Semantics-Aware Status Updates with Energy Harvesting Devices: Query Version Age of Information

Erfan Delfani and Nikolaos Pappas, *Senior Member, IEEE*

***Abstract*—In this work, we study the freshness and significance of information in an IoT status update system where an Energy Harvesting (EH) device samples an information source and forwards the update packets to a destination node through a direct channel. We introduce and optimize a semantics-aware metric, Query Version Age of Information (QVAoI), in the system along with other semantic metrics: Query Age of Information (QAoI), Version Age of Information (VAoI), and Age of Information (AoI). By employing the MDP framework, we formulate the optimization problem and determine the optimal transmission policies at the device, which involve deciding the time slots for updating, subject to the energy limitations imposed by the device's battery and energy arrivals. Through analytical and numerical results, we compare the performance of the semantics-aware QVAoI-Optimal, QAoI-Optimal, VoI-Optimal, and AoI-Optimal policies with a baseline greedy policy. All semantics-aware policies show significantly improved performance compared to the greedy policy. The QVAoI-Optimal policy, in particular, demonstrates a significant performance improvement by either providing fresher, more relevant, and valuable updates with the same amount of energy arrivals or reducing the number of transmissions in the system to maintain the same level of freshness and significance of information compared to the QAoI-Optimal and other policies.**

I. Introduction

Communicating timely and informative data is crucial for status update systems within real-time IoT networks. These systems involve the sensing and transmission of update packets from an information source through a network to a destination monitor for further processing [1], [2]. Information packets are sampled using IoT devices and shared within a network, facilitating various applications related to the monitoring and controlling of remote environments, particularly in the contexts of smart cities, intelligent industries, smart agriculture, metaverse, and healthcare. However, there are limitations in these networks, as IoT devices are highly energy-limited, and the network resources are restricted in terms of bandwidth, channel reliability or availability, and other factors. These limitations necessitate more effective, cost- and energy-efficient approaches for status updating, particularly when communication from a remote, low-energy IoT device to a destination network is involved.

The *semantics-aware communication* paradigm introduces novel approaches that address the transmission and utilization of the right amount of data at the right time to achieve the designated goals within status update systems [3]. This is accomplished by employing semantics-aware performance metrics and managing the information chain from generation to transmission and utilization within a communication network. Recent studies have demonstrated substantial benefits of this paradigm in status update systems [1].

At the core of semantics-aware communication are semantic metrics that capture the *freshness*, *relevance*, or *value* of information. The freshness of information relates to the staleness of information packets in the network from their generation until their utilization. Higher freshness corresponds to lower staleness of the data packets. The relevance of information relates to sampling the appropriate piece of information from the source. In contrast, the value of information involves providing the destination node with the right piece of information regarding the benefits of having it compared to the cost of its transmission. We can refer to the relevance and value of information as the *significance* of information. Various semantic metrics have been introduced in the literature, including Age of Information (AoI) [4], non-linear AoI [5], Age of Incorrect Information (AoII) [6], Query Age of Information (QAoI) [7], Version Age of Information (VAoI) [8], and state-aware AoI [9]. Among these, AoI is a freshness metric that quantifies the time elapsed since the generation time of the last successfully received packet at the destination node. Version AoI (VAoI) is a semantic metric that quantifies jointly the freshness and relevance of information, measuring the number of versions by which the receiver lags behind the source. Query AoI (QAoI) is another performance metric that represents the freshness and value of information. QAoI considers the AoI only when there is a request from the destination node, i.e., only in query instances when the information is assumed to be useful to the receiver.

In this work, we introduce the Query VAoI (QVAoI) metric, which captures all three attributes of semantic metrics. We investigate the advantages of employing and optimizing the VAoI and QVAoI metrics within a status update system compared to the AoI, and its query-based counterpart, QAoI. VAoI and QVAoI, being content-based metrics, capture the freshness and also the relevance and value of information, whereas AoI and QAoI do not consider the relevance of the updates. We aim to demonstrate that optimizing the VAoI and QVAoI within a status update system with an Energy Harvesting (EH) IoT device can lead to improved performance and reduced transmissions from the IoT device to the destination network without compromising the conveyed information, particularly when dealing with information sources evolving in versions.

The authors are with the Department of Computer and Information Science at Linköping University, Sweden, email: {erfan.delfani, nikolaos.pappas}@liu.se. This work has been supported by the Swedish Research Council (VR), ELLIIT, and the European Union (ETHER, 101096526, ELIXIRION, 101120135, and SOVEREIGN, 101131481).

II. Semantic metrics: Query Version Age of Information

In this section, we present the metrics AoI, QAoI, and VAoI, and introduce our proposed metric, QVAoI. AoI is defined as the time elapsed since the generation time of the last received update, i.e., $\Delta^{AoI}(t) = t - u(t)$, where $u(t)$ is the timestamp of the current update at the receiver. The AoI is the most widely used metric in the literature, and its optimization yields the freshest updates in status update systems across various configurations (see [1], [4], [10], [11]).

VAoI quantifies the number of versions by which the receiver lags behind the source, i.e., $\Delta^{VAoI}(t) = N_S(t) - N_R(t)$, where $N_S(t)$ and $N_R(t)$ represent the version numbers of the current updates at the source and receiver, respectively. By version, we refer to any significant change in the content or new generation of information at the source. VAoI has been studied in several works. The works [8], [12]–[17] focus on the scaling and growth of VAoI in gossiping networks with different topologies and system models. The works [18], [19] focus on the optimization of VAoI in gossiping networks, while the work [20] characterizes the higher-order statistics of VAoI in gossiping networks.

QAoI is an extension of AoI and considers the age at instances when there is a request from the destination node. This metric is suitable for *pull-based* status update systems, wherein the destination node requests and controls the generation or transmission of updates as needed and when they are valuable for utilization. A pull-based setup contrasts with a *push-based* setup, in which the source or transmitter decides to push updates toward the receiver at its discretion, regardless of requests from the receiver. If we presume a binary request arrival process $r(t)$, where $r(t)$ equals 1 when there is a request from the receiver and 0 otherwise, then QAoI is defined as $\Delta^{QAoI}(t) = r(t) \times \Delta^{AoI}(t)$. This metric penalizes system staleness only when requests are demanded. In a practical scenario where there is a lag (τ) between the request time and when the system controls are applied, the definition can be corrected to $\Delta^{QAoI}(t) = r(t-\tau) \times \Delta^{AoI}(t)$. QAoI has been studied in the works [7], [21]–[24].

Here, we introduce QVAoI metric as an extension of VAoI and QAoI, which considers both the content changes at the source and the queries from the destination. This metric combines the advantages of both VAoI and QAoI to represent relevant and valuable updates alongside the freshest ones. We define the QVAoI as the count of versions by which the receiver lags behind the source at query instances. This metric ensures that the system is not penalized in the absence of requests, and when the transmission of updates holds no value or utility for the receiver. The mathematical representation of QVAoI is:

$$\Delta^{QVAoI}(t) = r(t-\tau) \times \Delta^{VAoI}(t), \quad (1)$$

where $r(t)$ represents the binary request arrival process, and τ denotes the time lag between the request time and the instance when the system controls are applied.

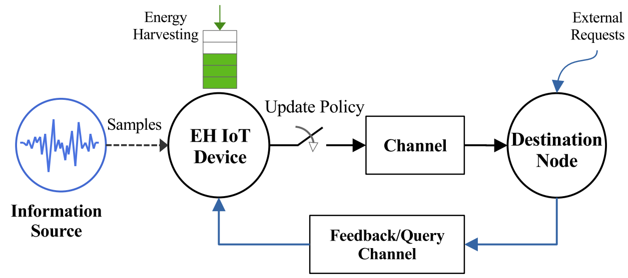

Fig. 1: The considered system model.

III. System Model

A. System Setup

We consider a status update system, as depicted in Fig. 1, where an Energy Harvesting (EH) IoT device samples and transmits update packets to a destination node. The IoT device can be a sensor or a measurement device, and the packets are sampled from an information source, typically a physical process. The destination node can either initiate a request or act as a request aggregator for a destination network, sending queries to the IoT device to obtain new updates and then delivering these updates back to the destination network.

Our objective is to derive an information handling or an *update policy* that optimizes the semantic metrics within this system while adhering to the energy constraints imposed on the EH device. This update policy determines the scheduling of optimal times for transmitting new updates, which results in the best performance in terms of semantic metrics. The system operates in discrete time slots, allowing the device to decide whether to transmit an update or remain idle during each slot. We assume that each time slot lasts long enough for the sampling and transmission of a new update from the device to the receiver. The EH device is equipped with a rechargeable battery of B units, collecting energy from ambient sources according to a Bernoulli distribution with probability β in each time slot. When the battery is empty, the device cannot send updates; otherwise, it must decide on the scheduling of transmissions. We assume a normalized battery unit so that each transmission consumes one energy unit from the battery. In this system, each update packet from the information source is labeled by a timestamp and a version number, with new versions generated with probability p_t in each time slot.

