Instant AoI Optimization in IoT Networks with Packet Combination

Jiadong Lou, Xu Yuan, and Nian-Feng Tzeng
University of Louisiana at Lafayette, Lafayette, Louisiana, USA

***Abstract*—This paper studies the freshness of data delivery, measured by the recently proposed Age of Information (AoI) metric, in Internet of Things (IoT) networks. Given IoT networks with plenty of edge devices to upload their sealed packets, re-packing multiple packets into one at each sink node by removing redundant packet headers could significantly improve transmission efficiency. We investigate such packet combination behaviors in transmitting the monitored/collected data from sensing nodes to accelerate data delivery, enabling the IoT edge server to acquire the latest updates timely. Two data acquisition modes, i.e., *Periodic Request* and *Proactive Request*, at the IoT edge server are considered. Under each mode, we derive the AoI formula, develop mathematical modeling, formulate the problem, and propose a low-complexity scheduling algorithm by leveraging packet combination with an aim to minimize the Instant AoI at the edge server. Through numerical results, we demonstrate the advantages of packet combination behaviors for AoI performance improvement.**

I. Introduction

Monitoring environmental changes or collecting data from sensing nodes plus timely uploading them to the edge server for upper layer applications have been the central theme of Internet of Things (IoT) networks, such as house monitoring in the smart home, danger detection in the smart plant, among others. These applications highly rely on the latest updates, making it absolutely important to promote freshness in updated data delivery, which has received much attention lately. Recently proposed *Age of Information* (AoI) [1] has been a new emerging and the most suitable performance metric that can measure information freshness from the standpoint of the IoT edge server. Defined as the time elapsed since the generation time of the latest arrival packet at a destination node, AoI can characterize the level of timely information delivery at a receiver side. In contrast to such conventional measures as delay and throughput, which can only capture the effectiveness of data collection and transmission in a network as a whole, AoI aims to quantify the timeliness of updates from a destination perspective.

Extensive work has dealt with the theoretical foundations of AoI, including packet generation rate control [1], [2], [3], queue management [4], [5], [6], and scheduling policy [7], [8], [9], [10]. Meanwhile, some practical settings or constraints have been taken into consideration when exploring AoI in real-world applications. They considered practical constraints including throughput [11], interference [12], and channel access [13], [14], [15]. Moreover, [16] has studied AoI under the setting where a mobile agent can traverse the ground terminals for data collection, whereas [17] has considered the request and response behaviors between users and the data distributor. Recent work extended AoI into IoT networks and network edge, with [18] aiming to minimize AoI under an averaged energy cost constraint at the IoT device and [19] pursuing a general AoI model for the sampling behaviors at the network edge. Although existing work has expanded the theoretical exploration of AoI into practical applications to some extent, plenty of important settings or application scenarios in IoT networks have not been well addressed yet.

In this paper, we study the AoI optimization problem in a practical IoT network scenario, where a sink node gathers the information updates from sensing nodes and then uploads them to the edge server in support of upper layer applications. To speed up data transmission from the sink node to the edge server, we consider the packet combination behavior at the sink node, by dropping redundant information in the headers of multiple packets (from sensing nodes) and re-packing useful information (including the latest updates) in payloads into a single packet for delivery, in a way similar to earlier packet combining in multi-stage interconnection networks for traffic hotspot avoidance [20].

We consider two application scenarios in practice for data acquisition at the edge server, i.e., *Periodic Request* and *Proactive Request*. In *Periodic Request* mode, the sink node periodically re-packs the latest updates from all sensing nodes for delivery to the edge server. In *Proactive Request* mode, the edge server makes a data request at an arbitrary time, whereas the sink node makes a fast decision on the latest packet delivery to the edge server within a given time interval. To meet the application needs, we introduce a new performance metric, *Instant AoI*, for capturing AoI at the time point when the sink node finishes each delivery, to better measure data freshness. In both modes, we propose different packet combination strategies, formulate the optimization problems, and design low-complexity algorithms with the aim to minimize Instant AoI at the IoT edge server. The simulation results demonstrate that our proposed packet combination strategies can achieve the minimum Instant AoI, serving to precisely measure the freshness of received data at the edge server.

The remainder of this paper is organized as follows. Section II describes the network model and states our problem. In Section III, we present the mathematical modeling of

For correspondence, please contact Prof. Xu Yuan (xu.yuan@louisiana.edu).

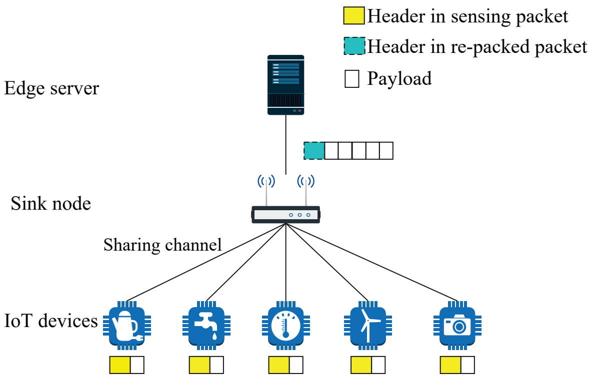

Fig. 1. An example of the IoT network.

packet combination and Instant AoI. In Sections IV and V, we analyze the Instant AoI variation in the periodic request and the proactive request modes, respectively, and present the corresponding low-complexity scheduling algorithms to find the optimal Instant AoI. Section VI provides our numerical results and Section VIII concludes our paper.

II. Network Model and Problem Statement

We consider an Internet of Things (IoT) application scenario comprising a set of sensing nodes $\mathcal{N}$, a sink node S, and an Edge Server (ES), as shown in Figure 1. Sensing nodes include different categories of IoT devices, for monitoring environmental changes and collecting the respective information updates, such as temperature, humidity, and others. Collected information is sampled and packed for wireless transmission to the sink node via a shared channel in a single hop. Assume the sink node also uses the same channel to transmit received updates to the ES. To schedule the transmission among all sensing nodes and the sink node, we consider the TDMA modulation scheme, which divides the time equally into a set of time slots to support interference-free transmission. All sensing nodes pack the monitored data into the same-sized packets, with each fitting to one time slot for transmission. That is, during a time slot, one sensing node can pack the new update into one packet that records instantly monitored information while delivering this packet to the sink node within the current time slot. The propagation and the transmission delay of each packet are both capsuled in one time slot.

