Timely Data Delivery for Heterogeneous IoT Applications

Verónica Toro-Betancur*†, Gopika Premsankar†, Lorenzo Corneo†, and Mario Di Francesco†
*Nokia Bell Labs, Finland †Aalto University, Finland

***Abstract*—Internet of Things applications require timely access to information collected from sensors deployed over large geographic areas. However, such applications often experience highly-varying network conditions that prevent the timely delivery of information updates related to source data from sensors. Moreover, IoT applications have different metrics of interest and patterns to request source data. This article explicitly addresses the timely delivery of information updates in heterogeneous IoT scenarios with different application-specific goals. For this purpose, it introduces new metrics based on age of information (AoI) to accurately describe timeliness of updates in such a context. Moreover, it analytically derives optimal update generation policies for different request patterns to minimize the overall update age in an IoT system and maximize fairness of updates. Finally, it carries out a thorough performance evaluation of the proposed policies for representative request patterns with a real-world dataset of Internet connectivity. The obtained results demonstrate that the proposed policies are competitive with those in the state of the art, with a two order of magnitude reduction in energy consumption and up to a 19.9% higher fairness.**

***Index Terms*—Information freshness; timeliness; data delivery; Internet of Things; probabilistic modeling**

I. Introduction

Modern applications rely on timely data provided by information sources distributed on a large scale and accessed over the Internet. Representative examples include cyber-physical systems [1] and different use cases in the Internet of Things (IoT) ranging from smart energy to intelligent transportation systems [2]. Information sources are primarily represented by sensors or servers providing data on their behalf, for instance, in the cloud or at the edge of the network [3, 4]. Data requested by applications dynamically change over time, thus, it is essential that updates are quickly and efficiently delivered to their destination or client.

Timely data delivery is affected by several factors. A very important one is freshness, defined by the time at which the data were generated at the source [5–7]. Age of Information (AoI) [8] is one of the most widely used metrics to characterize freshness and has received large attention in the literature [9] for diverse scenarios ranging from vehicular networks and video streaming [10, 11] to energy-harvesting information sources and IoT applications [12, 13]. However, freshness alone is not enough to characterize timely data delivery in IoT scenarios [14]. In fact, it is imperative to also consider network-specific factors including the delay – namely, the time taken by a message to traverse the network from source to destination – and the communication reliability [15]. Both are especially critical for heterogeneous IoT applications characterized by diverse network conditions and data request patterns, which depend on different use cases [16].

A large share of the literature has focused on freshness (Section V). Most works aimed at minimizing the average AoI experienced in a network [17–20], which is generally assumed to be under the control of the designer [9]. In contrast, modern applications have limited to no control on the underlying communication systems, especially in the Internet. Moreover, optimizing for AoI has mostly considered strict simplifying assumptions on the network; as a result, the related solutions have limited applicability to real-world scenarios [15]. This is especially true for large-scale IoT deployments characterized by heterogeneous network connectivity and traffic patterns [21].

This work explicitly addresses these limitations by targeting timely data delivery for diverse classes of applications in heterogeneous IoT environments. Different from the state of the art, it focuses on freshness as experienced by IoT applications when information updates become available to them. For this purpose, it **introduces novel metrics that accurately describe timeliness in heterogeneous IoT scenarios** by accounting not only for freshness, but also for the network delay and the communication reliability (Section II). Moreover, it **analytically derives optimal data generation policies for different request patterns** to minimize the overall update age in an IoT system and maximize fairness (Section III). Finally, it carries out **a thorough performance evaluation of several data generation policies with a real-world dataset of Internet connectivity** (Section IV). The obtained results demonstrate that the proposed policies are competitive with those in the state of the art in terms of update age, with a two order of magnitude reduction in energy consumption and up to a 19.9% higher fairness.

II. Background

This section first introduces the system model and then different metrics suitable to characterize timely data delivery for different classes of IoT applications.

A. System Model

The considered system comprises a *server* providing updates to a set $C = \{1, 2, ..., N\}$ of N geographically-distributed *clients* (Figure 1). In this context, the server is the endpoint for accessing information sources such as sensor devices in IoT deployments. Conversely, clients are the software components of an IoT application which obtain updates from

the (edge or cloud) server over the Internet. Clients experience different network delays that depend on their location, specifically, their distance from the server. In particular, a client i experiences a one-way delay d_i to reach the server. The system follows a request-response pattern: clients explicitly request information from the server, which then replies with the corresponding data. The server makes information available over time in discrete steps simply referred to as updates.

Different from existing literature [8, 9], we characterize timeliness as experienced by clients when information updates become available to them. This entails considering both the network delay and the message loss probability. Accordingly, the timeliness associated with an update from the server is characterized by the *effective update age*[1] at time t, defined as:

$$\Delta_i(t) = t - G(t) + d_i, \tag{1}$$

where $G(t)$ is the generation time of the last update at the server. As a consequence, the average update age experienced by client i is given by:

$$\overline{\Delta}_i = \lim_{T \to +\infty} \frac{1}{T} \int_0^T \Delta_i(t)\, dt. \tag{2}$$

Each request by client i may not be correctly received by the server; such unreliability is characterized in terms of the probability α_i that the corresponding message is correctly received. The client is assumed to adopt an error recovery mechanism based on acknowledgments and retransmissions [15]. Specifically, the server generates a new update as soon as it correctly receives a re-transmitted request to reduce the update age in presence of dropped requests [13].

B. Application-specific Age Metrics

The previous discussion characterized the update age associated with multiple requests sent by an individual client to the server. The following introduces different metrics that describe the update age in the entire network according to application-specific requirements.

