Age of Information in an $M/M/2$ Queue with In-Order Delivery

Yijie Huang, Zhiyuan Jiang
School of Communication and Information Engineering
Shanghai University
Shanghai, China
{nickyhuang, jiangzhiyuan}@shu.edu.cn

***Abstract*—The emergence of the concept of Age of Information provides a new method to quantify the freshness of information for time-critical network systems. In parallel server networks with out-of-order arrival of updates, this paper studies and analyzes the performance of the time average age with the in-order delivery mode, i.e., the updates are delivered to the destination node according to their generation timestamp. The time average age under the $M/M/2$ blocking and queuing models is evaluated by a stochastic-hybrid-system approach and a graphical decomposition method, respectively. Numerical results demonstrate that the theoretical expression is consistent with the simulated age, and compared with out-of-order delivery, the performance loss of in-order delivery is within 14.2%.**

***Index Terms*—Age of Information, status update system, in-order delivery, queuing theory**

I. Introduction

The importance of low-latency cyber-physical system applications continues to grow as a key driver of the fifth generation cellular communication systems. The timeliness of status updates has become an important performance measure of network optimization, which leads to a series of studies on the Age of Information (AoI) metric. AoI captures the freshness of information from the destination node, which can be simply defined as the time elapsed since the generation of the freshest packet delivered successfully. By taking the impact of both latency and throughput into account, AoI can track the entire process of network status update.

The research focusing on the analysis and optimization of AoI usually model the system as a stochastic process where the source submits updates to the destination. In queuing networks, packets can travel through multiple routes, which motivates AoI analysis of parallel server queues. Existing work in [1], [2] addressed the network with plentiful and limited resources. The authors derived the expression of the average age for the $M/M/\infty$ model and developed an approximation of the average age for the $M/M/2$ model. Upper and lower bounds for the above models were also derived. In [3], [4], the $M/M/c^*$ preemptive parallel server system was analyzed using Stochastic Hybrid System (SHS) and both homogeneous and heterogeneous servers cases are discussed. Scheduling for parallel severs was investigated in [5], [6], the optimality of Last-Generated-First-Served (LGFS) policy was emphatically established.

The results in these works that hold for out-of-order packet arrivals all involve a common problem. Any update currently in service would be preempted or discarded upon the source generating a new update or a fresher update finishing service before the ones with larger age. This causes the fact that these obsolete updates are terminated or invalid. However, some specific scenarios of wireless communication systems have the requirements for the integrity and strict ordering of data, making the results inapplicable.

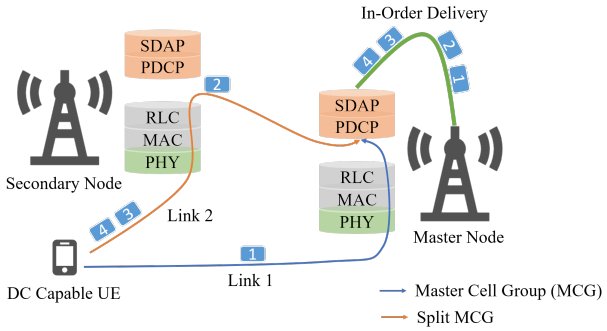

Fig. 1. Dual Connectivity (DC).

In particular, the in-order delivery requirement arises in the dual connection mode in 5G cellular systems. With limited numbers of 5G users initially, the 5G cell would not be sufficiently spread and the operators can only provide seamless service to the 5G user by inter-working with existing 4G Long -Term Evolution (LTE) networks [7]. Dual connectivity, otherwise known as E-UTRAN New Radio - Dual Connectivity (EN-DC), is the technology that enables a 4G and 5G connection to occur at the same time as shown in Fig. 1, which is adopted by Option 3 in non-standalone (NSA) to provide a combined radio access to user equipment (UE). The data from or to the UE can be transmitted by the split bearer via LTE-access as the master node and NR-access as the secondary node. The split of data for split bearers takes place at the Packet Data Convergence Protocol (PDCP) layer, where the security function requires the integrity protection of the control plane data, and the data transmission function requires the in-order transmission and reception of Protocol Data Units (PDUs).

The resequencing problem in the networks with multiple parallel links has been widely discussed in the last century. The mean resequencing delay for a $M/M/m$ system of different service rates is obtained in [8]. The mean resequencing delay

for a queueing system with two servers under a threshold-type policy is analyzed in [9]. [10] investigates the relationship between the mean resequencing delay and variations in packet service times for $M/H_k/\infty$ systems. As far as we know, no research has analyzed age in terms of in-order delivery of data while maintaining integrity. This paper attempts to study the timeliness of information with the in-order delivery mode.

In this work, we consider a system where a source node generates timestamped updates to a destination node. These timestamps can be regarded as the sequence number (SN) field of the PDU in PDCP layer, which can be used to check whether the data is delivered in sequence by sending and receiving entities of the PDCP layer. This paper analyzes the time-average AoI with in-order delivery mode in two typical cases. One is the $M/M/2$ blocking system, in which updates will not be dropped in the middle of the service. This setting can reduce the complexity of the system, and will not cause the system to be overloaded due to an excessively high arrival rate. The average age of the blocking system is obtained by the SHS approach which is good at solving problems with finite Markov states. The other is the $M/M/2$ queuing system that complies with a first-come-first-served (FCFS) discipline and maximizes data integrity. An approximation for the average age of the queuing system is derived from a graphical method.

The remainder of this paper is organized as follows. In Section II, the system model is described. In Section III, we give a brief introduction to SHS and use it to perform an age analysis of the $M/M/2$ blocking system. In IV, we consider the $M/M/2$ queuing system and provide an approximately expression close to the simulated age. Simulation results and conclusions are presented in Section V and VI.

II. System Model

The system is modeled with a source transmitting packets through a network to a remote monitor. As shown in Fig. 2, the model uses two communication links to act as random delays in the network, the first half of which is equivalent to a $M/M/2$ queuing model. Since the stochastic service time may cause the packets to arrive at the destination out of order, we add an intermediate node before the destination node to ensure orderly transmission. The intermediate node plays the role of reordering, and the packets it transmits to the monitor are complete and ordered.

