2020 18th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt)

Average Age of Information for a Multi-Source M/M/1 Queueing Model with Packet Management and Self-Preemption in Service

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*—We consider an M/M/1 status update system consisting of two independent sources and one server. We derive the average age of information (AoI) of each source using the stochastic hybrid systems (SHS) technique under the following packet management with self-preemptive serving policy. The system can contain at most two packets with different source indexes at the same time, i.e., one packet under service and one packet in the queue. When the system is empty, any arriving packet immediately enters the server. When the server is busy at an arrival of a packet, the possible packet of the same source in the system (either waiting in the queue or being served) is replaced by the fresh packet. Numerical results illustrate the effectiveness of the proposed packet management with self-preemptive serving policy compared to several baseline policies.**

***Index Terms*– Information freshness, age of information (AoI), multi-source queueing model, stochastic hybrid systems (SHS).**

I. Introduction

In many services provided by the upcoming wireless network generation, a destination needs status updates of various sensors which are assigned to monitor different random processes. One key enabler for these services is freshness of the sensors' information at the destination. 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.

The most related works to our paper are [1], [14] and [15]. 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. As alternatives to these *source-agnostic* packet management policies in [1], the works [14] and [15] proposed *source-aware* packet management policies where the arriving packets of a source cannot replace the packets of another source in the system.

In this paper, we consider an M/M/1 status update system in which two independent sources generate packets according to the Poisson process. We derive the average AoI of each source under the following packet management with self-preemptive serving policy. The system can contain at most two packets with different source indexes. When the system is empty, any arriving packet immediately enters the server. Our proposed policy is similar to [14] and [15] in that the queueing policy is source-aware, thus being different from the source-agnostic policy in [1]. However, as the main difference to [14] and [15],

ISBN 978-3-903176-29-4 © 2020 IFIP

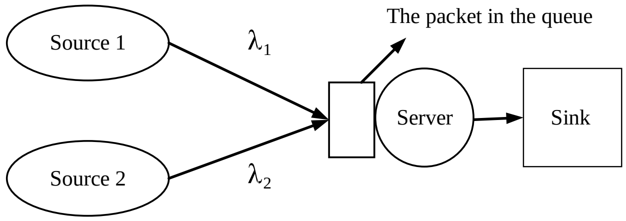

Fig. 1: The considered status update system.

when the server is busy at an arrival of a packet, the possible packet of the same source waiting in the queue *or under service* (i.e., a *self-preemptive* serving policy) is replaced by the arriving packet.

II. System Model

We consider a status update system consisting of two independent sources, one server, and one sink, as depicted 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 $\rho = \rho_1 + \rho_2 = \lambda/\mu$.

Packet management with a self-preemptive serving policy: The system (i.e., the queue and server) can contain at most two packets with different source indexes at the same time, i.e., 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 $c \in \{1, 2\}$ in the *system* is replaced if a new packet of the *same source* arrives. In other words, when a new packet of a source $c \in \{1, 2\}$ arrives, the possible packet of the same source waiting in the queue or under service (called *self-preemption*) is replaced by the fresh packet.

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

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

III. AoI Analysis Using the SHS Technique

Next, we use the SHS technique introduced in [1], to calculate the average AoI 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 $q(t) \in \mathcal{Q} = \{0, 1, \cdots, m\}$ 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 a transition from a state to itself (i.e., a self-transition) is possible. Through a self-transition, a reset of the continuous state $\mathbf{x}$ takes place, but the discrete state remains the same (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., $b_{q,j} \in \{0, 1\}, \forall j \in \{0, \cdots, n\}, q \in \mathcal{Q}$. 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, 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 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

$$\pi_q(t) = \mathbb{E}[\delta_{q,q(t)}] = \Pr(q(t) = q), \quad (1)$$

$$\mathbf{v}_q(t) = \mathbb{E}[\mathbf{x}(t)\delta_{q,q(t)}] = [v_{q0}(t) \cdots v_{qn}(t)]. \tag{2}$$

