Scheduling Algorithms for Optimizing Age of Information in Wireless Networks With Throughput Constraints

Igor Kadota, Abhishek Sinha, and Eytan Modiano, Fellow, IEEE

Abstract—Age of Information (AoI) is a performance metric that captures the freshness of the information from the perspective of the destination. The AoI measures the time that elapsed since the generation of the packet that was most recently delivered to the destination. In this paper, we consider a single-hop wireless network with a number of nodes transmitting time-sensitive information to a base station and address the problem of minimizing the expected weighted sum AoI of the network while simultaneously satisfying timely-throughput constraints from the nodes. We develop four low-complexity transmission scheduling policies that attempt to minimize AoI subject to minimum throughput requirements and evaluate their performance against the optimal policy. In particular, we develop a randomized policy, a Max-Weight policy, a Drift-Plus-Penalty policy, and a Whittle’s Index policy, and show that they are guaranteed to be within a factor of two, four, two, and eight, respectively, away from the minimum AoI possible. The simulation results show that Max-Weight and Drift-Plus-Penalty outperform the other policies, both in terms of AoI and throughput, in every network configuration simulated, and achieve near-optimal performance.

Index Terms—Age of information, throughput, scheduling, optimization, quality of service, wireless networks.

I. INTRODUCTION

The Age of Information (AoI) is a performance metric that measures the time that elapsed since the generation of the packet that was most recently delivered to the destination. This metric captures the freshness of the information from the perspective of the destination. Consider a cyber-physical system such as an automated industrial plant, a smart car, or a modern car, where a number of sensors are transmitting time-sensitive information to a monitor over unreliable wireless channels. Each sensor samples information from a physical phenomena (e.g. pressure of the tire, quantity of fuel, proximity to obstacles and engine rotational speed) and transmits this data to the monitor. Ideally, the monitor receives fresh information about every physical phenomena continuously. However, due to limitations of the wireless channel, this is often impractical. In such cases, the system has to manage the use of the available channel resources in order to keep the monitor updated. In this paper, we develop four low-complexity transmission scheduling policies and analyze their performance in terms of the freshness of the information at the monitor, namely the Age of Information.

Let every packet be time-stamped with the time it was generated. Denote by \( \tau_i[m] \) the time-stamp of the \( m \)th packet delivered by sensor \( i \) to the monitor. Assume that at time \( t \), the \( m \)th packet delivered by sensor \( i \) is the most recent. Then, the Age of Information associated with sensor \( i \) at time \( t \) is given by \( h_i(t) = t - \tau_i[m] \). While the monitor does not receive new packets from sensor \( i \), the value of \( h_i(t) \) increases linearly with \( t \), representing the information getting older. As soon as the monitor receives a new packet from sensor \( i \), the corresponding time-stamp is instantaneously updated from \( \tau_i[m] \) to \( \tau_i[m+1] \), reducing the value of \( h_i(t) \) by \( \tau_i[m+1] - \tau_i[m] \). Notice that at the moment packet \( (m+1) \) is delivered to the monitor, the value of \( h_i(t) \) matches the delay of the packet. This makes sense because, at that moment, the information at the monitor is as old as the information contained in packet \( (m+1) \). It follows naturally that a good AoI performance is achieved when packets with low delay are delivered regularly.

In order to provide good AoI performance, the scheduling policy must control how the channel resources are allocated to the different sensors in the network. Depending on the channel conditions and network configuration, this can mean that some sensors get to transmit repeatedly, while other sensors less often. The frequency at which information is delivered to the monitor is of particular importance in sensor networks. Clearly, a sensor that measures the quantity of fuel requires a lower update frequency (i.e. throughput) than a sensor that is measuring the proximity to obstacles in order to avoid collisions. For capturing this attribute, we associate a minimum timely-throughput requirement with each sensor in the network. Hence, in addition to providing good AoI performance, the scheduling policies should also fulfill timely-throughput constraints from the individual sensors.

A framework for modeling wireless networks with timely-throughput requirements was proposed in [2] together with two debt-based scheduling policies that fulfill any feasible...
requirements. Generalizations of this model to different network configurations were proposed in [3]–[5]. Scheduling policies that maximize throughput and also provide service regularity in wireless networks were studied in [6]–[8]. The problem of minimizing AoI was introduced in [9]–[11]. In [11]–[18], different queueing systems are analyzed and the optimal server utilization with respect to AoI is found. In [19]–[22], the authors optimize the process of generating information updates in order to minimize AoI. The design of scheduling policies based on AoI is considered in [23]–[35].

In [19]–[22], the authors optimize the process of generating information updates in order to minimize AoI. The design of scheduling policies based on AoI is considered in [23]–[35]. Most relevant to our paper are [6], [25], [27], [31]–[34]. In [6], the Time-Since-Last-Delivery \(^1\) (TSDL) is introduced as a measure of service regularity and a Max-Weight scheduling policy based on TSDL and queue length is developed and analyzed. In [25], a Greedy policy, which transmits the packet with highest current age, is shown to be AoI optimal for symmetric network and a scheduling policy based on the Whittle’s Index is developed. This work is extended in [32], where the authors develop and analyze a stationary randomized policy and a Max-Weight policy based on AoI. In [27], the AoI minimization problem is formulated as a Markov Decision Process and structural properties of the optimal scheduling policy are obtained. In [31], a round-robin policy and a stationary randomized policy are optimized in terms of AoI. In [34], a stationary randomized policy is optimized with respect to average AoI and peak AoI.

In this paper, we develop policies that minimize AoI subject to minimum throughput requirements, where timely-throughput is modeled as in [2], and evaluate their performance against an AoI lower bound. To the best of our knowledge, this is the first work to consider AoI-based policies that provably satisfy throughput constraints of multiple destinations simultaneously.

An important observation is that high throughput does not guarantee low AoI. Next, we provide two examples that illustrate the importance of low delay and service regularity. **Example 1:** consider an M/M/1 queue with high arrival rate and low service rate. In this system, the queue is often filled, resulting in high throughput and high packet delay. This high delay means that packets being served contain outdated information. Hence, despite the high throughput, the AoI may still be high. **Example 2:** consider a network with two nodes sharing the wireless channel during a given time-interval. The scheduling policy selects which node is allowed to transmit packets at any given time. Policy \( A \) selects node 1 repeatedly in the first half of the interval and node 2 in the second half. Policy \( B \) alternates between nodes 1 and 2 throughout the entire interval. Despite the fact that both policies have the same throughput, policy \( B \) may outperform policy \( A \) in terms of AoI due to its superior service regularity.

In this paper, we assume that nodes can generate a new packet with fresh information when scheduled. This assumption is motivated by applications in which end nodes can generate (or sample) information on-demand. This assumption allows us to gain insight into the scheduling problem. It is important to notice that the techniques employed in this paper may be applicable to related models with stochastic packet arrivals and buffering.

The remainder of this paper is outlined as follows. In Sec. II, the network model and performance metrics are formally presented. Then, in Sec. III, four low-complexity scheduling policies are proposed and analyzed. In Sec. IV, those policies are simulated and compared to the state-of-the-art in the literature. The paper is concluded in Sec. V.

**II. System Model**

Consider a single-hop wireless network with a base station (BS) receiving time-sensitive information from \( M \) nodes. Let the time be slotted, with slot index \( k \in \{1, 2, \cdots, K\} \), and consider a wireless channel that allows at most one packet transmission per slot. In each slot \( k \), the BS either idles or selects a node \( i \in \{1, 2, \cdots, M\} \) for transmission. Let \( u_i(k) \) be the indicator function that is equal to 1 when the BS selects node \( i \) during slot \( k \), and \( u_i(k) = 0 \) otherwise. When \( u_i(k) = 1 \), node \( i \) samples fresh information, generates a new packet and sends this packet over the wireless channel. The packet from node \( i \) is successfully received by the BS with probability \( p_i \in (0, 1] \) and a transmission error occurs with probability \( 1 - p_i \). The probability \( p_i \) does not change with time, but may differ between nodes.

The transmission scheduling policy controls the decision of the BS in each slot \( k \), which is represented by the set of values \( \{u_i(k)\}_{i=1}^{M} \). The **interference constraint** associated with the wireless channel imposes that

\[
\sum_{i=1}^{M} u_i(k) \leq 1, \quad \forall k \in \{1, \cdots, K\}, \tag{1}
\]

meaning that at any given slot \( k \), the scheduling policy can select at most one node for transmission. Let \( d_i(k) \) be the random variable that indicates when a packet from node \( i \) is delivered to the BS. If node \( i \) transmits a packet during slot \( k \), i.e. \( u_i(k) = 1 \), then \( d_i(k) = 1 \) with probability \( p_i \) and \( d_i(k) = 0 \) with probability \( 1 - p_i \). On the other hand, if node \( i \) does not transmit, i.e. \( u_i(k) = 0 \), then \( d_i(k) = 0 \) with probability one. It follows that

\[
\mathbb{E}[d_i(k)|u_i(k)] = p_i u_i(k) \quad \text{and, applying the law of iterated expectations}
\]

\[
\mathbb{E}[d_i(k)] = p_i \mathbb{E}[u_i(k)]. \tag{2}
\]

In this paper, we consider non-anticipative scheduling policies, i.e. policies that do not use future knowledge in making decisions. Denote by \( \Pi \) the class of non-anticipative policies and let \( \pi \in \Pi \) be an arbitrary admissible policy. Our goal is to design low-complexity scheduling policies that belong to \( \Pi \), provide close to optimal AoI performance and, at the same time, guarantee a minimum throughput level for each individual destination. Next, we formally introduce both performance metrics, throughput and AoI, and define a measure for "closeness to optimality".

