

Status Updating with Time Stamp Errors

Md. Nurul Absar Siddiky Ahmed Arafa
 Department of Electrical and Computer Engineering
 University of North Carolina at Charlotte, NC 28223
 msiddiky@charlotte.edu aarafa@charlotte.edu

Abstract—A status updating system is considered in which multiple processes are sampled and transmitted through a shared channel. Each process has its dedicated server that processes its samples before time stamping them for transmission. Time stamps, however, are prone to errors, and hence the status updates received may not be credible. Our setting models the time stamp error rate as a function of the servers' busy times. Hence, to reduce errors and enhance credibility, servers need to process samples on a relatively prolonged schedule. This, however, deteriorates timeliness, which is captured through the age of information (AoI) metric. An optimization problem is formulated whose goal to characterize the optimal processes' schedule and sampling instances to achieve the optimal trade-off between timeliness and credibility. The problem is first solved for a single process setting, where it is shown that a *threshold-based sleep-wake schedule* is optimal, in which the server wakes up and is allowed to process newly incoming samples only if the AoI surpasses a certain threshold that depends on the required timeliness-credibility trade-off. Such insights are then extended to the multi-process setting, where two main scheduling and sleep-wake policies, namely round-robin scheduling with threshold-waiting and asymmetric scheduling with zero-waiting, are introduced and analyzed.

I. INTRODUCTION

Several current and emerging applications in communications, networking and control require timely information processing and transfer in order to accurately achieve their goals. This has led to the emergence of the age of information (AoI) metric to assess the freshness of data at the destinations [1], defined as the difference between the current time and that of the time stamp of the latest received data [2]. In time-sensitive applications, it is crucial to measure the AoI accurately in order to take timely decisions. However, when time stamps are erroneous, the AoI value becomes unreliable, and the credibility of the decision-making process becomes questionable. In this work, we introduce the notion of *timeliness-credibility* trade-off through modeling analyzing the effects of time stamp errors on AoI optimization in a system where multiple processes are monitored through a shared communication channel, see Fig. 1.

Optimizing AoI, or maximizing timeliness and freshness of data, has been considered in a plethora of works in the literature. The pioneering work in queuing networks [2] and what follows in that line of research have shown that AoI-optimal policies are neither throughput-optimal (high server utilization) nor delay-optimal (low server utilization). Rather,

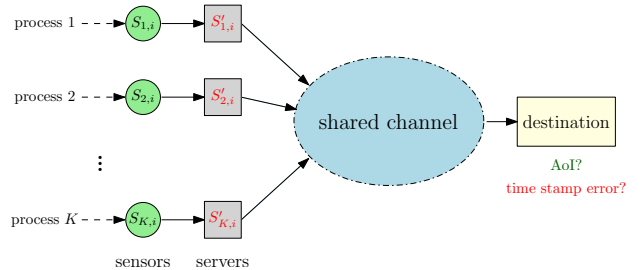


Fig. 1: System model: process k 's i th sample arrives at time $S_{k,i}$, yet is time-stamped as $S'_{k,i}$ by its server.

AoI aims at balancing server utilization to deliver fresh data. Other lines of research to which these ideas are extended include, e.g., energy harvesting communications [3], federated learning [4], gossip networks [5], data trading [6], coding [7], internet-of-things (IoT) networks [8], random access networks [9], edge computing [10], and privacy-preserving systems [11].

A notable challenge to achieve accurate AoI is the issue of *timestamping* [12], where timestamps on data packets are intentionally falsified, typically as part of an adversarial attack. This manipulation can occur through several mechanisms, including falsifying timestamps to make stale data appear fresh, introducing network delays, or due to natural factors such as sensor malfunctions or synchronization issues. These timestamp errors lead to inaccurate AoI calculations, which in turn result in misleading assessment of data freshness. This performance degradation is particularly concerning in energy-constrained IoT systems, remote monitoring, and decentralized networks, where accurate updates are critical. In [13], the effects of adversarial timestamp manipulation in gossip networks have been studied, demonstrating that even a compromised node in fully connected networks can significantly increase AoI and worsen how it scales with the size of the network.

Although timestamp manipulation has been explored in gossip networks, there is a lack of research on how timestamp inaccuracies affect AoI in conventional update systems. To address this gap, our work investigates the *optimization of both AoI and timestamp accuracy*, with the goal of enhancing system reliability and timeliness. Minimizing AoI alone is insufficient when timestamp errors are present, as these errors can lead to poor decision making in critical applications such as remote sensing and energy-constrained systems. Therefore, we propose integrating timestamp error management into AoI optimization to provide more accurate and reliable decision-making and system performance.

