Minimum Achievable Peak Age of Information Under Service Preemptions and Request Delay

Jaya Prakash Champati, *Member, IEEE,* Ramana R. Avula, *Student Member, IEEE,*
Tobias J. Oechtering, *Senior Member, IEEE* and James Gross, *Senior Member, IEEE*

***Abstract*—There is a growing interest in analysing freshness of data in networked systems. Age of Information (AoI) has emerged as a relevant metric to quantify this freshness at a receiver, and minimizing this metric for different system models has received significant research attention. However, a fundamental question remains: what is the minimum achievable AoI in any single-server-single-source queuing system for a given service-time distribution? We address this question for the average peak AoI (PAoI) statistic by considering generate-at-will source model, service preemptions, and request delays. Our main result is on the characterization of the minimum achievable average PAoI, and we show that it is achieved by a fixed-threshold policy among the set of all causal policies. We use the characterization to provide necessary and sufficient condition for preemptions to be beneficial for a given service-time distribution. Our numerical results, obtained using well-known distributions, demonstrate that the heavier the tail of a distribution the higher the performance gains of using preemptions.**

I. Introduction

The notion of information/data freshness is pervasive in networked systems. Understanding and quantifying this notion is essential for an efficient design of the networked systems to support the ever increasing demand for real-time status updates by emerging applications in Cyber-Physical Systems, Internet-of-Things, and information systems. Age of Information (AoI), proposed in [1], has emerged as a relevant performance metric for quantifying the freshness of status updates from the perspective of the receiver. It is defined as the time elapsed since the generation of freshest update available at the receiver. Unlike system delay, AoI accounts for the frequency of generation of updates by a source, since it linearly increases with time until an update with latest generation time is received. Whenever such an update is received AoI resets to the system delay of that update and thus indicating its age.

Given the above properties and its relevance to networked systems, minimizing AoI for a given service-time and/or inter-arrival-time distributions has received significant attention in the literature, e.g., see [2]–[7]. However, in the system models considered in these works, the update arrival instants can only be partially controlled, for example, by tuning the arrival rate. In contrast, the authors in [8], [9] considered the generate-at-will source model, where the source can generate an update at any time instant using a *scheduling policy*, thus completely controlling the time instants at which arrivals occur. Under this model no queueing is required, because by the definition of AoI, at any time instant, sending an old update from a queue would be suboptimal to sending a freshly generated update. Furthermore, for a given service-time distribution, the minimum AoI statistics achieved under this model will be lower than that of the system models where the arrivals instants are only partially controlled. For a single-source-single-server system, the authors in [8] solved for an optimal scheduling policy that minimizes the average AoI, while the authors in [9] solved the problem for any non-decreasing function of AoI.

Motivated by the fact that, allowing service preemptions on top of the generate-at-will source model could further reduce AoI, we ask the fundamental question *what is the minimum achievable AoI in a single-source-single-server queuing system for any given service-time distribution?* Recently, the authors in [10] studied this problem for average AoI and numerically computed optimal policies for exponential and shifted-exponential service-time distributions. However, for general service-time distributions this is an open problem. In contrast, in this work, we solve minimum achievability for average peak-AoI (PAoI). This AoI statistic was first studied in [11] and has received considerable attention in recent works, e.g., see [5], [12], [13]. To solve for minimum achievable average PAoI for general service-time distributions (possibly with infinite mean), we consider the information retrieval system shown in Figure 1, where the monitor strives to obtain latest information from the information source by scheduling requests for status updates. The request channel only allows one request at a time and incurs a constant delay d. When the source receives a request, it immediately generates an update and sends it to the preemptive server with independent and identically distributed (i.i.d.) service times[1].

If preemptions are not allowed, then the *no-threshold* policy, which sends a request immediately after the monitor receives an update, minimizes the average PAoI. When preemptions are allowed, two main challenges arise while solving for an optimal scheduling policy. First, computing the average PAoI under a preemptive scheduling policy is non-trivial due to its dependencies on the sequence of preemptions. Also, the request delay which induces idle time at the server further

Jaya P. Champati is with IMDEA Networks Institute, Madird, Spain. Ramana R. Avula, Tobias J. Oechtering, and James Gross are with the School of Electrical Engineering and Computer Science, KTH Royal Institute of Technology, Stockholm, Sweden. E-mail: jaya.champati@imdea.org, {avula, oech, jamesgr}@kth.se. The work was supported in part by the Strategic Research Agenda Program, Information and Communication Technology – The Next Generation (SRA ICT – TNG), through the Swedish Government, and in part by the Swedish Research Council (VR) under grant 2016-04404.

[1]A preliminary version of this work is published in IEEE INFOCOM 2020 [14].

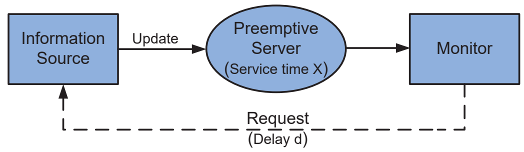

Fig. 1: A model for information retrieval with independently evolving source.

complicates the analysis. Second, the monitor's decision on when to send a new request to the source, possibly preempting an update under service, clearly depends on the service-time distribution and could potentially depend on the past decisions. Thus, minimizing the average PAoI under preemptions results in an infinite-horizon average cost Markov Decision Problem (MDP), where the state space and the action space are continuous. In general, for such a problem, it is hard to prove the existence of an optimal stationary deterministic policy among all randomized causal policies, which use the entire history of available information [15].

Our key result is that, a *fixed-threshold policy*, that chooses a fixed duration between requests, minimizes the average PAoI among all causal policies[2]. We prove this result in two steps. First, we formulate an MDP with appropriate cost functions and show that the policy for choosing the sequence of thresholds between any two AoI peaks is independent of the initial state and is also stationary. Second, we define costs for each decision within the two AoI peaks and show that the sequence of decisions converge to a stationary policy and that a fixed-threshold policy achieves the minimum cost. We then characterize the minimum achievable average PAoI in any single-source-single-server queuing system. Further, we present a necessary and sufficient condition for preemptions to be beneficial for a given service-time distribution. In our numerical analysis, three service-time distributions, namely, Erlang, Pareto, and log-normal are used to study how minimum average PAoI varies with different thresholds. Our results demonstrate that for smaller d values, the optimal threshold decreases as d increases for the service-time distributions considered. Also, we compare the performance of no-threshold policy and median-threshold policy, which chooses median as the fixed threshold, against the optimal fixed-threshold policy. An interesting observation is that, the heavier the tail of a distribution the higher the performance gains of using preemptions.

The rest of the paper is organized as follows. The related work is presented in Section II. In Section III, we formulate the average PAoI minimization problem. In Section IV, we present preliminary results that are used in Section V to obtain the optimal fixed-threshold policy. In Section VI, we discuss the conditions under which preemptions are beneficial. The numerical results are presented in Section VI, and finally, we conclude in Section VIII.

[2]In the seminal work [16] (in the field of computer science), the authors showed that a fixed threshold is optimal for restarting a randomized algorithm in order to minimize its expected run time, and it inspired our work. Nevertheless, in comparison to the problem studied in [16], minimizing expected PAoI is harder as the consecutive AoI peaks are not independent even under a stationary policy. Furthermore, we consider general service-time distributions and the optimality of a fixed-threshold policy is shown for the set of randomized causal policies.

II. Related Work

Research works in AoI literature can be broadly divided into two categories: 1) analysis of AoI, and 2) minimization of AoI. Analysis of AoI concerns with computation of AoI statistics for different system settings, e.g., see [11], [17]–[23], and [24] for a comprehensive survey. In this section, we focus on summarizing related works on the minimization of AoI under different system settings by further dividing them based on non-prememptive and preemptive service considerations.

A. Non-preemptive Service

In contrast to latency, AoI has interesting property that it increases at both low and high sampling rate for queueing systems using First-Come-First-Serve (FCFS) policy [2]. This property led to initial works focusing on quantifying and minimizing the average AoI for the M/M/1, M/D/1 and D/M/1 queues in [2], and for an M/M/1 queue with multi-sources in [3]. Minimizing average PAoI was considered in [5] for a multi-class M/G/1 system, and AoI violation probability minimization for the D/G/1 system was studied in [25], under FCFS policy. In contrast, the authors in [17], [19] studied Last-Come-First-Serve (LCFS) policy as it reduces AoI compared to FCFS policy. One may further reduce AoI compared to LCFS policy by considering packet discarding. Intuitively, even if an infinite capacity queue is available, when the server is busy, discarding the arriving packets except for the most recent packet would result in a lower AoI when compared with storing and transmitting any older packets. This motivated research efforts toward studying systems with no queue or a single capacity queue storing the latest packet [11], [21], [26], [27].

Some research works also considered multi-hop settings. The authors in [28], [29] studied average AoI and average PAoI minimization in a multi-hop wireless network with interference constraints and with packet flows between multiple source-destination pairs assuming that transmission time of a packet equals a unit time slot. Optimizing AoI was also extensively studied for the systems with energy-harvesting source, e.g., see [4], [30]. In contrast to above works, the generate-at-will source model was studied in [8], [9] where generation of a status update can be completely controlled. While the authors in [8] solved for optimal-waiting times between generation times to minimize the average AoI, the authors in [9] solved the problem for any non-decreasing function of AoI. When the request delay $d = 0$, the system we study is equivalent to the generate-at-will source model. Furthermore, in contrast to [8], [9], we consider service preemptions and address the minimum achievabile PAoI problem. Next, we summarize works that consider service preemptions.

B. Preemptive Service

Most of the works that considered service preemptions focused on analysing the average AoI and average PAoI for different queueing systems, e.g., see [17]–[20], [26], [31]–[33]. Minimizing average AoI for a single-server-single-source system under service preemptions was studied in [22], [34]–[36]. The authors in [34] studied the problem of whether to preempt or not preempt the current update in service in an M/GI/1/1 system, while the authors in [22] showed that the deterministic arrivals are optimal for a given arrival rate for the G/M/1/1 system. In [35], optimal policies for Bernoulli arrivals were studied, and optimal blocklength for packets was computed for Poisson arrivals in [36]. AoI minimization under service preemptions was also considered in multi-source and/or muti-server systems, e.g., see [7], [37]. In contrast to our problem, among other differences, the generate-at-will model was not considered in the above works.

The work by the authors in [10] is contemporary to ours. The authors studied a system equivalent to the generate-at-will source model [8], [9], and considered the problem of minimizing the average AoI in the system under service preemptions. Considering a fixed-threshold policy for doing preemptions, the authors first solve for an optimal *waiting time*[3]. Stating that it is hard to obtain a closed-form expression for the average AoI in terms of the fixed threshold and its corresponding optimal waiting time, the authors compute, numerically, the optimal fixed threshold for two service-time distributions, namely, exponential and shifted exponential. However, it is not shown if the proposed method results in an optimal policy for general service-time distributions. In this work, we consider the average PAoI minimization problem and provide a fixed-threshold policy that is optimal in the set of randomized causal policies[4]. Furthermore, we characterize the minimum achievable average PAoI.

III. System Model and Problem Statement

We study an information retrieval system shown in Figure 1, where a monitor (e.g., a mobile application) strives to obtain latest information (e.g., newsfeeds) from a source which evolves independently. When the source receives a request from the monitor, it instantaneously generates an information update (or simply update) and sends it to the preemptive server. The request is sent on a dedicated channel that serves one request at a time and induces a constant delay d per request. This delay d may model the transmission delay at the monitor. On the other hand, an update incurs a random service time, denoted by X, at the server before it reaches the monitor. We assume that the service times across the updates are i.i.d. Further, we consider that a new update always preempts an update under service. Let $F_X(\cdot)$, $f_X(\cdot)$ and $\mathbb{E}[X]$ denote the cumulative distribution function, probability density function and the mean of X, respectively. We use $x_{\min} \geq 0$ to denote the minimum value in the support of X. We note that, for $d = 0$, our model is equivalent to the generate-at-will source model [8], [9], where the monitor indicates to the source if an update was received (for instance by an ACK), and then the source decides (*push model*) when to generate the next update. A subtle difference is that, our model is a *pull model* as the monitor decides when to send the next update request and the source merely generates an update upon receiving the request. The pull model aptly models several applications; for example, in web browsing, the monitor represents a web browser while the source represents a web server.

[3]The idle time of the server after an update is received. Idling the server does not reduce the average PAoI but may reduce the average AoI.

[4]Whether a fixed-threshold policy is optimal for average AoI minimization remains an open problem.

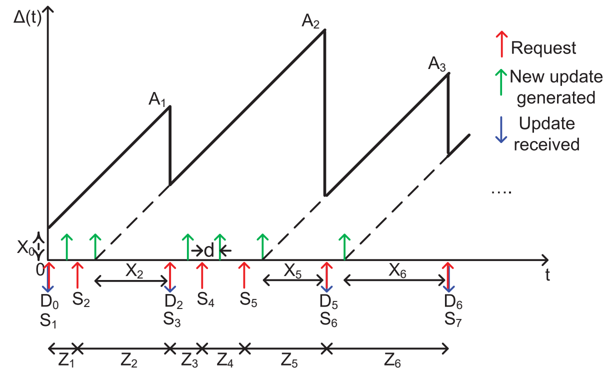

Fig. 2: A sample path of AoI under service preemptions.