We consider a forward unreliable channel from the device to the receiver, where data packets are transmitted, exhibiting a success probability denoted by p_s. We also consider an error-free backward channel from the receiver to the device, which can serve the queries and acknowledgments (ACK) to receive data packets successfully. The device becomes aware of the last timestamp and version stored at the destination node by receiving ACK feedback. We assume a query arrival process $r(t)$ from the receiver, which follows a Bernoulli distribution with a probability of q in each time slot, representing requests from the destination node or an external network.

B. Problem Formulation

Our objective is to determine an optimal update policy to optimize the time-averaged expected value of AoI, QAoI, VAoI, and QVAoI, as defined in Section II.

$$\Delta^{AoI}(t) = t - u(t), \tag{2}$$
$$\Delta^{VAoI}(t) = N_S(t) - N_R(t), \tag{3}$$
$$\Delta^{QAoI}(t) = r(t-1) \times \Delta^{AoI}(t), \tag{4}$$
$$\Delta^{QVAoI}(t) = r(t-1) \times \Delta^{VAoI}(t), \tag{5}$$

where t is the current time, $u(t)$ is the timestamp of the current update at the receiver, $N_S(t)$ and $N_R(t)$ represent the version numbers of the current updates at the source and receiver, respectively, and $r(t)$ is the binary query arrival process.

An update policy, denoted by π, is a sequence of actions taken by the IoT device at different time slots, i.e., $\pi = \big(a^\pi(0), a^\pi(1), a^\pi(2), \cdots\big)$, where $a^\pi(t)$ is the action realized at time t under the policy π. Specifically, $a^\pi(t) = 1$ indicates a transmission action, while $a^\pi(t) = 0$ denotes an idle action. Due to the stochastic nature of the system variables, the resulting semantic metric under policy π is a stochastic process, denoted by $\Delta^\pi(t)$, where $\Delta^\pi(t)$ can be either $\Delta^{AoI}(t)$, $\Delta^{QAoI}(t)$, $\Delta^{VAoI}(t)$, or $\Delta^{QVAoI}(t)$. This optimization problem can be formulated as follows:

$$\min_{\pi\in\Pi} \lim_{T\to\infty} \frac{1}{T} E\left[\sum_{t=0}^{T-1} \Delta^\pi(t) \Big| s_0\right], \tag{6}$$

where Π is the set of all feasible policies and s_0 is the system's initial state. (6) can be formulated as an infinite-horizon average cost Markov Decision Process (MDP) problem. The MDP problem is characterized by a tuple $< \mathcal{S}, \mathcal{A}, P, C >$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the set of actions, P is the state transition probability function, and C is the cost function.

- *States*: The state vector $s(t)$ is defined as $s(t) \triangleq [b(t), \Delta(t), r(t)]^T \in \mathcal{S}$, where $b(t)$ is the state of the battery, taking value in the set $B = \{0, 1, 2, \ldots, b_{\max}\}$. $\Delta(t) \in \{0, 1, 2, \cdots, \Delta_{\max}\}$ is either AoI or VAoI at the receiver, and $r(t) \in \{0, 1\}$ is the query process at time slot t. $r(t)$ is 1 when there is a query from the receiver and 0 otherwise. The state space, $\mathcal{S} = \big\{(b, \Delta, r) : \ b \in B, \Delta \in \{1, 2, \cdots, \Delta_{\max}\}, \text{ and } r \in \{0, 1\}\big\}$ is a finite set.
- *Actions*: At time t, $a(t) = 0$ represents the action of staying idle, while $a(t) = 1$ represents the action of transmitting an update. The action $a(t)$ is forced to be 0 when $b(t) = 0$ or $r(t) = 0$. We define the realized action as $d(t) \triangleq \mathbb{1}_{b(t)\neq 0} \times r(t) \times a(t)$, where $\mathbb{1}$ is the indicator function.
- *Transition probabilities*: Given the following equation,

$$P\left[s(t+1)|s(t), a(t)\right] = P\left[b(t+1)|b(t), r(t), a(t)\right] \times P\left[\Delta(t+1)|\Delta(t), r(t), a(t)\right] \times P\left[r(t+1)\right], \tag{7}$$

  the transition probabilities are presented in Section III-C.
- *Cost function*: The transition cost function is equal to the semantic metric at the next time slot, i.e., $C\big(s(t), a(t), s(t+1)\big) \triangleq \Delta(t+1)$, where $\Delta(t)$ can be either $\Delta^{AoI}(t)$, $\Delta^{QAoI}(t)$, $\Delta^{VAoI}(t)$, or $\Delta^{QVAoI}(t)$.

C. Transition probabilities

We provide the transition probabilities between the MDP states by introducing the following Bernoulli processes: the energy arrival process, $e(t)$, the channel success process, $c(t)$, and the version generation process, $z(t)$, given by:

$$e(t) = \begin{cases} 1 & \text{w.p. } \beta, \\ 0 & \text{w.p. } 1-\beta, \end{cases} \quad c(t) = \begin{cases} 1 & \text{w.p. } p_s, \\ 0 & \text{w.p. } 1-p_s, \end{cases} \quad z(t) = \begin{cases} 1 & \text{w.p. } p_t, \\ 0 & \text{w.p. } 1-p_t. \end{cases} \tag{8}$$

We can now characterize the evolution of the states:

$$b(t+1) = \min\left\{b(t) + e(t) - d(t), b_{\max}\right\}. \tag{9}$$

$$\Delta^{AoI}(t+1) = \begin{cases} 1, & d(t) = 1 \text{ and } c(t) = 1, \\ \min\left\{\Delta^{AoI}(t) + 1, \Delta_{\max}\right\}, & \text{o/w.} \end{cases} \tag{10}$$

$$\Delta^{VAoI}(t+1) = \begin{cases} z(t), & d(t) = 1 \text{ and } c(t) = 1, \\ \min\left\{\Delta^{VAoI}(t) + z(t), \Delta_{\max}\right\}, & \text{o/w.} \end{cases} \tag{11}$$

$$r(t+1) = \begin{cases} 1 & \text{w.p. } q, \\ 0 & \text{w.p. } 1-q. \end{cases} \tag{12}$$

The transition probabilities can be calculated according to (7) and the following equations:

$$P\big[b(t+1)\big|b(t), r(t), a(t)\big] = \begin{cases} \beta & d(t) = 0,\ b(t+1) = b(t) + 1, \\ \bar\beta & d(t) = 0,\ b(t+1) = b(t), \\ \beta & d(t) = 1,\ b(t+1) = b(t), \\ \bar\beta & d(t) = 1,\ b(t+1) = b(t) - 1. \end{cases} \tag{13}$$

$$P\left[\Delta^{AoI}(t+1)\big|\Delta^{AoI}(t), r(t), a(t)\right] \tag{14}$$
$$= \begin{cases} 1 & d(t) = 0,\ \Delta^{AoI}(t+1) = \Delta^{AoI}(t) + 1, \\ \bar p_s & d(t) = 1,\ \Delta^{AoI}(t+1) = \Delta^{AoI}(t) + 1, \\ p_s & d(t) = 1,\ \Delta^{AoI}(t+1) = 1. \end{cases}$$

$$P\left[\Delta^{VAoI}(t+1)\big|\Delta^{VAoI}(t), r(t), a(t)\right] \tag{15}$$
$$= \begin{cases} p_t & d(t) = 0,\ \Delta^{VAoI}(t+1) = \Delta^{VAoI}(t) + 1, \\ \bar p_t & d(t) = 0,\ \Delta^{VAoI}(t+1) = \Delta^{VAoI}(t), \\ p_t \bar p_s & d(t) = 1,\ \Delta^{VAoI}(t+1) = \Delta^{VAoI}(t) + 1, \\ \bar p_t \bar p_s & d(t) = 1,\ \Delta^{VAoI}(t+1) = \Delta^{VAoI}(t), \\ p_t p_s & d(t) = 1,\ \Delta^{VAoI}(t+1) = 1, \\ \bar p_t p_s & d(t) = 1,\ \Delta^{VAoI}(t+1) = 0. \end{cases}$$

$$P\left[r(t+1)\right] = \begin{cases} q & r(t+1) = 1, \\ 1 - q & r(t+1) = 0, \end{cases} \tag{16}$$

where $\bar\beta \triangleq 1 - \beta$, $\bar p_t \triangleq 1 - p_t$, and $\bar p_s \triangleq 1 - p_s$. The total probability theorem can also help in simplifying (7):

$$P[s(t+1)|s(t), a(t)] \tag{17}$$
$$= \sum_{(z,e,c)\in\{0,1\}^3} P\left[s(t+1)\big|s(t), a(t), z(t), e(t), c(t)\right] P_c P_e P_z,$$

with $P_c \triangleq P\left[c(t) = c\right]$, $P_e \triangleq P\left[e(t) = e\right]$, and $P_z \triangleq P\left[z(t) = z\right]$ given by (8).