Specifically, we consider a system in which time stamps from multiple processes are sent through a shared channel towards a destination. Each process has a dedicated server to process its samples and assign them time stamps before sending them on the channel. A server introduces time stamp errors with a rate that depends on its busy time. That is, to reduce the errors, a server needs to *sleep* for a while. This, in turn increases the AoI and reduces timeliness. Hence, a trade-off arises between minimizing AoI and minimizing time stamp errors. We introduce an optimization problem to characterize the optimal trade-off by optimizing the sleep-wake schedules of the servers. We first solve the problem for the single process setting. Towards that end, we show that the optimal sleep-wake schedule has a *threshold structure*: the server wakes up only if the AoI surpasses a certain threshold that depends on the target credibility of time stamps. We then build on these insights and present two main scheduling policies for the multiple-source setting: round-robin scheduling with threshold-waiting, and asymmetric scheduling with zero-waiting. We analyze and compare the performances of both policies and show that the optimal choice between them highly depends on the system parameters and the target time stamp credibility.

II. SYSTEM MODEL AND PROBLEM FORMULATION

We consider a status update system composed of K sensors and K servers, as shown in Fig. 1. Sensor k receives samples from a λ_k -Poisson process, and passes them to server k for time-stamping and transmission. Transmissions go through a shared communication channel that adds a random delay and can only be utilized by one server at a time.

Let π denote the transmission schedule. This schedule also defines a sleep-wake policy for servers (and sensors); to avoid samples becoming stale, sensor k does not acquire new samples and goes into sleep mode unless it is scheduled to transmit according to π . More precisely, when sensor k 's turn comes up for the i th time, it may choose to continue to sleep (or wait) for $W_{k,i}$ extra time units, after which it wakes up and becomes ready to receive new samples. Then, it receives its i th sample after $X_{k,i}$ time units. Note that $X_{k,i}$'s are independent and identically distributed (i.i.d.), $\sim \exp(\lambda_k)$ across process k samples. We denote by $\{W_{k,i}\}$ the servers' waiting policy.

Now, let $S_{k,i}$ denote the arrival time of the i th sample of the k th process. Such sample gets served immediately upon arrival, and reaches the destination at time

$$D_{k,i} = S_{k,i} + Y_{k,i}, \quad (1)$$

where $Y_{k,i}$'s are i.i.d. across samples and processes, denoting channel busy times. We assume that the destination is *unaware* of the values of the channel busy times $\{Y_{k,i}\}$; only the sensor-server side is aware.

Servers may introduce time stamp errors, in which the *received* time stamp of the process k 's i th sample is given by $S'_{k,i}$ as opposed to the true time stamp $S_{k,i}$. Errors occur at a rate that depends on the sleep-wake schedule of the sensors as

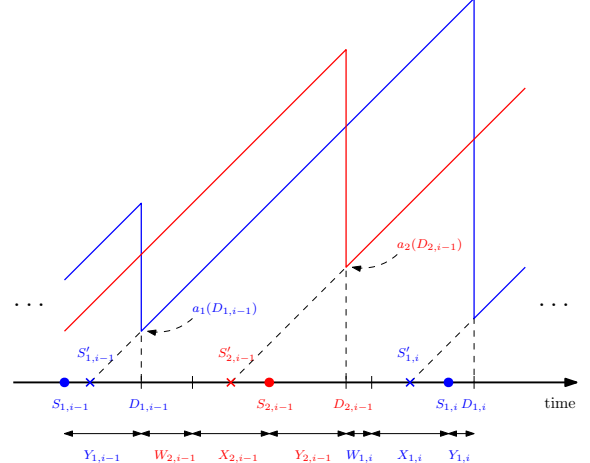


Fig. 2: Example AoI evolution at the destination for $K = 2$ processes (1 in blue; 2 in red). Filled circles denote true time stamps and crosses denote received (erroneous) time stamps.

we explain below. Statistically, we assume that $S'_{k,i}$ and $S_{k,i}$ are related as follows:

$$\mathbb{E}[S'_{k,i} | S_{k,i}, D_{k,i}] = S_{k,i}, \quad (2)$$

$$\text{Var}(S'_{k,i} | S_{k,i}, D_{k,i}) = h_k(S_{k,i} - X_{k,i} - S_{k,i-1}), \quad (3)$$

where $\mathbb{E}[\cdot]$ and $\text{Var}(\cdot)$ denote expectation and variance, respectively, and $h_k(\cdot)$ is some monotonically *decreasing convex* function. Hence, the server's introduced (and received) time stamp $S'_{k,i}$ is unbiased from the true time stamp $S_{k,i}$, yet its variance is inversely proportional with the inter-sampling duration. Therefore, to reduce errors of a certain server, one needs to reduce its sampling rate. The rationale is that *errors occur more often when servers do not get enough sleep time*. Such approach has been considered previously in, e.g., use-dependant channels [14]. While our results are presented for general functions $h_k(\cdot)$, our experiments will be focusing on exponentially-decaying functions given by

$$h_k(x) = e^{-\alpha_k x}, \quad (4)$$

for some parameter $\alpha_k \geq 0$ denoting server k 's *recovery rate*. That is, higher values of α_k represent faster recovery, in which server k introduces relatively less errors and can tolerate being awake for relatively longer periods of time, and vice versa.

