Age-based Scheduling: Improving Data Freshness for Wireless Real-Time Traffic

Ning Lu
Department of CS
Thompson Rivers University
Kamloops, BC, Canada
nlu@tru.ca

Bo Ji
Department of CIS
Temple University
Philadelphia, PA, USA
boji@temple.edu

Bin Li
Department of ECBE
University of Rhode Island
Kingston, Rhode Island, USA
binli@uri.edu

ABSTRACT

We consider the problem of scheduling real-time traffic with hard deadlines in a wireless ad hoc network. In contrast to existing real-time scheduling policies that merely ensure a minimal timely throughput, our design goal is to provide guarantees on both the timely throughput and data freshness in terms of *age-of-information* (AoI), which is a newly proposed metric that captures the "age" of the most recently received information at the destination of a link. The main idea is to introduce the AoI as one of the driving factors in making scheduling decisions. We first prove that the proposed scheduling policy is feasibility-optimal, i.e., satisfying the per-traffic timely throughput requirement. Then, we derive an upper bound on a considered data freshness metric in terms of AoI, demonstrating that the network-wide data freshness is guaranteed and can be tuned under the proposed scheduling policy. Interestingly, we reveal that the improvement of network data freshness is at the cost of slowing down the convergence of the timely throughput. Extensive simulations are performed to validate our analytical results. Both analytical and simulation results confirm the capability of the proposed scheduling policy to improve the data freshness without sacrificing the feasibility optimality.

CCS CONCEPTS

• **Networks → Ad hoc networks**; **Network performance analysis**;

KEYWORDS

Data freshness, wireless scheduling, age of information, real-time traffic, ad hoc networks

ACM Reference Format:
Ning Lu, Bo Ji, and Bin Li. 2018. Age-based Scheduling: Improving Data Freshness for Wireless Real-Time Traffic. In *Mobihoc '18: The Eighteenth ACM International Symposium on Mobile Ad Hoc Networking and Computing, June 26–29, 2018, Los Angeles, CA, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3209582.3209602

Mobihoc '18, June 26–29, 2018, Los Angeles, CA, USA

ACM ISBN 978-1-4503-5770-8/18/06…$15.00
https://doi.org/10.1145/3209582.3209602

1 INTRODUCTION

Scheduling wireless network traffic with hard packet deadlines plays an important role in providing quality-of-service (QoS) guarantees for mission-critical systems, such as wireless sensor monitoring, vehicle platooning, industrial automation, and other cyber-physical system. Consider the scenario where multiple sensors measure variables of a plant, and all measurements are required to send to a central controller through a wireless network for monitoring and control purposes. To keep the usefulness of the measurements, each data packet in the network has to be sent in a real-time manner[1]; otherwise, it would become stale and thus useless. Too many useless measurements might not be tolerable and could affect the stability and control performance of the system. Real-time scheduling algorithms aim at scheduling deadline-constrained packets in wireless networks to guarantee a minimum portion of packets to be delivered on time (i.e., timely throughput) for each traffic flow. It is worth noting that imposing a hard deadline on each packet's reception guarantees that the received packet at the time of reception is useful; while maintaining a certain timely throughput assures that a sufficient number of useful packets are received during a fixed period of time on average. The design of real-time scheduling algorithms has attracted many research attentions recently (e.g., [4, 5, 7, 11]). Although providing QoS guarantees such that the destination receives timely packets at a desired rate is crucial, it might not be sufficient to maintain information freshness for real-time systems (see [3]). For example, in the above scenario, the most recent measurement of a variable could become out-of-date to the controller if it waits too long before the next update, which can lead to disturbance in monitoring and control. Hence, measurements from sensors are expected to update timely at the controller's side, indicating that the data freshness at the information sink also has an important role to play.

Recently, the "age-of-information" (AoI) is proposed to capture the freshness of information updates in a system, which has received great attention because of rapid deployment of real-time applications [14]. By definition (e.g., given in [8]), AoI is the amount of time elapsed since the most recent update (at the destination) was generated (at the source). It naturally characterizes the data freshness from the receiver's perspective. Motivated by the AoI research and the importance of information freshness at the receiver for real-time applications (e.g., maintaining fresh measurements at the controller in wireless monitoring), we are particularly interested in designing real-time scheduling algorithms that guarantee

[1] There exist a number of factors that prevent the real-time delivery of packets, such as queueing delay due to channel contention among transmitters, and packet loss due to channel fluctuation.

a minimum portion of packets delivered on time as well as improve the AoI of real-time traffic. To the best of our knowledge, none of existing works has taken into consideration of providing both timely throughput and data freshness (in terms of AoI) guarantees for deadline-constrained traffic. The current focus of AoI research is on AoI analysis and optimization under different network settings (e.g., [2, 6, 8, 14]). In [6], Kadota *et al.* proposed a greedy policy (AoI-Greedy) for scheduling deadline-constrained traffic in a symmetric wireless network. However, this work only focuses on the AoI minimization without providing any guarantees on timely throughput, and is specifically suitable for broadcast (i.e., fully connected) networks. It is worth mentioning that our work is also related to improving the short-term performance of real-time traffic (i.e., how often the packets are delivered to the receiver). Quantifying and optimizing short-term performance of real-time traffic flows remains open due to the fact that the analysis of transient behavior (short-term) of a system is more difficult than the analysis of asymptotic behavior (long-term). As an effort to this end, in [3], Hou proposed to introduce a penalty on each flow if its short-term performance is below some specified requirement, and employ Brownian motion approximation to optimize the overall penalty of the system. We consider our work to be an effort towards that end too, where the AoI can be considered as a metric that captures the short-term performance.

In this paper, we consider the problem of scheduling deadline-constrained real-time traffic in an *ad hoc* wireless network with guarantees on both timely throughput and data freshness in terms of AoI at the receivers. Particularly, we adopt a frame-based model[2]: the network nodes communicate in frames, each of which consists of a fixed number of consecutive time slots of equal duration; for each link, packets arrive at the beginning of each frame and need to be transmitted by the end of the frame; otherwise the packets will be dropped. To provide the guarantee on timely throughput, we use a *virtual-queue* technique in designing the scheduling algorithm; while to improve the performance of data freshness, we introduce *AoI* as one of the driving factors in making scheduling decisions. We provide an example in Section 2 to illustrate how the AoI performance can be improved through making appropriate scheduling decisions on packet delivery.

The contributions of this work can be summarized as follows:

- We propose a real-time scheduling policy that provides guarantees on both the timely throughput and the data freshness for deadline-constrained traffic in ad hoc networks, by introducing the AoI as one of the driving factors in making scheduling decisions.
- We prove that the proposed scheduling policy is feasibility-optimal. We also derive an upper bound on the network data freshness defined in Section 2, which demonstrates that the data freshness at the receivers is guaranteed in the network. Particularly, we reveal that the improvement of data freshness is at the cost of slowing down the convergence of the timely throughput. We show that trading the system performance (i.e., convergence speed of the timely throughput) for data freshness can be done by tuning a control parameter. The introduction of this control parameter makes the traditional virtual-queue-length-based policies (e.g., [4, 5, 7, 11]) and the AoI-Greedy policy special cases of our proposed policy.

The remainder of the paper is organized as follows. Section 2 introduces the system model. Section 3 presents the scheduling design and shows that the proposed policy is feasibility-optimal. We also analyze the AoI performance and investigate the trade-off between the data freshness and the convergence speed of the timely throughput in Section 3. Simulation results are given in Section 4. We discuss the distributed implementation of the proposed scheduling policy in Section 5. Section 6 provides concluding remarks.

2 SYSTEM MODEL

We consider scheduling real-time traffic flows in an ad hoc wireless network with L links. Assume that time is slotted and each consecutive T time slots are grouped into one frame. Each link is associated with one real-time flow, and we assume that packets arrive at each link only at the beginning of each frame, and each packet has to be delivered by the end of the frame; otherwise, it will be dropped. Let $A_l[kT]$ denote the number of packet arrivals at link $l \in \{1, 2, \cdots, L\}$ in frame k, where $k = 0, 1, 2, \cdots$, which are independently distributed over links and identically distributed over time with mean λ_l, and $A_l[kT] \leq A_{\max}$ for some $A_{\max} < \infty$. Note that we also use $\mathbf{A}[kT] \triangleq (A_l[kT])_{l=1}^{L}$ to denote the arrival vector in frame k. The timely throughput requirement on each flow is that each link guarantees a minimum timely throughput of $\lambda_l(1-\gamma_l)$, where $\gamma_l \in (0, 1)$ is the maximum allowable packet dropping rate due to deadline expiry. For example, on average at most 20% of packets can be dropped or a timely throughput is at least $0.8 \times \lambda_l$ at link l when $\gamma_l = 0.2$.

We consider that L links share one common communication channel, and each link experiences an independent block fading, where the channel state is constant during a frame and can vary over frames. We use a random variable $C_l[kT]$ to capture the channel state information (CSI) of link l in frame k, which is the maximum number of packets that can be delivered in a time slot if link l is scheduled. We assume that $\mathbf{C}[kT] \triangleq (C_l[kT])_{l=1}^{L}$ are independently and identically distributed over time with $C_l[kT] \leq C_{\max}, \forall l, k$, for some $C_{\max} < \infty$. We consider the case where the CSI is known to the scheduler via channel probing at the beginning of each frame.

