Timely Status Update in Wireless Uplinks: Analytical Solutions with Asymptotic Optimality

Zhiyuan Jiang, Bhaskar Krishnamachari, Senior Member, IEEE, Xi Zheng, Sheng Zhou, Zhisheng Niu, Fellow, IEEE

Abstract—In a typical Internet of Things (IoT) application where a central controller collects status updates from multiple terminals, e.g., sensors and monitors, through a wireless multiaccess uplink, an important problem is how to attain timely status updates autonomously. In this paper, the timeliness of the status is measured by the recently proposed age-of-information (AoI) metric; both the theoretical and practical aspects of the problem are investigated: we aim to obtain a scheduling policy with minimum AoI and, meanwhile, requires little signaling exchange overhead. Towards this end, we first consider the set of arrival-independent and renewal (AIR) policies; the optimal policy thereof to minimize the time-average AoI is proved to be a round-robin policy with one packet (latest packet only and others are dropped) buffers (RR-ONE). The optimality is established based on a generalized Poisson-Arrival-See-Time-Average (PASTA) theorem. It is further proved that RR-ONE is asymptotically optimal among all policies in the massive IoT regime. The AoI steady-state stationary distribution under RR-ONE is also derived. An implementation scheme of RR-ONE is proposed which can accommodate dynamic terminal appearances and others are dropped) buffers (RR-ONE). The optimality is established based on a generalized Poisson-Arrival-See-Time-Average (PASTA) theorem. It is further proved that RR-ONE is asymptotically optimal among all policies in the massive IoT regime. The AoI steady-state stationary distribution under RR-ONE is also derived. An implementation scheme of RR-ONE is proposed which can accommodate dynamic terminal appearances with little overhead. In addition, considering scenarios where packets cannot be dropped, a Lyapunov optimization based max-AoI-weight policy is proposed which achieves better performance compared with state-of-the-art.

Index Terms—Internet of Things, status update, age-of-information, wireless multiaccess channel, queuing theory

I. INTRODUCTION

Internet of Things (IoT) represents one of the biggest paradigm shifts recently which can revolutionize the information technology and several aspects of everyday-life such as living, e-health and driving; it envisions to transform every physical object into an intelligent individual that is capable of sensing, communicating and computing. By 2021, Ericsson predicts that there will be around 28 billion IoT devices and a large share of them are empowered by wireless communication technologies [2]. The wireless communication community has dedicated a significant amount of efforts to accommodate such a massive number of emerging IoT devices; in particular, it is one of the main targets of the 5G system that 100-fold more connected devices per geographical area should be supported compared with current LTE systems [3]. In addition to the sheer amount of IoT devices, it is also desired to enhance the timeliness of services for time-critical applications whereby the service quality depends heavily on the freshness of the monitoring data collected from IoT devices, e.g., the Tactile Internet and autonomous driving [4].

Age-of-information (AoI) is a recently proposed metric specifically to quantify such timeliness [5]–[12]; it is a constantly evolving information monitoring delay at a destination node, or simply put, time elapsed since the last-updated packet’s generation. This definition jointly accounts for the delay introduced by sampling the information source and data communication, which distinguishes AoI from the conventional end-to-end communication (queuing and transmission) delay metric [13], [14]. Consider a concrete example in Fig. 1. The AoI only coincides with the communication delay at the time when a status update packet is delivered; another distinct difference is that the communication delay is defined for each packet; in contrast, the AoI is a constantly evolving measurement at the destination. AoI is particularly meaningful for status information that exhibits Markovian property—a new status renders an old status much less valuable or even useless; examples of such information include sensing information for temperature, pressure and etc. The remote estimation of information sources which have Markovian property is the problem of interest in this paper, and it has many applications in IoT wherein sensing plays a pivotal role. It was shown in [15, Lemma 1] that the mutual information between a Markov information source and a destination is a non-negative and non-increasing function of the AoI in a single-link scenario. Therefore, optimizing the AoI is equivalent of maximizing the mutual information between sources and destinations in remote estimations. In fact, it was further proposed [16] that for a particular class of applications, AoI
should be considered as an end metric which jointly considers several metrics, e.g., throughput, end-to-end delay and lossless or lossy designs, such that different systems with distinct characterizations in each metric can be compared fairly. The wireless communication system is therefore well motivated to optimize the AoI, however, this objective may deviate from the conventional throughput- or delay-oriented paradigms since it has been shown that the optimization of AoI leads to distinctively different system designs, e.g., sampling strategy and service principle [6].

One of the fundamental restrictions of wireless communication systems is that transmissions are subject to interference due to the broadcast nature of electromagnetic waves, leading to the fact that IoT terminals cannot transmit simultaneously; otherwise collisions happen and transmissions fail with no data delivered. Therefore, terminals should be carefully scheduled to avoid such collisions; as a result, delay is introduced. For instance, a simple scheduling strategy is that terminals take turns to update their status data to avoid collisions. Intriguingly, we will show that taking turns (a round-robin scheduling policy with proper packet management) is, to some extent, the optimal policy without entailing a large amount of signaling overhead.

The overhead issue touches upon another important design principle in wireless multiaccess uplinks, especially with massive distributed IoT devices, that is the policy design is preferably decentralized, i.e.,

**Definition 1 (Decentralized Scheduling):** A decentralized scheduling scheme is defined as a scheme wherein the transmission scheduling decisions are made autonomously at terminals and require only local information.

For instance, the carrier-sensing-medium-access (CSMA) protocol is a widely-used and successful application of decentralized protocol in wireless networks [17]. In particular, terminals transmit based on a contention protocol and scheduling decisions are made in a decentralized manner. However, the CSMA protocol is designed only for throughput maximization and may face severe challenges in status update systems. Note that this definition, in line with the CSMA scheme, allows certain signaling exchange, e.g., feedback from the central controller, as long as the scheduling decisions are made autonomously to avoid overhead scaling with the number of terminals.

Concerning the aforementioned scenario and corresponding challenges, the contributions of this paper include:

1. Among arrival-independent renewal (AIR) scheduling policies, whose decisions are independent of packet-arrival processes and hence decentralization-friendly, a round-robin policy with one-packet buffers (only retains the most up-to-date packet and others are discarded) at terminals (RR-ONE) is proved optimal. The proof technique leverages a generalized Poisson-arrival-see-time-average (PASTA) theorem which, as far as we know, has not been adopted in the related literature before.

2. RR-ONE is proved asymptotically optimal among all policies with a massive number of terminals. It is shown that the optimum time-average AoI is proportional to the number of terminals asymptotically; the optimum linear scaling factor is \( \frac{1}{2} \). RR-ONE is proved to achieve the optimum scaling factor. The pure CSMA protocol is however shown to have (at least) a scaling factor of 1; hence its time-average AoI is arbitrarily larger than RR-ONE asymptotically. In addition, the AoI steady-state distribution under RR-ONE is also derived.

3. A full-fledged decentralized implementation of RR-ONE is described; it is capable of adapting to dynamic terminal appearances which is essential for decentralized algorithms. Thereby, the only global information required for each terminal is the total number of terminals; this is obtained by a common broadcast message from the central controller.

4. Considering scenarios wherein arrival packets are queued and first-come-first-served (FCFS) at terminals without any packet management, e.g., packet dropping, we propose a Lyapunov optimization based max-AoI-weight policy. Based on simulation results, it outperforms both the pure CSMA scheme wherein terminals get equal transmission probability, and queue-aware CSMA scheme [18] wherein terminals with longer queues are prioritized.

The remainder of the paper is organized as follows. In Section II, the system model is introduced; the problem of AoI minimization is then formulated; for clarity, we present our main results here and detailed proofs and explanations are conveyed in the subsequent sections. In Section III, we show that RR-ONE is the optimal AIR policy. In Section IV, the asymptotic optimality is proved. In Section V, the stationary distribution under RR-ONE is derived. In Section VI, a decentralized protocol is presented. Section VII presents the max-AoI-weight policy without performing packet management, Section VIII presents simulation results. Finally, in Section IX, conclusions are drawn and discussions are made. The proofs of several lemmas are shown in the appendix.