We assess timeliness at the destination using AoI. For process k , the AoI at time t is defined as

$$a_k(t) = t - S'_{k,i}, \quad D_{k,i} \leq t < D_{k,i+1}. \quad (5)$$

An example of the AoI evolution at the destination for $K = 2$ processes is shown in Fig. 2. Note that due to time-stamping errors, the AoI value seen at the destination may not represent the true value of the AoI. We define by an *epoch* the time elapsed in between two consecutive deliveries of samples from a specific process. Let us denote by

$$L_{k,i} \triangleq D_{k,i} - D_{k,i-1} \quad (6)$$

the length of the i th epoch for process k . We are interested in

the long-term average AoI for process k defined by the area under its AoI curve. From Fig. 2, such quantity is given by

$$\overline{\text{AoI}}_k = \limsup_{n \rightarrow \infty} \frac{\sum_{i=1}^n \mathbb{E}[a_k(D_{k,i-1}) L_{k,i}] + \frac{1}{2} \mathbb{E}[L_{k,i}^2]}{\sum_{i=1}^n \mathbb{E}[L_{k,i}]} \quad (7)$$

As we can see from the above, the timeliness measured at the destination is *not always credible* due to time stamping errors. Therefore, one also needs to measure the long-term average error for a specific process when evaluating its timeliness. Such quantity is given by

$$\bar{\epsilon}_k = \limsup_{n \rightarrow \infty} \frac{1}{n} \sum_{i=1}^n \mathbb{E}[(S_{k,i} - S'_{k,i})^2] \quad (8)$$

Using (2) and (3), observe that one can reduce the value of $\bar{\epsilon}_k$ by increasing the inter-sampling duration. However, this may negatively impact timeliness. Hence, *a trade-off arises between timeliness and credibility*. Note that a schedule π combined with a waiting policy $\{W_{k,i}\}$ completely characterize the values of $\overline{\text{AoI}}_k$ and $\bar{\epsilon}_k$ for all processes. Our main goal is to optimize a weighted average of timeliness and credibility. That is, to solve the following optimization problem:

$$\min_{\pi, \{W_{k,i} \geq 0\}} \sum_{k=1}^K \beta_k \overline{\text{AoI}}_k + (1 - \beta_k) \bar{\epsilon}_k, \quad (9)$$

for some $\beta_k \in [0, 1]$, $\forall k$.

We first solve the single-process version of the above problem in the next section. Then, we present solutions for the multi-process version in the following one.

III. THE SINGLE PROCESS SETTING

In the case of $K = 1$ process, we drop the index k from all the variables, and drop the schedule π from the optimization problem in (9). That is, the only variable of the optimization problem in (9) is now the waiting policy $\{W_i\}$.

Towards characterizing the long-term average AoI, observe that the starting AoI of epoch i is given by

$$\begin{aligned} a(D_{i-1}) &= \begin{cases} Y_{i-1} - (S'_{i-1} - S_{i-1}), & S'_{i-1} \geq S_{i-1} \\ Y_{i-1} + S_{i-1} - S'_{i-1}, & S'_{i-1} < S_{i-1} \end{cases} \\ &= Y_{i-1} + S_{i-1} - S'_{i-1}. \end{aligned} \quad (10)$$

Next, we focus on *stationary deterministic* waiting policies in which the waiting time in epoch i is given by a deterministic function of the channel busy time in epoch $i - 1$. That is,

$$W_i \triangleq \omega(Y_{i-1}), \quad (11)$$

for some function $\omega(\cdot)$ to be optimized. Such waiting policy has been shown optimal in similar settings of AoI optimization [11], in which the channel busy times are i.i.d.¹ Such choice

¹Observe that the waiting policy is determined completely at the sensor-server side where full knowledge of the channel busy times Y_i 's is provided. Hence, the policy is implementable in our setting.

of waiting policies induces a stationary distribution across epochs. Specifically, the i th epoch length is now given by

$$L_i = \omega(Y_{i-1}) + X_i + Y_i, \quad (12)$$

and the long-term average AoI now reduces to the following:

$$\overline{\text{AoI}} = \frac{\mathbb{E}[a(D_{i-1}) L_i] + \frac{1}{2} \mathbb{E}[L_i^2]}{\mathbb{E}[L_i]}. \quad (13)$$