Since nearby links cannot be scheduled simultaneously due to interference, scheduling is required to determine a subset of links that could transmit simultaneously without interfering with each other in each time slot, called *feasible schedule* denoted by $\mathbf{S}[t] \triangleq (S_l[t])_{l=1}^{L}$, where $S_l[t] = 1$ if link l is scheduled in slot t and $S_l[t] = 0$, otherwise. We use $\mathcal{S}$ to denote the set of all feasible schedules.

A key technique in the design is the use of virtual queues [12]. Assume that each link l maintains a virtual queue to keep track of the number of dropped packets due to deadline expiry. Let $V_l[kT]$ denote the virtual queue length of link l at the beginning of frame k. The evolution of virtual queue l is given by

$$V_l[(k+1)T] = \max\left\{V_l[kT] + I_l[kT] - D_l[kT], 0\right\}, \qquad (1)$$

[2]The problem of real-time scheduling under general interference, channel, packet arrival, and deadline models could be extremely difficult. A more modest goal in research community is to attack this problem under the frame-based model, which is first considered in [4].

where $I_l[kT] \geq 0$ is the increase of the virtual queue and is equal to the number of dropped packets in frame k, i.e.,

$$I_l[kT] \triangleq A_l[kT] - \min\left\{\sum_{t=kT}^{(k+1)T-1} C_l[kT]S_l[t], A_l[kT]\right\}; \quad (2)$$

and the decrease of the virtual queue is $D_l[kT]$ with mean $\gamma_l\lambda_l$, and $D_l[kT] < D_{\max}$ for some $D_{\max} < \infty$. According to [9], if there is a scheduling policy that can stabilize all virtual queues, there always exist non-negative numbers $\alpha(\mathbf{a}, \mathbf{c}; \mathbf{s}_0, \mathbf{s}_1, \ldots, \mathbf{s}_{T-1})$ such that

$$\sum_{\mathbf{s}_0, \mathbf{s}_1, \ldots, \mathbf{s}_{T-1} \in \mathcal{S}} \alpha(\mathbf{a}, \mathbf{c}; \mathbf{s}_0, \mathbf{s}_1, \ldots, \mathbf{s}_{T-1}) = 1, \forall \mathbf{a}, \mathbf{c}, \quad (3)$$

$$\lambda_l(1-\gamma_l) \leq \sum_{\mathbf{a}} \mathrm{P}_{\mathbf{A}}(\mathbf{a}) \sum_{\mathbf{c}} \mathrm{P}_{\mathbf{C}}(\mathbf{c}) \sum_{\mathbf{s}_0, \mathbf{s}_1, \ldots, \mathbf{s}_{T-1} \in \mathcal{S}} \alpha(\mathbf{a}, \mathbf{c}; \mathbf{s}_0, \mathbf{s}_1, \ldots, \mathbf{s}_{T-1}) \min\left\{\sum_{\tau=1}^{T-1} c_l s_{\tau,l}, a_l\right\}, \forall l. \quad (4)$$

where $\mathbf{s}_\tau \triangleq (s_{\tau,l})_{l=1}^{L}$, $\tau \in \{0, 1, \ldots, T-1\}$, $\mathrm{P}_{\mathbf{A}}(\mathbf{a}) = \Pr\{\mathbf{A}[kT] = \mathbf{a}\}$ and $\mathrm{P}_{\mathbf{C}}(\mathbf{c}) = \Pr\{\mathbf{C}[kT] = \mathbf{c}\}$. The inequality (4) states that the average service provided to link l during a frame (the right hand side of (4)) should be no less than the average amount of traffic that needs to be delivered (the left hand side of (4)), in order to meet the timely throughput requirement of real-time traffic.

A scheduling algorithm is said to be *feasibility-optimal* if, for any given maximum allowable dropping rate vector $\boldsymbol{\gamma} = (\gamma_l)_{l=1}^{L}$ and channel state $\mathbf{C}$, for any arrival process that lies strictly within the *maximal satisfiable region* $\Lambda(\boldsymbol{\gamma}, \mathbf{C})$, it stabilizes all virtual queues in the sense that all virtual queue processes are positive recurrent. The maximal satisfiable region is defined as follows:

$$\Lambda(\boldsymbol{\gamma}, \mathbf{C}) = \big\{\boldsymbol{\lambda} : \exists \alpha(\mathbf{a}, \mathbf{c}; \mathbf{s}_0, \mathbf{s}_1, \ldots, \mathbf{s}_{T-1}) \geq 0, \text{ such that (3) and (4) hold}\big\}. \quad (5)$$

In addition to providing guarantees on timely throughput, we are also interested in improving data freshness of the network. To this end, we adopt the AoI in designing the real-time scheduling policy. We denote $R_l[kT]$ the AoI of link l up to frame k, i.e, the number of frames elapsed up to frame k since the most recent successful packet delivery on link l. The smaller $R_l[kT]$, the better data freshness. Note that the use of frame as the unit of AoI can be seen in related work [6]. This is also because we assume that the arrivals only happen at the beginning of each frame. To capture the dynamics of $R_l[kT]$, we introduce $\mathcal{H}_l[kT]$ to denote the event that at least one packet is delivered at link l in frame k, i.e.,

$$\mathcal{H}_l[kT] \triangleq \left\{\min\left\{\sum_{t=kT}^{(k+1)T-1} C_l[kT]S_l[t], A_l[kT]\right\} > 0\right\}.$$

In addition, let $\mathbb{I}_{\mathcal{H}_l[kT]}$ denote the indicator variable such that $\mathbb{I}_{\mathcal{H}_l[kT]} = 1$ if event $\mathcal{H}_l[kT]$ happens, and $\mathbb{I}_{\mathcal{H}_l[kT]} = 0$ otherwise. Thus, the evolution of AoI of link l can be described as follows.

$$R_l[(k+1)T] = \begin{cases} 1 & \text{if } \mathbb{I}_{\mathcal{H}_l[kT]} = 1; \\ R_l[kT] + 1 & \text{if } \mathbb{I}_{\mathcal{H}_l[kT]} = 0. \end{cases} \quad (6)$$

As we can see, $R_l[kT]$ grows linearly until link l has at least one packet delivery in a frame and then it drops to one at the beginning of the next frame.

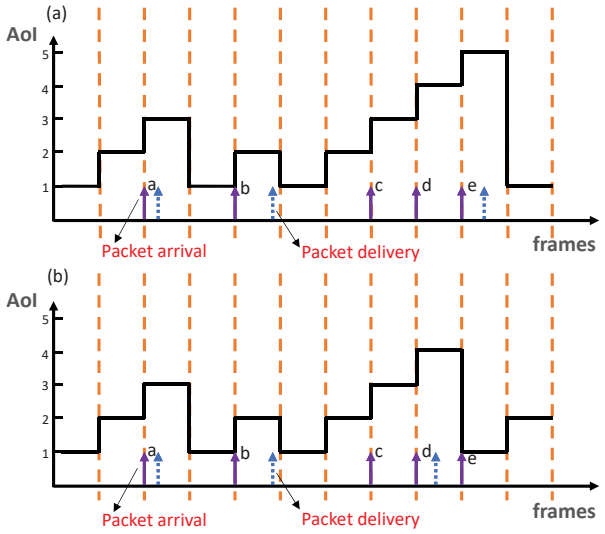


Figure 1: Example of AoI dynamics of a link

We provide an example in Fig. 1 to show how to improve AoI performance of a link by appropriately scheduling packet delivery. As shown in Fig. 1(a), the AoI of a link updates at the beginning of every frame. The delivery of packets a, b, and e result in the decrease of AoI. If we only focus on the time window shown in the figure (11 frames), Fig. 1(a) shows an average delivery ratio of 60% (3 packet deliveries over 5 packet arrivals) with an average AoI of 2.27 frames. While in Fig. 1(b), with the same packet arrival pattern and delivery ratio, the schedule of delivery of packet d instead of e leads to a drop of average AoI to 2 frames, indicating a better performance on data freshness.

In this work, we are interested in designing feasibility-optimal scheduling algorithms which can also improve the network data freshness, which is measured as the total weighted sum of the average steady-state AoI over all links in the network, i.e., $\sum_{l=1}^{L} \omega_l \mathbb{E}[\overline{R}_l]$, where $\omega_l \geq 0$ is a weighting parameter associated with link l and gives the preference of link l towards the information freshness, and $\overline{R}_l$ denotes the steady-state AoI of link l under some stabilizing scheduling policy. Next, we will develop the scheduling policy that not only uses virtual-queue lengths, but also considers the AoI in making scheduling decisions.

3 AGE-BASED REAL-TIME SCHEDULING

In this section, we first present the proposed scheduling policy and then show that it is feasibility-optimal. After that, we investigate the performance of data freshness under our policy. At last, we examine the trade-off between the data freshness and the convergence speed of the timely throughput.

3.1 Algorithm Description

The proposed Age-based real-time (AoI-RT) scheduling algorithm is given in Algorithm 1.