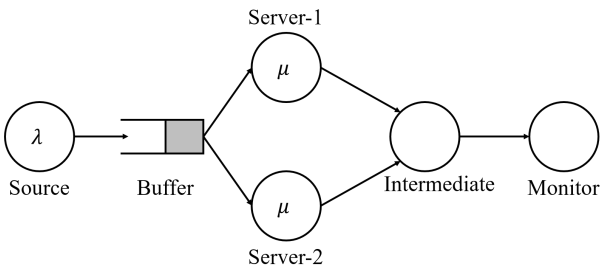

Fig. 2. The $M/M/2$ network model.

Based on this model, the AoI is analyzed from the perspective of the orderliness and integrity of the data. Take the FCFS $M/M/s$ queue as an example. Parallel server network allows multiple packets to be served together. Packets are numbered in the order in which they were generated. Under the setting of in-order delivery, so as not to discard the update package in service, if the packet $i+1$ completes the service before the packet i, it cannot be delivered to the destination immediately, but must wait in the intermediate node and be received or used by the monitor at the moment when the packet i completes its service.

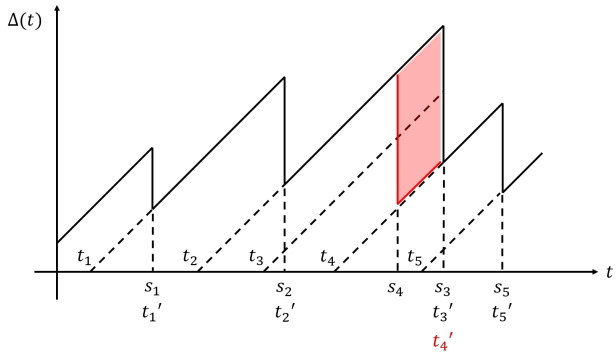

Fig. 3. The AoI evolution for system model.

The sample path of the status age is shown in Fig. 3, where the generation time and the service completion time of update packets are denoted by $t_1, t_2, ..., t_i$ and $s_1, s_2, ..., s_i$ respectively while true receptions at the monitor occur at times $t_1', t_2', ..., t_i'$. In the absence of an update submission, age increases linearly with time and decreases when successfully captured by the monitor. Assuming ignoring the queuing time for updates, a packet that ends service prematurely does not reset the age ($t_4 > t_3, s_4 < s_3$). And at the moment packets become in order, the intermediate node submits them all at the same time ($t_4' = t_3' = s_3$), leading to the age at the monitor dropping to the age of the newest update packet.

The main focus of this paper is to compute the time average AoI in such system, which can be represented by the area under the sawtooth waveforms in the Fig. 3. From the above figure, the red line refers to the treatment of the outdated update in the literature, i.e., the age is reset based on the update that ends the service fastest in the system. And the red area refers to the additional age cost that the system pays to maintain the integrity and ordering of the data. The two cases of whether the model is lossy or not are discussed separately. (i) The blocking scheme without the queuing buffer discards all updates that find the system to be full. (ii) The queuing scheme with an infinite buffer keeps incoming updates queuing up to be served.

III. Average Age for $M/M/2$ Blocking Model

A. Preliminaries of Stochastic Hybrid System for AoI

A method introduced in [11] is used to model our system as a stochastic hybrid system (SHS) [12], with which the time average AoI of parallel server queues can be calculated with low complexity. In the SHS model, the hybrid state $(q(t), \mathbf{x}(t))$ is composed of a discrete state and a continuous

state. The discrete state $q(t) \in \mathcal{Q}$ describes a continuous time Markov state related to the queuing system. The continuous state $\mathbf{x}(t) \in \mathbb{R}^{n+1}$ is a row vector capturing the evolution of a collection of age-related processes. This paper follows the simplified SHS in [11] for age analysis where $\mathbf{x}(t)$ is a piece-wise linear process.

The state transition of the system occurs when the packet arrives or completes service. For the Markov state $q(t)$, transition between states via directed transition edges l with transition rate $\lambda^{(l)}$, which indicates that the state changes from q_l to q_l' after an exponentially distributed time with parameter $\lambda^{(l)}$. Note that self-transitions is allowed in SHS. At the same time, transition l causes a discontinuous jump reset to the continuous state from $\mathbf{x}$ to $\mathbf{x}' = \mathbf{x}\mathbf{A}_l$, where $\mathbf{A}_l$ is a square matrix of size $n+1$ by $n+1$. The evolution of continuous states $\mathbf{x}(t)$ is based on $\dot{\mathbf{x}}(t) = \mathbf{b}_q \in \{0,1\}$, which means the age grows in a unit rate in the relevant states and keep unchanged in the unrelated states.

The discrete state probability and its correlation with the continuous state are denoted by

$$\pi_{\hat{q}}(t) = \mathbf{E}[\delta_{\hat{q},q(t)}] = \mathbf{P}[q(t) = \hat{q}], \tag{1}$$

$$\mathbf{v}_{\hat{q}}(t) = [v_{\hat{q}0}(t), ..., v_{\hat{q}n}(t)] = \mathbf{E}[\mathbf{x}(t)\delta_{\hat{q},q(t)}], \tag{2}$$

with the help of the Kronecker delta function δ. Given that the discrete state $q(t)$ is ergodic, the state probability converges to the stationary probability $\bar{\pi}_q$ which satisfying

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

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

According to the Lemma 1 derived in [11], the average age in an ergodic queuing system can be calculated.

Lemma 1: [11] If the discrete-state Markov chain $q(t)$ is ergodic with stationary distribution $\bar{\pi}$ and we can find a non-negative solution $\bar{\mathbf{v}} = [\bar{\mathbf{v}}_0, ..., \bar{\mathbf{v}}_m]$ such that

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

then the average age is given by $\Delta = \sum_{\bar{q} \in \mathcal{Q}} \bar{v}_{\bar{q}0}$.