Let $\mathcal{L}'_q$ denote the set of incoming transitions and $\mathcal{L}_q$ 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], [16], 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]

$$\bar{\pi}_q \textstyle\sum_{l \in \mathcal{L}_q} \lambda^{(l)} = \sum_{l \in \mathcal{L}'_q} \lambda^{(l)} \bar{\pi}_{q_l}, \ \ \forall q \in \mathcal{Q}, \tag{3}$$

$$\textstyle\sum_{q \in \mathcal{Q}} \bar{\pi}_q = 1. \tag{4}$$

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

$$\bar{\mathbf{v}}_q \textstyle\sum_{l \in \mathcal{L}_q} \lambda^{(l)} = \mathbf{b}_q \bar{\pi}_q + \sum_{l \in \mathcal{L}'_q} \lambda^{(l)} \bar{\mathbf{v}}_{q_l} \mathbf{A}_l, \ \ \forall q \in \mathcal{Q}. \tag{5}$$

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

$$\Delta_1 = \textstyle\sum_{q \in \mathcal{Q}} \bar{v}_{q0}. \tag{6}$$

Thus, the main goal in deriving the average AoI is to calculate $\bar{v}_{q0}, \ \forall q \in \mathcal{Q}$.

B. Average AoI Calculation

In our model, the state space of the Markov chain is $\mathcal{Q} = \{0, 1, 2, 3, 4\}$, where $q = 0$ 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; $q = 2$ indicates that a source 2 packet is under service and the queue is empty; $q = 3$ indicates that a source 1 packet is under service, and a source 2 packet is in the queue; and $q = 4$ indicates that a source 2 packet is under service, and a source 1 packet is in the queue. The continuous process is $\mathbf{x}(t) = [x_0(t) \ x_1(t) \ x_2(t)]$, 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 packet in the queue is delivered to the sink at time instant t.

The Markov chain for the discrete state $q(t)$ is shown in Fig. 2. 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 I. In the following, we explain the transitions presented in Table I:

- *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., $x'_0 = x_0$. This is because the arrival of source 1 packet does not yield an age reduction until it is delivered to the sink. Since the arriving source 1 packet is fresh and its age is zero, we have $x'_1 = 0$. Since with this arrival the queue is still empty, x_2 becomes irrelevant to the AoI of source 1, and thus, $x'_2 = 0$. Finally, we have

$$\mathbf{x}' = [x_0 \ x_1 \ x_2]\mathbf{A}_1 = [x_0 \ 0 \ 0]. \tag{7}$$

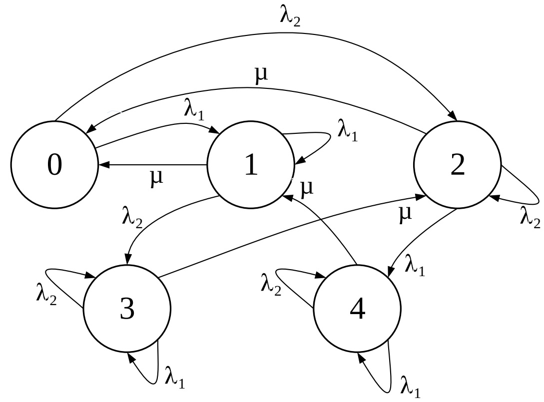

Fig. 2: The SHS Markov chain.

