Average Age of Information with Hybrid ARQ under a Resource Constraint

Elif Tuğçe Ceran, Deniz Gündüz, and András György
Department of Electrical and Electronic Engineering
Imperial College London
Email: {e.ceran14, d.gunduz, a.gyorgy}@imperial.ac.uk

Abstract

Scheduling the transmission of status updates over an error-prone communication channel is studied in order to minimize the long-term average *age of information* at the destination under a constraint on the average number of transmissions at the source node. After each transmission, the source receives an instantaneous ACK/NACK feedback, and decides on the next update without prior knowledge on the success of future transmissions. The optimal scheduling policy is first studied under different feedback mechanisms when the channel statistics are known; in particular, the standard automatic repeat request (ARQ) and hybrid ARQ (HARQ) protocols are considered. Structural results are derived for the optimal policy under HARQ, while the optimal policy is determined analytically for ARQ. For the case of unknown environments, an average-cost reinforcement learning algorithm is proposed that learns the system parameters and the transmission policy in real time. The effectiveness of the proposed methods is verified through numerical results.

Index Terms

Age of information, hybrid automatic repeat request (HARQ), constrained Markov decision process, reinforcement learning

I. Introduction

Motivated by the growing interest in timely delivery of information in status update systems, the *age of information (AoI)* has been introduced as a performance measure to quantify data

Part of this work was presented at the IEEE Wireless Communications and Networking Conference, Barcelona, Spain, April 2018 [1].

staleness at the receiver [2]–[4]. Consider a source node that samples an underlying time-varying process, and sends the sampled status of the process over an imperfect communication channel that introduces delays. The AoI characterizes the data staleness (or tardiness) at the destination node, and it is defined as the time that has elapsed since the most recent status update available at the destination was generated. Different from classical performance measures, such as the delay or throughput, AoI jointly captures the latency in transmitting updates and the rate at which they are delivered.

Our goal in this paper is to minimize the average AoI at the destination taking into account *retransmissions* due to errors over the noisy communication channel. Retransmissions are essential for providing reliability of status updates over error-prone channels, particularly in wireless settings. Here, we analyze the AoI for both the standard ARQ and hybrid ARQ (HARQ) protocols.

In the HARQ protocol, the receiver combines information from all previous transmission attempts of the same packet in order to increase the success probability of decoding [5], [6], [7]. The exact relationship between the probability of error and the number of retransmission attempts varies depending on the channel conditions and the particular HARQ method employed [5], [6], [7]. In general, the probability of successful decoding increases with each transmission, but the AoI of the received packet also increases. Therefore, there is an inherent trade-off between retransmitting previously failed status information with a lower error probability, or sending a fresh status update with higher error probability. We address this trade-off between the success probability and the freshness of the status update to be transmitted, and develop scheduling policies to minimize the expected average AoI.

In the standard ARQ protocol, if a packet cannot be decoded, it is retransmitted until a successful transmission happens. Note, however, that, when optimizing for the AoI, there is no point of retransmitting the same packet, since a newer packet with more up-to-date information is available at the sender at the time of retransmission. Thus, after the reception of a NACK feedback, the actual packet is discarded, and the most recent status of the underlying process is transmitted (the exact timing of the transmission may depend on the feedback, i.e., on the success history of previous transmissions).

We develop scheduling policies for both the HARQ and the standard ARQ protocols to minimize the expected average AoI under a constraint on the average number of transmissions, which is motivated by the fact that sensors sending status updates have usually limited energy

supplies (e.g., are powered via energy harvesting [8]); and hence, they cannot afford to send an unlimited number of updates, or increase the signal-to-noise-ratio in the transmission. First, we assume that the success probability before each transmission attempt is known (which, in the case of HARQ, depends on the number of previous unsuccessful transmission attempts); and therefore, the source node can judiciously decide when to retransmit and when to discard a failed packet and send a fresh update. Then, we consider transmitting status updates over an unknown channel, in which case the success probabilities of transmission attempts are not known *a priori*, and must be learned in an online fashion. This latter scenario can model sensors embedded in unknown or time-varying environments. We employ reinforcement learning (RL) algorithms to balance exploitation and exploration in an unknown environment, so that the source node can quickly learn the environment based on the ACK/NACK feedback signals, and can adapt its scheduling policy accordingly, exploiting its limited resources in an efficient manner.

The main contributions of this paper are outlined as follows:

- Average AoI is studied under a long-term average resource constraint imposed on the transmitter, which limits the average number of transmissions.
- Both retransmissions and pre-emption following a failed transmission are considered, corresponding, respectively, to the HARQ and ARQ protocols.
- The optimal preemptive transmission policy for the standard ARQ protocol is shown to be a threshold-type randomized policy, and is derived in closed-form.
- An average-cost RL algorithm; in particular, *average-cost SARSA with softmax*, is proposed to learn the optimal scheduling decisions when the transmission success probabilities are unknown.
- Extensive numerical simulations are conducted in order to show the effect of feedback, resource constraint and ARQ or HARQ mechanisms on the data freshness.

A. Related Work

Most of the earlier work on AoI consider queue-based models, in which the status updates arrive at the source node randomly following a memoryless Poisson process, and are stored in a buffer before being transmitted to the destination [3], [4]. Instead, in the so-called *generate-at-will* model, [2], [9]–[12], also adopted in this paper, the status of the underlying process can be sampled at any time by the source node.

A constant packet failure probability for a status update system is investigated for the first time in [13], where status updates arrive according to a Poisson process, while the transmission time for each packet is exponentially distributed. Packet loss and large queuing delay due to old packets in the queue result in an increase in the AoI. Different scheduling decisions at the source node are investigated; including the last-come-first-served (LCFS) principle, which always transmits the most up-to-date packet, and retransmissions with preemptive priority, which preempts the current packet in service when a new packet arrives.

Broadcasting of status updates to multiple receivers over an unreliable broadcast channel is considered in [10]. A low complexity sub-optimal scheduling policy is proposed when the AoI at each receiver and the transmission error probabilities to all the receivers are known. However, only work-conserving policies are considered in [10], which update the information at every time slot, since no constraint is imposed on the number of updates. Optimizing the scheduling decisions with multiple receivers is also investigated in [11], focusing on a perfect transmission medium, and an optimal scheduling algorithm for the MDP is shown to be threshold-type. To the best of our knowledge, [11] is the only prior work in the literature which applies RL in the AoI framework. However, their goal is to learn the data arrival statistics, and it does not consider an unreliable communication link. Moreover, we employ an average-cost RL method, which has significant advantages over discounted-cost methods, such as *Q-learning* [14].

