Average Age of Information in a Multi-Source M/M/1 Queueing Model with LCFS Prioritized Packet Management

Mohammad Moltafet and Markus Leinonen
Centre for Wireless Communications – Radio Technologies
University of Oulu, Finland
e-mail: {mohammad.moltafet, markus.leinonen}@oulu.fi

Marian Codreanu
Department of Science and Technology
Linköping University, Sweden
e-mail: marian.codreanu@liu.se

***Abstract*—In this paper, we consider an M/M/1 status update system consisting of two independent sources, one server, and one sink. We consider the following last-come first-served (LCFS) prioritized packet management policy. When the system is empty, any arriving packet immediately enters the server; when the server is busy, a packet of a source waiting in the queue is replaced if a new packet of the same source arrives and the fresh packet goes at the head of the queue. We derive the average age of information (AoI) of the considered M/M/1 queueing model by using the stochastic hybrid systems (SHS) technique. Numerical results illustrate the effectiveness of the proposed packet management policy compared to several baseline policies.**

I. Introduction

In many applications of Internet of things and cyber-physical control systems, freshness of the status information at receivers is a critical factor. Recently, the age of information (AoI) was proposed as a destination-centric metric to measure the information freshness in status update systems [1]–[3]. A status update packet contains the measured value of a monitored process and a time stamp representing the time when the sample was generated. Due to wireless channel access, channel errors, and fading, etc., communicating a status update packet through the network experiences a random delay. If at a time instant t, the most recently received status update packet contains the time stamp $U(t)$, AoI is defined as the random process $\Delta(t) = t - U(t)$. Thus, the AoI measures for each sensor the time elapsed since the last received status update packet was generated at the sensor. The average AoI is the most commonly used metric to evaluate the AoI [1]–[14].

The first queueing theoretic work on AoI is [2] where the authors derived the average AoI for a single-source M/M/1 first-come first-served (FCFS) queueing model. The average AoI for an M/M/1 last-come first-served (LCFS) queueing model with preemption was analyzed in [3]. The average AoI for different packet management policies in a single-source M/M/1 queueing model were derived in [8]. The work [11] was the first to investigate the average AoI in a multi-source setup. The authors of [11] derived the average AoI for a multi-source M/M/1 FCFS queueing model. The closed-form expressions for the average AoI and average peak AoI in a multi-source M/G/1/1 preemptive queueing model were derived in [12]. The authors of [13] derived an exact expression for the average AoI for a multi-source M/M/1 FCFS queueing model and an approximate expression for the average AoI for a multi-source M/G/1 FCFS queueing model having a general service time distribution.

The most related works to our paper are [1] and [14]. In [1], the authors introduced a powerful technique based on stochastic hybrid systems (SHS) to evaluate the AoI in continuous-time queueing systems. They considered a multi-source queueing model in which the packets of different sources are generated according to the Poisson process and are served according to an exponentially distributed service time. The authors derived the average AoI for two packet management policies: 1) LCFS with preemption under service (LCFS-S), and 2) LCFS with preemption only in waiting (LCFS-W). Under the LCFS-S policy, a new arriving packet preempts any packet that is currently under service (regardless of the source index). Under the LCFS-W policy, a new arriving packet replaces any older packet waiting in the queue (regardless of the source index); however, the new packet has to wait for any update packet that is currently under service to finish. In [14], the authors studied a different packet management policy without considering priority of serving based on the arrivals.

In this paper, we consider a status update system in which two independent sources generate packets according to the Poisson process and the packets are served according to an exponentially distributed service time. We consider an LCFS prioritized packet management policy as follows. When the system is empty, any arriving packet immediately enters the server. Differently from the policies studied in [1] and [14], when the server is busy at an arrival of a packet, the possible packet of the *same source* waiting in the queue is replaced by the arriving packet and this fresh packet goes *at the head* of the queue. We derive the average AoI for each source in the considered queueing model using the SHS technique.

II. System Model

We consider a status update system consisting of two independent sources, one server, and one sink, as depicted

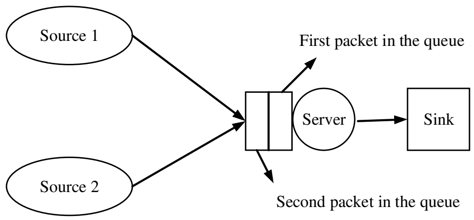

Fig. 1: The considered status update system.

in Fig. 1. Each source observes a random process at random time instants. The sink is interested in timely information about the status of these random processes. Status updates are transmitted as packets, containing the measured value of the monitored process and a time stamp representing the time when the sample was generated. We assume that the packets of sources 1 and 2 are generated according to the Poisson process with rates λ_1 and λ_2, respectively, and the packets are served according to an exponentially distributed service time with mean $1/\mu$. Let $\rho_1 = \lambda_1/\mu$ and $\rho_2 = \lambda_2/\mu$ be the load of source 1 and 2, respectively. Since packets of the sources are generated according to the Poisson process and the sources are independent, the packet generation in the system follows the Poisson process with rate $\lambda = \lambda_1 + \lambda_2$. The overall load in the system is $\rho = \rho_1 + \rho_2 = \lambda/\mu$.

We consider the following LCFS prioritized packet management policy. The queue can contain at most two packets at the same time, one packet of source 1 and one packet of source 2. When the system is empty, any arriving packet immediately enters the server. When the server is busy, a packet of a source $c \in \{1,2\}$ waiting in the queue is replaced if a new packet of the *same source* arrives, and the fresh packet goes *at the head* of the queue.

For each source, the AoI at the sink is defined as the time elapsed since the last successfully received packet was generated. Next, we present the formal definition of the AoI.