The sink node acts as the gateway and is responsible for intermediately transmitting packets collected from sensing nodes to the ES in support of certain upper layer applications. For each updated packet from a sensing node, two fields are included, i.e., packet header and payload. The header field has the length of at least 128 bits, whereas the payload may contain only a simple description of the monitored updates to occupy a little space. As a result, multiple packets with the same header suffer from inefficient packet delivery as the relatively longer header dominates the packet transmission time. This concern on inefficient packet delivery is mitigated greatly if the sink node can pack multiple updated packets into one by removing redundant packet headers to improve the transmission capability dramatically. This capability improvement results from that fewer time slots involved in delivering the collected updates. In this model, the ES supports some upper layer applications, seeking as best timely fresh data delivery as possible to meet specific tasks. After receiving updated packets from the sink node, the ES updates its maintained local data of each sensing node.

The behaviors of re-packing updated packets depend on specific application needs from upper layers, which can be generally categorized as follows: 1) *Periodic Request*. The upper layer applications request information collected from sensing nodes periodically to meet their business needs. This is the most common form of data collection in IoT networks. For example, in a smart factory, the monitoring system periodically records the log of factory environmental changes collected from all deployed sensing nodes. In this delivery mode, we assume the sink node packs the latest packets received from all sensing nodes and then delivers the combined packet to the ES once in every T time slots. Since transmission behaviors of the sink node are static, scheduling is performed mainly among the sensing nodes. 2) *Proactive Request*. The ES supports different types of real-time applications, allowing the upper layer to request information at will when needed. For example, a smart home monitoring system may request the freshest information from the sink node at an arbitrary time. Hence, it requires the sink node to make data delivery within a specific time interval. In this case, the ES acts as a proactive inquirer that makes data delivery requests irregularly.

III. Mathematical Modeling

In this section, we provide mathematical modeling for the packet combination and Age of Information (AoI).

A. Packet Combination

In our network model, as shown in Figure 1, we assume the sink node has the capability of packing multiple packets into one by discarding redundant packet headers while keeping useful payload information. This packing behavior can significantly reduce the time slots used for transmitting packets to the ES. To mathematically model such behavior, we denote λ as the re-packing rate at the sink node. Hence, if there are a total of L packets, L time slots are needed for transmission in the non-combination mode, but only $\lceil \lambda L \rceil$ time slots are needed in the combination mode, where $\lceil \cdot \rceil$ is the ceiling function and $1/L < \lambda < 1$. The certain value of λ relies on both the sizes of packet headers and payloads. In this paper, we only leverage its mathematical characteristic in analyzing data freshness and discuss its influence on Instant AoI in our simulations.

B. AoI Formulation

To capture the latest data updates, we adopt the recently proposed *Age of Information* (AoI) metric, which gauges data freshness from sensing nodes. At the ES, the latest updates from sensing nodes are included in the arrival re-packed packet, which is delivered from the sink node. Let $G_n^E(t)$ and

$A_n^E(t)$ denote the generation time of the latest packet from a sensing node n and its AoI at the ES at time t, respectively. Based on the definition of AoI, we have:

$$A_n^E(t) = t - G_n^E(t) \ . \tag{1}$$

It is seen that AoI increases linearly with time and decreases when a new packet is delivered. Similarly, at the sink node S, denoting $G_n^S(t)$ and $A_n^S(t)$ as the generation time of the latest packet directly transmitted from a sensing node n and its corresponding AoI at time t, respectively, we have:

$$A_n^S(t) = t - G_n^S(t) \ . \tag{2}$$

C. AoI Variation

To better understand the variation of AoI, we first derive the relationships of AoIs at the ES and the sink node. At the h-th delivery to the ES, we assume that there are a total of $L(h)$ packets at the sink node S. Then, it will take $\lceil \lambda L(h) \rceil$ time slots to transmit the combined packet. Assuming that at the sink node S, a packet starts its transmission at time slot $b_s(h)$ and ends at time slot $e_s(h)$, we have $e_s(h) = b_s(h) + \lceil \lambda L(h) \rceil - 1$. At the ES, the generation time of the latest arrival packet for a sensing node n is the same as its generation time at the sink node before being transmitted. Hence, we have:

$$G_n^E\big(b_s(h) + \lceil \lambda L(h) \rceil - 1\big) = G_n^S\big(b_s(h)\big) \ . \tag{3}$$

Putting (1), (2), and (3) together, we have the relationship of AoIs at the ES and at a sink node S for given sensing node n as follows:

$$\begin{aligned} A_n^E\big(e_s(h)\big) =& e_s(h) - G_n^E\big(e_s(h)\big) \\ =& b_s(h) + \lceil \lambda L(h) \rceil - 1 - G_n^S\big(b_s(h)\big) \\ =& A_n^S\big(b_s(h)\big) + \lceil \lambda L(h) \rceil - 1 \ . \end{aligned} \tag{4}$$

Equation (4) allows us to calculate the AoI at the ES based on the observation of AoI at the sink node S.

Next, we analyze the AoI variation at the sink node. Recall that when scheduled for packet transmission in a time slot, a sensing node will generate a new packet and finish transmitting it within this time slot. Denote $b_n(h)$ as the time slot for a sensing node n to make its latest transmission in delivering a packet to the sink node in the h-th transmission cycle. Since each transmitted packet can be fitted into one time slot, after being generated and uploaded at time slot $b_n(h)$, we have $G_n^S\big(b_n(h)\big) = b_n(h)$. Then, AoI at the sink node in time slot $b_n(h)$ can be calculated as: $A_n^S\big(b_n(h)\big) = b_n(h) - G_n^S(b_n(h)) = 0$. Since the sink node receives a packet from sensing node n in time slot $b_n(h)$ and transmits it at time slot $b_s(h)$, the AoI increases linearly between these two time slots to yield:

$$\begin{aligned} A_n^S\big(b_s(h)\big) &= A_n^S\big(b_n(h)\big) + \big(b_s(h) - b_n(h)\big) \\ &= b_s(h) - b_n(h) \ . \end{aligned} \tag{5}$$

Finally, with (4) and (5), AoI at the ES after receiving a (combined) packet from the sink node can be rewritten as $A_n^E\big(e_s(h)\big) = b_s(h) - b_n(h) + \lceil \lambda L(h) \rceil - 1$. On the other hand, if no new packet arrives, AoI at the ES will increase by 1 in each time slot, giving rise to:

$$A_n^E(t) = \begin{cases} b_s(h) - b_n(h) + \lceil \lambda L(h) \rceil - 1 \ , & \text{if } t = e_s(h); \\ A_n^E(t-1) + 1 \ , & \text{otherwise.} \end{cases} \tag{6}$$

D. Instant AoI

With the AoI variation formula, we explore the AoI metric over an infinite time span. To signify actual freshness from the application's point of view, we propose a more appropriate metric, i.e., *Instant AoI*, reflecting ES's AoI at the time immediately after a packet being delivered from the sink node.