IV. Analytical Results

In this section, we discuss the existence and structure of the optimal policies. We proceed with the QVAoI as the cost function since it generally encompasses other metrics.

Definition 1. *An MDP is weakly accessible if its states can be divided into two subsets, S_t and S_c. States in S_t are transient under any stationary policy, and any state s' in S_c can be reached from any state s in S_c under some stationary policy.*

Proposition 1. *The MDP problem* (6) *is weakly accessible.*

Proof. We demonstrate that any state $s' = (b', \Delta', r') \in \mathcal{S}$ is reachable from any other state $s = (b, \Delta, r) \in \mathcal{S}$ under a stationary stochastic policy π, where the action $a \in \{0, 1\}$ at each state is randomly selected with positive probability. The state $r' \in \{0, 1\}$ is accessible at each state independently of system actions and can remain unchanged with positive probability (w.p.p.). Therefore, we fix r' as states evolve in the remainder of this proof. The state $b' < b$ is reachable from b w.p.p. by executing action $a = 1$ for $(b - b')$ time slots, and $b' \geq b$ is reachable from b w.p.p. by executing action $a = 0$ for $(b' - b)$ slots. Upon reaching state b', regardless of subsequent actions, the battery state can remain unchanged w.p.p. Henceforth, we consider the battery state as b' for the rest of the proof. Similarly, the state $\Delta' < \Delta$ can be attained from Δ w.p.p. by executing action $a = 1$ for one time slot, followed by action $a = 0$ for Δ' slots. Conversely, the state $\Delta' \geq \Delta$ is accessible from Δ by executing action $a = 0$ for $(\Delta' - \Delta)$ slots. □

Proposition 2. *In the MDP problem* (6)*, the optimal average cost J^* achieved by an optimal policy π^* is the same for all initial states, and it satisfies the Bellman's equation:*

$$J^* + V(s) = \min_{a\in\{0,1\}} \left\{ C(s,a) + \sum_{s'\in\mathcal{S}} P\left(s' \middle| s,a\right) V(s') \right\}, \quad (18)$$

$$\pi^*(s) \in \arg\min_{a\in\{0,1\}} \left\{ C(s,a) + \sum_{s'\in\mathcal{S}} P\left(s' \middle| s,a\right) V(s') \right\}, \quad (19)$$

where $V(s)$ denotes the value function of the MDP problem, $P\left(s' \middle| s,a\right)$ is the transition function, and $C(s,a)$ is the average cost per slot, given by the transition costs,

$$C(s,a) = \sum_{s'\in\mathcal{S}} P\left(s' \middle| s,a\right) C\left(s,a,s'\right), \quad (20)$$

with $C(s,a,s') = r\Delta'$.

Proof. According to Proposition 1, the MDP problem (6) is weakly accessible. Consequently, by Proposition 4.2.3 in [25], the optimal average cost is invariant across all initial states. Moreover, Proposition 4.2.6 in [25] guarantees the existence of an optimal policy. According to Proposition 4.2.1 in [25], identifying J^* and $V(s)$ satisfying (18) enables determination of the optimal policy using (19). □

The optimal policy, denoted as π^*, relies on $V(s)$, which typically lacks a closed-form solution. Standard methods, like the (Relative) Value Iteration and Policy Iteration algorithms [25, Chapter 4], can solve this optimization problem.

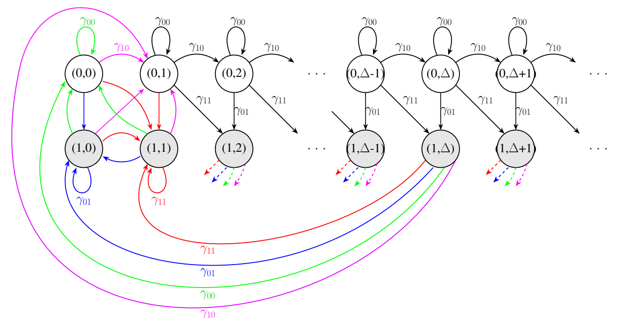

Fig. 2: The resulting Markov chain with threshold $\Delta_T = 0$ (greedy policy).

Definition 2. *Suppose there exists a $\Delta_T(b) > 0$ for each b such that the action $\pi(b, \Delta, r = 1)$ is 1 for $\Delta \geq \Delta_T(b)$, and 0 otherwise. In this case, π is a threshold policy.*

Theorem 1. *The optimal policy of the MDP problem* (6) *is a threshold policy.*

Proof. See Appendix A. □

A. Optimal Policy for a Single-Sized Battery

Having established that the optimal policy is a threshold policy, to provide insights into the benefits of semantics-aware information handling, we analyze the average QVAoI of a system with the minimum battery size, i.e., $b_{\text{max}} = 1$. For tractability, we further assume that the channel is reliable ($p_s = 1$) and the query always exists ($q = 1$). To this end, we examine the resulting Markov chains of the system for policies characterized by different threshold values Δ_T, where the action at state $s = (b, \Delta, r)$ is defined as follows:

$$a(b,\Delta) = \begin{cases} 0 & b = 0 \ \texttt{or} \ \ (b = 1 \ \texttt{and} \ \Delta < \Delta_T) \\ 1 & b = 1 \ \texttt{and} \ \Delta \geq \Delta_T \end{cases} \quad (21)$$

where we have excluded r from the state vector since it is always equal to 1 here. The state space for the Markov chains is defined by $\mathcal{S}_I = \left\{(b, \Delta) \middle| b \in \{0, 1\}, \Delta \in \{0, 1, 2, \cdots\}\right\}$. We consider three cases: $\Delta_T = 0$, $\Delta_T = 1$, and $\Delta_T \geq 2$. The resulting Markov chains are depicted in Figs. 2, 3, and 4. In these figures, for enhanced clarity and compact representation, we define parameters $\gamma_{00} \triangleq \bar{p}_t \bar{\beta}$, $\gamma_{01} \triangleq \bar{p}_t \beta$, $\gamma_{10} \triangleq p_t \bar{\beta}$, and $\gamma_{11} \triangleq p_t \beta$, and we use the same color for arrows indicating equal transition probabilities. The actions 0 and 1 are represented with circles filled with white and grey colors, respectively, at each state (b, Δ).

The average QVAoI for these Markov chains is given by:

$$\Delta_{Avg}^{QVAoI} = \sum_{s\in\mathcal{S}_I} \Delta \times \mu(s) = \sum_{(b,\Delta)\in\mathcal{S}_I} \Delta \times \mu(b,\Delta), \quad (22)$$

where $\mu(s)$ is the state-stationary probability of state $s = (b, \Delta)$. The state-stationary probabilities can be obtained by solving *balance equations*:

$$\boldsymbol{\mu} \boldsymbol{P_I} = \boldsymbol{\mu} \ \ \texttt{and} \ \sum_{s_i \in \mathcal{S}_I} \mu(s_i) = 1, \quad (23)$$

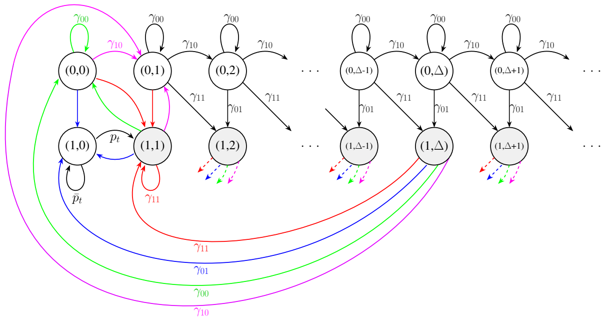

Fig. 3: The resulting Markov chain with threshold $\Delta_T = 1$.