Definition 1 (AoI). Let $t_{c,i}$ denote the time instant at which the ith status update packet of source c was generated, and $t'_{c,i}$ denote the time instant at which this packet arrives at the sink. At a time instant τ, the index of the most recently received packet of source c is given by $N_c(\tau) = \max\{i' | t'_{c,i'} \leq \tau\}$, and the time stamp of the most recently received packet of source c is $U_c(\tau) = t_{c,N_c(\tau)}$. The AoI of source c at the destination is defined as the random process $\Delta_c(t) = t - U_c(t)$. Let $(0,\tau)$ denote an observation interval. Accordingly, the time average AoI of the source c at the sink, denoted as $\Delta_{\tau,c}$, is defined as $\Delta_{\tau,c} = \frac{1}{\tau}\int_0^\tau \Delta_c(t)\mathrm{d}t$. The average AoI of source c, denoted by Δ_c, is defined as

$$\Delta_c = \lim_{\tau\to\infty} \Delta_{\tau,c}. \tag{1}$$

III. AoI Analysis Using the SHS Technique

Next, we use the SHS technique introduced in [1] to calculate the average AoI in (1) of each source in the system. In the following, we briefly present the main idea behind the SHS technique. We refer the readers to [1] for more details.

A. SHS Technique

The SHS technique models a queueing system through the states $(q(t), \mathbf{x}(t))$, where $q(t) \in \mathcal{Q} = \{0, 1, \ldots, m\}$ is a continuous-time finite-state Markov chain that describes the occupancy of the system and $\mathbf{x}(t) = [x_0(t)\ x_1(t) \cdots x_n(t)] \in \mathbb{R}^{1\times(n+1)}$ is a continuous process that describes the evolution of age-related processes at the sink. Following the approach in [1], we label the source of interest as source 1 and employ the continuous process $\mathbf{x}(t)$ to track the age of source 1 status updates at the sink.

The Markov chain $q(t)$ can be presented as a graph $(\mathcal{Q}, \mathcal{L})$ where each discrete state $q(t) \in \mathcal{Q}$ is a node of the chain and a (directed) link $l \in \mathcal{L}$ from node q_l to node q'_l indicates a transition from state $q_l \in \mathcal{Q}$ to state $q'_l \in \mathcal{Q}$. Note that unlike in a typical continuous-time Markov chain, a transition from a state to itself (i.e., a self-transition) is possible in the chain $q(t) \in \mathcal{Q}$. Through a self-transition, a reset of the continuous state $\mathbf{x}$ takes place, but the discrete state remains the same (for more details, see [1, Section III]).

A transition occurs when a packet arrives or departs in the system. Since the time elapsed between departures and arrivals is exponentially distributed, the transition $l \in \mathcal{L}$ from state q_l to state q'_l occurs with the exponential rate $\lambda^{(l)}\delta_{q_l,q(t)}$, where the Kronecker delta function $\delta_{q_l,q(t)}$ ensures that the transition l occurs only when the discrete state $q(t)$ is equal to q_l. When a transition l occurs, the discrete state q_l jumps to state q'_l, and the continuous state $\mathbf{x}$ is reset to $\mathbf{x}'$ according to a binary transition reset map matrix $\mathbf{A}_l \in \mathbb{R}^{(n+1)\times(n+1)}$ as $\mathbf{x}' = \mathbf{x}\mathbf{A}_l$. In addition, at each state $q(t) = q \in \mathcal{Q}$, the continuous state $\mathbf{x}$ evolves as a piece-wise linear function through the differential equation $\dot{\mathbf{x}}(t) = \frac{\partial \mathbf{x}(t)}{\partial t} = \mathbf{b}_q$, where $\mathbf{b}_q = [b_{q,0}\ b_{q,1} \cdots b_{q,n}]$ is a vector with binary elements, i.e., $b_{q,j} \in \{0,1\}, \forall j \in \{0,\ldots,n\}, q \in \mathcal{Q}$. If the age process $x_j(t)$ increases at a unit rate, we have $b_{q,j} = 1$; otherwise, $b_{q,j} = 0$.

To calculate the average AoI by using the SHS technique, the state probabilities of the Markov chain and the correlation vector between the discrete state $q(t)$ and the continuous state $\mathbf{x}(t)$ need to be calculated. Let $\pi_q(t)$ denote the probability of being in state q of the Markov chain and $\mathbf{v}_q(t)$ denote the correlation vector between the discrete state $q(t)$ and the continuous state $\mathbf{x}(t)$. Accordingly, we have

$$\pi_q(t) = \mathbb{E}[\delta_{q,q(t)}] = \Pr(q(t) = q), \tag{2}$$

$$\mathbf{v}_q(t) = \mathbb{E}[\mathbf{x}(t)\delta_{q,q(t)}] = [v_{q0}(t) \cdots v_{qn}(t)]. \tag{3}$$

Let $\mathcal{L}'_q$ denote the set of incoming transitions and $\mathcal{L}_q$ denote the set of outgoing transitions for state q, defined as

$$\mathcal{L}'_q = \{l \in \mathcal{L} : q'_l = q\}, \quad \mathcal{L}_q = \{l \in \mathcal{L} : q_l = q\}.$$

Following the ergodicity assumption of the Markov chain $q(t)$ in the AoI analysis [1], [15], the state probability vector $\boldsymbol{\pi}(t) = [\pi_0(t) \cdots \pi_m(t)]$ converges uniquely to the stationary vector $\bar{\boldsymbol{\pi}} = [\bar{\pi}_0 \cdots \bar{\pi}_m]$ satisfying [1]

$$\bar{\pi}_q \textstyle\sum_{l\in\mathcal{L}_q} \lambda^{(l)} = \sum_{l\in\mathcal{L}'_q} \lambda^{(l)} \bar{\pi}_{q_l}, \ \ \forall q \in \mathcal{Q}, \tag{4}$$