The scheduling decision is made on a per-frame basis. The idea is that, at the beginning of frame k, a maximum weighted schedule

Algorithm 1: AoI-RT Algorithm

In each frame k, the AoI-RT algorithm selects a schedule $\{\mathbf{S}^*[t]\}_{t=kT}^{(k+1)T-1}$ such that

$$\{\mathbf{S}^*[t]\}_{t=kT}^{(k+1)T-1} \in \underset{\substack{\mathbf{S}[t]\in\mathcal{S},\forall t=kT,\\ \ldots,(k+1)T-1}}{\arg\max} \sum_{l=1}^{L}((1-\zeta)V_l[kT]+\zeta\beta_l R_l[kT]) \times \min\left\{\sum_{t=kT}^{(k+1)T-1} C_l[kT]S_l[t], A_l[kT]\right\}.$$

where $0 \le \zeta \le 1$ and $\beta_l \ge 0, \forall l$, are some control parameters.

$\{\mathbf{S}^*[t]\}_{t=kT}^{(k+1)T-1}$ is selected among all possible schedules, by treating

$$\sum_{l=1}^{L}((1-\zeta)V_l[kT]+\zeta\beta_l R_l[kT]) \min\left\{\sum_{t=kT}^{(k+1)T-1} C_l[kT]S_l[t], A_l[kT]\right\}$$

in Algorithm 1 as the "weight" of a schedule $\{\mathbf{S}[t]\}_{t=kT}^{(k+1)T-1}$ over the frame k. The weight can be considered as a total sum of "link weight" over all links. For each link l, the link weight under a certain schedule is the production of $((1-\zeta)V_l[kT]+\zeta\beta_l R_l[kT])$ and the service scheduled to link l. Here, the virtual-queue length $V_l[kT]$ and the weighted AoI $\beta_l R_l[kT]$ are the two driving factors in making scheduling decisions. Note that the AoI-RT algorithm is similar to the traditional MaxWeight algorithm. The difference lies in that the weight now is the linear combination of the virtual-queue length and the weighted AoI parameterized by ζ.

Note that ζ is a parameter to control the data freshness performance of the algorithm. In particular, with $\zeta = 0$, the AoI-RT algorithm coincides with the pure virtual-queue-length-based (VQL-based) scheduling algorithms (e.g., [4, 5, 7, 11]), which is feasibility-optimal but provides no guarantees on data freshness in terms of AoI; while with $\zeta = 1$, the AoI-RT algorithm coincides with the greedy algorithm proposed in [6] (AoI-Greedy) that optimizes AoI but provides no guarantees on timely throughput. We will show that our policy with $0 < \zeta < 1$ can provide guarantees on both timely-throughput and AoI. It is worth noting that β_l is a parameter related to the link preference towards the data freshness.

While our age-based policy is similar to the regular scheduler proposed in [10], where the weight of a link is a combination of its congestion level (queue-length) and its *time-since-last-service* (TSLS), there are three major differences: 1) the regular scheduler strives to improve the service regularity from the sender's perspective (i.e., how often a link is scheduled) while our proposed AoI-RT algorithm improves the data freshness from the receiver's point of view (i.e., how often the packets are delivered to the receiver). Hence, guaranteeing TSLS is different from guaranteeing AoI; 2) the regular scheduler may serve a link even if it is empty while the AoI-RT algorithm always serves non-empty links, and since the regular scheduler may serve a link even if it is empty, it may lead to the waste of service; and 3) the regular scheduler is not designed for real-time traffic and therefore does not provide guarantees on timely throughput.

3.2 Feasibility Optimality

In this subsection, we will show that the proposed AoI-RT algorithm is feasibility-optimal, i.e., providing guarantees on timely throughput of real-time traffic.

Proposition 3.1. *The AoI-RT algorithm with any parameter $0 \le \zeta < 1$, is feasibility-optimal, i.e., for any arrival process that lies strictly within the maximal satisfiable region $\Lambda(\boldsymbol{\gamma}, \mathbf{C})$, the AoI-RT algorithm stabilizes the system, i.e.,*

$$\limsup_{K\to\infty} \frac{1}{K}\sum_{k=0}^{K-1}\sum_{l=1}^{L}\mathbb{E}[V_l[kT]] \le \frac{B}{\epsilon(1-\zeta)}, \tag{7}$$

where $B \triangleq \zeta G\sum_{l=1}^{L}\beta_l + \frac{1-\zeta}{2}L(A_{\max}^2+D_{\max}^2)$, $G \triangleq \min\{A_{\max}, TC_{\max}\}$, ϵ is some positive constant satisfying that $\boldsymbol{\lambda}+\epsilon\mathbf{1}$ still lies within the maximal satisfiable region, and $\mathbf{1}$ is an $L-$dimensional vector of all ones.

Proof. We apply the Lyapunov-drift analysis to obtain the feasibility optimality. Consider the Lyapunov function

$$W(\mathbf{V}[kT], \mathbf{R}[kT]) \triangleq \frac{1-\zeta}{2}\sum_{l=1}^{L}V_l^2[kT] + \zeta G\sum_{l=1}^{L}\beta_l R_l[kT], \tag{8}$$

where $\mathbf{V}[kT] \triangleq (V_l[kT])_{l=1}^{L}$ and $\mathbf{R}[kT] \triangleq (R_l[kT])_{l=1}^{L}$. It is shown in Appendix A that there exists a positive constant $\epsilon > 0$ satisfying $\boldsymbol{\lambda}+\epsilon\mathbf{1} \in \Lambda(\boldsymbol{\gamma}, \mathbf{C})$ such that

$$\begin{aligned}\Delta W[kT] \triangleq\ & \mathbb{E}\big[W(\mathbf{V}[(k+1)T], \mathbf{R}[(k+1)T]) \\ & - W(\mathbf{V}[kT], \mathbf{R}[kT])|\mathbf{V}[kT], \mathbf{R}[kT]\big] \\ \le\ & -\epsilon\sum_{l=1}^{L}(1-\zeta)V_l[kT] + B. \end{aligned}\tag{9}$$

Taking the expectation on the both sides of (9) and summing over $k = 0, 1, \ldots, K-1$, we have the desired result. □

Proposition 3.1 establishes the feasibility optimality of the AoI-RT algorithm. As we can see from (7), all the virtual queues are stabilized as long as $0 \le \zeta < 1$. However, when $\zeta = 1$, the sum of average virtual queue lengths becomes unbounded, which implies the AoI-Greedy policy is not able to meet timely throughput requirements. In the following subsection as well as simulation experiments, we will show that the proposed algorithm is also capable of improving the information freshness by reducing the AoI.

3.3 Data Freshness Guarantee

We analyze the AoI performance of our proposed algorithm in this subsection. Specifically, we derive an upper bound on the network data freshness defined in Section 2 under the AoI-RT algorithm, which demonstrates that the data freshness under our policy can be guaranteed. We also provide a lower bound on the network data freshness under general policies.

3.3.1 *Upper Bound Analysis.*

By Proposition 3.1, we have that $V_l[kT]$ and $R_l[kT]$ will converge to $\overline{V}_l^*$ and $\overline{R}_l^*$ in distribution in the steady state, where $\overline{V}_l^*$ and $\overline{R}_l^*$

denote the virtual-queue length and AoI of link l under the AoI-RT algorithm in the steady state, respectively. Then, we have the following proposition regarding the AoI performance.

Proposition 3.2. *The network data freshness under the AoI-RT algorithm by choosing $\beta_l = \frac{\omega_l}{\lambda_l(1-\gamma_l)}, \forall l$ is upper bounded by the following:*

$$\sum_{l=1}^{L}\omega_l\mathbb{E}\left[\overline{R}_l^*\right] \le \frac{G}{1+\epsilon}\sum_{l=1}^{L}\beta_l + \frac{1}{2(1+\epsilon)}\left(\frac{1}{\zeta}-1\right)\sum_{l=1}^{L}\mathbb{E}\left[\overline{A}_l^2+\overline{D}_l^2\right], \quad (10)$$

where $\epsilon > 0$ satisfies $\boldsymbol{\lambda}+\epsilon\mathbf{1} \in \Lambda(\boldsymbol{\gamma}, \mathbf{C})$, and $\overline{A}_l$ and $\overline{D}_l$ have the same distribution as $A_l[kT]$ and $D_l[kT]$, respectively for all l. Recall that $G \triangleq \min\{A_{\max}, TC_{\max}\}$.

Proof. See Appendix B for the details. □

Note that when ω_l, λ_l, and γ_l are given for all links, we can always find β_l for the AoI-RT algorithm such that $\omega_l = \beta_l\lambda_l(1-\gamma_l)$ holds. From Proposition 3.2, we can see that the considered network data freshness is upper bounded. The second term in (10) contains the second moment of the packet arrival processes and the channel fluctuations, showing that a better performance on data freshness in terms of AoI is dependent on less variations over the wireless channel and packet arrivals. *Although all the aforementioned factors are non-controllable, the AoI-RT algorithm provides a way of tuning data freshness performance via a controllable parameter ζ.* It can be seen that the performance of data freshness improves with the increase of ζ. It is worth noting that in the extreme case when $\zeta = 0$, the AoI-RT algorithm reduces to the pure VQL-based scheduling algorithms, and the upper bound given by Proposition 3.2 goes to infinity. This implies that the data freshness is not guaranteed by the conventional VQL-based scheduling policy even though it is feasibility-optimal; while in the extreme case when $\zeta = 1$, the second term disappears such that the AoI-Greedy policy yields the best AoI performance.

