policy. Because of these facts, service preemption does not affect the distribution of packet delivery times. Following the inductive construction argument used in the proof of Theorem 6.B.3 in [50], one can construct the packet transmission success and failure events one by one in policy P_1 and policy π_1 to prove this lemma. In particular, since the transmission errors are *i.i.d.* and they are not influenced by the scheduling policy, it is feasible to couple the packet transmission success and failure events in policy P_1 and policy π_1 in such a way that a packet from the flow with age $\Delta_{[i],P_1}(t)$ is successfully delivered at time t in policy P_1 if, and only if, a packet from the flow with age $\Delta_{[i],\pi_1}(t)$ is successfully delivered at time t in policy π_1. The details are omitted. □

We will compare policy P_1 and policy π_1 on a sample path by using the following Lemma:

Lemma 2. *(Inductive comparison)* Suppose that a packet is delivered at time t in policy P_1 and a packet is delivered at the same time t in policy π_1. The system state of policy P_1 is $\mathbf{\Delta}_{P_1}$ before the packet delivery, which becomes [redacted] after the packet delivery. The system state of policy π_1 is $\mathbf{\Delta}_{\pi_1}$ before the packet delivery, which becomes [redacted] after the packet delivery.

Under the conditions of Lemma 1, if (i) The packet generation and arrival times are synchronized across the N flows and (ii)

$$\Delta_{[i],P_1} \le \Delta_{[i],\pi_1},\ i = 1,\cdots,N, \tag{30}$$

then

[redacted] (31)

Proof. For synchronized packet generations and arrivals, let $W(t) = \max_i\{S_i : A_i \le t\}$ be the generation time of the freshest packet of each flow that has arrived at the queue by time t. At time t, because no packet that has arrived at the queue was generated later than $W(t)$, we can obtain

[redacted] (32)

Because (i) Policy P_1 follows the same scheduling discipline with the P-MAF-LGFS policy and (ii) The packet generation and arrival times are synchronized across the N flows, the delivered packet at time t in policy P_1 must be the freshest packet generated at time $W(t)$. Hence, in policy P_1, the flow associated with the delivered packet must have the minimum age after the delivery, given by

[redacted] (33)

Combining (32) and (33), yields

[redacted] (34)

Moreover, suppose that the packet delivered at time t in policy P_1 is from the flow with age value $\Delta_{[j],P_1}$ before the packet delivery. This indicates

[redacted] (35)

[redacted] (36)

According to Lemma 1, the packet delivered at time t in policy π_1 is from the flow with age value $\Delta_{[j],\pi_1}$ before the packet delivery. Hence,

[redacted] (37)

[redacted] (38)

1

Combining (30), (35), and (37), yields

$$\Delta'_{[i],P_1} = \Delta_{[i],P_1} \le \Delta_{[i],\pi_1} = \Delta'_{[i],\pi_1},\ i = 1,2,\cdots,j-1. \tag{39}$$

Moreover, combining (30), (36), and (38), yields

$$\Delta'_{[i],P_1} = \Delta_{[i+1],P_1} \le \Delta_{[i+1],\pi_1} \le \Delta'_{[i],\pi_1},\quad i = j,2,\cdots,N-1. \tag{40}$$

Finally, (31) follows from (34), (39), and (40). This completes the proof. □

Now we are ready to prove Theorem 1.

Proof of Theorem 1. Consider any work-conserving policy $\pi \in \Pi$. By Lemma 1, there exist policy P_1 and policy π_1 satisfying the same scheduling disciplines with policy P and policy π, respectively, and the packet delivery times in policy P_1 and policy π_1 are synchronized almost surely. For any given sample path of policy P_1 and policy π_1, $\mathbf{\Delta}_{P_1}(0^-) = \mathbf{\Delta}_{\pi_1}(0^-)$ at time $t = 0^-$. We consider two cases:

Case 1: When there is no packet delivery, the age of each flow grows linearly with a slope 1.