Definition. Let $\bar{A}_n^E(h)$ denote the Instant AoI corresponding to a sensing node n at the h-th transmission, then we have: $\bar{A}_n^E(h) = A_n^E\big(e_s(h)\big)$, where $e_s(h)$ is the time slot for the sink node to finish transmitting its h-th combined packet. Thus, the long-term averaged Instant AoI at the ES with respect to a sensing node n can be expressed as follows:

$$\bar{A}_n^E = \lim_{H \to \infty} \frac{1}{H} \sum_{h=1}^{H} A_n^E(h) \ , \tag{7}$$

where H indicates the total number of transmissions from the sink node S to the ES. Among all sensing nodes, the long-term averaged Instant AoI is expressed by:

$$\bar{A}^E = \frac{1}{|\mathcal{N}|} \sum_{n \in \mathcal{N}} \bar{A}_n^E \ . \tag{8}$$

Instant AoI has real-world importance since it takes into account the practical meaning for applications in IoT networks. Because shelving an arrival update degrades its freshness, applications often prefer to get fresh updates as soon as they are received. This is particularly true for real-time applications to avoid AoI deterioration over time after update delivery.

Based on the formula for Instant AoI, minimizing the long-term Instant AoI $\bar{A}^E$ involves parameters of $b_s(h)$, $b_n(h)$, and $L(h)$, relating to scheduling both sensing nodes and the sink node. The next two sections address the scheduling problem by taking into account two categories of sink node delivery behaviors, i.e., *Periodic Request* and *Proactive Request*.

IV. Periodic Request

In this section, we focus on the Instant AoI minimization problem under the scenario that the sink node periodically delivers combined packets to the ES. Such a delivery mode is practical, thus popular in many real applications. This mode is intuitive but its results are important, serving as the performance yardstick to assess the scheduling algorithms for more sophisticated delivery modes.

A. AoI Formula

In this mode, the sink node processes a delivery packet once every T time slots, called one transmission cycle. In each cycle (i.e., one delivery to the ES), the sink node combines the latest updates maintained for all sensing nodes before delivering to the ES. The combined packet will take $\lceil \lambda |\mathcal{N}| \rceil$ time slots for delivery in this cycle, where $|\mathcal{N}|$ is the total number of sensing nodes, each assumed to have one packet. We now analyze the Instant AoI variation with respect to all sensing nodes between two consecutive transmission cycles for a better understanding of the Instant AoI.

From (6), we see that the AoI at the ES with respect to a sensing node n increases linearly by the step of 1, until the sink node completes one delivery of the combined packet which contains updates from node n. We introduce a metric called the **Instant AoI profit**, denoted as $\Delta_n(h)$, to measure the decrease of Instant AoI when a sensing node n uploads a new packet to the sink node in the h-th cycle, expressed as:

$$\Delta_n(h) \triangleq \big(\bar{A}_n^E(h-1)+T\big)-\big(b_s(h)-b_n(h)+\lceil\lambda|\mathcal{N}|\rceil-1\big). \quad (9)$$

Let $h^{'}$ denote the most recent transmission cycle ($h^{'} < h$), in which sensing node n sends a packet to the sink node. The AoI at the ES with respect to sensing node n increases linearly from $h^{'}$-th to h-th transmission cycles, i.e.,

$$\bar{A}_n^E(h-1) + T = \bar{A}_n^E(h^{'}) + (h-h^{'}) \times T \ . \quad (10)$$

According to (6), (9), and (10), $\Delta_n(h)$ can be expressed as:

$$\begin{aligned}\Delta_n(h) =&(\bar{A}_n^E(h-1)+T) - (b_s(h) - b_n(h) + \lceil\lambda|\mathcal{N}|\rceil - 1)\\ =&\big(\bar{A}_n^E(h^{'})+(h-h^{'})\times T\big)-\big(b_s(h)-b_n(h)+\lceil\lambda|\mathcal{N}|\rceil-1\big)\\ =&\big(b_s(h^{'}) - b_n(h^{'}) + \lceil\lambda|\mathcal{N}|\rceil - 1 + (h-h^{'}) \times T\big)-\\ &\big(b_s(h) - b_n(h) + \lceil\lambda|\mathcal{N}|\rceil - 1\big)\\ =&b_n(h) - b_n(h^{'}) + \big(b_s(h^{'}) - b_s(h) + (h-h^{'}) \times T\big)\\ =&b_n(h) - b_n(h^{'}) \ , \end{aligned} \quad (11)$$

where $\bar{A}_n^E(h^{'}) = A_n^E(e_s(h^{'}))$ and $b_s(h) - b_s(h^{'}) = (h-h^{'}) \times T$. We observe that this profit is actually equal to the time interval between the current transmission time slot $b_n(h)$ and the most recent one $b_n(h^{'})$, for sensing node n.

B. Optimal Instant AoI Scheduling

Since all sensing nodes share the same channel, the interference among them exists, causing only one of them to transmit in each time slot. Besides, in each transmission cycle of T time slots, $\lceil\lambda|\mathcal{N}|\rceil$ time slots will be occupied by the sink node S's transmission. Notably, time slots used by all sensing nodes and the sink node cannot overlap so as to avoid interference. We aim to develop the optimal scheduling for the Instant AoI in the periodic request mode with all nodes transmitting free of interference. When setting the transmission cycle, we should always let $T > \lceil\lambda|\mathcal{N}|\rceil$; otherwise, the time slots in each cycle are too tight for the sink node to transmit combined packets. Let $\dot{T} \triangleq T - \lceil\lambda|\mathcal{N}|\rceil$ represent the number of time slots available for scheduling sensing nodes' transmissions. Indexing all sensing nodes with $(1, 2, \cdots, |\mathcal{N}|)$, then the scheduling policy in each transmission cycle is denoted by a vector: $P = (p_1, \cdots, p_{\dot{T}})$, in which each element represents a number from 1 to $|\mathcal{N}|$. If $p_j = m$, it means the node with index m is scheduled to upload its packet at the j-th time slot in the current transmission cycle to the sink node.