### A. Related Work

An AoI optimization problem can be posed as minimizing the time-average AoI at the receiver by controlling the sampling rate at the terminal. All the sampled data packets
TABLE I

<table>
<thead>
<tr>
<th>Description of Key Notations</th>
</tr>
</thead>
<tbody>
<tr>
<td>( N ): Number of terminals.</td>
</tr>
<tr>
<td>( n ): Terminal index.</td>
</tr>
<tr>
<td>( \lambda_n ): Status packet arrival rates of terminal-( n ).</td>
</tr>
<tr>
<td>( \pi ): An admissible policy.</td>
</tr>
<tr>
<td>( \bar{h}_n^{(T,N)} ): T-horizon time-average AoI of ( N ) terminals under policy ( \pi ).</td>
</tr>
<tr>
<td>( U_{n,\pi}(t) ): Scheduling decision for terminal-( n ) under policy ( \pi ) at time ( t ).</td>
</tr>
<tr>
<td>( A_n(t) ): Age of the packet in terminal-( n )’s buffer at time ( t ) (one-packet buffer).</td>
</tr>
<tr>
<td>( \tau_n ): The time since the last time terminal-( n ) is scheduled.</td>
</tr>
<tr>
<td>( { X^{(i)}<em>{n,\pi}, Y^{(i)}</em>{n,\pi} : i=1,2,\ldots } ): The resultant scheduling interval process of terminal-( n ) based on policy ( \pi ).</td>
</tr>
<tr>
<td>( m_n ): ( \mathbb{E}[X^{(i)}_{n,\pi}] \triangleq m_n, \forall k. )</td>
</tr>
<tr>
<td>( v_n ): ( \mathbb{E}[X^{(i)}_{n,\pi}] \triangleq v_n, \forall k. )</td>
</tr>
</tbody>
</table>

**II. SYSTEM MODEL, PROBLEM FORMULATION AND MAIN RESULTS**

Consider a base station (BS), alternatively referred to as a central controller, which is responsible for collecting status update packets from a large number of IoT devices as shown in Fig. 2. A time-slotted system is considered. The status update packets are generated and stored at terminal queues. The queue buffer size for every terminal is assumed to be identical and denote by \( B \). The number of packets generated at time \( t \) of terminal \( n \) is denoted by \( L_n(t) \) and \( L_n(t) \) is assumed to be a Bernoulli random variable with parameter \( \lambda_n \); the arrival processes \( \{ L_n(t), t = 1, 2, \ldots \} \) are independent over terminals and time. The number of terminals is denoted by \( N \). Let \( U_{n,\pi}(t) \) denote the scheduling decision of terminal \( n \) at time \( t \) for a given policy \( \pi \), i.e., \( U_{n,\pi}(t) = 1 \) if terminal \( n \) is scheduled and \( U_{n,\pi}(t) = 0 \) otherwise.

The T-horizon time-average AoI is denoted by

\[
\bar{h}_n^{(T,N)} \triangleq \frac{1}{TN} \sum_{t=1}^{T} \sum_{n=1}^{N} h_n,\pi(t),
\]

where the AoI at the \( t \)-th time slot for terminal \( n \) based on policy \( \pi \) is denoted by \( h_n,\pi(t) \), and the time horizon is \( T \) time slots. Denote time-average AoI over infinite time horizon as

\[
\bar{h}_n(\infty,\pi) \triangleq \lim_{T \to \infty} \bar{h}_n^{(T,N)}.
\]

The evolution of AoI can be written as

\[
h_n,\pi(t+1) = h_n,\pi(t) - U_{n,\pi}(t) \prod_{m \neq n} (1 - U_{m,\pi}(t)) g_n,\pi(t) + 1,
\]

where \( g_n,\pi(t) \) denotes the AoI reduction with a successful update from terminal \( n \). Consequently, we have \( g_n,\pi(t) = 0 \)

\footnote{The AoI evolution in this paper is based on discrete time slots; whereas the AoI changes continuously in Fig. 1 and Fig. 4 for ease of exposition.}
when queue-$n$ is empty at time $t$. The AoI for each terminal always increases by one after each time slot. Based on this definition (3), whenever a collision happens, i.e., more than one terminals transmit in the same time slot, no status is updated. Note that transmission failures only happen with collisions, otherwise the transmission is always assumed successful; this corresponds to the interference-limited regime which is emerging to be the main application scenario in the future ultra-dense networks [29], and therein failures due to noise are negligible. In addition, denote the time-average AoI of terminal-$n$ under policy $\pi$ as

$$\bar{h}_{n,\pi}(t) = \frac{1}{T} \sum_{t=1}^{T} h_{n,\pi}(t).$$

The status update procedure is described in Fig. 2. We assume the following sequence of events in each time slot. At the beginning of each time slot, a policy performs scheduling decisions including the following:

- **Terminal scheduling**: Decide which terminal, or a set of terminals\(^2\), updates and transmits in this time slot.
- **Packet management**: Once a terminal is scheduled, the policy also determines a packet management scheme, i.e., it can choose a packet from its queue to update in the time slot, or drop arbitrary packets. In contrast, packets are queued and served based on an FCFS manner without any packet management. Note that preemption is not considered in this paper, that is, the transmission of a packet cannot be interrupted.

Based on the decision, the scheduled terminal transmits its update packet (assuming one packet is transmitted in each time slot), and thereby the AoI is refreshed at the BS. Note that by this definition no transmission (service) preemption is considered, i.e., a CSMA-type collision avoidance mechanism is applied such that a terminal would not transmit when it senses others are occupying the channel and a terminal does not preempt its own transmission. Afterwards, packets arrive randomly at terminals (the age of newly arrived packets is zero) and then the ages of all packets and all AoIs increase by one. This marks the end of a time slot in our model. The AoI at one time slot is defined as the AoI at the end of the time slot, or drop arbitrary packets. In contrast, packets are queued and served based on an FCFS manner without any packet management. Note that preemption is not considered in this paper, that is, the transmission of a packet cannot be interrupted.

Based on the decision, the scheduled terminal transmits its update packet (assuming one packet is transmitted in each time slot), and thereby the AoI is refreshed at the BS. Note that by this definition no transmission (service) preemption is considered, i.e., a CSMA-type collision avoidance mechanism is applied such that a terminal would not transmit when it senses others are occupying the channel and a terminal does not preempt its own transmission. Afterwards, packets arrive randomly at terminals (the age of newly arrived packets is zero) and then the ages of all packets and all AoIs increase by one. This marks the end of a time slot in our model. The AoI at one time slot is defined as the AoI at the end of the time slot, or drop arbitrary packets. In contrast, packets are queued and served based on an FCFS manner without any packet management. Note that preemption is not considered in this paper, that is, the transmission of a packet cannot be interrupted.

The objective considered in this paper is to minimize the infinite-horizon time-average AoI (2) over all policies.\(^3\) As a first step, the following definition and Lemma 1 (cf. proof in Appendix B) enable us to only consider work-conserving non-collision (WCNC) policies without loss of optimality.

**Definition 2 (WCNC policy)**: A WCNC policy is defined as a policy that is not idle when there is at least one packet in terminal queues, nor schedule more than one terminals simultaneously.

**Lemma 1**: For a non-WCNC policy, there exists at least one WCNC policy that achieves lower AoI.

For practical concerns that the policy decisions should be decentralized, and also mathematical tractability, define AIR policies as follows. Denote the resultant scheduling interval process of terminal-$n$ based on policy $\pi$ as $X_{n,\pi}^{(k)}$, $k = 1, 2, \ldots$ where $k$ is the scheduling interval index. Define $R_{n,\pi}(t)$ as the counting process of scheduling times before time $t$ for terminal $n$, i.e., $R_{n,\pi}(t) \triangleq \sup \{ r : \sum_{k=0}^{r} X_{n,\pi}^{(k)} \leq t \}$.

**Definition 3 (AIR policy)**: A policy $\pi$ is an AIR policy if it is causal and the following conditions are both met.

1. The scheduling interval processes $\{X_{n,\pi}^{(k)}, n = 1, \ldots, N\}$ are independent of the packet arrival processes at terminals, with finite first and second raw moments denoted by $m_{n}$ and $v_{n}$ respectively.