Case 2: When a packet is successfully delivered, the evolution of the system state is governed by Lemma 2.

By induction over time, we obtain

$$\Delta_{[i],P_1}(t) \le \Delta_{[i],\pi_1}(t),\ i = 1,\cdots,N,\ t \ge 0. \tag{41}$$

For any symmetric and non-decreasing function p_t, it holds from (41) that for all sample paths and all $t \ge 0$ $p_t \circ \mathbf{\Delta}_{P_1}(t)$

$$\begin{aligned} &= p_t(\Delta_{1,P_1}(t),\cdots,\Delta_{N,P_1}(t)) \\ &= p_t(\Delta_{[1],P_1}(t),\cdots,\Delta_{[N],P_1}(t)) \\ &\le p_t(\Delta_{[1],\pi_1}(t),\cdots,\Delta_{[N],\pi_1}(t)) \\ &= p_t(\Delta_{1,\pi_1}(t),\cdots,\Delta_{N,\pi_1}(t)) \\ &= p_t \circ \mathbf{\Delta}_{\pi_1}(t). \end{aligned} \tag{42}$$

By Lemma 1, the state process $\{\mathbf{\Delta}_{P_1}(t), t \in [0,\infty)\}$ of policy P_1 has the same distribution with the state process $\{\mathbf{\Delta}_P(t), t \in [0,\infty)\}$ of policy P; the state process $\{\mathbf{\Delta}_{\pi_1}(t), t \in [0,\infty)\}$ of policy π_1 has the same distribution with the state process $\{\mathbf{\Delta}_\pi(t), t \in [0,\infty)\}$ of policy π. Hence, $\{p_t \circ \mathbf{\Delta}_{P_1}(t), t \in [0,\infty)\}$ has the same distribution with $\{p_t \circ \mathbf{\Delta}_P(t), t \in [0,\infty)\}$; $\{p_t \circ \mathbf{\Delta}_{\pi_1}(t), t \in [0,\infty)\}$ has the same distribution with $\{p_t \circ \mathbf{\Delta}_\pi(t), t \in [0,\infty)\}$. By substituting this and (42) into Theorem 6.B.30 of [50], we can obtain that (15) holds for all work-conserving policy $\pi \in \Pi$.

For non-work-conserving policies π, because the service times are exponentially distributed (i.e., memoryless) and *i.i.d.* across servers and time, server idling only postpones the delivery times of the packets. One can construct a coupling to show that for any non-work-conserving policy π, there exists a work-conserving policy π' whose age process is smaller than that of policy π in stochastic ordering; the details are omitted. As a result, (15) holds for all policies $\pi \in \Pi$. Finally, the equivalence between (15) and (16) follows from (4). This completes the proof. □

APPENDIX B PROOF OF THEOREM 2

Let $(\mathbf{\Delta}_\pi(t), \Xi_\pi(t))$ represent the *system state* of policy π at time t and $\{(\mathbf{\Delta}_\pi(t), \Xi_\pi(t)), t \in [0,\infty)\}$ be the *state process* of policy π. For notational simplicity, let policy P represent the NP-MASIF-LGFS policy, which is a non-preemptive, workconserving policy.

For single-server systems, the following *work conservation law* plays an important role in the scheduling literature (see, e.g., [59]–[61]): At any time, the expected total amount of time for completing the packets in the queue is invariant across different work-conserving policies. However, the work conservation law does not hold in multi-server systems: It often happens that some servers are busy while the rest servers are idle, which leads to inefficient utilization of the idle servers and sub-optimal scheduling performance. In the sequel, we use a *weak work-efficiency ordering* [57], [58] to compare different non-preemptive policies for multi-server systems.