The AoI in the presence of HARQ has been considered in [15], [16] and [17]. In [15] the affect of design decisions, such as the length of the transmitted codewords, on the average AoI is analyzed. The status update system is modeled as an M/G/1/1 queue in [16]; however, no resource constraint is considered, and the status update arrivals are assumed to be memoryless and random, in contrast to our work, which considers the *generate-at-will* model. Moreover, a specific coding scheme is assumed in [16], namely MDS (maximum distance separable) coding, which results in a particular formula for the successful decoding probabilities, whereas we allow general functions for the decoding probabilities. From a queuing system perspective, our model can be considered as a G/G/1/1 queue with optimization of packet arrivals and pre-emption. In [17], HARQ is considered in a zero-wait system, where as soon as an update is delivered, a new update goes into service, yet no resource constraint or pre-emption is taken into account.

In [2] and [18], the receiver can choose to update its status information by downloading an update over one of the two available channels, an unreliable free channel, modeling a Wi-Fi connection, and a reliable channel with a cost, modeling a cellular connection. They have

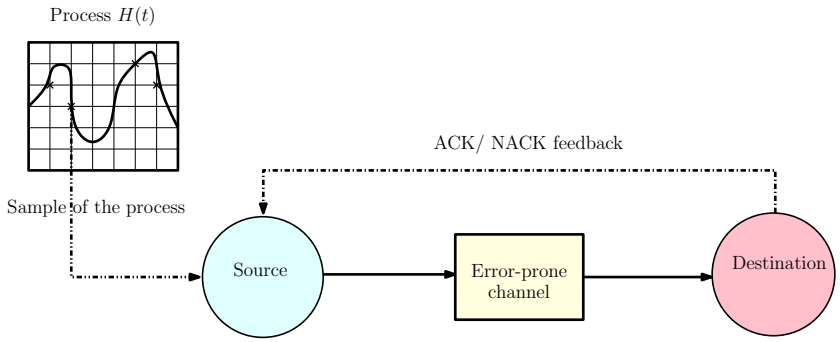


Figure 1. System model of a status update system over an error-prone point-to-point link in the presence of ACK/NACK feedback from the destination.

not considered the effect of retransmissions or any algorithm that learns the unknown system parameters; however, the Lagrangian formulation of our constrained optimization problem for the standard ARQ protocol is similar to the one considered in [2].

To the best of our knowledge, this is the first work in the literature that addresses a status update system with HARQ and in the presence of resource constraints. In addition, no previous work has studied the average AoI over a channel with unknown error probabilities, and employed an average-cost RL algorithm.

II. System Model and Problem Formulation

We consider a time-slotted status update system over an error-prone communication link (see Figure 1). The source monitors an underlying time-varying process, and can generate a status update at each time slot; known as the *generate-at-will* model [12]. The status updates are communicated from the source node to the destination over a time-varying channel. Each transmission attempt of a status update takes constant time, which is assumed to be equal to the duration of one time slot. We will normalize all the time durations by the duration of one time slot.

We assume that the channel changes randomly from one time slot to the next in an independent and identically distributed fashion, and the channel state information is available only at the destination node. We further assume the availability of an error- and delay-free single-bit feedback from the destination to the source node for each transmission attempt. Successful reception of a status update is acknowledged by an ACK signal, while a NACK signal is sent in case of a failure. In the classical ARQ protocol, a packet is retransmitted after each NACK feedback, until

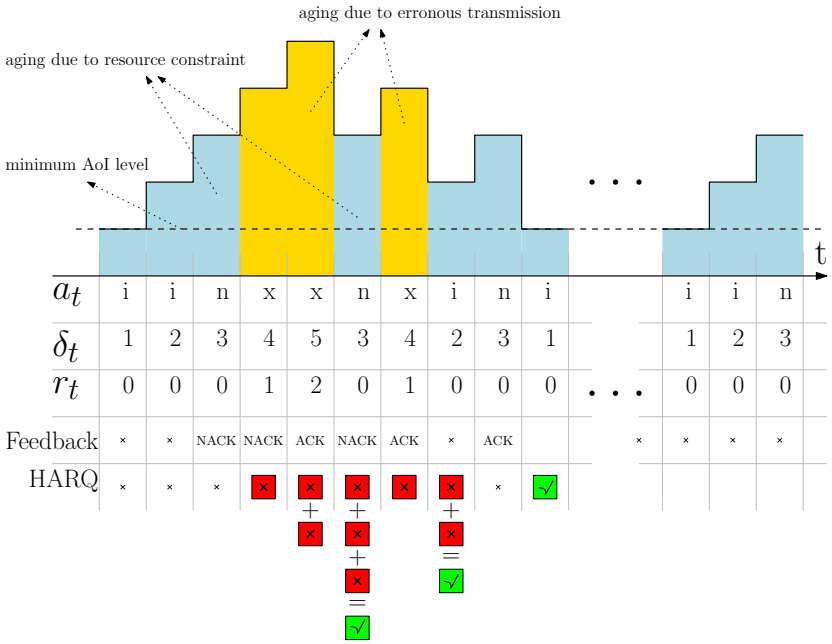


Figure 2. Illustration of the AoI in a slotted status update system with HARQ. (δ_t, r_t) represents the state of the system and the action is chosen based on the state (δ_t, r_t) and denoted by a_t. Packets with decoding errors (represented by red squares) are stored in the receiver and combined to decode the information successfully (represented by green squares).

it is successfully decoded (or, a maximum number of allowed retransmissions is reached), and the received signal is discarded after each failed transmission attempt. Therefore, the probability of error is the same for all retransmissions. Note, however, that, in the AoI framework with the classical ARQ protocol, there is no point in retransmitting a failed out-of-date status packet if it has the same error probability with a fresh update. Hence, we assume that, if the classical ARQ protocol is adopted, the source always removes failed packets, and transmits a fresh status update. On the other hand, in the HARQ protocol, the received signals from all previous transmission attempts for the same packet are combined for decoding. Therefore, the probability of error decreases with every retransmission. In general, the error probability of each retransmission attempt depends on the particular combination technique used by the decoder, as well as on the channel conditions [5].

The AoI measures the timeliness of the information at the receiver, and is defined as the number of time slots elapsed since the generation time of the most up-to-date packet successfully decoded at the receiver. Formally, denoting the generation time for any time slot t by $U(t)$, the

AoI, denoted by δ_t, is defined as

$$\delta_t \triangleq t - U(t). \tag{1}$$

We assume that a transmission decision is made at the beginning of each slot. The AoI increases by one when the transmission fails, while it decreases to one in the case of ARQ, or to the number of retransmissions in the case of HARQ, when a status update is successfully decoded.