TABLE I: SHS Markov chain states

State	Source index of the second packet in the queue	Source index of the first packet in the queue	Source index of the packet under service
0	-	-	-
1	-	-	1
2	-	-	2
3	-	1	1
4	-	2	1
5	2	1	1
6	1	2	1
7	-	1	2
8	-	2	2
9	2	1	2
10	1	2	2

$$\sum_{q\in\mathcal{Q}} \bar{\pi}_q = 1. \tag{5}$$

As it has been shown in [1, Theorem 4], under the ergodicity assumption of the Markov chain $q(t)$, the correlation vector $\mathbf{v}_q(t)$ converges to a nonnegative limit $\bar{\mathbf{v}}_q = [\bar{v}_{q0} \cdots \bar{v}_{qn}], \forall q \in \mathcal{Q}$, as $t \to \infty$ such that

$$\bar{\mathbf{v}}_q \sum_{l\in\mathcal{L}_q} \lambda^{(l)} = \mathbf{b}_q \bar{\pi}_q + \sum_{l\in\mathcal{L}'_q} \lambda^{(l)} \bar{\mathbf{v}}_{q_l} \mathbf{A}_l, \;\; \forall q \in \mathcal{Q}. \tag{6}$$

Finally, the average AoI of source 1 is calculated by [1, Theorem 4]

$$\Delta_1 = \sum_{q\in\mathcal{Q}} \bar{v}_{q0}. \tag{7}$$

As it can be observed in (7), calculating the average AoI using the SHS technique boils down to deriving $\bar{v}_{q0}, \ \forall q \in \mathcal{Q}$.

B. Average AoI Calculation

In the considered system model, the state space of the Markov chain is $\mathcal{Q} = \{0, 1, \ldots, 10\}$; the different states are presented in Table I. For example, $q = 0$ indicates that the server is idle, i.e., the system is empty; $q = 1$ indicates that a source 1 packet is under service and the queue is empty; $q = 5$ indicates that a source 1 packet is under service, the first packet in the queue (i.e., the packet that is at the head of the queue as depicted in Fig. 1) is a source 1 packet, and the second packet in the queue is a source 2 packet.

In our queueing model, the continuous process is $\mathbf{x}(t) = [x_0(t)\ x_1(t)\ x_2(t)\ x_3(t)]$, where $x_0(t)$ is the current AoI of source 1 at time instant t, $\Delta_1(t)$; $x_1(t)$ encodes what $\Delta_1(t)$ would become if the packet that is under service is delivered to the sink at time instant t; $x_2(t)$ encodes what $\Delta_1(t)$ would become if the first packet in the queue is delivered to the sink at time instant t; $x_3(t)$ encodes what $\Delta_1(t)$ would become if the second packet in the queue is delivered to the sink at time instant t. The transitions between the discrete states $q_l \to q'_l, \ \forall l \in \mathcal{L}$, and their effects on the continuous state $\mathbf{x}(t)$ are summarized in Table II. In the following, we explain the transitions presented in Table II:

- $l = 1$: A source 1 packet arrives at an empty system. With this arrival/transition, the AoI of source 1 does not change, i.e., $x'_0 = x_0$ (recall that the continuous state $\mathbf{x}$ is reset to $\mathbf{x}'$ when a transition occurs). This is because the arrival of source 1 packet does not yield an age reduction until it is delivered to the sink. However, since the arriving source 1 packet is fresh and its age is zero, we have $x'_1 = 0$. In addition, since with this arrival the queue is still empty, x_2 and x_3 become irrelevant to the AoI of source 1, and thus, $x'_2 = 0$ and $x'_3 = 0$. Finally, we have

$$\mathbf{x}' = [x_0\ x_1\ x_2\ x_3]\mathbf{A}_1 = [x_0\ 0\ 0\ 0]. \tag{8}$$

According to (8), it can be shown that $\mathbf{A}_1$ is given by

$$\mathbf{A}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}. \tag{9}$$

Then, by using (9), $\mathbf{v}_0\mathbf{A}_1$ is calculated as

$$\mathbf{v}_0\mathbf{A}_1 = [v_{00}\ v_{01}\ v_{02}\ v_{03}]\mathbf{A}_1 = [v_{00}\ 0\ 0\ 0]. \tag{10}$$

It can be seen from (8)-(10) that when we have $\mathbf{x}'$ for a transition $l \in \mathcal{L}$, it is easy to calculate $\mathbf{v}_{q_l}\mathbf{A}_l$. Thus, for the rest of the transitions, we just explain the calculation of $\mathbf{x}'$ and present the final expression of $\mathbf{v}_{q_l}\mathbf{A}_l$.

- $l = 2$: A source 2 packet arrives at an empty system. We have $x'_0 = x_0$, because this arrival does not change the AoI of source 1 at the sink. Since the arriving packet is a source 2 packet, x_1 is irrelevant and we have $x'_1 = 0$. Moreover, since the queue is empty, x_2 and x_3 become irrelevant, and we have $x'_2 = 0$ and $x'_3 = 0$.