Definition 10. *Weak work-efficiency ordering [57], [58]:* For any given I and a sample path realization of two nonpreemptive policies $\pi_1,\pi_2 \in \Pi_{np}$, policy π_1 is said to be *weakly more work-efficient* than policy π_2, if the following assertion is true: For each packet *j* executed in policy π_2, if

1. in policy π_2, a packet *j* starts service at time τ and completes service at time ν ($\tau \le \nu$),
2. in policy π_1, the queue is not empty during $[\tau,\nu]$, then in policy π_1, there always exists one corresponding packet j' that starts service during $[\tau,\nu]$. It is worth noting that the weak work-efficiency ordering does not require to specify which servers are used to process packets *j* and j'.

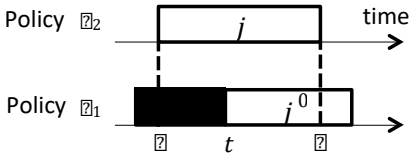


Fig. 6. An illustration of the weak work-efficiency ordering, where the service duration of a packet is indicated by a rectangle, without specifying which servers are used to process the packets. Suppose that policy π_1 is weakly more work-efficient than policy π_2. If (i) A packet *j* starts service at time τ and completes service at time ν in policy π_2, and (ii) The queue is not empty during $[\tau,\nu]$ in policy π_1, then in policy π_1 there exists one corresponding packet j' that starts service at some time *t* during $[\tau,\nu]$.

An illustration of the weak work-efficiency ordering is provided in Fig. 6. The weak work-efficiency ordering formalizes the following useful intuition for comparing two nonpreemptive policies π_1 and π_2: *If one packet j is delivered at time ν in policy π_2, then there exists one corresponding packet j' that has started its transmission shortly before time ν in policy π_1, as long as the queue is not empty.* The weak work-efficiency ordering was originally introduced for nearoptimal delay minimization in multi-server systems [57], [58]. In this paper, we use it for near-optimal age minimization in multi-server systems.

The following coupling lemma was established in [58] by using the property of NBU distributions:

Lemma 3. *(Coupling Lemma)* [58, Lemma 2] In continuoustime status updating systems, consider two non-preemptive policies $P,\pi \in \Pi_{np}$. For any given I, if (i) Policy P is work-conserving, and (ii) The packet service times are NBU, independent across the servers, and *i.i.d.* across the packets assigned to the same server, then there exist policy P_1 and policy π_1 in the same probability space which satisfy the same scheduling disciplines with policy P and policy π, respectively, such that

1. The state process $\{(\mathbf{\Delta}_{P_1}(t),\Xi_{P_1}(t)),t \in [0,\infty)\}$ of policy P_1 has the same distribution as the state process $\{(\mathbf{\Delta}_P(t),\Xi_P(t)),t \in [0,\infty)\}$ of policy P,
2. The state process $\{(\mathbf{\Delta}_{\pi_1}(t),\Xi_{\pi_1}(t)),t \in [0,\infty)\}$ of policy π_1 has the same distribution as the state process $\{(\mathbf{\Delta}_\pi(t),\Xi_\pi(t)),t \in [0,\infty)\}$ of policy π,
3. Policy P_1 is weakly more work-efficient than policy π_1 with probability one.

The proof of Lemma 3 is provided in [58].

We will compare the age of service information of policy P_1 and the age of policy π_1 on a sample path by using the following lemma:

Lemma 4. *(Inductive comparison)* Suppose that a packet starts service at time t in policy P_1 and a packet completes service (i.e., delivered to the destination) at the same time t in policy π_1. The system state of policy P_1 is $(\mathbf{\Delta}_{P_1},\Xi_{P_1})$ before the service starts, which becomes $(\mathbf{\Delta}'_{P_1},\Xi'_{P_1})$ after the service starts. The system state of policy π_1 is $(\mathbf{\Delta}_{\pi_1},\Xi_{\pi_1})$ before the service completes, which becomes [redacted] after the service completes. If the packet generation and arrival times are synchronized across the *N* flows and

$$\Xi_{[i],P_1} \le \Delta_{[i],\pi_1},\ i = 1,\cdots,N, \tag{43}$$

then

[redacted] (44)

