Minimizing Age of Processed Information in Wireless Networks

Chanikarn Nikunram*[†], Wasin Meesena*[‡], Stephen John Turner[†], Sucha Supittayapornpong[†]

Abstract—The freshness of real-time status processing of timesensitive information is crucial for several applications, including healthcare monitoring and autonomous vehicles. This freshness is considered in this paper for the system where unprocessed information is sent from sensors to a base station over a shared wireless network. The base station has a dedicated nonpreemptive processor with a constant processing time to process information from each sensor. The age of processed information is the time elapsed since the generation of the packet that was most recently processed by a processor. Our objective is to minimize the average age of processed information over an infinite timehorizon. We first show that a drop-free policy simplifies the system without sacrificing optimality. From this simplification, we propose three transmission-scheduling policies with 2-optimal guarantees for different requirements. A distributed Power-2 policy can be implemented without a central scheduler. With a central scheduler, both Back-Off and Max-Weight policies are near optimal with different advantages. The Back-Off policy guarantees a bound on the maximum age of processed information, while the Max-Weight policy achieves the lowest average age in simulation without the guarantee of bound. Simulation results confirm our theoretical findings.

Index Terms—Age of information, Transmission scheduling, Max-weight, Scheduling policy, Wireless network, Optimization

I. INTRODUCTION

Real-time status processing finds various applications, including healthcare monitoring [1] and autonomous vehicles [2], [3]. These applications collect real-time information from sensors over wireless communication, process the obtained information at a base station, and utilize the processed information for making decisions as illustrated in Figure 1. For example, an autonomous vehicle needs to process information in real-time to drive safely. The vehicle's control system receives data wirelessly from sensors, and data from each sensor is then processed by a dedicated processor.

To ensure that systems always make informed decisions based on up-to-date data, the freshness of processed information becomes one of the key factors. The freshness of information (AoI) ([4], [5] and references therein), which is the time elapsed since the generation of the packet that the base station most recently received. AoI has been extensively studied from queueing analysis, including single-server queues [6], multisource queues [7], non-preemptive queues [8], and parallel-server queues [9], to scheduling problems, such as throughput constraints [10], power constraints [11], [12], single-hop [13]–[16], multi-hop [17]–[19], both offline and online policies



Fig. 1. Time-sensitive information in a wireless network

[20]. Moreover, [21]–[23] have investigated AoI from the perspective of joint sampling and scheduling policies. However, those studies focus on the *age of information unprocessed* (AoIU) of packets that have just arrived at the base station. In some circumstances, the newly received information must be processed before the systems can use it. Thus, we must incorporate this processing time into scheduling procedures.

In this paper, the age of information processed (AoIP), which is the time elapsed since the generation of the packet that was most recently processed by a processor, is modeled to quantify the freshness of information for the real-time status processing. As shown in Figure 1, time-sensitive and unprocessed information is transmitted from sensors to a base station over a shared wireless channel. To process information from each sensor, the base station possesses a dedicated non-preemptive processor with a constant processing time, and each processor may process one packet at a time. Each processor is said to have fresher information when a new packet has been fully processed-rather than when a new packet arrives at the base station. Our goal is to develop scheduling policies for a resource-constrained wireless channel that minimizes the average age of processed information over an infinite time-horizon. While the AoIP can better capture the aspect of processed information, it is challenging to derive, compared to the AoIU. Thus, we concentrate on a subclass of scheduling policies, namely drop-free policies, that make the analysis of the average age of processed information more tractable. We show that this subclass of policies does not sacrifice any optimality and derives a lower performance bound to the optimal average sum of AoIP. Afterward, we present three drop-free scheduling policies, called the Power-2 policy, the Back-Off policy, and the Max-Weight policy. They all achieve 2-optimal performance guarantees with different advantages.

To elaborate, the Power-2 policy is low-complexity and fully distributed. On the other hand, with central scheduler employing a greedy approach, the Back-Off policy can signif-

[†]Vidyasirimedhi Institute of Science and Technology, Thailand. [‡]Carleton College, United States. *Both authors contributed equally to this research.



Fig. 2. A system model consists of N sensors and dedicated processors with different processing times. Each gray block represents a progress bar. For example, the total processing time of processor 2 is 3 slots, and the current progress is 1/3.

icantly outperform the Power-2 policy in terms of the average age. Furthermore, the Back-Off policy guarantees a bound on the maximum age of processed information. Lastly, the Max-Weight policy is a centralized policy that uses a Lyapunov Optimization technique. In simulation results, it provides the average age closest to the optimal. However, the Max-Weight policy cannot guarantee a bound on the maximum age of processed information.

The contributions of this paper are threefold:

- We model the age of processed information in a wireless network with processors and show that a class of dropfree policies tractably minimizes AoIP without sacrificing any optimality.
- We analyze a lower performance bound of the system and design three drop-free scheduling policies with 2-optimal guarantees.
- We simulate these policies and compare them with the lower performance bounds to confirm our analytical results and show that the Max-Weight and Power-2 policies perform near-optimal.

This paper is organized as follows. The system model is formally presented in Section II. Section III introduces the dropfree policies and analyzes their lower performance bounds. Three policies with different advantages are proposed and analyzed in Section IV. In Section V, these policies are simulated to evaluate their performances, followed by the conclusion in Section VI.

II. SYSTEM MODEL

A. Network and Processors

Consider a wireless network system with N nodes attempting to send packets, containing *time-sensitive, unprocessed information*, to their corresponding N processors at a base station, as shown in Figure 2. Each node and its corresponding processor are indexed by the index set $\mathcal{N} = \{1, \ldots, N\}$. For example, sensors in an autonomous vehicle send the measured data wirelessly to a central control system, the base station in our model, with a dedicated processor for each sensor.

Time is slotted, $t \in \{1, 2, ...\}$. In each slot t, at most one node can transmit a packet to the base station over a wireless channel due to interference, which is similar to [10], [13], [15], [16], [18], [20]–[22], [24]. Let $x_i(t)$ be a decision variable that equals 1 when node i transmits a packet to the base station

in slot t, and $x_i(t) = 0$ otherwise. Hence, the *interference* constraint is defined by

$$\sum_{i \in \mathcal{N}} x_i(t) \le 1, \quad \forall t \in \{1, 2, \ldots\}.$$
(1)

We assume that every transmission is always successful when the interference constraint is satisfied [13], [16], [22].

In this paper, we consider the class of all scheduling policies, denoted by Π , that make transmission decision $\{x_i(t)\}_{i \in \mathcal{N}}$ every slot such that the interference constraint is satisfied. Henceforth, we assume all transmissions satisfy the interference constraint. Every transmitted packet is assumed to take one slot to arrive at the base station [10], [11], [13], [16], [18]. When a packet arrives at the base station, it is redirected immediately to a corresponding processor.

All processors are non-preemptive. Each processor i can process one packet at a time, and the processing time takes D_i consecutive slots for any constant $D_i \in \mathbb{N}$. Therefore, when a new packet for processor i arrives at time t and the processor is idle, not processing any packet, the packet will be completely processed at the end of slot $t+D_i-1$ and will be considered as processed information in slot $t+D_i$. However, if processor i is busy (not idle), the newly arrived packet will be *dropped* and is no longer considered for processing. We call this situation wasteful transmission. Note that the system could introduce queues to store newly-arrived packets while processors are not idle. However, later we will show in Section III that those queues are unnecessary for optimality.

B. Wasteful Transmission

To prevent wasteful transmission, where a transmitted packet is dropped, we introduce $f_i(t)$ as an indicator variable that equals 1 if a transmission from node *i* in slot *t* will be wasteful, and $f_i(t) = 0$ otherwise. We assume all processors are initially idle, so $f_i(1) = 0, \forall i \in \mathcal{N}$. When node *i* transmits a packet in slot *t* knowing that the transmission will not be wasteful, i.e., $x_i(t) = 1$ and $f_i(t) = 0$, at the beginning of slot t + 1 the processor *i* will be *busy* processing this transmitted packet until the end of slot $t + D_i$ as illustrated in Figure 2. This implies that $f_i(\tau) = 1$ for $\tau \in \{t + 1, \dots, t + D_i - 1\}$ and $f_i(t + D_i) = 0$. In particular, any packet from node *i* transmitted during $\{t + 1, \dots, t + D_i - 1\}$ will be dropped, but a packet transmitted at time $t + D_i$ will get processed right after the previous processing completes.

If node *i* does not transmit in slot *t* and $f_i(t) = 0$, then $f_i(t+1) = 0$. However, if node *i* transmits a packet in slot *t* knowing that the transmission will be wasteful, i.e. $x_i(t) = 1$ and $f_i(t) = 1$, then the transmitted packet will be dropped in slot t+1 right after it has been redirected to processor *i*. Also, this wasteful transmission will not affect the values of $f_i(t)$. The dynamic of $f_i(t)$ is illustrated in Figure 3.

C. Age of Information

The Age of Information (AoI) represents how fresh the information received by the base station is. In this paper, we extend AoI to two metrics, considering unprocessed and



Fig. 3. The green dots represent transmissions from node *i*. The pink dots represent unwasteful transmissions when those packets that would not be dropped at the base station are generated and transmitted to the base station. The crossed $x_i(t)$ represents a wasteful transmission. A wasteful transmission affects $A_i^U(t)$ but does not affect $A_i^P(t)$, as show in slots 7, 15, and 16.

processed information. They are formally defined with their relationship below. We also assume that every packet is generated right before transmission, and its age starts at zero.

The Age of Information Unprocessed (AoIU) of node *i* at time *t* represents how old the transmitted packet that the base station most recently received from node *i* is. This AoIU is denoted by a positive integer $A_i^U(t)$. When node *i* transmits a packet at time *t*, i.e., $x_i(t) = 1$, and other nodes do not, this newly-generated packet from node *i* will arrive at the base station in slot t + 1, and $A_i^U(t + 1) = 1$ because the packet is generated right before the transmission and takes one slot to arrive. When there is no transmission from node *i*, the AoIU gets older by one slot, and $A_i^U(t + 1) = A_i^U(t) + 1$. The $A_i^U(t + 1)$ is illustrated in Figure 3 and is written as follows

$$A_{i}^{U}(t+1) = \begin{cases} 1 & \text{, if } x_{i}(t) = 1; \\ A_{i}^{U}(t) + 1 & \text{, otherwise.} \end{cases}$$
(2)

To measure the AoIU of the entire system when a policy π is employed, we consider the *Average Weighted Sum of AoIU* (AWSAoIU) as follows, for every $\pi \in \Pi$,

$$J_U^{\pi} = \lim_{T \to \infty} \frac{1}{NT} \sum_{t=1}^T \sum_{i \in \mathcal{N}} w_i A_i^U(t)$$
(3)

where w_i is a positive weight of node *i*.