Now let us further analyze the term $\mathbb{E}[a(D_{i-1}) L_i]$ in the numerator above. Using (2) and (3), one can see that the time stamp error $S_{i-1} - S'_{i-1}$ only depends on channel busy time Y_{i-2} , and is therefore independent from L_i . Further, one can show that its average value is equal to 0 as follows:

$$\begin{aligned} \mathbb{E}[S_{i-1} - S'_{i-1}] &= \mathbb{E}[\mathbb{E}[S_{i-1} - S'_{i-1} | S_{i-1}, D_{i-1}]] \\ &= \mathbb{E}[\mathbb{E}[S'_{i-1} | S_{i-1}, D_{i-1}] - \mathbb{E}[S'_{i-1} | S_{i-1}, D_{i-1}]] = 0. \end{aligned} \quad (14)$$

Therefore, using (10) we get that

$$\mathbb{E}[a(D_{i-1}) L_i] = \mathbb{E}[Y_{i-1} L_i]. \quad (15)$$

In other words, the time stamp error, on average, does not affect the long-term average AoI viewed from the destination.²

We now turn to the long-term average error. Since we have stationary distributions across epochs, we get

$$\begin{aligned} \bar{\epsilon} &= \mathbb{E}[(S_i - S'_i)^2] \\ &= \mathbb{E}[\mathbb{E}[(S_i - S'_i)^2 | S_i, D_i]] \\ &= \mathbb{E}[\mathbb{E}[(\mathbb{E}[S'_i | S_i, D_i] - S_i)^2 | S_i, D_i]] \\ &= \mathbb{E}[\text{Var}(S'_i | S_i, D_i)] \\ &= \mathbb{E}[h(S_i - X_i - S_{i-1})] \\ &= \mathbb{E}[h(Y_{i-1} + \omega(Y_{i-1}))]. \end{aligned} \quad (16)$$

The optimization problem is now given by

$$\min_{\omega(\cdot) \geq 0} \overline{\text{AoI}} + \frac{1 - \beta}{\beta} \bar{\epsilon}, \quad (17)$$

which can be equivalently represented as

$$\min_{\omega(\cdot) \geq 0} \overline{\text{AoI}}, \quad \text{s.t. } \bar{\epsilon} \leq \tau, \quad (18)$$

for some $\tau \geq 0$ [15]. We call the constraint in (18) the *credibility constraint*. We focus on analyzing the second formulation in (18) in the remainder of this section. Specifically, we follow Dinkelbach's approach [16] to transform the problem into the following auxiliary one:

$$\begin{aligned} p(\theta) &\triangleq \min_{\omega(\cdot) \geq 0} \mathbb{E}[(Y_{i-1} - \theta)(\omega(Y_{i-1}) + X_i + Y_i)] \\ &\quad + \frac{1}{2} \mathbb{E}[(\omega(Y_{i-1}) + X_i + Y_i)^2] \\ \text{s.t. } &\mathbb{E}[h(Y_{i-1} + \omega(Y_{i-1}))] \leq \tau, \end{aligned} \quad (19)$$

for some $\theta \in \mathbb{R}$. The optimal solution of problem (18) is now given by the unique θ^* that solves $p(\theta^*) = 0$, which can be

²This, however, necessitates the addition of a credibility measure as in (8).

found by, e.g., a bisection search.

The objective function of problem (19) can be further simplified as follows:

$$\begin{aligned} & \mathbb{E}[(Y_{i-1} - \theta)\omega(Y_{i-1})] + (\mu_Y - \theta)\left(\frac{1}{\lambda} + \mu_Y\right) \\ & + \frac{1}{2}\mathbb{E}[\omega(Y_{i-1})^2] + \mathbb{E}[\omega(Y_{i-1})]\left(\frac{1}{\lambda} + \mu_Y\right) \\ & + \frac{1}{\lambda^2} + \frac{1}{\lambda}\mu_Y + \frac{1}{2}\mu_{Y^2}, \end{aligned} \quad (20)$$

where μ_Y and μ_{Y^2} denote the first and second moments of Y_i , respectively. We now introduce the following Lagrangian for problem (19):