3.3.2 Lower Bound Analysis.

Next, we derive a fundamental lower bound on the network data freshness for a set of scheduling policies Π that can stabilize the system.

Proposition 3.3. *Under any policy $\pi \in \Pi$, the network data freshness is lower bounded by the following:*

$$\sum_{l=1}^{L}\omega_l\mathbb{E}\left[\overline{R}_l^{(\pi)}\right] \ge \frac{1}{2}\left(\sum_{l=1}^{L}\omega_l\right)\left(1+\frac{\sum_{l=1}^{L}\omega_l}{\Omega}\right), \quad (11)$$

where $\overline{R}_l^{(\pi)}$ denotes the AoI of link l under policy π in the steady state, and $\Omega \triangleq \max_{\{\mathbf{S}[\tau]\}_{\tau=0}^{T-1}}\sum_{l:(\sum_{\tau=0}^{T-1}S_l[\tau])>0}\omega_l$.

Proof. See Appendix C for the details. □

Specially, for a fully-connected non-fading network with $\omega_l = \omega$ and $A_l[kT] = A$ for all l and k, and each frame of one time slot, Proposition 3.3 implies $\sum_{l=1}^{L}\omega_l\mathbb{E}\left[\overline{R}_l^{(\pi)}\right] \ge \frac{1}{2}L(L+1)\omega$. Further, let per-link timely throughput requirement $\lambda_l(1-\gamma_l) = \frac{1}{L(1+\epsilon)}$ for all l. From Proposition 3.2, when $\zeta = 1$, $\sum_{l=1}^{L}\omega_l\mathbb{E}\left[\overline{R}_l^*\right] \le \omega GL^2$. Therefore, the network data freshness has an order of $\Theta(L^2)$ under the AoI-Greedy policy in the considered fully connected network, which also demonstrates the tightness of our upper and lower bounds on the network data freshness in this particular case.

3.4 Convergence Speed of Timely Throughput

In this subsection, we first establish the following proposition on the convergence speed of timely throughput, i.e., how fast the running average of the instantaneous timely throughput converges to the target average timely throughput with respect to the observation window size K (the number of frames observed).

Proposition 3.4. *The convergence speed of timely throughput under the AoI-RT algorithm is upper bounded as follows.*

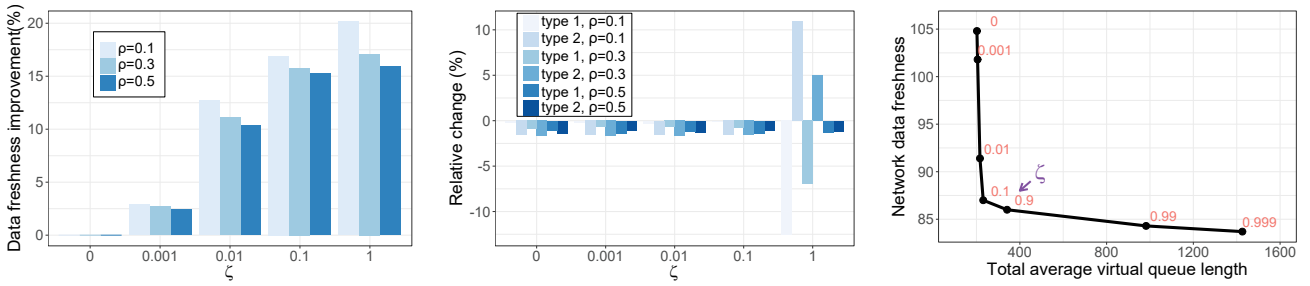
$$\mathbb{E}\left\|\frac{1}{K}\sum_{k=0}^{K-1}\mathbf{F}[kT]-\boldsymbol{\lambda}(1-\boldsymbol{\gamma})\right\|_1 \le \frac{1}{K}\sum_{l=1}^{L}V_l[0] + \frac{1}{K}\sum_{l=1}^{L}\mathbb{E}\left[V_l[KT]\right] + \frac{1}{\sqrt{K}}\sum_{l=1}^{L}\sqrt{Var\left(\overline{A}_l\right)+Var\left(\overline{D}_l\right)},$$

where $\overline{A}_l, \overline{D}_l$ have the same distribution as $A_l[kT]$, $D_l[kT]$, respectively for all l, $\|\cdot\|_1$ stands for the l_1 norm, $\mathbf{F}[kT] \triangleq (F_l[kT])_{l=1}^{L}$ denotes the instantaneous throughput vector, and

$$F_l[kT] \triangleq \min\left\{\sum_{t=kT}^{(k+1)T-1}C_l[kT]S_l[t], A_l[kT]\right\}.$$

Proof. Recall the dynamics of virtual queues (1). We have

$$\begin{aligned}
&\mathbb{E}\left\|\frac{1}{K}\sum_{k=0}^{K-1}\mathbf{F}[kT]-\boldsymbol{\lambda}(1-\boldsymbol{\gamma})\right\|_1 \\
&= \mathbb{E}\left\|\frac{1}{K}\sum_{k=0}^{K-1}\mathbf{F}[kT]-\frac{1}{K}\sum_{k=0}^{K-1}(\mathbf{A}[kT]-\mathbf{D}[kT]) + \frac{1}{K}\sum_{k=0}^{K-1}(\mathbf{A}[kT]-\mathbf{D}[kT])-\boldsymbol{\lambda}(1-\boldsymbol{\gamma})\right\|_1 \\
&= \mathbb{E}\left\|\frac{1}{K}(\mathbf{V}[0]-\mathbf{V}[KT]) + \frac{1}{K}\sum_{k=0}^{K-1}(\mathbf{A}[kT]-\mathbf{D}[kT])-\boldsymbol{\lambda}(1-\boldsymbol{\gamma})\right\|_1 \\
&\le \frac{1}{K}\left\|\mathbf{V}[0]\right\|_1 + \frac{1}{K}\mathbb{E}\left\|\mathbf{V}[KT]\right\|_1 + \mathbb{E}\left\|\frac{1}{K}\sum_{k=0}^{K-1}(\mathbf{A}[kT]-\mathbf{D}[kT])-\boldsymbol{\lambda}(1-\boldsymbol{\gamma})\right\|_1 \\
&= \frac{1}{K}\sum_{l=1}^{L}V_l[0] + \frac{1}{K}\sum_{l=1}^{L}\mathbb{E}\left[V_l[KT]\right] + \sum_{l=1}^{L}\mathbb{E}\left|\frac{1}{K}\sum_{k=0}^{K-1}(A_l[kT]-D_l[kT])-\lambda_l(1-\gamma_l)\right|. \quad (12)
\end{aligned}$$



(a) Improvement of network data freshness (b) Timely throughput performance (c) Data freshness vs. virtual-queue length ($\rho = 0.1$)

Figure 2: 100-link fully connected network

Note that each term in the last summation term in (12) can be upper bounded as follows.

$$\left(\mathbb{E}\left|\frac{1}{K}\sum_{k=0}^{K-1}(A_l[kT]-D_l[kT])-\lambda_l(1-\gamma_l)\right|\right)^2$$

$$\leq \mathbb{E}\left(\frac{1}{K}\sum_{k=0}^{K-1}(A_l[kT]-D_l[kT])-\lambda_l(1-\gamma_l)\right)^2 \quad (13)$$

$$= \mathrm{Var}\left(\frac{1}{K}\sum_{k=0}^{K-1}(A_l[kT]-D_l[kT])\right)$$

$$= \frac{1}{K^2}\sum_{k=0}^{K-1}\mathrm{Var}\left(A_l[kT]-D_l[kT]\right) \quad (14)$$

$$= \frac{1}{K}\left(\mathrm{Var}\left(\overline{A}_l\right)+\mathrm{Var}\left(\overline{D}_l\right)\right).$$

The inequality (13) is due to Jensen's inequality and the convexity of quadratic form. The equality (14) is due to the independence of $A_l[kT]$ and $D_l[kT]$ for all l and k. Thus,

$$\mathbb{E}\left|\frac{1}{K}\sum_{k=0}^{K-1}(A_l[kT]-D_l[kT])-\lambda_l(1-\gamma_l)\right|$$

$$\leq \frac{1}{\sqrt{K}}\sqrt{\mathrm{Var}\left(\overline{A}_l\right)+\mathrm{Var}\left(\overline{D}_l\right)}. \quad (15)$$