The Age of Information Processed (AoIP) of node *i* at time *t* represents how old the packet that processor *i* most recently finished processing is. This AoIP is denoted by a positive integer $A_i^P(t)$. When processor *i* has just finished processing a packet at the end of slot *t*, we know that $A_i^P(t+1) = D_i + A_i^U(t - D_i + 1) = D_i + 1$ because i) the processing time is D_i and ii) the received packet at the base station is not dropped and is immediately processed. In particular, this situation happens only when the packet is transmitted at time $t - D_i$, and the transmission is unwasteful, i.e., $x_i(t - D_i) = 1$ and $f_i(t - D_i) = 0$. When there is no processing completion of processor *i* at time *t*, the AoIP gets older by one slot, and $A_i^P(t+1) = A_i^P(t)+1$. The $A_i^P(t+1)$ is illustrated in Figure 3 and is written as follows

$$A_i^P(t+1) = \begin{cases} D_i + 1 & \text{, if } x_i(t-D_i) = 1 \text{ and} \\ & f_i(t-D_i) = 0; \\ A_i^P(t) + 1 & \text{, otherwise.} \end{cases}$$
(4)

To measure AoIP of the entire system when a policy π is employed, we consider the *Average Weighted Sum of AoIP* (AWSAOIP) as follows, for every $\pi \in \Pi$,

$$J_P^{\pi} = \lim_{T \to \infty} \frac{1}{NT} \sum_{t=1}^T \sum_{i \in \mathcal{N}} w_i A_i^P(t)$$
(5)

with the same weight w_i defined in (3).

The initial AoIU and AoIP can be any arbitrary positive integers. For simplicity, we assume that $A_i^U(1) = A_i^P(1) = 1$ for all $i \in \mathcal{N}$.

D. System Objective

In this paper, we develop scheduling policies that minimize the AWSAoIP in (5) under the interference constraint in (1) as follows

$$J_P^* = \min_{\pi \in \Pi} \left\{ \lim_{T \to \infty} \frac{1}{NT} \sum_{t=1}^T \sum_{i \in \mathcal{N}} w_i A_i^P(t) \right\}$$
(6a)

s.t.
$$\sum_{i \in \mathcal{N}} x_i(t) \le 1, \quad \forall t \in \{1, 2, ...\},$$
 (6b)

where J_P^* is the minimum AWSAoIP achieved under a scheduling policy that satisfies the interference constraint. While we aim to minimize the AWSAoIP, the evolution of $A_i^P(t)$ in (4) is too cumbersome to deal with, compared to $A_i^U(t)$ in (2). The next section considers a special class of policies that render our analysis tractable.

Note that, while the scheduling problem in (6) is deterministic, computing the optimal policy offline needs to search over a policy space, which is exponentially large with respect to Nand $\prod_{i\in\mathcal{N}} D_i$. For example, let $L_i(t)$ be the state of processor i at time t. At time t, a policy π could map the current system state, $\{A_i^P(t), L_i(t)\}_{i\in\mathcal{N}}$, to a scheduling decision at time t+1. In this case, searching for an optimal policy π for large N and $\prod_{i\in\mathcal{N}} D_i$ is intractable. Furthermore, implementing the policy requires a large memory space to store the policy mapping.

III. DROP-FREE POLICIES AND PRELIMINARY

We observe that wasteful transmission consumes network resources without improving AWSAoIP. Therefore, we consider a *drop-free scheduling policy* that avoids scheduling a transmission from nodes that will cause a packet drop, once the packet arrives at the base station. We denote the class of drop-free policies by Π^{DF} . Note that $\Pi^{DF} \subseteq \Pi$. Next, we introduce some useful properties of drop-free policies, derive a lower performance bound, and prove a preliminary lemma.

A. Drop-free Policies

For each node *i*, we begin with decomposing the first *T* slots into intervals between transmissions. Let $V_i[T]$ be the total number of packets transmitted by node *i* within these *T* slots, i.e., $\sum_{t=1}^{T} x_i(t) = V_i[T]$. Let $I_i[m]$ be the number of slots between the (m-1)th and *m*th packet deliveries from node *i* for all $m \in \{1, 2, \ldots, V_i[T]\}$, and let R_i be the number of remaining slots after the last transmission to slot *T*. Therefore, we have

$$T = \sum_{m=1}^{V_i[T]} I_i[m] + R_i, \quad \forall i \in \mathcal{N}.$$

$$(7)$$

Next, we show that drop-free policies lead to a simple relationship between unprocessed and processed ages.

Lemma 1. For any drop-free scheduling policy, the relationship between $A_i^U(t)$ in (2) and $A_i^P(t)$ in (4) always follows:

$$A_i^P(t+D_i) = A_i^U(t) + D_i, \quad \forall t \ge I_i[1] + 1, \ \forall i \in \mathcal{N}.$$
(8)

Proof. We will prove this by induction. Let π be a drop-free policy. Since $x_i(t) = 1$ implies that $f_i(t) = 0$, the evolution of $A_i^P(t)$ in (4) can be rewritten as

$$A_i^P(t+1) = \begin{cases} D_i + 1 & \text{, if } x_i(t-D_i) = 1; \\ A_i^P(t) + 1 & \text{, otherwise.} \end{cases}$$
(9)

Basic Step: Since $x_i(I_i[1]) = 1$, we know from (2) and (9) that

$$A_i^U(I_i[1]+1) + D_i = 1 + D_i = A_i^P(I_i[1]+D_i+1).$$

Inductive Step: Suppose $k \ge I_i[1] + 1$ is an integer such that

$$A_i^P(k+D_i) = A_i^U(k) + D_i.$$
 (10)

For k + 1 and $x_i(k) = 1$, we have from (2) and (9) that

$$A_i^U(k+1) + D_i = D_i + 1 = A_i^P(k+D_i+1).$$

For k + 1 and $x_i(k) = 0$, we have that

$$A_i^U(k+1) + D_i = A_i^U(k) + 1 + D_i$$

= $A_i^P(k+D_i) + 1 = A_i^P(k+D_i+1),$

where the first equality uses (2), the second equality uses the induction hypothesis (10), and the last equality uses (9). Hence, the equality in (8) holds by the proof of induction. \Box

The implication of Lemma 1 is that the relationship between AoIU and AoIP becomes much simpler under drop-free policies. The next lemma considers the relationship between AWSAoIU and AWSAoIP.

Lemma 2 (AWSAoIU and AWSAoIP). For any system with $\{w_i, D_i\}_{i \in \mathcal{N}}$, the following holds for every drop-free scheduling policy, $\pi \in \Pi^{DF}$,

$$J_P^{\pi} = J_U^{\pi} + \frac{1}{N} \sum_{i \in \mathcal{N}} w_i D_i, \qquad (11)$$

where J_P^{π} and J_U^{π} are defined repectively in (5) and (3).

Proof. Consider a drop-free policy $\pi \in \Pi^{DF}$. Manipulating the AWSA0IP in (5) gives

$$J_P^{\pi} = \lim_{T \to \infty} \frac{1}{NT} \sum_{i \in \mathcal{N}} w_i \left[\sum_{t=1}^{I_i[1] + D_i} A_i^P(t) + \sum_{t=I_i[1] + D_i + 1}^T A_i^P(t) \right]$$

Knowing the first term above disappears after applying the limit $T \to \infty$, we apply Lemma 1 as π is drop-free and obtain

$$J_P^{\pi} = \lim_{T \to \infty} \frac{1}{NT} \sum_{i \in \mathcal{N}} w_i \sum_{t=I_i[1]+1}^{T-D_i} \left(A_i^U(t) + D_i \right)$$
$$= \lim_{T \to \infty} \frac{1}{NT} \sum_{i \in \mathcal{N}} w_i \left[\sum_{t=1}^{T-D_i} \left(A_i^U(t) + D_i \right) - \sum_{t=1}^{I_i[1]} \left(A_i^U(t) + D_i \right) \right]$$

Knowing the second term in the RHS disappears in the limit, we use the definition of AWSAoIU in (3) to prove (11) from

$$J_P^{\pi} = \lim_{T \to \infty} \frac{1}{NT} \sum_{i \in \mathcal{N}} w_i \sum_{t=1}^{T-D_i} A_i^U(t) + \frac{1}{N} \sum_{i \in \mathcal{N}} w_i D_i.$$

Next, we argue that considering the class of drop-free policies does not affect the optimal AWSAoIP in (6a).

Lemma 3 (Drop-free Optimal Scheduler). For any system with $\{w_i, D_i\}_{i \in \mathcal{N}}$, there exists a drop-free optimal scheduler.

Proof. Assume that S is an optimal scheduler that knows which node to transmit in each slot to achieve the optimal AWSA0IP. If the scheduler S causes wasteful transmissions, then we can simply construct a new scheduler S^* by replicating S but without transmitting those packets that would have been dropped according to the scheduler S. By doing so, S^* is a drop-free scheduler, and its AWSA0IP is not higher than the optimal AWSA0IP under the optimal schedule S. Thus, S^* is a drop-free optimal scheduler.

Note that even if a processor is preemptive, meaning it has the capability to drop a packet it is processing, there still exists an optimal drop-free policy because we can tweak an optimal policy for preemptive processors by not scheduling those packets that would have been preempted later. In particular, the tweaked policy is drop-free and achieves the same optimality for the system with preemptive processors.

Combining Lemma 2 and Lemma 3, we have that

$$J_P^* = \min_{\pi \in \Pi^{DF}} J_P^{\pi} = \min_{\pi \in \Pi^{DF}} J_U^{\pi} + \frac{1}{N} \sum_{i \in \mathcal{N}} w_i D_i.$$
(12)

This simplifies our optimization only to minimize AWSAoIU over drop-free policies. However, finding the optimal AWSAoIU is challenging, so we mitigate this by deriving a lower performance bound and later show that our policies in Section IV achieve the bound.