$$\begin{aligned} \mathcal{L} = & \int \left(\left(y - \theta + \frac{1}{\lambda} + \mu_Y \right) \omega(y) + \frac{1}{2}\omega(y)^2 \right) f_Y(y) dy \\ & + (\mu_Y - \theta)\left(\frac{1}{\lambda} + \mu_Y\right) + \frac{1}{\lambda^2} + \frac{1}{\lambda}\mu_Y + \frac{1}{2}\mu_{Y^2} \\ & + \gamma \left(\int h(y + \omega(y)) f_Y(y) dy - \tau \right) - \int \eta(y)\omega(y) dy, \end{aligned} \quad (21)$$

where $f_Y(y)$ denotes the distribution of Y_i , whereas γ and $\eta(y)$ are Lagrange multipliers. Taking the functional derivative of the above with respect to $\omega(y)$, equating to 0, and rearranging, we get

$$y + \omega(y) + \gamma h'(y + \omega(y)) = \theta - \frac{1}{\lambda} - \mu_Y + \frac{\eta(y)}{f_Y(y)}, \quad (22)$$

where $h'(\cdot)$ denotes the derivative of $h(\cdot)$. Now observe that since $h(\cdot)$ is convex, and $\gamma \geq 0$, the function

$$H_\gamma(x) \triangleq x + \gamma h'(x) \quad (23)$$

is monotonically increasing. Hence, by (22) we have

$$\omega(y) = H_\gamma^{-1} \left(\theta - \frac{1}{\lambda} - \mu_Y + \frac{\eta(y)}{f_Y(y)} \right) - y. \quad (24)$$

By complementary slackness [15], we get $\eta(y) = 0$ if $\omega(y) > 0$, in which case $H_\gamma^{-1}(\theta - \frac{1}{\lambda} - \mu_Y) > y$. On the other hand, if $H_\gamma^{-1}(\theta - \frac{1}{\lambda} - \mu_Y) < y$, then we must have $\eta(y) > 0$ so as to increase the argument inside H_γ^{-1} and make $\omega(y)$ non-negative. This means, again by complementary slackness, that $\omega(y) = 0$. Combining the arguments, we finally have the optimal waiting policy that solves problem (19) given by

$$\omega^*(y) = \left[H_{\gamma^*}^{-1} \left(\theta - \frac{1}{\lambda} - \mu_Y \right) - y \right]^+ \quad (25)$$

where $[\cdot]^+ \triangleq \max(\cdot, 0)$ and γ^* denotes the optimal Lagrange multiplier associated with the credibility constraint.

The result above shows that the optimal waiting policy has a threshold structure; as long as the starting AoI of an epoch is below a certain threshold, given by $H_{\gamma^*}^{-1}(\theta - \frac{1}{\lambda} - \mu_Y)$, the sensor should continue in sleeping mode until the AoI surpasses that threshold, and then wake up. The threshold, however, remains partially unknown unless we can evaluate γ^* . We do so indirectly as follows. First, let us assume that

$\gamma^* = 0$. In this case, the threshold is simply given by

$$H_0^{-1} \left(\theta - \frac{1}{\lambda} - \mu_Y \right) = \theta - \frac{1}{\lambda} - \mu_Y. \quad (26)$$

We now use the above to check if the credibility constraint is satisfied. If it is not, then it must be that $\gamma^* > 0$, which means by complementary slackness that the credibility constraint is satisfied with equality. In this case, all we need to do is to find some threshold value ξ that solves

$$\mathbb{E} \left[h \left(Y_{i-1} + [\xi - Y_{i-1}]^+ \right) \right] = \tau, \quad (27)$$

which can be evaluated by, e.g., a bisection search since $h(\cdot)$ is monotonically decreasing.

The approach above provides the optimal solution of problem (19), i.e., it evaluates $p(\theta)$. The final step to link all this back to problem (18) is to find the optimal value θ^* that solves $p(\theta^*) = 0$. We have now proven the following theorem that summarizes the theoretical results in this section:

Theorem 1 *The optimal solution of problem (18) is given by a threshold-waiting policy $\omega^*(\cdot) = [\xi^* - \cdot]^+$. The threshold ξ^* is given by $\theta^* - \frac{1}{\lambda} - \mu_Y$, provided that the credibility constraint is satisfied. Otherwise, it is given by the solution of (27). The value of θ^* is such that $p(\theta^*) = 0$ in (19).*

IV. THE MULTI-PROCESS SETTING

We now use the results developed for the single process setting to present solutions for problem (9) in the case of $K \geq 2$ processes. We first note that finding the jointly optimal scheduling and waiting policies is highly nontrivial. One reason behind this is that the functions governing the time-stamp errors, $h_k(\cdot)$'s, can vary from one server/process to another. For example, let us consider the exponentially decaying model in (4) for $K = 2$. If $\alpha_1 > \alpha_2$, then server 1 recovers faster than server 2. Hence, it could be optimal to schedule process 1 more often than process 2 so as to allow a sufficient time for server 2 to recover and reduce errors. Thus, the often studied round-robin schedule (or maximum-age-first) in the AoI literature [17] may not be optimal in our setting. This, in addition to the fact that we need to evaluate optimal waiting times renders the problem challenging.