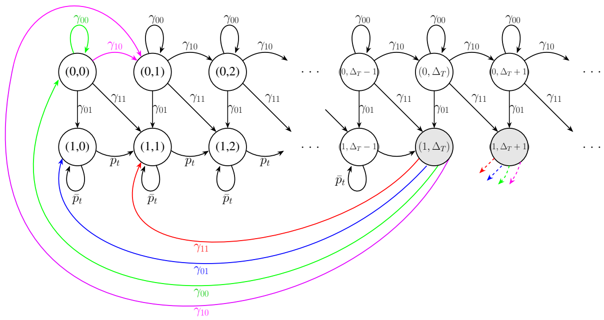

Fig. 4: The resulting Markov chain with threshold $\Delta_T \geq 2$.

where $\boldsymbol{\mu} = [\mu(s_1)\ \mu(s_2)\ \mu(s_3)\ \cdots]$ is a row vector representing the stationary distributions. $\boldsymbol{P_I}$ is the transition probability matrix whose $(i,j)^{th}$ element is the transition probability from state s_i to s_j, as defined in Section III-C with $b_{\text{max}} = 1$.

Theorem 2. *In a permanent query setup with a single-sized battery, the average QVAoI for a threshold policy is given by:*

$$\Delta_{Avg}^{QVAoI} = \begin{cases} \frac{p_t}{\beta} & \Delta_T = 0, \\ \frac{p_t}{\beta}\left(1 - \frac{\bar{p}_t\bar{\beta}\beta^2}{\beta^2 + p_t\bar{\beta}(p_t+\beta)}\right) & \Delta_T = 1, \\ \frac{\mathcal{N}(\Delta_T)}{\mathcal{D}(\Delta_T)} & \Delta_T \geq 2, \end{cases} \tag{24}$$

where Δ_T is the threshold, and $\mathcal{N}(\Delta_T)$ and $\mathcal{D}(\Delta_T)$ are:

$$\begin{aligned} \mathcal{N}(\Delta_T) = \beta^2 \Big[& (p_t^2 + \bar{p}_t^2\beta^2)(\Delta_T - 1)(\Delta_T + 2p_t) \\ & - 2\beta p_t \Big(2p_t^2(\Delta_T - 1) + p_t(\Delta_T - 2)(\Delta_T - 1) - \Delta_T^2 + \Delta_T - 1\Big)\Big] \\ & + 2p_t(\beta + p_t\bar{\beta})^2(p_t + \beta\Delta_T)\left(\frac{p_t\bar{\beta}}{\beta + p_t\bar{\beta}}\right)^{\Delta_T}, \end{aligned} \tag{25}$$

$$\mathcal{D}(\Delta_T) = 2\beta(\beta + p_t\bar{\beta})^2\left[\beta\Delta_T + p_t\left(\frac{p_t\bar{\beta}}{\beta + p_t\bar{\beta}}\right)^{\Delta_T}\right]. \tag{26}$$

The parameters p_t and β represent the version generation and energy arrival probabilities, respectively.

Proof. See Appendix B. □

Theorem 2 shows that the average QVAoI for $\Delta_T = 1$ is always less than that for $\Delta_T = 0$. However, for $\Delta_T \geq 2$, the average QVAoI can be either less than or greater than that for $\Delta_T = 1$, depending on Δ_T, p_t, and β.

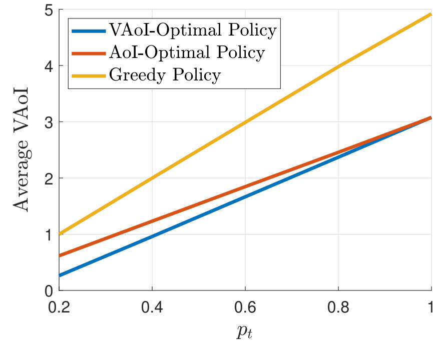

Fig. 5: Average VAoI vs. p_t.

V. Numerical Results

The MDP problem (6) can be solved using the Relative Value Iteration Algorithm (RVIA) to obtain the optimal policies. We evaluate the performance of the AoI-Optimal, VAoI-Optimal, QAoI-Optimal, and QVAoI-Optimal policies concerning the semantic metrics in the system. In query-based configurations, the QVAoI metric represents the freshest relevant and most valuable updates within the system, serving as the key performance metric. In push-based setups, the VAoI metric denotes the freshest relevant updates within the system, considered as the key performance metric. Moreover, to highlight the advantages of semantics-aware status updating, we include a greedy policy as a baseline. According to the greedy policy, the device transmits an update (upon request in a pull-based setup) as soon as energy becomes available.

A. The Impact of Version Generation Probability p_t

1) VAoI-optimal vs. AoI-optimal policy in optimizing VAoI: The primary advantage of VAoI over AoI is that VAoI considers the dynamics of the information source, utilizing the knowledge of the version generation probability, i.e., p_t. In this section, we demonstrate how this advantage can improve the system's performance in terms of maintaining the freshest relevant updates of information. We evaluate the average VAoI of the system for the VAoI-optimal, AoI-optimal, and greedy policies by varying p_t from 0.2 to 1 while fixing the parameters $\beta = 0.2$, $q = 1$, $p_s = 1$, $b_{\text{max}} = 15$, and $\Delta_{\text{max}} = 19$. It can be observed in Fig. 5 that, as expected, the average VAoI in the system deteriorates as the probability of version generation p_t increases, for all three policies. The reason is that the versions of the source become more stale as the energy resources, and thus the possibility of sending updates, are fixed and unchanged while the new versions are generated with a higher probability. In Fig. 5, it is evident that semantics-aware policies, VAoI-optimal and AoI-optimal policies, demonstrate superior performance compared to the greedy policy. Additionally, the VAoI-optimal policy exhibits superior performance in comparison to the AoI-optimal policy, especially when the value of p_t is low.

2) QVAoI-optimal vs. QAoI-optimal policy in optimizing QVAoI: In Fig. 6, the average QVAoI is depicted for QVAoI-optimal, QAoI-optimal, and greedy policies as a function of p_t, with the parameters $\beta = 0.2$, $q = 0.5$, $p_s = 1$, $b_{\text{max}} = 15$,

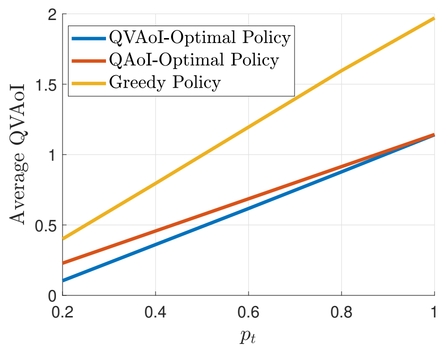

Fig. 6: Average QVAoI vs. p_t.

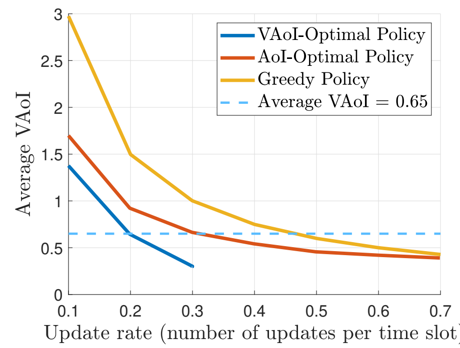

Fig. 8: Average VAoI vs. update rate.

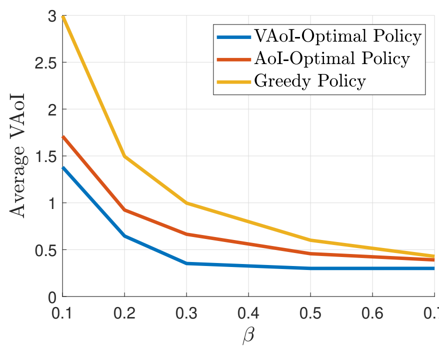

Fig. 7: Average VAoI vs. β.

and $\Delta_{\max} = 19$. Here, the semantics-aware policies also demonstrate superior performance compared to the greedy policy, and the QVAoI-optimal policy outperforms the QAoI-optimal policy for lower values of p_t.

These results emphasize that *incorporating semantics-aware metrics, leads to a more effective status updating policy concerning the freshness and significance of information within the system, compared to the greedy policy* (see the red and blue curves vs. the orange curve). Moreover, *the utilization of the VAoI and QVAoI metrics yield enhanced status updating policies concerning the freshness and significance of information, compared to the AoI and QAoI metrics, particularly in scenarios where the source versions evolve slowly, i.e., when p_t is low* (see the blue curve vs. the red curve). It is also evident that the VAoI-Optimal and QVAoI-Optimal policies align with the AoI-Optimal and QAoI-Optimal policies, respectively, when $p_t = 1$. In other words, *VAoI and QVAoI represent more general semantic metrics, reducing to AoI and VAoI when the source's version changes in each time slot.*

B. The Impact of Energy Arrival Probability β

1) VAoI-optimal vs. AoI-optimal policy in optimizing VAoI: We evaluate the average VAoI of the system across various policies by varying β within the range of 0.1 to 0.7, while setting the parameters $p_t = 0.3$, $q = 1$, $p_s = 1$, $b_{\max} = 15$, and $\Delta_{\max} = 19$. The results are depicted in Fig. 7. First, it can be observed that an increase in energy arrival probability improves the performance of all three policies, as it provides more energy for the device to send more frequent updates with a higher degree of freedom. For a high energy arrival rate, the three policies perform comparably well. However, as β decreases and the system experiences more severe energy limitations, the semantics-aware policies outperform the greedy policy by a considerable margin. Moreover, it is evident that the VAoI-Optimal policy performs better than the AoI-Optimal policy in optimizing the average VAoI in the system.