B. Lower Performance Bound

We consider the *long-term transmission rate* and derive its property before proceeding to the lower bound. The long-term transmission rate of node *i* under policy π is defined by, for every $\pi \in \Pi$,

$$r_{i}^{\pi} = \lim_{T \to \infty} \frac{1}{T} \sum_{t=1}^{T} x_{i}(t) = \lim_{T \to \infty} \frac{V_{i}[T]}{T}.$$
 (13)

Lemma 4 (Maximum Transmission Rates). For any system with $\{w_i, D_i\}_{i \in \mathcal{N}}$ and for any drop-free policy $\pi \in \Pi^{DF}$, the long-term transmission rate from each node is at most the reciprocal of the corresponding processing time. In particular, the following holds

$$r_i^{\pi} \le \frac{1}{D_i}, \quad \forall i \in \mathcal{N}.$$
 (14)

Proof. Fix node $i \in \mathcal{N}$. Consider the first T slots. Since π is a drop-free policy, the inter-delivery times during T slots must be at least D_i , i.e.,

$$I_i[m] \ge D_i, \quad \forall i \in \mathcal{N} \text{ and } m \in \{2, 3, \dots, V_i[T]\}$$

Applying the above inequality to (7), we get

$$T = \sum_{m=1}^{V_i[T]} I_i[m] + R_i \ge (V_i[T] - 1) D_i.$$

Manipulating the inequality above, applying the limit $T \to \infty$, and using the definition of the long-term transmission rate in (13) yields

$$\frac{1}{D_i} \ge \lim_{T \to \infty} \left[\frac{V_i[T]}{T} - \frac{1}{T} \right] = r_i^{\pi}.$$

Theorem 1 (Lower bound). For any system with $\{w_i, D_i\}_{i \in \mathcal{N}}$, the optimal AWSA0IP J_P^* has the lower bound L_B that is the solution of the following optimization problem, i.e., $J_P^* \ge L_B$ and

$$L_B = \min_{\pi \in \Pi^{DF}} \left\{ \frac{1}{N} \sum_{i \in \mathcal{N}} w_i \left(D_i + \frac{1}{2} + \frac{1}{2r_i^{\pi}} \right) \right\}$$
(15a)

$$s.t.\sum_{i\in\mathcal{N}}r_i^{\pi} \le 1 \tag{15b}$$

$$0 < r_i^{\pi} \le \frac{1}{D_i}, \quad \forall i \in \mathcal{N}.$$
 (15c)

Proof. Consider a drop-free policy $\pi \in \Pi^{DF}$ that satisfies the interference constraint throughout the first T slots. Fix node i, and recall that the notations $V_i[T]$, $I_i[m]$, and R_i are introduced in Section III-A for equality (7). For each interval $I_i[m]$, the unprocessed age $A_i^U(t)$ evolves as $\{1, 2, \ldots, I_i[m]\}$ according to the dynamic of $A_i^U(t)$ in (2). Similarly, the age evolves as $\{1, 2, \ldots, R_i\}$ during the remaining R_i slots. Therefore, the time-average age of information unprocessed of node i during the first T slots is as follows

$$\frac{1}{T}\sum_{t=1}^{T}A_{i}^{U}(t) = \frac{1}{T}\left[\sum_{m=1}^{V_{i}[T]}\frac{(I_{i}[m]+1)I_{i}[m]}{2} + \frac{(R_{i}+1)R_{i}}{2}\right]$$
(16)
$$= \frac{1}{2T}\left[\sum_{m=1}^{V_{i}[T]}I_{i}[m]^{2} + R_{i}^{2}\right] + \frac{1}{2}.$$

Applying Cauchy's Inequality to the above equality gives

$$\frac{1}{T}\sum_{t=1}^{T}A_{i}^{U}(t) \geq \frac{1}{2T}\frac{\left[\sum_{m=1}^{V_{i}[T]}I_{i}[m] + R_{i}\right]^{2}}{V_{i}[T] + 1} + \frac{1}{2}$$
$$= \frac{T}{2(V_{i}[T] + 1)} + \frac{1}{2}.$$

Applying the limit $T \to \infty$ to the above and using the definition of the long-term transmission rate (13), we obtain the lower bound of the time-average age of information unprocessed for node *i* as

$$\begin{split} \lim_{T \to \infty} \frac{1}{T} \sum_{t=1}^{T} A_i^U(t) \geq \frac{1}{2 \lim_{T \to \infty} \left[\frac{V_i[T]}{T} + \frac{1}{T}\right]} + \frac{1}{2} \\ \geq \frac{1}{2r_i^{\pi}} + \frac{1}{2}. \end{split}$$

Using the bound above, we can bound the AWSAoIU in (3) by

$$J_U^{\pi} = \frac{1}{N} \sum_{i \in \mathcal{N}} w_i \lim_{T \to \infty} \frac{1}{T} \sum_{t=1}^T A_i^U(t)$$

$$\geq \frac{1}{N} \left[\sum_{i \in \mathcal{N}} \frac{w_i}{2r_i^{\pi}} + \sum_{i \in \mathcal{N}} \frac{w_i}{2} \right].$$
(17)

The lower bound in (17) is a function of long-term transmission rates. Using the definition of these rates in (13) and summing over $i \in \mathcal{N}$, we have

$$\sum_{i\in\mathcal{N}} r_i^{\pi} = \lim_{T\to\infty} \frac{1}{T} \sum_{i\in\mathcal{N}} \sum_{t=1}^T x_i(t) = \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^T \sum_{i\in\mathcal{N}} x_i(t) \le 1$$
(18)

where the last inequality holds because the policy satisfies the interference constraint in (1). Minimizing both sides of (17) over drop-free policies and using (18) and (14) as the constraints give

$$\min_{\pi \in \Pi^{DF}} J_U^{\pi} \ge \min_{\pi \in \Pi^{DF}} \frac{1}{N} \sum_{i \in \mathcal{N}} w_i \left(\frac{1}{2r_i^{\pi}} + \frac{1}{2} \right)$$
(19)

where $\sum_{i \in \mathcal{N}} r_i^{\pi} \leq 1$ and $0 < r_i^{\pi} \leq \frac{1}{D_i}$ for all $i \in \mathcal{N}$. Finally, using (19) and (12) proves the theorem and (15).

Intuitively, Theorem 1 provides a lower performance bound of an optimal drop-free policy. It is later used to analyze the performances of three scheduling algorithms in Section IV. Note that our lower bound in (15) differs from [10], [22] in that i) it involves processing constraints and ii) it is under a class of drop-free policies.

Furthermore, it is possible to show that this lower bound also holds for another system with queues placed in front of the processors, where processors can be either preemptive or non-preemptive. In other words, if some drop-free scheduling policy achieves the bound, that policy is also near-optimal for the system with queues. This claim can be proven by considering a rate of processed packets. Since this rate cannot exceed the transmission rate and packets get older when being queued, we can derive a similar bound as in (15).

The bound in (15) can be made explicit by finding the set of *optimal rates* $\{r_i^*\}_{i\in\mathcal{N}}$ that minimizes (15a) under constraints (15b) and (15c). These optimal rates can be found from solving the convex optimization in (15) by any off-the-shelf solver, for example, CVX [25]. Substituting the optimal rates $\{r_i^*\}_{i\in\mathcal{N}}$ into (15a) gives

$$J_P^* \ge L_B = \frac{1}{N} \sum_{i \in \mathcal{N}} w_i \left(\frac{1}{2r_i^*} + \frac{1}{2} + D_i \right).$$
(20)

Furthermore, when $\sum_{i \in \mathcal{N}} \frac{1}{D_i} \leq 1$, it is easy to see that $r_i^* = \frac{1}{D_i}$ for all node *i*, and we have

$$J_P^* \ge L_B = \frac{1}{N} \sum_{i \in \mathcal{N}} w_i \left(\frac{3}{2}D_i + \frac{1}{2}\right).$$
(21)

C. Preliminary of Performance Guarantee

To guarantee the performance of a scheduling policy $\pi \in \Pi$, we say π is k-optimal if $J_P^{\pi} \leq k \times J_P^*$ where $J_P^* = \min_{\pi' \in \Pi} J_P^{\pi'}$. We first prove the following lemma, which will be used in the next section.

Lemma 5 (inter-deliver-optimal). When a scheduling policy has the property that $\left\lceil \frac{1}{r_i^*} \right\rceil \leq I_i[m] \leq 2 \left\lceil \frac{1}{r_i^*} \right\rceil$ for every $i \in \mathcal{N}$ and $m \in \mathbb{N}$, the policy is 2-optimal. Moreover, it is 4/3optimal if $\sum_{i \in \mathcal{N}} \frac{1}{D_i} \leq 1$.