We have the desired result by substituting (15) to (12). □

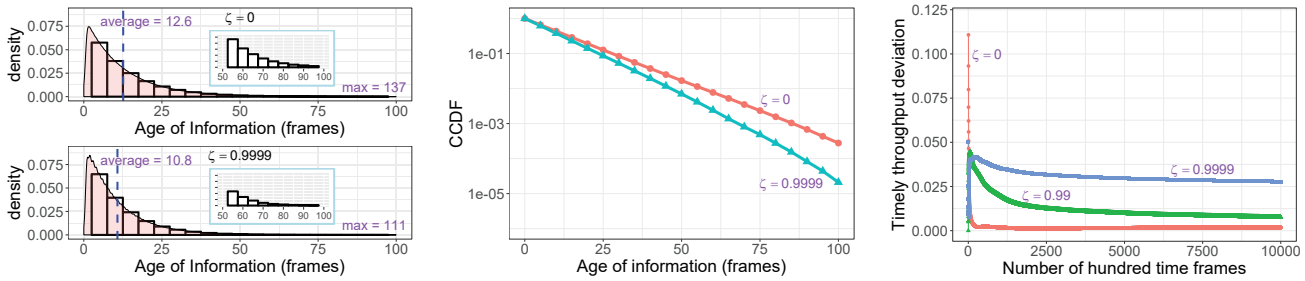
It is shown from Proposition 3.4 that the convergence speed of timely throughput depends on the virtual-queue length, and the random variations of the packet arrival process and the departure process of the virtual queue. Given the observation window size K, the increase of the total expected virtual-queue length slows down the convergence of timely throughput. Note that Proposition 3.1 also provides an upper bound on the expected total virtual-queue length in the steady state, which increases linearly with control parameter ζ. While from Proposition 3.2, it is known that we can improve the information freshness by increasing ζ. *The trade-off between data freshness and the convergence speed of timely throughput is now clear: when increasing ζ under the AoI-RT scheduling algorithm, the network data freshness is improved at the cost of slowing down the convergence of timely throughput.* The existence of this trade-off is due to the introduction of AoI in making scheduling decisions. It is interesting that we could significantly improve the data freshness in terms of AoI if a long convergence period is tolerable to the real-time applications (such as wireless sensor monitoring with soft timely throughput requirements [13]). We will also examine the trade-off in simulations to show how much AoI performance can be improved by slightly slowing down the convergence of timely throughput.

4 SIMULATION RESULTS

In this section, we present two sets of simulations to demonstrate the performance of the proposed AoI-RT algorithm.

In the first set of simulations, we consider a 100-link fully connected network (e.g., cellular networks). Particularly, among 100 links, there are N_1 links of type 1 each with a packet arrival (Bernoulli) rate of $\frac{0.32}{N_1}$, and N_2 links of type 2 each with an arrival (Bernoulli) rate of $\frac{0.32}{N_2}$. Note that $N_1 + N_2 = 100$. The ratio of N_1 to $N_1 + N_2$, denoted by ρ, can vary and we consider three different values of ρ: 0.1, 0.3, and 0.5. It can be seen that $\rho = 0.1$ and $\rho = 0.3$ lead to the scenarios where two types of links are of heterogeneous traffic; while when $\rho = 0.5$, all links have the same arrival rate. In addition, the frame has a length of one time slot, and for all links, $\gamma_l = 0.25$, $\beta_l = 1$, and $C_l[kT] = 1$ if scheduled. The simulation parameters are selected carefully such that the arrival process lies in the maximal satisfiable region $\Lambda(\boldsymbol{\gamma}, \mathbf{C})$.

The improvement of network data freshness in terms of AoI under our policy ($0 < \zeta < 1$), the VQL-based policy ($\zeta = 0$), and the AoI-Greedy policy ($\zeta = 1$) is shown in Fig. 2(a). The y-axis gives the improvement ratio of the network data freshness under different values of ζ to the network data freshness when $\zeta = 0$. As it can be seen, the performance of data freshness improves with the increase of ζ. From Fig. 2(a), we can also see that when the network presents certain traffic heterogeneity (the arrival rate difference between two types), a better improvement ratio can be obtained. For example, when $\rho = 0.1$ and $\zeta = 1$, the improvement ratio is about 20% compared to 15% when $\rho = 0.5$ (no heterogeneity) and $\zeta = 1$. This is because with traffic heterogeneity, it is easier for the scheduling algorithm to boost the AoI performance of links with lower traffic by reducing the service to links with higher traffic. The investigation of capability of meeting the timely throughput requirement is given in Fig. 2(b), where the relative change is defined by the



(a) AoI histogram comparison (link 5) (b) CCDF of AoI (link 5) (c) Convergence speed of timely throughput

Figure 3: 10-link ad hoc network

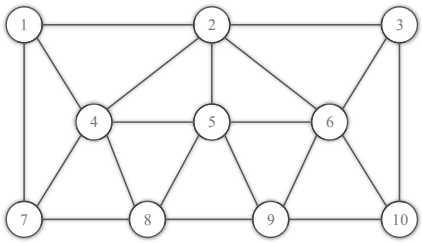


Figure 4: Conflict graph of the 10-link ad hoc network

difference (between the average timely throughput of a link and the timely throughput requirement) divided by the timely throughput requirement. It can be seen clearly that the relative change remains close to zero when varying ζ, showing that our policy can also provide the guarantee on timely throughput while improving AoI (as shown in Fig. 2(a)). However, when $\zeta = 1$, the timely throughput requirement cannot be satisfied under the AoI-Greedy policy. For example, when $\rho = 0.1$, the average timely throughput of type 1 links is over 10% below the requirement. Fig. 2(c) shows as expected that with the increase of ζ, the network data freshness improves; while the total average virtual-queue length increases (i.e., the convergence of timely throughput slows down[3]).

In the second set of simulations, we consider a 10-link ad hoc network in Fig. 4 with Bernoulli arrivals: $\lambda_5 = 0.1$ and $\lambda_l = 0.62$ for all other l. The frame has a length of two time slots. For all links, $\gamma_l = 0.2$, $\beta_l = 1$, and $C_l[kT]$ is Bernoulli with mean of 0.95 for all k.

We first investigate the AoI performance of link 5 (with lower traffic). Fig. 3(a) gives histograms of AoI over all the simulated time frames with normalized frequencies for link 5 when $\zeta = 0$ and $\zeta = 0.9999$, respectively. It illustrates that the occurrence of large values of AoI (e.g., > 50 frames) becomes less often when $\zeta = 0.9999$, indicating a better performance of data freshness. The complementary cumulative distribution function (CCDF) of AoI is further shown in Fig. 3(b), showing that when $\zeta = 0.9999$, the probability of seeing a large value of AoI is much smaller comparing the case where $\zeta = 0$ (e.g., can be almost 10 times less for a AoI value of 75). The convergence speed of timely throughput when varying ζ is shown in Fig. 3(c). As expected, when $\zeta = 0$, the timely throughput has the fastest convergence to the requirement. However, if the data freshness is more important to applications (as is often the case), we could trade for data freshness by increasing ζ.

Overall, the simulations demonstrate the ability of the proposed AoI-RT policy to improve the data freshness while providing timely throughput guarantees. Again, the VQL-based policy can only meet the timely throughput requirement; while the AoI-Greedy policy can only guarantee the data freshness at the receivers.

5 DISTRIBUTED IMPLEMENTATION

The proposed AoI-RT scheduling policy is centralized such that a centralized entity should compute the scheduling decision for every frame globally with necessary message passing. It is interesting to design low-complexity distributed implementations of the AoI-RT policy such that the scheduling decision can be made locally at each link with little or without message passing. We will consider the distributed implementation of our policy as future works. However, we would like to provide some discussions here. In fully connected networks, only one link can be scheduled for transmission at a time, which significantly reduces the scheduling decision space. A greedy solution that selects the maximum weighted schedule for every slot could be optimal and an algorithm similar to the FCSMA proposed in [9] could be used to implement such a greedy solution. Whereas for general ad hoc networks, the design of distributed algorithms could be very challenging. In [11], an asymptotically optimal algorithm is proposed for ad hoc networks with periodic packet arrivals and perfect channel conditions. Similarly, by utilizing the Markov approximation technique originally proposed in [1], we could design distributed algorithms of the AoI-RT policy for some special cases of ad hoc networks, which are amenable to low-complexity implementations.

6 CONCLUSION

In this work, we proposed the AoI-RT algorithm for scheduling real-time traffic with hard deadlines in ad hoc networks, which provides guarantees on both timely throughput and data freshness in terms of AoI. The algorithm takes the virtual-queue length and AoI of each link as driving factors in making scheduling decisions, which has been proved to be feasibility-optimal. We also derived

[3]Due to the lack of space, we only show the convergence of timely throughput in the second set of simulations, and both sets of simulations yield the same insight.

an upper bound of the data freshness metric to show that the performance of data freshness can be guaranteed under our policy. Particularly, we showed that the data freshness can be traded from the convergence speed of the timely throughput, which is done by tuning scheduling parameters. Simulation results further demonstrate the ability of our policy to improve the data freshness while providing the timely throughput guarantee. Future works include the distributed implementations of the proposed scheduling policy.

ACKNOWLEDGMENTS

This work is supported by the Natural Sciences and Engineering Research Council of Canada Discovery Grant: RGPIN-2017-05578; and the following US National Science Foundation grants: CNS-1717108, CNS-1651947, and CCF-1657162.