We introduce the *system update age* to characterize the overall update age based on the instantaneous values of the individual clients, as:

$$\overline{\Delta} = \frac{1}{N} \sum_{i \in C} \overline{\Delta}_i \alpha_i + \tilde{\Delta}_i \left(1 - \alpha_i\right). \tag{3}$$

The system update age considers all clients as equally contributing to timely information delivery, as long as the corresponding data successfully reaches the server. Indeed, the first term in the right-hand side of the equation explicitly refers to this aspect by means of the packet success probability (α_i) introduced in the previous section. Moreover, $\tilde{\Delta}_i$ denotes the delay experienced by client i when the first request is dropped and the update is retransmitted; i.e., $\tilde{\Delta}_i = 2d_i + x_i$, where x_i is a fixed waiting time before the server retransmits the update.

The system update age is defined in terms of the average values experienced by individual clients. However, the update age of the clients in the system varies: those with a reliable and low-latency connection receive more timely updates with respect to the others. Indeed, certain applications are sensitive to how the update age is spread across the clients [22]. For this reason, we additionally define the *update age fairness* based on Jain's fairness index [23] as:

$$\sigma = \frac{\left(\sum_{i \in C} \overline{\Delta}_i \alpha_i + \tilde{\Delta}_i \left(1 - \alpha_i\right)\right)^2}{N \sum_{i \in C} \left(\overline{\Delta}_i \alpha_i + \tilde{\Delta}_i \left(1 - \alpha_i\right)\right)^2}, \tag{4}$$

The value of $\sigma \in [0, 1]$ expresses how the update age is balanced among all clients, the higher the better. Note that there is an inherent trade-off between system update age and fairness. Optimal fairness implies maximizing σ; one option is to artificially increase the update age experienced by all the clients to match the highest in the system. This, in turn, dramatically increases the system update age.

The rest of this article proposes and evaluates policies that address different requirements of IoT applications, including to minimize system update age and to maximize fairness.

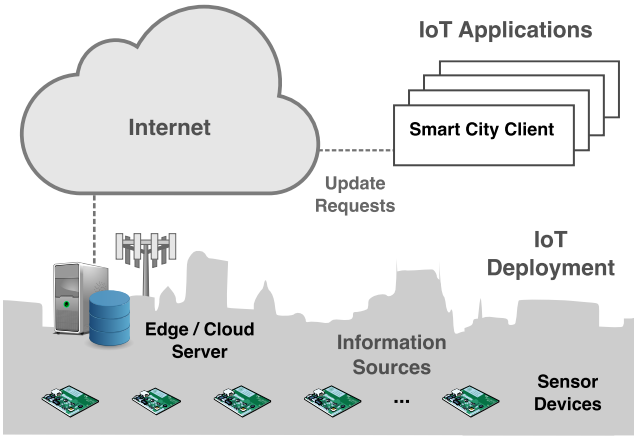

Fig. 1: System model.

III. Optimal Application-specific Update Age

This section analytically derives optimal policies for both system update age and fairness. For this purpose, a probabilistic characterization is derived next based on the distribution of the requests from clients. Specifically, the rest of this section considers special cases of particular interest for the distributions of the interarrival times, selected to cover a variety of use cases and according to existing studies [16, 24]. These distributions are: *uniform*, *exponential* and *normal*.

A. Minimizing System Update Age

Let $p_i(t)$ denote the probability density function (PDF) for the interarrival times of requests from client i. The expected value of update age in the interval $[t_l, t_r]$ is then given by:

$$\begin{aligned} M(t_l, t_r) &= \sum_{i=1}^{N} \int_{t_l}^{t_r} (t - t_l + d_i)\, p_i(t) \alpha_i\, dt \\ &= \sum_{i=1}^{N} \left[\int_{t_l}^{t_r} t p_i(t) dt + (d_i - t_l)\left(\theta_i(t_r) - \theta_i(t_l)\right) \right] \alpha_i, \end{aligned} \tag{5}$$

[1] This metric is simply called update age for brevity in the rest of the paper.

for the case where the last update is generated at time t_l, where $\theta_i(t)$ is the cumulative distribution function (CDF) corresponding to the PDF $p_i(t)$. In other words, Eq. (5) denotes the expected update age obtained by clients whose requests arrive in the interval $[t_l, t_r]$. Note that the equation does not exactly correspond to the system update age, as it does not involve $\tilde{\Delta}_i$. This makes no difference for the purpose of the optimization as α_i is given, therefore, it is enough to consider only the first term in Eq. (3) – namely, $\sum_{i\in C} \overline{\Delta}_i \alpha_i$.

Our formulation relies on $p_i(t)$ being available at the server $\forall i \in C$. In practice, such a distribution can be easily estimated by the server through a statistical test – such as the Kolmogorov-Smirnov test – after receiving enough requests.

Analytical expressions of $M(t_l, t_r)$ for the considered distributions are derived first, then an unconstrained optimization problem is solved to find the time t_r (i.e., $t_r = G_k(t)$) at which updates must be generated to minimize the expected update age ($\min M(t_l, t_r)$) for each distribution.

1) Uniform Distribution: The PDF and CDF for clients sending requests according to a uniform distribution over an interval $[t_i^-, t_i^+]$ are given by

$$p_i^u(t) = \begin{cases} \frac{1}{t_i^+ - t_i^-}, & t_i^- \le t \le t_i^+ \\ 0, & \text{elsewhere} \end{cases}$$

and

$$\theta_i^u(t) = \begin{cases} 0, & t < t_i^- \\ \frac{t - t_i^-}{t_i^+ - t_i^-}, & t_i^- \le t \le t_i^+ \\ 1, & t > t_i^+ \end{cases},$$

respectively. Then, it is:

$$\begin{aligned} &M_u(t_l, t_r) \\ &= \sum_{i=1}^{N_u} \left[\int_{t_{i,-}^*}^{t_{i,+}^*} \frac{t}{t_i^+ - t_i^-} dt + (d_i - t_l)\left(\theta_i^u(t_r) - \theta_i^u(t_l)\right) \right] \alpha_i \\ &= \sum_{i=1}^{N_u} \left[\frac{(t_{i,+}^*)^2 - (t_{i,-}^*)^2}{2(t_i^+ - t_i^-)} + (d_i - t_l)\left(\theta_i^u(t_r) - \theta_i^u(t_l)\right) \right] \alpha_i \end{aligned}$$

where N_u is the number of clients issuing requests with a uniform distribution, $t_{i,-}^* = \max(t_l, t_i^-)$, and $t_{i,+}^* = \min(t_r, t_i^+)$.