2. The counting processes $R_{n,\pi}(t), n = 1, \ldots, N$ are renewal processes.

By definition, the set of AIR policies is essentially a subset of all policies. The condition (1) is in fact reflecting the practical perspective that the scheduling decisions are desired to be independent of the packet arrival processes to enable decentralized implementation and reduce signaling exchange overhead. The condition (2) does enforce an additional constraint that the scheduling intervals are i.i.d.; however the distributions can be arbitrary as long as they have finite first and second moments. Note that, notwithstanding these conditions, it is found (Theorem 2) that the optimal AIR policy with proper packet management is asymptotically optimal among all policies in the massive IoT regime.

To be clear, we define non-causal policy specifically in this context, and the definition for causal policy follows immediately.

**Definition 4 (Non-causal policy)**: A policy $\pi$ that has knowledge of future packet arrival times is defined as non-causal policies.

### A. Main Results

The main results of the paper include two aspects, i.e., scheduling policies for scenarios with and without packet management. For the former wherein terminals apply packet management, the AoI performance of RR-ONE and optimality guarantees are analyzed as follows (Theorem 1 to 3).
Definition 5 (RR-ONE): RR-ONE, denoted by RR in the subscript, is defined as a policy that schedules the $n_{RR}$-th terminal at each time slot which satisfies

$$n_{RR} = \min \left\{ n : \tau_n = \max_{m=1, \ldots, N} \tau_m \right\}, \quad (5)$$

and only retains the last-arrival packet at each terminal. If no packet is present at the scheduled terminal, the terminal transmits a blank packet. The time since last update from terminal $m$ is denoted by $\tau_m$; assume $\tau_m = 0$, $\forall m$ when $t = 0$. $\square$

Theorem 1 (Optimality among AIR Policies): RR-ONE is the optimal AIR policy to minimize the time-average AoI, with

$$\bar{\mu}_{RR} = \frac{1}{N} \sum_{n=1}^{N} \frac{1}{\lambda_n} + \frac{N-1}{2}. \quad (6)$$

Theorem 2 (Asymptotic Optimality): RR-ONE is asymptotically optimal among all policies (including non-causal policies) in the massive IoT regime, i.e., it achieves the asymptotically optimal among all policies (including non-causal policies which know future packet arrivals). We emphasize that the asymptotic optimality of RR-ONE is among all policies, including policies requiring global information, e.g., terminals’ queue lengths, age of all packets, and even non-causal policies which know future packet arrivals (by observing that the proof of Theorem 2 also applies in this case).

Theorem 3 (AoI Stationary Distributions under RR-ONE): The AoI evolution of terminal-$n$ based on RR-ONE follows a Markov renewal process with a fixed renewal time of $N$ time slots, and the steady-state stationary distribution is

$$\mu_n(j) = \begin{cases} \frac{1 - (1 - \lambda_n)^j}{N}, & 1 \leq j \leq N; \\ \frac{(1 - \lambda_n)^{j-N}}{N} - \frac{1 - (1 - \lambda_n)^N}{N}, & j \geq N + 1, \end{cases} \quad (9)$$

where $\mu_n(j)$ denotes the probability that the steady state AoI of terminal-$n$ is $j$. $\square$

The steady-state stationary distribution in (9) is instantiated with an insight described in Fig. 3.

On the other hand, in scenarios wherein terminals do not apply packet managements, i.e., packets are served in a FCFS manner, we propose the max-Aoi-weight policy (Aoi-MW) based on the Lyapunov optimization technique [14].

Definition 6 (Aoi-MW): Aoi-MW is defined as a policy that schedules the $n_{MW}$-th terminal at each time slot which satisfies

$$n_{MW} = \arg \max_{n \in \{1, \ldots, N\}} \{2(h_{n, \pi}(t) + 1)g_{n, \pi}(t) - g^2_{n, \pi}(t)\}. \quad (10)$$

If no packet is present at the scheduled terminal, the terminal transmits a blank packet. $\square$

In Section III-VII, we will prove our main results and elaborate on their implementations and implications.

III. PROOF OF THEOREM 1: OPTIMAL AIR POLICY WITH PACKET MANAGEMENT

The quest for the optimal policy, among all policies with any given $N$, to minimize the time-average AoI seems elusive, because the problem can be essentially viewed as a restless multi-armed bandit problem with time- and arm-correlated reward functions [30]. Besides, there is a strong probability that the optimal policy requires global information exchange and hence decentralization-unfriendly. Therefore, in this section, we resolve to derive the optimal AIR policy to minimize the time-average Aoi in (2) following a generalized Poisson-arrival-see-time-average (PASTA) theorem, i.e., the arrival-time-average (ASTA) property with a Markov state process as the observed process and an independent outside observer [31]. First, consider the queue evolution of terminal-$n$ based on AIR policies; it is similar with an $M/G/1$ queue given the definition of AIR policies, with a subtle, but important, difference that the service (in this case the service time is the scheduling interval) begins immediately after a packet departure, even if there is no packet waiting in the queue. In the case that there is no packet in the queue, the service proceeds independently till the end, i.e., scheduled, during which period two possible circumstances can occur: 1) there are (at least one) packet arrivals and thereby one of the packets is updated under a certain packet management policy; 2) there is no packet arrival and consequently no packet is updated. It is clear that under this queue model, the optimal packet management, under arbitrary scheduling policy, is to always update the most up-to-date packet, i.e., the packet that arrives the last; the resultant queue is equivalent to having a buffer size of one and storing only the latest arrival packet. Note that this packet management policy is not necessarily optimal with preemptive service model due to service interruption [32]. Without loss of optimality, we only consider the one-packet buffer packet management policy in the rest of the section.

The age of the packet in queue-$n$ (buffer size is one) is denoted by $A_n(t)$, $t = 1, 2, \ldots$; a sample path of which is shown in the left of Fig. 4. Upon a packet arrival, e.g., $a_1$ in Fig. 4, the age $A_n(t)$ drops to one (measured at the end of the time slot) based on the procedure in Fig. 2. When terminal-$n$ is scheduled at the time of $s_i$, the AoI at the BS is updated to the age of the packet at terminal $n$, i.e., $A_n(s_i)$. Note that we prescribe a generalized age of $A_n(s_i)$ that between each update and next packet arrival, e.g., between $s_1$ and $a_1$, $A_n(t)$

2327-4662 (c) 2018 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
equals the AoI of terminal \( n \) at the BS although there is no packet in the queue during the time. By doing this, we make \( A_n(t) \) evolve independently of \( h_{n,\pi}(t) \) while not affecting the AoI update procedure; this is crucial for the ASTA property to apply.

Based on the renewal process condition of AIR policies, and following the same arguments in, e.g., [32], the time-average AoI can be readily calculated by the sum of the geometric areas \( Q_{k,n} \) in Fig. 4:

\[
\bar{h}_{n,\pi}(N) = \lim_{T \to \infty} \frac{1}{T} \sum_{k=1}^{K} Q_{k,n} = \lim_{T \to \infty} \frac{1}{T} \sum_{k=1}^{K} \frac{1}{\lambda_n} = \frac{\mu_n}{m_n},
\]

where \( K \) denotes the number of scheduled times until time \( T \) (the terminal index is omitted for brevity). When \( K \) goes to infinity, \( K \) also goes to infinity for reasonable scheduling policies since otherwise it is obvious that the AoI of a terminal with finite scheduling times will go to infinity when \( T \) is large enough. The last equality is based on the elementary renewal theorem [33]. It then follows that

\[
\bar{h}_{n,\pi}(N) = \frac{1}{m_n} \mathbb{E} \left[ \frac{X_{n,\pi}^{(k)} A_n(s_k)}{2} + (X_{n,\pi}^{(k)} - 1) \frac{X_{n,\pi}^{(k)}}{2} \right] \)
\[
= \frac{1}{m_n} \left( \mathbb{E} \left[ X_{n,\pi}^{(k)} \right] \mathbb{E} \left[ A_n(s_k) \right] + \frac{1}{2} (v_n - m_n) \right) \)
\[
= \frac{\mathbb{E} \left[ A_n(s_k) \right]}{2} + \frac{v_n - m_n}{2m_n} \)
\[
\geq \mathbb{E} \left[ A_n(s_k) \right] + \frac{m_n - 1}{2},
\]

where the equality (a) is based on the arrival-independent condition of AIR policies, and the inequality (b) follows from \( v_n \geq m_n^2 \); note that the equality holds when the scheduling interval is a constant \( m_n \).