To alleviate this hurdle, in this preliminary work on this problem we aim at developing policies that optimize the scheduling policy or the waiting policy *individually*, as opposed to jointly, and compare their performances against each other. We focus on two specific kinds of policies that we discuss next. We note that the detailed expressions of the resulting AoI's and time stamp errors from such policies are omitted due to space limits, and are demonstrated in the experimental results in Section V.

A. Round-Robin Scheduling with Threshold-Waiting

The first policy that we discuss considers a round-robin (RR) schedule, in which process 1 is scheduled for sampling, followed 2, all the way to K , and then the schedule repeats. We denote this schedule by π_{RR} .

As for the waiting policy, and given the results of the single process setting, we combine RR scheduling with a threshold-waiting policy, which is illustrated as follows. Instead of waiting prior to each process sampling, we only wait once before scheduling process 1. This is then followed by the scheduled RR transmissions. We note that such approach has been shown optimal in the relatively similar setting of [17]. Hence, let us focus on process 1's epoch. The waiting time at the beginning of epoch i is now given by the following function of epoch $i - 1$'s sum service times:

$$\omega \left(\sum_{k=1}^K Y_{k,i-1} \right) = \left[\xi - \sum_{k=1}^K Y_{k,i-1} \right]^+, \quad (28)$$

for some threshold ξ to be optimized. For simplicity of presentation, we refer to the above expression by ω_i^{RR} .

Now observe that for the k th process, we get from (6) that

$$L_{k,i} = \sum_{s=k+1}^K X_{s,i-1} + Y_{s,i-1} + \omega_i^{RR} + \sum_{s=1}^k X_{s,i} + Y_{s,i}. \quad (29)$$

The above expression, together with the definition of ω_i^{RR} in (28), shows that the location of the waiting time could possibly lead to different distributions of epochs across different processes. However, we note that the *sums* of their corresponding AoI's and time stamp errors would still be the same, denoted by $\overline{\text{AoI}}_k(\pi_{RR})$ and $\overline{\text{e}}_k(\pi_{RR})$, respectively. We omit such details due to space limits. Based on this, we focus on the case in which all β_k 's are equal in this work. This implies that the relationship between AoI and the time-stamp error remains consistent across all processes, leading to a uniform stationary behavior in the system's performance.³

Using the above expression, we simplify equations (7) and (8) and specialize them to the case of RR scheduling with threshold-waiting. The resulting expressions are used for specific service time distribution to get closed-form expressions for $\overline{\text{AoI}}_k(\pi_{RR})$ and $\overline{\text{e}}_k(\pi_{RR})$ in terms of the threshold ξ , which can then be found by, e.g., line search algorithms.

B. Asymmetric Scheduling with Zero-Waiting

The second policy under consideration is an asymmetric scheduling (AS) policy, indicated by π_{AS} . In here, process 1 is scheduled for m_1 sampling and transmission trials, followed by process 2 for m_2 trials, and so on until process K completes its m_k trials, after which the schedule repeats. In this asymmetric schedule, different from π_{RR} , we do not consider waiting prior to transmission.

Using π_{AS} , an epoch for any process k includes the same number of (possibly different) transmissions from every other process. Since the waiting time is 0 and all random variables are i.i.d., it follows that the system is stationary and all epochs' distributions are the same. We denote the corresponding AoI and time stamp error for process k by $\overline{\text{AoI}}_k(\pi_{AS})$ and $\overline{\text{e}}_k(\pi_{AS})$, respectively.

³The general case in which β_k 's are not equal can be solved by optimizing the *location* of the waiting time, and is to be analyzed in future work.

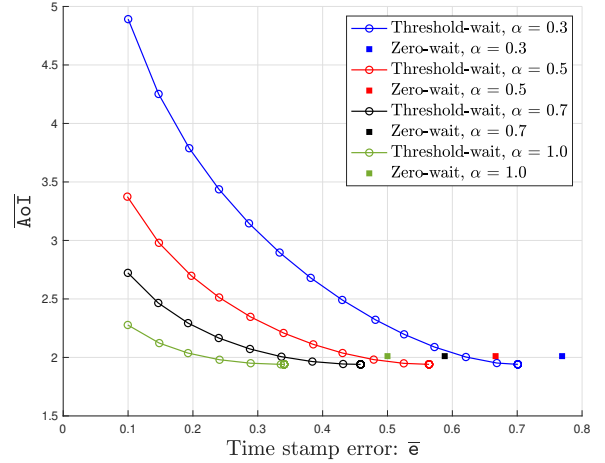


Fig. 3: Single process AoI vs. time-stamp error for different recovery rate α values.