The following three cases arise:

1) $t_{i,+}^* = t_i^+$ and $t_{i,-}^* = t_i^- \quad \forall i \in \mathcal{C}$, i.e., there is a probability of one that the request from the client arrives in the interval $[t_l, t_r]$;
2) $t_{i,+}^* = t_r$ and $t_{i,-}^* = t_l \quad \forall i \in \mathcal{C}$;
3) $t_{i,+}^* = t_r$ and $t_{i,-}^* = t_i^- \quad \forall i \in \mathcal{C}$.

Clearly, the first case does not depend on t_r and can be disregarded; moreover, the last two cases are equivalent. Therefore, we only consider the last one, in which

$$\theta_i^u(t_r) = \frac{t_r - t_i^-}{t_i^+ - t_i^-} \quad \text{and} \quad \theta_i^u(t_l) = 0.$$

leading to:

$$M_u(t_l, t_r) = \sum_{i=1}^{N_u} \left[\frac{t_r^2 - {t_i^-}^2}{2(t_i^+ - t_i^-)} + (d_i - t_l)\left(\frac{t_r - t_i^-}{t_i^+ - t_i^-}\right) \right] \alpha_i$$

To minimize the expected update age, we first obtain the derivative of the previous equation with respect to t_r:

$$\begin{aligned} \frac{\partial M_u(t_l, t_r)}{\partial t_r} &= \sum_{i=1}^{N_u} \left[\frac{2t_r}{2(t_i^+ - t_i^-)} + \frac{d_i - t_l}{t_i^+ - t_i^-} \right] \alpha_i \\ &= \sum_{i=1}^{N_u} \left[\frac{t_r - t_l + d_i}{t_i^+ - t_i^-} \right] \alpha_i. \end{aligned}$$

and then setting it to zero, leading to:

$$t_r = \frac{\sum_{i=1}^{N_u} \left[\frac{t_l - d_i}{t_i^+ - t_i^-} \right] \alpha_i}{\sum_{i=1}^{N_u} \frac{\alpha_i}{t_i^+ - t_i^-}} \tag{6}$$

2) Exponential Distribution: For requests whose interarrival times follow an exponential distribution, the PDF and CDF are:

$$p_i^e(t) = \lambda_i e^{-\lambda_i t}, \ \theta_i^e(t) = 1 - e^{-\lambda_i t},$$

where λ_i is the average rate, at which client i sends requests. Accordingly, the expected update age is given by:

$$\begin{aligned} M_e(t_l, t_r) = \sum_{i=1}^{N_e} \Bigg[& t_l e^{-\lambda_i t_l} - t_r e^{-\lambda_i t_r} \\ & + \left(e^{-\lambda_i t_l} - e^{-\lambda_i t_r}\right)\left(d_i + \frac{1}{\lambda_i}\right) \Bigg] \alpha_i \end{aligned} \tag{7}$$

where N_e is the number of clients sending requests according to an exponential distribution. The derivative of Eq. (7) with respect to t_r is:

$$\frac{\partial M_e(t_l, t_r)}{\partial t_r} = \sum_{i=1}^{N_e} \left[\lambda_i \left(t_r + d_i\right) e^{-\lambda_i t_r} \right] \alpha_i$$

For analytical tractability, we approximate[2] the term $e^{-\lambda_i t_r}$ with a second-order Taylor polynomial, leading to:

$$\begin{aligned} \frac{\partial M_e(t_l, t_r)}{\partial t_r} &\approx \sum_{i=1}^{N_e} \left[t_r \lambda_i - t_r^2 \lambda_i^2 + \lambda_i d_i - t_r \lambda_i^2 d_i \right] \alpha_i \quad \text{(8a)} \\ &\approx A_e t_r^2 + B_e t_r + C_e \quad \text{(8b)} \end{aligned}$$

where:

$$A_e = -\sum_{i=1}^{N_e} \lambda_i^2 \alpha_i, \ B_e = \sum_{i=1}^{N_e} (\lambda_i - \lambda_i^2 d_i)\alpha_i, \ C_e = \sum_{i=1}^{N_e} \lambda_i d_i \alpha_i.$$

Here, t_r can be easily found by setting Eq. (8b) equal to zero and solving the quadratic polynomial.

[2]Low-order approximation is feasible since λ_i and t_r are both small, i.e., in the order of hundreds of milliseconds.

3) Normal Distribution: Finally, requests following a normal distribution have the following PDF and CDF:

$$p_i^n(t) = \frac{1}{\sigma_i\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{t-\mu_i}{\sigma_i}\right)^2},\ \theta_i^n(t) = \frac{1}{2}\left[1 + \text{erf}\left(\frac{t-\mu_i}{\sigma_i\sqrt{2}}\right)\right]$$

respectively, where μ_i and σ_i are the mean and standard deviation for client i and erf$(\cdot)$ is the error function. In this case, the expected update age is:

$$\begin{aligned} &M_n(t_l, t_r) \\ &= \sum_{i=1}^{N_n}\left[\int_{t_l}^{t_r} \frac{t}{\sigma_i\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{t-\mu_i}{\sigma_i}\right)^2} dt \right. \\ &\quad \left. + \frac{d_i - t_l}{2}\left(1 + \text{erf}\left(\frac{t_r-\mu_i}{\sigma_i\sqrt{2}}\right) - 1 - \text{erf}\left(\frac{t_l-\mu_i}{\sigma_i\sqrt{2}}\right)\right)\right]\alpha_i \\ &= \sum_{i=1}^{N_n}\left[\frac{\sigma_i}{\sqrt{2\pi}}\left(e^{-\frac{1}{2}\left(\frac{t_l-\mu_i}{\sigma_i}\right)^2} - e^{-\frac{1}{2}\left(\frac{t_r-\mu_i}{\sigma_i}\right)^2}\right)\right. \\ &\quad \left. + \frac{\mu_i + d_i - t_l}{2}\left(\text{erf}\left(\frac{\mu_i - t_l}{\sigma_i\sqrt{2}}\right) - \text{erf}\left(\frac{\mu_i - t_r}{\sigma_i\sqrt{2}}\right)\right)\right]\alpha_i \end{aligned}$$