It is now clear that the main challenge is to calculate \( \mathbb{E} \left[ A_n(s_k) \right] \). First we have Lemma 2 (cf. proof in Appendix B) which shows that \( \{ A_n(t), t = 1, 2, \ldots \} \) is a Markov state process and its stationary distribution is given in (13).

**Lemma 2:** \( \{ A_n(t), t = 1, 2, \ldots \} \) is a Markov state process with the steady-state stationary distribution given as

\[
\mu_n(j) = \lambda_n (1 - \lambda_n)^{j-1},
\]

where \( \mu_n(j) \) denotes the probability that the steady state of terminal \( n \) is state \( j \) (age of packet at terminal-\( n \) equals \( j \)).

Then the challenge of calculating \( \mathbb{E} \left[ A_n(s_k) \right] \) is tackled by treating \( \mathbb{E} \left[ A_n(s_k) \right] \) as the average state value of a Markov state process by an independent outside observer. Armed with this, we invoke the ASTA property [31] which can be seen as a generalization of the well-known PASTA theorem to non-Poisson observers.

**Lemma 3:** [31, Theorem 3.14] Let \( U \) be a Markov state process and \( N \) be a counting process. Then ASTA holds for the pair \( (U, N) \) if \( U \) is left-continuous and the pair \( (U, N) \) is forward-pointwise independent, i.e., for all \( t > 0 \), \( U(t) \) and \( \{ N(t+s) - N(s) : s \geq 1 \} \) are independent.

Let \( U \) be \( \{ A_n(t), t = 1, 2, \ldots \} \) and \( N \) in Lemma 3 be the counting process of the number of scheduling times before time \( t \). Then based on the AIR policy conditions, \( U \) and \( N \) are independent. The continuous condition follows by design of update sequence described in Fig. 2. Therefore, we obtain

\[
\mathbb{E} \left[ A_n(s_k) \right] = \lim_{K \to \infty} \frac{1}{K} \sum_{k=1}^{K} A_n(s_k) = \mathbb{E} \left[ A_n(t) \right] = \frac{1}{\lambda_n}.
\]

In other words, the time-average of random sampling \( (s_k) \) of the Markov process \( \{ A_n(t), t = 1, 2, \ldots \} \) equals the steady-state average. Combining with (12), the time-average AoI is

\[
\bar{h}_{n,\pi}(N) = \frac{1}{N} \sum_{n=1}^{N} \bar{h}_{n,\pi}(N) \geq \frac{1}{N} \sum_{n=1}^{N} \left( \mathbb{E} \left[ A_n(s_k) \right] + \frac{m_n - 1}{2} \right) \)
\[
= \frac{1}{N} \sum_{n=1}^{N} \left( \frac{1}{\lambda_n} + \frac{m_n - 1}{2} \right).
\]

The update rate of all terminals equals one packet per time slot for WCNC policies; therefore, according to the elementary renewal theorem,

\[
\sum_{n=1}^{N} \frac{1}{m_n} = 1.
\]

It follows from the arithmetic-harmonic-mean inequality (equality holds when \( m_n = N, \forall n = 1, \ldots, N \)) and (15) that

\[
\bar{h}_{n,\pi}(N) \geq \frac{1}{N} \sum_{n=1}^{N} \frac{1}{\lambda_n} + \frac{N - 1}{2}.
\]

The equality holds in (17) under two conditions: 1) \( m_n = N, \forall n = 1, \ldots, N \); 2) the scheduling interval is a constant \( m_n \). These two conditions can be both satisfied with RR-ONE. For a sanity check, RR-ONE is indeed an AIR policy. With this, we can conclude the proof of Theorem 1.

**Remark 1:** Alert readers may be curious about the optimal policy in general. In particular, since a myopic policy (minimize the AoI in the next time slot) is optimal [9, Theorem 1] when we eliminate the randomness of arrival packets (\( \lambda = 1 \), i.e., the AoI is updated to one (or zero) every time, is it optimal in general? A counter example is given in Appendix A to show that the myopic policy is not optimal even with global state information (GSI), e.g., queue length, age of packets and arrival rates information, at least with finite horizon. Nevertheless, we conjecture that the myopic policy with GSI is close to optimal with infinite horizon, and it is adopted as a performance benchmark in Section VIII to investigate the value of GSI.

**IV. PROOF OF THEOREM 2: ASYMPTOTIC OPTIMALITY OF RR-ONE**

In IoT systems, one major challenge is to accommodate a large number of terminals while maintaining timely status updates. Hence, it is of central interest to consider the problem in the asymptotic regime. Towards this end, it will be shown in the following that RR-ONE, given its simple structure, is asymptotically optimal among all policies with arbitrary information and even non-causal packet arrival knowledge. First, we obtain two performance bounds and compare these with the achievable performance by RR-ONE; the conclusion
follows by showing that they have identical asymptotic scaling factors.

First, we introduce two lower bounds of the time-average AoI of any policies in Lemma 4 and Lemma 5 (cf. proof in Appendix B).

Lemma 4: The time-average AoI in (2) cannot be less than $\frac{N+1}{2}$ for any causal or non-causal policy, i.e.,

$$\bar{h}(\infty, N) \geq \frac{N+1}{2}, \quad \forall N = 1, 2, ..., \lambda_n \in [0, 1], n \in \{1, ..., N\}. \quad (18)$$

Lemma 5: The time-average AoI in (1) cannot be less than $\frac{1}{N} \sum_{n=1}^{N} \frac{1}{\lambda_n}$ for any causal or non-causal policy, i.e.,

$$\bar{h}(\infty, N) \geq \frac{1}{N} \sum_{n=1}^{N} \frac{1}{\lambda_n}, \quad \forall N = 1, 2, ..., \lambda_n \in [0, 1], \quad n \in \{1, ..., N\}. \quad (19)$$

It follows that the minimum time-average AoI, denoted by $\bar{h}_{\text{opt}}(\infty, N)$, cannot be less than either bound, i.e.,

$$\bar{h}_{\text{opt}}(\infty, N) \geq \max \left[ \frac{N+1}{2}, \frac{1}{N} \sum_{n=1}^{N} \frac{1}{\lambda_n} \right]. \quad (20)$$

After obtaining two lower bounds in Lemma 4 and 5, combining with the achievable AoI derived in (17), we can prove the asymptotic optimality of RR-ONE; the optimum scaling result follows immediately. Based on Lemma 4, Lemma 5 and Theorem 2, it follows that $\forall N, \lambda_1, ..., \lambda_N$,

$$\max \left[ \frac{N+1}{2}, \frac{1}{N} \sum_{n=1}^{N} \frac{1}{\lambda_n} \right] \leq \bar{h}_{\text{opt}}(\infty, N) \leq \frac{1}{N} \sum_{n=1}^{N} \frac{1}{\lambda_n} + \frac{N-1}{2}. \quad (21)$$

For any fixed $\lambda_n$, $n = 1, ..., N$, divide both sides of (21) by $N$, and let $N$ goes to infinity, we obtain

$$\lim_{N \to \infty} \max \left[ \frac{N+1}{2}, \frac{1}{N} \sum_{n=1}^{N} \frac{1}{\lambda_n} \right] = \frac{1}{2},$$

$$\lim_{N \to \infty} \frac{1}{N} \sum_{n=1}^{N} \frac{1}{\lambda_n} + \frac{N-1}{2} = \frac{1}{2}. \quad (22)$$

and therefore (7) follows. The scaling results for RR-ONE follows directly from (34). Similarly $\forall n$,

$$\lim_{x_n \to \infty} \max \left[ \frac{n+1}{2}, \frac{1}{N} \sum_{n=1}^{N} \frac{1}{x_n} \right] = \frac{1}{N},$$

and (8) follows, which concludes the proof of Theorem 2.

Remark 2: Theorem 2 shows that the optimum time-average AoI scales linearly with the number of terminals $N$ and expected inter-arrival time, i.e., $\frac{1}{\lambda_n}, \forall n$. The optimum scaling factors are also given. Moreover, we show that RR-ONE not only can achieve linear scaling, but also achieves the optimum scaling factors.