At any time, the monitor aims to have the freshest update. Note that this depends on the time instants at which monitor requests new information. A scheduling policy for information requests specifies these time instants. To be precise, let n denote the index of a request and also its corresponding update, then a scheduling policy $\boldsymbol{s} \triangleq \{S_n, n \geq 1\}$, where $S_n \in \mathbb{R}_{\geq 0}$ denotes the generation time of request n. Using the convention that request 1 is sent at time zero, the waiting time between requests n and $n+1$, denoted by Z_n, is given by $Z_n = S_{n+1} - S_n$. As a consequence of constant delay d incurred for each request and the minimum service time is $x_{\min}$, we have $Z_n \geq \max\{x_{\min}, d\}$ for all n. Note that the scheduling policy can be equivalently written as $\boldsymbol{s} = \{Z_n, n \geq 1\}$. When the monitor sends request n at time S_n, the corresponding update n is generated by the source at time $S_n + d$. The monitor waits for duration Z_n and sends the next request at $S_{n+1} = S_n + Z_n$, for which the update $n+1$ is generated at $S_{n+1} + d$. This update preempts update n if the service time X_n exceeds Z_n.

Let D_n denote the time at which information update n is received at the monitor. We assign $D_n = \infty$, if the update n is dropped due to preemption. We have

$$D_n = \begin{cases} S_n + X_n + d & \text{if update } n \text{ is received} \\ \infty & \text{otherwise} \end{cases}$$

In this system, AoI at the monitor at time t, denoted by $\Delta(t)$, is given by

$$\Delta(t) = t - \max_{n \in \mathbb{N}} \{S_n + d : D_n \leq t\}. \tag{1}$$

Note that $\Delta(t)$ increases linearly with t and drops instantaneously when an update is received. In Figure 2, we present a sample path of AoI under service preemptions. Here, we have used the convention that, a packet is received at time zero and the initial AoI $\Delta(0) = X_0$. Let k denote the kth AoI peak and $A_k(\boldsymbol{s})$ denote the corresponding PAoI value. Further,

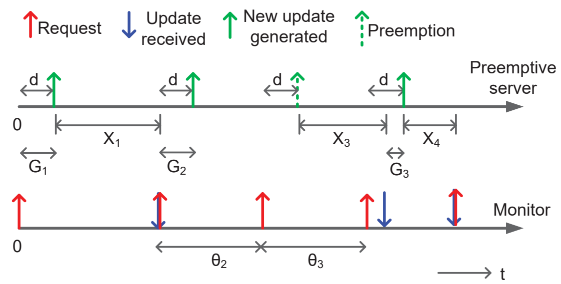

Fig. 3: Illustration of idle times G_k under a work-conserving policy.

let n_k denote the index of the update received just after the kth AoI peak. Note that between updates n_k and n_{k+1} there could be multiple updates that are preempted. We now have $A_k(\boldsymbol{s}) = \Delta(D_{n_k}^-)$, where $D_{n_k}^-$ is the time just before update n_k is received under $\boldsymbol{s}$.

In the following, we describe the policies of interest.

- *Work-conserving policy:* $Z_n = \min(\theta_n, X_n + d)$, for all n, where θ_n is a threshold for preemption and takes values from $[\max\{x_{\text{min}}, d\}, \infty) \cup \{\infty\}$. Under this policy, a request is sent if an update is received or the threshold θ_n is elapsed, which ever happens first. There might be scenarios where an update (for previous request) is received after a new request is sent but before the new request reaches the source. In this case, the update is accepted by the monitor and it does not generate another request for receiving this update; the source, as always, generates an update when the new request reaches it. We note that, under this policy, the server will be always busy when $d = 0$.
- *Threshold policy:* $Z_n = \min(\theta_n, X_n + d)$, for all n, where $\theta_n \in [\theta_{\text{min}}, \theta_{\text{max}}]$ is a threshold for preemption, $\theta_{\text{min}} > x_{\text{min}}$, $\theta_{\text{min}} \geq d$, and $\theta_{\text{max}} < \infty$. A threshold policy is a work-conserving policy with bounded thresholds.
- *Fixed-threshold policy:* $Z_n = \min(\theta, X_n + d)$, for all n, for some $\theta \in [\theta_{\text{min}}, \theta_{\text{max}}]$. We use $\boldsymbol{s}_\theta$ to denote this policy.
- *Min-threshold policy:* $Z_n = \max\{x_{\text{min}}, d\}$, for all n. We use $\underline{\boldsymbol{s}}$ to denote this policy.
- *No-threshold policy:* $Z_n = X_n + d$, for all n[5]. We use $\boldsymbol{s}_\infty$ to denote this policy. Under $\boldsymbol{s}_\infty$ a request is sent immediately after an update is received and preemptions are not allowed[6]. We note that $\boldsymbol{s}_\infty$ is the only non-preemptive work-conserving policy, where $\theta_n = \infty$, for all n.

Since a request incurs delay d before reaching the source, the preemptive server may stay idle during this period. Under work-conserving policies, the idle time of the server before generation of a new packet can be at most d as the monitor sends a new request immediately after it receives an update or the threshold expires. Let $G_k(\boldsymbol{s})$ denote the idle time of the server between $D_{n_k}(\boldsymbol{s})$ and the time at which the next update is generated. The timing diagram in Figure 3 complements Figure 2 by illustrating how the request delay d induces the idle times $G_k(\boldsymbol{s})$ at the preemptive server. It can be observed that, under work-conserving policy $\boldsymbol{s}$, we have $0 \leq G_k(\boldsymbol{s}) \leq d$.

Under a given policy $\boldsymbol{s}$, the average PAoI is defined as

$$\zeta(\boldsymbol{s}) \triangleq \lim_{K \to \infty} \frac{1}{K} \mathbb{E}_{\boldsymbol{s}} \left[\sum_{k=1}^{K} A_k(\boldsymbol{s}) \right], \tag{2}$$

where the expectation above is taken with respect to a probability distribution determined by $\boldsymbol{s}$ and the distribution of X. Let $\mathcal{S}$ denote the set of all admissible causal policies for which the limit in (2) exists. For any given service-time distribution $F_X(\cdot)$ and request delay d, we are interested in solving the average PAoI minimization problem $\mathcal{P}$:

$$\begin{aligned} &\underset{\boldsymbol{s} \in \mathcal{S}}{\text{minimize}} \quad \zeta(\boldsymbol{s}) \\ &\text{subject to} \quad Z_n \geq \max\{x_{\text{min}}, d\}, \ \forall n \geq 1. \end{aligned}$$

Let $\boldsymbol{s}^*$ denote an optimal policy, and ζ^* denote the minimum average PAoI achieved under $\boldsymbol{s}^*$. We define a threshold θ^* as follows:

$$\theta^* = \underset{\theta \in [\max\{x_{\text{min}}, d\}, \infty) \cup \{\infty\}}{\arg\min} \zeta(\boldsymbol{s}_\theta). \tag{3}$$

Our main result is that the fixed-threshold policy $\boldsymbol{s}_{\theta^*}$ achieves the minimum average PAoI.

In the sequel, we use $\eta_n := \theta_n - d$, and $x^+ = \max\{0, x\}$. The list of symbols used are summarized in Table I.

TABLE I: NOMENCLATURE

d	Request delay
n	Index of a request and its corresponding update
k	Index of AoI peak
X	Service time
x_{min}	The minimum value in the support of service time X
S_n	Generation time of request n
Z_n	The waiting time between requests n and $n+1$
D_n	The time at which update n is received at the monitor
$\Delta(t)$	Age of information at time t
A_k	k-th PAoI value
n_k	The index of the update received just after the k-th PAoI
$\check{X}_k$	The service time of update n_k
G_k	The idle time of the server between D_{n_k} and S_{n_k+1}
θ_n	Threshold for preemption of update n
$\theta_{\text{min}}, \theta_{\text{max}}$	The minimum and maximum values of threshold θ_n
η_n	$\theta_n - d$
Y_{k+1}	The duration between the time instances at which update n_k and n_{k+1} are received
$F_X(x)$	Cumulative distribution function of the service time X
$f_X(x)$	Probability density function of the service time X
$\boldsymbol{s}$	Scheduling policy
$\boldsymbol{s}_\theta$	Fixed-threshold policy
$\underline{\boldsymbol{s}}$	Min-threshold policy
$\boldsymbol{s}_\infty$	No-threshold policy
$\zeta(\boldsymbol{s})$	Average PAoI under a given policy $\boldsymbol{s}$
$\boldsymbol{s}^*$	Optimal policy that achieves minimum average PAoI
ζ^*	The minimum average PAoI
θ^*	Threshold of an optimal fixed-threshold policy

[5]For $d = 0$, no-threshold policy is equivalent to the zero-wait policy studied in [9].

[6]In scenarios where updates are lost in a network, no-threshold policy and in general any non-preemptive policy fails, and it is essential to consider preemptive policies. Service-time distributions corresponding to such scenarios are improper, i.e., $\lim_{x \to \infty} F(x) < 1$.

IV. Threshold Policies and Auxiliary Results

In this section, we define different classes of threshold policies and provide some important auxiliary results which will be used in the later parts of the paper. Let I_n denote the causal information available at nth request. It includes all the observed PAoI values, service times, and thresholds used prior to the nth request.

Definition 1. *A randomized-threshold causal policy specifies a probability distribution for choosing $\theta_n \in [\theta_{min}, \theta_{max}]$ using I_n.*

Let $\mathcal{S}_{\mathrm{T}}$ denote the set of all randomized-threshold causal policies. The constraint $\theta_n \in [\theta_{\min}, \theta_{\max}]$ in threshold policies is an artefact introduced to bound the MDP costs and facilitate the proof of convergence of the optimal policy to a stationary fixed-threshold policy. Note that, considering $x_{\min} < \theta_{\min}$[7] and $\theta_{\max} < \infty$ excludes min-threshold policy when $d < x_{\min}$, and no-threshold policy, respectively, from $\mathcal{S}_{\mathrm{T}}$. Nevertheless, for a given problem, choosing $\theta_{\min}$ arbitrarily close to $x_{\min}$ and $\theta_{\max}$ sufficiently large, the imposed constraints result in only a mild restriction of $\mathcal{S}_{\mathrm{T}}$. If $d > x_{\min}$, $\theta_{\min}$ takes value d on the left extreme. Figure 4 visualizes $\mathcal{S}_{\mathrm{T}}$ and other work-conserving policies using a Venn diagram.

In our proof for optimality of a fixed-threshold policy, we first show that an optimal policy belongs the set randomized-threshold policies which choose the same set of thresholds between two successfully received updates. We define this set of policies below.

Definition 2. *A repetitive randomized-threshold policy is a randomized-threshold causal policy under which the joint distributions for choosing the set of thresholds between any two AoI peaks are identical.*

Let $\mathcal{S}_{\mathrm{TR}}$ denote the set of all repetitive randomized-threshold policies, and $\mathcal{S}_{\theta}$ denote the set of all fixed-threshold policies. From the above definitions, we have $\mathcal{S}_{\theta} \subset \mathcal{S}_{\mathrm{TR}} \subset \mathcal{S}_{\mathrm{T}} \subset \mathcal{S}$.

From Figure 2, it is easy to infer that under any policy $\boldsymbol{s}$, we have, for all k,

$$\begin{aligned} A_{k+1}(\boldsymbol{s}) &= D_{n_{k+1}} - (S_{n_k} + d) \\ &= \underbrace{D_{n_{k+1}} - D_{n_k}}_{\triangleq Y_{k+1}(\boldsymbol{s})} + \underbrace{D_{n_k} - (S_{n_k} + d)}_{\triangleq \check{X}_k(\boldsymbol{s})}. \end{aligned} \tag{4}$$

Note that $\check{X}_k(\boldsymbol{s})$ is equal to X_{n_k}, the service time of update n_k. However, under preemptive policies $\check{X}_k(\boldsymbol{s})$ does not have the same distribution as X. The time $Y_{k+1}(\boldsymbol{s})$ denotes the duration between the time instances at which update n_k and n_{k+1} are received. Note that $Y_{k+1}(\boldsymbol{s})$ includes the idle time of the server after reception of update n_k. Therefore, introducing idle time penalizes PAoI and it is always beneficial to send a request immediately after receiving an update. This implies that an optimal policy belongs to the set of work-conserving policies. Hence, we arrive at the following lemma.

Lemma 1. *An optimal policy $\boldsymbol{s}^*$ belongs to the set of work-conserving policies.*

[7]An optimal policy $\boldsymbol{s}^*$ never chooses a $\theta_n < x_{\min}$. Thus, the constraint $x_{\min} < \theta_{\min}$ only excludes the case $\theta_n = x_{\min}$.

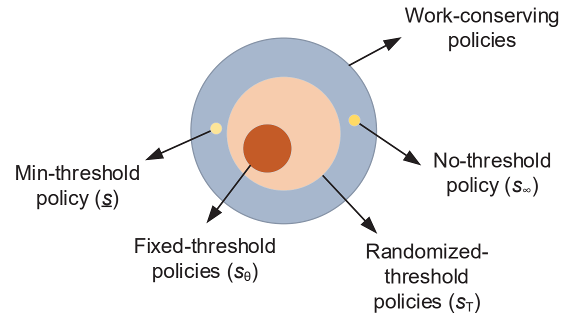

Fig. 4: Venn diagram showing different sets of policies.