The probability of error after r retransmissions, denoted by $g(r)$, depends on r and the particular HARQ scheme used for combining multiple transmission attempts (an empirical method to estimate $g(r)$ is presented in [6]). As in any reasonable HARQ strategy, we assume that $g(r)$ is non-increasing in the number of retransmissions r; that is, $g(r_1) \geq g(r_2)$ for all $r_1 \leq r_2$. Standard HARQ methods only allow a finite maximum number of retransmissions r_{max} [19]; however, in some cases we will allow r_{max} to be ∞.

For any time slot t, let $\delta_t \in \mathbb{Z}^+$ denote the AoI at the beginning of the time slot and $r_t \in \{0, \ldots, r_{max}\}$ denote the number of previous transmission attempts of the same packet. Then the state of the system can be described by $s_t \triangleq (\delta_t, r_t)$. At each time slot, the source node takes one of the three actions, denoted by $a \in \mathcal{A}$, where $\mathcal{A} = \{\mathrm{i}, \mathrm{n}, \mathrm{x}\}$: (i) remain idle ($a = \mathrm{i}$); (ii) transmit a new status update ($a = \mathrm{n}$); or (iii) retransmit the previously failed update ($a = \mathrm{x}$). The evolution of AoI for a slotted status update system is illustrated in Figure 2.

Note that if no resource constraint is imposed on the source, remaining idle is clearly sub-optimal since it does not contribute to decreasing the AoI. However, continuous transmission is typically not possible in practice due to energy or interference constraints. Accordingly, we impose a constraint on the average number of transmissions, denoted by $C_{max} \in (0, 1]$.

This leads to the CMDP formulation, defined by the 5-tuple $\big(\mathcal{S}, \mathcal{A}, \mathrm{P}, c, d\big)$ [20]: The countable set of states $(\delta, r) \in \mathcal{S}$ and the finite action set $\mathcal{A} = \{\mathrm{i}, \mathrm{n}, \mathrm{x}\}$ are already defined. P refers to the transition function, where $\mathrm{P}(s'|s, a) = \Pr(s_{t+1} = s' \mid s_t = s, a_t = a)$ is the probability that action a in state s at time t will lead to state s' at time $t + 1$, which will be explicitly defined in (4). The cost function $c : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, is the AoI at the destination, and is defined as $c((\delta, r), a) = \delta$ for any $(\delta, r) \in \mathcal{S}$, $a \in \mathcal{A}$, independent of action a. The transmission cost, $d : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is independent of the state and depends only on the action a, where $d = 0$ if $a = \mathrm{i}$, and $d = 1$, otherwise.

A policy is a sequence of decision rules $\pi_t : (\mathcal{S} \times \mathcal{A})^t \to [0, 1]$, which maps the past states and actions and the current state to a distribution over the actions, i.e., after the state-action sequence

$s_1, a_1, \ldots, s_{t-1}, a_{t-1}$, in state s_t, action a is selected with probability $\pi_t(a_t|s_1, a_1, \ldots, s_{t-1}, a_{t-1}, s_t)$. We will use $s_t^\pi = (\delta_t^\pi, r_t^\pi)$ and a_t^π to denote the sequences of states and actions, respectively, induced by policy $\pi = \{\pi_t\}$. A policy $\pi = \{\pi_t\}$ is called *stationary* if the distribution of the next action is independent of the past states and actions given the current state; that is, with a slight abuse of notation, $\pi_t(a_t|s_1, a_1, \ldots, s_{t-1}, a_{t-1}, s_t) = \pi(a_t|s_t)$ for all t and $(s_i, a_i) \in \mathcal{S} \times \mathcal{A}$. Finally, a policy is said to be deterministic if it chooses an action with probability one; with a slight abuse of notation, we will use $\pi(s)$ to denote the action taken with probability one in state s by a stationary deterministic policy.

Let $J^\pi(s_0)$ denote the infinite horizon average age, and $C^\pi(s_0)$ denote the expected average number of transmissions when policy π is employed with initial state s_0. Then the CMDP optimization problem can be stated as follows:

Problem 1.

$$\text{Minimize } J^\pi(s_0) \triangleq \limsup_{T\to\infty} \frac{1}{T}\, \mathbb{E}\left[\sum_{t=1}^{T} \delta_t^\pi \Big| s_0\right], \tag{2}$$

$$\text{subject to } C^\pi(s_0) \triangleq \limsup_{T\to\infty} \frac{1}{T}\, \mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}[a_t^\pi \neq \mathrm{i}] \Big| s_0\right] \leq C_{max}. \tag{3}$$

A policy π which is a solution of the above minimization problem is called optimal, and we are interested in finding optimal policies. Without loss of generality, we assume that the sender and the receiver are synchronized at the beginning of the problem, that is, $s_0 = (1, 0)$; and s_0 will be omitted from the notation for simplicity.

Before formally defining the transition function P in our AoI problem, we present a simple observation that allows to simplify P. It is easy to see that retransmitting a packet immediately after a failed attempt is better than retransmitting it after waiting for some slots. This is obviously true since waiting increases the age, without increasing the success probability. The difference in the waiting time is illustrated in Figure 3 for a simple scenario, where the first transmission of a status update results in a failure, while the retransmission is successful.

Proposition 1. *For any policy π there exists another policy π' (not necessarily distinct from π) such that $J^{\pi'}(s_0) \leq J^\pi(s_0)$, $C^{\pi'}(s_0) \leq C^\pi(s_0)$, and π' takes a retransmission action only following a failed transmission, that is, $Pr(a_{t+1}^{\pi'} = \mathrm{x}|a_t^{\pi'} = \mathrm{i}) = 0$.*

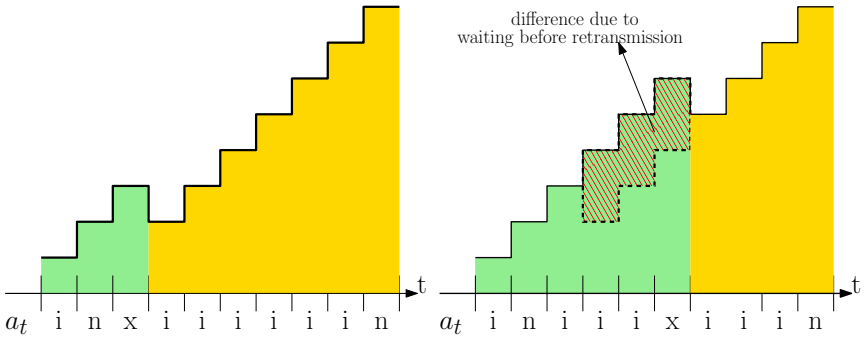


Figure 3. The difference of the AoI for policies without and with idle slots before retransmissions. The figure on the left shows the evolution of age (height of the bars) when retransmission occurs immediately after an error in transmission whereas the figure on the right represents the evolution of age when retransmission occurs after some idle slots.