This enhancement becomes more tangible when comparing the average number of updates needed by each policy to maintain the same level of average VAoI. This comparison is illustrated in Fig. 8, where the average VAoI is plotted against the update rate, i.e., the average number of updates per slot. In this figure, the horizontal dashed line represents an average VAoI level of 0.65. The intersection point of this line with the outcomes of the three policies determines the average update rate needed by each to maintain a performance level of 0.65 in terms of average VAoI. As can be seen, the VAoI-Optimal policy requires an update rate of only 0.2, whereas the AoI-Optimal policy and the greedy policy require update rates of 0.31 and 0.48, respectively. This demonstrates improvements of 55% and 140% in average update rate, correspondingly.

Another result that can be inferred from Fig. 8 is that, under the VAoI-Optimal policy, the average update rate per slot never exceeds the level of p_t (0.3 in the figure). In other words, according to the VAoI-Optimal policy, the update rate per slot is maximized at p_t, unless constrained by energy arrivals.

2) QVAoI-optimal vs. QAoI-optimal policy in optimizing QVAoI: In Figures 9 and 10, the average QVAoI for QVAoI-Optimal, QAoI-Optimal, and greedy policies is illustrated as a function of β and update rate, respectively, with $p_t = 0.3$, $q = 0.5$, $p_s = 1$, $b_{\max} = 15$, and $\Delta_{\max} = 19$. In Fig. 9, we observe that both semantics-aware policies outperform the greedy policy, while the QVAoI-Optimal policy shows superior performance compared to the QAoI-Optimal policy. The average QVAoI is depicted as a function of the update rate in Fig. 10, where the semantics-aware policies prove to require fewer updates to maintain a specific level of average QVAoI in the system. For instance, to achieve an average QVAoI of 0.25, the QVAoI-Optimal policy results in an update rate of 0.19, while QAoI-Optimal and greedy policies require 0.28 and 0.38, respectively. This demonstrates an improvement of 47% and 100% in the average update rate. In this pull-based scenario, it is evident that the average update rate of the three

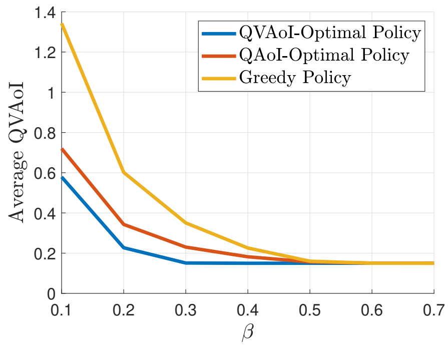

Fig. 9: Average QVAoI vs. β.

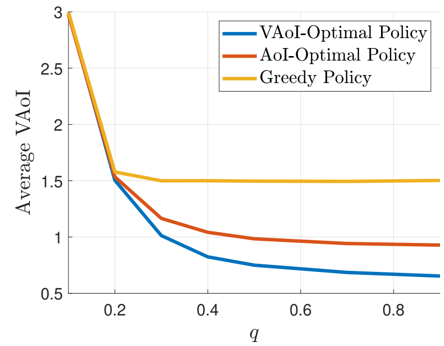

Fig. 11: Average VAoI vs. q.

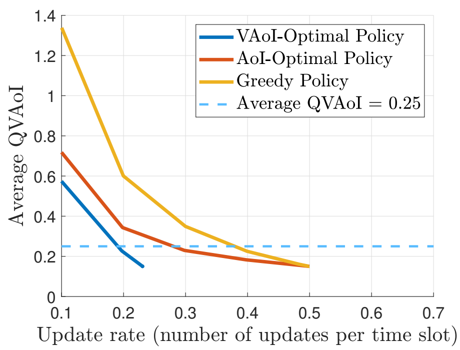

Fig. 10: Average QVAoI vs. update rate.

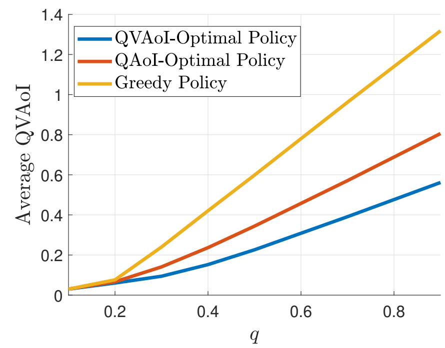

Fig. 12: Average QVAoI vs. q.

policies never surpasses the q value. Notably, QVAoI-Optimal exhibits the best performance with a lower update rate.

These results signify that *in a semantics-aware communication system, we can reduce the number of updates and thus the costs without compromising the conveyed information.* This is particularly important because reducing the number of transmissions in an Energy Harvesting IoT system leads to a significant improvement in energy efficiency.

C. The Impact of Request Arrival Probability q

1) VAoI-optimal vs. AoI-optimal policy in optimizing VAoI: The average VAoI of the system as a function of q is depicted in Fig. 11, with parameters $\beta = 0.2$, $p_t = 0.3$, $p_s = 1$, $b_{\text{max}} = 15$, and $\Delta_{\text{max}} = 19$. In this figure, we observe that q acts as a limitation in the system's performance. For low request arrival probabilities, less than β, all three policies perform poorly and similarly. However, for high values of q, the semantics-aware policies perform better, with the VAoI-Optimal policy demonstrating superior performance compared to the AoI-Optimal policy. This result indicates that a pull-based scenario does not enhance the average VAoI as long as the updates are always valuable to the receiver, or the receiver is always ready to utilize them. In fact, The pull-based scenario is advantageous when there is a limitation on the utilization or value of updates on the receiver side, as will be discussed in the next section.

2) QVAoI-optimal vs. QAoI-optimal policy in optimizing QVAoI: In Fig. 12, the average QVAoI is depicted as a function of q. We have fixed the parameters $\beta = 0.2$, $p_t = 0.3$, $p_s = 1$, $b_{\text{max}} = 15$, and $\Delta_{\text{max}} = 19$. It can be seen that the average QVAoI increases with q for all three policies, with the QAoI-optimal policy demonstrating superior performance. The reason behind this increasing behavior is noteworthy. The first reason is that the query process $r(t)$ emerges as a weight in the objective function of the MDP problem (6). As the probability of query arrivals increases, the expected value in this objective function increases. However, even after normalizing the objective function to the expected value of query arrivals, i.e., $\lim_{T\to\infty} \frac{1}{T} E\left[\sum_{t=0}^{T-1} r(t)\right] = q$, the increasing behavior persists (see Fig. 13). This can be explained by noting that for higher demands when the probability of query arrival increases, there are more time slots in which the device must decide on transmitting. Given the fixed energy arrival rate, the device should set higher thresholds on VAoI values at query instances to limit the number of updates to the maximum permitted by the energy constraint. Consequently, according to this policy, with higher thresholds, there would be more query instances followed by no action, resulting in an increase in average QVAoI. Therefore, increasing pressure on the device and issuing extra requests (more than 0.3 in Fig. 13) cannot improve the system's average QVAoI. However, concerning the average VAoI, as the probability of query arrivals increases, updates occur frequently enough to reduce the average VAoI, as illustrated in Fig. 11.

D. The Impact of Channel Success Probability p_s

In Figs. 14 and 15, the average VAoI and average QVAoI of the system are depicted as functions of p_s. We have

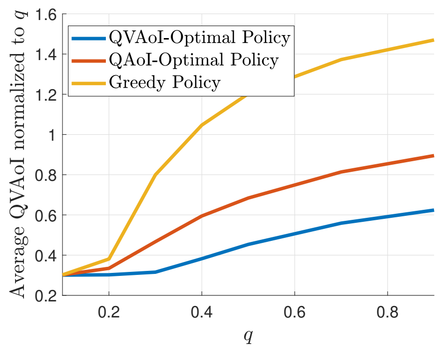

Fig. 13: Average QVAoI (normalized to q) vs. q.