TABLE II: Transition rates for the Markov chain

l	$q_l \to q'_l$	$\lambda^{(l)}$	$\mathbf{x}\mathbf{A}_l$	$\mathbf{v}_{ql}\mathbf{A}_l$
1	$0 \to 1$	λ_1	$[x_0\ 0\ 0\ 0]$	$[v_{00}\ 0\ 0\ 0]$
2	$0 \to 2$	λ_2	$[x_0\ 0\ 0\ 0]$	$[v_{00}\ 0\ 0\ 0]$
3	$1 \to 3$	λ_1	$[x_0\ x_1\ 0\ 0]$	$[v_{10}\ v_{11}\ 0\ 0]$
4	$1 \to 4$	λ_2	$[x_0\ x_1\ 0\ 0]$	$[v_{10}\ v_{11}\ 0\ 0]$
5	$2 \to 7$	λ_1	$[x_0\ 0\ 0\ 0]$	$[v_{20}\ 0\ 0\ 0]$
6	$2 \to 8$	λ_2	$[x_0\ 0\ 0\ 0]$	$[v_{20}\ 0\ 0\ 0]$
7	$1 \to 0$	μ	$[x_1\ 0\ 0\ 0]$	$[v_{11}\ 0\ 0\ 0]$
8	$2 \to 0$	μ	$[x_0\ 0\ 0\ 0]$	$[v_{20}\ 0\ 0\ 0]$
9	$3 \to 3$	λ_1	$[x_0\ x_1\ 0\ 0]$	$[v_{30}\ v_{31}\ 0\ 0]$
10	$4 \to 4$	λ_2	$[x_0\ x_1\ 0\ 0]$	$[v_{40}\ v_{41}\ 0\ 0]$
11	$3 \to 6$	λ_2	$[x_0\ x_1\ 0\ x_2]$	$[v_{30}\ v_{31}\ 0\ v_{32}]$
12	$4 \to 5$	λ_1	$[x_0\ x_1\ 0\ 0]$	$[v_{40}\ v_{41}\ 0\ 0]$
13	$5 \to 5$	λ_1	$[x_0\ x_1\ 0\ 0]$	$[v_{50}\ v_{51}\ 0\ 0]$
14	$5 \to 6$	λ_2	$[x_0\ x_1\ 0\ x_2]$	$[v_{50}\ v_{51}\ 0\ v_{52}]$
15	$6 \to 6$	λ_2	$[x_0\ x_1\ 0\ x_3]$	$[v_{60}\ v_{61}\ 0\ v_{63}]$
16	$6 \to 5$	λ_1	$[x_0\ x_1\ 0\ 0]$	$[v_{60}\ v_{61}\ 0\ 0]$
17	$7 \to 7$	λ_1	$[x_0\ 0\ 0\ 0]$	$[v_{70}\ 0\ 0\ 0]$
18	$7 \to 10$	λ_2	$[x_0\ 0\ 0\ x_2]$	$[v_{70}\ 0\ 0\ v_{72}]$
19	$8 \to 8$	λ_2	$[x_0\ 0\ 0\ 0]$	$[v_{80}\ 0\ 0\ 0]$
20	$8 \to 9$	λ_1	$[x_0\ 0\ 0\ 0]$	$[v_{80}\ 0\ 0\ 0]$
21	$9 \to 9$	λ_1	$[x_0\ 0\ 0\ 0]$	$[v_{90}\ 0\ 0\ 0]$
22	$9 \to 10$	λ_2	$[x_0\ 0\ 0\ x_2]$	$[v_{90}\ 0\ 0\ v_{92}]$
23	$10 \to 10$	λ_2	$[x_0\ 0\ 0\ x_3]$	$[v_{100}\ 0\ 0\ v_{103}]$
24	$10 \to 9$	λ_1	$[x_0\ 0\ 0\ 0]$	$[v_{100}\ 0\ 0\ 0]$
25	$3 \to 1$	μ	$[x_1\ x_2\ 0\ 0]$	$[v_{31}\ v_{32}\ 0\ 0]$
26	$4 \to 2$	μ	$[x_1\ 0\ 0\ 0]$	$[v_{41}\ 0\ 0\ 0]$
27	$5 \to 4$	μ	$[x_1\ x_2\ 0\ 0]$	$[v_{51}\ v_{52}\ 0\ 0]$
28	$6 \to 7$	μ	$[x_1\ 0\ x_3\ 0]$	$[v_{61}\ 0\ v_{63}\ 0]$
29	$7 \to 1$	μ	$[x_0\ x_2\ 0\ 0]$	$[v_{70}\ v_{72}\ 0\ 0]$
30	$8 \to 2$	μ	$[x_0\ 0\ 0\ 0]$	$[v_{80}\ 0\ 0\ 0]$
31	$9 \to 4$	μ	$[x_0\ x_2\ 0\ 0]$	$[v_{90}\ v_{92}\ 0\ 0]$
32	$10 \to 7$	μ	$[x_0\ 0\ x_3\ 0]$	$[v_{100}\ 0\ v_{103}\ 0]$