We consider the general scenario where the time slots in each transmission cycle are not necessarily enough to schedule all sensing nodes' transmission. The metric, Instant AoI profit, will be used here to derive the long-term averaged Instant AoI. Let $\mathcal{V}(h)$ denote the set of sensing nodes that upload new packets to the sink node S in the h-th transmission cycle. For each node in set $\mathcal{V}(h)$, its Instant AoI will be reduced by $\Delta_n(h)$. By adding T to the Instant AoI of all sensing nodes, and subtracting the AoI profit (i.e., $\Delta_n(h)$) from the Instant AoI of those nodes in $\mathcal{V}(h)$, we have Instant AoI at the current cycle h, based on that in the $(h-1)$-th cycle, as follows:

$$\bar{A}^E(h) = \bar{A}^E(h-1) + \frac{1}{|\mathcal{N}|}\big(|\mathcal{N}| \cdot T - \sum_{n\in\mathcal{V}(h)} \Delta_n(h)\big) \ . \quad (12)$$

According to (12), we can recursively calculate the long-term averaged Instant AoI of all nodes at the H-th cycle, denoted as $\bar{A}^E(H)$, starting from the initial state $\bar{A}^E(0)$. By defining $\mathbb{I}_{(\cdot)} = 1$ if $(\cdot)$ is satisfied; $= 0$, otherwise, we have:

$$\begin{aligned}\bar{A}^E(H) =&\bar{A}^E(0) + \frac{1}{|\mathcal{N}|}\Big(\sum_{h=1}^{H}\big(|\mathcal{N}| \cdot T - \sum_{n\in\mathcal{V}(h)} \Delta_n(h)\big)\Big)\\ =&\bar{A}^E(0) + HT - \frac{1}{|\mathcal{N}|}\sum_{n\in\mathcal{N}}\sum_{h=1}^{H}\big(\mathbb{I}_{(n\in\mathcal{V}(h))}\Delta_n(h)\big).\end{aligned}$$

The last term in the formula is calculated by adding the Instant AoI profit cumulatively in each transmission cycle for all sensing nodes scheduled in the h-th cycle. Recall that Instant AoI profit of a sensing node equals the time slot interval between two consecutive transmissions. If the last time slot for a node n to transmit an update till its H-th transmission cycle is $\tilde{b}_n(H)$, we have:

$$\sum_{h=1}^{H}\Big(\mathbb{I}_{(n\in\mathcal{V}(h))}\Delta_n(h)\Big) = \tilde{b}_n(H) \ . \quad (13)$$

This value for a node n equals the total number of time slots from its start to its last transmission slot. In other words, $\tilde{b}_n(H)$ is the index for the latest time slot, when node n delivers a new update. Then, the long-term averaged Instant AoI of all sensing nodes can be calculated as follows:

$$\begin{aligned}\bar{A}^E =& \lim_{H\to\infty}\frac{1}{H}\sum_{h=1}^{H}\bar{A}^E(h)\\ =&\lim_{H\to\infty}\frac{1}{H}\Big\{H\bar{A}^E(0) + \Big(T + \cdots + HT\Big)\\ &- \frac{1}{|\mathcal{N}|}\sum_{h=1}^{H}\sum_{n\in\mathcal{N}}\Big(\sum_{h=1}^{H}\Big(\mathbb{I}_{n\in\mathcal{V}(h)}\Delta_n(h)\Big)\Big)\Big\}\\ =&\bar{A}^E(0) + \frac{H+1}{2}T - \lim_{H\to\infty}\frac{1}{H}\frac{1}{|\mathcal{N}|}\sum_{h=1}^{H}\sum_{n\in\mathcal{N}}\tilde{b}_n(h).\end{aligned}$$

Hence, minimizing long-term averaged Instant AoI over H transmission cycles can be equally represented by:

$$\text{OPT-1} \quad \max \sum_{h=1}^{H} \sum_{n \in \mathcal{N}} \tilde{b}_n(h)$$
$$s.t. \text{ Constraints: } \tilde{b}_n(h) \leq hT \quad .$$

Based on our analysis, we have transformed the Instant AoI minimization problem to OPT-1, which focuses only on maximizing the sum of indices of time slots for sensing nodes updating packets among all transmission cycles. To solve this problem, we propose an algorithm to schedule transmissions of all sensing nodes over all transmission cycles, aiming to achieve the long-term averaged optimal Instant AoI. The main idea of this algorithm is to maximize the sum of the last transmission time slot indices of all sensing nodes in each transmission cycle. In each cycle, we non-repeatedly select nodes with the minimum index in the last transmission time slots and assign an available time slot in the current transmission cycle to it in turn, so as to increase its corresponding index. The details of our algorithm are listed in Algorithm 1.

Algorithm 1 Min Instant AoI Periodic Request

```
In each transmission cycle, repeat the following:
Initialization: P = ∅, I = ∅ and j = Ṫ.
while 1 ≤ j ≤ Ṫ do
    Choose node n ∈ N and n ∉ I with the minimum
    value of b̃_n.
    Schedule node n in the j-th time slot, p_j = n.
    j = j − 1.
    Add node n to set I, i.e., I = I ∪ n.
end while
```

It can be easily proved that the Algorithm 1 can provide the optimal scheduling for OPT-1. Notably, the optimal scheduling derived from Algorithm 1 is the Round Robbin policy. This result is expected because of the regularity that comes from the periodic request. Such a scheduler possesses the advantages of low control overhead and easy implementation.

V. Proactive Request

In this section, we study the Instant AoI minimization problem under the scenario that the upper layer applications send proactive requests to the ES for acquiring the latest updates from the sink node once every T time slots. The sink node shall always make a fast decision in order for the latest packet punctually delivered to the ES within the coming T time slots.

In this proactive request mode, the sink node can perform packet combining more flexibly. One combined packet may consist of updates received from only a portion of sensing nodes. As a result, time slots taken by a sink node's transmission depend on the number of packets that are uploaded and combined. After receiving all packets from scheduled sensing nodes, the sink node delivers a combined packet to the ES. Here, scheduling refers to the decision for a transmission cycle, i.e., T time slots, in which the sink node determines whether a sensing node is scheduled to upload its new update and how many packets are to be combined before delivery to the ES in order to minimize Instant AoI at the ES.

A. AoI Model

We now analyze the AoI model in the proactive request mode. Such a request from the ES is hard to be predicted since it depends on the upper layer behaviors in practice. Instead of taking into account the long-term AoI and predicting the distribution of requests, we take the Instant AoI after each request as the main metric since we aim to ensure information freshness from the user perspective upon receiving the requested packets. If there is no request, the scheduling of sensing nodes to send their updates is not considered here. Thus, to model the Instant AoI minimization problem, we define the state when a request arrives as follows:

- The complete information of current AoIs at the ES before an update request, with respect to all sensing nodes, $A_n^E(0)$.
- The duration time T of a request.