Let us denote by $X_{k,i}^{(j)}$ and $Y_{k,i}^{(j)}$ the j th inter-arrival and service times of the k th process in the i th epoch, respectively, where the epoch index is counted with respect to process 1 without loss of generality. Therefore, we have

$$L_{k,i} = \sum_{s=k+1}^K \sum_{j=1}^{m_s} X_{s,i-1}^{(j)} + Y_{s,i-1}^{(j)} + \sum_{s=1}^k \sum_{j=1}^{m_s} X_{s,i}^{(j)} + Y_{s,i}^{(j)}. \quad (30)$$

Based on the above expression, we specialize the equations in (7) and (8) and derive the AoI and time stamp error expressions for AS scheduling with zero waiting. Given a service time distribution, one can then find the optimal selection of the number of trials m_k for process k .

V. NUMERICAL RESULTS

In this section, we present some numerical results to further illustrate the theoretical analysis of this paper. We focus on showing the AoI vs. time stamp error trade-off under different system settings. From the optimization problem in (9), such a trade-off can be characterized by varying the values of β_k 's. In our simulations, and in agreement with our theoretical results, we focus on the case in which $\beta_k = \beta$, $\forall k$. The service time distribution is $\sim \exp(1/\mu_Y)$.

We first present results for the single process setting. In Fig. 3, we vary the value of $\beta \in [0, 1]$ to show how the AoI behaves with respect to time stamp error. We set $\lambda = 9$ and $\mu_Y = 1$. Clearly, the higher the value of β the better the AoI and the worse the error, and vice versa. Moreover, as the recovery rate α increases, the trade-off behaves better: for a relatively higher value of α , one can achieve lower errors for the same AoI values. As a baseline, we show the single AoI-error pair achieved by the zero-wait policy. It is clear from the figure that the optimal threshold-wait policy outperforms zero-wait in terms of both AoI and error for relatively higher values of β .

Next, we compare the behavior of π_{RR} and π_{AS} for $K = 2$ processes. We set $\lambda_1 = \lambda_2 = 6$, $\alpha_2 = 50$, and $\mu = 1.5$,

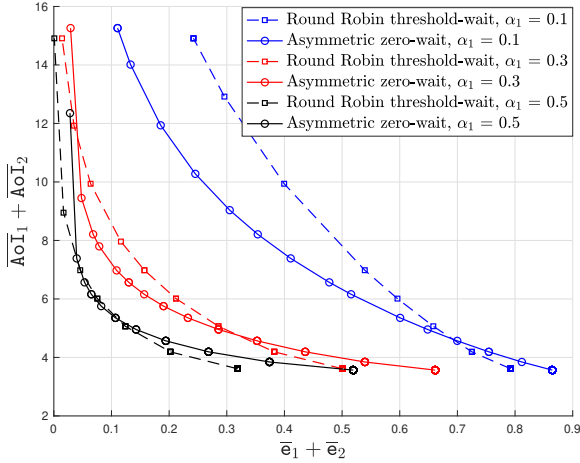


Fig. 4: Two processes sum AoI vs. sum time stamp error for different process 1 recovery rate α_1 values.

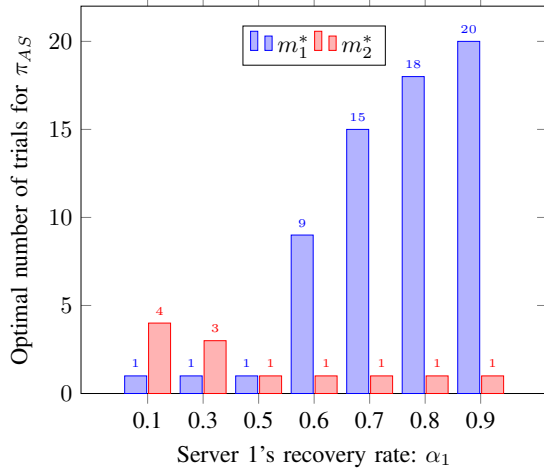


Fig. 5: Optimal AS policy behavior: (m_1^*, m_2^*) vs. α_1 .

and plot the sum AoI vs. sum time stamp error (by varying β) in Fig. 4. The results generally show that the behavior of π_{RR} relative to π_{AS} depends on the value of α_1 and β . For instance, for smaller α_1 and smaller β , π_{AS} is favored upon π_{RR} . While for larger α_1 and larger β the situation is reversed. For intermediate values no specific policy dominates the other. This shows that the choice of the scheduling and waiting policy for this problem is highly dependent on the system dynamics, especially the servers' recovery rates.