The transition probabilities are given as follows (omitting the parenthesis from the state variables (δ, r)):

$$\begin{aligned}
&\mathrm{P}(\delta+1, 0|\delta, r, \mathrm{i}) = 1,\\
&\mathrm{P}(\delta+1, 1|\delta, r, \mathrm{n}) = g(0),\\
&\mathrm{P}(1, 0|\delta, r, \mathrm{n}) = 1-g(0),\\
&\mathrm{P}(\delta+1, r+1|\delta, r, \mathrm{x}) = g(r),\\
&\mathrm{P}(r+1, 0|\delta, r, \mathrm{x}) = 1-g(r),
\end{aligned} \tag{4}$$

and $\mathrm{P}(\delta', r'|\delta, r, a) = 0$ otherwise. Note that the above equations set the retransmission count to 0 after each successful transmission, and it is not allowed to take a retransmission action in states where the transmission count is 0. Also, the property in Proposition 1 is enforced by the first equation in (4), that is, $\mathrm{P}(\delta+1, 0|\delta, r, \mathrm{i}) = 1$ (since retransmissions are not allowed in states $(\delta, 0)$).

III. Lagrangian Relaxation and the Structure of the Optimal Policy

In this section, we derive the structure of the optimal policy for Problem 1 based on [20], [21]. While there exists a stationary and deterministic optimal policy for countable-state finite-action average-cost MDPs [22]–[24], this is not necessarily true for CMDPs [20], [21]. To solve the

CMDP, we start with rewriting the problem in its Lagrangian form. The average Lagrangian cost of a policy π with Lagrange multiplier $\eta \geq 0$ is defined as

$$J_\eta^\pi = \lim_{T\to\infty} \frac{1}{T}\mathbb{E}\left[\sum_{t=1}^{T} \delta_t^\pi\right] - \eta\left(C_{max} - \frac{1}{T}\mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}[a_t^\pi \neq \mathrm{i}]\right]\right), \tag{5}$$

and, for any η, the optimal achievable cost J_η^* is defined as $J_\eta^* \triangleq \min_\pi J_\eta^\pi$. This formulation is equivalent to an unconstrained countable-state average-cost MDP, in which the instantaneous overall cost becomes $\delta_t + \eta\mathbb{1}[a_t^\pi \neq \mathrm{i}]$. It is well-known that there exists an optimal stationary deterministic policy for this problem. In particular, there exists a function $h_\eta(\delta, r)$, called the *differential cost function*, satisfying the following *Bellman optimality equations* for the countable-state MDP [23], [24]:

$$h_\eta(\delta, r) + J_\eta^* = \min_{a\in\{\mathrm{i,n,x}\}} \big(\delta + \eta\cdot\mathbb{1}[a \neq \mathrm{i}] + \mathbb{E}\left[h_\eta(\delta', r')\right]\big), \tag{6}$$

where (δ', r') is the next state obtained from (δ, r) after taking action a. Note that the function h_η satisfying (6) is unique up to an additive factor, and with selecting this additive factor properly, it also satisfies

$$h_\eta(\delta, r) = \mathbb{E}\left[\sum_{t=0}^{\infty} (\delta_t + \eta\cdot\mathbb{1}[a \neq \mathrm{i}] - J_\eta^*)\big|\delta_0 = \delta, r_0 = r\right],$$

in which case it is called the *value function*, and denoted by V_η. We also introduce the *state-action cost function*:

$$Q_\eta(\delta, r, a) \triangleq \delta + \eta\cdot\mathbb{1}[a \neq \mathrm{i}] + \mathbb{E}\left[h_\eta(\delta', r')\right] \ . \tag{7}$$

Then the optimal policy, for any $(\delta, r) \in \mathcal{S}$, takes the action achieving the minimum in (6):

$$\pi_\eta^*(\delta, r) \in \arg\min_{a\in\{\mathrm{i,n,x}\}} \big(Q_\eta(\delta, r, a)\big) \ . \tag{8}$$

Focusing on deterministic policies, it is possible to characterize optimal policies for our CMDP problem: Combining Theorem 4.4 of [20] with Theorem 2.5 of [21] and its proof, we obtain the following result:

Theorem 1. *An optimal stationary policy for the CMDP in Problem 1, which randomizes in at most one state, exists. Alternatively, an optimal stationary policy, which is a mixture of two deterministic policies, exists; that is, there exist Lagrange multipliers $\eta_1, \eta_2 \geq 0$, and a mixing coefficient $\mu \in [0, 1]$, such that the mixture policy $\pi_{\eta_1,\eta_2,\mu}^* \triangleq \mu\pi_{\eta_1}^* + (1-\mu)\pi_{\eta_2}^*$ is optimal for Problem 1, and the constraint in* (3) *is satisfied with equality.*

In the theorem we use $\pi^*_{\eta_i}$ to denote the optimal policy for the unconstrained MDP with Lagrange multiplier η_i. One can think of the optimal policy $\pi^*_{\eta_1,\eta_2,\mu}$ for the CMDP as a randomized policy between two deterministic policies: in any state $s = (\delta, r)$, it chooses action $\pi^*_{\eta_1}(s)$ with probability μ and $\pi^*_{\eta_2}(s)$ with probability $1-\mu$, independently at each time slot. While Theorem 1 presents the general structure of the optimal policy, it does not provide any guidance on how to select η_1, η_2 and μ, which will be provided next. Note that $\pi^*_{\eta_i}$ may not be unique due to multiple reasons: we can have policies with a different balance between the AoI and the transmission cost; or, different policies may yield the same performance. In what follows, we define $\pi^*_{\eta_i}$ to be any Lagrangian-optimal policy for η_i with the minimum average AoI.