Proof. For synchronized packet generations and arrivals, let $W(t) = \max\{S_i : A_i \le t\}$ be the generation time of the freshest packet of each flow that has arrived at the queue by time *t*. At time *t*, because no packet that has arrived at the queue was generated later than $W(t)$, we can obtain

[redacted] (45)

[redacted] (46)

Because policy P_1 follows the same scheduling discipline with the NP-MASIF-LGFS policy, each packet starts service in policy P_1 must be from the flow with the maximum age of served information $\Xi_{[1],P_1}$ (denoted as flow n^*), and the delivered packet must be the freshest packet that was generated at time $W(t)$. In other words, the age of served information of flow n^* is reduced from the maximum age of served information $\Xi_{[1],P_1}$ to the minimum age of served information [redacted], and the ages of served information of the other $(N-1)$ flows remain unchanged. Hence,

$$\Xi'_{[i],P_1} = \Xi_{[i+1],P_1},\ i = 1,\cdots,N-1, \tag{47}$$

$$\Xi'_{[N],P_1} = t - W(t). \tag{48}$$

In policy π_1, the delivered packet can be any packet from any flow. For all possible cases of policy π_1, it must hold that

[redacted] (49)

By combining (43), (47), and (49), we have

[redacted]

In addition, combining (46) and (48), yields

[redacted]

By this, (44) is proven. □

Now we are ready to prove Theorem 2.

Proof of Theorem 2. Recall that policy P is non-preemptive and work-conserving. Consider any non-preemptive policy $\pi \in \Pi_{np}$. By Lemma 3, there exist policy P_1 and policy π_1 satisfying the same scheduling disciplines with policy P and policy π,

respectively, and policy P_1 is weakly more workefficient than policy π_1 with probability one.

Next, we construct another policy [redacted] in the same probability space with policy P_1 and policy π_1: Because policy P_1 is weakly more work-efficient than policy π_1 with probability one, if

1. In policy π_1, a packet j starts service at time τ and completes service at time ν ($\tau \le \nu$),
2. In policy P_1, the queue is *not empty* during $[\tau,\nu]$, then in policy P_1, there exists one corresponding packet j' that starts service during $[\tau,\nu]$. Let $t \in [\tau,\nu]$ be the service starting time of packet j' in policy P_1, then in policy π_1', packet j is

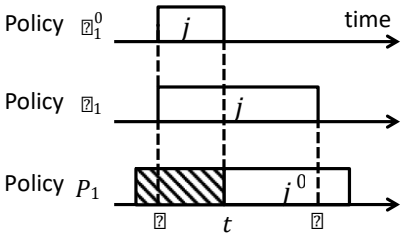


Fig. 7. An illustration of the construction of policy [redacted], where the queue is *not empty* during $[\tau,\nu]$ in policy P_1. The service completion time t of packet j in policy [redacted] is smaller than the service completion time ν of packet j in policy π, and is equal to the service starting time t of packet j' in policy P_1.

constructed to start service at time τ and complete service at time t, as illustrated in Fig. 7. On the other hand, if

1. In policy π_1, a packet j starts service at time τ and completes service at time ν ($\tau \le \nu$),
2. In policy P_1, the queue is *empty* during $[\tau,\nu]$, then in policy [redacted], packet j is constructed to start service at time τ and complete service at time ν. The initial age of policy [redacted] is chosen to be the same as that of other policies. Hence, [redacted].

The policy π_1' constructed above satisfies the following two useful properties:

Property (i): The service completion time of each packet in policy [redacted] is equal to or earlier than that in policy π. Hence,

[redacted] (50)

holds with probably one.

Property (ii): If the queue is not empty at time t in policy P_1 and a packet completes service at time t in policy [redacted], then a packet starts service at the same time t in policy P_1.