B. SHS Analysis

Consider a single source with two parallel homogeneous servers system, where a source generates fresh updates as a rate λ Poisson process and the servers are identical with service rate μ. To prevent update packet from aging due to queuing and avoid interference from preempting packets in service, new arrivals are discarded if the servers are both busy. In this setting, this system can be as a $M/M/2$ blocking queue. Referring to the reassignment procedure in [3], an arriving fresh packet is given priority to server 2. Meanwhile the packet at server 2 is reassigned to server 1 if server 1 is idle, otherwise discard the newcomer. Since the two servers are homogeneous and memoryless, reassignment has no impact on the service time of the packet.

The continuous state is $\mathbf{x}(t) = [x_0(t), x_1(t), x_2(t), x_3(t)]$ where $x_0(t)$ is the age at the monitor, $x_i(t)$ ($i \in \{1,2\}$) is the age of the packet at server i, and $x_3(t)$ is the age of the freshest packet waiting in the intermediate node. Apparently, server 2 holds the fresher packet than server 1, i.e. $x_2 \leq x_1$.

For this system, the discrete state is $q(t) = q \in \mathcal{Q} = \{000, 010, 101, 110, 111\}$ which describes the occupancy of servers and the intermediate node. For $q_1q_2q_3 \in \mathcal{Q}$, $q_i = 0$ or 1 ($i \in \{1,2\}$) indicates that the server i is idle and that the server i is serving an update packet, respectively. And q_3 indicates whether any packets were not delivered at the intermediate nodes. In the in-order delivery mode, the update packet must be used by the receiver in the order in which it was generated. Therefore, the packet that completes service in advance needs to wait for the packet generated earlier to complete service and be submitted together. Specially, packet that completes service out of order will be forwarded to the intermediate node, freeing the server to become idle. For convenience, the following re-indexes the state as 0 to 4.

The evolution of the age vector $\mathbf{x}(t)$ in each discrete state is given by

$$\dot{\mathbf{x}}(t) = \mathbf{b}_q = \begin{cases} [1 & 0 & 0 & 0] & q = 000, \\ [1 & 0 & 1 & 0] & q = 010, \\ [1 & 1 & 0 & 1] & q = 101, \\ [1 & 1 & 1 & 0] & q = 110, \\ [1 & 1 & 1 & 1] & q = 111. \end{cases} \tag{6}$$

The interpretation of (6) is that the age at monitor $x_0(t)$ grows at unit rate in each q but $x_i(t)$ increases at unit rate only when server i is busy with a relevant packet or the intermediate node is not empty. Specifically, the age of the packet at the intermediate node continues to grow because it has not been submitted yet.

The SHS Markov chain is shown in Fig. 4 where the transitions are labeled $l \in \{1, 2, ..., 9\}$. When a transition l occurs, the continuous state jumps from $\mathbf{x}$ to $\mathbf{x}' = \mathbf{x}\mathbf{A}_l$ at the rate of $\lambda^{(l)}$. The above parameters are listed in Table I, together with $\mathbf{v}_{q_l}\mathbf{A}_l$ to make simultaneous equations according to Lemma 1.

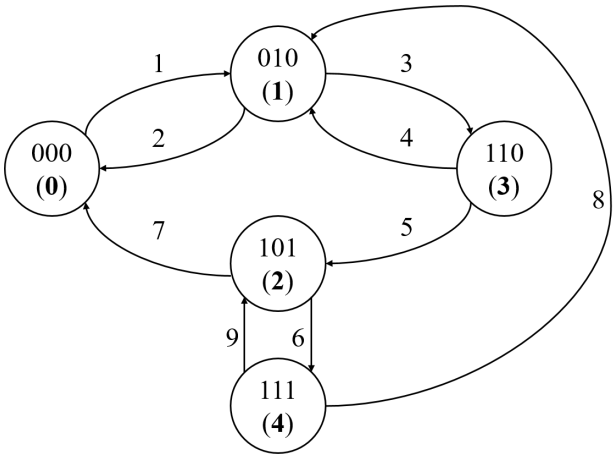

Fig. 4. The SHS Markov chain for the $M/M/2$ blocking system when packets are required to be submitted in order.

l	$q_l \to q_l'$	$\lambda^{(l)}$	$\mathbf{x}' = \mathbf{x}\mathbf{A}_l$	$\bar{\mathbf{v}}_{q_l}\mathbf{A}_l$
1	$0 \to 1$	λ	$[x_0 \quad 0 \quad 0 \quad 0]$	$[\bar{v}_{00} \quad 0 \quad 0 \quad 0]$
2	$1 \to 0$	μ	$[x_2 \quad 0 \quad 0 \quad 0]$	$[\bar{v}_{12} \quad 0 \quad 0 \quad 0]$
3	$1 \to 3$	λ	$[x_0 \quad x_2 \quad 0 \quad 0]$	$[\bar{v}_{10} \quad \bar{v}_{12} \quad 0 \quad 0]$
4	$3 \to 1$	μ	$[x_1 \quad 0 \quad x_2 \quad 0]$	$[\bar{v}_{31} \quad 0 \quad \bar{v}_{32} \quad 0]$
5	$3 \to 2$	μ	$[x_0 \quad x_1 \quad 0 \quad x_2]$	$[\bar{v}_{30} \quad \bar{v}_{31} \quad 0 \quad \bar{v}_{32}]$
6	$2 \to 4$	λ	$[x_0 \quad x_1 \quad 0 \quad x_3]$	$[\bar{v}_{20} \quad \bar{v}_{21} \quad 0 \quad \bar{v}_{23}]$
7	$2 \to 0$	μ	$[x_3 \quad 0 \quad 0 \quad 0]$	$[\bar{v}_{23} \quad 0 \quad 0 \quad 0]$
8	$4 \to 1$	μ	$[x_3 \quad 0 \quad x_2 \quad 0]$	$[\bar{v}_{43} \quad 0 \quad \bar{v}_{42} \quad 0]$
9	$4 \to 2$	μ	$[x_0 \quad x_1 \quad 0 \quad x_2]$	$[\bar{v}_{40} \quad \bar{v}_{41} \quad 0 \quad \bar{v}_{42}]$

TABLE I
TABLE OF TRANSITIONS OF MARKOV CHAIN IN FIGURE

Lemma 2: Suppose that $S_1, S_2, ..., S_n$ are n mutually independent random variables having exponential distribution with parameter $\lambda_1, \lambda_2, ..., \lambda_n$, i.e., $S_i \sim \exp(\lambda_i)$.