For any η, let C_η denote the average number of transmissions under the optimal policy π^*_η, and J^*_η denote the average AoI for π^*_η. Note that by our definition of π^*_η, the definitions of J^*_η and C_η are unambiguous (also note that C_η and J^*_η can be computed directly by finding the stationary distribution of the chain, or estimated empirically by running the MDP with policy π^*_η). Since η effectively represents the cost of a single transmission in (6) and (7), as η increases, the average number of transmissions of the optimal policy cannot increase, and as a result, the AoI cannot decrease; that is, C_η and J^*_η are monotone functions of η: if $\eta_1 < \eta_2$, we have $C_{\eta_1} \geq C_{\eta_2}$ and $J^*_{\eta_1} \leq J^*_{\eta_2}$. Therefore, given the values of η_1 and η_2, one can find the optimal mixing coefficient μ by solving $\mu C_{\eta_1} + (1-\mu)C_{\eta_2} = C_{max}$, which has a solution for $\mu \in [0,1]$ if $C_{\eta_1} \geq C_{max} \geq C_{\eta_2}$. Given the whole curve $\mathcal{JC} = \{(C_\eta, J^*_\eta) : \eta \geq 0\}$ parametrized by η, the mixture policies defined in Theorem 1 span the lower convex hull of $\mathcal{JC}$. Thus, the optimal Lagrange multipliers η_1 and η_2 can be found by identifying which two points of $\mathcal{JC}$ determine the lower convex hull at C_{max}. Let $\mathrm{lch}(\mathcal{JC})$ denote the set of η such that (C_η, J^*_η) belongs to the lower convex hull of $\mathcal{JC}$. For any $\eta \in \mathrm{lch}(\mathcal{JC})$, π^*_η is the optimal policy for the CMDP with constraint $C_{max} = C_\eta$, and there is no need to mix the two policies. For any other η, let $\eta_1, \eta_2 \in \mathrm{lch}(\mathcal{JC})$ be the two points on the lower convex hull of $\mathcal{JC}$, such that the segment connecting $(C_{\eta_1}, J^*_{\eta_1})$ and $(C_{\eta_2}, J^*_{\eta_2})$ touches the lower convex hull of $\mathcal{JC}$ with x-coordinate C_{max}. Then the optimal mixing coefficient can be obtained by setting the average number of transmissions, $\mu C_{\eta_1} + (1-\mu)C_{\eta_2}$ of the mixture to C_{max}; that is,

$$\mu = \frac{C_{max} - C_{\eta_2}}{C_{\eta_1} - C_{\eta_2}} \ , \tag{9}$$

and the optimal policy is

$$\pi^*_{\eta_1,\eta_2,\mu} = \mu \pi^*_{\eta_1} + (1-\mu)\pi^*_{\eta_2}. \tag{10}$$

Note that, the whole $\mathcal{JC}$ curve needs to be generated and its lower convex hull needs to be determined in order to find the optimal choices of η_1, η_2 and μ, which is not suitable for practical applications. In Section IV, a computationally efficient heuristic algorithm is proposed to find these parameters.

IV. An Iterative Algorithm to Minimize the AoI under an Average Cost Constraint

By the discussion at the end of the last section, we have to find two points on the lower convex hull of $\mathcal{JC}$, whose cost value is approximately C_{max}. This problem is easy if we have a closed form description of $\mathcal{JC}$. Otherwise, we have to generate elements on $\mathcal{JC}$ (i.e., determine the optimal policy and the associated cost for different values of η), and determine the aforementioned two points based on these elements.

We remark that our state space is countably infinite, since the age can be arbitrarily large (r_{max} may also be infinite). However, in practice we can approximate the countable state space with a large but finite space by setting a maximum bound on the age (which will be denoted by N), and by selecting a finite r_{max} (whenever the chain would leave this constrained state space, we truncate the value of the age and/or the retransmission number to N and r_{max}, respectively); this gives a finite state space approximation to the problem similarly to [2], [11]. Clearly, letting N and r_{max} go to infinity, the optimal policy for the restricted state space will converge to that of the original problem.

When we consider the finite state space approximation of our problem, we can employ the *relative value iteration* (RVI) [23] algorithm to solve (6) for any given η; and hence, find the optimal policy π_η^*. Note that the finite state space approximation is needed for the practical implementation of the RVI algorithm since each iteration in the RVI requires the computation of the value function for each state-action pair, which cannot be completed in finite time for an infinite state space. The pseudo code of the RVI algorithm is given in Algorithm 1. To simplify the notation, the dependence on η is suppressed in the algorithm for h, V and Q.

After presenting an algorithm that can compute the optimal policy π_η^* for any given η (more precisely, an arbitrarily close approximation thereof), we need to find the values for the Lagrange multipliers η_1 and η_2. In general, we would need to generate the whole $\mathcal{JC}$ curve to determine its lower convex hull. This could be approximated by computing (C_η, J_η^*) for a fine grid of η values, but this approach might be computationally demanding (note that generating each point

Algorithm 1: Relative value iteration (RVI) algorithm for a given η.

Input : Lagrange parameter η, error probability $g(r)$

(δ^{ref}, r^{ref}) /* choose an arbitrary but fixed reference state */
$n \leftarrow 0$ /* iteration counter */
$h_0^{N \times r_{max}} \leftarrow \mathbf{0}$ /* initialization */
while 1 /* until convergence */
do
  for *state* $s = (\delta, r) \in [1, \dots, N] \times [1, \dots, r_{max}]$ **do**
    for *action* $a \in \mathcal{A}$ **do**
      $Q_{n+1}(\delta, r, a) \leftarrow \delta + \eta \cdot \mathbb{1}[a^\pi \neq \mathrm{i}] + \mathbb{E}\left[h_n(\delta', r')\right]$
    end
    $V_{n+1}(\delta, r) \leftarrow \min_a(Q_{n+1}(\delta, r, a))$
    $h_{n+1}(\delta, r) \leftarrow V_{n+1}(\delta, r) - V_{n+1}(\delta^{ref}, r^{ref})$
  end
  if $|h_{n+1} - h_n| \leq \epsilon$ **then**
    /* compute the optimal policy */
    for $(\delta, r) \in [1, \dots, N] \times [1, \dots, r_{max}]$ **do**
      $\pi_\eta^*(\delta, r) \leftarrow \arg\min_a(Q(\delta, r, a))$
    end
    return π^*
  else
    increase the iteration counter: $n \leftarrow n + 1$
  end
end

requires running an instance of the RVI). Instead, we can use the following heuristic: With the aim of finding a single η value with $C_\eta \approx C_{max}$, we start with an initial parameter η^0, and run an iterative algorithm updating η as $\eta^{m+1} = \eta^m + \alpha_m(C_{\eta^m} - C_{max})$ for a step size parameter $\alpha_m = 1/\sqrt{m}$ (note that for each step we need to run the RVI algorithm to be able to determine C_{η^m}). We continue this iteration until $|C_{\eta^m} - C_{max}|$ becomes smaller than a given threshold, and denote the resulting value as η^*. We can increase or decrease the η^* value until η^* and its modification satisfy the conditions (note that in case of a finite state space, which is an approximation we always use in computing an optimal policy numerically, π_η, and consequently C_η and J_η^*, are piecewise constant functions of η, and so η must be changed sufficiently to

change the average transmission cost).