**A. Minimum Throughput Requirement**

Let \( q_i \) be a strictly positive real value that represents the minimum throughput requirement of node \( i \). Using the random

\[\text{Authorized licensed use limited to: MIT Libraries. Downloaded on December 17, 2020 at 00:09:48 UTC from IEEE Xplore. Restrictions apply.}\]
variable $d_i^T(k)$, we define the long-term throughput of node $i$ when policy $\pi$ is employed as

$$q_i^\pi := \lim_{K \to \infty} \frac{1}{K} \sum_{k=1}^K \mathbb{E}[d_i^T(k)].$$

(3)

Then, we express the minimum throughput constraint of each individual node as

$$q_i^\pi \geq q_i, \quad \forall i \in \{1, \cdots, M\}.$$  

(4)

In this paper, we assume that $\{q_i\}_{i=1}^M$ is a feasible set of minimum throughput requirements, i.e., there exists a policy $\pi \in \Pi$ that satisfies all $K$ interference constraints in (1) and all $M$ throughput constraints in (4) simultaneously. As shown in [2, Lemma 5], the inequality

$$\sum_{i=1}^M q_i p_i \leq 1,$$

(5)

is a necessary and sufficient condition for the feasibility of $\{q_i\}_{i=1}^M$. Throughout this paper, we assume that (5) is satisfied with strict inequality. Next, we present the AoI metric.

B. Age of Information

The Age of Information depicts how old the information is from the perspective of the BS. Let $h_i(k)$ be the positive integer that represents the AoI associated with node $i$ at the beginning of slot $k$. If the BS does not receive a packet from node $i$ during slot $k$, then $h_i(k+1) = h_i(k) + 1$, since the information at the BS is one slot older. In contrast, if the BS receives a packet from node $i$ during slot $k$, then $h_i(k+1) = 1$, because the received packet was generated at the beginning of slot $k$. The evolution of $h_i(k)$ follows

$$h_i(k+1) = \begin{cases} 1, & \text{if } d_i(k) = 1 \\ h_i(k) + 1, & \text{otherwise}. \end{cases}$$

(6)

The average AoI of node $i$ during the first $K$ slots is captured by $\mathbb{E} \left[ \sum_{k=1}^K h_i(k) \right] / K$, where the expectation is with respect to the randomness in the channel and the scheduling policy. For measuring the freshness of the information of the entire network when policy $\pi$ is employed, we use the Expected Weighted Sum AoI

$$\mathbb{E}[J_i^T] = \frac{1}{KM} \mathbb{E} \left[ \sum_{k=1}^K \sum_{i=1}^M \alpha_i h_i(k) \right].$$

(7)

where $\vec{h}(1) = [h_1(1), \cdots, h_M(1)]^T$ is the vector of initial AoI in (6) and $\alpha_i > 0$ is the weight of node $i$. For simplicity, we assume that $h_i(1) = 1, \forall i$, and omit $\vec{h}(1)$ henceforth.

C. Optimization Problem

With the definitions of AoI and throughput, we present the optimization problem that is central to this paper.

AoI Optimization:

$$\text{OPT}^* = \min_{\pi \in \Pi} \left\{ \lim_{K \to \infty} \frac{1}{KM} \mathbb{E} \left[ \sum_{k=1}^K \sum_{i=1}^M \alpha_i h_i(k) \right] \right\}$$

(8a)

s.t. $\bar{q}_i^\pi \geq q_i, \quad \forall i$; 

$$\sum_{i=1}^M u_i(k) \leq 1, \quad \forall k.$$  

(8b)

(8c)

The minimum throughput constraints are depicted in (8b) and the interference constraints are in (8c). The scheduling policy that results from (8a)-(8c) is referred to as AoI-optimal.

For a given network setup $(M, p_i, q_i, \alpha_i)$, let OPT$^*$ be the Expected Weighted Sum AoI achieved by the AoI-optimal policy $\pi^*$. Similarly, let OPT$^\eta$ be the AoI achieved by some policy $\eta \in \Pi$. The optimality ratio of $\eta$ is given by

$$\psi^\eta = \frac{\text{OPT}^\eta}{\text{OPT}^*},$$

(9)

and we say that policy $\eta$ is $\psi^\eta$-optimal. Naturally, the lower the value of $\psi^\eta$, the better is the AoI performance of policy $\eta$. The lowest $\psi^\eta$ achievable by a policy $\eta$ that satisfies (8b) and (8c) is $\psi^\eta = 1$. In general, if policy $\eta$ does not satisfy (8b) or (8c), then $\psi^\eta$ could be lower than unity. The optimality ratio $\psi^\eta$ is used in the upcoming sections to compare the performance of different scheduling policies.

III. SCHEDULING POLICIES

In this section, we propose four low-complexity scheduling policies with strong AoI performances. The first three provably satisfy the throughput constraints for every feasible set $\{q_i\}_{i=1}^M$ and the fourth accounts for the throughput constraints, but provides no guarantee. To evaluate the AoI performance of each policy, we find their corresponding optimality ratio $\psi^\eta$. Moreover, in Sec. IV, we simulate and compare these policies to the state-of-the-art in the literature.

Prior to introducing the policies, we obtain a lower bound to the AoI optimization (8a)-(8c) which is used in the derivation of the optimality ratios $\psi^\eta$. Then, we present four scheduling policies: 1) Optimal Stationary Randomized policy; 2) Max-Weight policy; 3) Drift-Plus-Penalty policy; and 4) Whittle’s Index policy. The first is obtained by solving the AoI optimization (8a)-(8c) over the class of Stationary Randomized Policies. The second and third policies are derived using Lyapunov Optimization [36]. The fourth policy is obtained by using the Restless Multi-Armed Bandit framework [37].

A. Lower Bound

In this section, we use a sample path argument to derive a lower bound to the AoI optimization (8a)-(8c).

$$L_B = \min_{\pi \in \Pi} \left\{ \frac{1}{2M} \sum_{i=1}^M \alpha_i \left( \frac{1}{q_i^\pi} + 1 \right) \right\}$$

(10a)

s.t. $q_i^\pi \geq q_i, \quad \forall i$;  

$$\sum_{i=1}^M u_i(k) \leq 1, \quad \forall k.$$  

(10b)

(10c)

Proof: Consider a scheduling policy $\pi \in \Pi$ that satisfies all throughput and interference constraints running on a network for the time-horizon of $K$ slots. Let $\Omega$ be the sample space associated with this network and let $\omega \in \Omega$ be a sample path. For a given sample path $\omega$, the total number of packets delivered by node $i$ during the $K$ slots is denoted $D_i(K) = \sum_{k=1}^K d_i(k)$ and the inter-delivery time associated

Authorized licensed use limited to: MIT Libraries. Downloaded on December 17,2020 at 00:09:48 UTC from IEEE Xplore. Restrictions apply.
with each of those deliveries is denoted \( I_i[m] \). In particular, let \( I_i[m] \) be the number of slots between the \((m-1)\)th and \( m \)th packet deliveries from node \( i \), \( \forall m \in \{1, \cdots, D_i(K)\} \). After the last packet delivery from node \( i \), the number of remaining slots is \( R_i \). Hence, the time-horizon can be written as
\[
K = \sum_{m=1}^{D_i(K)} I_i[m] + R_i, \quad \forall i \in \{1, 2, \cdots, M\}.
\] (11)

According to the evolution of \( h_i(k) \) in (6), the slot that follows the \((m-1)\)th packet delivery from node \( i \) has an AoI of \( h_i(k) = 1 \). Since the \( m \)th packet is delivered only after \( I_i[m] \) slots, we know that \( h_i(k) \) evolves as \( \{1, 2, \cdots, I_i[m]\} \). This pattern is repeated throughout the entire time-horizon, including the last \( R_i \) slots. As a result, the time-average Age of Information of node \( i \) can be expressed as
\[
\frac{1}{K} \sum_{k=1}^{K} h_i(k) = \frac{1}{K} \left( \sum_{m=1}^{D_i(K)} \left( \frac{(I_i[m] + 1)I_i[m]}{2} + \frac{(R_i + 1)R_i}{2} \right) \right) = \frac{1}{2} \left( \frac{D_i(K)}{K} \sum_{m=1}^{D_i(K)} I_i^2[m] + \frac{R_i^2}{K} + 1 \right),
\] (12)

where the last equality uses (11) to replace the two linear terms by \( K \).

Define the operator \( \overline{M}[x] \) that computes the sample mean of any set \( x \). In particular, let the sample mean of \( I_i[m] \) and \( I_i^2[m] \) be
\[
\overline{M}[I_i] = \frac{1}{D_i(K)} \sum_{m=1}^{D_i(K)} I_i[m];
\] (13)
\[
\overline{M}[I_i^2] = \frac{1}{D_i(K)} \sum_{m=1}^{D_i(K)} I_i^2[m].
\] (14)

Substituting \( \overline{M}[I_i^2] \) into (12) and then applying Jensen’s inequality, yields
\[
\frac{1}{K} \sum_{k=1}^{K} h_i(k) \geq \frac{1}{2} \left( \frac{D_i(K)}{K} \left( \overline{M}[I_i] \right)^2 + \frac{R_i^2}{K} + 1 \right),
\] (15)
combining (11) into (13) and then substituting the result in (15), gives
\[
\frac{1}{K} \sum_{k=1}^{K} h_i(k) \geq \frac{1}{2} \left( \frac{K - R_i}{K} \frac{1}{D_i(K)} + \frac{R_i^2}{K} + 1 \right).
\] (16)

By minimizing the LHS of (16) analytically with respect to the variable \( R_i \), we have
\[
\frac{1}{K} \sum_{k=1}^{K} h_i(k) \geq \frac{1}{2} \left( \frac{K}{D_i(K) + 1} + 1 \right).
\] (17)

Applying the limit \( K \to \infty \) to (18) and using the definition of throughput in (3), gives
\[
\lim_{K \to \infty} \frac{1}{K} \sum_{k=1}^{K} \mathbb{E}[h_i(k)] \geq \frac{1}{2} \left( \frac{1}{q_i^R} + 1 \right).
\] (19)
Combining (19) and the objective function in (7), yields
\[
\lim_{K \to \infty} \mathbb{E}[J_R^n] = \lim_{K \to \infty} \frac{1}{2M} \sum_{i=1}^{M} \alpha_i \sum_{k=1}^{K} \mathbb{E}[h_i(k)] \geq \frac{1}{2M} \sum_{i=1}^{M} \alpha_i \left( \frac{1}{q_i^R} + 1 \right).
\] (20)

Finally, substituting (20) into the AoI optimization (8a)-(8c) gives the Lower Bound (10a)-(10c).