- $l=3$: A source 1 packet is under service and a source 1 packet arrives. In this transition, we have $x_0' = x_0$ because there is no departure. The delivery of the packet under service reduces the AoI to x_1 and thus, we have $x_1' = x_1$. Since the arriving source 1 packet is fresh and its age is zero, we have $x_2' = 0$. Since there is only one packet in the queue, x_3 becomes irrelevant, and we have $x_3' = 0$. The reset map of transition $l = 4$ can be derived similarly.
- $l=5$: A source 2 packet is under service and a source 1 packet arrives. In this transition, we have $x_0' = x_0$ because there is no departure. Since the packet under service is a source 2 packet, x_1 is irrelevant, and thus, we have $x_1' = 0$. Since the arriving source 1 packet is fresh and its age is zero, we have $x_2' = 0$. Since there is only one packet in the queue, x_3 becomes irrelevant, and we have $x_3' = 0$. The reset map of transition $l = 6$ can be derived similarly.
- $l=7$: A source 1 packet completes service and is delivered to the sink. With this transition, the AoI is reset to the age of the source 1 packet that just completed service, and thus, $x_0' = x_1$. Since the system enters state $q = 0$, x_1, x_2, and x_3 become irrelevant, and thus, we have $x_1' = 0$, $x_2' = 0$, and $x_3' = 0$. The reset map of transition $l = 8$ can be derived similarly.
- $l=9$: A source 1 packet is under service, a source 1 packet is in the queue, and a source 1 packet arrives. The source 1 packet in the queue is replaced by the fresh source 1 packet. In this transition, we have $x_0' = x_0$ because there is no departure. The delivery of the packet under service reduces the AoI to x_1, and thus, we have $x_1' = x_1$. Since the arriving source 1 packet is fresh and its age is zero, we have $x_2 = 0$. Since there is only one packet in the queue, x_3 becomes irrelevant, and we have $x_3' = 0$. The reset maps of transitions $l = 10$, $l = 17$, and $l = 19$ can be derived similarly.
- $l=11$: A source 1 packet is under service, a source 1 packet is in the queue, and a source 2 packet arrives. In this transition, we have $x_0' = x_0$ because there is no departure. The delivery of the packet under service reduces the AoI to x_1, and thus, we have $x_1' = x_1$. According to the packet management policy, the arriving packet of source 2 goes at the head of the queue while the source 1 packet waiting in the queue moves as the second packet in the queue. Consequently, the first packet in the queue is a source 2 packet, and we have $x_2' = 0$. The delivery of the second packet in the queue, which due to the packet management is now a source 1 packet, reduces the AoI to x_2, and thus, we have $x_3' = x_2$. The reset maps of transitions $l = 12$, $l = 18$, and $l = 20$ can be derived similarly.
- $l=13$: A source 1 packet is under service, the first packet in the queue is a source 1 packet, the second packet in the queue is a source 2 packet, and a source 1 packet arrives. According to the packet management policy, the source 1 packet that is at the head of the queue is replaced by the fresh source 1 packet. In this transition, we have $x_0' = x_0$ because there is no departure. The delivery of the packet under service reduces the AoI to x_1, and thus, we have $x_1' = x_1$. Since the arriving source 1 packet is fresh and its age is zero, we have $x_2' = 0$. Since the second packet in the queue is a source 2 packet, x_3 is irrelevant, and we have $x_3' = 0$. The reset maps of transitions $l = 15$, $l = 21$, and $l = 23$ can be derived similarly.
- $l=14$: A source 1 packet is under service, the first packet in the queue is a source 1 packet, the second packet in the queue is a source 2 packet, and a source 2 packet arrives. According to the packet management policy, the source 2 packet in the queue is replaced by the arriving source 2 packet and this fresh packet goes at the head of the queue. In this transition, we have $x_0' = x_0$ because there is no departure. The delivery of the packet under service reduces the AoI to x_1, and thus, we have $x_1' = x_1$. Due to the packet management policy, the arriving packet of source 2 goes at the head of the queue while the source 1 packet which was at head of the queue moves as the second packet in the queue. Thus, we have $x_2' = 0$ and $x_3' = x_2$. The reset maps of transitions $l = 16$, $l = 22$, and $l = 24$ can be derived similarly.
- $l=25$: A source 1 packet is in the queue, and a source 1 packet completes service and is delivered to the sink. With this transition, the AoI is reset to the age of the source 1 packet that just completed service, and thus, $x_0' = x_1$. Since the source 1 packet in the queue goes to the server, we have, $x_1' = x_2$. Since the source 1 packet in the queue goes to the server, the queue becomes empty, and thus, we have $x_2' = 0$ and $x_3' = 0$. The reset maps of transitions $l = 26$, $l = 29$, and $l = 30$ can be derived similarly.
- $l=27$: The first packet in the queue is a source 1 packet, the second packet in the queue is a source 2 packet, and the source 1 packet completes service and is delivered to the sink. With this transition, the AoI is reset to the age of the source 1 packet that just completed service, i.e., $x_0' = x_1$. Since the first packet in the queue goes to the server, we have $x_1' = x_2$. Finally, since the queue holds only one source 2 packet, we have $x_2' = 0$ and $x_3' = 0$. The reset maps of transitions $l = 28$, $l = 31$, and $l = 32$ can be derived similarly.

Recall that our goal is to find $\bar{v}_{q0}, \forall q \in \mathcal{Q}$, to calculate the average AoI of source 1 in (7). In this regard, first we determine $\mathbf{b}_q, \forall q \in \mathcal{Q}$, and the stationary probability vector $\bar{\boldsymbol{\pi}}$. Then, by solving the linear equations in (6), we calculate $\bar{v}_{q0}, \forall q \in \mathcal{Q}$.

The evolution of $\mathbf{x}(t)$ at each discrete state $q(t) = q$ is determined by $\mathbf{b}_q$, i.e., $\dot{\mathbf{x}} = \mathbf{b}_q$. Thus, the first element of $\mathbf{b}_q$ is equal to 1 in all discrete states, $b_{q,1} = 1,\ \forall q \in \mathcal{Q}$. This is because the AoI of source 1, $\Delta_1(t) = x_0(t)$, increases at a unit rate with time in all discrete states. The second element of $\mathbf{b}_q$ is equal to 1 if there is a relevant packet (i.e., a packet of source 1) under service at state $q(t) = q$. The third element of $\mathbf{b}_q$ is equal to 1 if the first packet in the queue is a relevant packet at state $q(t)=q$. The fourth element of $\mathbf{b}_q$ is equal to 1 if the second packet in the queue is a relevant packet at state $q(t)=q$. Thus, $\mathbf{b}_q$ for different states are determined by

$$\mathbf{b}_q=\begin{cases}[1\ 0\ 0\ 0]\,,\ q=0,\\ [1\ 1\ 0\ 0]\,,\ q=1,\\ [1\ 0\ 0\ 0]\,,\ q=2,\\ [1\ 1\ 1\ 0]\,,\ q=3,\\ [1\ 1\ 0\ 0]\,,\ q=4,\\ [1\ 1\ 1\ 0]\,,\ q=5,\end{cases}\quad \mathbf{b}_q=\begin{cases}[1\ 1\ 0\ 1]\,,\ q=6,\\ [1\ 0\ 1\ 0]\,,\ q=7,\\ [1\ 0\ 0\ 0]\,,\ q=8,\\ [1\ 0\ 1\ 0]\,,\ q=9,\\ [1\ 0\ 0\ 1]\,,\ q=10.\end{cases} \tag{11}$$