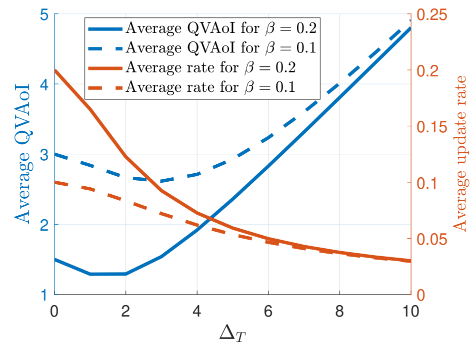

Fig. 16: Average QVAoI and update rate for single-sized battery vs. Δ_T.

fixed the parameters $\beta = 0.2$, $p_t = 0.3$, $b_{\max} = 15$, and $\Delta_{\max} = 19$, while $q = 1$ and $q = 0.5$, respectively. As expected, an increase in channel success probability improves the performance of the system across different policies, with the VAoI-Optimal and QVAoI-Optimal policies demonstrating the best performance.

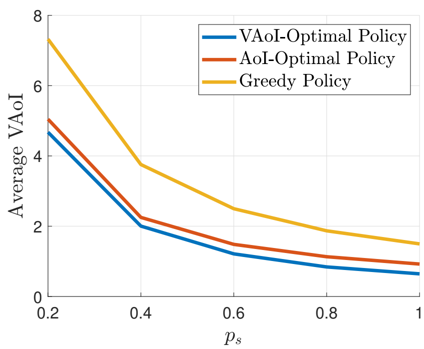

Fig. 14: Average VAoI for different policies vs. p_s.

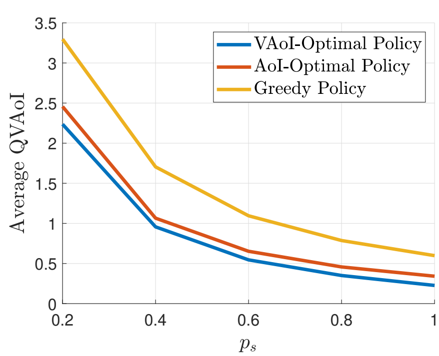

Fig. 15: Average QVAoI for different policies vs. p_s.

E. Average QVAoI for the Single-Sized Battery

We derived the closed-form equations for the average QVAoI of a threshold policy in a status update system with $q = 1$ and $b_{\max} = 1$ in Section IV-A. In Fig. 16, the average QVAoI is depicted along the left vertical axis versus different threshold levels Δ_T and for two values of energy arrival probability β, where $p_t = 0.3$. It can be observed that for $\beta = 0.2$ (shown with the solid blue curve), the minimum average QVAoI is obtained by setting the threshold Δ_T equal to 1. However, by reducing β to 0.1 (the dashed blue curve), the optimal threshold becomes 3. Here, two conclusions can be drawn. First, the semantics-aware policies outperform the greedy policy, i.e., when $\Delta_T = 0$. Secondly, it is an intriguing finding that, *in scenarios with highly restricted energy arrivals, sending fewer updates results in a fresher system.* This becomes clearer when we compare the update rate per time slot for the optimal policy with a higher threshold to that of a greedy policy with threshold zero, as depicted along the right vertical axis in Fig. 16 in red. For instance, when $\beta = 0.1$, the update rate corresponding to the optimal policy with a threshold of 3 is 0.72, while the greedy policy consumes all the arrived energy and transmits updates as frequently as possible, with an average rate of 0.1. This occurs while the average QVAoI for the optimal policy is 2.61 and for the greedy policy is 3. This demonstrates that the *semantics-aware policy results in fewer updates and a fresher system* simultaneously.

VI. Conclusion

In this study, we addressed the optimization of freshness and significance of information in a status update system wherein an EH device was tasked with scheduling the transmission of measured update packets from an information source to a destination node. We introduced a semantics-aware metric, QVAoI, and identified the QVAoI-Optimal, QAoI-Optimal, VAoI-Optimal, and AoI-Optimal policies by formulating and solving MDP problems. Through comparison with a greedy policy, we demonstrated that these semantics-aware policies delivered superior performance regarding the freshness and significance of information. Moreover, we illustrated that the QVAoI-Optimal and VAoI-Optimal policies can achieve fresher and more significant updates from the device or reduce the number of transmissions without compromising the freshness and significance of information compared to the QAoI-Optimal and AoI-Optimal policies, respectively.

References

[1] R. D. Yates, Y. Sun, D. R. Brown, S. K. Kaul, E. Modiano, and S. Ulukus, "Age of information: An introduction and survey," *IEEE Journal on Selected Areas in Commun.*, vol. 39, no. 5, 2021.

[2] M. A. Abd-Elmagid, N. Pappas, and H. S. Dhillon, "On the role of age of information in the internet of things," *IEEE Commun. Magazine*, vol. 57, no. 12, 2019.

[3] M. Kountouris and N. Pappas, "Semantics-empowered communication for networked intelligent systems," *IEEE Commun. Magazine*, vol. 59, no. 6, 2021.

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

[5] X. Zheng, S. Zhou, Z. Jiang, and Z. Niu, "Closed-form analysis of non-linear age of information in status updates with an energy harvesting transmitter," *IEEE Trans. on Wireless Commun.*, vol. 18, no. 8, 2019.

[6] A. Maatouk, S. Kriouile, M. Assaad, and A. Ephremides, "The age of incorrect information: A new performance metric for status updates," *IEEE/ACM Trans. on Networking*, vol. 28, no. 5, 2020.

[7] F. Chiariotti, J. Holm, A. E. Kalør, B. Soret, S. K. Jensen, T. B. Pedersen, and P. Popovski, "Query age of information: Freshness in pull-based communication," *IEEE Trans. on Commun.*, vol. 70, no. 3, 2022.

[8] R. D. Yates, "The age of gossip in networks," in *2021 IEEE ISIT*, 2021.

[9] E. Delfani, G. J. Stamatakis, and N. Pappas, "State-aware timeliness in energy harvesting iot systems monitoring a markovian source," *arXiv preprint arXiv:2405.03628*, 2024.

[10] 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 Commun.*, vol. 197, 2023.

[11] N. Pappas, M. A. Abd-Elmagid, B. Zhou, W. Saad, and H. S. Dhillon, *Age of Information: Foundations and Applications*. Cambridge University Press, 2022.

[12] B. Buyukates, M. Bastopcu, and S. Ulukus, "Version age of information in clustered gossip networks," *IEEE Journal on Selected Areas in Information Theory*, vol. 3, no. 1, 2022.

[13] P. Kaswan and S. Ulukus, "Timely gossiping with file slicing and network coding," in *IEEE ISIT*, 2022.

[14] ——, "Susceptibility of age of gossip to timestomping," in *IEEE Information Theory Workshop (ITW)*, 2022.

[15] P. Mitra and S. Ulukus, "Asuman: Age sense updating multiple access in networks," in *58th Annual Allerton Conference on Communication, Control, and Computing*, 2022.

[16] P. Kaswan and S. Ulukus, "Age of gossip in ring networks in the presence of jamming attacks," in *56th Asilomar Conference on Signals, Systems, and Computers*, 2022.

[17] P. Mitra and S. Ulukus, "Age-aware gossiping in network topologies," *IEEE Trans. on Commun.*, 2024.

[18] E. Delfani and N. Pappas, "Version age-optimal cached status updates in a gossiping network with energy harvesting sensor," in *2023 21st WiOpt Conference*, 2023.

[19] P. Mitra and S. Ulukus, "A learning based scheme for fair timeliness in sparse gossip networks," *arXiv preprint arXiv:2310.01396*, 2023.

[20] M. A. Abd-Elmagid and H. S. Dhillon, "Distribution of the age of gossip in networks," *Entropy*, vol. 25, no. 2, 2023.

[21] M. Hatami, M. Leinonen, Z. Chen, N. Pappas, and M. Codreanu, "On-demand aoi minimization in resource-constrained cache-enabled iot networks with energy harvesting sensors," *IEEE Trans. on Commun.*, vol. 70, no. 11, 2022.

[22] M. E. Ildiz, O. T. Yavascan, E. Uysal, and O. T. Kartal, "Pull or wait: How to optimize query age of information," *IEEE Journal on Selected Areas in Information Theory*, 2023.

[23] T. K. Oğuz, E. T. Ceran, E. Uysal, and T. Girici, "Implementation and evaluation of age-aware downlink scheduling policies in push-based and pull-based communication," *Entropy*, vol. 24, no. 5, 2022.

[24] P. Agheli, N. Pappas, P. Popovski, and M. Kountouris, "Effective communication: When to pull updates?" in *IEEE ICC*, 2024.