Next, we define deterministic-repetitive threshold policies and compute $\zeta(\boldsymbol{s})$ for this class of policies.

Definition 3. *A deterministic-repetitive-threshold policy uses the same sequence of deterministic thresholds between any two AoI peaks.*

Let $\{\theta_i, i \geq 1\}$ denote a sequence of deterministic thresholds. Then, a deterministic-repetitive-threshold policy $\boldsymbol{s}$ repeats this sequence between any two peaks. We emphasize that θ_i is the threshold for ith request between any two AoI peaks, but not the threshold for ith request from time zero; here, we have abused the notation for brevity. In the following lemma, we characterize $\check{X}_k(\boldsymbol{s})$ and $Y_{k+1}(\boldsymbol{s})$. Recall that $\eta_i = \theta_i - d$.

Lemma 2. *For a deterministic-repetitive-threshold policy $\boldsymbol{s}$, $\check{X}_k(\boldsymbol{s})$ are i.i.d., $G_k(\boldsymbol{s})$ are i.i.d., and $Y_{k+1}(\boldsymbol{s})$ are identically distributed (but not i.i.d.). Further, we have*

$$\mathbb{E}[\check{X}(\boldsymbol{s})] = \int_0^{\theta_1} x f_X(x) dx + \sum_{j=1}^{\infty} \prod_{i=1}^{j} \mathbb{P}\{X_i > \theta_i\} \int_0^{\theta_{j+1}} x f_X(x) dx, \tag{5}$$

$$\begin{aligned} \mathbb{E}[Y(\boldsymbol{s})] =& \mathbb{E}[G(\boldsymbol{s})] + \mathbb{E}[\check{X}(\boldsymbol{s})] + \\ & \sum_{j=1}^{\infty} \prod_{i=1}^{j} \mathbb{P}\{X_i > \theta_i\} F_X(\theta_{j+1}) \sum_{i=1}^{j} \theta_i, \end{aligned} \tag{6}$$

$$\begin{aligned} \mathbb{E}[G(\boldsymbol{s})] = & \sum_{j=0}^{\infty} \prod_{i=1}^{j} \mathbb{P}\{X_i > \theta_i\} \times \\ & \left[F_X(\eta_{j+1}) d + \int_{\eta_{j+1}}^{\theta_{j+1}} (\theta_{j+1} - x) f_X(x) dx \right], \end{aligned} \tag{7}$$

and $\zeta(\boldsymbol{s}) = \mathbb{E}[\check{X}(\boldsymbol{s})] + \mathbb{E}[Y(\boldsymbol{s})]$.

Proof. The proof is given in Appendix A. □

Using the result in Lemma 2 we compute $\zeta(\boldsymbol{s}_\theta)$, the average PAoI under a fixed-threshold policy.

Corollary 1. *For a fixed-threshold policy $\boldsymbol{s}_\theta$, we have the average PAoI $\zeta(\boldsymbol{s}_\theta) = \mathbb{E}[\check{X}(\boldsymbol{s}_\theta)] + \mathbb{E}[Y(\boldsymbol{s}_\theta)]$, where*

$$\mathbb{E}[\check{X}(\boldsymbol{s}_\theta)] = \frac{\int_0^{\theta} x f_X(x) dx}{F_X(\theta)}, \tag{8}$$

$$\mathbb{E}[Y(\boldsymbol{s}_\theta)] = \mathbb{E}[\check{X}(\boldsymbol{s}_\theta)] + \mathbb{E}[G(\boldsymbol{s}_\theta)] + \frac{\theta\mathbb{P}(X > \theta)}{F_X(\theta)}, \quad (9)$$

$$\mathbb{E}[G(\boldsymbol{s}_\theta)] = \frac{F_X(\eta)d + \int_\eta^\theta (\theta - x) f_X(x)dx}{F_X(\theta)}, \quad (10)$$

and $\eta = \theta - d$.

Proof. The proof is given in Appendix B. □

The following lemma provides a simplified expression for $\mathbb{E}[Y(\boldsymbol{s}_\theta)]$ when $F_X(\cdot)$ is continuously differentiable.

Lemma 3. *If* $F_X(\cdot)$ *is continuously differentiable and* $F_X(x_{min}) = 0$, *then*

$$\mathbb{E}[Y(\boldsymbol{s}_\theta)] = \frac{\theta - \int_0^\eta F_X(x)dx}{F_X(\theta)}. \quad (11)$$

Proof. The proof is given in Appendix C. □

We note that the condition in Lemma 3 is satisfied by well-known continuous probability distributions including Erlang, Pareto, and log-normal, and we use it for computing average PAoI in our numerical analysis. Finally, in the following corollary, we present the average PAoI for min-threshold policy and no-threshold policy.

Corollary 2. *For a given distribution* $F_X(\cdot)$, *the average PAoIs achieved by the min-threshold policy* $\underline{\boldsymbol{s}}$ *and the no-threshold policy* $\boldsymbol{s}_\infty$ *are given by*

$$\zeta(\underline{\boldsymbol{s}}) = \zeta(\boldsymbol{s}_{\max\{x_{min},d\}}) \quad \textit{and} \quad \zeta(\boldsymbol{s}_\infty) = 2\mathbb{E}[X] + d.$$

Proof. The proof is given in Appendix D. □

Remark: Even though $F_X(x)$ and $f_X(x)$ are zero for $0 \le x < x_{\min}$, in the expressions above we chose to use lower limit 0 for the integrals. This is done for simplicity in the expressions; one can equivalently replace the lower limits by $x_{\min}$. Similarly, η in the lower limit can be replaced by $\max\{x_{\min}, \eta\}$.

V. Minimum Achievable Average PAoI

In this section, we first present a fixed-threshold policy that is optimal among randomized-threshold causal policies. Next, in any single-source-single-server queuing system, we present the optimal policy among all work-conserving policies and provide an expression for the minimum average PAoI.

Theorem 1. *Given the distribution of service times* $F_X(\cdot)$, *there exists a fixed-threshold policy* $\boldsymbol{s}_{\theta^\dagger}$ *in* $\mathcal{S}_\theta$ *that is optimal in* $\mathcal{S}_T$, *where* $\theta^\dagger$ *is given by*

$$\theta^\dagger \triangleq \underset{\theta \in [\theta_{min}, \theta_{max}]}{\arg\min} \ \zeta(\boldsymbol{s}_\theta). \quad (12)$$

Proof. The proof of the theorem is given in two steps. First, we formulate an infinite horizon average cost MDP problem equivalent to $\mathcal{P}$ in the domain of $\mathcal{S}_{\text{T}}$ and show that an optimal policy $\boldsymbol{s}^\dagger$ belongs to $\mathcal{S}_{\text{TR}}$. Next, we consider the decision process between two successive updates and show the independence of the optimal policy with the past decisions. Further, we prove that the fixed-threshold policy $\boldsymbol{s}_{\theta^\dagger}$ minimizes the average PAoI. The details are provided in Appendix E. □

Now, as illustrated in Figure 4, for a given problem, by choosing $\theta_{\min}$ arbitrarily close to $x_{\min}$ and $\theta_{\max}$ sufficiently large, the set $\mathcal{S}_{\text{T}} \cup \{\boldsymbol{s}_\infty, \underline{\boldsymbol{s}}\}$ can closely approximate the set of work-conserving policies. Therefore, from Theorem 1 and Lemma 1, it immediately follows that $\min(\zeta(\boldsymbol{s}_{\theta^\dagger}), \zeta(\boldsymbol{s}_\infty), \zeta(\underline{\boldsymbol{s}}))$ is the minimum achievable PAoI. Using Corollary 2, we arrive at the following result.

Theorem 2. *Given the service-time distribution* $F_X(\cdot)$, *the minimum average PAoI is given by*

$$\zeta^* = \min(\zeta(\boldsymbol{s}_{\theta^\dagger}), 2\mathbb{E}[X] + d, \zeta(\boldsymbol{s}_{\max\{x_{min},d\}})), \quad (13)$$

and thus, the optimal policy $\boldsymbol{s}^*$ *is either* $\boldsymbol{s}_{\theta^\dagger}$ *or* $\boldsymbol{s}_\infty$ *or* $\underline{\boldsymbol{s}}$, *whichever achieves* ζ^*.

From Theorem 2 and by the definition of θ^* (defined in (3)), we conclude $\theta^* = \theta^\dagger$, if $\zeta^* = \zeta(\boldsymbol{s}_{\theta^\dagger})$; $\theta^* = \max\{x_{\min}, d\}$, if $\zeta^* = \zeta(\boldsymbol{s}_{\max\{x_{\min},d\}})$; $\theta^* = \infty$, if $\zeta^* = 2\mathbb{E}[X] + d$.

Consider a single-source-single-server queuing system with a given service time distribution, having any arrival process and any queuing policy, e.g., FCFS/LCFS, preemptions/no preemptions, packet drops/no drops etc. We observe that, in our system, the update arrivals at the server is determined by the scheduled requests from the monitor, which is a design choice that allows for achieving a lower AoI when compared to systems with same service-time distribution but with some pre-determined structure for the arrival process. Furthermore, in systems where the update arrivals is a design choice, but has some form of queuing, AoI will be higher as queuing an update makes it stale. Combining the above two observations and the fact that we allow service preemption, which further reduces the minimum achievable PAoI, we conclude that the minimum average PAoI in any single-source-single-server queuing system will be at least the minimum average PAoI in our system. Therefore, the following corollary follows from Theorem 2.

Corollary 3. *In any single-source-single-server queuing system, given the service-time distribution* $F_X(\cdot)$, *service times are i.i.d., and the request delay is* d, *the minimum achievable average PAoI is given by* $\zeta(\boldsymbol{s}_{\theta^*})$, *where* θ^* *is defined in* (3).

VI. When are Preemptions Beneficial?

In this section, we study the conditions under which preemptions are beneficial, i.e., allowing preemptions will result in a strictly lower average PAoI. From Theorem 2, a necessary and sufficient condition for preemptions to be beneficial is as follows:

$$\exists\, \theta \ge 0 \text{ s. t. } \min(\zeta(\boldsymbol{s}_{\theta^\dagger}), \zeta(\boldsymbol{s}_{\max\{x_{\min},d\}})) < 2\mathbb{E}[X] + d. \quad (14)$$

In the following we consider an example distribution and obtain the condition under which preemptions are beneficial.

Case Study: Consider a random service time X that takes value t_1 with probability p and t_2 with probability $1-p$, where $0 < t_1 < t_2$. Also, consider that $d = 0$. The distribution of X can be written as follows:

$$f(x) = p\delta(x - t_1) + (1 - p)\delta(x - t_2),$$

$$F_X(x) = pu(x-t_1) + (1-p)u(x-t_2),$$

where $\delta(\cdot)$ and $u(\cdot)$ are Dirac delta function and unit-step function, respectively. For this distribution, $x_{\min} = t_1$ and therefore $\zeta(\boldsymbol{s}_{\max\{x_{\min},d\}}) = \zeta(\boldsymbol{s}_{x_{\min}}) = t_1(1+p)/p$. Note that choosing threshold $\theta < t_1$ or $\theta > t_2$ does not reduce average PAoI. Therefore, we compute $\zeta(\boldsymbol{s}_\theta)$ for $t_1 < \theta \leq t_2$. Using Corollary 1, and noting that, $\mathbb{E}[G_k(\boldsymbol{s}_\theta)] = 0$ for $d = 0$, we compute

$$\begin{aligned}
\zeta(\boldsymbol{s}_\theta) &= \mathbb{E}[\check{X}(\boldsymbol{s})] + \mathbb{E}[Y(\boldsymbol{s})] \\
&= \frac{2\int_{t_1}^{\theta} x f(x)dx}{F_X(\theta)} + \frac{\theta \mathbb{P}(X > \theta)}{F_X(\theta)} \\
&= \frac{2pt_1}{p} + \frac{\theta(1-p)}{p} \\
&= \frac{2pt_1 + (1-p)\theta}{p} > t_1(1+p)/p \ \text{ for all } \theta > t_1.
\end{aligned}$$

From the last step above we conclude that $\min(\zeta(\boldsymbol{s}_{x_{\min}}), \zeta(\boldsymbol{s}_{\theta^\dagger})) = \zeta(\boldsymbol{s}_{x_{\min}})$.

Since $\mathbb{E}[X] = pt_1 + (1-p)t_2$, using (14), preemptions are beneficial iff $\zeta(\boldsymbol{s}_{x_{\min}}) < 2\mathbb{E}[X]$, which implies

$$t_2 > \frac{t_1}{1-p}\left[1 + \frac{1}{p} - 2p\right]. \tag{15}$$

The condition in (15) establishes a lower bound on t_2 for preemptions to be beneficial. For example, if $p = \frac{1}{2}$ and $t_1 = 1$, then preemptions are beneficial if t_2 is greater than 2. In this case, whenever an update is not received within the duration t_1, it is optimal to send a new request just after t_1.