TABLE I: Transition rates for the Markov chain

l	$q_l \to q'_l$	$\lambda^{(l)}$	$\mathbf{x}\mathbf{A}_l$	$\mathbf{A}_l$	$\mathbf{v}_{q_l}\mathbf{A}_l$
1	$0 \to 1$	λ_1	$[x_0 \ 0 \ 0]$	$\begin{bmatrix}1&0&0\\0&0&0\\0&0&0\end{bmatrix}$	$[v_{00} \ 0 \ 0]$
2	$0 \to 2$	λ_2	$[x_0 \ 0 \ 0]$	$\begin{bmatrix}1&0&0\\0&0&0\\0&0&0\end{bmatrix}$	$[v_{00} \ 0 \ 0]$
3	$1 \to 1$	λ_1	$[x_0 \ 0 \ 0]$	$\begin{bmatrix}1&0&0\\0&0&0\\0&0&0\end{bmatrix}$	$[v_{10} \ 0 \ 0]$
4	$2 \to 2$	λ_2	$[x_0 \ 0 \ 0]$	$\begin{bmatrix}1&0&0\\0&0&0\\0&0&0\end{bmatrix}$	$[v_{20} \ 0 \ 0]$
5	$1 \to 3$	λ_2	$[x_0 \ x_1 \ 0]$	$\begin{bmatrix}1&0&0\\0&1&0\\0&0&0\end{bmatrix}$	$[v_{10} \ v_{11} \ 0]$
6	$2 \to 4$	λ_1	$[x_0 \ 0 \ 0]$	$\begin{bmatrix}1&0&0\\0&0&0\\0&0&0\end{bmatrix}$	$[v_{20} \ 0 \ 0]$
7	$3 \to 3$	λ_1	$[x_0 \ 0 \ 0]$	$\begin{bmatrix}1&0&0\\0&0&0\\0&0&0\end{bmatrix}$	$[v_{30} \ 0 \ 0]$
8	$3 \to 3$	λ_2	$[x_0 \ x_1 \ 0]$	$\begin{bmatrix}1&0&0\\0&1&0\\0&0&0\end{bmatrix}$	$[v_{30} \ v_{31} \ 0]$
9	$4 \to 4$	λ_1	$[x_0 \ 0 \ 0]$	$\begin{bmatrix}1&0&0\\0&0&0\\0&0&0\end{bmatrix}$	$[v_{40} \ 0 \ 0]$
10	$4 \to 4$	λ_2	$[x_0 \ 0 \ x_2]$	$\begin{bmatrix}1&0&0\\0&0&0\\0&0&1\end{bmatrix}$	$[v_{40} \ 0 \ v_{42}]$
11	$1 \to 0$	μ	$[x_1 \ 0 \ 0]$	$\begin{bmatrix}0&0&0\\1&0&0\\0&0&0\end{bmatrix}$	$[v_{11} \ 0 \ 0]$
12	$2 \to 0$	μ	$[x_0 \ 0 \ 0]$	$\begin{bmatrix}1&0&0\\0&0&0\\0&0&0\end{bmatrix}$	$[v_{20} \ 0 \ 0]$
13	$3 \to 2$	μ	$[x_1 \ 0 \ 0]$	$\begin{bmatrix}0&0&0\\1&0&0\\0&0&0\end{bmatrix}$	$[v_{31} \ 0 \ 0]$
14	$4 \to 1$	μ	$[x_0 \ x_2 \ 0]$	$\begin{bmatrix}1&0&0\\0&0&0\\0&1&0\end{bmatrix}$	$[v_{40} \ v_{42} \ 0]$

According to (7), it can be shown that the binary matrix $\mathbf{A}_1$ is given by

$$\mathbf{A}_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{8}$$

Then, by using (8), $\mathbf{v}_0\mathbf{A}_1$ is calculated as

$$\mathbf{v}_0\mathbf{A}_1 = [v_{00} \ v_{01} \ v_{02}]\mathbf{A}_1 = [v_{00} \ 0 \ 0]. \tag{9}$$

It can be seen from (7)-(9) 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 expressions of $\mathbf{A}_l$ and $\mathbf{v}_{q_l}\mathbf{A}_l$.