To calculate the stationary probabilities, we use (4) and (5). Using (4) and the transition rates of the different states presented in Table II, it can be shown that the stationary probability vector $\bar{\boldsymbol{\pi}}$ satisfies $\bar{\boldsymbol{\pi}}\mathbf{D}=\bar{\boldsymbol{\pi}}\mathbf{Q}$ with

$$\begin{aligned}\mathbf{D}=&\mathrm{diag}[\lambda,\lambda+\mu,\lambda+\mu,\lambda+\mu,\lambda+\mu,\lambda+\mu,\lambda+\mu,\lambda+\mu,\\ &\lambda+\mu,\lambda+\mu,\lambda+\mu],\end{aligned}$$

$$\mathbf{Q}=\begin{bmatrix} 0 & \lambda_1 & \lambda_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\ \mu & 0 & 0 & \lambda_1 & \lambda_2 & 0 & 0 & 0 & 0 & 0 & 0\\ \mu & 0 & 0 & 0 & 0 & 0 & 0 & \lambda_1 & \lambda_2 & 0 & 0\\ 0 & \mu & 0 & \lambda_1 & 0 & 0 & \lambda_2 & 0 & 0 & 0 & 0\\ 0 & 0 & \mu & 0 & \lambda_2 & \lambda_1 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & \mu & \lambda_1 & \lambda_2 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & \lambda_1 & \lambda_2 & \mu & 0 & 0 & 0\\ 0 & \mu & 0 & 0 & 0 & 0 & 0 & \lambda_1 & 0 & 0 & \lambda_2\\ 0 & 0 & \mu & 0 & 0 & 0 & 0 & 0 & \lambda_2 & \lambda_1 & 0\\ 0 & 0 & 0 & 0 & \mu & 0 & 0 & 0 & 0 & \lambda_1 & \lambda_2\\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mu & 0 & \lambda_1 & \lambda_2 \end{bmatrix}.$$

Applying (5), we derive the stationary vector $\bar{\boldsymbol{\pi}}$. Consequently, the stationary probabilities $\bar{\pi}_0$ and $\bar{\pi}_1$ are calculated as

$$\begin{aligned}\bar{\pi}_0&=\frac{\rho+2\rho_1\rho_2+1}{\epsilon}, \qquad (12)\\ \bar{\pi}_1&=\frac{\rho_1\left(\rho_1\rho_2(\rho+4)+(\rho_1+1)^2+(\rho+1)\rho_2^2+2\rho_2\right)}{(\rho+1)\epsilon},\end{aligned}$$

where

$$\begin{aligned}\epsilon=&\rho_1^4\rho_2+3\rho_1^2\rho_2^2\rho+\rho_1^3+2\rho+\rho_1\rho_2(4\rho^2+7\rho+2)\\ &+\rho_1\rho_2^4+\rho_2^3+2\rho^2+1.\end{aligned}$$

For known $\bar{\pi}_0$ and $\bar{\pi}_1$ in (12), the probabilities $\{\bar{\pi}_2,\bar{\pi}_3,\bar{\pi}_4,\bar{\pi}_5\}$ are given as

$$\begin{aligned}&\bar{\pi}_2=\rho\bar{\pi}_0-\bar{\pi}_1,\ \ \bar{\pi}_3=\frac{\rho_1\bar{\pi}_1}{1+\rho_2}, \qquad (13)\\ &\bar{\pi}_4=\frac{\rho_1(\rho+2)+1}{1+\rho_1}(\rho\bar{\pi}_0-\bar{\pi}_1),\ \bar{\pi}_5=\frac{\rho\rho_1(\rho_1(\rho+2)+1)}{1+\rho}\bar{\pi}_0\\ &+\left(\frac{\rho_1^2\rho_2}{(1+\rho)(1+\rho_2)}-\frac{\rho_1(\rho_1(\rho+2)+1)}{1+\rho}\right)\bar{\pi}_1.\end{aligned}$$

For known $\{\bar{\pi}_0,\bar{\pi}_1,\bar{\pi}_2,\bar{\pi}_3,\bar{\pi}_4,\bar{\pi}_5\}$ in (12) and (13), the probabilities $\{\bar{\pi}_6,\bar{\pi}_7,\bar{\pi}_8,\bar{\pi}_9,\bar{\pi}_{10}\}$ are given as

$$\begin{aligned}&\bar{\pi}_6=\frac{\rho_2(\bar{\pi}_3+\bar{\pi}_5)}{1+\rho_1},\ \bar{\pi}_7=(1+\rho)\bar{\pi}_1-\rho_1\bar{\pi}_0-\bar{\pi}_3, \qquad (14)\\ &\bar{\pi}_8=\frac{\rho_2\bar{\pi}_2}{1+\rho_1},\ \bar{\pi}_9=(1+\rho_1)\bar{\pi}_4-\rho_2\bar{\pi}_1-\bar{\pi}_5,\\ &\bar{\pi}_{10}=\frac{\rho_2}{1+\rho_1}\bigg(\left(1+\rho_1\right)\left(\bar{\pi}_4+\bar{\pi}_1\right)-\rho_1\bar{\pi}_0-\bar{\pi}_3-\bar{\pi}_5\bigg).\end{aligned}$$