Proof. Considering the first T slots with the similar derivation in (16) and using the condition that R_i , which is lower than $I_i[V_i[T] + 1]$, and $I_i[m]$ do not exceed $2\left\lceil \frac{1}{r_i^*} \right\rceil$, we have

$$\frac{1}{T} \sum_{t=1}^{T} A_i^U(t) = \frac{1}{2T} \left[\sum_{m=1}^{V_i[T]} I_i[m]^2 + R_i^2 \right] + \frac{1}{2}$$
$$\leq \frac{2 \left[\frac{1}{r_i^*} \right]}{2T} \left[\sum_{m=1}^{V_i[T]} I_i[m] + R_i \right] + \frac{1}{2}$$
$$= \left[\frac{1}{r_i^*} \right] + \frac{1}{2}.$$

Since $I_i[m] \ge \left\lceil \frac{1}{r_i^*} \right\rceil \ge \frac{1}{r_i^*} \ge D_i$, π is a drop-free policy. Therefore, by Lemma 2 and the inequality above, we obtain

$$J_P^{\pi} = J_U^{\pi} + \sum_{i \in \mathcal{N}} w_i D_i$$

=
$$\lim_{T \to \infty} \frac{1}{T} \sum_{t=1}^T \sum_{i \in \mathcal{N}} w_i A_i^U(t) + \sum_{i \in \mathcal{N}} w_i D_i$$

$$\leq \sum_{i \in \mathcal{N}} w_i \left(\left\lceil \frac{1}{r_i^*} \right\rceil + \frac{1}{2} + D_i \right).$$
(22)

Since $\left[\frac{1}{r_i^*}\right] \leq \frac{1}{r_i^*} + D_i$ as $D_i \geq 1$, the inequality (22) yields that π is 2-optimal according to (20) as

$$J_P^{\pi} \le \sum_{i \in \mathcal{N}} w_i \left(\frac{1}{r_i^*} + \frac{1}{2} + 2D_i \right)$$
$$\le 2 \sum_{i \in \mathcal{N}} w_i \left(D_i + \frac{1}{2} + \frac{1}{2r_i^*} \right) = 2J_P^*$$

Also, if $\sum_{i \in \mathcal{N}} \frac{1}{D_i} \leq 1$, then $r_i^* = \frac{1}{D_i}$ for all $i \in \mathcal{N}$ and, the inequality (22) yields that π is 4/3-optimal according to (21) and

$$J_P^{\pi} \le \sum_{i \in \mathcal{N}} w_i \left(D_i + \frac{1}{2} + D_i \right)$$
$$\le \frac{4}{3} \sum_{i \in \mathcal{N}} w_i \left(\frac{3}{2} D_i + \frac{1}{2} \right) \le \frac{4}{3} J_P^*.$$

IV. SCHEDULING POLICIES

In this section, we present three scheduling policies, namely, Power-2, Back-Off, and Max-Weight, with their performance guarantees.

A. Power-2 Policy

The Power-2 policy minimizes AWSAoIP with a 2-optimal guarantee. It is distributed, cyclic, and has low complexity. The policy is inspired by the work in [16] for a different problem that guarantees the upper bound of AoI without minimizing AWSAoIU explicitly. Here, we present our policy and derive its performance guarantee.

Without loss of generality, node indices are rearranged such that $r_1^* \ge r_2^* \ge \ldots \ge r_N^*$. We define $D_i^{PW} = 2^{\lceil \log_2(1/r_i^*) \rceil}$ as a fixed inter-delivery time for source node *i*. Note that $D_1^{PW} \le D_2^{PW} \le \ldots \le D_N^{PW}$. Then, for each node *i*, we determine a *basic time* B_i , in which every packet from node *i* is transmitted in slot *t* where $t \equiv B_i \mod D_i^{PW}$. Intuitively, B_i is the first slot node *i* transmits a packet and it keeps transmitting every D_i^{PW} slots. Therefore, node *i* only needs know B_i and D_i^{PW} to make transmission decisions independently from other nodes, assuming the starting time is synchronized.

The procedure for determining the basic time $\{B_i\}_{i \in \mathcal{N}}$ that the policy is drop-free and satisfies the interference constraint in (1) is summarized in Algorithm 1 with an example provided below. **Algorithm 1:** Procedure for determining basic time $\{B_i\}$

Input :
$$\{D_i^{PW}\}_{i \in \mathcal{N}}$$

Output: $\{B_i\}_{i \in \mathcal{N}}$
for $i \in \{1, \dots, N\}$ do
 $\mathcal{P}_i = \{t \in \{1, 2, \dots, D_i^{PW}\} | t \neq B_j \mod D_j^{PW}, \forall j < i\}$
 $B_i = \min \mathcal{P}_i$
end

Example: A system consisting of three nodes with $\{D_i\}_{i=1}^3 = \{2,2,4\}$ and $\{w_i\}_{i=1}^3 = \{20,5,1\}$. Solving problem (15) obtains $\{r_i^*\}_{i=1}^3 \approx \{\frac{1}{2}, \frac{1}{2.9}, \frac{1}{6.5}\}$, and we set $\{D_i^{PW}\}_{i=1}^3 = \{2^{\lceil \log_2(1/r_i^*) \rceil}\}_{i=1}^3 = \{2,4,8\}$. Starting from the first node, we have $\mathcal{P}_1 = \{1,2\}$ and $B_1 = 1$. Next, we have $\mathcal{P}_2 = \{2,4\}$ and $B_2 = 2$. Lastly, we have $\mathcal{P}_3 = \{4,8\}$ and $B_3 = 4$. Therefore, node 1 transmits a packet in slots $\{1,3,5,7,\ldots\}$ without communicating with the other nodes. Similarly, node 2 transmits in slots $\{2,6,10,14,\ldots\}$, and node 3 transmits in slots $\{4,12,20,\ldots\}$.

Now, we show that Algorithm 1 always constructs the basic times $\{B_i\}_{i \in \mathcal{N}}$ before showing that the interference constraint in (1) is satified by the transmissions generated by these basic times.

Theorem 2 (Existence of Basic Times). For any system with $\{w_i, D_i\}_{i \in \mathcal{N}}$, Algorithm 1 always leads to non-empty \mathcal{P}_i for all $i \in \mathcal{N}$.

Proof. Let $\{D_i^{PW}\}_{i \in \mathbb{N}}$ be the result of a given network setting. We use induction to show that \mathcal{P}_i is non-empty for all $i \in \mathcal{N}$.

Basic Step: Since $D_i^{PW} \ge 1$, we have that $\mathcal{P}_1 = \{1, 2, \dots, D_1^{PW}\}$ and is non-empty.

Inductive Step: Let $k \in \{1, 2, ..., N-1\}$ such that all \mathcal{P}_k are non-empty. We will show that \mathcal{P}_{k+1} is non-empty as follows.

$$\begin{aligned} |\mathcal{P}_{k+1}| \\ &= \left| \left\{ 1, 2, \dots, D_{k+1}^{PW} \right\} / \bigcup_{j=1}^{k} \left\{ t \in \left\{ 1, \dots, D_{k+1}^{PW} \right\} \right\} \\ &\left| t \equiv B_j \mod D_j^{PW} \right| \end{aligned} (23) \\ &\geq D_{k+1}^{PW} - \sum_{j=1}^{k} \left| \left\{ t \in \left\{ 1, 2, \dots, D_{k+1}^{PW} \right\} \right\} \end{aligned}$$

$$\left|t \equiv B_j \mod D_j^{PW}\right|$$
. (24)

Because all D_i^{PW} are power of 2 and $D_1^{PW} \leq \ldots \leq D_{k+1}^{PW}$, we know that D_{k+1}^{PW} is divisible by D_j^{PW} for every $j \leq k$. It

follows that

$$\left|\left\{t \in \{1, 2, \dots, D_{k+1}^{PW}\} \mid t \equiv B_j \mod D_j^{PW}\right\}\right| = \frac{D_{k+1}^{PW}}{D_j^{PW}}$$

Applying the property above to (23) gives

$$|\mathcal{P}_{k+1}| \ge D_{k+1}^{PW} - \sum_{j=1}^{k} \frac{D_{k+1}^{PW}}{D_j^{PW}} = D_{k+1}^{PW} \left(1 - \sum_{j=1}^{k} \frac{1}{D_j^{PW}}\right).$$

Using the facts that $\frac{1}{D_i^{PW}} \leq 2^{-\log_2(1/r_i^*)} = r_i^*$ and $\sum_{i \in \mathcal{N}} r_i^* \leq 1$, as $\{r_i^*\}_{i \in \mathcal{N}}$ satisfies (15b), we have

$$|\mathcal{P}_{k+1}| \ge D_{k+1}^{PW} \left(1 - \sum_{j=1}^{k} r_j^*\right) \ge D_{k+1}^{PW} \left(r_{k+1}^*\right)$$
$$\ge D_{k+1}^{PW} \frac{1}{D_{k+1}^{PW}} = 1.$$

Therefore, \mathcal{P}_{k+1} is non-empty, which proves the theorem. \Box

Theorem 3 (No Concurrent Transmissions). For any system with $\{w_i, D_i\}_{i \in \mathcal{N}}$, the Power-2 policy satisfies the interference constraint in (1).

Proof. Let $\{D_i^{PW}\}_{i \in N}$ be the result of a given network setting. Suppose that the Power-2 policy violates the interference constraint. There exist $i, j \in \mathcal{N}$ in which i > j and $t \in \mathbb{N}$ such that $t \equiv B_i \mod D_i^{PW}$ and $t \equiv B_j \mod D_j^{PW}$, as nodes i and j are scheduled to transmit in the same slot. However, since D_i^{PW} is divisible by D_j^{PW} , the fact that $t \equiv B_i \mod D_i^{PW}$ leads to $t \equiv B_i \mod D_j^{PW}$. It follows that $B_j \equiv B_i \mod D_j^{PW}$. However, by the procedure for constructing B_i in Algorithm 1, the fact that $B_i \not\equiv B_j \mod D_j^{PW}$ leads to a contradiction. Therefore, the policy satisfies the interference constraint.

We finally state the performance guarantee of the Power-2 policy.

Theorem 4 (Performance Guarantee of Power-2). For any system with $\{w_i, D_i\}_{i \in \mathcal{N}}$, the Power-2 policy is 2-optimal, and it is 4/3-optimal if $\sum_{i \in \mathcal{N}} \frac{1}{D_i} \leq 1$.

Proof. To determine the AWSAoIP of the Power-2 policy over an infinite time horizon, we can simply ignore the first inter-delivery time of every node. According to the policy, $I_i[m] = 2^{\lceil \log_2(1/r_i^*) \rceil}$ for every $i \in \mathcal{N}$ and $m \in \{2, 3, 4, \ldots\}$. It holds that

$$I_{i}[m] = 2^{\lceil \log_{2}(1/r_{i}^{*}) \rceil} \leq 2 \cdot 2^{\log_{2}(1/r_{i}^{*})} \leq 2 \lceil 1/r_{i}^{*} \rceil$$

and $I_{i}[m] = 2^{\lceil \log_{2}(1/r_{i}^{*}) \rceil} \geq \left\lceil 2^{\log_{2}(1/r_{i}^{*})} \right\rceil = \lceil 1/r_{i}^{*} \rceil.$

Applying Lemma 5 proves the theorem.

Intuitively, the Power-2 policy utilizes a set of fixed interdeliver times to achieve the 2-optimal guarantee with simple and distributed implementation. However, this comes at a cost of under-utilization of network resources, i.e., no transmission even when some processor is idle. The next policy improves on this issue.