Example - Refreshing a Webpage: The above insights are directly useful for the problem of when to refresh a webpage. Consider that a webpage loads in time $t_1 = 0.1$ sec with probability 0.5 or half a minute, i.e., $t_2 = 30$ sec with probability 0.5. Since $t_2 > 2$, (15) is satisfied and hence a user should always refresh immediately after waiting 0.1 sec to minimize PAoI. This solution is also intuitive; because, $t_2 \gg t_1$ the user can greedily refresh the webpage after t_1 duration in order to reduce the expected time to load the webpage. Thus, solving for minimum PAoI provides a sensible solution for the refresh times demonstrating the applicability of this seemingly theoretical problem.

Note that the service-time distribution in the above example is simple enough to compute θ^* analytically and use (14) to infer whether preemptions will be beneficial or not. In general, it is not straightforward to do so for any service-time distribution. In the following lemma, we provide a sufficient condition that is useful to infer if preemptions are beneficial for a given class of distributions.

Lemma 4. *For any single-source-single-server queueing system, a sufficient condition for preemptions to be beneficial for minimizing average PAoI is as follows:*

$$\exists\, \theta \geq 0 \text{ such that } \mathbb{E}[X] < \mathbb{E}[X - \theta | X > \theta] + \frac{\theta}{2}.$$

Proof. The proof is given in Appendix F. □

From Lemma 4, we infer that a sufficient condition is the existence of a θ that satisfies $\mathbb{E}[X - \theta | X > \theta] > \mathbb{E}[X]$.

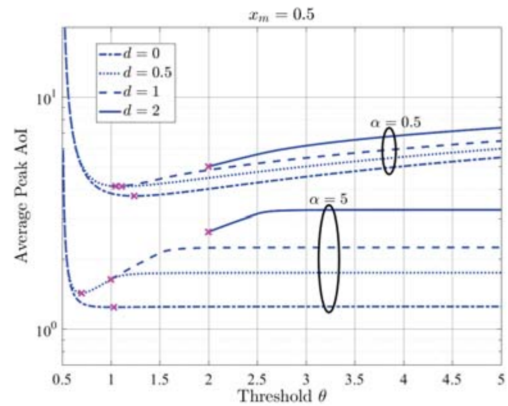

Fig. 5: Average PAoI vs. θ under the Pareto service-time distribution for different α and $x_m = 0.5$.

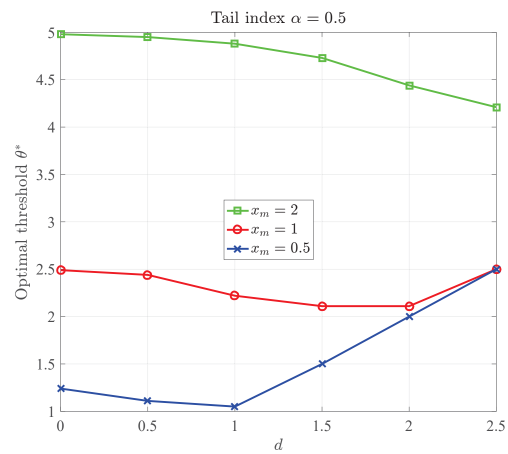

Fig. 6: Optimal threshold versus d for different x_m and $\alpha = 0.5$.

This condition implies that given an elapsed time θ, the expected residual should be greater than the mean value. This is satisfied by heavy-tailed distributions and hyper-exponential distributions [38].

VII. Numerical Analysis

In this section, we compute optimal fixed thresholds for Pareto, log-normal, and Erlang service-time distributions. Pareto and log-normal distributions are used to illustrate the effectiveness of preemptions for heavy-tailed distributions, and the Erlang distribution is chosen due to the fact that it models a tandem of exponential (memoryless) servers. We compare the average PAoI achieved by no-threshold policy, optimal fixed-threshold policy $\boldsymbol{s}_{\theta^\dagger}$, and median-threshold policy, which uses median of the service-time distribution as the fixed threshold. We study the median-threshold policy because it can be useful in cases where the distribution of the service times is not known apriori but the sample median can be computed from

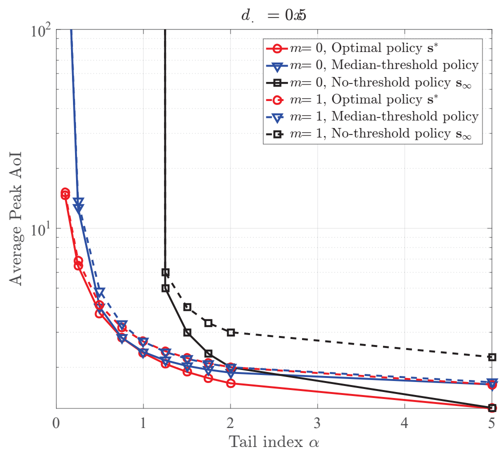

Fig. 7: Average PAoI achieved by different policies under the Pareto service-time distribution with varying α and x_m=0.5.

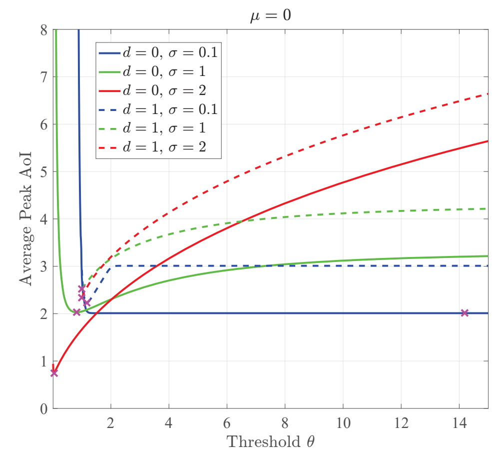

Fig. 8: Average PAoI vs. θ under log-normal distribution for different σ and $\mu = 0$.

the history of service times. Furthermore, unlike sample mean, sample median is always finite and is an unbiased estimate. Finally, we conclude this section with a case study on the effect of variance of a distribution on the minimum average PAoI. All the numerical computations are implemented in MATLAB.

A. Pareto Service-Time Distribution

The Pareto distribution is characterized by two parameters $\{x_m, \alpha\}$, where x_m is the scale parameter and α is the tail index. The smaller the α, the heavier the tail. The default values are $x_m = 0.5$ and $\alpha = 0.5$. In Figure 5, we plot the average PAoI $\zeta(\boldsymbol{s}_\theta)$, computed using Corollary 1 and Lemma 3, by varying the threshold θ for α equal to 0.5 and 5. The minimum values of $\zeta(\boldsymbol{s}_\theta)$ are indicated by the points in magenta. Observe that, for $d = 0$, $\zeta(\boldsymbol{s}_\theta)$ is convex in θ for both the α values. Therefore, in this case we obtain $\theta^* = \theta^\dagger$. For $d > 0$, Since θ values are lower bounded by d, we observe that $\theta^* = d$ for some parameter values. Further, for $\alpha = 5$, as θ increases PAoI converges to $2\mathbb{E}[X] + d$, the value achieved by the no-threshold policy. For $\alpha = 0.5$, this convergence does occur, albeit at larger threshold values.

In Figure 6, we plot the optimal threshold θ^* as d increases. It can be observed that θ^* initially decreases as d increases to compensate for the increase of PAoI due to d. As d increases beyond a certain value, θ^* attains d as it is bounded below by d. In Figure 7, we compare the average PAoI achieved by different policies for different d values. Observe that for higher α values, i.e., the distribution has a light tail, the no-threshold policy achieves lower average PAoI values, comparable with that of the optimal policy. On the other hand, for $\alpha \leq 1$, the distribution has a heavy tail and infinite mean, and thus the average PAoI under the no-threshold policy is infinity. In contrast, the optimal policy achieves finite average PAoI values in this case, and this illustrates the effectiveness of preemptions for heavy-tailed distributions. Furthermore, the median-threshold policy performs consistently well and is an attractive choice when the parameters $\{x_m, \alpha\}$ are not known apriori, but only an estimate of the median is available. Finally, we note that the PAoI values achieved by a policy is insensitive to $d = 0$ and $d = 1$ when the tail is heavier.

B. Log-normal Service-Time Distribution

Log-normal distribution is characterized by the parameters (μ, σ), where μ is the mean and σ is the standard deviation of the underlying normal distribution. For this distribution the higher the σ, the heavier the tail. The default value for μ is 0. Since the median value is given by e^μ, its default value is 1. In Figure 8, we plot $\zeta(\boldsymbol{s}_\theta)$ by varying the threshold θ. Again, the minimum values of $\zeta(\boldsymbol{s}_\theta)$ are indicated by the points in magenta. Observe that, for a given d the minimum PAoI that can be achieved is lower for higher σ value. This is interesting as it suggests that the higher the variance, the lower the average PAoI that can be achieved. Note that this statement is not true for Pareto distribution (cf. Figure 5). In Section VII-D, we will further study the affect of variance for Pareto distribution. In Figure 9, we compare the average PAoI achieved by different policies. Under the no-threshold policy, it increases exponentially with σ as $\mathbb{E}[X] = e^{(\mu + \frac{\sigma^2}{2})}$. Under both median-threshold policy and optimal policy, average PAoI decreases with σ. Note that, for $d = 0$, we compute $\theta^* = 0$, and the minimum average PAoI goes to 0 as σ goes to infinity. However, in Figure 9 we lower bound the thresholds by 0.01 and thus the minimum average PAoI converges to this value.

C. Erlang Service-Time Distribution

Erlang distribution is characterized by two parameters $\{k, \lambda\}$, where k is the shape parameter and λ is the rate parameter. In Figure 10, we plot the average PAoI by varying the threshold θ and $d = 1$. The minimum values of $\zeta(\boldsymbol{s}_\theta)$ are indicated by the points in magenta. Recall that, for $k = 1$ the Erlang distribution results in an exponential distribution. For this case, from Figure 10 we observe that the function $\zeta(\boldsymbol{s}_\theta)$ is concave, and therefore $\boldsymbol{s}^*$ in this case is min-threshold policy with $\theta^* = d = 1$, and $\zeta^* = 2$. In contrast, for $k \geq 2$, the

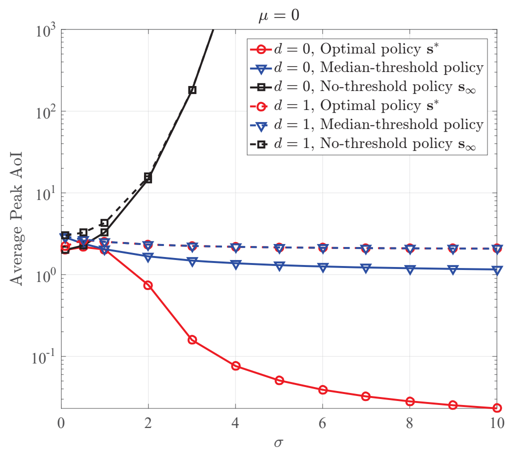

Fig. 9: Average PAoI achieved by different policies under log-normal distribution for varying σ and $\mu = 0$.

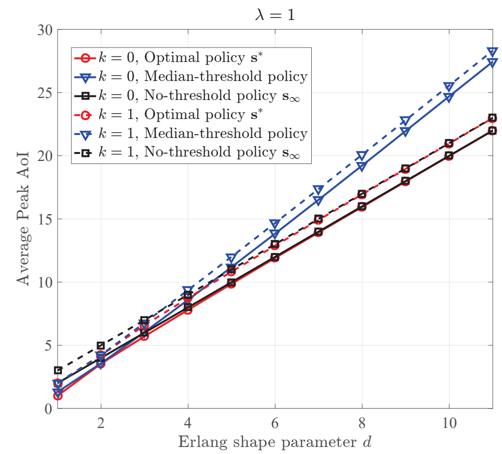

Fig. 11: Average PAoI achieved by different policies under the Erlang service-time distribution with varying k and $\lambda = 1$.

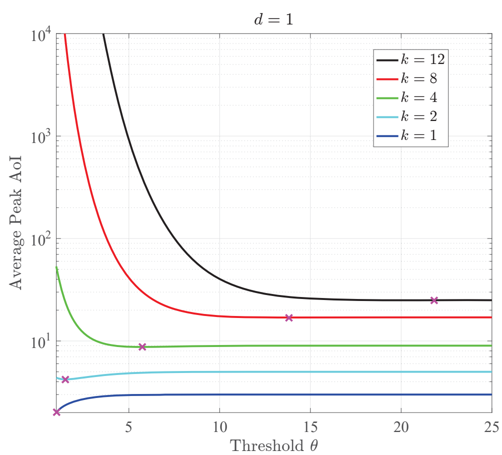

Fig. 10: Average PAoI vs. θ under the Erlang service-time distribution for different k and $\lambda = 1$.

functions are convex in θ and we obtain $\theta^* = \theta^\dagger$. In Figure 11, we compare the average PAoI achieved by different policies. It can be observed that no-threshold policy has average PAoI close to $\zeta(\boldsymbol{s}^*)$. This is because the sufficiency condition $\mathbb{E}[X - \theta | X > \theta] > \mathbb{E}[X]$ is not satisfied by the Erlang distribution for any θ [38], and thus allowing preemptions does not significantly reduce average PAoI. The average PAoI under median-threshold policy is relatively higher and also diverges from both no-threshold and $\boldsymbol{s}^*$ when k increases, thus suggesting that using preemptions with arbitrary threshold could in fact penalize the average PAoI. Therefore, it is important to verify first if preemptions are beneficial for a given service-time distribution. The conditions provided in (14) and Lemma 4 are potentially useful toward this end.