To obtain the expression in (20), we applied Jensen’s inequality twice and minimized (16) analytically with respect to \( R_i \). Each of those steps could have led to a loose lower bound \( L_B \). However, in the next section, we use this lower bound to obtain a tight optimality ratio, \( \psi^R < 2 \), for a Stationary Randomized policy. Moreover, we evaluate the tightness of \( L_B \) using numerical results in Sec. IV.

B. Optimal Stationary Randomized Policy

Denote by \( \Pi_R \) the class of Stationary Randomized Policies and let \( R \in \Pi_R \) be a scheduling policy that, in each slot \( k \), selects node \( i \) with probability \( \mu_i \in (0, 1) \) and idles with probability \( \mu_{idle} \). Each policy in \( \Pi_R \) is fully characterized by the set of scheduling probabilities \( \{\mu_i\}_{i=1}^{M} \), where \( \mu_i = \mathbb{E}[u_i(k)], \forall i, \forall k \) and \( \mu_{idle} = 1 - \sum_{i=1}^{M} \mu_i \). Next, we find the Optimal Stationary Randomized policy \( R^* \) that solves the AoI optimization (8a)-(8c) over the class \( \Pi_R \subset \Pi \) and derive the associated optimality ratio \( \psi^R \).

Proposition 2: Consider a policy \( R \in \Pi_R \) with scheduling probabilities \( \{\mu_i\}_{i=1}^{M} \). The long-term throughput and the expected time-average AoI of node \( i \) can be expressed as
\[
\hat{q}_i^R = p_i \mu_i;
\] (21)
\[
\lim_{K \to \infty} \frac{1}{K} \sum_{k=1}^{K} \mathbb{E}[h_i(k)] = \frac{1}{p_i \mu_i}.
\] (22)

Proof: In any given slot \( k \), the BS receives a packet from node \( i \) if this node is scheduled and the corresponding packet transmission is successful. The probability of this event is \( p_i \mu_i \). Moreover, the inter-delivery times \( I_i[m] \) of node \( i \) are i.i.d. with \( \mathbb{P}[I_i[m] = n] = p_i \mu_i (1 - p_i \mu_i)^{n-1}, \forall n, i \in \{1, 2, \cdots\} \).

Clearly, under policy \( R \), the sequence of packet deliveries is a renewal process. Thus, we can use renewal theory to derive (21) and (22). In particular, by the definition of

\[\text{Authorized licensed use limited to: MIT Libraries. Downloaded on December 17, 2020 at 00:09:48 UTC from IEEE Xplore. Restrictions apply.}\]
long-term throughput (3) and the expression for the expected time-average AoI of node $i$, we have

$$\hat{q}_i^R = \lim_{K \to \infty} \frac{1}{K} \sum_{k=1}^{K} \mathbb{E}[d_i(k)] \stackrel{(a)}{=} \frac{1}{\mathbb{E}[I[m]]} p_i \mu_i, \quad (23)$$

$$\lim_{K \to \infty} \frac{1}{K} \sum_{k=1}^{K} \mathbb{E}[h_i(k)] \stackrel{(b)}{=} \frac{2 \mathbb{E}[I^2[m]]}{2 \mathbb{E}[I[m]]} + \frac{1}{p_i \mu_i}. \quad (24)$$

where (a) follows from the elementary renewal theorem and (b) from its generalization for renewal-reward processes [38, Sec. 5.7].

Substituting both expressions from Proposition 2 into the AoI optimization (8a)-(8c) gives the equivalent optimization problem over the class $\Pi_R$ presented below.

\textbf{Optimization over Randomized Policies:}

$$\text{OPT}_{R^*} = \min_{R \in \Pi_R} \left\{ \frac{1}{M} \sum_{i=1}^{M} \alpha_i}{p_i \mu_i} \right\} \quad (25a)$$

s.t. $p_i \mu_i \geq q_i$, $\forall i$; \quad (25b)

$$\sum_{i=1}^{M} \mu_i \leq 1. \quad (25c)$$

Notice that under the class $\Pi_R$, conditions (25c) and (8c) are equivalent. The Optimal Stationary Randomized policy $R^*$ is characterized by the set $\{\mu_i^*\}_{i=1}^{M}$ that solves (25a)-(25c).

\textbf{Theorem 3 (Optimality Ratio for $R^*$)}: The optimality ratio of $R^*$ is such that $\psi^R < 2$, namely the Optimal Stationary Randomized policy is 2-optimal for every network setup.

\textbf{Proof:} Let $\hat{q}_i^R$ be the throughput associated with the policy that solves the Lower Bound (10a)-(10c). Consider the policy $R \in \Pi_R$ with long-term throughput $\hat{q}_i^R = p_i \mu_i = \hat{q}_i^L$ for each node $i$. Since $\hat{q}_i^R = \hat{q}_i^L$, it follows that $R$ satisfies all throughput constraints. Comparing $L_B$ in (10a) with the objective function associated with $R$, namely $\text{OPT}_{R^*}$, yields

$$\frac{\text{OPT}_{R^*}}{2} < L_B \rightarrow \psi^R = \text{OPT}_{R^*} = \text{OPT}_{R^*} \leq \frac{\text{OPT}_{R}}{L_B} < 2, \quad (26)$$

where $\text{OPT}^*$ comes from (8a) and $\text{OPT}_{R^*}$ from (25a). Recall that $L_B \leq \text{OPT}^* \leq \text{OPT}_{R^*}$. \quad (26)

\textbf{Corollary 4} The Optimal Stationary Randomized policy $R^*$ is also the solution for the Lower Bound problem (10a)-(10c).

\textbf{Proof:} Using the same argument as in the proof of Theorem 3, in particular $\hat{q}_i^R = p_i \mu_i = \hat{q}_i^L$, it follows that the scheduling policy that solves the Optimization over Randomized policies (25a)-(25c) also solves the Lower Bound (10a)-(10c).

\textbf{Proof:} To find the set of scheduling probabilities $\{\mu_i^*\}_{i=1}^{M}$ that solve the optimization problem (25a)-(25c), we analyze the KKT Conditions. Let $\{\lambda_i^*\}_{i=1}^{M}$ be the KKT multipliers associated with the relaxation of (25b) and $\gamma$ be the multiplier associated with the relaxation of (25c). Then, for $\lambda_i \geq 0, \forall i, \gamma \geq 0$ and $\mu_i \in [q_i/p_i, 1], \forall i$, we define

$$\mathcal{L}(\mu_i, \lambda_i, \gamma) = \frac{1}{M} \sum_{i=1}^{M} \alpha_i}{p_i \mu_i} +$$

$$+ \sum_{i=1}^{M} \lambda_i (q_i - p_i \mu_i) + \gamma \left( \sum_{i=1}^{M} \mu_i - 1 \right), \quad (27)$$

and, otherwise, we define $\mathcal{L}(\mu_i, \lambda_i, \gamma) = +\infty$. Then, the KKT Conditions are (i)

1) Stationarity: $\nabla_{\mu_i} \mathcal{L}(\mu_i, \lambda_i, \gamma) = 0$;
2) Complementary Slackness: $\gamma (\sum_{i=1}^{M} \mu_i - 1) = 0$;
3) Complementary Slackness: $\lambda_i (q_i - p_i \mu_i) = 0, \forall i$;
4) Primal Feasibility: $p_i \mu_i \geq q_i, \forall i$, and $\sum_{i=1}^{M} \mu_i \leq 1$;
5) Dual Feasibility: $\lambda_i \geq 0, \forall i$, and $\gamma \geq 0$.

Since $q_i$ is strictly positive, the function $\mathcal{L}(\mu_i, \lambda_i, \gamma)$ is convex on the interval of interest $\mu_i \in [q_i/p_i, 1]$. Therefore, if there exists a vector $\{\mu_i^*\}_{i=1}^{M}$ $\{\lambda_i^*\}_{i=1}^{M}$ that satisfies all KKT Conditions, then this vector is unique. As a result, the scheduling policy $R^* \in \Pi_R$ that optimizes (25a)-(25c) is also unique and is characterized by $\{\mu_i^*\}_{i=1}^{M}$. Next, we find the vector $\{\mu_i^*\}_{i=1}^{M}$.