Remark 3: By setting $\lambda_n = 1, \forall n$ in Theorem 2 which is equivalent to a scenario wherein the age after every update is always one, it is immediately obvious that RR-ONE is optimal in this setting with arbitrary $N$.

Corollary 2: The time-average AoI achieved by a uniformly random scheduling policy with one-packet buffers (UN-ONE) is at least

$$\bar{h}_{\text{UN}}(\infty, N) \geq N. \quad (24)$$

Proof: Consider running the UN-ONE policy in system $A_0$ (Lemma 4). Obviously this gives us a lower bound on the UN-ONE performance. In this case, the system state is fully characterized by the AoI of each terminal, and the AoI for each terminal (omitting the terminal index for brevity) evolves as

$$p_{h,h+1} = \left( \frac{1}{N} \right), p_{h,1} = \frac{1}{N}, p_{h,i} = 0, \forall i \neq 1, h+1. \quad (25)$$

Note that although the AoI transitions of different terminals are not independent by observing, e.g., only one terminal can be scheduled at each time slot, the time-average AoI in (2) only concerns with the marginal distribution of AoI for each terminal. The AoI transition Markov chain of one terminal is shown in Fig. 5.

Therefore, the steady state AoI for each terminal is exponentially distributed with parameter $\frac{1}{N}$. The time-average AoI by running the UN-ONE policy in system $A_0$ is hence $N$. Therefore, (24) follows immediately.

Remark 4: Based on Corollary 2, UN-ONE, which in fact can be seen as a performance lower bound of pure CSMA scheme without considering the contention time overhead, has

Fig. 4. Age of the packet at terminal-$n$ assuming one packet buffer (left) and AoI at the BS (right), wherein $a_i$ denotes the arrival time of the $i$-th packet at terminal-$n$, $s_i$ denotes the $i$-th scheduling time of terminal-$n$, and $A_i(s_i)$ denotes the age of the packet updated at time $s_i$.
Since terminal queue, then the AoI is updated to certain value given by (26). If no packet is present at the successive RR-ONE scheduling and then is updated to a by noticing the fact that the AoI increases by $N$ during
the transition probability chain (the terminal index is omitted for brevity) is defined as

$$\Pr \{ A_n = a | \tau_n = \tau \} = \lambda_n (1 - \lambda_n)^{\tau - a}, \quad a = 0, ..., \tau - 1,$$

$$\Pr \{ A_n = h_n | \tau_n = \tau \} = (1 - \lambda_n)^{\tau}. \quad (26)$$

If the terminal-$n$ is scheduled in this time slot, then the AoI of terminal-$n$, i.e., $h_n$, is updated to the realization of $A_n$. Consider $N$ embedded Markov chains which describe the AoI transition for $N$ respective terminals between successive scheduling based on RR-ONE. The state $S$ of the $n$-th Markov chain (the terminal index is omitted for brevity) is defined as the AoI at the scheduled time slot. The transition probability matrix is therefore

$$p_{ss'} \triangleq \Pr \{ S_{k+1} = s' | S_k = s \} = \begin{cases} \lambda_n (1 - \lambda_n)^{s' - s}, & \text{if } s' \in \{1, ..., N\}; \\ (1 - \lambda_n)^N, & \text{if } s' = s + N; \\ 0, & \text{otherwise}, \end{cases} \quad (27)$$

by noticing the fact that the AoI increases by $N$ during successive RR-ONE scheduling and then is updated to a certain value given by (26). If no packet is present at the terminal queue, then the AoI is updated to $s' = s + N$. In the following, the resultant steady-state stationary distribution of the Markov chain, denoted by $\mu_n(s)$ where $s = 1, 2, ...$ denotes state value, is derived. It follows that

$$\mu_n(j) = \sum_{i \in \{1, 2, ...\}} \mu_n(i) p_{ij}. \quad (28)$$

Since

$$p_{ij} = \lambda_n (1 - \lambda_n)^{j - i - 1}, \quad \forall i \in \{1, 2, ...\}, j \in \{1, ..., N\}, \quad (29)$$

the steady-state stationary distribution for the first $N$ states is

$$\mu_n(j) = \lambda (1 - \lambda)^{j-1}, \quad \forall j \in \{1, ..., N\}. \quad (30)$$

For the $j$-th state with $j > N$, its previous state must be state $j - N$, and hence

$$\mu_n(j) = \mu_n(j - N) (1 - \lambda_n)^N = ... = \mu_n(l) (1 - \lambda_n)^{mN} = \lambda_n (1 - \lambda_n)^{j - 1}, \quad \forall j = mN + l, \quad (31)$$

where $m$ and $l$ are integers larger or equal to one. Therefore, it has been shown that the steady-state stationary distribution of the Markov chain is a geometric distribution with parameter $\lambda_n$.

After addressing the steady state of the embedded Markov chains, we are ready to derive the steady-state stationary distribution of the AoI evolution process $\{Y_n(t), t = 1, 2, ...\}$ at all time slots. Notice that it is a Markov renewal process with fixed renewal interval $N$ based on RR-ONE. Moreover, the state evolution between renewal (scheduling) is deterministic; the AoI increases by one every time slot. Therefore, the steady-state stationary distribution of $\{Y_n(t), t = 1, 2, ...\}$ is derived as

$$\mu_n(j) = \sum_{m=1}^{\min[j,N]} q_{n,m,j} = \sum_{m=1}^{\min[j,N]} \frac{1}{N} \lambda (1 - \lambda_n)^{j - m}, \quad (32)$$

where $q_{n,m,j}$ is the fraction of time that the AoI state transits to state $(j - m + 1)$ at the scheduled time slot and then reaches state-$j$ before the next scheduling. Therefore

$$j - (j - m + 1) = N - 1,$$

such that state-$j$ is within reach, and hence $m \leq N$; this, combining with $m \leq j$, explains the minimum operation in (32). The steady-state stationary distribution is hence given in (9) after some mathematical manipulations. The time-average AoI can be directly derived from this distribution, i.e.,

$$\bar{Y}(\infty, N) = \frac{1}{N} \sum_{n=1}^{N} \sum_{j=1}^{\infty} \mu_n(j) j = \frac{1}{N} \sum_{n=1}^{N} \frac{1}{\lambda_n} + N - 1 - \frac{N}{2}. \quad (34)$$

This coincides with (17).

### VI. DECENTRALIZED RR-ONE ALGORITHM

A fully decentralized RR-ONE-based scheduling algorithm (DRR) is proposed in this section. The proposed algorithm is described in Algorithm 1, where we assume dynamic terminal appearance. However, we assume at each time slot at most one event can happen: a terminal appears or disappears.

The essence of DRR can be summarized as follows. The system always runs a round-robin status update protocol by assigning each terminal a unique time slot ($\gamma_n$) to update and rotating among terminals. Without new terminal appearances or disappearances, the system runs collision-free and every time slot is utilized. The BS feeds an ACK back in this case. Terminals dynamically appear at or disappear from the system. With a new terminal appearance, it updates immediately by design, and causing a collision inevitably since previously all time slots are utilized. The BS then feeds back a common message indicating a collision, with also the information...
Algorithm 1: DRR

1. Initialization:
   2. (Terminals) Set $\gamma_n = n \mod N$, $w_n = N$.
   3. (BS) Set $N$ to the number of initial terminals and $W = 1$.
   4. (Newly appeared terminals) Set $\gamma_n = 0$, $w_n = 1$.

5. At each terminal:
   6. if Packet arrives then
      7. Replace the current queued packet with the new one.
   8. if $\gamma_n = 0$ then
      9. Transmit the queued packet (a blank packet if none).
     10. Set $\gamma_n \leftarrow w_n$.
    11. else
     12. Keep silent in this time slot.
     13. At the BS, after receiving updates:
    14. if Successful update or a blank update then
     15. Send an ACK.
    16. else if Receive nothing then
     17. Send a NACK.
    18. else
       (Collision) Set $N \leftarrow N + 1$. Send a packet indicating a collision, containing information of $N$ and $W$.
     19. Set $W \leftarrow (W - 1) \mod N$.
     20. After receiving feedback at each terminal:
    21. if Collision then
       22. if Terminal is newly appeared then
          23. $w_n \leftarrow N$, $\gamma_n \leftarrow W$, $\gamma_n \leftarrow \gamma_n - 1$.
       24. else
          25. $w_n \leftarrow N$, $\gamma_n \leftarrow N$, $\gamma_n \leftarrow \gamma_n - 1$.
       26. else if NACK then
          27. $w_n \leftarrow w_n - 1$, $\gamma_n \leftarrow \gamma_n - 1$.
       28. else
          29. ACK received, $\gamma_n \leftarrow \gamma_n - 1$.
     30. Return to Step 5.