- *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 at the sink. Since the arriving packet is a source 2 packet, x_1 is irrelevant, and thus, we have $x'_1 = 0$. Moreover, since the queue is empty, x_2 becomes irrelevant, and thus, we have $x'_2 = 0$.
- *l=3*: A source 1 packet is under service and a source 1 packet arrives. According to the self-preemptive serving policy, the source 1 packet that is under service is preempted by the arriving source 1 packet. In this transition, we have $x'_0 = x_0$ because there is no departure. Since the arrived source 1 packet which entered the server through preemption is fresh and its age is zero, we have $x'_1 = 0$. Since the queue is empty, x_2 becomes irrelevant, and thus, we have $x'_2 = 0$. The reset map of transition $l = 4$ can be derived using the similar arguments.
- *l=5*: A source 1 packet is under service 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 $x'_1 = x_1$. Since the packet in the queue is a source 2 packet, x_2 is irrelevant, and thus, we have $x'_2 = 0$. The reset map of transition $l = 6$ can be derived using the similar arguments.
- *l=7*: A source 1 packet is under service, the packet in the queue is a source 2 packet, and a source 1 packet arrives. According to the self-preemptive policy, the source 1 packet that is under service is preempted by the arriving source 1 packet. In this transition, we have $x'_0 = x_0$ because there is no departure. Since the arrived source 1 packet which entered the server through preemption is fresh and its age is zero, we have $x'_1 = 0$. Since the packet in the queue is a source 2 packet, x_2 is irrelevant, and thus, we have $x'_2 = 0$. The reset maps of transitions $l = 8$, $l = 9$, and $l = 10$ can be derived using the similar arguments.
- *l=11*: A source 1 packet completes service and is delivered to the sink. With this transition, the AoI at the sink 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 $q = 0$, we have $x'_1 = 0$, and $x'_2 = 0$. The reset map of transition $l = 12$ can be derived using the similar arguments.
- *l=13*: The packet in the queue is a source 2 packet and the source 1 packet in the server completes service and is delivered to the sink. With this transition, the AoI at the sink is reset to the age of the source 1 packet that just completed service, i.e., $x'_0 = x_1$. Since the packet that goes to the server is a source 2 packet, x_1 is irrelevant, and thus, we have $x'_1 = 0$. In addition, since with this transition the queue becomes empty, we have $x'_2 = 0$. The reset map of transition $l = 14$ can be derived using the similar arguments.

Recall that our goal is to find $\bar{v}_{q0}, \forall q \in \mathcal{Q}$, to calculate the average AoI of source 1 in (6). 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 (5), 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 packet in the queue is a relevant packet at state $q(t) = q$. Thus, $\mathbf{b}_q$ for different states are given as

$$\mathbf{b}_q = \begin{cases} [1 \ \ 0 \ \ 0], & q = 0, \\ [1 \ \ 1 \ \ 0], & q = 1, \\ [1 \ \ 0 \ \ 0], & q = 2, \\ [1 \ \ 1 \ \ 0], & q = 3, \\ [1 \ \ 0 \ \ 1], & q = 4. \end{cases} \tag{10}$$

To calculate the stationary probabilities, we use (3) and (4). Using (3) and the transition rates among different states presented in Table I, it can be shown that the stationary probability vector $\bar{\boldsymbol{\pi}}$ satisfies $\bar{\boldsymbol{\pi}}\mathbf{D} = \bar{\boldsymbol{\pi}}\mathbf{Q}$ with

$$\mathbf{D} = \ \mathrm{diag}[\lambda, \lambda+\mu, \lambda+\mu, \lambda+\mu, \lambda+\mu],$$
$$\mathbf{Q} = \begin{bmatrix} 0 & \lambda_1 & \lambda_2 & 0 & 0 \\ \mu & \lambda_1 & 0 & \lambda_2 & 0 \\ \mu & 0 & \lambda_2 & 0 & \lambda_1 \\ 0 & 0 & \mu & \lambda & 0 \\ 0 & \mu & 0 & 0 & \lambda \end{bmatrix}.$$

Applying (4), the stationary probabilities are given as

$$\bar{\boldsymbol{\pi}} = \frac{1}{2\rho_1\rho_2 + \rho + 1}\left[1 \quad \rho_1 \quad \rho_2 \quad \rho_1\rho_2 \quad \rho_1\rho_2\right]. \tag{11}$$