By substituting the stationary probability vector $\bar{\boldsymbol{\pi}}$ defined through (12)-(14) and $\mathbf{b}_q$ in (11) into (6) and solving the corresponding system of linear equations, the values of $\bar{v}_{q0}$, $\forall q\in\mathcal{Q}$, are calculated. Finally, by substituting these obtained values into (7), the average AoI of source 1 in the considered queueing model is given as

$$\Delta_1=\frac{\sum_{i=0}^{13}\rho_1^i\psi_i}{\mu\rho_1\left(1+\rho_1\right)\left(\sum_{j=0}^{11}\rho_1^j\xi_j\right)}, \tag{15}$$

$$\begin{aligned}
\psi_0&=\rho_2^7+5\rho_2^6+12\rho_2^5+18\rho_2^4+18\rho_2^3+12\rho_2^2+5\rho_2+1,\\
\psi_1&=3\rho_2^8+22\rho_2^7+76\rho_2^6+159\rho_2^5+222\rho_2^4+213\rho_2^3+138\rho_2^2\\
&\quad+56\rho_2+11,\\
\psi_2&=3\rho_2^9+31\rho_2^8+155\rho_2^7+465\rho_2^6+917\rho_2^5+1240\rho_2^4\\
&\quad+1162\rho_2^3+737\rho_2^2+292\rho_2+56,\\
\psi_3&=\rho_2^{10}+14\rho_2^9+108\rho_2^8+506\rho_2^7+1512\rho_2^6+3015\rho_2^5\\
&\quad+4123\rho_2^4+3878\rho_2^3+2440\rho_2^2+974\rho_2+176,\\
\psi_4&=\rho_2^{10}+21\rho_2^9+179\rho_2^8+900\rho_2^7+2910\rho_2^6+6228\rho_2^5\\
&\quad+8994\rho_2^4+8764\rho_2^3+5590\rho_2^2+2446\rho_2+385,\\
\psi_5&=10\rho_2^9+142\rho_2^8+910\rho_2^7+3458\rho_2^6+8401\rho_2^5+13379\rho_2^4\\
&\quad+13983\rho_2^3+9294\rho_2^2+3594\rho_2+625,\\
\psi_6&=44\rho_2^8+493\rho_2^7+2507\rho_2^6+7428\rho_2^5+13704\rho_2^4\\
&\quad+15958\rho_2^3+11403\rho_2^2+4553\rho_2+777,\\
\psi_7&=112\rho_2^7+1025\rho_2^6+4167\rho_2^5+9536\rho_2^4+12961\rho_2^3\\
&\quad+10313\rho_2^2+4372\rho_2+743,\\
\psi_8&=182\rho_2^6+1356\rho_2^5+4319\rho_2^4+7324\rho_2^3+6780\rho_2^2\\
&\quad+3144\rho_2+536,\\
\psi_9&=196\rho_2^5+1156\rho_2^4+2744\rho_2^3+3151\rho_2^2+1656\rho_2+280,\\
\psi_{10}&=140\rho_2^4+617\rho_2^3985\rho_2^2+619\rho_2+99,\\
\psi_{11}&=64\rho_2^3+188\rho_2^2+157\rho_2+21,\\
\psi_{12}&=17\rho_2^2+25\rho_2+2,\quad \psi_{13}=2\rho_2,\\
\xi_0&=\psi_0,\ \xi_1=2\rho_2^8+13\rho_2^7+46\rho_2^6+102\rho_2^5+151\rho_2^4+153\rho_2^3\\
&\quad+104\rho_2^2+44\rho_2+9,\\
\xi_2&=\rho_2^9+8\rho_2^8+45\rho_2^7+158\rho_2^6+363\rho_2^5+562\rho_2^4+592\rho_2^3\\
&\quad+415\rho_2^2+179\rho_2+37,\\
\xi_3&=9\rho_2^8+67\rho_2^7+275\rho_2^6+711\rho_2^5+1207\rho_2^4+1364\rho_2^3\\
&\quad+1003\rho_2^2+444\rho_2+92,\\
\xi_4&=\rho_2^8+42\rho_2^7+258\rho_2^6+830\rho_2^5+1637\rho_2^4+2056\rho_2^3\\
&\quad+1623\rho_2^2+745\rho_2+154,\\
\xi_5&=7\rho_2^7+121\rho_2^6+577\rho_2^5+1433\rho_2^4+2099\rho_2^3+1833\rho_2^2\\
&\quad+886\rho_2+182,\\
\xi_6&=21\rho_2^6+220\rho_2^5+790\rho_2^4+1449\rho_2^3+146\rho_2^2+760\rho_2+154,\\
\xi_7&=35\rho_2^5+251\rho_2^4+652\rho_2^3+809\rho_2^2+469\rho_2+92,\\
\xi_8&=35\rho_2^4+174\rho_2^3+299\rho_2^2+204\rho_2+37,\\
\xi_9&=21\rho_2^3+67\rho_2^2+60\rho_2+9,\ \xi_{10}=7\rho_2^2+11\rho_2+1,\ \xi_{11}=\rho_2.
\end{aligned}$$

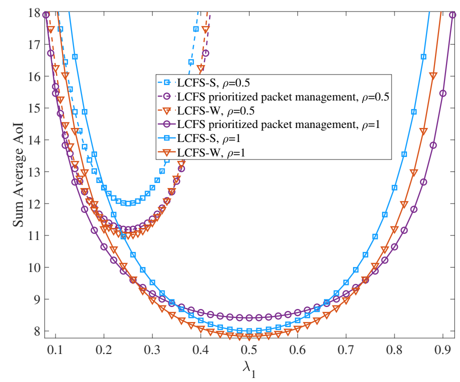

Fig. 2: Sum average AoI for different values of ρ under different management policies with $\mu = 1$.

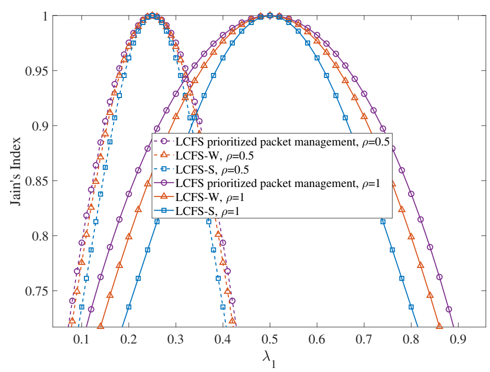

Fig. 3: Sum average AoI for different values of ρ under different management policies with $\mu = 1$.