B. Back-Off Policy

The Back-Off policy employs a greedy approach to select a node for transmission without causing wasteful transmissions. This is done by restricting the inter-delivery time to be at least $D_i^{BO} = \lceil 1/r_i^* \rceil$ for every node $i \in \mathcal{N}$. Later, we show that the policy is 2-optimal and outperforms the Power-2 policy in simulations. In particular, it has bounded maximum AoIP.

The Back-Off policy maintains a list of candidate nodes for drop-free transmission in every slot. Let $b_i(t)$ be an indicator variable that equals 0 if node *i* is a candidate for slot *t*. If $b_i(t)$ equals 1, node *i* is *backed off* from the list. We initialize $b_i(t) = 0 \ \forall i \in \mathcal{N}$ and $\forall t \in \mathbb{N}$, and then $b_i(t)$ is updated dynamically as follows. If node *i* transmits during slot *t*, it will back off from the list for $D_i^{BO} - 1$ slots. That is $b_i(\tau) = 1$ where $\tau \in \{t+1, t+2, \ldots, t+D_i^{BO}-1\}$. Since $\lceil 1/r_i^* \rceil \geq D_i$, the Back-Off policy is drop-free.

There could be multiple candidate nodes in a slot. Therefore, we introduce a *countdown time* $C_i(t)$ as a variable that quantifies how urgent node *i* should transmit in slot *t*. This $C_i(t)$ is initialized by $C_i(1) = D_i^{BO}$ and evolves as follows

$$C_{i}(t) = \begin{cases} D_{i}^{BO} & ,t > D_{i}^{BO} \text{ and } x_{i}(t - D_{i}^{BO}) = 1; \\ \infty & ,b_{i}(t) = 1; \\ C_{i}(t - 1) - 1 & , \text{otherwise.} \end{cases}$$
(25)

That is, when node *i* is backed-off, the countdown time is infinite. If it has just stopped being backed-off, then $C_i(t)$ is set to D_i^{BO} . If the node is not backed-off and it does not transmit a packet, then the countdown time reduces by 1. Intuitively, the lower $C_i(t)$ is, the more urgent node *i* needs to transmit a packet in slot *t*. Thus, in each slot *t*, the Back-Off policy schedules a transmission from node *j* where

$$j = \underset{i \in \mathcal{N}: b_i(t) = 0}{\operatorname{arg\,min}} C_i(t)$$

Also, if all nodes are backed off, $b_i(t) = 1$ for every $i \in \mathcal{N}$, nothing is transmitted during slot t. The Back-Off policy is summarized in Algorithm 2.

Example: Table I illustrates the Back-Off policy using the same setup in Section IV-A. Given $\{D_i\}_{i=1}^3 = \{2, 2, 4\}$ and $\{w_i\}_{i=1}^3 = \{20, 5, 1\}$, we have $\{r_i^*\}_{i=1}^3 \approx \{\frac{1}{2}, \frac{1}{2,9}, \frac{1}{6.5}\}$, and $\{D_i^{BO}\}_{i=1}^3 = \{\lceil 1/r_i^*\rceil\}_{i=1}^3 = \{2, 3, 7\}$. We initialize the countdown times with $\{C_i(1)\}_{i=1}^3 = \{2, 3, 7\}$. Although processor 3 takes $D_3 = 4$ slots to process information and a transmission in the 8th slot by node 3 is not wasteful, $f_3(8) = 0$, node 3 is still backed-off until t = 10. The countdown time is reset later in the 11th slot, $C_3(11) = 7$.

The performance of the Back-Off policy can be analyzed by showing that every countdown time never reaches a zero as stated in the following lemma. Note that we set $D_i^{BO} = \lceil 1/r_i^* \rceil$ for all $i \in \mathcal{N}$, so $\sum_{i=1}^N \frac{1}{D_i^{BO}} \leq 1$ always holds.

Lemma 6 (No-zero). If $\sum_{i=1}^{N} \frac{1}{D_i^{BO}} \leq 1$, then we have that $C_i(t) \geq 1$ for every $i \in \mathcal{N}$ and every $t \in \{1, 2, \ldots\}$.

Algorithm 2: Back-Off Policy

t	1	2	3	4	5	6	7	8	9	10	11	12
$C_1(t)$	1	∞	2	8	2	∞	2	∞	2	∞	2	8
$C_2(t)$	3	2	∞	∞	3	2	∞	∞	3	Ź	∞	∞
$C_3(t)$	7	6	5	A	∞	∞	∞	∞	∞	∞	7	ø
$b_3(t)$	0	0	0	0	1	1	1	1	1	1	0	0
$f_3(t)$	0	0	0	0	1	1	1	0	0	0	0	0
TABLE I												

AN EXAMPLE OF THE BACK-OFF POLICY

Proof. Since this proof requires defining new definitions and proving other three lemmas, it is provided in the Appendix. The high-level idea is as follows. We use the proof of contradiction by first assuming that there exists a non-stop transmission period such that some countdown reaches 0. However, due to the countdown times and back-off periods, the number of transmissions during this non-stop transmission period from each node is limited. As a result, the maximum total number of transmissions during this period is less than the length of the period, suggesting that there must be a slot with no transmission during the non-stop period and contradicting the assumption that the period is non-stop. \Box

We then use Lemma 6 to prove the following guarantee.

Theorem 5 (Performance Guarantee of Back-Off). For any system with $\{w_i, D_i\}_{i \in \mathcal{N}}$, the Back-Off policy is 2-optimal, and it is 4/3-optimal if $\sum_{i \in \mathcal{N}} \frac{1}{D_i} \leq 1$.

Proof. Since the back-off period is $D_i^{BO} - 1 = \lceil 1/r_i^* \rceil - 1$, the interdelivery time $I_i[m]$ is at least $\lceil 1/r_i^* \rceil$. From Lemma 6, the countdown times are always positive, so the inter-delivery time $I_i[m]$ is no longer than the sum of back-off period and the longest countdown period, i.e., $2 \lceil 1/r_i^* \rceil - 1$. Therefore, we have $\lceil 1/r_i^* \rceil \leq I_i[m] \leq 2 \lceil 1/r_i^* \rceil$, and the policy is drop-free. Invoking Lemma 5 proves the theorem.

Corollary 1 (Bounded Maximum AoIP of Back-Off). *The Back-Off policy has bounded maximum AoIP satisfying*

$$A_i^P(t) \le 2 \left\lceil \frac{1}{r_i^*} \right\rceil + D_i - 1 \quad \forall i \in \mathcal{N}, \forall t \ge I_i[1] + D_i + 1.$$

Proof. Since the Back-Off policy is drop-free, we can write $A_i^P(t)$ as $A_i^U(t-D_i) + D_i$ for every $t \ge I_i[1] + D_i + 1$ by Lemma 1. Since, the maximum inter-delivery time is at most $2 \lceil 1/r_i^* \rceil - 1$ and every packet takes one slot to arrive at the base station, we have $A_i^U(t-D_i) \le 2 \lceil 1/r_i^* \rceil - 1$ and

$$A_i^P(t) = A_i^U(t - D_i) + D_i \le 2\left[\frac{1}{r_i^*}\right] + D_i - 1.$$

The Back-Off policy still has unused slots while some processors are idle. The next section considers a greedy scheduling policy that always makes unwasteful transmission when possible.

C. Max-Weight Policy

The Max-Weight policy is a drop-free policy that requires a central scheduler. It is derived from a Lyapunov Optimization technique [5], [10]. We first derive the policy and then prove its performance.

The network state is defined by $S(t) = \{A_i^U(t), f_i(t)\}_{i \in \mathcal{N}}$, where $A_i^U(t)$ and $f_i(t)$ are AoIU and the wasteful transmission indicator defined in Section II. We consider the linear Lyapunov function

$$L(S(t)) = \frac{1}{N} \sum_{i \in \mathcal{N}} \tilde{\alpha}_i A_i^U(t)$$
(26)

where every $\tilde{\alpha}_i$ is some positive constant. The Lyapunov drift is defined by

$$\Delta(S(t)) = L(S(t+1)) - L(S(t)).$$
(27)

The evolution of $A_i^U(t+1)$ in (2) can be rewritten as

$$A_i^U(t+1) = x_i(t) + (1 - x_i(t)) \left(A_i^U(t) + 1 \right).$$
 (28)

Substituting (28) and (26) to (27), we get

$$\Delta(S(t)) = \frac{1}{N} \sum_{i \in \mathcal{N}} \tilde{\alpha}_i \left[A_i^U(t+1) - A_i^U(t) \right]$$
$$= -\frac{1}{N} \sum_{i \in \mathcal{N}} x_i(t) \tilde{\alpha}_i A_i^U(t) + \frac{1}{N} \sum_{i \in \mathcal{N}} \tilde{\alpha}_i.$$
(29)

The Max-Weight policy minimizes the Lyapunov drift in (29) to reduce the progress of AoIU in every slot. Since the only controllable variable is the transmission $x_i(t)$, the policy selects node *i* that i) causes no wasteful transmission, $f_i(t) = 0$, and ii) has the largest coefficient, $\tilde{\alpha}_i A_i^U(t)$. In particular, node *j* is selected for transmission when

$$j = \underset{i \in \mathcal{N}: f_i(t) = 0}{\arg \max} \tilde{\alpha}_i A_i^U(t)$$

The Max-Weight policy with $\tilde{\alpha}_i = w_i/r_i^*$ for every $i \in \mathcal{N}$ is summarized in Algorithm 3. Recall that r_i^* is the optimal

Algorithm 3: Max-Weight Policy

rate of the problem in (15). The performance of Max-Weight is proven in Theorem 6.

Theorem 6 (Performance Guarantee of Max-Weight). For any system with $\{w_i, D_i\}_{i \in \mathcal{N}}$ and corresponding optimal rate $\{r_i^*\}_{i \in \mathcal{N}}$, the Max-Weight policy with $\tilde{\alpha}_i = w_i/r_i^*$ for all $i \in \mathcal{N}$ is 2-optimal, and it is 5/3-optimal if $\sum_{i \in \mathcal{N}} \frac{1}{D_i} \leq 1$.