By substituting (10) and (11) into (5) and solving the corresponding linear equations, the values of $\bar{v}_{q0}, \ \forall q \in \mathcal{Q}$, are calculated. Finally, by substituting the results into (6), the average AoI of source 1 in the considered queueing model is given as

$$\Delta_1 = \frac{\sum_{i=1}^{5} \rho_1^i \psi_i + (\rho_2+1)^2}{\mu\rho_1 \left(1+\rho_1\right)^2 \left(\rho_1^2(2\rho_2+1) + (\rho_2+1)^2(2\rho_1+1)\right)}, \tag{12}$$

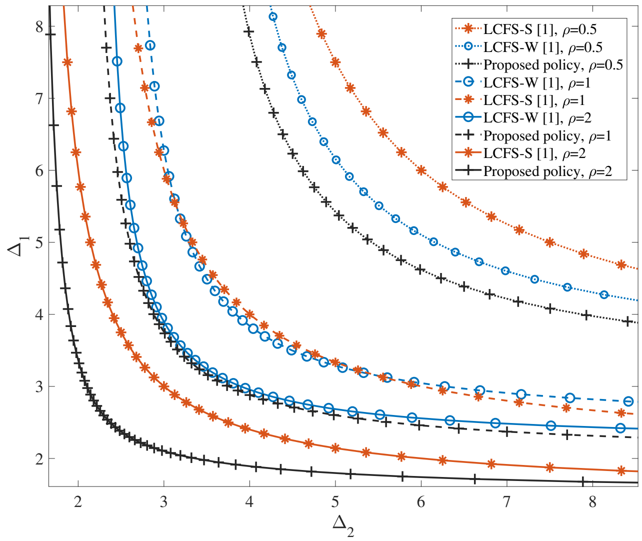

Fig. 3: The average AoI of sources 1 and 2 for different values of ρ under different queueing policies with $\mu = 1$.

where

$$\psi_1 = 6\rho_2^2 + 11\rho_2 + 5,$$
$$\psi_2 = 13\rho_2^2 + 24\rho_2 + 10,$$
$$\psi_3 = 10\rho_2^2 + 27\rho_2 + 10,$$
$$\psi_4 = 3\rho_2^2 + 14\rho_2 + 5,$$
$$\psi_5 = 3\rho_2 + 1.$$

IV. Numerical Results

In this section, we show the effectiveness of the proposed packet management policy compared to the policies proposed in [1]. Fig. 3 depicts the contours of achievable average AoI pairs (Δ_1, Δ_2) for fixed values of system load $\rho = \rho_1 + \rho_2$ under different packet management policies with normalized service rate $\mu = 1$. As it can be seen, the proposed policy outperforms both the LCFS-S and LCFS-W policies derived in [1] in terms of average AoI. In addition, we can observe that the proposed policy provides the minimum sum average AoI in the system. Moreover, we can observe that by increasing the value of ρ the minimum sum average AoI decreases.

V. Conclusions

We considered an M/M/1 status update system consisting of two independent sources and one server. We proposed the packet management with self-preemptive serving policy in which when the server is busy at an arrival of a packet, the possible packet of the same source in the system (waiting in the queue or under service) is replaced by the arriving packet. We derived the average AoI of each source using the SHS technique. The numerical results showed that for the M/M/1 queueing systems, the proposed packet management with self-preemptive serving policy results in a lower sum average AoI compared to the LCFS-S and LCFS-W policies derived in [1].

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, pp. 406–410.

[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," *Accepted, IEEE Int. Symp. Inform. Theory*, [Online]. https://arxiv.org/abs/2001.03959v1, 2020.

[15] ——, "Average age of information in a multi-source M/M/1 queueing model with LCFS prioritized packet management," *Accepted, IEEE Int. Conf. on Comp. Commun.*, 2020.

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