D. Effect of Variance

Consider a scenario where there are multiple parallel preemptive servers with equal mean service time that connect the information source and the monitor. Only one of them should be chosen to send the packets from the source to the monitor. For this scenario, a natural choice is to use the prememptive server which provides the lowest possible PAoI. An interesting finding from our simulations is that the server with largest variance in the service times has lower minimum achievable PAoI. In Table II, we present the minimum PAoI ζ^* that is achieved under three preemptive servers, each having Pareto-service distribution with equal mean 1 but different scale parameter x_m and tail index α, and the request delay to each of the servers is zero. Observe that as α increases ζ^* increases. This implies that the larger the variance the lower the PAoI value that can be achieved using preemptions. If this statement is true, in general, is left for the future work.

TABLE II: Paerto-service distributions with mean 1. Request delay $d = 0$.

$\{x_m, \alpha\}$	mean	Variance	$\theta^\dagger$	min PAoI (ζ^*)
$\{\frac{1}{3}, \frac{3}{2}\}$	1	∞	0.753	1.263
$\{\frac{1}{2}, 2\}$	1	∞	1.110	1.661
$\{\frac{5}{6}, 6\}$	1	$\frac{1}{24}$	1.713	1.994

VIII. Conclusion

We have studied a problem of finding the minimum achievable average PAoI for a given service-time distribution. To this end, we have considered generate-at-will source model with service preemptions and request delays d. Using an MDP formulation we have shown that a fixed-threshold policy achieves minimum average PAoI in the set of randomized-threshold causal policies. The minimum achievable average PAoI in any single-source-single-server queuing system is then given by the minimum average PAoI achieved by an optimal fixed-threshold policy. Using the fact that no-threshold policy is optimal among all non-preemptive policies, we establish necessary and sufficient conditions for the service-time distributions under which allowing preemptions result in a lower minimum average PAoI. In the numerical analysis, we have used the Pareto and log-normal service-time distributions to illustrate the effectiveness of preemptions for heavy-tailed distributions. Based on the inference from Section VII-D, we conjecture that when two distributions are drawn from the same family and has the same mean but different variances, then the minimum achievable average PAoI will be lower for the distribution with higher variance.

We conclude by presenting some open problems. A theoretical study on the effect of variance on the minimum average PAoI, proving/disproving the above conjecture, would be interesting. One may consider minimum achievability for other functions of AoI including the average AoI. Also, it would be interesting to consider multiple information sources and study how preemptions affect both sampling and communication scheduling.

Appendix

A. Proof of Lemma 2

We first analyse $\check{X}_k(\boldsymbol{s})$. Recall that n_k is the index of the kth received update and $\boldsymbol{s}$ repeats the same sequence $\{\theta_i, i \geq 1\}$ between any two peaks. Consider update $n_{k-1}+1$, the first update that is generated after $k-1$th packet is received, if $X_{n_{k-1}+1} \leq \theta_1$, then it will be received successfully. In this case, we set $n_k = n_{k-1}+1$ and $\check{X}_k(\boldsymbol{s}) = X_{n_{k-1}+1}$. If $X_{n_{k-1}+1} > \theta_1$, then update $n_{k-1}+1$ will be preempted by sending request $n_{k-1}+2$. In this case the above statements can be similarly repeated by comparing $X_{n_{k-1}+2}$ and θ_2. Since $X_{n_{k-1}+i}$ and X_i are i.i.d., from the above analysis we characterize $\check{X}_k(\boldsymbol{s})$ as follows:

$$\check{X}_k(\boldsymbol{s}) = \begin{cases} X_1 & X_1 \leq \theta_1 \\ X_2 & X_1 > \theta_1, X_2 \leq \theta_2 \\ X_3 & X_1 > \theta_1, X_2 > \theta_2, X_3 \leq \theta_3 \\ \vdots & \end{cases}$$

The above characterization of $\check{X}_k(\boldsymbol{s})$ is true for any k as $\boldsymbol{s}$ is a deterministic-repetitive threshold policy. Since X_n are i.i.d. we infer that $\check{X}_k(\boldsymbol{s})$ are also i.i.d. In the following we write $\check{X}_k(\boldsymbol{s})$ using indicator functions.

$$\begin{aligned} \check{X}_k(\boldsymbol{s}) =& X_1 \mathbb{1}\{X_1 \leq \theta_1\} + \\ & \sum_{j=1}^{\infty} \prod_{i=1}^{j} \mathbb{1}\{X_i > \theta_i\} X_{j+1} \mathbb{1}\{X_{j+1} \leq \theta_{j+1}\} \end{aligned} \tag{16}$$

Taking expectation on both sides we arrive at (5).

To analyse $Y_{k+1}(\boldsymbol{s})$, we first need to compute $G_k(\boldsymbol{s})$ – the time instant at which a new update is generated after $D_{n_k}(\boldsymbol{s})$. Let i_k denote the the number of requests sent between AoI peaks $k-1$ and k after which update k is received successfully. This implies $\check{X}_k(\boldsymbol{s}) = X_{n_{k-1}+i_k} \leq \theta_{i_k}$. If a request is sent at time $D_{n_k}(\boldsymbol{s})$, then $G_k(\boldsymbol{s}) = d$. This happens when $\check{X}_k(\boldsymbol{s}) \leq \eta_{i_k}$. If $\eta_{i_k} < \check{X}_k(\boldsymbol{s}) \leq \theta_{i_k}$, then we obtain $G_k(\boldsymbol{s}) = \theta_{i_k} - \check{X}_k(\boldsymbol{s})$. See Figure 12 for an illustration of $G_k(\boldsymbol{s})$ for some policy. Observe that, just before the peak A_{k+1} the monitor has sent a new request as θ_2 is elapsed. For the new request the threshold should be θ_3. However, before this request could reach the source, the update for the previous request is received. As noted before, this update will be accepted by the monitor, no new request will be generated, and the monitor replaces threshold θ_3 by θ_1 in accordance with the deterministic-repetitive threshold policy.

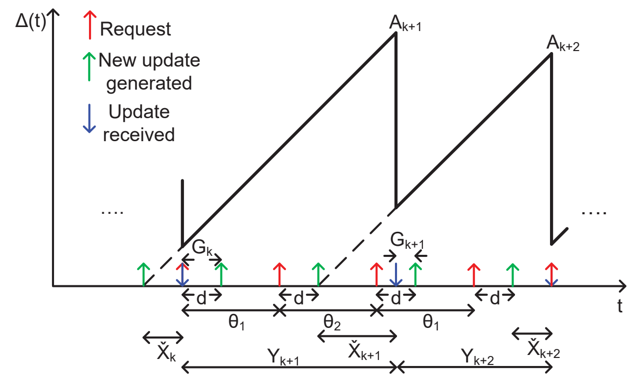

Fig. 12: An illustration of the server idle times G_k.

Using similar analysis as in the case of $\check{X}_k(\boldsymbol{s})$, we obtain

$$G_k(\boldsymbol{s}) = \begin{cases} d & X_1 \leq \eta_1 \\ (\theta_1 - X_1) & \eta_1 < X_1 \leq \theta_1 \\ d & X_1 > \theta_1, X_2 \leq \eta_2, \\ (\theta_2 - X_2) & X_1 > \theta_1, \eta_2 < X_2 \leq \theta_2 \\ \vdots & \end{cases}$$

It is easy to see that $G_k(\boldsymbol{s})$ are i.i.d., and we obtain

$$\begin{aligned} \mathbb{E}[G_k(\boldsymbol{s})] = \sum_{j=0}^{\infty} \prod_{i=1}^{j} \mathbb{P}\{X_i > \theta_i\} \times \Big[F_X(\eta_{j+1}) d + \\ \int_{\eta_{j+1}}^{\theta_{j+1}} (\theta_{j+1} - x) f_X(x) dx \Big]. \end{aligned}$$

At time $D_{n_k}(\boldsymbol{s}) + G_k(\boldsymbol{s})$, update n_k+1 is generated. If $X_{n_{k+1}} \leq \theta_1$, then it will be received successfully and $Y_{k+1}(\boldsymbol{s})$ equals $G_k(\boldsymbol{s}) + X_{n_k+1}$. Otherwise, $G_k(\boldsymbol{s}) + \theta_1$ will get added to $Y_{k+1}(\boldsymbol{s})$ and then $X_{n_{k+2}}$ is compared with θ_2 and the arguments are repeated. From this analysis and noting that X_{n_k+i} and X_i are i.i.d., we characterized $Y_{k+1}(\boldsymbol{s})$ as follows[8]:

$$Y_{k+1}(\boldsymbol{s}) = \begin{cases} G_k(\boldsymbol{s}) + X_1 & X_1 \leq \theta_1 \\ G_k(\boldsymbol{s}) + \theta_1 + X_2 & X_1 > \theta_1, X_2 \leq \theta_2 \\ G_k(\boldsymbol{s}) + \theta_1 + \theta_2 + X_3 & X_1 > \theta_1, X_2 > \theta_2, X_3 \leq \theta_3 \\ \vdots & \end{cases}$$

The above equation is further simplified in (17). Note that $Y_{k+1}(\boldsymbol{s})$ and $Y_k(\boldsymbol{s})$ have the same distribution but are dependent through $G_k(\boldsymbol{s})$. Nevertheless, $Y_{k+1}(\boldsymbol{s})$ and $Y_{k-1}(\boldsymbol{s})$ are independent for all k. Again, taking expectation on both sides and noting that X_i are i.i.d. we arrive at (6).

[8]Even though we use the same $\{X_i\}$ to characterize $Y_{k+1}(\boldsymbol{s})$ as well $\check{X}_k(\boldsymbol{s})$ and $G_k(\boldsymbol{s})$, one should note that actual service time random variables that result in $\check{X}_k(\boldsymbol{s})$ and $G_k(\boldsymbol{s})$ are different from that of $Y_{k+1}(\boldsymbol{s})$. We, however, chose to use the same $\{X_i\}$ as it doesn't change the intended results and avoids introducing any additional notation.

$$Y_{k+1}(\boldsymbol{s})=G_k(\boldsymbol{s})+X_1\mathbb{1}\{X_1\le\theta_1\}+\sum_{j=1}^{\infty}\prod_{i=1}^{j}\mathbb{1}\{X_i>\theta_i\}\Big[\mathbb{1}\{X_{j+1}\le\theta_{j+1}\}\sum_{i=1}^{j}\theta_i+X_{j+1}\mathbb{1}\{X_{j+1}\le\theta_{j+1}\}\Big]$$

$$= G_k(\boldsymbol{s}) + \check{X}_{k+1}(\boldsymbol{s}) + \sum_{j=1}^{\infty}\prod_{i=1}^{j}\mathbb{1}\{X_i>\theta_i\}\mathbb{1}\{X_{j+1}\le\theta_{j+1}\}\sum_{i=1}^{j}\theta_i \tag{17}$$

Since $\check{X}_k(\boldsymbol{s})$ are i.i.d., and $Y_k(\boldsymbol{s})$ have identical distribution, and $A_{k+1}(\boldsymbol{s}) = \check{X}_k(\boldsymbol{s})+Y_{k+1}(\boldsymbol{s})$, we conclude that $A_k(\boldsymbol{s})$ for all k have identical distribution with mean $\mathbb{E}[\check{X}(\boldsymbol{s})]+\mathbb{E}[Y(\boldsymbol{s})]$. Therefore,

$$\zeta(\boldsymbol{s}) = \lim_{K\to\infty}\frac{1}{K}\mathbb{E}_{\boldsymbol{s}}\left[\sum_{k=1}^{K}A_k(\boldsymbol{s})\right] = \mathbb{E}[\check{X}(\boldsymbol{s})] + \mathbb{E}[Y(\boldsymbol{s})].$$

B. Proof of Corollary 1

Substituting $\theta_i = \theta$ for all i in (5), we obtain

$$\mathbb{E}[\check{X}(\boldsymbol{s}_\theta)] \overset{(a)}{=} \int_0^\theta x f_X(x)dx + \sum_{j=1}^{\infty}\mathbb{P}(X>\theta)^j\int_0^\theta x f_X(x)dx \tag{18}$$

$$\overset{(b)}{=} \int_0^\theta x f_X(x)dx\sum_{j=0}^{\infty}\mathbb{P}(X>\theta)^j \overset{(c)}{=} \frac{\int_0^\theta x f_X(x)dx}{F_X(\theta)}. \tag{19}$$

In step (a) we have used $\mathbb{E}[X\mathbb{1}\{x\le\theta\}] = \int_0^\theta x f_X(x)dx$. In step (c) we have used the sum for infinite geometric series.