Denote b_s as the time slot when the sink node starts to transmit a combined packet and L as the total number of new packets uploaded from sensing nodes. Since the transmission from the sink node to the ES shall finish within T time slots, we have the following constraint:

$$b_s + \lceil \lambda L \rceil - 1 \leq T \ . \tag{14}$$

In addition, it takes at least L time slots for scheduled sensing nodes to deliver their update packets, so the sink node's uploading shall wait for finishing update delivery, leading to

$$b_s > L \ . \tag{15}$$

Before the sink node starts transmitting, there are $(b_s - 1)$ time slots left for sensing nodes to upload new updates. Denote A_n^E as the Instant AoI at the ES with respect to node n after delivering a combined packet that represents L updates. According to (6), we have:

$$A_n^E = \begin{cases} b_s - b_n + \lceil \lambda L \rceil - 1, & \text{if node n delivers a new update to the sink node;} \\ A_n^E(0) + b_s + \lceil \lambda L \rceil - 1 \ , & \text{otherwise,} \end{cases} \tag{16}$$

where b_n is the index of the time slot scheduled for node n to upload its update. Similarly, we apply the Instant AoI profit, denoted as Δ_n, which is the decrement in Instant AoI when the sink node uploads a combined packet including the update from a sensing node n. Then, we have:

$$\begin{aligned} \Delta_n &= (A_n^E(0) + b_s + \lceil \lambda L \rceil - 1) - (b_s - b_n + \lceil \lambda L \rceil - 1) \\ &= A_n^E(0) + b_n \ . \end{aligned} \tag{17}$$

Hence, the averaged Instant AoIs of all sensing nodes, from which whatever or not send new packets, denoted as A^E, can be calculated as follows:

$$\begin{aligned}A^E &= \frac{1}{|\mathcal{N}|}\Big(\sum_{n\in\mathcal{N}}\big(A_n^E(0)+b_s+\lceil\lambda L\rceil-1\big)-\sum_{n\in\mathcal{V}}\Delta_n\Big)\\ &= \frac{1}{|\mathcal{N}|}\Big(\sum_{n\in\mathcal{N}}\big(A_n^E(0)+b_s+\lceil\lambda L\rceil-1\big)\\ &\quad -\sum_{n\in\mathcal{V}}\big(A_n^E(0)+b_n\big)\Big)\\ &= \frac{1}{|\mathcal{N}|}\big(\sum_{n\in\complement_{\mathcal{N}}\mathcal{V}}A_n^E(0)-\sum_{n\in\mathcal{V}}b_n+|\mathcal{N}|\cdot(b_s+\lceil\lambda L\rceil-1)\big),\end{aligned}$$

where $\mathcal{V}$ is the set of nodes that send new updates included in the combined packet and $|\mathcal{V}| = L$. $\complement$ represents the complement set. As a result, the Instant AoI minimization problem is formulated as follows:

$$\begin{aligned}\text{OPT-2}\quad &\min\ A^E\\ &s.t.\ \text{Constraints: (14) and (15)}\ .\end{aligned}$$

B. Optimal Instant AoI Scheduling

Since two key parameters, L and b_s, are involved in OPT-2, we will examine each term of the objective function in order to determine whether they can be optimized independently. We first examine the term of $\sum_{n\in\complement_{\mathcal{N}}\mathcal{V}}A_n^E(0)$. For a given value of L, which equals the size of set $\mathcal{V}$, the minimum value can be obtained by choosing $|\complement_{\mathcal{N}}\mathcal{V}|$ nodes with their corresponding $A_n^E(0)$ values from the smallest to the biggest. As a result, the optimal value of this first term relies only on parameter L. We use $f(L)$ to represent the minimum value of $\sum_{n\in\complement_{\mathcal{N}}\mathcal{V}}A_n^E(0)$ for a specific value of L.

We next analyze the term of $\sum_{n\in\mathcal{V}}b_n$. Since only one packet can be transmitted from a sensing node to the sink node in each time slot, the values of b_n for nodes chosen in the total of (b_s-1) available time slots should be the non-repeating natural numbers in the range of $(1,2\cdots,b_s-1)$. Given L, the maximum value of $\sum_{n\in\mathcal{V}}b_n$ results from adding the last L values in the natural number sequence of $(1,2,\cdots,b_s-1)$. Hence, the term of $\sum_{n\in\mathcal{V}}b_n$ can be expressed by $\sum_{n\in\mathcal{V}}b_n = (b_s-L)+\cdots+(b_s-1)=\frac{(2b_s-L-1)}{2}L$. Then, the objective function can be represented as:

$$A^E = \frac{1}{|\mathcal{N}|}\big(f(L)-\frac{(2b_s-L-1)}{2}L+|\mathcal{N}|(b_s+\lceil\lambda L\rceil-1)\big).$$

Since the objective function is re-written into expressions involving two parameters b_s and L, we have the derivation of b_s as follows: $\frac{\partial A^E}{\partial b_s}=1-\frac{1}{|\mathcal{N}|}\geq 0$, which indicates the objective function increases independently with respect to b_s. Since $b_s > L$ and b_s is the natural number that represents the time slot index, we take the minimum value of $b_s = L+1$, to minimize A^E. Then, the objective function relies solely on the parameter L and can be expressed by:

$$A^E = \frac{1}{|\mathcal{N}|}\big(f(L)-\frac{(L+1)}{2}L+|\mathcal{N}|(L+\lceil\lambda L\rceil)\big)\ .$$

This new objective function is important, helping us design an algorithm governed only by L in optimizing the Instant AoI.

Then, we propose the following algorithm to solve OPT-2, as shown in Algorithm 2, where we define the scheduling policy as a vector: $P = (p_1,\cdots,p_{(b_s-1)})$, in which each element is a number from 1 to $|\mathcal{N}|$. If $p_j = m$, the node with index m is scheduled to upload its packet in the j-th time slot. The main idea is to find the optimal value of L, which represents the number of nodes that can be scheduled to transmit new updates. We iteratively calculate the optimal Instant AoI under different L values. Since L is countable, after all iterations, we can identify the optimal L minimizing the averaged Instant AoI. Next, we select L sensing nodes having the minimum AoI values, and assign one time slot to each of them in turn for transmitting a new packet. After all of them finish transmissions, the sink node combines received packets and then starts delivering it in the next time slot.