To assess stationarity, $\nabla_{\mu_i} \mathcal{L}(\mu_i, \lambda_i, \gamma) = 0$, we calculate the partial derivative of $\mathcal{L}(\mu_i, \lambda_i, \gamma)$ with respect to $\mu_i$. It follows from the derivative that

$$\frac{\alpha_i}{M p_i \mu_i^2} + \lambda_i \mu_i = \gamma, \forall i. \quad (28)$$

From complementary slackness, $\gamma (\sum_{i=1}^{M} \mu_i - 1) = 0$, we know that either $\gamma = 0$ or $\sum_{i=1}^{M} \mu_i = 1$. Equation (28) shows that the value of $\gamma$ can only be zero if $\lambda_i = 0$ and $\mu_i \to \infty$, which violates $\mu_i \in [q_i/p_i, 1]$. Hence, we obtain

$$\gamma > 0 \text{ and } \sum_{i=1}^{M} \mu_i = 1. \quad (29)$$

Notice that $\sum_{i=1}^{M} \mu_i = 1$ implies in $\mu_{idle} = 0$.

Based on dual feasibility, $\lambda_i \geq 0$, we can separate nodes $i \in \{1, \cdots, M\}$ into two categories: nodes with $\lambda_i > 0$ and nodes with $\lambda_i = 0$.

\textbf{Category 1)} node $i$ with $\lambda_i > 0$. It follows from complementary slackness, $\lambda_i (q_i - p_i \mu_i) = 0$, that

$$\mu_i = \frac{q_i}{p_i}. \quad (30)$$

Plugging this value of $\mu_i$ into (28) gives the inequality $\lambda_i \mu_i = \gamma - \gamma_i > 0$, where we defined the constant

$$\gamma_i := \frac{\alpha_i q_i}{M p_i^2}. \quad (31)$$

\textbf{Category 2)} node $i$ with $\lambda_i = 0$. It follows from (28) that

$$\gamma = \gamma_i \left( \frac{q_i}{p_i \mu_i} \right)^2 \rightarrow \mu_i = \frac{q_i}{p_i \sqrt{\gamma_i}}. \quad (32)$$

In summary, for any fixed value of $\gamma > 0$, the scheduling probability of node $i$ is

$$\mu_i = \frac{q_i}{p_i \max \left\{ 1; \sqrt{\gamma} \right\}}. \quad (33)$$

\textbf{Theorem 5 (Optimal Stationary Randomized Policy):} The scheduling probabilities $\{\mu_i^*\}_{i=1}^{M}$ that result from Algorithm 1 are the unique solution to (25a)-(25c) and, thus, characterize the Optimal Stationary Randomized policy $R^*$.

Notice that for a decreasing value of $\gamma$, the probability $\mu_i$ remains fixed or increases. Our goal is to find the value of $\gamma^*$ that gives $\{\mu_i^*\}_{i=1}^{M}$ satisfying the condition $\sum_{i=1}^{M} \mu_i^* = 1$. \quad (33)
Algorithm 1 Unique Solution to KKT Conditions

1: \( \gamma_i \leftarrow \alpha_i p_i / M q_i^2, \forall i \in \{1, 2, \cdots, M\} \)
2: \( \gamma \leftarrow \max_i \{\gamma_i\} \)
3: \( \mu_i \leftarrow (q_i / p_i) \max\{1 : \sqrt{\gamma_i / \gamma}\}, \forall i \)
4: \( S \leftarrow \mu_1 + \mu_2 + \cdots + \mu_M \)
5: while \( S < 1 \) do
6: \( \text{decrease } \gamma \) slightly
7: \( \text{repeat steps 3 and 4 to update } \mu_i \) and \( S \)
8: end while
9: \( \mu_i^* = \mu_i, \forall i, \) and \( \gamma^* = \gamma \)
10: return \((\mu_1^*, \mu_2^*, \cdots, \mu_M^*, \gamma^*)\)

Proposed algorithm to find \( \gamma^* \): start with \( \gamma = \max\{\gamma_i\} \). Then, according to (33), all nodes have \( \mu_i = q_i / p_i \) and, by the feasibility condition in (5), it follows that

\[
\sum_{i=1}^{M} \mu_i = \sum_{i=1}^{M} \frac{q_i}{p_i} \leq 1.
\]

Now, by gradually decreasing \( \gamma \) and adjusting \((\mu_i)_{i=1}^{M}\) according to (33), we can find the unique \( \gamma^* \) that fulfills \( \sum_{i=1}^{M} \mu_i^* = 1 \). The solution \( \gamma^* \) exists since \( \gamma \to 0 \) implies in \( \sum_{i=1}^{M} \mu_i \to \infty \). The uniqueness of \( \gamma^* \) follows from the monotonicity of \( \mu_i \) with respect to \( \gamma \). This process is described in Algorithm 1 and illustrated in Fig. 1.

Algorithm 1 outputs the set of scheduling probabilities \((\mu_i^*)_{i=1}^{M}\) and the parameter \( \gamma^* \). The set \((\lambda_i^*)_{i=1}^{M}\) is obtained using (28). Hence, the unique vector \((\mu_i^*)_{i=1}^{M}, (\lambda_i^*)_{i=1}^{M}, \gamma^*)\) that solves the KKT Conditions is found.

In order to fulfill the throughput constraints (25b), every scheduling policy in \( \Pi_R \) must allocate at least \( \mu_i \geq q_i / p_i \) to each node \( i \). What differentiates policies in \( \Pi_R \) is how they distribute the remaining resources, \( 1 - \sum_{i=1}^{M} q_i / p_i \), between nodes. According to Algorithm 1, the Optimal Stationary Randomized policy \( R^* \) supplies additional resources, \( \mu_i^* > q_i / p_i \), to nodes with high value of \( \gamma_i \), namely nodes with a high priority \( \alpha_i \) or a low value of \( q_i / p_i \). Notice that if a node with low \( q_i / p_i \) was given the minimum required amount of resources, it would rarely transmit and its AoI would be high. In contrast, policy \( R^* \) allocates the minimum required, \( \mu_i^* = q_i / p_i \), to nodes with low priority \( \alpha_i \) or high \( q_i / p_i \).

The policies \( R \in \Pi_R \) discussed in this section are as simple as possible. They select nodes randomly, according to fixed scheduling probabilities \((\mu_i)_{i=1}^{M}\) calculated offline by Algorithm 1. Despite their simplicity, it was shown that \( R^* \) is 2-optimal regardless of the network setup \((M, p_i, q_i, \alpha_i)\).

In the following sections, we develop scheduling policies that take advantage of additional information, such as the current AoI of each node, for selecting nodes in an adaptive manner.

C. Max-Weight Policy

Using techniques from Lyapunov Optimization [36, chapter 3], we derive the Max-Weight policy associated with the AoI optimization (8a)-(8c). Max-Weight is a scheduling policy designed to reduce the expected increase in the Lyapunov Function. The Lyapunov Function outputs a positive scalar that is large when the network is in undesirable states, namely when nodes have high AoI or less throughput than the minimum required \( q_i \). Intuitively, the Max-Weight policy keeps the network in desirable states by controlling the growth of the Lyapunov Function. Prior to presenting the Max-Weight policy, we introduce the notions of throughput debt, network state, Lyapunov Function and Lyapunov Drift.

Let \( x_i(k) \) be the throughput debt associated with node \( i \) at the beginning of slot \( k \). The throughput debt evolves as

\[
x_i(k+1) = k q_i - \sum_{t=1}^{k} d_i(t).
\]

The value of \( k q_i \) can be interpreted as the minimum number of packets that node \( i \) should have delivered by slot \( k+1 \) and \( \sum_{t=1}^{k} d_i(t) \) is the total number of packets actually delivered in the same interval. Define the operator \((.)^+ = \max\{(\cdot), 0\}\) that computes the positive part of a scalar. Then, the positive part of the throughput debt is given by \( x_i^+(k) = \max(x_i(k); 0) \). A large debt \( x_i^+(k) \) indicates to the scheduling policy \( \pi \in \Pi \) that node \( i \) is lagging behind in terms of throughput. In fact, strong stability of the process \( x_i^+(k) \), namely

\[
\lim_{K \to \infty} \frac{1}{K} \sum_{k=1}^{K} \mathbb{E}[x_i^+(k)] < \infty,
\]

is sufficient to establish that the minimum throughput constraint, \( \dot{q}^* \geq q_i \), is satisfied [36, Theorem 2.8].

Denote by \( S_k = (h_i(k), x_i^+(k))_{i=1}^{M} \) the network state at the beginning of slot \( k \) and define the Lyapunov Function by

\[
L(S_k) := \frac{1}{2} \sum_{i=1}^{M} \left( \alpha_i h_i^2(k) + V \left[ x_i^+(k) \right]^2 \right),
\]

where \( V \) is a strictly positive real value that depicts the importance of the throughput constraints. Observe that \( L(S_k) \) is large when nodes have high AoI or high throughput debt. To measure the expected change in the Lyapunov Function from one slot to the next, we define the Lyapunov Drift

\[
\Delta(S_k) := \mathbb{E} \left\{ L(S_{k+1}) - L(S_k) \right\} S_k.
\]

The Max-Weight policy is designed to keep \( L(S_k) \) small by reducing \( \Delta(S_k) \) in every slot \( k \). Next, we present an upper bound on \( \Delta(S_k) \) that can be readily used to design the Max-Weight policy. The derivation of this upper bound is centered around the evolution of \( h_i(k) \) in (6) and the evolution of \( x_i^+(k) \)
The complete derivation can be found in Appendix A and the upper bound follows

\[
\Delta(S_k) \leq - \sum_{i=1}^{M} \mathbb{E} \{ u_i(k) \mid S_k \} W_i(k) + B(k),
\]

where \( W_i(k) \) and \( B(k) \) are given by

\[
W_i(k) = \frac{\alpha_i p_i}{2} h_i(k)[h_i(k) + 2] + V p_i x^+_i(k); \quad (40)
\]

\[
B(k) = \sum_{i=1}^{M} \left\{ \alpha_i \left[ h_i(k) + \frac{1}{2} \right] + V \left[ x^+_i(k) q_i + \frac{1}{2} \right] \right\}. \quad (41)
\]

Both \( W_i(k) \) and \( B(k) \) are fully characterized by the network state \( S_k \) and network setup \((M, p_i, q_i, \alpha_i)\). Hence, both can be used by admissible policies for making scheduling decisions. However, notice that the term \( B(k) \) in (39) is not affected by the choice of \( u_i(k) \). Thus, for minimizing the upper bound in (39), the Max-Weight policy selects, in each slot \( k \), the node with highest value of \( W_i(k) \), with ties being broken arbitrarily. Denote the Max-Weight policy as \( MW \).