Next we approximate the values of η_1 and η_2 by $\eta^* \pm \xi$ where ξ is a small perturbation, such that $C_{\eta_1} \geq C_{max} \geq C_{\eta_2}$. Then, the mixing coefficient can be chosen similarly to (9) in Section III

$$\mu = \frac{C_{max} - C_{\eta^*+\xi}}{C_{\eta^*-\xi} - C_{\eta^*+\xi}} \ . \tag{11}$$

Numerical results obtained by implementing the above heuristics in order to minimize the average AoI with HARQ will be presented in Section VII. In the next section, we focus on the simpler scenario with the classical ARQ protocol.

V. AoI with Classical ARQ Protocol under an Average Cost Constraint

In the classical ARQ protocol, failed transmissions are discarded at the destination and the receiver tries to decode each retransmission as a new message. In the context of AoI, there is no point in retransmitting an undecoded packet since the probability of a successful transmission is the same for a retransmission and for the transmission of a new update. Hence, the state space reduces to $\delta \in \{1, 2, \ldots\}$ as $r_t = 0$ for all t, and the action space reduces to $\mathcal{A} \in \{\mathrm{n}, \mathrm{i}\}$, and the probability of error $p \triangleq g(0)$ is fixed for every transmission attempt. [1] State transitions in (4), Bellman optimality equations [23], [24] for the countable-state MDP in (6), and the RVI algorithm with the finite state approximation can all be simplified accordingly. We define

$$Q_\eta(\delta, \mathrm{i}) \triangleq \delta + h_\eta(\delta + 1), \tag{12}$$

$$Q_\eta(\delta, \mathrm{n}) \triangleq \delta + \eta + p h_\eta(\delta + 1) + (1 - p) h_\eta(1), \tag{13}$$

where $h_\eta(\delta)$ is the optimal differential value function satisfying the Bellman optimality equation

$$h_\eta(\delta) + J_\eta^* \triangleq \min \{Q_\eta(\delta, \mathrm{i}), Q_\eta(\delta, \mathrm{n})\}, \ \forall \delta \in \{1, 2, \ldots\}. \tag{14}$$

Thanks to these simplifications, we are able to provide a closed-form solution to the corresponding Bellman equations in (12), (13) and (14).

[1] This simplified model with classical ARQ protocol and Lagrangian relaxation is equivalent to the work in [2] when η is considered to be the cost of a single transmission and the assumption of a perfect transmission channel in [2] is ignored.

Lemma 1. *The policy that satisfies the Bellman optimality equations for the standard ARQ protocol is deterministic, and has a threshold structure:*

$$\pi^*(\delta) = \begin{cases} \text{n} & \textit{if } \delta \geq \Delta_\eta, \\ \text{i} & \textit{if } \delta < \Delta_\eta. \end{cases}$$

for some integer Δ_η that depends on η.

Proof. The proof is given in Appendix A. □

The next lemma characterizes the possible values of the threshold defined in Lemma 1.

Lemma 2. *Under the standard ARQ protocol, the optimal value of the threshold Δ_η can be found in closed-form:*

$$\Delta_\eta^* \in \left\{ \left\lfloor \frac{\sqrt{2\eta(1-p)+p}-p}{1-p} \right\rfloor, \left\lceil \frac{\sqrt{2\eta(1-p)+p}-p}{1-p} \right\rceil \right\}.$$

Proof. The proof is given in Appendix B. □

From the proof of the lemma one can easily deduce that the transmission cost (per time slot) of the threshold policy for any integer threshold Δ is given by

$$C^\Delta = \frac{1}{\Delta(1-p)+p}, \tag{15}$$

and the corresponding AoI is

$$J^\Delta = \frac{(\Delta(1-p)+p)^2+p}{2(1-p)(\Delta(1-p)+p)} + \frac{1}{2}.$$

Expressing J^Δ as a function of C^Δ, and relaxing the integrality constraint on Δ, one can see that

$$J^\Delta = \frac{1}{2(1-p)C^\Delta} + \frac{1}{2} + \frac{pC^\Delta}{2(1-p)}$$

is a convex function of C^Δ. Thus, for all positive integers Δ, the points (C^Δ, J^Δ) lie on the lower convex hull of the graph $\mathcal{JC} = \{(C_\eta, J_\eta^*), \eta \geq 0\}$, and no other deterministic policy achieves the lower convex hull (recall the discussion after Theorem 1). Therefore, by (10)), if $C_{max} \in (C^\Delta, C^{\Delta+1})$ for some Δ, then the optimal policy is a mixture of the threshold policies with thresholds Δ and $\Delta+1$. These threshold values can be found by inverting (15), and taking the closest integers to the resulting non-integer threshold value. In particular, defining $\Delta_{C_{max}} \triangleq \frac{1/C_{max}-p}{1-p}$, $\Delta_1 \triangleq \lfloor \Delta_{C_{max}} \rfloor$ and $\Delta_2 \triangleq \lceil \Delta_{C_{max}} \rceil$, we obtain that the optimal policy is a mixture of the threshold policies of Lemma 1 with thresholds Δ_1 and Δ_2 and mixture coefficient

$\mu = \frac{C_{max}-C^{\Delta_2}}{C^{\Delta_1}-C^{\Delta_2}}$. The resulting policy $\pi^*_{C_{max}}$ can be written in closed form: if $\Delta_{C_{max}}$ is an integer then $\pi^*_{C_{max}}(\delta) = \mathrm{n}$ if $\delta \geq \Delta_{C_{max}}$ and i otherwise. If $\Delta_{C_{max}}$ is not an integer, then $\pi^*_{C_{max}}(\delta) = \mathrm{n}$ if $\delta \geq \lceil \Delta_{C_{max}} \rceil$, $\pi^*_{C_{max}}(\delta) = \mathrm{i}$ if $\delta < \lfloor \Delta_{C_{max}} \rfloor$, while $\pi^*_{C_{max}}(\mathrm{n}|\delta) = 1-\mu$ and $\pi^*_{C_{max}}(\mathrm{i}|\delta) = \mu$ for $\delta = \lfloor \Delta_{C_{max}} \rfloor$. This proves the following theorem:

Theorem 2. *For any $C_{max} \in (0, 1]$, the stationary policy $\pi^*_{C_{max}}$ defined above is an optimal policy (i.e., a solution of Problem 1) under the ARQ protocol.*

Numerical results obtained for the above algorithm will be presented and compared with those from the HARQ protocol in Section VII.