[25] D. P. Bertsekas, *Dynamic Programming and Optimal Control, Vol. II*, 3rd ed. Athena Scientific, 2007.

Appendix A
Proof of Theorem 1

Proof. The Bellman equation at state $s=(b,\Delta,r)$ is given by:

$$J^*+V(s)=\min_{a\in\{0,1\}}\underbrace{\left\{\sum_{s'\in S}P\left[s'\middle|s,a\right]\left(r\Delta'+V(s')\right)\right\}}_{\triangleq\, Q(a)} \quad (27)$$

$$a^*(s)=\arg\min_{a\in\{0,1\}}Q(a)=\begin{cases}0, & DV(s)\geq 0,\\ 1, & DV(s)<0,\end{cases} \quad (28)$$

where $DV(s)\triangleq V^1(s)-V^0(s)$, $V^0(s)\triangleq Q(a=0)$, and $V^1(s)\triangleq Q(a=1)$.

As can be seen, the optimal action $a^*(s)$ is related to the sign of $DV(s)$. When $b=0$ or $r=0$, the action $a=0$ is forced and $DV(s)=0$. For other cases where $b>0$ and $r=1$, from Section III-C we have:

$$\begin{aligned}&V^0(s)=\textstyle\sum_{s'\in S}P\left[s'\middle|s,a=0\right]\left(\Delta'+V(s')\right)\\&=\textstyle\sum_{r'\in\{0,1\}}\sum_{\substack{z\in\{0,1\}\\e\in\{0,1\}}}\left\{\left(\Delta+z\right)+V(b+e,\Delta+z,r')\right\}P_eP_zP_{r'},\\&V^1(s)=\textstyle\sum_{s'\in S}P\left[s'\middle|s,a=1\right]\left(\Delta'+V(s')\right)\\&=\textstyle\sum_{r'\in\{0,1\}}\sum_{\substack{z\in\{0,1\}\\e\in\{0,1\}}}\Big\{\bar{p}_s\left[(\Delta+z)+V(b+e-1,\Delta+z,r')\right]\\&+p_s\left[z+V(b+e-1,z,r')\right]\Big\}P_eP_zP_{r'}.\end{aligned} \quad (29)$$

In what follows, we demonstrate that $DV(s)=DV(b,\Delta,r)$ is a decreasing (non-increasing) function of Δ, i.e., for $\Delta^-\leq\Delta^+$, we show that $DV(b,\Delta^+,r)\leq DV(b,\Delta^-,r)$ or $DV(b,\Delta^+,r)-DV(b,\Delta^-,r)\leq 0$. This results in the threshold policy because if $DV(s)$ is negative for a Δ_T, it will also be negative for $\Delta\geq\Delta_T$, and the optimal action remains 1. By simplification of $DV(b,\Delta^+,r)$ and $DV(b,\Delta^-,r)$ based on (29) we obtain the following equations:

$$\begin{aligned}&DV(b,\Delta^+,r)-DV(b,\Delta^-,r)\\&=V^1(b,\Delta^+,r)-V^1(b,\Delta^-,r)-\left[V^0(b,\Delta^+,r)-V^0(b,\Delta^-,r)\right]\\&=\textstyle\sum_{r',z,e}\Big\{\overbrace{p_s\left(\Delta^--\Delta^+\right)}^{\leq 0}\\&\quad+\bar{p}_s\Big[V(b+e-1,\Delta^++z,r')-V(b+e-1,\Delta^-+z,r')\Big]\\&\quad-\Big[V(b+e,\Delta^++z,r')-V(b+e,\Delta^-+z,r')\Big]\Big\}P_eP_zP_{r'}\end{aligned}$$

Therefore, to verify the inequality $DV(b,\Delta^+,r)-DV(b,\Delta^-,r)\leq 0$, it is sufficient to show that

$$\begin{aligned}&\bar{p}_s\left[V(b-1,\Delta^+,r)-V(b-1,\Delta^-,r)\right]\\&\qquad-\left[V(b,\Delta^+,r)-V(b,\Delta^-,r)\right]\leq 0,\end{aligned}$$

for $b>0$ and $\Delta^-\leq\Delta^+$. To proceed with the proof, we use the VIA and mathematical induction. VIA converges to the value function of Bellman's equation regardless of the initial value of $V_0(s)$, i.e., $\lim_{k\to\infty}V_k(s)=V(s),\ \forall s\in S$.

$$V_{k+1}(s)=\min_{a\in\{0,1\}}\left\{\sum_{s'\in S}P\left[s'\middle|s,a\right]\left(r\Delta'+V_k(s')\right)\right\}. \quad (30)$$

Therefore, it is sufficient to prove the following inequality for all $k \in \{0,1,2,\cdots\}$:

$$\bar{p}_s\big[V_k(b-1,\Delta^+,r)-V_k(b-1,\Delta^-,r)\big] \\ -\big[V_k(b,\Delta^+,r)-V_k(b,\Delta^-,r)\big] \le 0. \quad (31)$$

Assuming $V_0(s)=0$ for all $s\in S$, equation (31) is true for $k=0$. Now, by extending assumption (31) for $k>0$, we aim to prove its validity for $k+1$, i.e.,

$$\bar{p}_s\big[V_{k+1}(b-1,\Delta^+,r) - V_{k+1}(b-1,\Delta^-,r)\big] \\ - \big[V_{k+1}(b,\Delta^+,r) - V_{k+1}(b,\Delta^-,r)\big] \le 0. \quad (32)$$

the VIA equation (30) is given by $V_{k+1}(s) = \min\{V^0_{k+1}(s), V^1_{k+1}(s)\}$, by defining:

$$V^0_{k+1}(s) \triangleq \textstyle\sum_{s'\in S} P\big[s'\big|s,a=0\big]\big(r\Delta'+V_k(s')\big), \\ V^1_{k+1}(s) \triangleq \textstyle\sum_{s'\in S} P\big[s'\big|s,a=1\big]\big(r\Delta'+V_k(s')\big), \quad (33)$$

where

$$V^0_{k+1}(s)=\textstyle\sum_{r',z,e}\Big\{\big(\Delta+z\big)+V_k(b+e,\Delta+z,r')\Big\}P_eP_zP_{r'}, \\ V^1_{k+1}(s)=\textstyle\sum_{r',z,e}\Big\{\bar{p}_s\big[\,(\Delta+z)+V_k(b+e-1,\Delta+z,r')\big] \\ +p_s\big[z+V_k(b+e-1,z,r')\big]\Big\}P_eP_zP_{r'}. \quad (34)$$

The inequality (32) can further be simplified:

$$\bar{p}_s\Big[\min\{V^0_{k+1}(b-1,\Delta^+,r),V^1_{k+1}(b-1,\Delta^+,r)\} \quad (35) \\ -\min\{V^0_{k+1}(b-1,\Delta^-,r),V^1_{k+1}(b-1,\Delta^-,r)\}\Big] \\ -\Big[\min\{V^0_{k+1}(b,\Delta^+,r),V^1_{k+1}(b,\Delta^+,r)\} \\ -\min\{V^0_{k+1}(b,\Delta^-,r),V^1_{k+1}(b,\Delta^-,r)\}\Big] \le 0$$

We consider four cases to proceed with the proof of (35).

$$\text{Case 1.} \begin{cases} V^0_{k+1}(b-1,\Delta^-,r) \le V^1_{k+1}(b-1,\Delta^-,r), \\ V^0_{k+1}(b,\Delta^+,r) \le V^1_{k+1}(b,\Delta^+,r). \end{cases}$$
$$\text{Case 2.} \begin{cases} V^0_{k+1}(b-1,\Delta^-,r) \le V^1_{k+1}(b-1,\Delta^-,r), \\ V^0_{k+1}(b,\Delta^+,r) > V^1_{k+1}(b,\Delta^+,r). \end{cases}$$
$$\text{Case 3.} \begin{cases} V^0_{k+1}(b-1,\Delta^-,r) > V^1_{k+1}(b-1,\Delta^-,r), \\ V^0_{k+1}(b,\Delta^+,r) \le V^1_{k+1}(b,\Delta^+,r). \end{cases}$$
$$\text{Case 4.} \begin{cases} V^0_{k+1}(b-1,\Delta^-,r) > V^1_{k+1}(b-1,\Delta^-,r), \\ V^0_{k+1}(b,\Delta^+,r) > V^1_{k+1}(b,\Delta^+,r). \end{cases}$$

We prove the inequality (35) for case 1; a similar approach can be utilized to prove the other cases.