Define $T = \min\{S_1, S_2, ..., S_n\}$ and $I = \{i : S_i = T\}$.

Then, the minimum $T \sim \exp(\sum_{k=1}^{n} \lambda_k)$, and the probability $P(I = i) = \frac{\lambda_i}{\sum_{k=1}^{n} \lambda_k}$.

Consider the $M/M/s$ model in queuing theory on the basis of Lemma 2 [13], the transition rate corresponding to the leaving event is $s\mu$ and the probability that any packet completes the service at the earliest is $\frac{1}{s}$ when the servers are fully loaded. Therefore, the transition rates from state $3, 4$ goes to state $1, 2$ are all $2\mu \times \frac{1}{2} = \mu$ in this two server system. We now describe the transitions l according to Table I:

- $l = 1$: A fresh packet is assigned to server 2. The age at the monitor $x_0' = x_0$ is unchanged, $x_1' = 0$ because there is no packet being served on server 1, $x_3' = 0$ because it's irrelevant in state 1, and $x_2' = 0$ because the packet just arriving is fresh new.
- $l = 2$: The packet at server 2 completes service and is delivered to the monitor. The system returns to state 0. The age at the monitor changes to $x_0' = x_2$, and $x_1' = x_2' = x_3' = 0$ because they are all irrelevant in state 0.
- $l = 3$: A fresh packet arrives and is assigned to server 2, $x_2' = 0$. The age at the monitor $x_0' = x_0$ is unchanged, $x_1' = x_2$ because the packet reassigned to server 1 was previously at server 2, and x_3' remains zero for its irrelevance.
- $l = 4$: The packet at server 1 completes service and is delivered to the monitor. The age at the monitor is reset to $x_0' = x_1$, $x_1' = 0$ since server 1 becomes idle, $x_2' = x_2$ because the packet at server 2 is still in service, and $x_3' = 0$ since there are no packets in the intermediate node.
- $l = 5$: The packet at server 2 completes service but is not submitted to the monitor due to the order. The system moves to state 2. The age at the monitor $x_0' = x_0$ since no delivery occurred, $x_1' = x_1$ because the packet remains in place, $x_2' = 0$ on account of its irrelevance in state 2, and $x_3' = x_2$ for the reason that the packet just finishing service at server 2 is forwarded to the intermediate node
- $l = 6$: A fresh packet arrives and is assigned to server 2, $x_2' = 0$. x_0', x_1' and x_3' keep unchanged.
- $l = 7$: The packet at server 1 completes service in state 2, which will be delivered to the monitor together with all packets at the intermediate node at the same time. The age at the monitor $x_0' = x_3$ since the packet at the intermediate node has smaller age, $x_1' = x_2' = x_3' = 0$ since they are all irrelevant in state 0.
- $l = 8$: The packet at server 1 completes service and is delivered to the monitor together with packets hold by the intermediate node, causing the age at the monitor x_0' to drop to x_3, $x_2' = x_2$ keeps unchanged, and $x_1' = x_3' = 0$ for they are irrelevant in state 1.
- $l = 9$: The packet at server 2 completes service and is placed in the intermediate node, x_3' changes to x_2. At this time, x_0' and x_1' keep unchanged, $x_2' = 0$ because it's irrelevant in state 2.

The specific steps to find the average age are given below.

First step, use (3) and (4) to calculate the stationary probability of discrete states $\bar{\pi}$, which is as follows,

$$\begin{aligned}
\bar{\pi}_0 &= \frac{2\mu^2(\lambda+\mu)}{\lambda^3 + 3\lambda^2\mu + 4\lambda\mu^2 + 2\mu^3}, \\
\bar{\pi}_1 &= \frac{\lambda\mu(\lambda+2\mu)}{\lambda^3 + 3\lambda^2\mu + 4\lambda\mu^2 + 2\mu^3}, \\
\bar{\pi}_2 &= \frac{\lambda^2\mu}{\lambda^3 + 3\lambda^2\mu + 4\lambda\mu^2 + 2\mu^3}, \\
\bar{\pi}_3 &= \frac{\lambda^2(\lambda+2\mu)}{2(\lambda^3 + 3\lambda^2\mu + 4\lambda\mu^2 + 2\mu^3)}, \\
\bar{\pi}_4 &= \frac{\lambda^3}{2(\lambda^3 + 3\lambda^2\mu + 4\lambda\mu^2 + 2\mu^3)}.
\end{aligned} \tag{7}$$

Second step, list the equations to solve for $\bar{\mathbf{v}}$ from Equation and Table I that

$$\begin{aligned}
\lambda\bar{\mathbf{v}}_0 &= \bar{\pi}_0\mathbf{b}_0 + \mu[\bar{v}_{12} \quad 0 \quad 0 \quad 0] + \mu[\bar{v}_{23} \quad 0 \quad 0 \quad 0], \\
(\lambda+\mu)\bar{\mathbf{v}}_1 &= \bar{\pi}_1\mathbf{b}_1 + \lambda[\bar{v}_{00} \quad 0 \quad 0 \quad 0] + \mu[\bar{v}_{31} \quad 0 \quad \bar{v}_{32} \quad 0] \\
&\quad + \mu[\bar{v}_{43} \quad 0 \quad \bar{v}_{42} \quad 0], \\
(\lambda+\mu)\bar{\mathbf{v}}_2 &= \bar{\pi}_2\mathbf{b}_2 + \mu[\bar{v}_{30} \quad \bar{v}_{31} \quad 0 \quad \bar{v}_{32}] \\
&\quad + \mu[\bar{v}_{40} \quad \bar{v}_{41} \quad 0 \quad \bar{v}_{42}], \\
2\mu\bar{\mathbf{v}}_3 &= \bar{\pi}_3\mathbf{b}_3 + \lambda[\bar{v}_{10} \quad \bar{v}_{12} \quad 0 \quad 0], \\
2\mu\bar{\mathbf{v}}_4 &= \bar{\pi}_4\mathbf{b}_4 + \lambda[\bar{v}_{20} \quad \bar{v}_{21} \quad 0 \quad \bar{v}_{23}].
\end{aligned} \tag{8}$$