where N_n is the number of clients whose requests have inter-arrival times following a normal distribution. The derivative of $M_n(t_l, t_r)$ with respect to t_r is:

$$\frac{\partial M_n(t_l,t_r)}{\partial t_r} = \left[e^{\frac{1}{2}\left(\frac{t_r-\mu_i}{\sigma_i}\right)^2}\left[\frac{1}{\sqrt{2\pi}}\left(t_r - \frac{\mu_i}{2}\right) + \frac{\mu_i - d_i - t_l}{\sigma_i\sqrt{2}}\right]\right]\alpha_i$$

For analytical tractability, we approximate the exponential term $e^{\frac{1}{2}\left(\frac{t_r-\mu_i}{\sigma_i}\right)^2}$ with a first-order Taylor polynomial:

$$\begin{aligned} \frac{\partial M_n(t_l,t_r)}{\partial t_r} &\approx \left[\left(1 - \frac{1}{2}\left(\frac{t_r-\mu_i}{\sigma_i}\right)^2\right)\right. \\ &\quad \left.\left[\frac{1}{\sqrt{2\pi}}\left(t_r - \frac{\mu_i}{2}\right) + \frac{\mu_i - d_i - t_l}{\sigma_i\sqrt{2}}\right]\right]\alpha_i \qquad (9a) \\ &\approx A_n t_r^3 + B_n t_r^2 + C_n t_r + D_n \qquad (9b) \end{aligned}$$

where

$$\begin{aligned} A_n &= -\sum_{i=1}^{N_n}\frac{\alpha_i}{2\sigma_i^2\sqrt{2\pi}} \\ B_n &= \sum_{i=1}^{N_n}\frac{\alpha_i}{2\sigma_i^2\sqrt{2\pi}}\left(2 + \frac{\mu_i}{2} + \frac{\sqrt{\pi}(\mu_i - d_i - t_l)}{\sigma_i}\right) \\ C_n &= \sum_{i=1}^{N_n}\frac{\alpha_i}{2\sigma_i^2\sqrt{2\pi}}\left(2\sigma_i^2 - \mu_i - \mu_i^2 + \frac{2\sqrt{\pi}(\mu_i - d_i - t_l)}{\sigma_i}\right) \\ D_n &= \sum_{i=1}^{N_n}\alpha_i\left[\frac{\mu_i^3}{4\sigma_i^2\sqrt{2\pi}} + \frac{\mu_i(\mu_i - d_i - t_l)}{2\sigma_i^3\sqrt{2}} - \frac{\mu_i}{2\sqrt{2\pi}} + \frac{\mu_i - d_i - t_l}{\sigma_i\sqrt{2}}\right] \end{aligned}$$

Then, t_r can be found by setting Eq. (9b) equal to zero and solving the polynomial.

B. Maximizing Update Age Fairness

This section analytically characterizes the maximization of update age fairness for the considered probability distributions. That is, we solve $\max_{t_r}\sigma$, where σ is as in Eq. (4). The previous section has obtained analytical expressions of $M(t_l, t_r) = \sum_{i\in C}\overline{\Delta}_i\alpha_i$ for each distribution. Therefore, it is straightforward to extend such results to the maximization of σ, as follows. Let us first define the following quantities:

$$\begin{aligned} \phi(t_r) &= \left(\sum_{i\in C}\overline{\Delta}_i\alpha_i + \tilde{\Delta}_i(1-\alpha_i)\right)^2 \\ &= \left(M(t_l,t_r) + \sum_{i\in C}\tilde{\Delta}_i(1-\alpha_i)\right)^2, \\ \psi(t_r) &= N\sum_{i\in C}\left(\overline{\Delta}_i\alpha_i + \tilde{\Delta}_i(1-\alpha_i)\right)^2. \end{aligned}$$

from which it is

$$\frac{d\phi(t_r)}{dt_r} = 2\left[M(t_l,t_r) + \sum_{i\in C}\tilde{\Delta}_i(1-\alpha_i)\right]\frac{d}{dt_r}M(t_l,t_r)$$

Then,

$$\frac{d\sigma}{dt_r} = \frac{\frac{d\phi(t_r)}{dt_r}\psi(t_r) - \frac{d\psi(t_r)}{dt_r}\phi(t_r)}{\psi(t_r)^2}$$

The maximum is obtained by solving $\frac{d\sigma}{dt_r} = 0$ for t_r.

1) Uniform distribution: The derivative of σ is obtained by using $M_u(t_l, t_r)$ and $\frac{d}{dt_r}M_u(t_l, t_r)$ as in Section III-A1 for the uniform distribution. Solving $\frac{d}{dt_r}\sigma = 0$ leads to a polynomial of order 7. Therefore, the time t_r at which new updates need to be generated is determined by solving such a polynomial numerically.

2) Exponential distribution: In this case, the expressions for $M_e(t_l, t_r)$ and $\frac{d}{dt_r}M_e(t_l, t_r)$ derived in Section III-A2 for the exponential distribution are used. Moreover, the following approximations are employed for analytical tractability and to express the optimization problem as a polynomial.

$$e^{-\lambda_i t} \approx 1 - \lambda_i t \text{ with } t \in \{t_l, t_r\}.$$

The obtained eigth order polynomial is solved numerically.