Algorithm 2 Min Instant AoI Proactive Delivery

Initialization: $L = 1$, $l = 0$, $c = MAX$, $c_1 = 0$, $c_2 = 0$, $c_3 = 0$, $P = \varnothing$, and $S = \varnothing$.
Sort nodes based on the value of $A_n^E(0)$ in ascending order, recorded in the set S.
while $1 \leq L \leq T$ **do**
  if $L + \lceil\lambda L\rceil \leq T$ **then**
    Calculate the sum of $A_n^E(0)$ except for the last L elements in sequence S, recorded as c_1.
    Calculate the value of $(L+1)L/2$, recored as c_2.
    Calculate $|\mathcal{N}|\cdot(L+\lceil\lambda L\rceil)$, recorded as c_3.
    if $c > c_1 - c_2 + c_3$ **then**
      $c = c_1 - c_2 + c_3$ and $l = L$.
    end if
  end if
end while
Choose the last l nodes in S to transmit packets in turn, $P = (S[|\mathcal{N}|-l+1],\cdots,S[|\mathcal{N}|])$.
Sink node starts transmitting the combined packet at time slot $(l+1)$.

With Algorithm 2, given any request from the ES, the sink nodes can make a fast decision on scheduling sensing nodes to transmit their latest updates in a way that minimizes long-term averaged Instant AoI.

VI. Numerical Results

In this section, we evaluate the performance of our proposed solutions for packet combination under both periodic and proactive request modes.

A. Periodic Request Mode

In the periodic request mode, we consider IoT networks with the number of sensing nodes equal to 20, 30, and 50, respectively. The amounts of time slots in each transmission cycle vary from 20 to 80 with the step of 10, while the re-packing rate varies from 0.2 to 0.6 with the step of 0.1. For comparison, we consider the non-combination scheduling

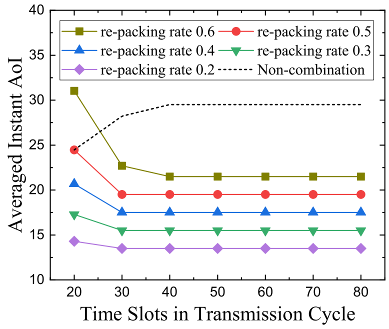

Fig. 2. Averaged Instant AoI versus the time slot count in a transmission cycle for our packet combination scheme under different re-packing rates and 20 sensing nodes.

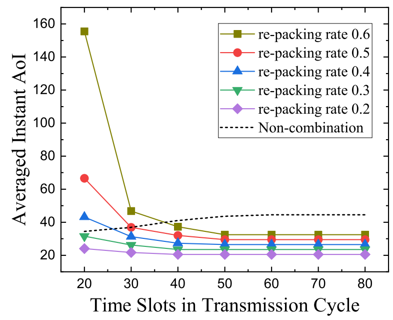

Fig. 3. Averaged Instant AoI versus the time slot count in a transmission cycle for our packet combination scheme under different re-packing rates and 30 sensing nodes.

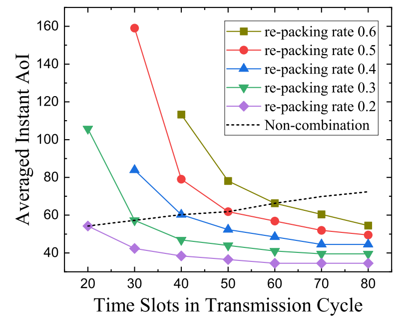

Fig. 4. Averaged Instant AoI versus the time slot count in a transmission cycle for our packet combination scheme under different re-packing rates and 50 sensing nodes.

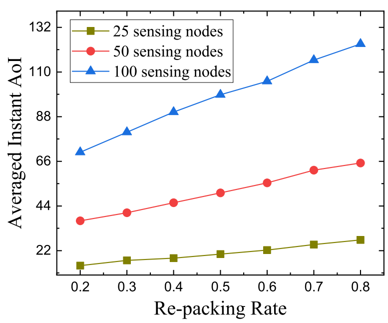

Fig. 5. Instant AoI comparison under different sensing node counts and re-packing rates.

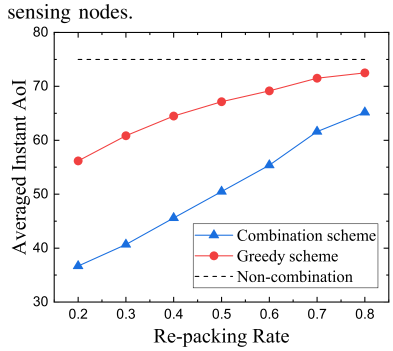

Fig. 6. Our combination scheme compared to both non-combination and greedy combination schemes under 50 sensing nodes.

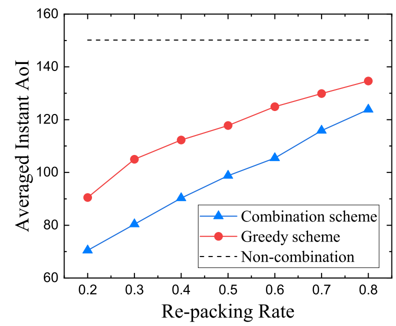

Fig. 7. Our combination scheme compared to both non-combination and greedy combination schemes under 100 sensing nodes.

mode where sensing nodes upload their packets in turn to the sink node in the first half of each transmission cycle, while in the second half of this cycle, the sink node forwards all received packets to the ES without combining. The order for sensing nodes to upload is based on their current Instant AoI values obtained in the previous cycle. The one with the largest Instant AoI is scheduled to transmit with the highest priority.

Figs. 2, 3, and 4 show the optimal long-term averaged Instant AoIs at the ES under the IoT networks with sensing node counts equal to 20, 30, and 50, respectively, versus the time slot amounts in each transmission cycle and re-packing rates. The averaged Instant AoI for non-combination mode is also included, as indicated by a dotted line in each figure. From these figures, we can see the long-term averaged Instant AoIs first decrease with an increase in the time slot amounts in each transmission cycle and then remain unchanged under the packet combination mode. The reason is that more time slots accommodate more latest updates from sensing nodes for simultaneous delivery to the ES, thus reducing the Instant AoI. However, as the number of time slots in each transmission cycle is large enough (known as the threshold) for all sensing nodes to transmit their updates, Instant AoI reaches a low constant and remains unchanged afterwards. In Fig. 2 (or Fig. 3), for example, the threshold of the time slot amount appears to be 40 (or 50), beyond which the averaged Instant AoI is constant.

In the non-combination mode, however, the averaged Instant AoIs become worse with an increase in the time slot amount until its plateau. The reason is that the sink node cannot start transmitting its received packets to the ES until all scheduled sensing nodes finish sending their update packets, so packets at the sink nodes have to wait before delivery in sequence. If not all sensing nodes can be scheduled, the waiting time equals the scheduled sensing node count, which is always one half of the total time slot amount. Hence, Instant AoI deteriorates with an increase in the number of time slots a transmission cycle contains. When the number is large enough for all sensing nodes to send their updates, the Instant AoI will not change any more afterwards. The dotted lines in Figs. 2, 3, and 4 signify such trends.