By substituting (6) and (7) into (8), the final result can be obtained after solving 13 equations. With rate λ Poisson packets to the homogeneous parallel server system with service rate μ and normalized load $\rho = \lambda/2\mu$, the average age at monitor is given by

$$\begin{aligned}
\Delta &= \bar{v}_{00} + \bar{v}_{10} + \bar{v}_{20} + \bar{v}_{30} + \bar{v}_{40} \\
&= \frac{\lambda+\mu}{\lambda\mu}\bar{\pi}_0 + \frac{3\lambda^2 + 4\lambda\mu + 2\mu^2}{2\lambda\mu(\lambda+\mu)}(\bar{\pi}_1 + \bar{\pi}_2) \\
&\quad + \frac{4\lambda^2 + 5\lambda\mu + 2\mu^2}{2\lambda\mu(\lambda+\mu)}(\bar{\pi}_3 + \bar{\pi}_4) \\
&= \frac{4\lambda^4 + 11\lambda^3\mu + 14\lambda^2\mu^2 + 12\lambda\mu^3 + 4\mu^4}{2\lambda\mu(\lambda+\mu)(\lambda^2 + 2\lambda\mu + 2\mu^2)} \\
&= \frac{1}{2\mu}\left[\frac{16\rho^4 + 22\rho^3 + 14\rho^2 + 6\rho + 1}{\rho(2\rho+1)(2\rho^2 + 2\rho + 1)}\right].
\end{aligned} \tag{9}$$

IV. AVERAGE AGE FOR $M/M/2$ QUEUING MODEL

A. Graphical Analysis

In this section, an $M/M/2$ model with an infinite queuing buffer is considered. This system will not discard any packets

generated by the source. When generated, packets that find the system busy are queued in the buffer. When leaving the server, packets that complete service in order will be directly submitted to the monitor, and packets that complete service ahead of time will wait at the intermediate node before received by the monitor. The consideration of waiting buffer adds a factor of queuing time to the delay of packets, which brings the risk of congestion to the system.

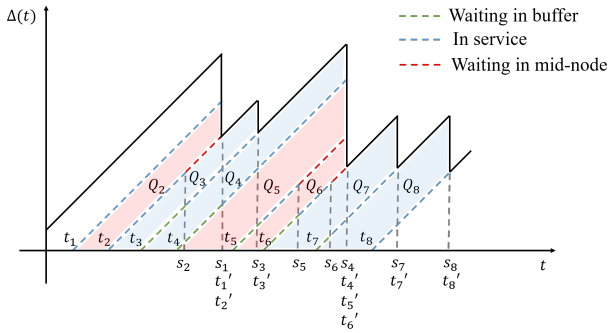

Fig. 5. The sample path of AoI in $M/M/2$ queuing model.

Fig. 5 shows the evolution and composition of AoI in the $M/M/2$ queuing model. In order to more intuitively see what constitutes the system time of each packet, three different colored lines are used to indicate the process of a packet being queued, serviced, and waiting to be delivered. As mentioned in Section II, the packet i arrive into the system at time t_i, where the interarrival time $t_i - t_{i-1}$ is i.i.d. and follows a negative exponential distribution with parameter λ. The service time of each packet is also an exponential i.i.d. random variable with parameter μ. The main difference in the in-order delivery mode is that the time packet i delivered to the monitor successfully is not necessarily equal to the time it departs the server due to the ordering requirement. As marked in Fig. 5, $t_2' \neq s_2$ because the packet 2 need to wait in the intermediate node until the packet 1 completes service.

With the help of the graphical method in [14], the integral area are divided into the sum of several trapezoidal areas $Q_1, Q_2, ..., Q_n$. Denote the queue waiting time, interarrival time, service time and additional waiting time of the ith packet as W_i, X_i, S_i and A_i, respectively, then its system time can be written as $T_i = W_i + S_i + A_i$, from which the time average AoI can be calculated by

$$\Delta = \lambda(\mathbf{E}[X^2]/2 + \mathbf{E}[XT]). \tag{10}$$

Considering the computational complexity of (10), we classify all the packets in the system into the following two categories:

- Type a: This kind of packets can be delivered to the monitor as soon as they complete service, i.e., the time they spend at the intermediate node is zero because they are the oldest packets in the system at the moment. For example, packet 1, 3, 4, 7 and 8 in Fig. 5.
- Type b: This kind of packets should wait for a while in the intermediate node before delivery since there is a packet carrying earlier timestamp is still being served. For example, packet 2, 5 and 6 in Fig. 5.