VII. MAX-AOI-WEIGHT POLICY FOR SCHEDULING WITHOUT PACKET MANAGEMENT

Without packet management, the status packets are assumed to be queued with infinite buffers at terminals before transmissions and the service is based on FCFS discipline in this paper. Therefore, all generated packets are eventually transmitted and updated to the BS. This assumption is reasonable in scenarios where not only the latest status is important, but also the BS is interested in keeping all the status data, and their status evolution processes, for analytic purposes. Another reasonable scenario is that terminals cannot apply packet management due to hardware limitations.

Note that the ASTA property does not apply in this case, due to the fact that the age of the head-of-line packet at each terminal changes to the next-in-line packet’s age when the terminal is scheduled. Therefore, the independence property does not hold in Lemma 3; this presents a difficulty in analyzing the time-average AoI explicitly. Therefore, we resolve to the Lyapunov optimization technique [14] to derive AoI-MW by minimizing the drift of a Lyapunov function. The quadratic Lyapunov function is adopted, i.e.,

$$L(h_\pi(t)) \triangleq \frac{1}{N} \sum_{n=1}^{N} h_{n,\pi}(t)^2,$$  \hspace{1cm} (35)

where $h_{n}(t) \triangleq [h_{1,\pi}(t), \ldots, h_{N,\pi}(t)]^\top$, and the Lyapunov drift function is hence

$$\Delta(h_\pi(t)) \triangleq \mathbb{E}[L(h_\pi(t+1)) - L(h_\pi(t)) | h_\pi(t)].$$  \hspace{1cm} (36)

The AoI dynamics defined in (3) is rewritten as

$$h_{n,\pi}(t+1) = h_{n,\pi}(t) - U_{n,\pi}(t) g_{n,\pi}(t) + 1$$  \hspace{1cm} (37)

with $\sum_{n=1}^{N} U_{n,\pi}(t) \leq 1$, and $g_{n,\pi}(t)$ equals the age difference, or the inter-arrival time (at buffer of terminal-$n$), between the last-received status packet at destination for terminal-$n$ and the head-of-line packet in this time slot; as stated before, in case of there is no packet in the buffer of the scheduled terminal, $g_{n,\pi}(t) = 0$. It follows that

$$\mathbb{E}[h_{n,\pi}(t+1)^2 - h_{n,\pi}(t)^2 | h_{n,\pi}(t)] = -\mathbb{E}[U_{n,\pi}(t) h_{n,\pi}(t)] G_{n,\pi}(t) + 2h_{n,\pi}(t) + 1,$$  \hspace{1cm} (38)

where $G_{n,\pi}(t) \triangleq 2(h_{n,\pi}(t) + 1) g_{n,\pi}(t) - g_{n,\pi}(t)^2$. Then the drift can be written as

$$\Delta(h_\pi(t)) = -\frac{1}{N} \sum_{n=1}^{N} \mathbb{E}[U_{n,\pi}(t) | h_{n,\pi}(t)] G_{n,\pi}(t) + \frac{1}{N} \sum_{n=1}^{N} [2h_{n,\pi}(t) + 1].$$  \hspace{1cm} (39)

AoI-MW should minimize the Lyapunov drift at any time slot such that the Lyapunov function is minimized in the long run [14]. Observe that only the first term in (39) concerns with the scheduling action, and hence the AoI-MW policy is defined as at each time slot,

$$n_{AoI-MW} = \arg \max_{n \in \{1, \ldots, N\}} \{2(h_{n,\pi}(t) + 1) g_{n,\pi}(t) - g_{n,\pi}^2(t)\},$$  \hspace{1cm} (40)

about the current number of terminals (including the newly appeared), i.e., $N$, and the current spot of rotation, i.e., $W$. The new terminal will occupy the newly created time slot at the end of the rotation and the collided time slot still belongs to its original owner, and therefore no collision will happen between the two terminals in the next round. With a terminal (T0) disappearance, the BS receives nothing in its time slot. Note that a terminal that does not have any packet to update when it is scheduled would transmit a special blank packet such that the BS can distinguish between terminal disappearance and no packet to update. When T0 disappears, the BS then feeds back a NACK. Every terminal in the system then subtracts its record of the number of terminals by one and the next-in-line terminal (T1) will occupy T0’s spot in the next round.
The decentralized implementation of AoI-MW can be based on the queue-aware CSMA scheme [18], and replacing the queue weights by the maximization term in (40). Note that the Lyapunov drift framework can incorporate arbitrary linear terminal weights (denoted by \( w_n \)) and i.i.d. transmission error (error probability denoted by \( p_{e,n} \)) naturally by multiplying the maximization term in (40) by \( w_n (1 - p_{e,n}) \) [34].

**Remark 5 (Performance of AoI-MW):** The performance analysis for max-weight type policies usually follow the standard approach that proves its performance advantage over the optimal randomized policy [14] which can achieve throughput-optimality under mild conditions. However, in the AoI case the optimal randomized policy is found to be loosely connected to the optimal AoI scheduling [34], leading to loose AoI bounds that sometimes are even useless given the fact the performance of the derived Lyapunov-drift-based policy is very close to optimum revealed by simulations [34]. Therefore, we evaluate the AoI-MW performance based on simulations in Section VIII (Fig. 10).

**VIII. SIMULATION RESULTS**

In this section, computer simulation based experiments are conducted to evaluate the AoI performance of scheduling policies. The optimum performance is obtained numerically by formulating the problem as an Markov decision process (MDP) and solving it by relative value iterations for average cost function [35]. Note that, similar with most practical applications, the MDP based approach suffers from the curse of dimensionality and hence only small-scale problems can be solved thereby. Specifically, it is observed that the state space size grows exponentially with the number of terminals, and hence the scalability is significantly limited. Nevertheless, we obtain the minimum time-average AoI of a 2-terminal case and compare its performance with RR-ONE. An finite-state approximation is made for the MDP which originally has infinite state space; note that the AoI can grow to infinity. However, the optimality is intact by arguing that the optimal policy given by solving the finite state MDP does not allow the AoI grows to our prescribed AoI limit. The state space of the MDP is defined as

\[
(h_1, a_1) \times (h_2, a_2),
\]

where \( h_i \) denotes the AoI at the BS for terminal \( i \), \( a_i \) denotes the age of the packet at terminal \( i \) (assuming the terminal adopts the one-packet buffer packet management policy), and \( 1 \leq h_i, a_i \leq h_{\text{max}}, \ i = 1, 2 \), where \( h_{\text{max}} \) is the age limit. The transition probability matrix follows straightforwardly; it is omitted here for brevity, along with the relative value iteration procedure which is well known. The performance of RR-ONE is obtained by running RR-ONE for \( 10^5 \) time slots and calculating the time-average AoI. In addition, we also simulate the pure CSMA which schedules a terminal uniformly random at each time slot, and an age-greedy policy which chooses the terminal with the largest AoI. The age-greedy policy is found optimal without considering random packet arrivals [9].

It is observed from Fig. 6 that the performance gap between RR-ONE and the optimum given by numerically solving the MDP is larger with lower packet arrival rates; on the other hand, RR-ONE achieves the optimum when \( \lambda \) approaches one based on Remark 3. Since RR-ONE is proved optimal among AIR policies, the optimal scheduling policy with low packet arrival rates must be a non-AIR policy. Specifically, the following intuition explains this. Suppose that the probability of both terminals having arrival packets in the same time slot is negligible when arrival rates are sufficiently low; then the optimal policy is immediately obvious that it should schedule the terminal with a packet arrival at each time slot; note that this policy is not an AIR policy because the scheduling decision depends on packet arrivals which violates Condition 1 of AIR policy definition. The optimum performance in this case is also obvious; it should be the same with what is shown in Lemma 5, i.e., completely determined by the inter-arrival time of packets and this can be observed from Fig 6. In our recent work [36], a Whittle’s index [37] policy which accounts for the heterogeneity of arrival rates and thus with better performance is derived, whereas at the cost of losing the explicit AoI expressions and simple decentralized implementation. Nevertheless, it is noted that the performance...
Fig. 8. Performance comparisons with a large number of terminals; the arrival rates are uniformly randomly generated from $[0, 1]$.