Next, we use *Property (ii)* to show that, almost surely,

[redacted] (51)

At time $t = 0^-$, because $\Xi_P(0^-) \le \mathbf{\Delta}_P(0^-)$ and [redacted], we can obtain [redacted].

This further implies that

[redacted] (52)

For any time $t > 0$, there could be three cases:

Case 1: If the queue is empty at time t in policy P_1, then (51) holds naturally at time t because all packets have started services in policy P_1 (otherwise, the queue is not empty).

Case 2: If the queue is not empty at time t in policy P_1 and a packet completes service at time t in policy [redacted], according to *Property (ii)*, a packet starts service at time t in policy P_1. Hence, the evolution of the system state before and after time t is governed by Lemma 4.

Case 3: If the queue is not empty at time t in policy P_1 and no packet completes service at time t in policy [redacted], there may exist some packet that starts service at time t in policy P_1. Therefore, the age of each flow in policy π_1' grows linearly with a slope 1 at time t; the age of served information of each flow in policy P_1 may either grow linearly with a slope 1 or drop to a lower value. By comparison, the age of served information of each flow in policy P_1 grows at a speed no faster than the age of each flow in policy [redacted].

By induction over time and considering the above three cases, (51) is proven.

Furthermore, for any symmetric and non-decreasing function p_t, it holds from (50) and (51) that for all sample paths and all $t \ge 0$ $p_t \circ \Xi_{P_1}(t)$

$$\begin{aligned}
&= p_t(\Xi_{1,P_1}(t),\cdots,\Xi_{N,P_1}(t)) \\
&= p_t(\Xi_{[1],P_1}(t),\cdots,\Xi_{[N],P_1}(t)) \\
&\le p_t(\Delta_{[1],\pi_1'}(t),\cdots,\Delta_{[N],\pi_1'}(t)) \\
&= p_t(\Delta_{1,\pi_1'}(t),\cdots,\Delta_{N,\pi_1'}(t)) \\
&= p_t \circ \mathbf{\Delta}_{\pi_1'}(t) \\
p_t &\le \circ \mathbf{\Delta}_{\pi_1}(t). \qquad (53)
\end{aligned}$$

By Lemma 3, the state process $\{(\mathbf{\Delta}_{P_1}(t),\Xi_{P_1}(t)),t \in [0,\infty)\}$ of policy P_1 has the same distribution with the state process $\{(\mathbf{\Delta}_P(t),\Xi_P(t)),t \in [0,\infty)\}$ of policy P; the state process $\{(\mathbf{\Delta}_{\pi_1}(t),\Xi_{\pi_1}(t)),t \in [0,\infty)\}$ of policy π_1 has the same distribution with the state process $\{(\mathbf{\Delta}_\pi(t),\Xi_\pi(t)),t \in [0,\infty)\}$ of policy π. Hence, $\{p_t \circ \Xi_{P_1}(t),t \in [0,\infty)\}$ has the same distribution with $\{p_t \circ \Xi_P(t),t \in [0,\infty)\}$; $\{p_t \circ \mathbf{\Delta}_{\pi_1}(t),t \in [0,\infty)\}$ has the same distribution with $\{p_t \circ \mathbf{\Delta}_\pi(t),t \in [0,\infty)\}$. By substituting this and (53) into Theorem 6.B.30 of [50], we can obtain that (23) holds for all policy $\pi \in \Pi_{np}$. According to (4), the first inequality in (24) is equivalent to (23). The second inequality in (24) holds naturally. This completes the proof. □

APPENDIX C PROOF OF THEOREM 3

Let the age vector $\mathbf{\Delta}_\pi(t) = (\Delta_{1,\pi}(t),\cdots,\Delta_{N,\pi}(t))$ represent the *system state* of policy π at time t and $\{\mathbf{\Delta}_\pi(t),t = 0,T_s,2T_s,\cdots\}$ be the *state process* of policy π. Recall that $\Delta_{[i],\pi}(t)$ is the ith largest component of the age vector $\mathbf{\Delta}_\pi(t)$. For notational simplicity, let policy P represent the DT-MAFLGFS policy, which is a non-preemptive, work-conserving policy. We first present two lemmas that are useful to prove Theorem 3.