3) Normal distribution: Similarly to previous distributions, $M_n(t_l, t_r)$ and $\frac{d}{dt_r}M_n(t_l, t_r)$ from Section III-A3 for the normal distribution are used to maximize update age fairness. Furthermore, the following approximations are used for analytical tractability:

$$\begin{aligned} e^{\frac{1}{2}\left(\frac{t-\mu_i}{\sigma_i}\right)^2} &\approx 1 - \frac{1}{2}\left(\frac{t-\mu_i}{\sigma_i}\right)^2, \\ \text{erf}\left(\frac{\mu_i - t}{\sigma_i\sqrt{2}}\right) &\approx \frac{2}{\sqrt{\pi}}\left(\frac{\mu_i - t}{\sigma_i\sqrt{2}} - \frac{1}{3}\left(\frac{\mu_i - t}{\sigma_i\sqrt{2}}\right)^3\right), \end{aligned}$$

where $t \in \{t_l, t_r\}$. This leads to a polynomial of order 12, which can be solved numerically.

IV. Evaluation

The proposed policies for optimal application-specific update age are evaluated next in terms of: *system update age*, measured according to Eq. (3); *update age fairness*, as expressed in Eq. (4); and energy expenditure, based on the number of updates generated by the server. The rest of this section first introduces the considered policies, then details the simulation setup, and finally presents the obtained results.

A. Considered Policies

This section first introduces optimal policies that minimize the system update age or maximize fairness. It then presents simple offline and online algorithms to derive policies that mitigate the impact of the heterogeneity in the conditions experienced by clients with respect to the update age.

Minimum system update age (min-sys). The server follows the policy introduced in Section III-A.

Maximum update age fairness (max-fairness). The server adopts the policy presented in Section III-B.

As soon as possible (ASAP). The server generates new data upon receiving a new request and immediately sends it to the client. Specifically, updates are executed at times $\rho_{i,k} : i \in C, k \in \mathbb{Z}^+$, where $\rho_{i,k}$ is the request time of update k from client i. The server also sends the data at the time they are generated; as a result, each client experiences an average update age of exactly d_i, which is also the best possible value.

Wait for farthest (W4F). The server generates individual updates for each client such that the update age experienced by each of them is approximately equal to that of the farthest client. Let f be such a client, i.e., $f = \arg\max_{i \in C} d_i$. To this end, the server first predicts the request time of all clients and then generates an update per client at a time $g_{i,k}$ such that $\rho_{i,k} - g_{i,k} + d_i \approx d_f$. Clearly, $g_{f,k} = \rho_{f,k}$ for all $k \in \mathbb{Z}^+$. The request time of all clients is predicted as the mean interarrival time of the L last requests and taking into account the last request time. That is, if update $k-1$ for client i was requested at time $\rho_{i,k-1}$ and μ_i is the mean interarrival time of the client's requests, then $g_{i,k} = \rho_{i,k-1} + \mu_i - (d_f - d_i)$. Therefore, updates are executed at times $g_{i,k} : i \in C, k \in \mathbb{Z}^+$.

Offline-periodic. The server creates updates periodically at every T time and sends the latest update to each request received from the clients. Under this policy, updates are generated at times: $t_1 + nT : n \in \mathbb{Z}^+ \cup \{0\}$, where t_1 is the generation time of the first update. This approach does not aim to minimize the system update age nor maximize fairness; however, it has been employed as a baseline that does not depend on the number of requests [25].

Online-cached. The server generates updates every T time as in the previous case but caches the last m values locally. Clients located near the server are sent cached updates, whereas the farthest clients are sent fresher updates. The cached update is determined based on the one-way delay of the farthest client f. Specifically, the n-th cached update is sent to client i, with $n \leq m$, if and only if $d_f - T \leq nT + d_i \leq d_f$. This is a simple policy that attempts to maximize fairness.

B. Simulation Setup

The three distributions introduced in Section III are considered to model the interarrival times of the requests: uniform, exponential and normal. The mean for each of them is taken as a random number between 10 ms and 200 ms, which are typical values for IoT applications [16, 24]. In particular, two random values are sampled for each client to determine its lower and upper bounds for the uniform distribution, i.e., t_i^- and t_i^+ in Section III-A1. The exponential distribution is defined by its mean value, therefore, only one random number is sampled for each client. Finally, two values are needed for the normal distribution to completely define it: mean (μ_i) and standard deviation (σ_i). These values are determined per client by sampling two random numbers in the previously described range. Then, the mean is taken as the middle point between them and the standard deviation is set to their difference.

TABLE I: Simulation parameters.

Parameter	Value
Number of clients	400
Round-trip time range	[1.4, 290.7] ms
Simulation time	5 min
Time intervals	100 ms
Number of time intervals	3,000
Mean interarrival time range	$[10, 200]$ ms
Period T for periodic and cached updates	50 ms

The delay between servers and clients is drawn from a real-world dataset in [26], which includes round-trip times between thousands of clients and servers in RIPE Atlas [27] – a global platform for Internet measurements. The evaluation makes use of 400 clients and a server located in Richmond, Virginia. These specific data are used as they include a large number of clients, cover a wide range of network infrastructure, and are large enough to provide significant variability in the network latency. The round-trip times in the considered dataset range between 1.4 ms and 290.7 ms. Finally, the probability of packet loss for each client is uniformly chosen between 0 and 0.5 to account for possibly unreliable (e.g., wireless) connectivity [28].

The experiment runs for a simulated time of 5 minutes divided into 3,000 intervals of 100 ms. In each interval, the clients requests are received and the system update age and fairness are calculated for all the proposed policies. Then, the mean over all intervals is reported in the evaluation figures together with error bars representing the standard deviation. For the periodic and cached policy, a period of $T = 50$ ms is used. Table I summarizes the simulation parameters.

C. Obtained Results

This section includes a thorough performance evaluation of the proposed update age minimization / fairness maximization schemes for the uniform distribution, and the serving policies presented in Section IV-A. First, the performance of both system update age and fairness is evaluated for every policy and request distribution. Then, an analysis of energy consumption of the proposed policies is provided.