In order to compute $\mathbb{E}[Y(\boldsymbol{s}_\theta)]$, we first compute $\mathbb{E}[G(\boldsymbol{s}_\theta)]$ by substituting $\theta_i = \theta$ and $\eta_i = \eta$ for all i in (7).

$$\mathbb{E}[G(\boldsymbol{s}_\theta)] = \sum_{j=0}^{\infty}\prod_{i=1}^{j}\mathbb{P}\{X_i>\theta\}\Big[F_X(\eta)d + \int_\eta^\theta(\theta-x)f_X(x)dx\Big]$$

$$= \frac{F_X(\eta)d + \int_\eta^\theta(\theta-x)f_X(x)dx}{F_X(\theta)} \tag{20}$$

By substituting $\theta_i = \theta$ for all i and using (18) and (20) in (6), we obtain

$$\mathbb{E}[Y(\boldsymbol{s}_\theta)] = \mathbb{E}[G(\boldsymbol{s}_\theta)] + \mathbb{E}[\check{X}(\boldsymbol{s}_\theta)] + \sum_{j=1}^{\infty}\mathbb{P}(X>\theta)^j F_X(\theta)j\theta$$

$$= \mathbb{E}[G(\boldsymbol{s}_\theta)] + \mathbb{E}[\check{X}(\boldsymbol{s}_\theta)] + \theta F_X(\theta)\sum_{j=1}^{\infty}j\mathbb{P}(X>\theta)^j$$

$$= \mathbb{E}[G(\boldsymbol{s}_\theta)] + \mathbb{E}[\check{X}(\boldsymbol{s}_\theta)] + \frac{\theta\mathbb{P}(X>\theta)}{F_X(\theta)}. \tag{21}$$

C. Proof of Lemma 3

Given that $F_X(\cdot)$ is continuously differentiable, in the following we use integral by parts.

Case 1: $\eta > x_{\text{min}}$. For this case, we have

$$\int_0^\theta x f_x(x)dx = \int_{x_{\text{min}}}^\theta x f_x(x)dx$$

$$= \theta F_X(\theta) - x_{\text{min}}F_X(x_{\text{min}}) - \int_{x_{\text{min}}}^\theta F_X(x)dx$$

$$= \theta F_X(\theta) - \int_0^\theta F_X(x)dx \tag{22}$$

Further,

$$\int_\eta^\theta(\theta-x)f_x(x)dx$$

$$= \theta[F_X(\theta) - F_X(\eta)] - [\theta F_X(\theta) - \eta F_X(\eta) - \int_\eta^\theta F_X(x)dx]$$

$$= -F_X(\eta)d + \int_\eta^\theta F_X(x)dx. \tag{23}$$

From (23) and (10), we obtain

$$\mathbb{E}[G(\boldsymbol{s}_\theta)] = \frac{F_X(\eta)d + \int_\eta^\theta(\theta-x)f_X(x)dx}{F_X(\theta)} = \frac{\int_\eta^\theta F_X(x)dx}{F_X(\theta)}. \tag{24}$$

From (9) we have

$$\mathbb{E}[Y(\boldsymbol{s}_\theta)] = \mathbb{E}[\check{X}(\boldsymbol{s}_\theta)] + \mathbb{E}[G(\boldsymbol{s}_\theta)] + \frac{\theta\mathbb{P}(X>\theta)}{F_X(\theta)}$$

$$= \frac{\theta F_X(\theta) - \int_0^\theta F_X(x)dx}{F_X(\theta)} + \frac{\int_\eta^\theta F_X(x)dx}{F_X(\theta)} + \frac{\theta\mathbb{P}(X>\theta)}{F_X(\theta)}$$

$$= \frac{\theta - \int_0^\eta F_X(x)dx}{F_X(\theta)}.$$

In the second step above, we have used (22) and (24).

Case 2: $\eta \le x_{\text{min}}$. For this case $F(\eta) = 0$, and we have

$$\int_\eta^\theta(\theta-x)f_x(x)dx = \int_{x_{\text{min}}}^\theta(\theta-x)f_x(x)dx$$

$$\Longrightarrow \int_\eta^\theta(\theta-x)f_x(x)dx + \int_{x_{\text{min}}}^\theta f_x(x)dx = \theta F_X(\theta).$$

Again, using the above expression in (9), we obtain

$$\mathbb{E}[Y(\boldsymbol{s}_\theta)] = \theta + \frac{\theta\mathbb{P}(X>\theta)}{F_X(\theta)} = \frac{\theta}{F_X(\theta)}.$$

Also, for $\eta \le x_{\text{min}}$, we have

$$\frac{\theta - \int_0^\eta F_X(x)dx}{F_X(\theta)} = \frac{\theta}{F_X(\theta)}.$$

Hence the result is proven.

D. Proof of Corollary 2

Both $\underline{\boldsymbol{s}}$ and $\boldsymbol{s}_\infty$ are deterministic-repetitive-threshold policies using fixed thresholds $\theta = \max\{x_{\text{min}}, d\}$ and $\theta = \infty$, respectively. Therefore, applying the result in Corollary 1 to these policies, we obtain $\zeta(\underline{\boldsymbol{s}}) = \zeta(\boldsymbol{s}_{\max\{x_{\text{min}},d\}})$, and for $\zeta(\boldsymbol{s}_\infty)$ we have

$$\mathbb{E}[\check{X}(\boldsymbol{s}_\infty)] = \frac{\int_0^\infty x f_X(x)dx}{F_X(\infty)} = \mathbb{E}[X]$$

and

$$\mathbb{E}[G(\boldsymbol{s}_\infty)] = \lim_{\theta\to\infty} \frac{F_X(\eta)d + \int_\eta^\theta (\theta - x) f_X(x)dx}{F_X(\theta)} = d.$$

Using above results for $\mathbb{E}[\check{X}(\boldsymbol{s}_\infty)]$ and $\mathbb{E}[G(\boldsymbol{s}_\infty)]$ in (9), we obtain $\mathbb{E}[Y(\boldsymbol{s}_\infty)] = \mathbb{E}[X]+d$. Therefore, $\zeta(\boldsymbol{s}_\infty) = 2\mathbb{E}[X]+d$.

E. Proof of Theorem 1

In this proof, we use the notation F_1^N to denote the sequence $[F_1, \ldots, F_N]$ and $\mathcal{A}^N$ to denote the N-fold Cartesian product of a set $\mathcal{A}$. Let $I_{k,r} = \{A_1^{k-1}, \check{X}_0^{k-1}, \tilde{I}_1^{k-1}, \theta_{k,1}, \ldots, \theta_{k,r-1}\}$ denote the causal information available to the scheduler at rth request after $(k-1)$th update, where $\tilde{I}_k = \{\theta_{k,1}, \ldots, \theta_{k,\check{R}_k}\}$ denotes the sequence of threshold values between $(k-1)$th and kth updates and $\check{R}_k = n_k - n_{k-1}$. Here, $I_{k,0}$ denotes the information state exactly at $(k-1)$th update. Further, we use $i_{k,r}$ to denote a realization of $I_{k,r}$ and $\delta_{k,r}(i_{k,r})$ to denote the conditional distribution function of the threshold $\theta_{k,r}$ given $i_{k,r}$. Let $\eta_{k,r} = \theta_{k,r} - d$ and $\bar{F}_X(x) = 1 - F_X(x)$. A randomized-threshold causal policy $\boldsymbol{s}$ specifies a causal sub-policy at each update $k-1$, denoted by $\mu_k(i_{k,0})$, where μ_k specifies the conditional distributions $\delta_{k,r}(i_{k,r})$ for all requests between the $(k-1)$th and kth updates. For a given $i_{k,0}$, the sub-policy μ_k belongs to $\mathcal{U}$, which is the set of randomized sub-policies that specify the distributions of thresholds between two successive updates. For a given $i_{k,r}$, the distribution $\delta_{k,r}$ belongs to $\mathcal{F}$, which is the set of valid probability distribution functions.

Now, we solve for an optimal policy in $\mathcal{S}_\text{T}$ in two steps. First, we formulate an infinite-horizon average cost MDP problem with the decision epochs as the times at which the updates are received. In the next step, we consider the decision epochs as the times at which requests are sent between any two successive updates.

Step 1: The identified infinite-horizon average cost MDP problem equivalent to $\mathcal{P}$ has the following elements:

- *State:* the service time of the previous update, $\check{X}_{k-1} \in \mathbb{R}_{\geq 0}$,
- *Action:* the sequence of conditional distribution functions,
$$\mu_k(i_{k,0}) = \left\{\delta_{k,r}(i_{k,r}) \middle| r \in \mathbb{N}\right\}$$
- *Cost function:* the expected PAoI given $i_{k,0}$,
$$\begin{aligned} c_k(i_{k,0}, \mu_k) &= \mathbb{E}_{\mu_k}\left[A_k | I_{k,0} = i_{k,0}\right] \\ &= \check{x}_{k-1} + \mathbb{E}_{\mu_k}\left[G_k + B_k + \check{X}_k \middle| I_{k,0} = i_{k,0}\right], \end{aligned}$$
where $B_k = Y_k - (G_k + \check{X}_k)$ denotes the time lost due to preemptions.

Here, using the result from the Lemma 2, we obtain

$$\begin{aligned} \alpha_X(\mu_k) &=: \mathbb{E}_{\mu_k}\left[\check{X}_k | I_{k,0} = i_{k,0}\right] = \mathbb{E}_{\mu_k}\left[\check{X}_k\right] \\ &= \mathbb{E}_{\mu_k}\left[\sum_{r=1}^{\infty} \prod_{m=1}^{r-1} \bar{F}_X(\theta_{k,m}) \int_0^{\theta_{k,r}} x f_X(x)dx\right], \end{aligned}$$

$$\begin{aligned} \beta_X(\mu_k) &=: \mathbb{E}_{\mu_k}\left[B_k | I_{k,0} = i_{k,0}\right] \\ &= \mathbb{E}_{\mu_k}\left[Y_k | I_{k,0} = i_{k,0}\right] - \mathbb{E}_{\mu_k}\left[G_k + \check{X}_k | I_{k,0} = i_{k,0}\right] \\ &= \mathbb{E}_{\mu_k}\left[Y_k\right] - \mathbb{E}_{\mu_k}\left[G_k + \check{X}_k\right] \\ &= \mathbb{E}_{\mu_k}\left[\sum_{r=1}^{\infty} \prod_{m=1}^{r} \bar{F}_X(\theta_{k,m})\theta_{k,r}\right], \end{aligned}$$

$$\begin{aligned} \gamma_X(\mu_k) &=: \mathbb{E}_{\mu_k}\left[G_k | I_{k,0} = i_{k,0}\right] = \mathbb{E}_{\mu_k}\left[G_k\right] \\ &= \mathbb{E}_{\mu_k}\Bigg[\sum_{r=1}^{\infty} \prod_{m=1}^{r-1} \bar{F}_X(\theta_{k,m}) \Big[F_X(\eta_{k,r})d \\ &\qquad + \int_{\eta_{k,r}}^{\theta_{k,r}} (\theta_{k,r} - x) f_X(x)dx\Big]\Bigg], \end{aligned}$$

where $\alpha_X : \mathcal{U} \to \mathbb{R}$, $\beta_X : \mathcal{U} \to \mathbb{R}$, and $\gamma_X : \mathcal{U} \to \mathbb{R}$ are deterministic functions. Therefore, we can express the cost function as

$$c_k(\check{x}_{k-1}, \mu_k) = \check{x}_{k-1} + \alpha_X(\mu_k) + \beta_X(\mu_k) + \gamma_X(\mu_k). \quad (25)$$

Now, the problem $\mathcal{P}$ in the domain of $\mathcal{S}_\text{T}$ is equivalent to the infinite horizon average cost problem given by

$$\boldsymbol{s}^\dagger = \arg\min_{\boldsymbol{s}\in\mathcal{S}_\text{T}} \left\{ \lim_{K\to\infty} \frac{1}{K} \mathbb{E}_{\boldsymbol{s}}\left[\sum_{k=1}^{K} c_k(\check{x}_{k-1}, \mu_k)\right]\right\}, \quad (26)$$

where $\boldsymbol{s}^\dagger$ is the optimal policy. Note that for a given policy $\boldsymbol{s} \in \mathcal{S}_\text{T} \subset \mathcal{S}$, we have $\alpha_X(\mu_k) < \infty$, $\beta_X(\mu_k) < \infty$, and $\gamma_X(\mu_k) < \infty$ because the limit in (2) exists for all $\boldsymbol{s} \in \mathcal{S}$. Given $\check{x}_1$, let J_K denote the minimum expected cumulative cost over a finite horizon $k = [1, \cdots, K]$, given by

$$J_K(\check{x}_0) = \min_{\mu_1^K \in \mathcal{U}^K} \left\{\mathbb{E}_{\mu_1^K}\left[\sum_{k=1}^{K} c_k(\check{x}_{k-1}, \mu_k)\right]\right\}. \quad (27)$$