Lemma 5. *(Coupling Lemma)* In discrete-time status updating systems, consider policy P and any non-preemptive, workconserving policy $\pi \in \Pi_{np}$. For any given I, if (i) The transmission errors are *i.i.d.* with an error probability $q \in [0,1)$ and (ii) The transmission time of each packet is equal to T_s, then there exist policy P_1 and policy π_1 in the same probability space which satisfy the same scheduling disciplines with policy P and policy π, respectively, such that

1. The state process $\{\mathbf{\Delta}_{P_1}(t), t = 0, T_s, 2T_s, \cdots\}$ of policy P_1 has the same distribution as the state process $\{\mathbf{\Delta}_P(t), t = 0, T_s, 2T_s, \cdots\}$ of policy P,
2. The state process $\{\mathbf{\Delta}_{\pi_1}(t), t = 0, T_s, 2T_s, \cdots\}$ of policy π_1 has the same distribution as the state process $\{\mathbf{\Delta}_{\pi}(t), t = 0, T_s, 2T_s, \cdots\}$ of policy π,
3. If a packet from the flow with age $\Delta_{[i],P_1}(t)$ at time t is successfully delivered at time $(t+T_s)$ in policy P_1, then almost surely, a packet from the flow with age $\Delta_{[i],\pi_1}(t)$ at time t is successfully delivered at time $(t + T_s)$ in policy π_1; and vice versa.

Proof. By employing the inductive construction argument used in the proof of Theorem 6.B.3 in [50], one can construct the packet transmission success and failure events one by one in policy P_1 and policy π_1 to prove this lemma. In particular, since the transmission errors are *i.i.d.* and they are not influenced by the scheduling policy, it is feasible to couple the packet transmission success and failure events in policy P_1 and policy π_1 in such a way that a packet from the flow with age $\Delta_{[i],P_1}(t)$ at time t is successfully delivered at time $(t+T_s)$ in policy P_1 if, and only if, a packet from the flow with age $\Delta_{[i],\pi_1}(t)$ at time t is successfully delivered at time $(t + T_s)$ in policy π_1. □

Notice that policy P_1 and policy π_1 are two distinct policies, so the flow with age $\Delta_{[i],P_1}(t)$ in policy P_1 and the flow with age $\Delta_{[i],\pi_1}(t)$ at time t in policy π_1 are likely representing different flows. However, policy P_1 and policy π_1 are coupled in such a way that the packet deliveries for these two flows occur simultaneously at time $(t + T_s)$.

We will compare policy P_1 and policy π_1 on a sample path by using the following lemma:

Lemma 6. *(Inductive comparison)* Under the conditions of Lemma 5, if (i) The packet generation and arrival times are synchronized across the N flows and (ii)

$$\Delta_{[i],P_1}(t) \le \Delta_{[i],\pi_1}(t),\ i = 1,\cdots,N, \tag{54}$$

then

$$\Delta_{[i],P_1}(t + T_s) \le \Delta_{[i],\pi_1}(t + T_s),\ i = 1,\cdots,N. \tag{55}$$