As for the system time for these two types of packets,

$$T_{i,a} = W_{i,a} + S_{i,a}, \tag{11}$$

$$\begin{aligned} T_{i,b} &= W_{i,b} + S_{i,b} + A_{i,b} \\ &= W_{i-n,a} + S_{i-n,a} - \sum_{k=i-n+1}^{i} X_{k,b}, \end{aligned} \tag{12}$$

where $i - n$ is the index of the packet which keeps the ith packet waiting before delivery. The transformation of (12) eliminates the variable $A_{i,b}$ and reduces uncertainty in calculations. Based on the derivation method in [2], we can express the overall average AoI by finding the average AoI for each type as follows,

$$\begin{aligned} \Delta &= \lambda \left(p_a \left(\frac{\mathbf{E}[X_{i,a}^2]}{2} + \mathbf{E}[X_{i,a}T_{i,a}] \right) \right. \\ &\quad \left. + p_b \left(\frac{\mathbf{E}[X_{i,b}^2]}{2} + \mathbf{E}[X_{i,b}T_{i,b}] \right) \right) \\ &= \lambda \left(p_a \left(\frac{\mathbf{E}[X_a^2]}{2} + \mathbf{E}[W_a X_a] + \mathbf{E}[X_a S_a] \right) \right. \\ &\quad + p_b \left(\frac{\mathbf{E}[X_b^2]}{2} + \mathbf{E}[X_b W_a] + \mathbf{E}[X_b S_a] \right. \\ &\quad \left. \left. - \mathbf{E}[X_b] \sum_{n=1}^{\infty} \mathbf{E}\left[\sum_{k=1}^{N} X_b | N = n \right] \mathbf{Pr}(N = n) \right) \right), \end{aligned} \tag{13}$$

where p_a, p_b are the probability that a packet is of type a or b, respectively.

B. The Average AoI Computation

1) Probability of a Certain Packet Type: To isolate the impact of queuing time when analyzing probability of some events, we borrow ideas from [2] and consider two different cases that whether there is a packet in the system just prior to the start of service of packet i. Let M_i denote the number of packets in the system on the eve of the packet i being served.

- $M_i = 0$: The probability that there are no packets in the system before the packet i is about to start service is equal to the steady state probability that the system is empty, which can be found in [15] that

$$p_0 = \left[\sum_{k=0}^{s-1} \frac{1}{k!} \left(\frac{\lambda}{\mu} \right)^k + \frac{1}{s!} \frac{1}{1-\rho} \left(\frac{\lambda}{\mu} \right)^s \right]^{-1} \tag{14}$$

  where $\rho = \frac{\lambda}{s\mu}$ is the system utilization and s takes 2 in the $M/M/2$ system. Any packet that starts service in this case is of type a because no packet generated earlier than it is still being served in the system.
- $M_i \neq 0$: Obviously, the probability that the system is busy with a packet before the packet i starts service is $1 - p_0$. In this case, the probability that a packet is of type a or b is $\frac{1}{2}$ due to the memoryless property.

In summary, the probability of a certain packet type is

$$p_a = p_0 \times 1 + (1 - p_0) \times \frac{1}{2} = \frac{1}{2}(1 + p_0), \tag{15}$$

$$p_b = p_0 \times 0 + (1 - p_0) \times \frac{1}{2} = \frac{1}{2}(1 - p_0). \tag{16}$$

2) Expected values of Service Time: The service time for a packet of type a or b does not have the same distribution as a typical service time since different types of packets have different constraints. Based on Bayes' theorem, we obtain the following conditional mean of service time for two types of packets,

$$\mathbf{E}[S_a] = \frac{p_0+3}{2(p_0+1)}\frac{1}{\mu}, \tag{17}$$

$$\mathbf{E}[S_b] = \frac{1}{2\mu}. \tag{18}$$

Proof: Denote the event that a packet i belongs to type a and b by $A(i)$ and $B(i)$, respectively. According to Bayes' formula that $P(X|Y)P(Y) = P(Y|X)P(X)$, here comes

$$\begin{aligned} &f_{S|A(i)}(S_i = s|A(i))\mathbf{Pr}(A(i)) \\ &= \mathbf{Pr}(A(i)|S_i = s)f_S(s) \\ &= p_0 f_S(s) + (1-p_0)\mathbf{Pr}(S_i > S_{i-k}|S_i = s)f_S(s). \end{aligned} \tag{19}$$

Therefore, the expected value is

$$\begin{aligned} \mathbf{E}(S_a) &\triangleq \mathbf{E}(S_i = s|A(i)) \\ &= \frac{\int_0^\infty s(p_0 + (1-p_0)(1-e^{-\mu s}))\mu e^{-\mu s}\mathrm{d}s}{p_a} \\ &= \frac{p_0\frac{1}{\mu} + (1-p_0)(\frac{1}{\mu} - \frac{1}{4\mu})}{p_0 + \frac{1}{2}(1-p_0)} \end{aligned} \tag{20}$$

Similarly, the mean service time of the packet of type b is

$$\mathbf{E}(S_b) = \frac{(1-p_0)\int_0^\infty s e^{-\mu s}\mu e^{-\mu s}\mathrm{d}s}{(1-p_0)\int_0^\infty e^{-\mu s}\mu e^{-\mu s}\mathrm{d}s} = \frac{1}{2\mu}. \tag{21}$$

■

3) Expected values of Interarrival Time: Computing the arrival time distribution for a packet of type a and b involves a joint distribution with the queuing time, which is too difficult to derive. So we assume the type of a packet is determined by $X_i + S_i \leqslant S_{i-k}$, which is reasonable when the server load is relatively small because the queuing time can be ignored at this situation. Under this approximation, the conditional mean interarrival time and its square can be written as follows,

$$\mathbf{E}[X_a] = \frac{1}{\lambda} + \frac{1}{\lambda+\mu} - \frac{1}{\lambda+2\mu}, \tag{22}$$

$$\mathbf{E}[X_b] = \frac{1}{\lambda+\mu}, \tag{23}$$

$$\mathbf{E}[X_a^2] = \frac{2}{\lambda^2} + \frac{2}{(\lambda+\mu)^2} + \frac{2}{\lambda(\lambda+\mu)} - \frac{6}{\lambda(\lambda+2\mu)}, \tag{24}$$

$$\mathbf{E}[X_b^2] = \frac{2}{(\lambda+\mu)^2}. \tag{25}$$

The derivation of above conditional distribution is consistent with that of the service time.