**Theorem 6:** The Max-Weight policy satisfies any feasible set of minimum throughput requirements \( \{q_i\}_{i=1}^{M} \).

**Theorem 7 (Optimality Ratio for MW):** For any given network setup \((M, p_i, q_i, \alpha_i)\), the optimality ratio of \( MW \) is such that

\[
\psi^{MW} \leq 4 + \frac{1}{L_B} \left[ V - \frac{2}{M} \sum_{i=1}^{M} \alpha_i \right]. \quad (42)
\]

In particular, for every network with \( V \leq 2 \sum_{i=1}^{M} \alpha_i / M \), the Max-Weight policy is 4-optimal.

The proofs of Theorems 6 and 7 are provided in Appendices B and C, respectively. Both theorems follow from the analysis of the expression in (39).

Recall that the Optimal Stationary Randomized policy \( R^* \) selects nodes randomly, according to fixed scheduling probabilities \( \{\mu_i\}_{i=1}^{M} \). In contrast, the Max-Weight policy \( MW \) uses feedback from the network, namely \( h_i(k) \) and \( x^+_i(k) \), to guide scheduling decisions. Despite the added complexity, we expect the feedback loop to improve the performance of \( MW \). In fact, numerical results in Sec. IV demonstrate that \( MW \) outperforms \( R^* \) in every network configuration simulated. However, by comparing Theorems 3 and 7, it might seem that \( R^* \) yields a better performance than \( MW \). This is because the analysis associated with \( MW \) is more challenging, leading to an optimality ratio \( \psi^{MW} \) that is less tight than \( \psi^{R^*} \). Next, we develop a policy called Drift-Plus-Penalty policy and show that it is 2-optimal.

### D. Drift-Plus-Penalty Policy

The Drift-Plus-Penalty policy is derived using a similar technique as the Max-Weight policy. The main difference between these two policies is that the Drift-Plus-Penalty is designed to reduce the sum of the Lyapunov Drift and a Penalty Function, while the Max-Weight policy reduces only the Lyapunov Drift. As we will see, this difference will improve the optimality ratio of the Drift-Plus-Penalty policy significantly.

Based on the AoI minimization (8a), we define the Penalty Function as follows

\[
P'(S_k) := \frac{1}{2} \sum_{i=1}^{M} \beta_i \mathbb{E}[h_i(k+1) \mid S_k], \quad (43)
\]

where \( S_k = (h_i(k), x^+_i(k))_{i=1}^{M} \) is the network state at the beginning of slot \( k \) and \( \beta_i \) is a positive real value associated with node \( i \). Observe that \( P'(S_k) \) is large when nodes have high AoI. Similarly to (38), we define the Lyapunov Drift as

\[
\Delta'(S_k) := \mathbb{E} \{ L'(S_{k+1}) - L'(S_k) \mid S_k \}, \quad (44)
\]

with associated Lyapunov Function

\[
L'(S_k) := \frac{V'}{2} \sum_{i=1}^{M} [x^+_i(k)]^2 , \quad (45)
\]

where \( V' \) is a strictly positive real value that represents the importance of the throughput constraints. Notice that, as opposed to the Lyapunov Function in (37), the expression in (45) does not contain the term with \( h_i(k) \). This is because the AoI term is already present in the Penalty Function.

The Drift-Plus-Penalty policy is designed to minimize an upper bound on \( \Delta'(S_k) + P'(S_k) \) at every slot \( k \). The upper bound is derived in Appendix D of the supplementary material by manipulating (43)-(45). The expression for the upper bound follows

\[
\Delta'(S_k) + P'(S_k) \leq - \sum_{i=1}^{M} \mathbb{E} \{ u_i(k) \mid S_k \} W'_i(k) + B'(k),
\]

where \( W'_i(k) \) and \( B'(k) \) are given by

\[
W'_i(k) = \frac{\beta_i p_i}{2} h_i(k) + V' p_i x^+_i(k); \quad (47)
\]

\[
B'(k) = \sum_{i=1}^{M} \left\{ \frac{\beta_i}{2} [h_i(k) + 1] + V' x^+_i(k) q_i + \frac{V'}{2} \right\}. \quad (48)
\]

The values of \( W'_i(k) \) and \( B'(k) \) can be easily calculated by any admissible policy and thus can be used for making scheduling decisions. For minimizing the upper bound in (46), the Drift-Plus-Penalty policy selects, in each slot \( k \), the node with highest value of \( W'_i(k) \), with ties being broken arbitrarily. Denote the Drift-Plus-Penalty policy as \( DPP \).

**Theorem 8:** The DPP policy satisfies any feasible set of minimum throughput requirements \( \{q_i\}_{i=1}^{M} \).

**Theorem 9 (Optimality Ratio for DPP):** For any given network setup \((M, p_i, q_i, \alpha_i)\), by choosing the constant \( \beta_i = \alpha_i / \mu_i p_i \), the optimality ratio of \( DPP \) is such that

\[
\psi^{DPP} \leq 2 + \frac{1}{L_B} \left[ V' - \frac{1}{M} \sum_{i=1}^{M} \alpha_i \right]. \quad (49)
\]

In particular, for every network with \( V' \leq \sum_{i=1}^{M} \alpha_i / M \), the Drift-Plus-Penalty policy is 2-optimal.
The proofs of Theorems 8 and 9 are provided in Appendices E and F of the supplementary material, respectively. The Lyapunov Function (45) with a quadratic term in $x_i^+(k)$ has a central role in showing that the DPP policy satisfies any feasible requirements $\{q_i\}_{i=1}^M$. Notice that this is also true for MW. The Penalty Function (43) with a linear term in $h_i(k)$ is central to show that the DPP policy is 2-optimal. Recall that the MW policy with a quadratic $h_i(k)$ was shown to be 4-optimal. Comparing Theorems 7 and 9, we can clearly see this improvement in the optimality ratio. However, this improvement is limited to the mathematical analysis. Numerical results in Sec. IV suggest that DPP and MW have similar performances. Next, we develop an index policy based on Whittle’s Index [37] that is surprisingly similar to MW.

E. Whittle’s Index Policy

To find Whittle’s Index, we transform the AoI optimization (8a)-(8c) into a relaxed Restless Multi-Armed Bandit (RMAB) problem. This is possible because every node in the network evolves as a restless bandit. To obtain the relaxed RMAB problem, we first substitute the $K$ interference constraints in (8c) by the single time-averaged constraint

$$\frac{1}{K} \sum_{k=1}^{K} \sum_{i=1}^{M} E[u_i(k)] \leq 1. \quad (50)$$

Next, we relax this time-averaged constraint by placing (50) into the objective function (8a) together with the associated Lagrange Multiplier $C \geq 0$. The resulting optimization problem is called relaxed RMAB and its solution lays the foundation for the design of Whittle’s Index. A detailed description of this method can be found in [37], [39].

One of the challenges associated with this method is that Whittle’s Index is only defined for problems that are indexable. Unfortunately, it can be shown that due to the throughput constraints, $q_i^*_t \geq q_i$, the relaxed RMAB resulting from the transformation of the AoI optimization is not indexable. To overcome this, we relax the throughput constraints (8b), placing them into the objective function of (8a)-(8c) as follows

**Relaxed AoI Optimization:**

$$\hat{\text{OPT}}_D = \min_{\pi \in \Pi} \left\{ \lim_{K \to \infty} \frac{1}{K} \sum_{k=1}^{K} \sum_{i=1}^{M} \left[ \alpha_i E[h_i(k)] + \theta_i \left( \frac{q_i}{p_i} - E[u_i(k)] \right) \right] \right\} \quad (51a)$$

s.t. $\theta_i \geq 0, \ \forall i; \quad (51b)$

$$\sum_{i=1}^{M} u_i(k) \leq 1, \ \forall k. \quad (51c)$$

Each Lagrange Multiplier $\theta_i$ is associated with a relaxation of $q_i^* \geq q_i$. These multipliers are called throughput incentives for they represent the penalty incurred by scheduling policies that deviate from the corresponding throughput constraint. Applying the transformation described at the beginning of this section to the relaxed AoI optimization (51a)-(51c) yields

**Doubly relaxed RMAB:**

$$\hat{\text{OPT}}_{D} = \min_{\pi \in \Pi} \left\{ \lim_{K \to \infty} \frac{1}{K} \sum_{k=1}^{K} \sum_{i=1}^{M} \left[ \alpha_i E[h_i(k)] + \left( C - \theta_i \right) E[u_i(k)] - \frac{C}{M} + \theta_i q_i \right] \right\} \quad (52a)$$

s.t. $\theta_i \geq 0, \ \forall i; \quad (52b)$

$$C \geq 0. \quad (52c)$$

Next, we solve the doubly relaxed RMAB, establish that the relaxed AoI optimization is indexable and obtain a closed-form expression for the Whittle’s Index.

The doubly relaxed RMAB is separable and thus can be solved for each individual node. Observe that a scheduling policy running on a network with a single node $i$ can only choose between selecting node $i$ for transmission during slot $k$ or idling. The scheduling policy that optimizes (52a)-(52c) for a given node $i$ is characterized next.