Fig. 9. Performance comparisons with a large number of terminals and FCFS or LCFS service disciplines; the arrival rates are identical for all terminals, and $\lambda = 0.1/N$ (left), $\lambda = 0.1/N$ (right).

Fig. 10. Performance comparisons between queue-aware CSMA and AoI-MW for two terminals with different arrival rates.

gap is bounded (within 0.5 time slots) in this 2-terminal case by observing the RR-ONE performance and performance bound in Lemma 5. The arrival rates in Fig. 6 and 7 are no less than 0.3 due to the curse of dimensionality of MDP as mentioned before. When the arrival rates are too low, $h_{\max}$ can be so large that the MDP state space is beyond the memory size allowed by MATLAB.

Fig. 7 shows the performance with heterogeneous packet arrival rates for terminals; the arrival rate $\lambda_1$ is fixed to 0.5 packets per time slot and $\lambda_2$ varies from 0.3 to 1. The bounded gap is still within 0.5 time slots, despite the fact that RR-ONE disregards the heterogeneity of arrival rates completely. Nonetheless, it is observable that the gap between optimum and RR-ONE is larger compared with Fig. 6. Moreover, the gap between RR-ONE and the age-greedy policy is relatively larger when the arrival rates difference between terminals increases, showing that the age-greedy is more sensitive to arrival statistics heterogeneity.

We increase the number of terminals and enter the massive IoT regime in Fig. 8. The MDP-based optimal solution is computationally intractable in this regime and hence we adopt the myopic policy with GSI as an approximation of the optimum. The myopic policy with GSI leverages all the global information (though no future knowledge) to make a scheduling decision that minimizes the one-step expected AoI cost in the MDP formulation; by comparing it with RR-ONE helps us to understand how much GSI benefits RR-ONE. Based on Fig. 8, it is shown that the myopic policy with GSI outperforms RR-ONE only slightly, due to the reason that the packet management of using one-packet buffers eliminates most of the randomness of packet arrivals; most packets are dropped by packet management due to staleness and hence their randomness has no effect. The performance of pure CSMA is also shown; it has been proved in Corollary 2 that its linear scaling factor is (at least) 1 compared with 1/2 for RR-ONE; this can be observed in the figure.

In Fig. 9 and 10, policies without packet management are simulated. The arrival packets are queued at terminals and all arrived packets have to be delivered to the BS. We compare our proposed AoI-MW with pure CSMA (every terminal with equal transmission probability) and queue-aware CSMA [18] (terminals are prioritized based on their queue lengths). The AoI-MW scheme is implemented as follows. The weight of each terminal is given by (40); based on the queue-CSMA scheme proposed in [18], AoI-MW is implemented by replacing the weight of queue-length [18] with (40). The frame structure is shown here.

**Contest mini-slots**

In the contention mini-slots, each of which has a length of $\delta = \frac{1}{10}T_s$, where $T_s$ is the length of a data slot, all terminals participate in the $p$-CSMA random access wherein the transmission probability of each terminal is determined by the scheduling policies and the mapping functions derived in [18]. The terminals that are elected by the contention transmit in the data slot; note that collisions can also happen due to
multiple terminals elected at the same time, in which case
the transmissions fail. The length of one time slot is the
combination of contention mini-slots and one data slot.

Fig. 9 shows that when packet arrival rates of terminals are
equal, queue-aware CSMA and Aol-MW achieve similar
AoI performance, while pure CSMA performs much worse
since it is oblivious to the instantaneous age of packets in the
queues. Note that even the pure CSMA can achieve stability in
this case with identical arrival rates, however in terms of AoI,
the AoI awareness (queue-aware CSMA can be considered
as semi-AoI-aware since the queue length can approximate
AoI) is very important. Furthermore, we identify the advantage
of Aol-MW over queue-aware CSMA in scenarios with non-
identical arrival rates in Fig. 10; for ease of exposition, we
consider a case with two terminals. It is observed that when
the traffic load is high, i.e., \( \lambda_1 + \lambda_2 \rightarrow 1 \), and the arrival rates
difference between two terminals is large, the performance
gap between Aol-MW and queue-aware CSMA is evident.
This is because the queue length can no longer approximate
the AoI well in the regime. It is also observed that LCFS
performs better than pure CSMA in the non-asymptotic regime, especially when
\( \lambda N \) remains constant. Since the BS knows the current
ages of packets of all terminals and the arrival distributions,
impacts on the optimal time-average AoI. Future work should consider
further improvements.

In this section, we assume that the BS knows the current
ages of packets of all terminals and the arrival distributions,
i.e., \( \lambda_i, i \in \{1, \ldots, N\} \) to make a centralized scheduling
decision. In this case, it will be shown the myopic policy is
not optimal by constructing a counter example. Consider a
case with a time horizon \( T = 2 \), and the number of terminals
is \( N = 2 \). The arrival rates for terminal 1 and 2 are \( \lambda_1 = \delta \)
and \( \lambda_2 = 1 - \delta \), respectively, and \( 0 < \delta < 1 \). The initial
ages are set as \( h_i(0) \), \( i = 1, 2 \), and the initial age gains of
the packets in the queues are \( g_i(0), i = 1, 2 \). Assuming the
following conditions are met:

\[
\frac{g_2(0)}{2} < g_1(0) < g_2(0),
\]

\[
h_2(0) - g_2(0) > h_1(0),
\]

then the expected average age under the myopic policy, which
always schedules the terminal with the largest \( g_i(t) \) at time \( t \),
is shown here

\[
\frac{1}{4} (h_{1,\text{mos}}(1) + h_{1,\text{mos}}(2) + h_{2,\text{mos}}(1) + h_{2,\text{mos}}(2))
= C - \frac{1}{4} (g_2(0) + \delta(1 - \delta)(h_2(0) + 1)
+ \delta^2(g_1(0) + h_1(0) + 1 + \delta)(g_1(0) + g_2(0))
+ (1 - \delta)^2(h_2(0) + 1))
= C - \frac{1}{4} (g_2(0) + h_2(0) + 1 + \delta(g_1(0) - g_2(0))
+ \delta^2(h_1(0) + 1 - g_1(0))),
\]

and the expected average age under the optimal policy, which
is not hard to figure out in this simple case, is

\[
\frac{1}{4} (h_{1,\text{mos}}(1) + h_{1,\text{mos}}(2) + h_{2,\text{mos}}(1) + h_{2,\text{mos}}(2))
= C - \frac{1}{4} (g_1(0) + \delta(1 - \delta)(g_1(0) + h_2(0) + 1)
+ \delta^2(g_1(0) + g_2(0)) + (1 - \delta)^2(g_1(0) + g_2(0))
+ (1 - \delta)^2(g_1(0) + h_2(0) + 1))
= C - \frac{1}{4} (2g_1(0) + h_2(0) + 1 - \delta(h_2(0) + 1 - g_2(0))),
\]

where

\[
C = \frac{1}{2} (h_1(0) + h_2(0)) + 1.5.
\]

By setting the parameter \( \delta \) sufficiently small, we can see that
the average age of the myopic policy is strictly smaller than
the optimal policy which schedules terminal 1 at the first time
slot although the current myopic choice is terminal 2.