The optimal finite-horizon solution to (27) can be obtained using the backward recursion of the stochastic Bellman's dynamic programming [15] given by

$$V_k(i_{k,0}) = \min_{\mu_k\in\mathcal{U}} \left\{c_k(\check{x}_{k-1}, \mu_k) + \mathbb{E}_{\mu_k}\left[V_{k+1} \middle| I_{k,0} = i_{k,0}\right]\right\},$$

where the value function V_k denotes the optimal expected cumulative cost-to-go from k to K. Since we consider a finite-horizon without a terminal cost, we initialize the recursion with $V_{K+1} = 0$. Thus, for $k = K$, we have

$$V_K(i_{K,0}) = \check{x}_{K-1} + \underbrace{\min_{\mu_K\in\mathcal{U}} \left\{\alpha_X(\mu_K) + \beta_X(\mu_K) + \gamma_X(\mu_K)\right\}}_{\tilde{V}_K}$$

where $\tilde{V}_K$ is a constant for all $i_{K,0}$ since α_X, β_X, and γ_X do not depend on $i_{K,0}$. Similarly, for $k = K-1$,

$$V_{K-1}(i_{K-1,0}) = \check{x}_{K-2} + \tilde{V}_{K-1} + \tilde{V}_K, \quad (28)$$

where

$$\tilde{V}_{K-1} = \min_{\mu_{K-1}\in\mathcal{U}} \left\{2\alpha_X(\mu_{K-1}) + \beta_X(\mu_{K-1}) + \gamma_X(\mu_{K-1})\right\},$$

$$\mu_{K-1}^\dagger = \operatorname*{argmin}_{\mu_{K-1}\in\mathcal{U}} \left\{2\alpha_X(\mu_{K-1}) + \beta_X(\mu_{K-1}) + \gamma_X(\mu_{K-1})\right\}.$$

Here, $\tilde{V}_{K-1}$ is a constant and the optimal sub-policy $\mu_{K-1}^\dagger$ is independent of $i_{K-1,0}$. Now, for some $k = m$ such that $1 < m \leq K-1$, we assume that the optimal sub-policy satisfies $\mu_m^\dagger = \mu_{K-1}^\dagger$ and the value function has the same structure as in (28), that is given by

$$V_m(i_{m,0}) = \check{x}_{m-1} + \textstyle\sum_{l=m}^K \tilde{V}_l,$$

where $\tilde{V}_m^K$ are some constants. Next, for $k = m-1$, we get (29), where $\tilde{V}_k$ is a constant for all $i_{k,0}$ and $\mu_k^\dagger = \mu_{K-1}^\dagger$. Therefore, using backward induction, for all $1 \le k < K$, we have that $\mu_k^\dagger = \mu^\dagger$, where $\mu^\dagger$ is independent of $i_{k,0}$ and is given by

$$\mu^\dagger = \underset{\mu \in \mathcal{U}}{\operatorname{argmin}} \Big\{ 2\alpha_X(\mu) + \beta_X(\mu) + \gamma_X(\mu) \Big\}. \tag{30}$$

Further, we have

$$\begin{aligned} V_1(\check{x}_0) = J_K(\check{x}_0) &= \min_{\mu_1^K \in \mathcal{U}^K} \left\{ \mathbb{E}_{\mu_1^K} \left[\sum_{k=1}^K c_k(\check{x}_{k-1}, \mu_k) \right] \right\} \\ &= \check{x}_0 + K \cdot \min_{\mu \in \mathcal{U}} \Big\{ 2\alpha_X(\mu) + \beta_X(\mu) + \gamma_X(\mu) \Big\}. \end{aligned}$$

That is, for any arbitrary $K > 1$, we have

$$\begin{aligned} &\min_{\mu_1^K \in \mathcal{U}^K} \left\{ \frac{1}{K} \mathbb{E}_{\mu_1^K} \left[\sum_{k=1}^K c_k(\check{x}_{k-1}, \mu_k) \right] \right\} \\ &= \frac{\check{x}_0}{K} + \min_{\mu \in \mathcal{U}} \Big\{ 2\alpha_X(\mu) + \beta_X(\mu) + \gamma_X(\mu) \Big\}. \end{aligned}$$

Thus, the minimum expected PAoI at the limit $K \to \infty$ is given by

$$\begin{aligned} \zeta^\dagger &= \min_{\mu_1^K \in \mathcal{U}^K} \left\{ \lim_{K \to \infty} \frac{1}{K} \mathbb{E}_{\mu_1^K} \left[\sum_{k=1}^K c_k(\check{x}_{k-1}, \mu_k) \right] \right\} \\ &= \lim_{K \to \infty} \left[\frac{\check{x}_0}{K} + \min_{\mu \in \mathcal{U}} \Big\{ 2\alpha_X(\mu) + \beta_X(\mu) + \gamma_X(\mu) \Big\} \right] \\ &= 2\alpha_X(\mu^\dagger) + \beta_X(\mu^\dagger) + \gamma_X(\mu^\dagger). \end{aligned} \tag{31}$$

Hence, the optimal policy $\boldsymbol{s}^\dagger$ that minimizes $\mathcal{P}$ among $\mathcal{S}_{\mathrm{T}}$ specifies $\mu^\dagger$ at each update, independent of the current information, i.e., $\boldsymbol{s}^\dagger \in \mathcal{S}_{\mathrm{TR}}$.

Step 2: In the following, we drop the index k and ignore the information $I_{k,0}$, as the optimal policy $\boldsymbol{s}^\dagger$ is invariant with respect to k and $I_{k,0}$. Here, we solve (30) by changing the decision epochs of the MDP problem to the times at which requests are sent between any two successive updates. Let $I'_r = \{\theta_1, \dots, \theta_{r-1}\}$ denote the causal information sequence at rth request after an update and c' denotes the cost defined as

$$\begin{aligned} c'(\theta_r) =& 2\int_0^{\theta_r} x f_X(x)dx + \theta_r \bar{F}_X(\theta_r) \\ &+ F_X(\eta_r)d + \int_{\eta_r}^{\theta_r} (\theta_r - x) f_X(x)dx, \end{aligned}$$

such that, for any $\mu \in \mathcal{U}$, we have

$$\begin{aligned} \zeta(\mu) &= 2\alpha_X(\mu) + \beta_X(\mu) + \gamma_X(\mu) \\ &= \mathbb{E}_\mu \left[\sum_{r=1}^\infty \prod_{m=1}^{r-1} \bar{F}_X(\theta_m) c'(\theta_r) \right]. \end{aligned}$$

Let $\omega = \{\theta_i | i \in \mathbb{N}\}$ be a realization of μ for which, we have the sequence $\{J_r\}$ defined by

$$J_r = \prod_{m=1}^{r-1} \bar{F}_X(\theta_m) c'(\theta_r). \tag{32}$$

The derivative of $c'(\theta_r)$ with respect to θ_r given by

$$\frac{dc'(\theta_r)}{d\theta_r} = \theta_r f_X(\theta_r) + \int_0^{\theta_r} f_X(x)dx - \int_{\eta_r}^{\theta_r} f_X(x)dx$$

is non-negative since $0 \le \eta_r \le \theta_r$. Hence, $c'(\theta_r)$ is an increasing function of θ_r. Further, for all $r \ge 1$, we have $\theta_r \in [\theta_{\min}, \theta_{\max}]$, where $\theta_{\min} = x_{\min} + \epsilon$, $\epsilon > 0$. That is, we have $0 \le c'(\theta_r) \le c'(\theta_{\max}) < \infty$. Further, we have $0 \le \bar{F}_X(\theta_r) < 1$ for all $r \ge 1$. Therefore, $J_r \to 0$ as $r \to \infty$ and consequently, for a sufficiently large R, we have

$$\sum_{r=R+1}^\infty J_r \approx 0. \tag{33}$$

Let $\zeta_R^\dagger$ be the minimum expected cumulative cost over the finite horizon $[1, \cdots, R]$, which is given by

$$\zeta_R^\dagger = \min_{\delta_1^R \in \mathcal{F}^R} \left\{ \mathbb{E}_{\delta_1^R} \left[\sum_{r=1}^R \prod_{m=1}^{r-1} \bar{F}_X(\theta_m) c'(\theta_r) \right] \right\}. \tag{34}$$

Similar to Step 1, the optimal solution to (34) can be obtained using the backward recursion of the stochastic Bellman's dynamic programming [15] given by

$$\zeta_r(i'_r) = \min_{\delta_r \in \mathcal{F}} \left\{ \mathbb{E}_{\delta_r} \left[\prod_{m=1}^{r-1} \bar{F}_X(\theta_m) c'(\theta_r) + \zeta_{r+1}(I'_{r+1}) \right] \right\},$$

where the value function ζ_r denotes the optimal expected cumulative cost-to-go from r to R. As (33) is true for any realization ω of μ, we have $\zeta_{R+1} \approx 0$. Now, for $r = R$,

$$\zeta_R(i'_R) = \prod_{m=1}^{R-1} \bar{F}_X(\theta_m) \underbrace{\min_{\delta_R \in \mathcal{F}} \Big\{ \mathbb{E}_{\delta_r} \Big[c'(\theta_r) \Big] \Big\}}_{\tilde{\zeta}_R}. \tag{35}$$

From (35), it is easy to see that $\tilde{\zeta}_R$ is a constant and the optimal distribution $\delta_R^\dagger$ is independent of i'_R. Next, for some $l > 1$, we assume that the optimal distribution $\delta_l^\dagger$ is independent of i'_l and the value function has the same structure as in (35), that is given by

$$\zeta_l(i'_l) = \prod_{m=1}^{l-1} \bar{F}_X(\theta_m) \times \tilde{\zeta}_l,$$

for some constant $\tilde{\zeta}_l > 0$. Next, for $r = l-1$, we have

$$\zeta_r(i_r) = \prod_{m=1}^{r-1} \bar{F}_X(\theta_m) \underbrace{\min_{\delta_r \in \mathcal{F}} \Big\{ \mathbb{E}_{\delta_r} \Big[c'(\theta_r) + \tilde{\zeta}_l \bar{F}_X(\theta_r) \Big] \Big\}}_{\tilde{\zeta}_r}, \tag{36}$$

where $\tilde{\zeta}_r$ is a constant for all i'_r. Therefore, using backward induction, we have that all $\delta_r^\dagger$ are independent of i'_r, where $r \in [1, \dots, R]$. As the backward induction is true for any

$$V_k(i_{k,0}) = \min_{\mu_k \in \mathcal{U}} \left\{ \check{x}_{k-1} + \alpha_X(\mu_k) + \beta_X(\mu_k) + \gamma_X(\mu_k) + \mathbb{E}_{\mu_k} \left[\check{X}_k + \sum_{l=k+1}^{K} \tilde{V}_l \middle| I_{k,0} = i_{k,0} \right] \right\}$$

$$= \check{x}_{k-1} + \underbrace{\min_{\mu_k \in \mathcal{U}} \left\{ 2\alpha_X(\mu_k) + \beta_X(\mu_k) + \gamma_X(\mu_k) \right\}}_{\tilde{V}_k} + \sum_{l=k+1}^{K} \tilde{V}_l. \quad (29)$$

arbitrarily large R, it is also true for the optimal sub-policy $\mu^\dagger$. Next, we drop i'_r and rewrite (36) in terms of $\tilde{\zeta}_r$ as

$$\tilde{\zeta}_r = \min_{\delta_r \in \mathcal{F}} \left\{ \mathbb{E}_{\delta_r} \left[c'(\theta_r) + \tilde{\zeta}_{r+1} \bar{F}_X(\theta_r) \right] \right\}, \quad (37)$$

Now, let $\theta_r^\dagger$ be given by

$$\theta_r^\dagger = \underset{\theta_r \in [\theta_{\min}, \theta_{\max}]}{\operatorname{argmin}} \left\{ c'(\theta_r) + \tilde{\zeta}_{r+1} \bar{F}_X(\theta_r) \right\}, \quad (38)$$

Here, we denote a deterministic distribution with $\mathbf{1}_\theta$ for which $\mathbb{P}(\theta_r = \theta) = 1$. From (38), at each backward iteration, we have that $\delta_r^\dagger = \mathbf{1}_{\theta_r^\dagger}$ minimizes (37) since, for any $\delta_r \in \mathcal{F}$, we have

$$c'(\theta_r^\dagger) + \tilde{\zeta}_{r+1} \bar{F}_X(\theta_r^\dagger) \le \mathbb{E}_{\delta_r} \left[c'(\theta) + \tilde{\zeta}_{r+1} \bar{F}_X(\theta) \right].$$

Let $T : \mathbb{R}_{\ge 0} \to \mathbb{R}_{\ge 0}$ be the Bellman's operator, given by

$$T(U) = \min_{\theta \in [\theta_{\min}, \theta_{\max}]} \left\{ c'(\theta) + U \bar{F}_X(\theta) \right\}.$$

Using the similar argument as in [15, Theorem 7.6.2], for any U_1 and U_2 in $\mathbb{R}_{\ge 0}$, we have

$$\left| T(U_1) - T(U_2) \right| \le \left| U_1 - U_2 \right| \max_{\theta \in [\theta_{\min}, \theta_{\max}]} \left\{ \bar{F}_X(\theta) \right\}.$$

Therefore, the Bellman's operator forms a contraction mapping for all $\theta \in [\theta_{\min}, \theta_{\max}]$. Using Banach's fixed point theorem, for some $\theta^\dagger \in [\theta_{\min}, \theta_{\max}]$, we have that there exists a unique fixed point $\tilde{\zeta}^\dagger$ to the recursive equation (37). Similar to the case of an infinite horizon discounted cost MDP problem discussed in [15, Theorem 7.6.2], where the conclusion is that a stationary (but state-dependent) policy is optimal for the infinite-horizon, we conclude that using a policy with the fixed-threshold $\theta^\dagger$ at all requests minimizes average PAoI, i.e., there exists a fixed-threshold policy $\boldsymbol{s}_{\theta^\dagger} \in \mathcal{S}_\theta$ which is optimal. Therefore, using Corollary 1, we obtain the optimal $\theta^\dagger$, which is given by

$$\theta^\dagger \triangleq \underset{\theta \in [\theta_{\min}, \theta_{\max}]}{\arg\min}\ \zeta(\boldsymbol{s}_\theta), \quad (39)$$