Proof. Let $x_i(t)$ represent the decision variable with respect to the Max-Weight policy, π^{MW} . Since the sum of the optimal rate is at most 1 as in (18) and the policy selects the node with the highest $\tilde{\alpha}_i A_i^U(t)$, we have

$$\sum_{i\in\mathcal{N}:f_i(t)=0} x_i(t)\tilde{\alpha}_i A_i^U(t) \ge \sum_{i\in\mathcal{N}:f_i(t)=0} r_i^*\tilde{\alpha}_i A_i^U(t).$$
(30)

Applying (30) to (29), we obtain

$$\Delta(S(t)) \leq -\frac{1}{N} \sum_{i \in \mathcal{N}: f_i(t)=1} x_i(t) \tilde{\alpha}_i A_i^U(t) + \frac{1}{N} \sum_{i \in \mathcal{N}} \tilde{\alpha}_i - \frac{1}{N} \sum_{i \in \mathcal{N}: f_i(t)=0} r_i^* \tilde{\alpha}_i A_i^U(t).$$

Since node *i* with $f_i(t) = 1$ is not selected for transmission, leading to $x_i(t) = 0$, the first term in the RHS above is 0 and we have

$$\Delta(S(t)) \le \frac{1}{N} \sum_{i \in \mathcal{N}} \tilde{\alpha}_i - \frac{1}{N} \sum_{i; f_i(t) = 0} r_i^* \tilde{\alpha}_i A_i^U(t).$$

Rewriting the last sum above to

$$\frac{1}{N} \sum_{i \in \mathcal{N}} r_i^* \tilde{\alpha}_i A_i^U(t) \left(1 - f_i(t)\right)$$

gives

$$\Delta(S(t)) \leq \frac{1}{N} \sum_{i \in \mathcal{N}} \tilde{\alpha}_i - \frac{1}{N} \sum_{i \in \mathcal{N}} r_i^* \tilde{\alpha}_i A_i^U(t) + \frac{1}{N} \sum_{i \in \mathcal{N}} r_i^* \tilde{\alpha}_i A_i^U(t) f_i(t).$$

Summing the above over $t \in \{1, 2, ..., T\}$ and dividing by T Applying the above inequality to (31), we achieve vields

$$\frac{L(S(T+1))}{T} - \frac{L(S(1))}{T}$$

$$\leq \frac{1}{N} \sum_{i \in \mathcal{N}} \tilde{\alpha}_i - \frac{1}{NT} \sum_{t=1}^T \sum_{i \in \mathcal{N}} r_i^* \tilde{\alpha}_i A_i^U(t)$$

$$+ \frac{1}{NT} \sum_{t=1}^T \sum_{i \in \mathcal{N}} r_i^* \tilde{\alpha}_i A_i^U(t) f_i(t)$$

Knowing $\frac{L(S(t+1))}{T}$ is positive, we manipulate the above to

$$\frac{1}{NT} \sum_{t=1}^{T} \sum_{i \in \mathcal{N}} r_i^* \tilde{\alpha}_i A_i^U(t) \le \frac{L(S(1))}{T} + \frac{1}{N} \sum_{i \in \mathcal{N}} \tilde{\alpha}_i + \frac{1}{NT} \sum_{t=1}^{T} \sum_{i \in \mathcal{N}} r_i^* \tilde{\alpha}_i A_i^U(t) f_i(t).$$

Setting $\tilde{\alpha}_i = w_i/r_i^*$ in the above and applying the limit $T \rightarrow$ ∞ with knowing that L(S(1)) is a constant, we get

$$\lim_{T \to \infty} \frac{1}{NT} \sum_{t=1}^{T} \sum_{i \in \mathcal{N}} w_i A_i^U(t) \le \frac{1}{N} \sum_{i \in \mathcal{N}} \frac{w_i}{r_i^*} + \lim_{T \to \infty} \frac{1}{NT} \sum_{t=1}^{T} \sum_{i \in \mathcal{N}} w_i A_i^U(t) f_i(t).$$
(31)

Next, we bound the last term in (31). Fix node *i*. Consider packet transmissions during T slots as in (7). For m > 1, the slot that follows the (m-1)th packet delivery from node i has $A_i^U(t) = 1$ and $f_i(t) = 1$, and $A_i^U(t)f_i(t)$ evolves as $\{1, 2, \ldots, D_i - 1, 0, \ldots, 0\}$ until the slot that the mth packet is transmitted. Therefore, the sum of $A_i^U(t)f_i(t)$ during the inter-deliver times between the (m-1)th and mth transmissions is $D_i(D_i-1)/2$, and the sum during the remaining R_i slots is at most $D_i(D_i-1)/2$. Since early slots before the first transmission, $f_i(t)$ is set to 0, and we have

$$\frac{1}{T} \sum_{t=1}^{T} A_i^U(t) f_i(t) \\
\leq \frac{1}{T} \left[(V_i[T] - 1) \frac{D_i(D_i - 1)}{2} + \frac{D_i(D_i - 1)}{2} \right] \\
= \frac{D_i(D_i - 1)}{2} \frac{V_i[T]}{T}.$$

Taking limit $T \to \infty$ to the above and using the definition of the long-term transmission rate, we have

$$\lim_{T \to \infty} \frac{1}{T} \sum_{t=1}^{T} A_i^U(t) f_i(t) \le \frac{D_i (D_i - 1)}{2} r_i^{\pi^{MW}}$$

Applying Lemma 4 to the above as π^{MW} is drop-free yields

$$\lim_{T \to \infty} \frac{1}{T} \sum_{t=1}^{T} A_i^U(t) f_i(t) \le \frac{D_i - 1}{2}.$$

$$\lim_{T \to \infty} \frac{1}{NT} \sum_{t=1}^{T} \sum_{i \in \mathcal{N}} w_i A_i^U(t)$$
$$\leq \frac{1}{N} \sum_{i \in \mathcal{N}} \frac{w_i}{r_i^*} + \frac{1}{N} \sum_{i \in \mathcal{N}} w_i \frac{D_i - 1}{2}.$$

Combining Lemma 2 with π^{MW} and the above inequality gives

$$J_P^{\pi^{MW}} \le \frac{1}{N} \left[\sum_{i \in \mathcal{N}} \frac{w_i}{r_i^*} + \sum_{i \in \mathcal{N}} w_i \frac{D_i - 1}{2} + \sum_{i \in \mathcal{N}} w_i D_i \right]$$
(32)

Comparing (32) with (20), the Max-Weight policy is 2-optimal as

$$J_P^{\pi^{MW}} \le 2\left[\frac{1}{N}\sum_{i\in\mathcal{N}}\left(\frac{w_i}{2r_i^*} + \frac{w_i}{2} + w_iD_i\right)\right] \le 2J_P^*.$$

Furthermore, if $\sum_{i \in \mathcal{N}} \frac{1}{D_i} \leq 1$, then $r_i^* = \frac{1}{D_i}$. Hence, substituting $r_i^* = \frac{1}{D_i}$ to (32) and comparing with (21), we have

$$J_P^{\pi^{MW}} \le \frac{1}{N} \left[\sum_{i \in \mathcal{N}} w_i D_i + \sum_{i \in \mathcal{N}} w_i \frac{D_i - 1}{2} + \sum_{i \in \mathcal{N}} w_i D_i \right]$$
$$\le \frac{5}{3} \left[\frac{1}{N} \sum_{i \in \mathcal{N}} \left(\frac{3}{2} w_i D_i + \frac{w_i}{2} \right) \right] \le \frac{5}{3} J_P^*.$$

Thus, the Max-Weight policy is 5/3-optimal if $\sum_{i \in \mathcal{N}} \frac{1}{D_i} \leq$ 1.

V. SIMULATION RESULTS

This section evaluates the performance of Power-2, Back-Off, and Max-Weight policies in two aspects: AWSAoIP and maximum AoIP. We use the CVXPY package [26] to solve the convex optimization in (15) for optimal transmission rates. For each scheduling policy, we simulate 10^6 slots to ensure AWSAoIP converges.

A. Average age of information processed

We evaluate AWSAoIP under the three policies in networks with a varying number of nodes. We begin with a group of 5 nodes with processing times (24, 152, 70, 37, 54) and weights (4.1, 7.2, 1.1, 3.0, 1.4). These processing times yield $\sum_{i=1}^{5} \frac{1}{D_i} \approx 0.12$. In Figure 4, we simulate 40 network setups. Each setup contains a multiple of groups, and the largest setup has $5 \times 40 = 200$ nodes.

The simulation result shows that both centralized policies-Back-Off and Max-Weight policies— are near-optimal as their AWSA0IP are close to the lower bound in (15). For large networks, Max-Weight policy slightly outperforms the Back-Off policy. The Power-2 policy trades performance for the simple distributed algorithm, yet it still achieves 2-optimal. The AWSAoIPs from these policies are flat until N = 40because $\sum_{i=1}^{N} \frac{1}{D_i} \leq 1$, so the systems are processor bound. After N = 40, the systems are communication bound and AWSAoIP increases linearly with the number of nodes.



Fig. 4. AWSAoIPs with varying numbers of nodes



Fig. 5. The distributions of $A_1^P(t)$ with $D_1 = 2$ and $A_9^P(t)$ with $D_1 = 16$ under the Back-Off and Max-Weight policies. The red dashed lines represent the bounds on the maximum AoI derived in Corollary (1).

B. Maximum age of information processed

We illustrate that the Back-Off policy guarantees a bound on the maximum AoIP as in Corollary 1. Consider a network setup consisting of 9 nodes, where $D_1 = 2$ and $D_i = 16$ for $i \in \{2, 3, \ldots, 9\}$ and all weights are 1. The $J_P^{\pi^{BO}}$ and $J_P^{\pi^{MW}}$ in this setup are 199.5 and 202.0 respectively. Figure 5 shows the frequencies of $A_1^P(t)$ and $A_9^P(t)$ under both policies, and the red dash lines are the maximum AoIP from Corollary 1. The Back-Off policy keeps all AoIPs within the maximum. Furthermore, it is possible to construct a setup for the Max-Weight policy that the frequency of event $A_1^P(t) = y$ is non-zero while $r_1^* = 0.5$ for any positive integer y, and the maximum AoIP is unbounded linearly by $c(1/r_i^*)$ for any constant c.