System update age. The system update age is shown in Figure 2a for all serving policies and distributions. Unsurprisingly, the ASAP approach provides the lowest update age under all distributions, as it generates updates immediately on receiving a request. Our proposed min-sys approach achieves similar

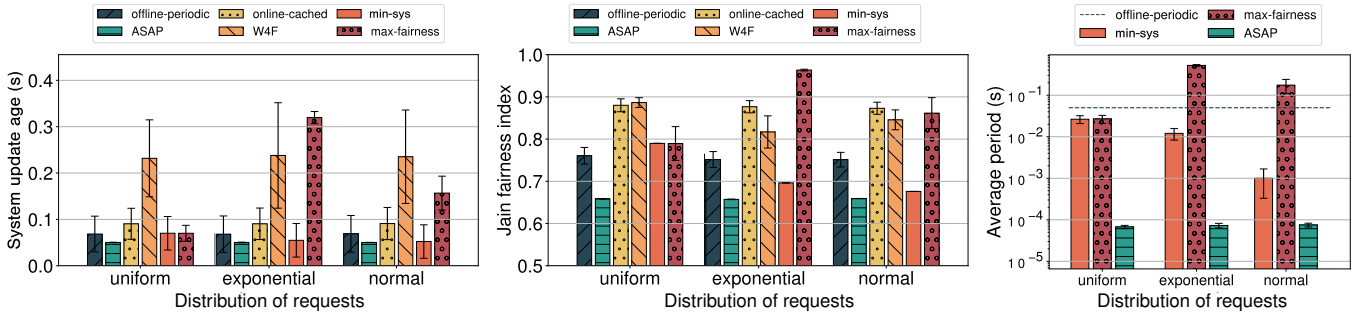

Fig. 2: (a) System update age and (b) update age fairness for all considered serving policies and request distributions in a single-server scenario. (c) Average period of the update generation under all considered distribution of requests.

system update age under exponential and normal distributions of the clients requests. Specifically, it only achieves 9.1% and 4% higher update age for the exponential and normal distributions, respectively. The online-cached policy provides a higher update age than the offline-periodic scheme, as it penalizes the near clients, by sending them cached updates, instead of the most recent ones. It is clear that, for all distributions, the W4F and max-fairness policies provide the highest update age, since they target to equalize the update age, by attempting to make all clients experience the same update age as the farthest client. It is important to note that the standard deviation of the system update age is approximately equal to that of the requests interarrival times, under the W4F policy. Since the updates are generated with respect to the average of the last L received requests per client, the spread in update age provided by this approach is determined by the variance of the requests and transmission delay. Similarly, the max-fairness policy presents a high variance under the normal distribution, due to the approximations made in Section III-B3 and the high order of the polynomial that needs to be solved at the server every time a new update is generated.

Update age fairness. The average fairness provided by each serving policy is presented in Figure 2b, for all distributions of the clients' requests. As discussed in Section II, the fairness is measured with Jain's fairness index [23] and therefore, the maximum value is one, which represents the case where all clients experience exactly the same update age. As expected, the worst update age fairness is provided by the ASAP approach, which completely neglects this metric in order to ensure the smallest system update age, as seen in Figure 2a.

Note that none of the approaches achieve perfect fairness. This is due to the random one-way delay, which makes it impossible for the server to schedule updates that provide the exact same update age to all clients. Interestingly, the W4F policy is outperformed by the online-cached policy under the exponential and normal distributions. Moreover, the online-cached policy slightly outperforms max-fairness under the normal distribution. This is because the online-cached policy does not depend on the distribution of the requests and therefore it achieves similar performance under all of them. Other approaches that depend on the distribution of the requests (such as W4F, max-fairness and min-sys) achieve lower fairness under distributions with higher standard deviation. Moreover, the trade-off between system update age and fairness is clearly exhibited by the min-sys and max-fairness schemes. The first achieves the lowest system update age under the exponential and normal distributions, at the cost of low fairness; whereas the latter achieves a very high fairness for these distributions by introducing a high system update age.

It is important to note that min-sys consistently achieves better fairness than the ASAP serving policy. In particular, it achieves 19.9%, 5.9% and 2.5% higher fairness for the uniform, exponential and normal distribution, respectively. This shows that a higher fairness can be obtained by considering upcoming requests from all clients, instead of serving each client independently. Moreover, the max-fairness policy achieves lower system update age than the W4F policy under uniform and normal distributions, which also shows the benefit of considering all upcoming requests.

Energy consumption. The energy consumption of the proposed min-sys and max-fairness serving policies is evaluated as the number of updates generated by the server, as commonly adopted in the literature [25]. In the following, we consider the average period of the updates – namely, the number of updates normalized with respect to time – as it is independent from the amount of requests at the sources.

Specifically, Figure 2c shows the average period of the proposed min-sys and max-fairness policies compared to that of the offline-periodic and ASAP policies. Interestingly, the max-fairness policy presents the lowest energy consumption under the exponential and normal distributions of requests. This shows how this approach aims to maximize fairness by sending old data to the clients. On the other hand, the ASAP policy incurs the highest energy consumption, as it generates updates every time a new request is received. However, the min-sys scheme saves more energy under the uniform, exponential and normal distributions. In particular, the min-sys and max-fairness policies approach the average period of the offline-periodic policy under the uniform distribution, since such a distribution is bounded in time; therefore, the

min-sys and max-fairness schemes are able to consider the whole time slot during which requests are expected. Thus, updates are created accordingly, resulting in a similar update period as the one created by the offline-periodic policy. This entails a similar system update age and fairness for such approaches, as seen in Figures 2a and 2b. The min-sys policy achieves two and one orders of magnitude higher average period than the ASAP policy under the exponential and normal distributions, respectively. Moreover, the system update age achieved by the min-sys approach under such distributions is very similar to that of ASAP, while also achieving slightly higher fairness, as shown by Figures 2a and 2b, respectively. This shows that the proposed approach outperforms the typical ASAP scheme in terms of fairness and energy consumption, by leveraging requests that follow exponential and normal probability distributions.