And, the minimum expected PAoI among $\mathcal{S}_{\mathrm{T}}$ is given by

$$\zeta(\boldsymbol{s}_{\theta^\dagger}) = \frac{1}{F_X(\theta^\dagger)} \times \left[2\int_0^{\theta^\dagger} x f_X(x)dx + \theta^\dagger \bar{F}_X(\theta^\dagger) + F_X(\theta^\dagger - d)d + \int_{\theta^\dagger - d}^{\theta^\dagger} (\theta^\dagger - x) f_X(x)dx \right].$$

F. Proof of Lemma 4

Recall that, for a work-conserving policy $G(\boldsymbol{s}) \le d$. Since $\boldsymbol{s}_\theta$ is a work-conserving policy, we have

$$\mathbb{E}[G(\boldsymbol{s}_\theta)] \le d. \quad (40)$$

Now, consider that there exists θ such that

$$\begin{aligned}
&\mathbb{E}[X] < \mathbb{E}[X - \theta | X > \theta] + \frac{\theta}{2} \\
&\overset{(a)}{\Leftrightarrow} \mathbb{E}[X] + \frac{\theta}{2} < \frac{\int_\theta^\infty x f_X(x)dx}{\mathbb{P}(X > \theta)} \\
&\Leftrightarrow 2\mathbb{P}(X > \theta)\mathbb{E}[X] + \theta\mathbb{P}(X > \theta) < 2\int_\theta^\infty x f_X(x)dx \\
&\Leftrightarrow 2(1 - F_X(\theta))\mathbb{E}[X] + \theta\mathbb{P}(X > \theta) < 2(\mathbb{E}[X] - \int_0^\theta x f_X(x)dx) \\
&\Leftrightarrow 2\int_0^\theta x f_X(x)dx + \theta\mathbb{P}(X > \theta) < 2\mathbb{E}[X] F_X(\theta) \\
&\Leftrightarrow 2\mathbb{E}[\check{X}(\boldsymbol{s}_\theta)] + \frac{\theta\mathbb{P}(X > \theta)}{F_X(\theta)} < 2\mathbb{E}[X] \\
&\overset{(b)}{\Longrightarrow} \mathbb{E}[G(\boldsymbol{s}_\theta)] + 2\mathbb{E}[\check{X}(\boldsymbol{s}_\theta)] + \frac{\theta\mathbb{P}(X > \theta)}{F_X(\theta)} < 2\mathbb{E}[X] + d \\
&\overset{(c)}{\Leftrightarrow} \zeta(\boldsymbol{s}_\theta) < 2\mathbb{E}[X] + d.
\end{aligned}$$

In step (a) we have used the following equation.

$$\mathbb{E}[X - \theta | X > \theta] = \frac{\int_\theta^\infty (x - \theta) f_X(x)dx}{\mathbb{P}(X > \theta)} = \frac{\int_\theta^\infty x f_X(x)dx}{\mathbb{P}(X > \theta)} - \theta.$$

In step (b), we have used (40), and step (c) follows from Corollary 1.

References

[1] S. Kaul, M. Gruteser, V. Rai, and J. Kenney, "Minimizing age of information in vehicular networks," in *Proc. IEEE SECON*, 2011.

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

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

[4] B. T. Bacinoglu, E. T. Ceran, and E. Uysal-Biyikoglu, "Age of information under energy replenishment constraints," in *Proc. Information Theory and Applications Workshop (ITA)*, 2015.

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

[6] J. P. Champati, H. Al-Zubaidy, and J. Gross, "Statistical guarantee optimization for aoi in single-hop and two-hop FCFS systems with periodic arrivals," *IEEE Transactions on Communications*, vol. 69, no. 1, pp. 365–381, 2021.

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

[8] R. D. Yates, "Lazy is timely: Status updates by an energy harvesting source," in *Proc. IEEE ISIT*, 2015.

[9] Y. Sun, E. Uysal-Biyikoglu, R. D. Yates, C. E. Koksal, and N. B. Shroff, "Update or wait: How to keep your data fresh," *IEEE Transactions on Information Theory*, vol. 63, no. 11, pp. 7492–7508, Nov 2017.

[10] A. Arafa, R. D. Yates, and H. V. Poor, "Timely cloud computing: Preemption and waiting," in *Proc. Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, 2019, pp. 528–535.

[11] M. Costa, M. Codreanu, and A. Ephremides, "On the age of information in status update systems with packet management," *IEEE Transactions on Information Theory*, vol. 62, no. 4, pp. 1897–1910, April 2016.

[12] Q. He, D. Yuan, and A. Ephremides, "On optimal link scheduling with min-max peak age of information in wireless systems," in *Proc. IEEE ICC*, May 2016, pp. 1–7.

[13] C. Xu, H. H. Yang, X. Wang, and T. Q. S. Quek, "On peak age of information in data preprocessing enabled IoT networks," *CoRR*, vol. abs/1901.09376, 2019.

[14] J. P. Champati, R. R. Avula, T. J. Oechtering, and J. Gross, "On the minimum achievable age of information for general service-time distributions," in *Proc. IEEE INFOCOM*, 2020, pp. 456–465.

[15] V. Krishnamurthy, *Partially Observed Markov Decision Processes*. Cambridge University Press, 2016.

[16] M. Luby, A. Sinclair, and D. Zuckerman, "Optimal speedup of Las Vegas algorithms," *Information Processing Letters*, vol. 47, pp. 173–180, 1993.

[17] S. Kaul, R. Yates, and M. Gruteser, "Status updates through queues," in *Proc. Conference on Information Sciences and Systems (CISS)*, 2012.

[18] K. Chen and L. Huang, "Age-of-information in the presence of error," *CoRR*, vol. abs/1605.00559, 2016.

[19] E. Najm and R. Nasser, "Age of information: The gamma awakening," in *Proc. ISIT*, July 2016, pp. 2574–2578.

[20] E. Najm, R. D. Yates, and E. Soljanin, "Status updates through M/G/1/1 queues with HARQ," in *Proc. IEEE ISIT*, June 2017, pp. 131–135.

[21] Y. Inoue, H. Masuyama, T. Takine, and T. Tanaka, "A general formula for the stationary distribution of the age of information and its application to single-server queues," *CoRR*, vol. abs/1804.06139, 2018.

[22] A. Soysal and S. Ulukus, "Age of information in G/G/1/1 systems: Age expressions, bounds, special cases, and optimization," *CoRR*, vol. abs/1905.13743, 2019.

[23] J. P. Champati, H. Al-Zubaidy, and J. Gross, "On the distribution of AoI for the GI/GI/1/1 and GI/GI/1/2* systems: Exact expressions and bounds," in *Proc. IEEE INFOCOM*, April 2019, pp. 37–45.

[24] Y. Sun, I. Kadota, R. Talak, E. Modiano, and R. Srikant, *Age of Information: A New Metric for Information Freshness*. Morgan & Claypool, 2019.

[25] J. P. Champati, H. Al-Zubaidy, and J. Gross, "Statistical guarantee optimization for age of information for the D/G/1 queue," in *Proc. IEEE INFOCOM Workshop*, April 2018, pp. 130–135.

[26] E. Najm and E. Telatar, "Status updates in a multi-stream M/G/1/1 preemptive queue," *CoRR*, vol. abs/1801.04068, 2018.

[27] A. Soysal and S. Ulukus, "Age of information in G/G/1/1 systems," *CoRR*, vol. abs/1805.12586, 2018.

[28] R. Talak, S. Karaman, and E. Modiano, "Minimizing age-of-information in multi-hop wireless networks," in *Proc. Allerton Conference on Communication, Control, and Computing*, Oct 2017, pp. 486–493.

[29] S. Farazi, A. G. Klein, and D. R. Brown, "Fundamental bounds on the age of information in general multi-hop interference networks," in *Proc. IEEE INFOCOM Workshop*, April 2019, pp. 96–101.

[30] B. T. Bacinoglu, Y. Sun, E. Uysal, and V. Mutlu, "Optimal status updating with a finite-battery energy harvesting source," *Journal of Communications and Networks*, vol. 21, no. 3, pp. 280–294, June 2019.

[31] 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, 2019.

[32] S. Farazi, A. G. Klein, and D. Richard Brown, "Average age of information in multi-source self-preemptive status update systems with packet delivery errors," in *Proc. Asilomar Conference on Signals, Systems, and Computers*, 2019, pp. 396–400.

[33] J. Xu, I.-H. Hou, and N. Gautam, "Status updates in a multi-stream M/G/1/1 preemptive queue," *CoRR*, vol. abs/2004.11847, 2020.

[34] V. Kavitha, E. Altman, and I. Saha, "Controlling packet drops to improve freshness of information," *CoRR*, vol. abs/1807.09325, 2018.

[35] B. Wang, S. Feng, and J. Yang, "When to preempt? age of information minimization under link capacity constraint," *Journal of Communications and Networks*, vol. 21, no. 3, pp. 220–232, 2019.

[36] R. Wang, Y. Gu, H. Chen, Y. Li, and B. Vucetic, "On the age of information of short-packet communications with packet management," in *Proc. IEEE GLOBECOM*, 2019, pp. 1–6.

[37] Y. Sun, E. Uysal-Biyikoglu, and S. Kompella, "Age-optimal updates of multiple information flows," in *Proc. IEEE INFOCOM Workshop*. IEEE, 2018, pp. 136–141.

[38] A. P. A. van Moorsel and K. Wolter, "Analysis of restart mechanisms in software systems," *IEEE Transactions on Software Engineering*, vol. 32, no. 8, pp. 547–558, Aug 2006.

Jaya Prakash Champati received his bachelor of technology degree from the National Institute of Technology Warangal, India in 2008, and master of technology degree from the Indian Institute of Technology (IIT) Bombay, India in 2010. He received his PhD in Electrical and Computer Engineering from the University of Toronto, Canada in 2017. From 2017-2020, he was a post-doctoral researcher with the division of Information Science and Engineering, EECS, KTH Royal Institute of Technology, Sweden. He is currently a Research Assistant Professor at IMDEA Networks Institute, Madrid, Spain. His general research interest is in the design and analysis of algorithms for scheduling problems that arise in networking and information systems. Currently, his focus is on freshness (Age of Information) and delay analysis for time-critical control applications in Cyber-Physical Systems (CPS) and the Internet of Things (IoT), and computational offloading in edge computing systems. Prior to joining PhD he worked at Broadcom Communications, where he was involved in developing the LTE MAC layer. He was a recipient of the best paper award at IEEE National Conference on Communications, India, 2011.

Ramana R. Avula (S'18) received his bachelor and master of technology degrees in Electrical Engineering from the Indian Institute of Technology, Madras, India in 2015. From 2015 to 2017, he was a Senior Engineer at the Robert Bosch Engineering and Business Solutions, India. Since 2017, he is pursuing his Ph.D. degree at the Division of Information Science and Engineering, KTH Royal Institute of Technology, Sweden. His current research interests include statistical signal processing and inference with emphasis on consumer privacy for smart metering systems.

Tobias J. Oechtering (S'01-M'08-SM'12) received his Dipl-Ing degree in Electrical Engineering and Information Technology in 2002 from RWTH Aachen University, Germany, his Dr-Ing degree in Electrical Engineering in 2007 from the Technische Universität Berlin, Germany. In 2008 he joined KTH Royal Institute of Technology, Stockholm, Sweden and has been a Professor since 2018. In 2009, he received the "Förderpreis 2009" from the Vodafone Foundation.

Dr. Oechtering is currently Senior Editor of IEEE Transactions on Information Forensic and Security since May 2020 and served previously as Associate Editor for the same journal since June 2016, and IEEE Communications Letters during 2012-2015. He has served on numerous technical program committees for IEEE sponsored conferences, and he was general co-chair for IEEE ITW 2019. His research interests include communication and information theory, communication for networked control, statistical signal processing, and physical layer privacy and security.

James Gross received his Ph.D. degree from TU Berlin in 2006. From 2008-2012 he was Assistant Professor and head of the Mobile Network Performance Group at RWTH Aachen University, as well as a member of the DFG-funded UMIC Research Centre of RWTH. Since November 2012, he has been with the Electrical Engineering and Computer Science School, KTH Royal Institute of Technology, Stockholm, where he is professor for machine-to-machine communications. He served as Director for the ACCESS Linnaeus Centre from 2016 to 2019, while he is currently a member of the board of KTHs Innovative Centre for Embedded Systems. His research interests are in the area of mobile systems and networks, with a focus on critical machine-to-machine communications, cellular networks, resource allocation, as well as performance evaluation methods. He has authored about 150 (peer-reviewed) papers in international journals and conferences. His work has been awarded multiple times, including the Best Paper Award at ACM MSWiM 2015, the Best Demo Paper Award at IEEE WoWMoM 2015, the Best Paper Award at IEEE WoWMoM 2009, and the Best Paper Award at European Wireless 2009. In 2007, he was the recipient of the ITG/KuVS dissertation award for his Ph.D. thesis. He is also co-founder of R3 Communications GmbH, a Berlin-based start-up in the area of ultrareliable low-latency wireless networking for industrial automation.