VI. Learning to minimize AoI in an unknown environment

In the CMDP formulation presented in Sections IV and V, we have assumed that the channel error probabilities for all retransmissions are known in advance. However, in most practical scenarios, these error probabilities may not be known at the time of deployment, or may change over time. Therefore, in this section, we assume that the source node does not have *a priori* information about the decoding error probabilities, and has to learn them. We employ an online learning algorithm to learn $g(r)$ over time without degrading the performance significantly.

The literature for average-cost RL is quite limited compared to discounted cost problems [14], [25]. SARSA [25] is a well-known RL algorithm, originally proposed for discounted MDPs, that learns the optimal policy for an MDP based on the action performed by the current policy in a recursive manner. For average AoI minimization in Problem 1, an average cost version of the SARSA algorithm is employed with *Boltzmann* (*softmax*) exploration. The resulting algorithm is called *average-cost SARSA with softmax*.

As indicated by (6) and (7) in Section III, $Q_\eta(s_n, a_n)$ of the current state-action pair can be represented in terms of the immediate cost of the current state-action pair and the differential state-value function $h_\eta(s_{n+1})$ of the next state. Notice that, one can select the optimal actions by only knowing $Q_\eta(s, a)$ and choosing the action that will give the minimum expected cost as in (8). Thus, by only knowing $Q_\eta(s, a)$, one can find the optimal policy π^* without knowing the transition probabilities P characterized by $g(r)$ in (4).

Similarly to SARSA, *average-cost SARSA with softmax* starts with an initial estimation of $Q_\eta(s, a)$ and finds the optimal policy by estimating state-action values in a recursive manner. In the n^{th} time iteration, after taking action a_n, the source observes the next state s_{n+1}, and the

Algorithm 2: Average-cost SARSA with softmax

Input : Lagrange parameter η /* error probability $g(r)$ is unknown */

$n \leftarrow 0$ /* time iteration */

$\tau \leftarrow 1$ /* softmax temperature parameter */

$Q_\eta^{N\times M\times 3} \leftarrow 0$ /* initialization of Q */

$J_\eta \leftarrow 0$ /* initialization of the gain */

for n **do**

  OBSERVE the current state s_n

  for $a \in \mathcal{A}$ **do**

    /* since it is a minimization problem, use minus Q function in softmax */

    $$\pi(a|s_n) = \frac{\exp(-Q_\eta(s_n,a)/\tau)}{\sum_{a'\in\mathcal{A}} \exp(-Q_\eta(s_n,a')/\tau)}$$

  end

  SAMPLE a_n from $\pi(a|S_n)$

  OBSERVE the next state s_{n+1} and cost $c_n = \delta_n + \eta 1_{\{a_n=1,2\}}$

  for $a \in \mathcal{A}$ **do**

    /* softmax is also used for the next state s_{n+1}, so that it is on-policy */

    $$\pi(a|s_{n+1}) = \frac{\exp(-Q_\eta(s_{n+1},a_{n+1})/\tau)}{\sum_{a'_{n+1}\in\mathcal{A}} \exp(-Q_\eta(s_{n+1},a'_{n+1})/\tau)}$$

  end

  SAMPLE a_{n+1} from $\pi(a_{n+1}|s_{n+1})$

  UPDATE

  $\alpha_n \leftarrow 1/\sqrt{n}$ /* update parameter */

  $Q_\eta(s_n,a_n) \leftarrow Q_\eta(s_n,a_n) + \alpha_n[\delta + \eta \cdot \mathbb{1}[a_n \neq \mathrm{i}] - J_\eta + Q_\eta(s_{n+1},a_{n+1}) - Q_\eta(s_n,a_n)]$

  $J_\eta \leftarrow J_\eta + 1/n[\delta + \eta \cdot \mathbb{1}[a_n \neq \mathrm{i}] - J_\eta]$ /* update J_η at every step */

  $n \leftarrow n+1$ /* increase the iteration */

end

instantaneous cost value c_n. Based on this, the estimate of $Q_\eta(s,a)$ is updated by weighing the previous estimate and the estimated expected value of the current policy in the next state s_{n+1}. Also note that, in general, c_n is not necessarily known before taking action a_n because it does not know the next state s_{n+1} in advance. In our problem, the instantaneous cost c_n is the sum

of AoI at the destination and the cost of transmission, i.e. $\delta_n + \eta \cdot \mathbb{1}[a_n \neq \text{i}]$; hence, it is readily known at the source node.

In each time slot, the learning algorithm

- observes the current state $s_n \in \mathcal{S}$,
- selects and performs an action $a_n \in \mathcal{A}$,
- observes the next state $s_{n+1} \in \mathcal{S}$ and the instantaneous cost c_n,
- updates its estimate of $Q_\eta(s_n, a_n)$ using the current estimate of J_η by

$$Q_\eta(s_n, a_n) \leftarrow Q_\eta(s_n, a_n) + \alpha_n[\delta + \eta \cdot \mathbb{1}[a_n \neq \text{i}] - J_\eta + Q_\eta(s_{n+1}, a_{n+1}) - Q_\eta(s_n, a_n)], \quad (16)$$

  where α_n is the update parameter (learning rate) in the n^{th} iteration.
- updates its estimate of J_η based on empirical average.

The details of the algorithm are given in Algorithm 2. We update the gain J_η at every time slot based on the empirical average, instead of updating it at non-explored time slots.

As we discussed earlier, with the accurate estimate of $Q_\eta(s, a)$ at hand the transmitter can decide for the optimal actions for a given η as in (8). However, until the state-action cost function is accurately estimated, the transmitter action selection method should balance the *exploration* of new actions with the *exploitation* of actions known to perform well. In particular, the *Boltzmann* action selection method, which chooses each action probabilistically relative to expected costs, is used in this paper. The source assigns a probability to each action for a given state s_n, denoted by $\pi(a|s_n)$:

$$\pi(a|s_n) \triangleq \frac{\exp(-Q_\eta(s_n, a)/\tau)}{\sum_{a' \in \mathcal{A}} \exp(-Q_\eta(s_n, a')/\tau)}, \quad (17)$$

where τ is called the temperature parameter such that high τ corresponds to more uniform action selection (exploration) whereas low τ is biased toward the best action (exploitation).