4) Expected value of Cumulative Sum: The last term of Eq. (13) involves computing the expected value of the accumulated sum. The number of packets between a type b packet and its previous most recent type a packet is denoted by $N-1$, where N is the number of interarrival times that need to be summed. Since the probability of a packet being of a certain type depends on the number of packets in the system when it is going to be served, the events of what type a packet precedes are independent of each other. Therefore, N follows a geometric distribution that

$$\mathbf{Pr}(N = n) = p_a \times p_b^{n-1} = p_a(1-p_a)^{n-1}. \tag{26}$$

With the above result, the expected value can be easily obtained as

$$\begin{aligned} &\mathbf{E}[X_{i,b}\sum_{k=i-n+1}^{i} X_{k,b}] \\ &= \mathbf{E}[X_{i,b}^2] + \mathbf{E}[X_{i,b}]\mathbf{E}[\sum_{k=i-n+1}^{i} X_{k,b}] \\ &= \mathbf{E}[X_b^2] + \mathbf{E}[X_b]\sum_{n=2}^{\infty}(n-1)\mathbf{E}[X_b]p_a(1-p_a)^{n-1} \\ &= \mathbf{E}[X_b^2] + \mathbf{E}[X_b]^2\frac{1-p_a}{p_a}. \end{aligned} \tag{27}$$

5) Approximation of Average Age: Due to the complexity of computing the expected value of the product of two random variables, some components of the final average age expression are approximated. The interarrival time of a packet is related to its own queuing waiting time, and is independent of the queuing waiting time of its preceding packets, i.e.,

$$\begin{aligned} \mathbf{E}[X_a W_a] &\approx \mathbf{E}[XW] \\ &= \int_0^\infty x\mathbf{E}[W_i|X_i = x]f_X(x)\mathrm{d}x \\ &= \frac{\lambda^2}{2\mu^2(2\mu+\lambda)(2\mu-\lambda)}, \end{aligned} \tag{28}$$

$$\begin{aligned} \mathbf{E}[X_b W_a] &\approx \mathbf{E}[X]\mathbf{E}[W] \\ &= \frac{1}{\lambda} \times \frac{\lambda^2}{\mu(2\mu+\lambda)(2\mu-\lambda)}, \end{aligned} \tag{29}$$

where the distribution about W are taken from [2] and [15]. Another approximation comes from the expectation of multiplying X_t and S_t $(t \in a, b)$ for ignoring the effect of X_i on whether the system is empty just prior to a packet starting service.

$$\mathbf{E}[X_a S_a] \approx \mathbf{E}[X]\mathbf{E}[S_a], \tag{30}$$

$$\mathbf{E}[X_b S_a] \approx \mathbf{E}[X]\mathbf{E}[S_a]. \tag{31}$$

The final approximate average AoI can be obtained by substituting Eq. (24)-(31) into Eq. (13).

V. Simulation Results

We run the simulation of duration 100,000 and compare the time average age obtained from the Monte Carlo simulation with the theoretical results when μ takes 0.5 and 1, respectively. The time average age Δ as a function of the server utilization $\rho = \lambda/2\mu$ under two models are plotted below.

Fig. 6 verifies the time average age for $M/M/2$ blocking model. It is shown that the theoretically evaluated results are almost consistent with the simulation results, which proves the feasibility and correctness of the solution using the SHS approach. In addition, the blocking mode keeps the average

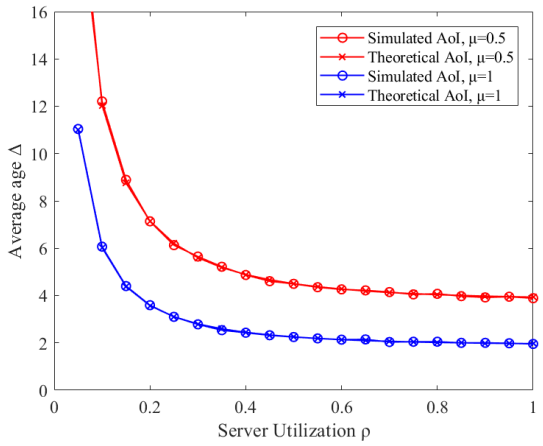

Fig. 6. The time average AoI for $M/M/2$ blocking model.

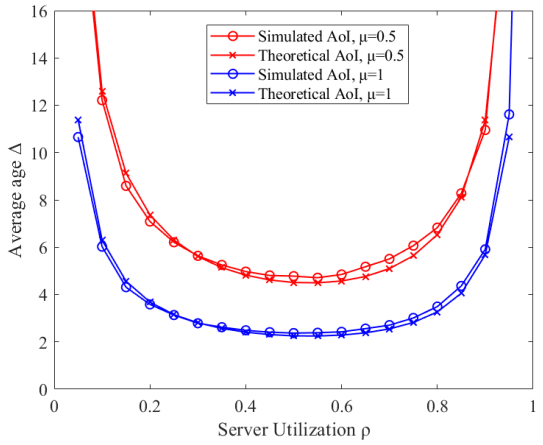

Fig. 7. The time average AoI for $M/M/2$ queuing model.

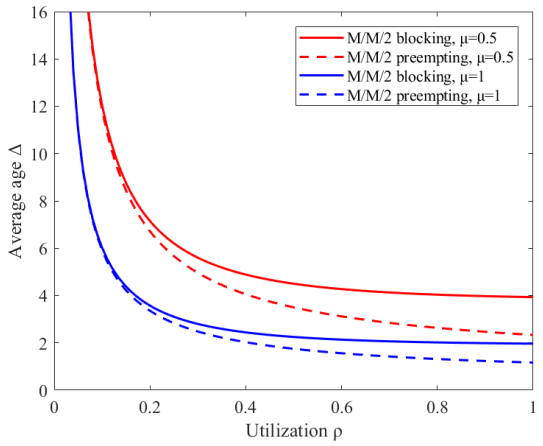

Fig. 8. The time average AoI for $M/M/2$ blocking model versus that for $M/M/2$ preempting model.