**APPENDIX A**

**MYOPIC POLICY WITH GSI IS NOT OPTIMAL WITH FINITE HORIZON**

In this section, we assume that the BS knows the current
ages of packets of all terminals and the arrival distributions,
i.e., \( \lambda_i, i \in \{1, \ldots, N\} \) to make a centralized scheduling
decision. In this case, it will be shown the myopic policy is
not optimal by constructing a counter example. Consider a
case with a time horizon \( T = 2 \), and the number of terminals
is \( N = 2 \). The arrival rates for terminal 1 and 2 are \( \lambda_1 = \delta \)
and \( \lambda_2 = 1 - \delta \), respectively, and \( 0 < \delta < 1 \). The initial
ages are set as \( h_i(0) \), \( i = 1, 2 \), and the initial age gains of
the packets in the queues are \( g_i(0), i = 1, 2 \). Assuming the
following conditions are met:

\[
\frac{g_2(0)}{2} < g_1(0) < g_2(0),
\]

\[
h_2(0) - g_2(0) > h_1(0),
\]

then the expected average age under the myopic policy, which
always schedules the terminal with the largest \( g_i(t) \) at time \( t \),
is shown here

\[
\frac{1}{4} (h_{1,\text{mos}}(1) + h_{1,\text{mos}}(2) + h_{2,\text{mos}}(1) + h_{2,\text{mos}}(2))
= C - \frac{1}{4} (g_2(0) + \delta(1 - \delta)(h_2(0) + 1)
+ \delta^2(g_1(0) + h_1(0) + 1 + \delta)(g_1(0) + g_2(0))
+ (1 - \delta)^2(h_2(0) + 1))
= C - \frac{1}{4} (g_2(0) + h_2(0) + 1 + \delta(g_1(0) - g_2(0))
+ \delta^2(h_1(0) + 1 - g_1(0))),
\]

and the expected average age under the optimal policy, which
is not hard to figure out in this simple case, is

\[
\frac{1}{4} (h_{1,\text{mos}}(1) + h_{1,\text{mos}}(2) + h_{2,\text{mos}}(1) + h_{2,\text{mos}}(2))
= C - \frac{1}{4} (g_1(0) + \delta(1 - \delta)(g_1(0) + h_2(0) + 1)
+ \delta^2(g_1(0) + g_2(0)) + (1 - \delta)^2(g_1(0) + g_2(0))
+ (1 - \delta)^2(g_1(0) + h_2(0) + 1))
= C - \frac{1}{4} (2g_1(0) + h_2(0) + 1 - \delta(h_2(0) + 1 - g_2(0))),
\]

where

\[
C = \frac{1}{2} (h_1(0) + h_2(0)) + 1.5.
\]

By setting the parameter \( \delta \) sufficiently small, we can see that
the average age of the myopic policy is strictly smaller than
the optimal policy which schedules terminal 1 at the first time
slot although the current myopic choice is terminal 2.

**APPENDIX B**

**PROOF OF LEMMAS**

**Proof of Lemma 1:** It is straightforward that the performance
of a non-WCNC policy is stochastically dominated by
a WCNC policy which is identical to the non-WCNC policy,
except that the WCNC policy schedules an arbitrary terminal
(resp. schedules one of the terminals) when the non-WCNC policy is idle (resp. schedules multiple terminals resulting in collisions).

**Proof of Lemma 2:** At each time slot, the probability of a packet arrival is λ and therein the age decreases to one; otherwise, with probability 1 − λ, the age increases by one. The Markovian property holds obviously. The steady-state stationary distribution is hence a geometric distribution with parameter λ.

**Proof of Lemma 4:** Consider a system A0 wherein the A0 after each update is fixed to one, instead of determined by the random arrival packets. Then for any scheduling policy π, we have
\[ \tilde{h}_{π}^{(T,N)} \geq \tilde{h}_{π,A0}^{(T,N)}, \]  
where \( \tilde{h}_{π,A0}^{(T,N)} \) denotes the time-average AoI of policy π in system A0. The proof for (47) is simply based on stochastic dominance, and currently omitted for brevity. By [9, Theorem 1], the optimal policy in A0 is the age-greedy policy by noticing that no asymmetric transmission failure or weights are considered in this paper. Therefore it is straightforward to derive the optimum time-average AoI in A0 since no randomness exists. The optimum AoI in A0 is therefore
\[ \tilde{h}_{\text{opt},A0}^{(\infty,N)} = \frac{N + 1}{2}, \]  
(48)

It is sufficient to consider the optimal scheduling scheme in A0 to obtain a AoI lower bound since all scheduling schemes perform better in A0 compared with a system with random packet arrivals. The conclusion follows immediately.

The proof also applies to non-causal policies, by noticing that the future knowledge of packet arrival times would not benefit the policy performance in A0 since the packet arrivals in A0 are deterministic, i.e., \( \lambda_i = 1, \forall i \).

**Proof of Lemma 5:** To prove this result, we consider a collision-free system A1 wherein the uplink transmissions can be multiplexed, i.e., an arbitrary number of terminals can update successfully in the same time slot. It is obvious that, similar with Lemma 4, for any policy π,
\[ \tilde{h}_{π}^{(T,N)} \geq \tilde{h}_{π,A1}^{(T,N)}. \]  
(49)

It is also obvious that the time-average AoI in system A1 is the time-average packet inter-arrival time at terminal queues. Therefore, the conclusion follows immediately.

The proof also applies to non-causal policies, by noticing that the future knowledge of packet arrival times would not benefit the policy performance in A1 since all the terminals are scheduled in A1, regardless of packet arrival times.

**REFERENCES**


Zhiyuan Jiang received his B.E., Ph.D. degrees from the Electronic Engineering Department of Tsinghua University in 2010, 2015, respectively. He is currently an Associate Professor with the School of Communication and Information Engineering, Shanghai University. He visited University of Southern California during 2013-2014 and 2017-2018. He worked as an experienced researcher and a wireless signal processing scientist for Ericsson and Intel Labs in 2015-2016 and 2018, respectively. His main research interests include sequential decision making in wireless networks and the design, implementation of multi-antenna systems.

Bhaskar Krishnamachari is Ming Hsieh Faculty Fellow and Professor of Electrical Engineering and Computer Science at the Viterbi School of Engineering at the University of Southern California. He received his B.E. in Electrical Engineering from the Cooper Union in New York City in 1998, and his M.S. and Ph.D. in Electrical Engineering from Cornell University in 1999 and 2002, respectively. He is Director of the USC Viterbi Center for Cyber-Physical Systems and the Internet of Things (CCS). His research interests are in the design, analysis, implementation and empirical evaluation of algorithms and protocols for next-generation wireless networks, including for the Internet of Things. His work has received several best paper awards including at IPSN and Mobicip. He has received the NSF CAREER Award and the ASEE Terman Award. He is the author of a textbook titled Networking Wireless Sensors published by Cambridge University Press, and serves as an editor for the IEEE/ACM Transactions on Networking.

Xi Zheng received her B.S. degree in Physics from Tsinghua University, Beijing, China, in 2014. Currently, she is a Ph.D. student in Electronic Engineering at Tsinghua University, Beijing, China. Her research interests include resource management in wireless networks and green wireless communications.

Zhisheng Niu graduated from Northern Jiaotong University (currently Beijing Jiaotong University), China, in 1985, and got his M.E. and D.E. degrees from Toyohashi University of Technology, Japan, in 1989 and 1992, respectively. During 1992-94, he worked for Fujitsu Laboratories Ltd., Japan, and in 1994 joined with Tsinghua University, Beijing, China, where he is now a professor at the Department of Electronic Engineering. He was a Visiting Researcher at National Institute of Information and Communication Technologies (NICT), Japan (10/1995 – 02/1996), Hitachi Central Research Laboratory, Japan (02/1997 – 02/1998), Saga University, Japan (01/2001 – 02/2001), Polytechnic University of New York, USA (01/2002 – 02/2002), University of Hamburg, Germany (09/2014 – 10/2014), and University of Southern California, USA (11/2014-12/2014). His major research interests include queueing theory, traffic engineering, mobile Internet, radio resource management of wireless networks, and green communication and networks.


Dr. Niu has published 100+ journal and 200+ conference papers in IEEE and IEICE publications and co-received the Best Paper Awards from the 13th, 15th and 19th Asia-Pacific Conference on Communication (APCC) in 2007, 2009, and 2013, respectively, International Conference on Wireless Communications and Signal Processing (WCSP’13), and the Best Student Paper Award from the 25th International Teletraffic Congress (ITC25). He received the Outstanding Young Researcher Award from Natural Science Foundation of China in 2009 and the Best Paper Award from IEEE Communication Society Asia-Pacific Board in 2013. He was also selected as a distinguished lecturer of IEEE Communication Society (2012-15) as well as IEEE Vehicular Technologies Society (2014-16). He is a fellow of both IEEE and IEICE.