VI. CONCLUSIONS AND FUTURE RESEARCH

This paper considered a wireless network with nodes transmitting unprocessed packets to their corresponding nonpreemptive processors at the base station over a shared wireless channel. We designed three scheduling policies: Power-2, Back-Off, and Max-Weight to minimize the Average Weighted sum *AoIP* of the system. All policies are 2-optimal and have different advantages. The Power-2 policy is simple and distributed. The Back-Off policy is near-optimal and has a maximum AoIP bound guarantee. The Max-Weight policy achieves closest to the system's lower bound. We confirm the results through simulations. Our subsequent work will consider a system model with unreliable wireless channels, where we have some preliminary results. Future research can consider the periodic generation of packets, minimum throughput constraints, and random processing times.

REFERENCES

- O. Salem and A. Mehaoua, "A secure framework for remote healthcare monitoring using the internet of medical things," in *ICC 2022 - IEEE International Conference on Communications*, 2022, pp. 1233–1238.
- [2] T. Zeng, O. Semiari, W. Saad, and M. Bennis, "Joint communication and control system design for connected and autonomous vehicle navigation," in *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, 2019, pp. 1–6.
- [3] H. Wang, B. Kim, J. Xie, and Z. Han, "E-auto: A communication scheme for connected vehicles with edge-assisted autonomous driving," in *ICC* 2019 - 2019 IEEE International Conference on Communications (ICC), 2019, pp. 1–6.
- [4] 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 Communications*, vol. 39, no. 5, pp. 1183–1210, 2021.
- [5] Y. Sun, I. Kadota, R. Talak, and E. Modiano, "Age of information: A new metric for information freshness," *Synthesis Lectures on Communication Networks*, vol. 12, no. 2, pp. 1–224, 2019.
- [6] S. K. Kaul, R. D. Yates, and M. Gruteser, "Real-time status: How often should one update?" in *Proceedings of the IEEE INFOCOM* 2012, Orlando, FL, USA, March 25-30, 2012, A. G. Greenberg and K. Sohraby, Eds. Orlando, FL, USA: IEEE, 2012, pp. 2731–2735. [Online]. Available: https://doi.org/10.1109/INFCOM.2012.6195689
- [7] M. Moltafet, M. Leinonen, and M. Codreanu, "On the age of information in multi-source queueing models," *IEEE Transactions on Communications*, vol. 68, no. 8, pp. 5003–5017, 2020.
- [8] D. Deng, Z. Chen, Y. Jia, L. Liang, S. Fang, and M. Wang, "Age of information in a multiple stream m/g/1/1 non-preemptive queue," in *ICC* 2021 - *IEEE International Conference on Communications*. Montreal, QC, Canada: IEEE, 2021, pp. 1–6.
- [9] R. D. Yates, "Status updates through networks of parallel servers," in 2018 IEEE International Symposium on Information Theory, ISIT 2018, Vail, CO, USA, June 17-22, 2018. ail, CO, USA: IEEE, 2018, pp. 2281– 2285. [Online]. Available: https://doi.org/10.1109/ISIT.2018.8437907
- [10] I. Kadota, A. Sinha, and E. Modiano, "Scheduling algorithms for optimizing age of information in wireless networks with throughput constraints," *IEEE/ACM Transactions on Networking*, vol. 27, no. 4, pp. 1359–1372, 2019.
- [11] H. Tang, J. Wang, L. Song, and J. Song, "Minimizing age of information with power constraints: Multi-user opportunistic scheduling in multi-state time-varying channels," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 5, pp. 854–868, 2020.
- [12] A. M. Arafa, J. Yang, S. Ulukus, and H. V. Poor, "Timely status updating over erasure channels using an energy harvesting sensor: Single and multiple sources," *IEEE Transactions on Green Communications and Networking*, vol. 6, pp. 6–19, 2022.
- [13] Y.-P. Hsu, E. Modiano, and L. Duan, "Scheduling algorithms for minimizing age of information in wireless broadcast networks with random arrivals," *IEEE Transactions on Mobile Computing*, vol. 19, pp. 2903–2915, 2020.
- [14] I. Kadota, A. Sinha, E. Uysal-Biyikoglu, R. Singh, and E. Modiano, "Scheduling policies for minimizing age of information in broadcast wireless networks," *IEEE/ACM Transactions on Networking*, vol. 26, no. 6, pp. 2637–2650, 2018.
- [15] I. Kadota and E. H. Modiano, "Minimizing the age of information in wireless networks with stochastic arrivals," *IEEE Transactions on Mobile Computing*, vol. 20, pp. 1173–1185, 2021.
- [16] C. Li, S. Li, Y. Chen, Y. T. Hou, and W. Lou, "Aoi scheduling with maximum thresholds," in 39th IEEE Conference on Computer Communications, INFOCOM 2020, Toronto, ON, Canada, July 6-9, 2020. Toronto, ON, Canada: IEEE, 2020, pp. 436–445. [Online]. Available: https://doi.org/10.1109/INFOCOM41043.2020.9155514

- [17] O. Vikhrova, F. Chiariotti, B. Soret, G. Araniti, A. Molinaro, and P. Popovski, "Age of information in multi-hop networks with priorities," in *GLOBECOM 2020 - 2020 IEEE Global Communications Conference*, 2020, pp. 1–6.
- [18] S. Farazi, A. G. Klein, and D. R. Brown, "Fundamental bounds on the age of information in multi-hop global status update networks," *Journal* of Communications and Networks, vol. 21, no. 3, pp. 268–279, 2019.
- [19] A. M. Bedewy, Y. Sun, and N. B. Shroff, "The age of information in multihop networks," *IEEE/ACM Transactions on Networking*, vol. 27, no. 3, pp. 1248–1257, 2019.
- [20] Y.-P. Hsu, E. Modiano, and L. Duan, "Scheduling algorithms for minimizing age of information in wireless broadcast networks with random arrivals," *IEEE Transactions on Mobile Computing*, vol. 19, no. 12, pp. 2903–2915, 2020.
- [21] Z. Jiang, S. Zhou, Z. Niu, and Y. Cheng, "A unified sampling and scheduling approach for status update in multiaccess wireless networks," in 2019 IEEE Conference on Computer Communications, INFOCOM 2019, Paris, France, April 29 - May 2, 2019. Paris, France: IEEE, 2019, pp. 208–216. [Online]. Available: https://doi.org/10.1109/INFOCOM.2019.8737404
- [22] C. Li, S. Li, Y. Chen, Y. T. Hou, and W. Lou, "Minimizing age of information under general models for iot data collection," *IEEE Trans. Netw. Sci. Eng.*, vol. 7, no. 4, pp. 2256–2270, 2020. [Online]. Available: https://doi.org/10.1109/TNSE.2019.2952764
- [23] A. M. Bedewy, Y. Sun, S. Kompella, and N. B. Shroff, "Optimal sampling and scheduling for timely status updates in multi-source networks," *IEEE Transactions on Information Theory*, vol. 67, pp. 4019– 4034, 2021.
- [24] E. Gindullina, L. Badia, and D. Gündüz, "Age-of-information with information source diversity in an energy harvesting system," *IEEE Transactions on Green Communications and Networking*, vol. 5, no. 3, pp. 1529–1540, 2021.
- [25] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 2.1," http://cvxr.com/cvx, Mar. 2014.
- [26] S. Diamond and S. Boyd, "Cvxpy: A python-embedded modeling language for convex optimization," *Journal of Machine Learning Research*, vol. 17, no. 83, pp. 1–5, 2016. [Online]. Available: http://jmlr.org/papers/v17/15-408.html

APPENDIX

The proof of Lemma 6 requires defining new definitions and proving the other three lemmas. Therefore, we begin with the requirements and prove Lemma 6 at the end of this appendix.

First, we introduce several definitions. Define

$$S(t) = \begin{cases} i & , x_i(t) = 1 \text{ for some } i \in \mathcal{N}; \\ 0 & , x_i(t) = 0 \text{ for all } i \in \mathcal{N} \end{cases}$$

to indicate which node is selected to transmit a packet during slot t as shown in Table II. That is, if $S(t) \neq 0$, then $C_{S(t)}(t)$ is the countdown time of the node selected to transmit a packet in slot t.

We define the *wait gap* G_t as the set of the time slots that the node selected to transmit in slot t has waited since it stopped being backed off. For example, in Table II, node i = 3 is scheduled for a transmission in slot t = 12, and we have that $G_{12} = \{12, 11\}$. Generally, it holds that

$$G_{t} = \begin{cases} \{t, t-1, \dots, t-D_{S(t)}^{BO} + C_{S(t)}(t)\} \\ , \sum_{n \in \mathcal{N}} x_{n}(t) = 1; \\ \emptyset & , \sum_{n \in \mathcal{N}} x_{n}(t) = 0. \end{cases}$$

We then define a *Connected Chain* as an order of wait gaps $G_{t_1}, G_{t_2}, \ldots, G_{t_h}$ where $t_1 > t_2 > \ldots > t_h$ and $G_{t_i} \cap G_{t_{i+1}} \neq \emptyset$ for $i = 1, \ldots, h-1$. The last condition and the definition of wait gap implies that a connected chain is a duration that has a packet transmission in every slot. Finally, define

t	1	2	3	4	5	6	7	8	9	10	11	12
S(t)	1	2	1	3	1	2	1	0	1	2	1	3
$C_1(t)$	2	8	2	∞								
$C_2(t)$	3	Ź	∞	∞	3	2	∞	∞	3	Ź	∞	∞
$C_3(t)$	7	6	5	A	∞	∞	∞	∞	∞	∞	7	ø
$b_3(t)$	0	0	0	0	1	1	1	1	1	1	0	0
$f_3(t)$	0	0	0	0	1	1	1	0	0	0	0	0
TABLE II												

Extension of Table I with S(t). Also, we have $G_4 = \{1, 2, 3, 4\}, G_3 = \{3\}, G_2 = \{1, 2\}, G_1 = \{1\}, \text{and } G_8 = \emptyset.$ Furthermore, (G_4, G_2, G_1) is a Connected Chain, but (G_4, G_3, G_2, G_1) is not since $G_3 \cap G_2 = \emptyset.$

a First Zero Connected Chain as an order of $G_{t_1}, G_{t_2}, \ldots, G_{t_h}$ that is a Connected Chain and $C_{S(t_1)}(t_1) = 0$, and there is no slot t for $0 < t < t_1$ that $C_{S(t)}(t) = 0$.