Summary. The presented results show that the proposed min-sys serving policy obtains a system update age that is similar to ASAP, while saving energy and achieving higher fairness. Min-sys is especially beneficial for applications where the interarrival time of the requests follow an exponential or a normal probability distribution. On the other hand, the proposed max-fairness policy achieves a high fairness by sporadically generating updates, which results in an increased update age at the clients. Such a behavior is expected, due to the unconstrained optimization method employed. However, the best trade-off between system update age and update age fairness is provided by the online-cached approach. Such a policy does not depend on the distribution of the requests, thus, it is also ideal for most types of applications. Moreover, online-cached does not generate more updates than the commonly used offline-periodic scheme, therefore, the energy consumption at the server is minimal.

V. Related Work

Most approaches to optimize timely data delivery target minimizing the average AoI experienced in a network [29, 30], whereas some works address fairness. Among them, Han et al. [31] consider resource scheduling in wireless systems using orthogonal frequency division multiple access that can account for incomplete knowledge on the system. Yang et al. [32] leverage proportional fairness to maximize AoI in the context of content provisioning with unmanned aerial vehicles. Different from all these works, we devise metrics based on update age which characterize timely data delivery for different classes of applications, including fairness. Moreover, we obtain optimal update generation policies for these metrics for diverse request patterns that correspond to realistic use cases.

Optimal policies have also been considered in the context of content caching [30, 33]. Specifically, Tang et al. [33] devise an update policy that minimizes AoI for time-varying content with a certain popularity distribution. Furthermore, Bastopcu and Ulukus [30] consider a system with one information source, a cache, and end users; they analytically characterize such a system to obtain optimal policies for content updates. In contrast, this work devises policies that do not rely on intermediate caches, but take advantage of the heterogeneity inherent in IoT applications.

There is also a significant share of work explicitly addressing AoI in the specific context of the IoT [25, 34, 35]. Several solutions target energy-harvesting devices [10, 36–40], which heavily rely on dynamic and potentially unpredictable environmental conditions. In contrast, we are not restricted to such a specific use case but propose policies that rather depend on different application-specific age metrics and request patterns.

Some research explicitly considers systems with multiple sources or clients [29, 30, 35, 41, 42]. In doing so, they rely on strong assumptions on how the different elements are distributed, generally close to each other. Instead, we consider IoT deployments that are possibly scattered over large geographical areas, resulting in highly-varying network conditions due to the location of the devices and their communication technologies. Moreover, we evaluate our proposed solution by using a real-world data set of Internet connectivity.

Our work also significantly distinguishes itself from the state of the art in that it does not assume any form of control on the underlying network infrastructure. In fact, most of the above-mentioned solutions require access to the queues in the network elements [9]. This implies that network operators grant application designers access to their infrastructure for improving timeliness, which is unfeasible in the context of the IoT. In contrast, the policies presented here are applicable to diverse application scenarios as they require no control over the underlying network infrastructure.

VI. Conclusion

This article has introduced an accurate model to describe timely delivery of information in heterogeneous IoT scenarios along with age-based metrics that characterize the goals of different application classes. Then, it has analytically derived optimal policies that minimize the system update age and maximize update age fairness for diverse types of request patterns corresponding to realistic use cases. A performance evaluation based on a large-scale Internet dataset demonstrated that the proposed policies achieve timely data delivery that is competitive with those in the state of the art, while consuming up to 100 times less energy due to a more efficient generation of the updates. Moreover, they achieve up to 19.9% higher update age fairness, which is beneficial to applications requiring balancing update age. As a future work, the model could be extended to account for different distributions of the network delays and message losses. Moreover, a multi-objective optimization problem could be formulated to jointly account for update age, fairness, and energy consumption.

Acknowledgment

This work was partially supported by the Research Council of Finland under grant number 357533.

REFERENCES

[1] A. Elgabli, H. Khan, M. Krouka, and M. Bennis, "Reinforcement Learning Based Scheduling Algorithm for Optimizing Age of Information in Ultra Reliable Low Latency Networks," in *IEEE ISCC*, 2019, pp. 1–6.

[2] P. Kortoçi, A. Merabi, C. Joe-Wong, and M. Di Francesco, "Incentivizing opportunistic data collection for time-sensitive IoT applications," in *IEEE SECON*, July 2021.

[3] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. H. Katz, A. Konwinski, G. Lee, D. A. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "Above the clouds: A Berkeley view of cloud computing," Tech. Rep. UCB/EECS-2009-28, Feb 2009. [Online]. Available: http://www2.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS-2009-28.html

[4] G. Premsankar, M. Di Francesco, and T. Taleb, "Edge Computing for the Internet of Things: A Case Study," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 1275–1284, April 2018.

[5] J. Cho and H. Garcia-Molina, "Synchronizing a database to improve freshness," in *ACM SIGMOD*, 2000.

[6] N. Lu, B. Ji, and B. Li, "Age-based scheduling: Improving data freshness for wireless real-time traffic," in *ACM MobiHoc*, 2018.

[7] Q. He, D. Yuan, and A. Ephremides, "Optimizing freshness of information: On minimum age link scheduling in wireless systems," in *WiOpt*, 2016.

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

[9] R. D. Yates, Y. Sun, D. R. Brown, S. K. Kaul, E. Modiano, and S. Ulukus, "Age of information: An introduction and survey," *IEEE JSAC*, vol. 39, no. 5, pp. 1183–1210, 2021.

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

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

[12] R. D. Yates, M. Tavan, Y. Hu, and D. Raychaudhuri, "Timely cloud gaming," in *IEEE INFOCOM*, 2017.