REFERENCES

[1] Chen, M., Liew, S. C., Shao, Z., and Kai, C. Markov approximation for combinatorial network optimization. *IEEE transactions on information theory 59*, 10 (2013), 6301–6327.
[2] Costa, M., Codreanu, M., and Ephremides, A. On the age of information in status update systems with packet management. *IEEE Transactions on Information Theory 62*, 4 (2016), 1897–1910.
[3] Hou, I.-H. On the modeling and optimization of short-term performance for real-time wireless networks. In *Proc. of IEEE INFOCOM* (San Francisco, CA, USA, April 2016).
[4] Hou, I.-H., Borkar, V., and Kumar, P. A theory of qos for wireless. In *Proc. of IEEE INFOCOM* (Rio de Janeiro, Brazil, April 2009).
[5] Jaramillo, J. J., Srikant, R., and Ying, L. Scheduling for optimal rate allocation in ad hoc networks with heterogeneous delay constraints. *IEEE Journal on Selected Areas in Communications 29*, 5 (2011), 979–987.
[6] Kadota, I., Uysal-Biyikoglu, E., Singh, R., and Modiano, E. Minimizing the age of information in broadcast wireless networks. In *Proc. of IEEE Allerton Conference* (Monticello, IL, USA, Octorber 2016).
[7] Kang, X., Hou, I., Ying, L., et al. On the capacity requirement of largest-deficit-first for scheduling real-time traffic in wireless networks. In *Proc. of ACM MobiHoc* (Hangzhou, China, June 2015).
[8] Kaul, S., Yates, R., and Gruteser, M. Real-time status: How often should one update? In *Proc. of IEEE INFOCOM* (Orlando, FL, USA, April 2012).
[9] Li, B., and Eryilmaz, A. Optimal distributed scheduling under time-varying conditions: A fast-csma algorithm with applications. *IEEE Transactions on Wireless Communications 12*, 7 (2013), 3278–3288.
[10] Li, B., Li, R., and Eryilmaz, A. Throughput-optimal scheduling design with regular service guarantees in wireless networks. *IEEE/ACM Transactions on Networking 23*, 5 (2015), 1542–1552.
[11] Lu, N., Li, B., Srikant, R., and Ying, L. Optimal distributed scheduling of real-time traffic with hard deadlines. In *Proc. of IEEE CDC* (Las Vegas, NV, USA, December 2016).
[12] Neely, M. J. Stochastic network optimization with application to communication and queueing systems. *Synthesis Lectures on Communication Networks 3*, 1 (2010), 1–211.
[13] Seno, L., Cena, G., Valenzano, A., and Zunino, C. Bandwidth management for soft real-time control applications in industrial wireless networks. *IEEE Transactions on Industrial Informatics 13*, 5 (2017), 2484–2495.
[14] Sun, Y., Uysal-Biyikoglu, E., Yates, R., Koksal, C. E., and Shroff, N. B. Update or wait: How to keep your data fresh. *IEEE Transactions on Information Theory 63*, 11 (2017), 7492–7508.

A PROOF OF INEQUALITY (9)

The evolution of $R_l[kT]$ (6) can be rewritten in a more compact form, i.e.,

$$\begin{aligned} R_l[(k+1)T] &= R_l[kT](1-\mathbb{I}_{\mathcal{H}_l[kT]})+1 \\ &= R_l[kT]-R_l[kT]\mathbb{I}_{\mathcal{H}_l[kT]}+1. \end{aligned} \tag{16}$$

To make our proof more concise, we omit the time index kT and use $\mathbf{x}^+$ to denote $\mathbf{x}[(k+1)T]$ for any $\mathbf{x}$. Hence, (16) becomes

$$R_l^+ = R_l - R_l\mathbb{I}_{\mathcal{H}_l}+1. \tag{17}$$

Multiplying β_l on both sides of (16), and summing over all links, we have

$$\sum_{l=1}^{L}\beta_l R_l^+ = \sum_{l=1}^{L}\beta_l R_l - \sum_{l=1}^{L}\beta_l R_l\mathbb{I}_{\mathcal{H}_l}+\sum_{l=1}^{L}\beta_l. \tag{18}$$

We are now ready to prove the inequality (9). Indeed, we have

$$\begin{aligned} \Delta W \triangleq& \mathbb{E}[W(\mathbf{V}^+,\mathbf{R}^+)-W(\mathbf{V},\mathbf{R})|\mathbf{V},\mathbf{R}] \\ =&\mathbb{E}\Bigg[\frac{1-\zeta}{2}\sum_{l=1}^{L}(V_l^+)^2+\zeta G\sum_{l=1}^{L}\beta_l R_l^+ \\ &\qquad -\frac{1-\zeta}{2}\sum_{l=1}^{L}V_l^2-\zeta G\sum_{l=1}^{L}\beta_l R_l\Bigg|\mathbf{V},\mathbf{R}\Bigg] \\ \le&\frac{1-\zeta}{2}\sum_{l=1}^{L}\mathbb{E}[(V_l+I_l-D_l)^2-V_l^2|\mathbf{V},\mathbf{R}] \\ &\qquad +\zeta G\mathbb{E}\left[\sum_{l=1}^{L}\beta_l R_l^+-\sum_{l=1}^{L}\beta_l R_l\middle|\mathbf{V},\mathbf{R}\right], \end{aligned} \tag{19}$$

where the last step follows from the dynamics of virtual queues (1) and the fact that $(\max\{\cdot,0\})^2\le(\cdot)^2$. By plugging (18) into (19), we have

$$\begin{aligned} \Delta W \le& \frac{1-\zeta}{2}\sum_{l=1}^{L}\mathbb{E}[(V_l+I_l-D_l)^2-V_l^2|\mathbf{V},\mathbf{R}] \\ &+\zeta G\mathbb{E}\left[\sum_{l=1}^{L}\beta_l-\sum_{l=1}^{L}\beta_l R_l\mathbb{I}_{\mathcal{H}_l}\middle|\mathbf{V},\mathbf{R}\right] \\ =&(1-\zeta)\sum_{l=1}^{L}\mathbb{E}\left[V_l(I_l-D_l)+\frac{1}{2}(I_l-D_l)^2\middle|\mathbf{V},\mathbf{R}\right] \\ &+\zeta G\sum_{l=1}^{L}\beta_l-\zeta G\mathbb{E}\left[\sum_{l=1}^{L}\beta_l R_l\mathbb{I}_{\mathcal{H}_l}\middle|\mathbf{V},\mathbf{R}\right] \\ \le&(1-\zeta)\sum_{l=1}^{L}\lambda_l(1-\gamma_l)V_l+B-\zeta G\mathbb{E}\left[\sum_{l=1}^{L}\beta_l R_l\mathbb{I}_{\mathcal{H}_l}\middle|\mathbf{V},\mathbf{R}\right] \\ &-\mathbb{E}\left[\sum_{l=1}^{L}(1-\zeta)V_l\min\left\{\sum_{t=kT}^{(k+1)T-1}C_l S_l[t],A_l\right\}\middle|\mathbf{V},\mathbf{R}\right], \end{aligned} \tag{20}$$

where the last step is true for B that is defined in Proposition 3.1 and uses the fact that I_l denotes the number of dropped packets (see (2)). Note that $\mathcal{H}_l$ is the event that at least one packet is delivered at link l in frame k. We use $G\triangleq\min\{A_{\max},TC_{\max}\}$ to represent the maximum number of packets that can be delivered within one frame, and therefore we have

$$\begin{aligned} &\mathbb{E}\left[\sum_{l=1}^{L}\beta_l R_l\mathbb{I}_{\mathcal{H}_l}\middle|\mathbf{V},\mathbf{R}\right] \\ &\ge\frac{1}{G}\mathbb{E}\left[\sum_{l=1}^{L}\beta_l R_l\min\left\{\sum_{t=kT}^{(k+1)T-1}C_l S_l[t],A_l\right\}\middle|\mathbf{V},\mathbf{R}\right]. \end{aligned} \tag{21}$$

By plugging (21) into (20), we have

$$\Delta W \le (1-\zeta)\sum_{l=1}^{L}\lambda_l(1-\gamma_l)V_l + B - \sum_{l=1}^{L}\mathbb{E}\left[((1-\zeta)V_l+\zeta\beta_l R_l)\min\left\{\sum_{t=kT}^{(k+1)T-1} C_l S_l[t], A_l\right\}\middle|\mathbf{V},\mathbf{R}\right]. \tag{22}$$

Since the arrival process is strictly within the maximal satisfiable region, there exists an $\epsilon > 0$ such that

$$\lambda_l(1-\gamma_l) \le -\epsilon + \sum_{\mathbf{a}}\mathrm{P_A}(\mathbf{a})\sum_{\mathbf{c}}\mathrm{P_C}(\mathbf{c})\sum_{\mathbf{s}_0,\mathbf{s}_1,\ldots,\mathbf{s}_{T-1}\in\mathcal{S}} \alpha(\mathbf{a},\mathbf{c};\mathbf{s}_0,\mathbf{s}_1,\ldots,\mathbf{s}_{T-1})\min\left\{\sum_{\tau=1}^{T-1} c_l s_{\tau,l}, a_l\right\}, \forall l. \tag{23}$$