We first assume that a First Zero Connected Chain exists, which later leads to a contradiction and there is no such a chain. Suppose a First Zero Connected Chain $\Gamma = (G_{t_1}, \ldots, G_{t_k})$ exists such that $C_{i^*}(t_1) = 0$ for some node i^* . We denote $G^* = G_{t_1} \cup \ldots \cup G_{t_k}$ and $L_C = |G^*|$. We assume that Γ is with the longest length. This chain leads to the following three lemmas, as follow:

Lemma 7 (countdown time in the chain). For the First Zero Connected Chain Γ , it holds that $C_{S(t)}(t) \leq t_1 - t$ for every $t \in G^*$.

Proof. Let P(j) be the statement that $C_{S(t)}(t) \leq t_1 - t$ for all $t \in G_{t_j}$, where j = 1, 2, ..., k.

Basic Step: Since $C_{S(t_1)}(t_1) = 0$, we then have that $C_{S(t_1)}(t) = t_1 - t$ for every $t \in G_{t_1}$ by the evolution of the countdown time in in (25). Since node $S(t_1)$ does not get to transmit a packet during slot $t \in G_{t_1}/\{t_1\}$, the countdown time of the selected node in slot t must not exceed the countdown time of node $S(t_1)$. That is,

$$C_{S(t)}(t) \le C_{S(t_1)}(t) = t_1 - t \tag{33}$$

for all $t \in G_{t_1}/\{t_1\}$. Combining with the fact that $C_{S(t_1)}(t_1) = 0, P(1)$ is true.

Inductive Step: Let q be a natural number such that q < k and P(q) is true. That is, for every $t \in G_{t_q}$, $C_{S(t)}(t) \leq t_1 - t$.

Since $G_{t_q} \cap G_{t_{q+1}} \neq \emptyset$, we can see that $t_{q+1} \in G_{t_q}$ and

$$C_{S(t_{q+1})}(t_{q+1}) \le t_1 - t_{q+1}.$$
 (34)

Using the similar approach as in the Basic Step, the evolution of $C_{S(t_{a+1})}(t)$ in (25) implies that

$$C_{S(t_{q+1})}(t) = C_{S(t_{q+1})}(t_{q+1}) + (t_{q+1} - t)$$
(35)

for all $t \in G_{t_{q+1}}$.

Since $S(t_{q+1})$ does not get to transmit a packet during slot $t \in G_{t_{q+1}}/{t_{q+1}}$, the countdown time of the selected node must not exceed the countdown time of node $S(t_{q+1})$. That is, combining with (34) and (35), we have

$$C_{S(t)}(t) \le C_{S(t_{q+1})}(t)$$

$$\le (t_1 - t_{q+1}) + (t_{q+1} - t) = t_1 - t \qquad (36)$$

for all $t \in G_{t_{q+1}}/\{t_{q+1}\}$. Thus, combining with (34), P(q+1) is true.

By the proof of induction, we can conclude that, for every $t \in G^*$, $C_{S(t)}(t) \leq t_1 - t$.

Lemma 8 (processing time shorter than the chain). For the First Zero Connected Chain Γ , it holds that $D_{S(t)}^{BO} \leq L_C - 1$ for every $t \in G^*$.

Proof. Assume that there exists some $t' \in G^*$ such that $D_{S(t')}^{BO} \geq L_C$. There must be some $q \in \{1, 2, \ldots, k\}$ that $t' \in G_{t_q}$ and $t' \neq t_q$ We know that

$$|G_{t'}| = \left| \left\{ t', t' - 1, \dots, t' - D_{S(t')}^{BO} + C_{S(t')}(t') \right\} \right|$$

= $D_{S(t')}^{BO} - C_{S(t')}(t') + 1.$ (37)

Applying Lemma 7 and the assumption that $D_{S(t')}^{BO} \ge L_C$ to (37), we have

$$|G_{t'}| \ge L_C - (t_1 - t') + 1.$$
(38)

Thus, we consider $G' = G_{t_1} \cup G_{t_2} \cup \ldots \cup G_{t_q} \cup G_{t'}$. Since $t' \in G_{t_q}$, we can easily prove that $(G_{t_1}, G_{t_2}, \ldots, G_{t_q}, G_{t'})$ is a First Zero Connected Chain. However, we know that $\{t_1, t_1 - 1, \ldots, t' + 1\} \subseteq G'/G_{t'}$ and

$$|G'| = |G'/G_{t'}| + |G_{t'}|$$

$$\geq (t_1 - t') + |G_{t'}|$$

$$\geq (t_1 - t') + L_C - (t_1 - t') + 1 = L_C + 1, \quad (39)$$

which is longer than $L_C = |G^*|$. This causes a contradiction since Γ is the longest First Zero Connected Chain. Therefore, $D_{S(t)}^{BO} \leq L_C - 1$ for every $t \in G^*$.

Lemmas 7 and 8 are used to prove the following lemma.

Lemma 9 (maximum transmissions in the chain). For the First Zero Connected Chain Γ , the number of transmissions from node *i* during $t \in G^* \setminus \{t_1\}$ is at most $\lfloor (L_C - 1) / D_i^{BO} \rfloor$, for every $i \in \mathcal{N}$.

Proof. For node *i* that $D_i^{BO} > L_C - 1$, it does not get to transmit a packet within $t \in G^* \setminus \{t_1\}$ at all by Lemma 8. In other words, it transmits at most $\lfloor (L_C - 1) / D_i^{BO} \rfloor = 0$. Now consider node *i* such that $D_i^{BO} \leq L_C - 1$. If $L_C - 1$ is divisible by D_i^{BO} , then we can partition these

If $L_C - 1$ is divisible by D_i^{BO} , then we can partition these $L_C - 1$ slots into $\frac{L_C - 1}{D_i^{BO}}$ intervals of consecutive D_i^{BO} slots. For node *i*, the inter-delivery time is always at least D_i^{BO} . Thus, each interval can transmit at most 1 packet from node *i*. That is, during these $L_C - 1$ slots, there are at most $\frac{L_C - 1}{D_i^{BO}} = \lfloor (L_C - 1) / D_i^{BO} \rfloor$ transmissions from node *i*.

If L_C-1 is not divisible by D_i^{BO} , we let z_i be the remainder where $1 \le z_i < D_i^{BO}$. Again, we partition $G^* \setminus \{t_1\}$ into $\lfloor (L_C-1)/D_i^{BO} \rfloor + 1$ intervals where the last interval is $\{t_1, t_1 + 1, \dots, t_1 + z_i - 1\}$ of length z_i and the rest are $\lfloor (L_C-1)/D_i^{BO} \rfloor$ intervals where each interval is a consecutive D_i^{BO} slots. Using the similar argument, we can transmit at most $\lfloor (L_C-1)/D_i^{BO} \rfloor + 1$ where each interval has one transmission from node *i*. If we get exact $\lfloor (L_C-1)/D_i^{BO} \rfloor + 1$ transmissions, then node *i* must transmit one packet during the second-to-last interval and another one during the last interval. However, when we transmit a packet during the second-to-last interval, $C_i(t)$ will be set to ∞ for $D_i^{BO} - 1$ slots, then become D_i^{BO} , and keep decreasing by 1 in every slot until node *i* transmits another packet. We can easily check that, during the last interval, $C_i(t)$ will always greater than $t_1 - t$. However, by Lemma 7, node *i* would not be able to transmit a packet during the last z_i slots, contradicting the assumption that node *i* transmits a packet during the last interval. Therefore, for every $i \in \mathcal{N}$, there are at most $\lfloor (L_C - 1) / D_i^{BO} \rfloor$ transmissions from node *i* within $t \in G^* \setminus \{t_1\}$.

Now, we will show that if such First Zero Connected Chain Γ exists, it leads to a contradiction.

Lemma 6 (No-zero). If $\sum_{i=1}^{N} \frac{1}{D_i^{BO}} \leq 1$, then we have that $C_i(t) \geq 1$ for every $i \in \mathcal{N}$ and every $t \in \{1, 2, \ldots\}$.

Proof. We assume that there exists some t^* and i^* that $C_{i^*}(t^*) = 0$, so we must be able to detect a First Zero Connected Chain $\Gamma = (G_{t_1}, G_{t_2}, \ldots, G_{t_k})$ where $t^* = t_1$. To be concise, we denote $G^* = G_{t_1} \cup G_{t_2} \cup \ldots \cup G_{t_k}$ and $L_C = |G^*|$. Since there might be multiple First Zero Connected Chains, we assume that Γ is the First Zero Connected Chain with the largest L_C .

By Lemma 9, the total number of transmissions during interval $G^* \setminus \{t_1\}$ is at most

;

$$\sum_{\in \mathcal{N}} \left\lfloor (L_C - 1) / D_i^{BO} \right\rfloor \leq \sum_{i \in \mathcal{N}} (L_C - 1) / D_i^{BO}$$
$$= \sum_{i \in \mathcal{N}} (L_C - 1) / \left\lceil \frac{1}{r_i^*} \right\rceil$$
$$\leq \sum_{i \in \mathcal{N}} (L_C - 1) r_i^*$$
$$\leq L_C - 1. \tag{40}$$

Since the chain is connected, there must be exactly $L_C - 1$ transmissions during during interval. Thus, the inequalities in (40) must be equalities. That is, $L_C - 1$ is divisible by D_i^{BO} for every $i \in \mathcal{N}$, and node i must exactly transmit $\lfloor (L_C - 1) / D_i^{BO} \rfloor = (L_C - 1) / D_i^{BO}$ packets during this interval. We then consider node $i^* = S(t_1)$. Since $C_{i^*}(t_1) = 0$, this means that node i^* cannot transmit during interval $\{t_1, t_1 - 1, \ldots, t_1 - D_{i^*}^{BO}\}$. That is, during interval $G^* \setminus \{t_1\}$, node i^* can transmit at most $\left\lceil \frac{L_C - D_{i^*O}^{BO} - 1}{D_{i^*O}^{BO}} \right\rceil = \frac{L_C - 1}{D_{i^*O}^{BO}} - 1$, resulting in a contradiction. Therefore, $C_i(t) \ge 1$ for every $i \in \mathcal{N}$ and $t = 1, 2, \ldots$