Proof. For synchronized packet generations and arrivals, let $W(t) = \max_i\{S_i : A_i \le t\}$ be the generation time of the freshest packet of each flow that has arrived at the queue by time t. Because (i) The packet transmission time is T_s and (ii) No packet that has arrived at the queue by time t was generated after time $W(t)$, we can obtain

$$\Delta_{[i],\pi_1}(t + T_s) \ge t + T_s - W(t),\ i = 1,\cdots,N. \tag{56}$$

Without loss of generality, suppose that there are l transmission errors and $(N - l)$ successful packet deliveries at time $(t+T_s)$ in policy P_1. Because (i) Policy P_1 follows the same scheduling discipline with the DT-MAF-LGFS policy and (ii) The packet generation and arrival times are synchronized across the N flows, each delivered packet must be the freshest packet generated at time $W(t)$. Hence, the flows associated with these delivered packets must have the minimum age at time $(t+T_s)$, given by

$$\Delta_{[i],P_1}(t + T_s) = t + T_s - W(t),\ i = l + 1,\cdots,N. \tag{57}$$

Combining (56) and (57), yields

$$\Delta_{[i],P_1}(t + T_s) = t + T_s - W(t) \le \Delta_{[i],\pi_1}(t + T_s),\quad i = l + 1,\cdots,N. \tag{58}$$

Moreover, suppose that the transmission errors at time $(t + T_s)$ are from the flows with age values $(\Delta_{[j_1],P_1}(t),\Delta_{[j_2],P_1}(t),\cdots,\Delta_{[j_l],P_1}(t))$ at time t, which are sorted such that $j_1 \ge j_2 \ge \cdots \ge j_l$. Because $\Delta_{[i],P_1}(t)$ is the ith largest component of the age vector $\mathbf{\Delta}_{P_1}(t)$, we have

$$\Delta_{[j_1],P_1}(t) \ge \Delta_{[j_2],P_1}(t) \ge \cdots \ge \Delta_{[j_l],P_1}(t). \tag{59}$$

If flow n is one of the flows that encounter a transmission error at time $t + T_s$ in policy P_1, then

$$\Delta_{n,P_1}(t + T_s) = \Delta_{n,P_1}(t) + T_s. \tag{60}$$

From (57), (59), and (60), the components of vector $\mathbf{\Delta}_{P_1}(t + T_s)$ are $\Delta_{[j_1],P_1}(t) + T_s,\Delta_{[j_2],P_1}(t) + T_s,\cdots,\Delta_{[j_l],P_1}(t) + T_s$ and $(N - l)$ numbers with the same value $t + T_s - W(t)$. Hence,

$$\Delta_{[i],P_1}(t + T_s) = \Delta_{[j_i],P_1}(t) + T_s,\ i = 1,\cdots,l. \tag{61}$$

According to Lemma 5, there are l transmission errors at time $(t + T_s)$ in policy π_1, which are from the flows with age values $(\Delta_{[j_1],\pi_1}(t),\Delta_{[j_2],\pi_1}(t),\cdots,\Delta_{[j_l],\pi_1}(t))$ at time t. Because $j_1 \ge j_2 \ge \cdots \ge j_l$, we have

$$\Delta_{[j_1],\pi_1}(t) \ge \Delta_{[j_2],\pi_1}(t) \ge \cdots \ge \Delta_{[j_l],\pi_1}(t). \tag{62}$$

If flow n is one of the flows that encounter a transmission error at time $t + T_s$ in policy π_1, then

$$\Delta_{n,\pi_1}(t + T_s) = \Delta_{n,\pi_1}(t) + T_s. \tag{63}$$

From (62) and (63), one can observe that $\Delta_{[j_1],\pi_1}(t) + T_s,\Delta_{[j_2],\pi_1}(t) + T_s,\cdots,\Delta_{[j_l],\pi_1}(t) + T_s$ are l components of vector $\mathbf{\Delta}_{\pi_1}(t + T_s)$. Hence,

$$\Delta_{[i],\pi_1}(t + T_s) \ge \Delta_{[j_i],\pi_1}(t) + T_s,\ i = 1,\cdots,l. \tag{64}$$

Combining (54), (61), and (64), yields

$$\begin{aligned}&\Delta_{[i],P_1}(t + T_s)\\ =&\Delta_{[j_i],P_1}(t) + T_s\\ \le&\Delta_{[j_i],\pi_1}(t) + T_s\\ \le&\Delta_{[i],\pi_1}(t + T_s),\ i = 1,\cdots,l.\end{aligned} \tag{65}$$

Finally, (55) follows from (58) and (65). This completes the proof. □

Now we prove Theorem 3.