The results of Figs. 2, 3, and 4 also exhibit that long-term averaged Instant AoIs decrease as the re-packing rate drops. This is expected because a lower re-packing rate causes more packets to be combined into one, taking fewer time slots for transmission. Accordingly, more updates from sensing nodes can be delivered to the ES simultaneously, resulting in lower Instant AoI values.

Meanwhile, we also observe that the combining scheme does not always outperform its non-combination counterpart, depending on the re-packing rate. For example, in Figure 4, when the number of time slots is less than 50 for the re-packing rate of 0.5, the averaged Instant AoIs across 50 sensing nodes under the packet combination scheme are larger than that under its non-combination counterpart. The reason is that the sink node always combines the latest updates of all sensing nodes into one packet. The sink node keeps waiting for updates from all scheduled sensing nodes in the current transmission cycle, thereby taking more time slots and raising the AoIs of other updated packets. Moreover, old packets may be packed into the combined one for delivery if no new updated packet is received, yielding much larger AoI values. As a result, the Instant AoI is worse under the combination scheme than under the non-combination scheme. This result calls for a proper choice of the re-packing rate in order to improve Instant AoI in the packet combining scheme.

B. Proactive Request Mode

In the proactive request mode, we conduct experiments in the IoT networks respectively with 25, 50, and 100 sensing nodes. We let the re-packing rates vary from 0.2 to 0.8 with the increment of 0.1. To simulate the unpredictable user request behavior, we assume request arrival follows the Poisson process. The current AoI values corresponding to all sensing nodes when a request arrives are assigned randomly from 1 to N, for $N = 25$, 50, and 100, respectively. In our evaluation, every simulation setting runs 50 times, with each for a duration of 2000 time slots. The duration of requests in every 100 time slots is simulated to follow an exponential distribution with the mean of $\frac{1}{\lambda} = 0.25, 0.5$, and 1 respectively, for the IoT networks with 25, 50, and 100 sensing nodes.

The outcomes of non-combination and greedy combination schemes are obtained as well for comparing with those of our proposed combining scheme under its associated algorithms. The non-combination scheme is the same as what is described in Section VI-A. For the greedy scheme, we adopt the packet combination in a way that maximizes the number of sensing nodes able to transmit new update packets upon each request.

Fig. 5 demonstrates the optimal averaged Instant AoIs at the ES under the IoT networks with sensing node amounts of 25, 50, and 100, respectively, when varying re-packing rates. From this figure, we observe that the averaged Instant AoI increases as the re-packing rate rises. The reason is that a combined packet takes more time slots to finish its transmission if the re-packing rate is larger, requiring more time slots to complete one request and thus hurting Instant AoI. It is also observed that when the number of sensing nodes grows, the averaged Instant AoI rises as expected since then more sensing nodes cannot be scheduled to transmit their packets in each request.

Figs. 6 and 7 show the comparative averaged Instant AoI results of our proposed combination scheme and its two counterparts, with 50 and 100 sensing nodes, respectively, in the IoT networks. From the figures, we find that our proposed optimal combination scheme under the proactive request mode always outperforms both non-combination and greedy combination counterparts. In the non-combination scheme, new packets uploaded from sensing nodes take more time slots for delivery to the ES when compared with our combination scheme, for the same amount of new packets. These new packets stagnate at the sink node for a longer time before delivered to the ES, thus resulting in worse Instant AoI performance. For greedy combination transmission, the sensing nodes try their best to upload new packets in the hope of lowering their own Instant AoI values, but for sensing nodes with no updated packet, their AoI values keep increasing; the longer time for a request to finish, the larger increase in their Instant AoI value. In contrast, our proposed optimal combination scheme takes into consideration all sensing nodes both with and without update packets via its optimal scheduling to determine the number of packets to be uploaded, resulting in the minimum Instant AoI for each request.

VII. Related Work

Extensive research has explored the theoretical foundation of AoI, [1], [2], [3], [4], [5], [6], [7], [8], [9], [10]. Specifically, in [1], AoI was first proposed to optimize packet generation rate at a single source node under the *First Come First Served* (FCFS) scheduling while in both [1] and [2], it was shown that AoI differs significantly from delay in regard to measuring data freshness. Besides, various packet queuing management schemes have been pursued in [4], [5], and [6], highlighting that discarding the old packets helps improve AoI performance. Meanwhile, the scheduling policy, *Last Generated First Served* (LGFS) with/without preemption, has been considered in [7], [8], [21], under different network setting (i.e., single-hop, multi-server single-hop, multi-hop, respectively) to analyze the averaged AoI. The LGFS policy combined with replication has been considered recently in [9] for a multi-server system, with an extended Max-Age-First LGFS policy provided later in [10].

The AoI under different network constraints has also been explored. In [11], AoI under the throughput constraint has been considered, whereas [12] and [22] have dealt with AoI under general interference in the wireless sensor networks and multi-hop networks, respectively. All these efforts offer the theoretical foundation of AoI, without taking into account the practical meanings or settings. In the practical network scenarios, AoI was first studied in the broadcast network [23], [24]. Specifically, [23] explores the long-run averaged AoI in the setting where a base station (BS) first receives packets from multiple sources and then broadcasts each packet to its destined user. In [24], the wireless broadcast network with unreliable channels is analyzed for the BS to distribute information to multiple clients. Then, AoI is pursed in the gossip network [25] and a two-hop energy-harvesting network [26]. The energy problem is also explored in [27] and [28], to study its impact on AoI optimization, considering the energy replenishment constraints and the scheduling to optimize AoI under energy requirements.

The influence of practical channel access modulation on AoI has been pursued. Specifically, [13] has considered the scheduled access with slotted ALOHA to schedule the packet transmission, whereas [13] and [14] have adopted the token turns channel access scheduling to schedule multiple terminals for communicating with one BS while ensuring AoI performance. In [15], the minimal Age scheduling with the TDMA modulation in a general single-hop network has been studied.

Recently, exploring AoI in IoT networks or the network edge has also received much attention. In particular, the optimal status sampling and updating process are developed in [18] to minimize the averaged AoI under an average energy cost constraint at the IoT device. A general model for the sampling behavior at the network edge is studied in [19] minimizing AoI, deemed more relevant to real-world applications. Moreover, AoI performance in some special network scenarios is also considered. For example, [16] studies the AoI optimization problem between a central station and a set of ground terminals via a mobile agent that can travel across the ground terminals. In addition, [17] considers the request and response behavior under a newly defined AoI metric that accounts for the data freshness of only users' requests.