[13] H. Wang, X. Xie, X. Li, and J. Yang, "Scheduling schemes for age optimization in iot systems with limited retransmission times," *IEEE Internet of Things J.*, vol. 9, no. 21, pp. 21 458–21 468, 2022.

[14] Q. Abbas, S. A. Hassan, H. K. Qureshi, K. Dev, and H. Jung, "A comprehensive survey on age of information in massive IoT networks," *Computer Communications*, vol. 197, pp. 199–213, 2023.

[15] E. Uysal, O. Kaya, S. Baghaee, and H. B. Beytur, *Age of Information in Practice*. Cambridge University Press, 2023, p. 297–326.

[16] P. Schulz, M. Matthe, H. Klessig, M. Simsek, G. Fettweis, J. Ansari, S. A. Ashraf, B. Almeroth, J. Voigt, I. Riedel, A. Puschmann, A. Mitschele-Thiel, M. Muller, T. Elste, and M. Windisch, "Latency critical IoT applications in 5G: Perspective on the design of radio interface and network architecture," *IEEE Communications Magazine*, vol. 55, no. 2, pp. 70–78, 2017.

[17] A. Kosta, N. Pappas, A. Ephremides, and V. Angelakis, "Age and value of information: Non-linear age case," in *IEEE ISIT*, 2017.

[18] C. Kam, S. Kompella, G. D. Nguyen, and A. Ephremides, "Effect of message transmission path diversity on status age," *IEEE Transactions on Information Theory*, 2016.

[19] M. Costa, M. Codreanu, and A. Ephremides, "On the age of information in status update systems with packet management," *IEEE Transactions on Information Theory*, 2016.

[20] E. Najm and R. Nasser, "Age of information: The gamma awakening," in *IEEE ISIT*, 2016.

[21] T. Qiu, N. Chen, K. Li, M. Atiquzzaman, and W. Zhao, "How can heterogeneous Internet of Things build our future: A survey," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 3, pp. 2011–2027, 2018.

[22] L. Fisser and A. Timm-Giel, "Minimizing Age of Information for Distributed Control in Smart Grids," in *IEEE SmartGridComm*, 2021.

[23] R. Jain, D.-M. Chiu, and W. Hawe, "A quantitative measure of fairness and discrimination for resource allocation in shared computer systems," *CoRR*, vol. cs.NI/9809099, 1998.

[24] 3GPP, "5G; Service requirements for cyber-physical control applications in vertical domains," TS 22.104, 05 2022, version 17.7.0.

[25] L. Corneo, C. Rohner, and P. Gunningberg, "Age of information-aware scheduling for timely and scalable Internet of Things applications," in *IEEE INFOCOM*, 2019.

[26] L. Corneo, M. Eder, N. Mohan, A. Zavodovski, S. Bayhan, W. Wong, P. Gunningberg, J. Kangasharju, and J. Ott, "Surrounded by the clouds: A comprehensive cloud reachability study," in *ACM Web Conference*, 2021.

[27] V. Bajpai, S. J. Eravuchira, and J. Schönwälder, "Lessons Learned From Using the RIPE Atlas Platform for Measurement Research," *ACM SIGCOMM CCR*, 2015.

[28] O. Ayan, H. Murat Gürsu, A. Papa, and W. Kellerer, "Probability analysis of age of information in multi-hop networks," *IEEE Networking Letters*, vol. 2, no. 2, pp. 76–80, 2020.

[29] Q. Jing, X. Wang, P. Zhou, K. Liu, and W. Wu, "Minimizing the age of multisource information with budget constraint in Internet of Things," *IEEE Internet of Things Journal*, vol. 9, no. 8, pp. 6173–6183, 2022.

[30] M. Bastopcu and S. Ulukus, "Cache freshness in information updating systems," in *CISS*, 2021.

[31] B. Han, Y. Zhu, Z. Jiang, M. Sun, and H. D. Schotten, "Fairness for freshness: Optimal age of information based OFDMA scheduling with minimal knowledge," *IEEE Transactions on Wireless Communications*, vol. 20, no. 12, pp. 7903–7919, 2021.

[32] P. Yang, K. Guo, X. Xi, T. Q. S. Quek, X. Cao, and C. Liu, "Fresh, fair and energy-efficient content provision in a private and cache-enabled UAV network," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 1, pp. 97–112, 2022.

[33] H. Tang, P. Ciblat, J. Wang, M. Wigger, and R. Yates, "Age of information aware cache updating with file- and age-dependent update durations," in *WiOpt*, 2020.

[34] B. Zhou and W. Saad, "Optimal sampling and updating for minimizing age of information in the internet of things," in *IEEE GLOBECOM*, 2018.

[35] M. A. Abd-Elmagid and H. S. Dhillon, "Distribution of AoI in EH-powered multi-source systems with source-aware packet management," in *IEEE ICC*, 2022.

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

[37] Y. Sun, E. Uysal-Biyikoglu, R. Yates, C. E. Koksal, and N. B. Shroff, "Update or wait: How to keep your data fresh," in *IEEE INFOCOM*, 2016.

[38] T. Bacinoglu and E. Uysal-Biyikoglu, "Scheduling status updates to minimize age of information with an energy harvesting sensor," in *IEEE ISIT*, 2017.

[39] X. Wu, J. Yang, and J. Wu, "Optimal status update for age of information minimization with an energy harvesting source," *IEEE Trans. on Green Communications and Networking*, 2018.

[40] S. Feng and J. Yang, "Optimal status updating for an energy harvesting sensor with a noisy channel," *IEEE INFOCOM WKSHPS*, 2018.

[41] N. Pappas, J. Gunnarsson, L. Kratz, M. Kountouris, and V. Angelakis, "Age of information of multiple sources with queue management," in *IEEE ICC*, 2015.

[42] A. M. Bedewy, Y. Sun, S. Kompella, and N. B. Shroff, "Age-optimal sampling and transmission scheduling in multi-source systems," in *ACM MobiHoc*, 2019.