Case 1. $V^0_{k+1}(b-1,\Delta^-,r) \le V^1_{k+1}(b-1,\Delta^-,r)$ and $V^0_{k+1}(b,\Delta^+,r) \le V^1_{k+1}(b,\Delta^+,r)$. In this case, equation (35) is further simplified:

$$\bar{p}_s\Big[V^0_{k+1}(b-1,\Delta^+,r)-V^0_{k+1}(b-1,\Delta^-,r)\Big] \\ +\underbrace{\bar{p}_s\min\{0,V^1_{k+1}(b-1,\Delta^+,r)-V^0_{k+1}(b-1,\Delta^+,r)\}}_{\le 0} \\ -\Big[V^0_{k+1}(b,\Delta^+,r)-V^0_{k+1}(b,\Delta^-,r)\Big] \\ +\underbrace{\min\{0,V^1_{k+1}(b,\Delta^-,r)-V^0_{k+1}(b,\Delta^-,r)\}}_{\le 0} \le 0, \quad (36)$$

where we have used $\min\{x,y\} = x+\min\{0,y-x\}$. The second and last terms are negative (non-positive), thus it suffices to show that:

$$\bar{p}_s\Big[V^0_{k+1}(b-1,\Delta^+,r) - V^0_{k+1}(b-1,\Delta^-,r)\Big] \\ -\Big[V^0_{k+1}(b,\Delta^+,r) - V^0_{k+1}(b,\Delta^-,r)\Big] \le 0. \quad (37)$$

According to (34), it can be written as follows:

$$\bar{p}_s\textstyle\sum_{r',z,e}\Big\{\big(\Delta^+-\Delta^-\big)+V_k(b+e-1,\Delta^++z,r') \\ -V_k(b+e-1,\Delta^-+z,r')\Big\}P_eP_zP_{r'} \\ -\textstyle\sum_{r',z,e}\Big\{\big(\Delta^+-\Delta^-\big)+V_k(b+e,\Delta^++z,r') \\ -V_k(b+e,\Delta^-+z,r')\Big\}P_eP_zP_{r'} \le 0 \\ \Leftrightarrow \textstyle\sum_{r',z,e}\Big\{\overbrace{(1-\bar{p}_s)(\Delta^--\Delta^+)}^{\le 0} \\ +\bar{p}_s\big[V_k(b+e-1,\Delta^++z,r') - V_k(b+e-1,\Delta^-+z,r')\big] \\ -\big[V_k(b+e,\Delta^++z,r')-V_k(b+e,\Delta^-+z,r')\big]\Big\}P_eP_zP_{r'} \le 0$$

where the first term in the summation is negative since $\Delta^- \le \Delta^+$ and $1-\bar{p}_s = p_s > 0$. The remaining terms are also negative according to the induction assumption (31), and the proof is complete. □

Appendix B
Proof of Theorem 2

Proof. The balance equation at each state for the Markov chain in Figs. 2, 3, and 4 can be directly written as follows:

- Case 1: $\Delta_T = 0$.

$$\begin{cases} \mu(0,0)=\gamma_{00}\mu(0,0)+\gamma_{00}\sum_{\delta=0}^{\infty}\mu(1,\delta), \\ \mu(0,1)=\gamma_{10}\mu(0,0)+\gamma_{00}\mu(0,1)+\gamma_{10}\sum_{\delta=0}^{\infty}\mu(1,\delta), \\ \mu(0,\Delta)=\gamma_{00}\mu(0,\Delta)+\gamma_{10}\mu(0,\Delta-1), \quad \Delta\ge 2, \\ \mu(1,0)=\gamma_{01}\mu(0,0)+\gamma_{01}\sum_{\delta=0}^{\infty}\mu(1,\delta), \\ \mu(1,1)=\gamma_{11}\mu(0,0)+\gamma_{01}\mu(0,1)+\gamma_{11}\sum_{\delta=0}^{\infty}\mu(1,\delta), \\ \mu(1,\Delta)=\gamma_{11}\mu(0,\Delta-1)+\gamma_{01}\mu(0,\Delta), \quad \Delta\ge 2. \end{cases} \quad (38)$$

From the first equation (38), we have $\sum_{\delta=0}^{\infty}\mu(1,\delta) = \frac{1-\gamma_{00}}{\gamma_{00}}\mu(0,0)$. After some mathematical manipulation, all other stationary probabilities can also be defined as a function of $\mu(0,0)$. By substituting them into $\sum_{b\in\{0,1\}}\sum_{\Delta\in\{0,1,2,\cdots\}}\mu(b,\Delta)=1$, we obtain $\mu(0,0)$ and subsequently all the state-stationary probabilities $\mu(b,\Delta)$.

$$\mu(0,0) = \frac{\bar{p}_t\beta\bar{\beta}}{\beta+p_t\bar{\beta}}. \quad (39)$$

Finally, the average QVAoI can be determined using Equation (22), as stated in Theorem 2.

- Case 2: $\Delta_T = 1$.

$$\begin{cases}\mu(0,0)=\gamma_{00}\mu(0,0)+\gamma_{00}\sum_{\delta=1}^{\infty}\mu(1,\delta),\\ \mu(0,1)=\gamma_{10}\mu(0,0)+\gamma_{00}\mu(0,1)+\gamma_{10}\sum_{\delta=1}^{\infty}\mu(1,\delta),\\ \mu(0,\Delta)=\gamma_{00}\mu(0,\Delta)+\gamma_{10}\mu(0,\Delta-1), \quad \Delta\geq 2,\\ \mu(1,0)=\gamma_{01}\mu(0,0)+\bar{p}_t\mu(1,0)+\gamma_{01}\sum_{\delta=1}^{\infty}\mu(1,\delta),\\ \mu(1,1)=\gamma_{11}\mu(0,0)+\gamma_{01}\mu(0,1)\\ \qquad\qquad +p_t\mu(1,0)+\gamma_{11}\sum_{\delta=1}^{\infty}\mu(1,\delta),\\ \mu(1,\Delta)=\gamma_{11}\mu(0,\Delta-1)+\gamma_{01}\mu(0,\Delta), \quad \Delta\geq 2.\end{cases} \tag{40}$$

Following the same approach as in Case 1, the stationary probabilities and the average QVAoI will be determined.

$$\mu(0,0)=\frac{p_t\bar{p}_t\beta\bar{\beta}}{\beta^2+p_t\bar{\beta}(p_t+\beta)}. \tag{41}$$

- Case 3: $\Delta_T \geq 2$.

$$\begin{cases}\mu(0,0)=\gamma_{00}\mu(0,0)+\gamma_{00}\sum_{\delta=\Delta_T}^{\infty}\mu(1,\delta),\\ \mu(0,1)=\gamma_{10}\mu(0,0)+\gamma_{00}\mu(0,1)+\gamma_{10}\sum_{\delta=\Delta_T}^{\infty}\mu(1,\delta),\\ \mu(0,\Delta)=\gamma_{00}\mu(0,\Delta)+\gamma_{10}\mu(0,\Delta-1), \quad \Delta\geq 2,\\ \mu(1,0)=\gamma_{01}\mu(0,0)+\bar{p}_t\mu(1,0)+\gamma_{01}\sum_{\delta=\Delta_T}^{\infty}\mu(1,\delta),\\ \mu(1,1)=\gamma_{11}\mu(0,0)+\gamma_{01}\mu(0,1)+p_t\mu(1,0)\\ \qquad\qquad +\bar{p}_t\mu(1,1)+\gamma_{11}\sum_{\delta=\Delta_T}^{\infty}\mu(1,\delta),\\ \mu(1,\Delta)=\gamma_{11}\mu(0,\Delta-1)+\gamma_{01}\mu(0,\Delta)\\ \qquad\qquad +p_t\mu(1,\Delta-1)+\bar{p}_t\mu(1,\Delta), \quad 2\leq\Delta<\Delta_T,\\ \mu(1,\Delta_T)=\gamma_{11}\mu(0,\Delta_T-1)+\gamma_{01}\mu(0,\Delta_T)\\ \qquad\qquad +p_t\mu(1,\Delta_T-1),\\ \mu(1,\Delta)=\gamma_{11}\mu(0,\Delta-1)+\gamma_{01}\mu(0,\Delta), \quad \Delta\geq\Delta_T+1.\end{cases} \tag{42}$$

Similar to Cases 1 and 2, the stationary probabilities and the average QVAoI can be determined, concluding the proof of Theorem 2.

$$\mu(0,0)=\frac{p_t\bar{p}_t\beta\bar{\beta}}{(\beta+p_t\bar{\beta})\left[\beta\Delta_T+p_t\left(\frac{p_t\bar{\beta}}{\beta+p_t\bar{\beta}}\right)^{\Delta_T}\right]}. \tag{43}$$

□