**Proposition 10 (Threshold Policy): Consider the doubly relaxed RMAB problem (52a)-(52c) associated with a single node $i$. The optimal scheduling policy is a Threshold policy that, in each slot $k$, selects node $i$ when $h_i(k) \geq H_i$ and idles when $1 \leq h_i(k) < H_i$. For positive fixed values of $C$ and $\theta_i$, if $C > \theta_i$, the expression for the threshold is**

$$H_i = \left[ \frac{3}{2} - \frac{1}{p_i} + \sqrt{\left( \frac{1}{p_i} - \frac{1}{2} \right)^2 + \frac{2(C - \theta_i)}{p_i \alpha_i}} \right]. \quad (53)$$

**Otherwise, if $C \leq \theta_i$, the threshold is $H_i = 1$.**

Proposition 10 follows from [25, Proposition 4]. Next, we define the condition for indexability and establish that the relaxed AoI optimization is indexable. For a given value of $C$, let $I_i(C) = \{h_i(k) \in \mathbb{N} | h_i(k) < H_i \}$ be the set of states $h_i(k)$ in which the Threshold policy idles. The doubly relaxed RMAB associated with node $i$ is indexable if the set $I_i(C)$ increases monotonically from $\emptyset$ to $\mathbb{N}$, as the value of $C$ increases from $0$ to $+\infty$. Furthermore, the relaxed AoI optimization is indexable if this condition holds for all nodes. The condition on $I_i(C)$ follows directly from Proposition 10 and is true for all nodes $i$. Thus, we establish that the relaxed AoI optimization (51a)-(51c) is indexable.

Given indexability, we define Whittle’s Index. Let $C_i(h_i(k))$ be the Whittle’s Index associated with node $i$ in state $h_i(k)$. By definition, $C_i(h_i(k))$ is the infimum value of $C$ that makes both scheduling decisions (transmit or idle) equally desirable to the Threshold policy while in state $h_i(k)$. The scheduling decisions are equally desirable when the multiplier $C$ is such that $H_i = h_i(k) + 1$. Using (53) to solve this equation for the value of $C$ gives the following expression for the Index

$$C_i(h_i(k)) = \frac{1}{2} \alpha_i p_i h_i(k) \left( h_i(k) + \frac{2}{p_i} - 1 \right) + \theta_i. \quad (54)$$

After establishing indexability and obtaining the expression for $C_i(h_i(k))$, we define Whittle’s Index policy. The Whittle’s Index policy selects, in each slot $k$, the node with highest value of $C_i(h_i(k))$, with ties being broken arbitrarily. Denote the Whittle’s Index policy as $WI$. 

Authorized licensed use limited to: MIT Libraries. Downloaded on December 17,2020 at 00:09:48 UTC from IEEE Xplore. Restrictions apply.
Theorem 11 (Optimality Ratio for $W_I$): For any given network setup $(M, p_i, q_i, \alpha_i)$, the optimality ratio of $W_I$ is such that

$$\psi_{WI} \leq 8 + \frac{1}{L_D} \left[ \sum_{i=1}^{M} \theta_i - \frac{7}{2M} \sum_{i=1}^{M} \alpha_i \right].$$  \hspace{1cm} (55)$$

In particular, for every network with $\sum_{i=1}^{M} \theta_i \leq 7 \sum_{i=1}^{M} \alpha_i/2$, the Whittle’s Index policy is 8-optimal.

The proof of Theorem 11 is provided in Appendix G of the supplementary material. The arguments used for deriving $\psi_{WI}$ are analogous to the ones for deriving $\psi_{MW}$ in Theorem 7.

Those similarities come from the fact that policies $MW$ and $WI$ are almost identical. Comparing the expressions for $W_i(k)$ and $C_i(h_i(k))$, in (40) and (54), respectively, we can see that both have the term $\alpha_p h_i^2(k)/2$ and both have an isolated throughput term: $W_i(k) = V p_i x_i^k + C_i(h_i(k))$. Naturally, we expect the performance of both policies to be similar in terms of AoI. The key difference between $MW$ and $WI$ lies in the throughput term. While the term $V p_i x_i^k$ guarantees that $MW$ satisfies the throughput constraint, $q_i^* \geq q_i$, the positive scalar $\theta_i$ represents an incentive for $WI$ to comply with the constraint, but provides no guarantee.

The benefit of using a fixed $\theta_i$ is that there is no need to keep track of $x_i^k(k)$ for each node and at every slot $k$.

The results in this section hold for any given set of positive throughput incentives $\{\theta_i\}_{i=1}^M$. Next, we propose an algorithm that finds the values of $\theta_i$ which maximize a lower bound on the Lagrange Dual problem associated with the relaxed AoI optimization (51a)-(51c). Observe that $OPT_D$ in (52a) is the Lagrange Dual function associated with (51a)-(51c). Thus, we can define the Lagrange Dual problem as $\max_{\chi_i,\theta_i} \{OPT_D\}$ subject to $C \geq 0$ and $\theta_i \geq 0, \forall i$. Since this dual problem is challenging to address, we consider a lower bound:

$$\max_{C,\chi_i} \{\tilde{L}(C, \chi_i)\} \leq \max_{C,\theta_i} \{\tilde{OPT}_D\} \leq \text{OPT}^*,$$  \hspace{1cm} (56)

subject to $\chi_i = C - \theta_i$, $C \geq 0$ and $\theta_i \geq 0$ for all nodes $i$, where

$$\tilde{L}(C, \chi_i) = \frac{1}{M} \sum_{i=1}^{M} \frac{\alpha_i}{p_i} - \frac{C}{M} \left[ 1 - \sum_{i=1}^{M} \frac{q_i}{p_i} \right]$$

$$+ \sum_{i=1}^{M} \frac{\alpha_i}{M} \left[ \frac{2\chi_i}{\alpha p_i} + \frac{1}{p_i} \right] - \frac{\chi_i q_i}{\alpha p_i}$$

$$- \frac{1}{p_i} \frac{1}{2}.$$  \hspace{1cm} (57)

The throughput incentives $\theta_i$ that result from the maximization of $\tilde{L}(C, \chi_i)$ are given by Algorithm 2. They are used in the next section to simulate the Whittle’s Index policy. Simulation results show that the values of $\{\theta_i\}_{i=1}^M$ from Algorithm 2 reduce the throughput debt when compared to $\theta_i = 0$.

Algorithm 2 Throughput Incentives

1. $\chi_i \leftarrow \alpha_i p_i [1/(q_i)^2 - (1/p_i - 1/2)]/2, \forall i$
2. $C \leftarrow \max \{\chi_i\}$
3. $\phi^*_{\chi_i} \leftarrow p_i \sqrt{2} \min \{\chi_i/\alpha p_i + (1/p_i - 1/2)^2, \forall i\}$
4. $S \leftarrow \phi_1 + \phi_2 + \cdots + \phi_M$
5. while $S < 1$ do
6. decrease $C$ slightly
7. repeat steps 3 and 4 to update $\phi_i$ and $S$
8. end while
9. $C^* = C$ and $\chi_i = \min \{C^*: \chi_i\}$ and $\theta_i = C^* - \chi_i, \forall i$
10. return $\{\theta_i, \theta_i^*, \theta_i^*\}_{i=1}^M$

IV. Simulation Results

In this section, we simulate six transmission scheduling policies: 1) Optimal Randomized, $R^*$; 2) Max-Weight, $MW$; 3) Drift-Plus-Penalty, $DPP$; 4) Whittle’s Index with $\theta_i$, $WI$; 5) Whittle’s Index with $\theta_i = 0$, $WP$; and 6) Largest Weighted-Debt First, $LD$. The first four policies are developed in Sec. III and the last two are proposed in [25] and [2], respectively. Policy $WP$ was proposed in [25] for minimizing the AoI in broadcast wireless networks without throughput requirements. It is analogous to $WI$ but with $\theta_i = 0, \forall i$. Policy $LD$ selects, in each slot $k$, the node with highest value of $x_i(k)/p_i$, where $x_i(k)$ is the throughput debt (35). It was shown in [2] that $LD$ satisfies any set of feasible throughput requirements $\{q_i\}_{i=1}^M$. Notice that $LD$ does not account for AoI.

We simulate a network with $M$ nodes, each having different parameters. Node $i$ has weight $\alpha_i = (M + 1 - i)/M$, channel reliability $p_i = 1/M$ and minimum throughput requirement $q_i = c_i p_i/M$, where $\epsilon \in [0, 1)$ represents the hardness of satisfying the throughput constraints $\hat{q}_i^* \geq q_i$. The larger the value of $\epsilon$, the more challenging are the constraints. Notice that $\epsilon < 1$ is necessary for the feasibility of $\{q_i\}_{i=1}^M$.

The values of $V$ and $V'$ represent the importance of the throughput constraints for $MW$ and $DPP$, respectively. A lower value of $V$ (or $V'$) reduces the priority of the throughput debt and increases the priority of AoI minimization. Recall that for any positive $V$ and $V'$, both $MW$ and $DPP$ are guaranteed to satisfy any feasible throughput requirements in the long run. Policies $R^*$, $WI$, $WP$ and $LD$ are not affected by $V$ nor $V'$.