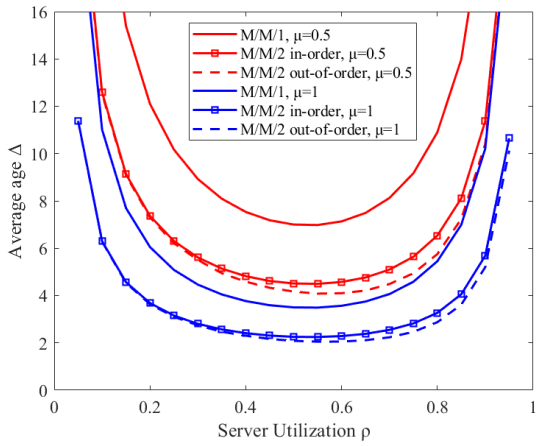

Fig. 9. The time average AoI for $M/M/2$ queuing model compared with that for the conventional out-of-order $M/M/2$ model and the $M/M/1$ model.

age from trending to infinity as the arrival rate λ increases at the expense of the receding rate.

Fig. 7 presents the comparison between the approximate age of the $M/M/2$ queuing system and the simulation results. Despite the gap, the approximate results are generally close to the actual simulated age. The more obvious gap at high arrival rate comes from the approximate analysis based on low arrival rate mentioned in Section IV.

In Fig. 8, the time average age of the $M/M/2$ blocking system and the results of the $M/M/2$ preempting systems from in [3] are compared. As the update frequency getting progressively faster, the preempting scheme reduces the age more than the blocking scheme with in-order delivery since it is always serving the newest packets.

In Fig. 9, the approximate time average age of the $M/M/2$ queuing model are compared with that of the model without completeness and in-order requirements from [2] and the $M/M/1$ age. It is shown that the performance of our model is actually the upper bound of the results of the $M/M/2$ out-of-order model in the literature, which is reasonable because under the common $M/M/2$ model, the in-order delivery mode takes more time to wait for sorting at the intermediate node before the monitor. The relative AoI gap of in-order delivery is within 14.2%. Conventional model can achieve lower age but suffers from packet loss. And an additional server makes the age roughly half of that in the single route case where the order is ensured.

VI. Conclusions

This paper analyzes the time average AoI in parallel homogeneous server queue under the in-order delivery mode. The models with and without queuing buffers are considered separately. For $M/M/2$ blocking system, the SHS approach are used to obtain the expression of the average age by first describing the occupancy status of service facilities and the continuous time variation of age-related processes, and then solving a set of simple age balance equations. For $M/M/2$ queuing system, the AoI is evaluated by a graphical method and a quite close approximation of the average age is derived.

In general, our research is the first to consider the effect of in-order delivery on the AoI performance. The above theoretical results have been verified by simulation. Furthermore, by comparison with the results in the existing literature, it is

found that the blocking scheme does not waste any network service resources despite its AoI performance is inferior to the preempting scheme. The blocking model can also be extended to more servers, although the number of discrete states when solving for age will also increases by the power of 2. As for the queuing model with infinite buffer, the average age of the system is affected due to the maintenance of data integrity. The performance loss of average age with in-order delivery is less than 14.2% compared to out-of-order delivery.

ACKNOWLEDGMENT

This work was supported by the National Key Research and Development Program of China under Grants 2019YFE0196600, the National Natural Science Foundation of China (NSFC) under Grants 62071284, 61871262, 61901251 and 61904101, the program for Professor of Special Appointment (Eastern Scholar) at Shanghai Institutions of Higher Learning, and Shanghai Institute for Advanced Communication and Data Science (SICS).

REFERENCES

[1] C. Kam, S. Kompella, and A. Ephremides, "Age of information under random updates," in *2013 IEEE International Symposium on Information Theory*, pp. 66–70, IEEE, 2013.

[2] C. Kam, S. Kompella, G. D. Nguyen, and A. Ephremides, "Effect of message transmission path diversity on status age," *IEEE Transactions on Information Theory*, vol. 62, no. 3, pp. 1360–1374, 2015.

[3] R. D. Yates, "Status updates through networks of parallel servers," in *2018 IEEE International Symposium on Information Theory (ISIT)*, pp. 2281–2285, IEEE, 2018.

[4] A. Javani, M. Zorgui, and Z. Wang, "Age of information in multiple sensing," in *2020 Information Theory and Applications Workshop (ITA)*, pp. 1–10, IEEE, 2020.

[5] A. M. Bedewy, Y. Sun, and N. B. Shroff, "Optimizing data freshness, throughput, and delay in multi-server information-update systems," in *2016 IEEE International Symposium on Information Theory (ISIT)*, pp. 2569–2573, IEEE, 2016.

[6] Y. Sun, E. Uysal-Biyikoglu, and S. Kompella, "Age-optimal updates of multiple information flows," in *IEEE INFOCOM 2018-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pp. 136–141, IEEE, 2018.

[7] M. Agiwal, H. Kwon, S. Park, and H. Jin, "A survey on 4g-5g dual connectivity: Road to 5g implementation," *IEEE Access*, vol. 9, pp. 16193–16210, 2021.

[8] S. Agrawal and R. Ramaswamy, "Analysis of the resequencing delay for m/m/m systems," in *Proceedings of the 1987 ACM SIGMETRICS conference on Measurement and modeling of computer systems*, pp. 27–35, 1987.

[9] I. Iliadis and L.-C. Lien, "Resequencing delay for a queueing system with two heterogeneous servers under a threshold-type scheduling," *IEEE Transactions on Communications*, vol. 36, no. 6, pp. 692–702, 1988.

[10] S. Chowdhury, "The mean resequencing delay for m/h/sub k//infinity systems," *IEEE transactions on software engineering*, vol. 15, no. 12, pp. 1633–1638, 1989.

[11] R. D. Yates and S. K. Kaul, "The age of information: Real-time status updating by multiple sources," *IEEE Transactions on Information Theory*, vol. 65, no. 3, pp. 1807–1827, 2018.

[12] J. P. Hespanha, "Modelling and analysis of stochastic hybrid systems," *IEE Proceedings-Control Theory and Applications*, vol. 153, no. 5, pp. 520–535, 2006.

[13] V. G. Kulkarni, *Modeling, analysis, design, and control of stochastic systems*. Springer, 2014.

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

[15] L. Kleinrock, *Queueing systems: theory*. John Wiley, 1975.