VIII. Conclusion

This paper conducts an in-depth study on data freshness in the applications of IoT networks. The packet combination behaviors have been explored to accelerate the delivery of useful updates from sensing nodes to the IoT edge server, for prompt acts accordingly. We have addressed two data acquisition modes widely used by applications and advocated the Instant AoI for better measuring data freshness in IoT network applications. Rigorous mathematical models are developed to characterize the Instant AoI under each mode. By analyzing the Instant AoI variation at the sink node, we have designed the low-complexity algorithms to deal with scheduling sensing nodes and the sink node. The theoretical proofs have shown that our proposed algorithms can achieve the optimal long-term averaged Instant AoI. The simulation results demonstrate the advantages of our proposed packet combination scheme when compared to its non-combining and greedy combining counterparts in terms of AoI improvement. Our mathematical development, algorithmic solutions, and results presented in this paper shed light on AoI studies by taking into account the important characteristics of real-world IoT networks.

References

[1] S. Kaul, R. Yates, and M. Gruteser, "Real-time status: How often should one update?" in *Proceedings of IEEE International Conference on Computer Communications (INFOCOM)*, 2012, pp. 2731–2735.

[2] R. D. Yates and S. Kaul, "Real-time status updating: Multiple sources," in *Proceedings of IEEE International Symposium on Information Theory (ISIT)*, 2012, pp. 2666–2670.

[3] Y. Sun, E. Uysal-Biyikoglu, R. Yates, C. E. Koksal, and N. B. Shroff, "Update or wait: How to keep your data fresh," in *Proceedings of IEEE International Conference on Computer Communications (INFOCOM)*, 2016, pp. 1–9.

[4] M. Costa, M. Codreanu, and A. Ephremides, "Age of information with packet management," in *Proceedings of IEEE International Symposium on Information Theory (ISIT)*, 2014, pp. 1583–1587.

[5] ——, "On the age of information in status update systems with packet management," *IEEE Transactions on Information Theory*, vol. 62, no. 4, pp. 1897–1910, 2016.

[6] N. Pappas, J. Gunnarsson, L. Kratz, M. Kountouris, and V. Angelakis, "Age of information of multiple sources with queue management," in *Proceedings of IEEE International Conference on Communications (ICC)*, 2015, pp. 5935–5940.

[7] S. K. Kaul, R. D. Yates, and M. Gruteser, "Status updates through queues," in *Proceedings of Annual Conference on Information Sciences and Systems (CISS)*, 2012, pp. 1–6.

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

[9] ——, "Minimizing the age of the information through queues," 2017. [Online]. Available: https://arxiv.org/abs/1709.04956

[10] Y. Sun, E. Uysal-Biyikoglu, and S. Kompella, "Age-optimal updates of multiple information flows," 2018. [Online]. Available: https://arxiv.org/abs/1801.02394

[11] I. Kadota, A. Sinha, and E. Modiano, "Optimizing age of information in wireless networks with throughput constraints," in *Proceedings of IEEE International Conference on Computer Communications (INFOCOM)*, 2018, pp. 1844–1852.

[12] R. Talak, S. Karaman, and E. Modiano, "Optimizing information freshness in wireless networks under general interference constraints," in *Proceedings of ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, 2018, pp. 61–70.

[13] R. D. Yates and S. K. Kaul, "Status updates over unreliable multiaccess channels," in *Proceedings of IEEE International Symposium on Information Theory (ISIT)*, 2017, pp. 331–335.

[14] Z. Jiang, B. Krishnamachari, X. Zheng, S. Zhou, and Z. Niu, "Decentralized status update for age-of-information optimization in wireless multiaccess channels," in *Proceedings of IEEE International Symposium on Information Theory (ISIT)*, 2018, pp. 2276–2280.

[15] T.-W. Kuo, "Minimum age tdma scheduling," in *Proceedings of IEEE International Conference on Computer Communications (INFOCOM)*, 2019.

[16] V. Tripathi, R. Talak, and E. Modiano, "Age optimal information gathering and dissemination on graphs," in *Proceedings of IEEE International Conference on Computer Communications (INFOCOM)*, 2019.

[17] B. Yin, S. Zhang, Y. Cheng, L. X. Cai, Z. Jiang, S. Zhou, and Z. Niu, "Only those requested count: Proactive scheduling policies for minimizing effective age-of-information," in *Proceedings of IEEE International Conference on Computer Communications (INFOCOM)*, 2019.

[18] B. Zhou and W. Saad, "Optimal sampling and updating for minimizing age of information in the internet of things," in *Proceedings of IEEE Global Communications Conference (GLOBECOM)*, 2018, pp. 1–6.

[19] C. Li, S. Li, and Y. T. Hou, "A general model for minimizing age of information at network edge," in *Proceedings of IEEE International Conference on Computer Communications (INFOCOM)*, 2019.

[20] P.-C. Yew, N.-F. Tzeng, and D. Lawrie, "Distributing hot-spot addressing in large-scale multiprocessors," *IEEE Transactions on Computers*, vol. C-36, no. 4, pp. 388–395, 1987.

[21] A. M. Bedewy, Y. Sun, and N. B. Shroff, "The age of information in multihop networks," 2017. [Online]. Available: https://arxiv.org/abs/1712.10061

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

[23] Y.-P. Hsu, E. Modiano, and L. Duan, "Scheduling algorithms for minimizing age of information in wireless broadcast networks with random arrivals," 2017. [Online]. Available: https://arxiv.org/abs/1712.07419

[24] I. Kadota, A. Sinha, E. Uysal-Biyikoglu, R. Singh, and E. Modiano, "Scheduling policies for minimizing age of information in broadcast wireless networks," 2018. [Online]. Available: https://arxiv.org/abs/1801.01803

[25] J. Selen, Y. Nazarathy, L. L. H. Andrew, and L. V. Hai, "The age of information in gossip networks," *Environmental Technology*, vol. 27, no. 8, pp. 863–873, 2013.

[26] A. Arafa and S. Ulukus, "Age-minimal transmission in energy harvesting two-hop networks," in *Proceedings of IEEE Global Communications Conference (GLOBECOM)*, 2017, pp. 1–6.

[27] B. T. Bacinoglu, E. T. Ceran, and E. Uysal-Biyikoglu, "Age of information under energy replenishment constraints," in *Proceedings of Information Theory and Applications Workshop (ITA)*, 2015, pp. 25–31.

[28] R. D. Yates, "Lazy is timely: Status updates by an energy harvesting source," in *Proceedings of IEEE International Symposium on Information Theory (ISIT)*, 2015, pp. 3008–3012.