Two performance metrics are used to evaluate scheduling policies. Figs. 2, 4 and 6 measure the Expected Weighted Sum AoI, $E[J_{PK}]$, defined in (7) and compare it with the lower bound $L_B$ in (10a). Figs. 3 and 5 measure the maximum normalized throughput debt, defined as $\max_i \{x_i^k(K + 1)/K q_i\}$. Figs. 2 and 3 display the evolution over time, for $K \in \{5 \times 10^4, 5 \times 10^4, 5 \times 10^5, 5 \times 10^6, 10^6\}$, of a network with $M = 15$, $\epsilon = 0.9$ and $V = V' = 1$. Each data point in Figs. 2 and 3 is an average over the results of $10^5/K$ simulations. Figs. 4 and 5 show simulations of networks with different sizes, namely $M \in \{5, 10, \ldots, 25, 30\}$, and fixed $\epsilon = 0.9$ and $V = V' = M^2$. Fig. 6 shows networks with varying throughput constraints, namely $\epsilon \in \{0.7, 0.75, 0.85, 0.95, 0.999\}$, and fixed $M = 30$ and $V = V' = M^2$. Each data point in Figs. 4, 5 and 6 is an average over the results of 10 simulations and each simulation runs for a total of $K = M \times 10^6$ slots. Figs. 2 and 3 show the effects of low $V$ and $V'$ on $MW$ and $DPP$. A lower value of $V$ (or $V'$) gives lower priority.
to the throughput debt and, as a result, the network may take longer to achieve the desired throughput, especially when the number of nodes $M$ and/or $\varepsilon$ are large. This convergence time is illustrated in Fig. 3. The advantage of having low $V$ is the (slight) improvement in EWSAoI. Comparing $MW$ and $DPP$ in Figs. 2 and 4, it can be seen that when $V = V'$ goes from 15 to 1, the EWSAoI of $MW$ decreases from 16.93 to 16.50 and $DPP$ decreases from 17.26 to 16.61, i.e. less than 5% improvement when $V$ and $V'$ are decreased by a factor of 225.

Simulations clearly support the theoretical results and discussions in Sec. III. Specifically, i) Figs. 2, 4 and 6 show that the performance of $R^*$ is a factor of 2 away from the lower bound, while the performance of $MW$, $DPP$, $WI$ and $WP$ are comparable to the lower bound in every network configuration simulated; and ii) Figs. 3 and 5 show that by changing the throughput incentive from $\theta = 0$ ($WP$) to the optimal $\theta^*$ ($WI$), the throughput debt is reduced but is still not zero. Hence, as expected, $WI$ and $WP$ are not guaranteed to satisfy the throughput requirements. From (i) and (ii) we conclude that $MW$ and $DPP$ have superior performance in terms of both AoI and throughput.

V. CONCLUDING REMARKS

In this paper, we considered a single-hop wireless network with a number of nodes transmitting time-sensitive information to a base station over unreliable channels. We addressed the problem of minimizing the Expected Weighted Sum AoI of the network while satisfying minimum throughput requirements from the individual nodes. Four low-complexity scheduling policies were developed: Optimal Stationary Randomized policy, Max-Weight policy, Drift-Plus-Penalty policy, and Whittle’s Index policy. The performance of each policy was evaluated both analytically and through simulation. The Max-Weight policy and the Drift-Plus-Penalty policy demonstrated the best performance in terms of both AoI and throughput. Interesting related models that can be analyzed using similar techniques include single-hop networks in which i) packets arrive to each node according to a stochastic process and are enqueued in separated (per node) queues; or ii) some
links can be activated simultaneously and scheduling decisions are for subsets of links.

**APPENDIX A**

**UPPER BOUND ON THE LYAPUNOV DRIFT OF MW**

In this appendix, we obtain the expressions in (39)-(41), which represent an upper bound on the Lyapunov Drift. Consider the network state \( S_k = (h_i(k), x_i^+(k))_{i=1}^M \), the Lyapunov Function \( L(S_k) \) in (37) and the Lyapunov Drift \( \Delta(S_k) \) in (38). Substituting (37) into (38), we get

\[
\Delta(S_k) = \frac{1}{2} \sum_{i=1}^M \alpha_i \mathbb{E} \left\{ h_i^2(k + 1) - h_i^2(k) \mid S_k \right\} + \frac{V}{2} \sum_{i=1}^M \mathbb{E} \left\{ [x_i^+(k + 1)]^2 - [x_i^+(k)]^2 \mid S_k \right\}.
\]

(58)

Next, we find expressions for \([x_i^+(k + 1)]^2 - [x_i^+(k)]^2\) and \(h_i^2(k + 1) - h_i^2(k)\) which are then substituted into (58).

To obtain the expression associated with the throughput debt, we use the following recursion

\[
x_i(k + 1) = x_i(k) - d_i(k) + q_i, \quad \forall k,
\]

(59)

with \(x_i(1) = 0\). Notice that (59) is equivalent to (35). Squaring \(x_i^+(k + 1)\), yields

\[
[x_i^+(k + 1)]^2 = \left[ \max\{x_i(k) - d_i(k) + q_i; 0\} \right]^2 \\
\leq \left[ \max\{x_i^+(k) - d_i(k) + q_i; 0\} \right]^2 \\
\leq [x_i^+(k) - d_i(k) + q_i]^2.
\]

(60)

Manipulating (60), gives

\[
[x_i^+(k + 1)]^2 - [x_i^+(k)]^2 \leq -2x_i^+(k)[d_i(k) - q_i] + 1.
\]

(61)

Finally, by taking the conditional expectation of (61) and applying (2), we get the upper bound

\[
\mathbb{E} \left\{ [x_i^+(k + 1)]^2 - [x_i^+(k)]^2 \mid S_k \right\} \\
\leq -2x_i^+(k) \left( p_i \mathbb{E} \{u_i(k) \mid S_k\} - q_i \right) + 1.
\]

(62)

To obtain the expression associated with the Aol, we calculate \(\mathbb{E} \{h_i^2(k + 1) \mid S_k\}\) using the evolution of \(h_i(k)\) in (6). It follows that

\[
\mathbb{E} \{ h_i(k + 1)^2 \mid S_k \} = p_i \mathbb{E} \{u_i(k) \mid S_k\} + (h_i(k) + 1)^2 (1 - p_i \mathbb{E} \{u_i(k) \mid S_k\}),
\]

(63)

Manipulating (63), we get

\[
\mathbb{E} \{ h_i(k + 1)^2 - h_i(k)^2 \mid S_k \} \\
= -p_i \mathbb{E} \{u_i(k) \mid S_k\} h_i(k) [h_i(k) + 2] + 2h_i(k) + 1.
\]

(64)

Substituting (62) and (64) into the Lyapunov Drift in (58), yields the expressions in (39)-(41).

**APPENDIX B**

**PROOF OF THEOREM 6**

**Theorem 6.** The Max-Weight policy satisfies any feasible set of minimum throughput requirements \( \{q_i\}_{i=1}^M \).

**Proof:** The expression for the Lyapunov Drift (39) is central to the analysis in this appendix and is rewritten below for convenience.

\[
\Delta(S_k) \leq - \sum_{i=1}^M \mathbb{E} \{u_i(k) \mid S_k\} W_i(k) + B(k),
\]

where \(W_i(k)\) and \(B(k)\) are given by

\[
W_i(k) = \frac{\alpha_i p_i}{2} h_i(k) [h_i(k) + 2] + V p_i x_i^+(k);
\]

\[
B(k) = \sum_{i=1}^M \left\{ \alpha_i \left( h_i(k) + \frac{1}{2} \right) + V \left( x_i^+(k) q_i + \frac{1}{2} \right) \right\}.
\]

Recall that the Max-Weight policy minimizes the RHS of (39) by selecting \(i = \arg \max \{W_i(k)\}\) in every slot \(k\). Hence, any other policy \(\pi \in \Pi\) yields a lower (or equal) RHS. Consider a Stationary Randomized Policy \(R \in \Pi_R\) that, in each slot \(k\), selects node \(i\) with probability \(\mu_i \in (0, 1]\). Then, it follows that

\[
\sum_{i=1}^M \mathbb{E} \{u_i(k) \mid S_k\} W_i(k) \geq \sum_{i=1}^M \mu_i W_i(k).
\]

(65)

Substituting (65) into the equation of the Lyapunov Drift gives

\[
\Delta(S_k) \leq - \sum_{i=1}^M \mu_i W_i(k) + B(k)
\]

\[
\leq - \sum_{i=1}^M \frac{\alpha_i p_i \mu_i}{2} \left[ h_i(k) - \frac{1}{p_i \mu_i} + 1 \right]^2 + \sum_{i=1}^M \frac{\alpha_i}{2 p_i \mu_i} + \n
+ \frac{VM}{2} - V \sum_{i=1}^M (\mu_i p_i - q_i) x_i^+(k).
\]

(66)

Consider the Cauchy-Schwarz inequality

\[
\left\{ \sum_{i=1}^M \alpha_i p_i \mu_i \left[ h_i(k) - \frac{1}{p_i \mu_i} + 1 \right] \right\} \left\{ \sum_{i=1}^M \alpha_i \right\}
\geq \left\{ \sum_{i=1}^M \alpha_i \left[ h_i(k) - \frac{1}{p_i \mu_i} + 1 \right] \right\}^2.\]

(67)

Applying this inequality to (66) yields

\[
\Delta(S_k) \leq \sum_{i=1}^M \alpha_i \left[ 2 p_i \mu_i - V \mu_i p_i - q_i x_i^+(k) \right] + \frac{VM}{2} - \frac{1}{2} \left\{ \sum_{i=1}^M \frac{\alpha_i}{p_i \mu_i} \right\}^{-1}
\times \left\{ \sum_{i=1}^M \alpha_i \left[ h_i(k) - \frac{1}{p_i \mu_i} + 1 \right]^2 \right\}.
\]

(68)
and rearranging the terms

\[
\left\{ \sum_{i=1}^{M} \frac{2V \alpha_i}{p_i \mu_i} \right\} \left\{ \sum_{i=1}^{M} (\mu_i p_i - q_i) x_i^+(k) \right\} + \left\{ \sum_{i=1}^{M} \alpha_i \right\} \left\{ \sum_{i=1}^{M} \frac{\alpha_i}{p_i \mu_i} + VM \right\}.
\]