Proof of Theorem 3. Consider any non-preemptive, workconserving policy $\pi \in \Pi_{np}$. By Lemma 5, there exist policy P_1 and policy π_1 satisfying the same scheduling disciplines with policy P and policy π, respectively, such that if a packet from the flow with age $\Delta_{[i],P_1}(t)$ at time t is successfully delivered at time $(t+T_s)$ in policy P_1, then almost surely, a packet from the flow with age $\Delta_{[i],\pi_1}(t)$ at time t is successfully delivered at time $(t + T_s)$ in policy π_1; and vice versa.

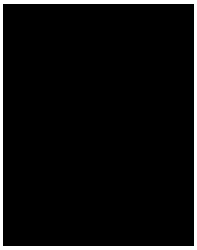
For any given sample path of policy P_1 and policy π_1, the initial system state is $\mathbf{\Delta}_{P_1}(0) = \mathbf{\Delta}_{\pi_1}(0)$ at time $t = 0$. The evolution of the system state is governed by Lemma 6. By induction over time, we obtain

$$\Delta_{[i],P_1}(t) \le \Delta_{[i],\pi_1}(t),\ i = 1,\cdots,N,\ t = 0,T_s,2T_s,\cdots. \tag{66}$$

The rest of the proof is quite similar to that of Theorem 1 and hence are omitted. □

Yin Sun received his B.Eng. and Ph.D. degrees in Electronic Engineering from Tsinghua University, in 2006 and 2011, respectively. He was a Postdoctoral Scholar and Research Associate at the Ohio State University from 2011-2017 and an Assistant Professor in the Department of Electrical and Computer Engineering at Auburn University from 2017-2023. He is currently the Godbold Associate Professor in the Department of Electrical and Computer Engineering at Auburn University, Alabama. His research interests include wireless networks, machine learning, semantic communications, age of information, information theory, and robotic control. He is also interested in applying AI and machine learning techniques in agricultural, food, and nutrition sciences.

He has been an Associate Editor of the *IEEE Transactions on Network Science and Engineering*, an Editor of the *Journal of Communications and Networks*, an Editor of the *IEEE Transactions on Green Communications and Networking*, a Guest Editor of five special journal issues, and an Organizing Committee Member of several conferences. He founded the Age of Information (AoI) Workshop in 2018 and the Modeling and Optimization in Semantic Communications (MOSC) Workshop in 2023. His articles received the best student paper award of the IEEE/IFIP WiOpt 2013, best paper award of the IEEE/IFIP WiOpt 2019, runner-up for the best paper award of ACM MobiHoc 2020, and 2021 Journal of Communications and Networks (JCN) best paper award. He co-authored a monograph *Age of Information: A New Metric for Information Freshness*. He received the Auburn Author Award of 2020, the National Science Foundation (NSF) CAREER Award in 2023, and was named a Ginn Faculty Achievement Fellow in 2023. He is a Senior Member of the IEEE and a Member of the ACM.



Sastry Kompella earned his Ph.D. in Electrical and Computer Engineering from the Virginia Polytechnic Institute and State University in 2006. Currently, he serves as the Chief Scientist for Nexcepta, Inc., an advanced R&D company providing cutting-edge technical solutions and mission-critical capabilities to the Department of Defense (DoD). Prior to this role, he held the position of Section Head for the wireless network research section within the Information Technology Division at the U.S. Naval Research Laboratory in Washington, DC, USA. His research interests encompass various aspects of wireless networks, including cognitive radio, dynamic spectrum access, and age of information.