IV. Numerical Results

In this section, we show the effectiveness of the proposed packet management policy in terms of the sum average AoI and fairness between different sources in the system. The fairness is assessed by the Jain's fairness index which is defined as $J(\Delta_1, \Delta_2) = \dfrac{(\Delta_1 + \Delta_2)^2}{2(\Delta_1^2 + \Delta_2^2)}$ [16, Section. 3]. $J(\Delta_1, \Delta_2)$ is continuous and lies in $[0.5, 1]$, where $J(\Delta_1, \Delta_2) = 1$ indicates the fairest situation in the system.

Fig. 2 illustrates the average AoI of sources 1 and 2 for different values of ρ under different packet management policies. From this figure, we observe that when the system can not choose λ_1 and λ_2, the best policy to achieve a low value of the sum average AoI depends on the system parameters. Fig. 3 depicts the Jain's fairness index for the average AoI of sources 1 and 2 as a function of λ_1 under different packet management policies. From this figure, we observe that the proposed policy outperforms the existing policies from the fairness perspective.

V. Conclusions

We considered an M/M/1 status update system consisting of two independent sources, one server, and one sink. We proposed the LCFS prioritized packet management policy in which when a new packet of a source arrives, the possible packet of the same source waiting in the queue is replaced by the arriving packet and this fresh packet goes at the head of the queue. We derived the average AoI for each source using the SHS technique. The numerical results illustrated the effectiveness of the proposed policy compared to the LCFS-S and LCFS-W policies in terms of fairness and sum average AoI.

Acknowledgements

This research has been financially supported by the Infotech Oulu, the Academy of Finland (grant 323698), and Academy of Finland 6Genesis Flagship (grant 318927). M. Codreanu would like to acknowledge the support of the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie Grant Agreement No. 793402 (COMPRESS NETS). The work of M. Leinonen has also been financially supported in part by the Academy of Finland (grant 319485). M. Moltafet would like to acknowledge the support of Finnish Foundation for Technology Promotion and HPY Research Foundation.

References

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

[2] S. Kaul, R. Yates, and M. Gruteser, "Real-time status: How often should one update?" in *Proc. IEEE Int. Conf. on Computer. Commun. (INFOCOM)*, Orlando, FL, USA, Mar. 25–30, 2012, pp. 2731–2735.

[3] S. K. Kaul, R. D. Yates, and M. Gruteser, "Status updates through queues," in *Proc. Conf. Inform. Sciences Syst. (CISS)*, Princeton, NJ, USA, Mar. 21–23, 2012, pp. 1–6.

[4] A. Kosta, N. Pappas, and V. Angelakis, "Age of information: A new concept, metric, and tool," *Foun. and Trends in Net.*, vol. 12, no. 3, pp. 162–259, 2017.

[5] M. Moltafet, M. Leinonen, and M. Codreanu, "Worst case age of information in wireless sensor networks: A multi-access channel," *IEEE Wireless Commun. Lett.*, vol. 9, no. 3, pp. 321–325, Mar. 2020.

[6] M. Moltafet, M. Leinonen, M. Codreanu, and N. Pappas, "Power minimization in wireless sensor networks with constrained AoI using stochastic optimization," in *Proc. Annual Asilomar Conf. Signals, Syst., Comp.*, Pacific Grove, USA, Nov. 3–6, 2019. [Online]. https://arxiv.org/pdf/1912.02421v1.

[7] S. Kaul, M. Gruteser, V. Rai, and J. Kenney, "Minimizing age of information in vehicular networks," in *Proc. Commun. Society. Conf. on Sensor, Mesh and Ad Hoc Commun. and Net.*, Salt Lake City, UT, USA, Jun. 27–30, 2011, pp. 350–358.

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

[9] Y. Inoue, H. Masuyama, T. Takine, and T. Tanaka, "The stationary distribution of the age of information in FCFS single-server queues," in *Proc. IEEE Int. Symp. Inform. Theory*, Aachen, Germany, Jun. 25–30, 2017, pp. 571–575.

[10] E. Najm and R. Nasser, "Age of information: The gamma awakening," in *Proc. IEEE Int. Symp. Inform. Theory*, Barcelona, Spain, Jul. 10–16, 2016, pp. 2574–2578.

[11] R. D. Yates and S. Kaul, "Real-time status updating: Multiple sources," in *Proc. IEEE Int. Symp. Inform. Theory*, Cambridge, MA, USA, Jul. 1–6, 2012, pp. 2666–2670.

[12] E. Najm and E. Telatar, "Status updates in a multi-stream M/G/1/1 preemptive queue," in *Proc. IEEE Int. Conf. on Computer. Commun. (INFOCOM)*, Honolulu, HI, USA, Apr. 15–19, 2018, pp. 124–129.

[13] M. Moltafet, M. Leinonen, and M. Codreanu, "On the age of information in multi-source queueing models," *Submitted to IEEE Trans. Commun*, [Online]. https://arxiv.org/abs/1911.07029v1, 2019.

[14] ——, "Average age of information for a multi-source M/M/1 queueing model with packet management," *Submitted to IEEE Int. Symp. Inform. Theory*, [Online]. https://arxiv.org/abs/2001.03959v1, 2020.

[15] A. Maatouk, M. Assaad, and A. Ephremides, "On the age of information in a CSMA environment," *IEEE/ACM Trans. Net.*, Early Access 2020.

[16] R. Jain, D. Chiu, and W. Hawe, "A quantitative measure of fairness and discrimination for resource allocation in shared systems," *Digital Equipment Corporation, DEC-TR-301, Tech. Rep., 1984*, available: http://www1.cse.wustl.edu/ jain/papers/ftp/fairness.pdf.