Therefore, we have

$$\begin{aligned}&(1-\zeta)\sum_{l=1}^{L}\lambda_l(1-\gamma_l)V_l\\ &\le -\epsilon\sum_{l=1}^{L}(1-\zeta)V_l + \sum_{l=1}^{L}(1-\zeta)V_l\sum_{\mathbf{a}}\mathrm{P_A}(\mathbf{a})\sum_{\mathbf{c}}\mathrm{P_C}(\mathbf{c})\\ &\quad\sum_{\mathbf{s}_0,\mathbf{s}_1,\ldots,\mathbf{s}_{T-1}\in\mathcal{S}}\alpha(\mathbf{a},\mathbf{c};\mathbf{s}_0,\mathbf{s}_1,\ldots,\mathbf{s}_{T-1})\min\left\{\sum_{\tau=1}^{T-1} c_l s_{\tau,l}, a_l\right\}\end{aligned} \tag{24}$$

Next, we focus on the second term of the right-hand-side of the above inequality.

$$\begin{aligned}&\sum_{l=1}^{L}(1-\zeta)V_l\sum_{\mathbf{a}}\mathrm{P_A}(\mathbf{a})\sum_{\mathbf{c}}\mathrm{P_C}(\mathbf{c})\\ &\quad\sum_{\mathbf{s}_0,\mathbf{s}_1,\ldots,\mathbf{s}_{T-1}\in\mathcal{S}}\alpha(\mathbf{a},\mathbf{c};\mathbf{s}_0,\mathbf{s}_1,\ldots,\mathbf{s}_{T-1})\min\left\{\sum_{\tau=1}^{T-1} c_l s_{\tau,l}, a_l\right\}\\ &\le \sum_{\mathbf{a}}\mathrm{P_A}(\mathbf{a})\sum_{\mathbf{c}}\mathrm{P_C}(\mathbf{c})\sum_{\mathbf{s}_0,\mathbf{s}_1,\ldots,\mathbf{s}_{T-1}\in\mathcal{S}}\alpha(\mathbf{a},\mathbf{c};\mathbf{s}_0,\mathbf{s}_1,\ldots,\mathbf{s}_{T-1})\\ &\quad\sum_{l=1}^{L}\left((1-\zeta)V_l+\zeta\beta_l R_l\right)\min\left\{\sum_{\tau=1}^{T-1} c_l s_{\tau,l}, a_l\right\}\\ &\le \sum_{\mathbf{a}}\mathrm{P_A}(\mathbf{a})\sum_{\mathbf{c}}\mathrm{P_C}(\mathbf{c})\\ &\quad\max_{\substack{\mathrm{S}[t]\in\mathcal{S},\forall t=kT,\\ \ldots,(k+1)T-1}}\sum_{l=1}^{L}\left((1-\zeta)V_l+\zeta\beta_l R_l\right)\min\left\{\sum_{t=kT}^{(k+1)T-1} C_l S_l[t], A_l\right\}\\ &= \sum_{l=1}^{L}\mathbb{E}\left[((1-\zeta)V_l+\zeta\beta_l R_l)\min\left\{\sum_{t=kT}^{(k+1)T-1} C_l S_l[t], A_l\right\}\middle|\mathbf{V},\mathbf{R}\right],\end{aligned} \tag{25}$$

where the second last step follows from the definition of the AoI-RT algorithm.

By combining (24) and (25) and substituting them into (22), we have the desired result.

B PROOF OF PROPOSITION 3.2

We consider the following quadratic Lyapunov function: $W(\mathbf{V},\mathbf{R}) \triangleq \frac{1-\zeta}{2}\sum_{l=1}^{L} V_l^2$. Then, we have

$$\begin{aligned}\Delta W(\mathbf{V},\mathbf{R}) &= \mathbb{E}[W(\mathbf{V}^+,\mathbf{R}^+) - W(\mathbf{V},\mathbf{R})|\mathbf{V},\mathbf{R}]\\ &= \mathbb{E}\left[\frac{1-\zeta}{2}\sum_{l=1}^{L}(V_l^+)^2 - \frac{1-\zeta}{2}\sum_{l=1}^{L}V_l^2\middle|\mathbf{V},\mathbf{R}\right]\\ &\le \frac{1-\zeta}{2}\sum_{l=1}^{L}\mathbb{E}\left[(V_l + I_l - D_l)^2 - V_l^2|\mathbf{V},\mathbf{R}\right]\\ &\le (1-\zeta)\sum_{l=1}^{L}\mathbb{E}\left[V_l(A_l - D_l)|\mathbf{V},\mathbf{R}\right] + \frac{1-\zeta}{2}\sum_{l=1}^{L}\mathbb{E}\left[I_l^2 + D_l^2|\mathbf{V},\mathbf{R}\right]\\ &\quad - \sum_{l=1}^{L}\mathbb{E}\left[(1-\zeta)V_l\min\left\{\sum_{t=kT}^{(k+1)T-1} C_l S_l[t], A_l\right\}\middle|\mathbf{V},\mathbf{R}\right].\end{aligned}$$

Taking expectation on both sides with respect to the steady-state distribution of $(\mathbf{V},\mathbf{R})$, i.e., $(\overline{\mathbf{V}},\overline{\mathbf{R}})$, and using the fact that $\mathbb{E}[\Delta W(\overline{\mathbf{V}},\overline{\mathbf{R}})] = 0$, we have

$$\begin{aligned}0 &\le (1-\zeta)\sum_{l=1}^{L}\lambda_l(1-\gamma_l)\mathbb{E}\left[\overline{V}_l^*\right] + \frac{1-\zeta}{2}\sum_{l=1}^{L}\mathbb{E}\left[\overline{A}_l^2 + \overline{D}_l^2\right]\\ &\quad - \sum_{l=1}^{L}\mathbb{E}\left[(1-\zeta)\overline{V}_l^*\min\left\{\overline{A}_l, \overline{C}_l\sum_{\tau=0}^{T-1}\overline{S}_{\tau,l}^*\right\}\right],\end{aligned}$$

which implies

$$\begin{aligned}&\sum_{l=1}^{L}\mathbb{E}\left[(1-\zeta)\overline{V}_l^*\min\left\{\overline{A}_l, \overline{C}_l\sum_{\tau=0}^{T-1}\overline{S}_{\tau,l}^*\right\}\right]\\ &\quad\le (1-\zeta)\sum_{l=1}^{L}\lambda_l(1-\gamma_l)\mathbb{E}\left[\overline{V}_l^*\right] + \frac{1-\zeta}{2}\sum_{l=1}^{L}\mathbb{E}\left[\overline{A}_l^2 + \overline{D}_l^2\right].\end{aligned}$$

By adding $\zeta\sum_{l=1}^{L}\beta_l\mathbb{E}\left[\overline{R}_l^*\min\left\{\overline{A}_l, \overline{C}_l\sum_{\tau=0}^{T-1}\overline{S}_{\tau,l}^*\right\}\right]$ on both sides, we have

$$\begin{aligned}&\sum_{l=1}^{L}\mathbb{E}\left[\left((1-\zeta)\overline{V}_l^* + \zeta\beta_l\overline{R}_l^*\right)\min\left\{\overline{A}_l, \overline{C}_l\sum_{\tau=0}^{T-1}\overline{S}_{\tau,l}^*\right\}\right]\\ &\quad\le (1-\zeta)\sum_{l=1}^{L}\lambda_l(1-\gamma_l)\mathbb{E}\left[\overline{V}_l^*\right] + \frac{1-\zeta}{2}\sum_{l=1}^{L}\mathbb{E}\left[\overline{A}_l^2 + \overline{D}_l^2\right]\\ &\quad\quad + \zeta\sum_{l=1}^{L}\beta_l\mathbb{E}\left[\overline{R}_l^*\min\left\{\overline{A}_l, \overline{C}_l\sum_{\tau=0}^{T-1}\overline{S}_{\tau,l}^*\right\}\right].\end{aligned} \tag{26}$$

Since the arrival process is strictly within the maximal satisfiable region $\Lambda(\boldsymbol{\gamma},\mathbf{C})$ and the optimal solution always occurs at extreme points in linear programming, under our proposed AoI-RT algorithm, we have

$$\begin{aligned}&\sum_{l=1}^{L}(1+\epsilon)\lambda_l(1-\gamma_l)((1-\zeta)V_l+\zeta\beta_l R_l)\\ &\quad\le \sum_{l=1}^{L}\mathbb{E}\left[((1-\zeta)V_l+\zeta\beta_l R_l)\min\left\{C_l\sum_{t=kT}^{(k+1)T-1} S_l^*[t], A_l\right\}\middle|\mathbf{V},\mathbf{R}\right].\end{aligned}$$