For simplicity of exposition, we divide inequality (69) into four terms \( LHS_1 + LHS_2 \leq RHS_1 + RHS_2 \). Taking their expectation with respect to \( S_k \), summing them over \( k \in \{1, 2, \cdots, K\} \) and then dividing them by \( KM \), gives

\[
LHS_1 = \left\{ \sum_{i=1}^{M} \frac{2V \alpha_i}{p_i \mu_i} \right\} \left\{ \frac{1}{KM} \sum_{k=1}^{K} \sum_{i=1}^{M} (\mu_i p_i - q_i) E \left[ x_i^+(k) \right] \right\}
\]

\[
LHS_2 = \frac{1}{KM} \sum_{k=1}^{K} E \left[ \left\{ \sum_{i=1}^{M} \alpha_i \right\} \left\{ \sum_{i=1}^{M} \frac{\alpha_i}{p_i \mu_i} + VM \right\} \right]^2
\]

\[
RHS_1 = - \left\{ \sum_{i=1}^{M} \frac{2 \alpha_i}{p_i \mu_i} \right\} \frac{1}{KM} \sum_{k=1}^{K} E[\Delta(S_k)]
\]

\[
RHS_2 = \frac{1}{M} \left\{ \sum_{i=1}^{M} \alpha_i \right\} \left\{ \sum_{i=1}^{M} \frac{\alpha_i}{p_i \mu_i} + VM \right\}.
\]

From the definition of Lyapunov Drift (38) and the fact that the Lyapunov Function (37) is non-negative, the expression of \( RHS_1 \) can be simplified as follows

\[
RHS_1 \leq \left\{ \sum_{i=1}^{M} \frac{2 \alpha_i}{p_i \mu_i} \right\} L(S_1) \frac{1}{KM},
\]

recall that \( h_i(1) = 1 \) and \( x_i(1) = 0 \). Hence, the Lyapunov Function \( L(S_1) \) is a positive finite constant.

Since \( RHS_2 \) is non-negative, it follows that the inequality can be reduced to \( LHS_1 \leq RHS_1 + RHS_2 \). Using (74) and applying the limit \( K \to \infty \) yields

\[
\lim_{K \to \infty} \left\{ \frac{1}{K} \sum_{k=1}^{K} E \left[ x_i^+(k) \right] \right\} \leq \frac{1}{2V} \left\{ \sum_{i=1}^{M} \frac{\alpha_i}{p_i \mu_i} + VM \right\}.
\]

Hence, by rearranging the terms, we can show that for any given node \( i \), we have strong stability

\[
\lim_{K \to \infty} \frac{1}{K} \sum_{k=1}^{K} E \left[ x_i^+(k) \right] < \infty,
\]

what establishes condition (36).

\[\text{APPENDIX C}\]

\[\text{PROOF OF THEOREM 7}\]

\[\text{Theorem 7 (Optimality Ratio for MW).} \] For any given network setup \((M, p_i, q_i, \alpha_i)\), the optimality ratio of MW is such that

\[
\psi^{MW} \leq 4 + \frac{1}{LB} \left[ V - \frac{2}{M} \sum_{i=1}^{M} \alpha_i \right]. \tag{77}
\]

In particular, for every network with \( V \leq 2 \sum_{i=1}^{M} \alpha_i/M \), the Max-Weight policy is 4-optimal.

\[\text{Proof:} \] Consider the analysis in Appendix B. In particular, the inequality \( LHS_1 + LHS_2 \leq RHS_1 + RHS_2 \) presented in (70)-(73). Applying Jensen’s inequality twice to \( LHS_2 \), yields

\[
\frac{1}{M} \left\{ \frac{1}{K} \sum_{k=1}^{K} E \left[ \sum_{i=1}^{M} \alpha_i \left( \frac{h_i(k) - 1}{p_i \mu_i} + 1 \right) \right] \right\}^2 \leq LHS_2
\]

\[
M \left\{ E \left[ J^K_{MW} \right] - \frac{1}{M} \sum_{i=1}^{M} \alpha_i \left( \frac{1}{p_i \mu_i} - 1 \right) \right\}^2 \leq LHS_2. \tag{78}
\]

Since \( LHS_1 \) is non-negative, it follows that the inequality can be reduced to \( LHS_2 \leq RHS_1 + RHS_2 \). Using equations (74) and (78), and then applying the limit \( K \to \infty \) yields

\[
\lim_{K \to \infty} \left\{ E \left[ J^K_{MW} \right] - \frac{1}{M} \sum_{i=1}^{M} \alpha_i \left( \frac{1}{p_i \mu_i} - 1 \right) \right\}^2 \leq \frac{1}{2M} \left\{ \sum_{i=1}^{M} \alpha_i \right\} \left\{ \sum_{i=1}^{M} \frac{\alpha_i}{p_i \mu_i} + VM \right\}
\]

\[
\lim_{K \to \infty} E \left[ J^K_{MW} \right] \leq \frac{1}{M} \sum_{i=1}^{M} \frac{\alpha_i}{p_i \mu_i} + V \leq OPT_{MW} \leq \frac{2}{M} \sum_{i=1}^{M} \frac{\alpha_i}{p_i \mu_i} + V. \tag{79}
\]

Analogously to the proof of Theorem 3, let \( q^L_t \) be the long-term throughput associated with the policy that solves the Lower Bound optimization (10a)-(10c). Then, evaluating \( LB \) from (10a) gives

\[
LB = \frac{1}{2M} \sum_{i=1}^{M} \frac{\alpha_i}{q^L_t} + \frac{1}{2M} \sum_{i=1}^{M} \alpha_i. \tag{80}
\]

Now, for each node \( i \), we impose the following scheduling probability \( \mu_i = \frac{q^L_t}{p_i} \). Then, evaluating (79) gives

\[
OPT_{MW} \leq \frac{2}{M} \sum_{i=1}^{M} \frac{\alpha_i}{q^L_t} + V. \tag{81}
\]

Comparing (80) and (81), yields

\[
LB \leq OPT_{MW} \leq 4LB + \left[ V - \frac{2}{M} \sum_{i=1}^{M} \alpha_i \right] \tag{82}
\]

what establishes the expression in (42).
ACKNOWLEDGMENT
The authors would like to thank Prof. H.-W. Lee for the helpful discussions regarding the Drift-Plus-Penalty policy and the anonymous reviewers for their insightful comments.

REFERENCES

Igor Kadota received the B.S. degree in electronic engineering from the Technological Institute of Aeronautics (ITA), Brazil, in 2010, the S.M. degree in telecommunication from ITA in 2013, and the S.M. degree in communication networks from the Massachusetts Institute of Technology (MIT) in 2016, where he is currently pursuing the Ph.D. degree with the Laboratory for Information and Decision Systems (LIDS).

His research is on modeling, analysis, and design of communication networks, with the emphasis on wireless networks and real-time traffic.

Mr. Kadota received the Best Paper Award at the IEEE INFOCOM 2018 for his work on Age of Information.
Abhishek Sinha received the B.E. degree in electronics and telecommunication engineering from Jadavpur University, Kolkata, India, the M.E. degree in telecommunication engineering from the Indian Institute of Science, Bangalore, India, and the Ph.D. degree from the Massachusetts Institute of Technology in 2017, where he worked in the Laboratory for Information and Decision Systems (LIDS).

He was a Senior Engineer at Qualcomm Research, San Diego, CA, USA. He is currently an Assistant Professor with the Department of Electrical Engineering at IIT Madras. His areas of interests include network control, information theory, machine learning, and applied probability.

Dr. Sinha was a recipient of several awards, including the Best Paper Award in INFOCOM 2018, the Best Paper Award in MobiHoc 2016, the Prof. Jnansaran Chatterjee Memorial Gold Medal, the T.P. Saha Memorial Gold Centered Silver Medal from Jadavpur University, and the Jagadis Bose National Science Talent Search (JBNSTS) Scholarship, Kolkata, India.

Eytan Modiano (F’12) received the B.S. degree in electrical engineering and computer science from the University of Connecticut at Storrs in 1986, and the M.S. and Ph.D. degrees in electrical engineering from the University of Maryland at College Park, College Park, MD, USA, in 1989 and 1992, respectively.

He was a Naval Research Laboratory Fellow from 1987 to 1992, a National Research Council Post-Doctoral Fellow from 1992 to 1993, and a member of the Technical Staff at Massachusetts Institute of Technology (MIT) Lincoln Laboratory from 1993 to 1999. He joined the faculty at MIT in 1999. He is currently a Professor with the Department of Aeronautics and Astronautics and an Associate Director of the Laboratory for Information and Decision Systems (LIDS), MIT. His research interest is on modeling, analysis, and design of communication networks and protocols.

Dr. Modiano was a co-recipient of the Infocom 2018 Best Paper Award, the MobiHoc 2018 Best Paper Award, the MobiHoc 2016 Best Paper Award, the Wiopt 2013 Best Paper Award, and the Sigmetrics 2006 Best Paper Award. He is the Editor-in-Chief of the IEEE/ACM TRANSACTIONS ON NETWORKING, and served as an Associate Editor for the IEEE TRANSACTIONS ON INFORMATION THEORY and the IEEE/ACM TRANSACTIONS ON NETWORKING. He was the Technical Program Co-Chair of the IEEE Wiopt 2006, IEEE INFOCOM 2007, ACM MobiHoc 2007, and DRCN 2015. He served on the IEEE Fellows Committee in 2014 and 2015. He is an Associate Fellow of the AIAA.