Finally, the column chart in Fig. 5 illustrates the optimal number of trials for π_{AS} , m_1^* and m_2^* , vs. different values of server 1's recovery rate α_1 . Here, we set $\alpha_2 = 0.5$, $\lambda_1 = \lambda_2 = 90$, $\mu = 50$, and $\beta = 0.5$. As α_1 increases, server 1 recovers relatively faster than server 2, and therefore m_1^* increases while m_2^* decreases as seen in the figure. This trend underscores the importance of tailoring transmission strategies to specific server recovery rates to optimize timeliness and credibility.

VI. CONCLUSION

The impact of time stamp errors on the credibility of AoI in status updating systems has been investigated. Samples from multiple processes are acquired by sensors and then processed by servers to be sent through a shared channel. Through modeling time stamp error rates as a function of the servers' busy times, a trade-off has been introduced: allowing servers more sleeping time, and hence more time to recover, decreases time stamp errors, but increases AoI. An optimization problem has been formulated to characterize the optimal timeliness-credibility trade-off by designing scheduling and server sleep-wake policies. Solutions have been presented first for the single process setting, in which the optimal sleep-wake policy has been shown to have a threshold structure. For the multi-process setting, round-robin (symmetric) and asymmetric scheduling have been studied. Our results show that server recovery rates can highly affect the timeliness-credibility trade-off curves, and that scheduling policies should be chosen based on the system parameters, including processes' sampling rates, channel service rate and relative recovery rates among the servers.

REFERENCES

- [1] R. D. Yates, Y. Sun, R. D. Brown, S. K. Kaul, E. Modiano, and S. Ulukus. Age of information: An introduction and survey. *IEEE J. Sel. Areas Commun.*, 39(5):1183–1210, May 2021.
- [2] S. Kaul, R. D. Yates, and M. Gruteser. Real-time status: How often should one update? In *Proc. IEEE INFOCOM*, March 2012.
- [3] A. Arafa, J. Yang, S. Ulukus, and H. V. Poor. Timely status updating over erasure channels using an energy harvesting sensor: Single and multiple sources. *IEEE Trans. Green Commun. Netw.*, 6(1):6–19, 2022.
- [4] H. Yang, A. Arafa, T. Q. S. Quek, and H. V. Poor. Age-based scheduling policy for federated learning in mobile edge networks. In *Proc. IEEE ICASSP*, April 2020.
- [5] R. D. Yates. The age of gossip in networks. In *Proc. IEEE ISIT*, July 2021.
- [6] J. Heand, Q. Ma, M. Zhang, and J. Huang. Optimal fresh data sampling and trading. In *Proc. IEEE WiOpt*, October 2021.
- [7] H. Sac, T. Bacinoglu, E. Uysal-Biyikoglu, and G. Durisi. Age-optimal channel coding blocklength for an M/G/1 queue with HARQ. In *Proc. IEEE SPAWC*, June 2018.
- [8] M. A. Abd-Elmagid, N. Pappas, and H. S. Dhillon. On the role of age of information in the internet of things. *IEEE Commun. Mag.*, 57(12):72–77, December 2019.
- [9] F. Zhao, N. Pappas, C. Ma, X. Sun, T. Q. S. Quek, and H. H. Yang. Optimizing information freshness in mobile networks with age-threshold aloha. In *Proc. IEEE ISIT*, July 2024.
- [10] X. He, C. You, and T. Q. S. Quek. Age-based scheduling for mobile edge computing: A deep reinforcement learning approach. *IEEE Trans. Mobile Comput.*, 23(10):9881–9897, February 2024.
- [11] A. Arafa and K. Banawan. Private status updating with erasures: A case for retransmission without resampling. In *Proc. IEEE ICC*, May 2023.
- [12] W. Minnaard. Timestamping NFTS. Master's thesis, University of Amsterdam, July 2014.
- [13] P. Kaswan and S. Ulukus. Timestamping vulnerability of age-sensitive gossip networks. *IEEE Trans. Commun.*, 72(7):4193–4205, July 2024.
- [14] D. Ward. *Remote Estimation Over Use-Dependant Channels*. PhD thesis, University of Maryland, 2017.
- [15] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge university press, 2004.
- [16] W. Dinkelbach. "On nonlinear fractional programming". *Manag. Sci.*, 13(7):492–498, 1967.
- [17] K. Banawan, A. Arafa, and K. G. Seddik. Timely multi-process estimation over erasure channels with and without feedback: Signal-independent policies. *IEEE J. Sel. Areas Inf. Theory*, 4:607–623, November 2023.