Then, taking expectation on both sides of above inequality with respect to the steady-state distribution of $(\mathbf{V}, \mathbf{R})$, we have

$$\sum_{l=1}^{L}(1+\epsilon)\lambda_l(1-\gamma_l)\mathbb{E}((1-\zeta)\overline{V}_l^* + \zeta\beta_l\overline{R}_l^*)$$
$$\leq \sum_{l=1}^{L}\mathbb{E}\left[((1-\zeta)\overline{V}_l^* + \zeta\beta_l\overline{R}_l^*)\min\left\{\overline{A}_l, \overline{C}_l\sum_{\tau=0}^{T-1}\overline{S}_{\tau,l}^*\right\}\right]. \quad (27)$$

By selecting $\omega_l = \beta_l\lambda_l(1-\gamma_l), \forall l$ and plugging (27) into (26), we have

$$\begin{aligned}\sum_{l=1}^{L}\omega_l\mathbb{E}\left[\overline{R}_l^*\right] &= \sum_{l=1}^{L}\beta_l\lambda_l(1-\gamma_l)\mathbb{E}\left[\overline{R}_l^*\right]\\ &\leq \frac{1}{1+\epsilon}\sum_{l=1}^{L}\beta_l\mathbb{E}\left[\overline{R}_l^*\min\left\{\overline{A}_l, \overline{C}_l\sum_{\tau=0}^{T-1}\overline{S}_{\tau,l}^*\right\}\right]\\ &\quad + \frac{1-\zeta}{2\zeta(1+\epsilon)}\sum_{l=1}^{L}\mathbb{E}\left[\overline{A}_l^2 + \overline{D}_l^2\right]\\ &\overset{(a)}{\leq} \frac{G}{1+\epsilon}\sum_{l=1}^{L}\beta_l\mathbb{E}[\overline{R}_l^*\mathbb{I}_{\overline{\mathcal{H}}_l}] + \frac{1-\zeta}{2\zeta(1+\epsilon)}\sum_{l=1}^{L}\mathbb{E}\left[\overline{A}_l^2 + \overline{D}_l^2\right]\\ &\overset{(b)}{=} \frac{G}{1+\epsilon}\sum_{l=1}^{L}\beta_l + \frac{1}{2(1+\epsilon)}\left(\frac{1}{\zeta}-1\right)\sum_{l=1}^{L}\mathbb{E}\left[\overline{A}_l^2 + \overline{D}_l^2\right],\end{aligned}$$

where step (a) uses the fact that G denotes the maximum number of packets that can be delivered within one frame; and (b) uses the following equality:

$$\sum_{l=1}^{L}\beta_l\mathbb{E}[\overline{R}_l^*\mathbb{I}_{\overline{\mathcal{H}}_l}] = \sum_{l=1}^{L}\beta_l, \quad (28)$$

which is from taking expectation on both sides of (18) with respect to the steady-state distribution of $\mathbf{R}$.

C PROOF OF PROPOSITION 3.3

For any policy π, given $\overline{\mathcal{H}}^{(\pi)} \triangleq \left\{l : \min\left\{\overline{A}_l, \overline{C}_l\sum_{\tau=0}^{T-1}\overline{S}_{\tau,l}^{(\pi)}\right\} > 0\right\}$, we have

$$\left(\sum_{l\in\overline{\mathcal{H}}^{(\pi)}}\omega_l\overline{R}_l^{(\pi)}\right)^2 = \left(\sum_{l\in\overline{\mathcal{H}}^{(\pi)}}\sqrt{\omega_l}\sqrt{\omega_l}\overline{R}_l^{(\pi)}\right)^2$$
$$\leq \left(\sum_{l\in\overline{\mathcal{H}}^{(\pi)}}\omega_l\right)\sum_{l\in\overline{\mathcal{H}}^{(\pi)}}\omega_l\left(\overline{R}_l^{(\pi)}\right)^2,$$

which is due to Cauchy-Schwarz inequality. This implies

$$\sum_{l\in\overline{\mathcal{H}}^{(\pi)}}\omega_l\left(\overline{R}_l^{(\pi)}\right)^2 \geq \frac{\left(\sum_{l\in\overline{\mathcal{H}}^{(\pi)}}\omega_l\overline{R}_l^{(\pi)}\right)^2}{\sum_{l\in\overline{\mathcal{H}}^{(\pi)}}\omega_l}. \quad (29)$$

Multiplying ω_l on both sides of (16), summing over all links, and taking expectation on both sides, we have

$$\mathbb{E}\left[\sum_{l\in\overline{\mathcal{H}}^{(\pi)}}\omega_l\overline{R}_l^{(\pi)}\right] = \sum_{l=1}^{L}\omega_l. \quad (30)$$

Taking expectation on both sides of (29) and plugging in (30), we have

$$\mathbb{E}\left[\sum_{l\in\overline{\mathcal{H}}^{(\pi)}}\omega_l\left(\overline{R}_l^{(\pi)}\right)^2\right] \geq \mathbb{E}\left[\frac{\left(\sum_{l\in\overline{\mathcal{H}}^{(\pi)}}\omega_l\overline{R}_l^{(\pi)}\right)^2}{\sum_{l\in\overline{\mathcal{H}}^{(\pi)}}\omega_l}\right]$$
$$\geq \frac{\left(\mathbb{E}\left[\sum_{l\in\overline{\mathcal{H}}^{(\pi)}}\omega_l\overline{R}_l^{(\pi)}\right]\right)^2}{\mathbb{E}\left[\sum_{l\in\overline{\mathcal{H}}^{(\pi)}}\omega_l\right]} = \frac{\left(\sum_{l=1}^{L}\omega_l\right)^2}{\mathbb{E}\left[\sum_{l\in\overline{\mathcal{H}}^{(\pi)}}\omega_l\right]}, \quad (31)$$

where the second inequality is due to Jensen's inequality and the fact that $g(a,b) = \frac{a^2}{b}$ is convex.

Next, we obtain the following regarding the quadratic term of R^+ from (17),

$$\begin{aligned}\sum_{l=1}^{L}\omega_l\left(R_l^+\right)^2 &= \sum_{l=1}^{L}\omega_l\left(R_l - R_l\mathbb{I}_{\mathcal{H}_l} + 1\right)^2\\ &= \sum_{l=1}^{L}\omega_l R_l^2 + \sum_{l=1}^{L}\omega_l R_l^2\mathbb{I}_{\mathcal{H}_l}^2 - 2\sum_{l=1}^{L}\omega_l R_l^2\mathbb{I}_{\mathcal{H}_l}\\ &\quad + 2\sum_{l=1}^{L}\omega_l R_l - 2\sum_{l=1}^{L}\omega_l R_l\mathbb{I}_{\mathcal{H}_l} + \sum_{l=1}^{L}\omega_l\\ &= \sum_{l=1}^{L}\omega_l R_l^2 - \sum_{l\in\mathcal{H}}\omega_l R_l^2 + 2\sum_{l=1}^{L}\omega_l R_l - 2\sum_{l\in\mathcal{H}}\omega_l R_l + \sum_{l=1}^{L}\omega_l,\end{aligned} \quad (32)$$

where $\mathcal{H} \triangleq \{l : \mathbb{I}_{\mathcal{H}_l} = 1\}$. Taking expectation on both sides of (32) with respect to the steady-state distribution of $(\mathbf{V}, \mathbf{R})$, we have

$$2\sum_{l=1}^{L}\omega_l\mathbb{E}\left[\overline{R}_l^{(\pi)}\right] = 2\mathbb{E}\left[\sum_{l\in\overline{\mathcal{H}}^{(\pi)}}\omega_l\overline{R}_l^{(\pi)}\right] + \mathbb{E}\left[\sum_{l\in\overline{\mathcal{H}}^{(\pi)}}\omega_l\left(\overline{R}_l^{(\pi)}\right)^2\right] - \sum_{l=1}^{L}\omega_l.$$

After plugging in (30), we have

$$\sum_{l=1}^{L}\omega_l\mathbb{E}\left[\overline{R}_l^{(\pi)}\right] = \frac{1}{2}\mathbb{E}\left[\sum_{l\in\overline{\mathcal{H}}^{(\pi)}}\omega_l\left(\overline{R}_l^{(\pi)}\right)^2\right] + \frac{1}{2}\sum_{l=1}^{L}\omega_l. \quad (33)$$

By plugging (31) into (33), we have

$$\sum_{l=1}^{L}\omega_l\mathbb{E}\left[\overline{R}_l^{(\pi)}\right] \geq \frac{1}{2}\left(\sum_{l=1}^{L}\omega_l\right)\left(1 + \frac{\sum_{l=1}^{L}\omega_l}{\mathbb{E}\left[\sum_{l\in\overline{\mathcal{H}}^{(\pi)}}\omega_l\right]}\right). \quad (34)$$

Since $\mathbb{E}\left[\sum_{l\in\overline{\mathcal{H}}^{(\pi)}}\omega_l\right]$ is upper bounded by the maximum of the summation of ω_l over all scheduled links l over a frame regardless of scheduling policies, i.e.,

$$\mathbb{E}\left[\sum_{l\in\overline{\mathcal{H}}^{(\pi)}}\omega_l\right] \leq \max_{\{\mathbf{S}[\tau]\}_{\tau=0}^{T-1}}\sum_{l:(\sum_{\tau=0}^{T-1}S_l[\tau])>0}\omega_l, \quad (35)$$

the proposition holds.