In addition, the constrained structure of the average AoI problem requires additional modifications to the algorithm, which is achieved in this paper by updating the Lagrange multiplier according to the empirical resource consumption. In each time slot, we keep track of a value η resulting in a transmission cost close to C_{max}, and then find and apply a policy that is optimal (given the observations so far) for the MDP with Lagrangian cost as in Algorithm 2.

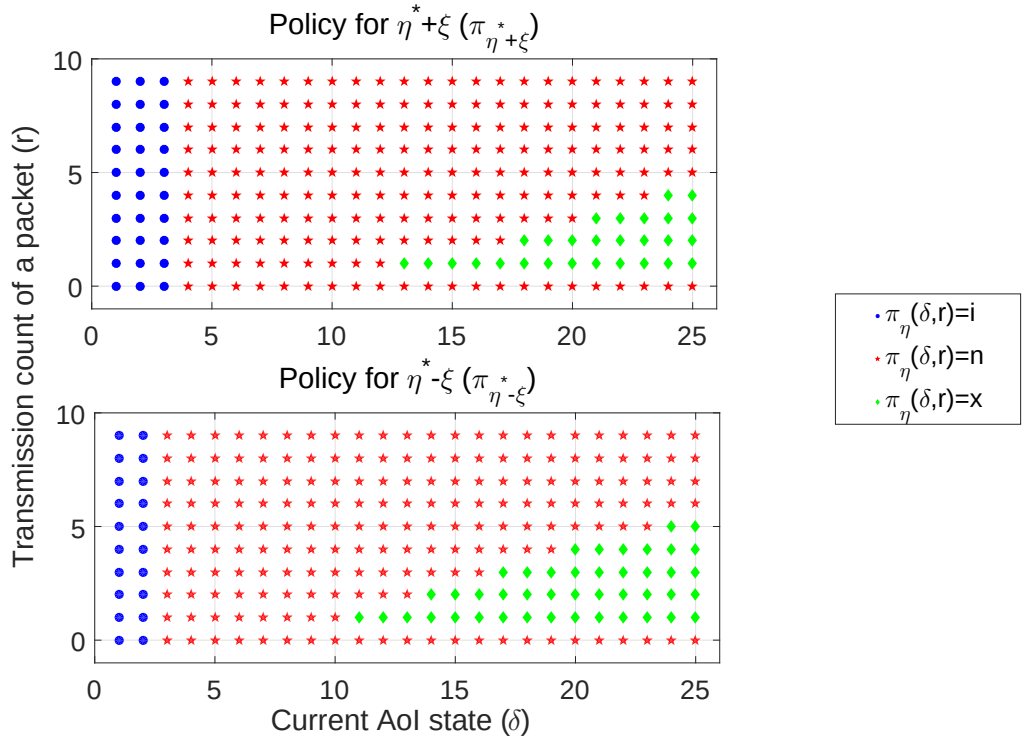
The performance of *average-cost SARSA with softmax*, and its comparison with the RVI algorithm will be presented in the next section.

VII. Numerical Results

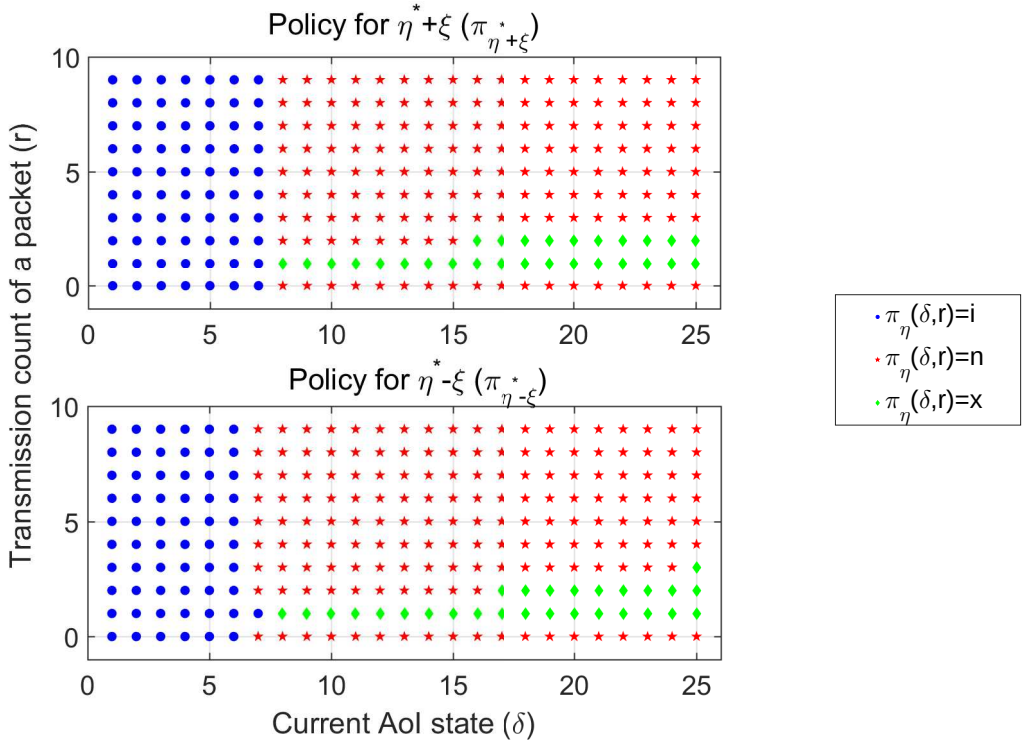
In this section, we provide numerical results for all the proposed algorithms, and compare the achieved average performances. For the simulations employing HARQ, motivated by previous research on HARQ [5], [6], [7], we assume that decoding error reduces exponentially with the number of retransmission, that is, $g(r) \triangleq p_0\lambda^r$ for some $\lambda \in (0,1)$, where p_0 denotes the error probability of the first transmission, and r is the retransmission count (set to 0 for the first transmission). The exact value of the rate λ depends on the particular HARQ protocol and the channel model. Note that ARQ corresponds to the case with $\lambda = 1$ and $r_{max} = 0$. Following the *IEEE 802.16* standard [19], the maximum number of retransmissions is set to $r_{max} = 3$; however, we will present results for other r_{max} values as well. We note that we have also run simulations for HARQ with relatively higher r_{max} values and $r_{max} = \infty$, and the improvement on the performance is not observable beyond $r_{max} = 3$. Numerical results for different p_0, λ and C_{max} values, corresponding to different channel conditions and HARQ schemes, will also be provided.

Figure 4 illustrates the deterministic policies obtained by RVI and the search for η^* for given C_{max} and p_0 values, while λ is set to 0.5. The final policies are generated by randomizing between $\pi^*_{\eta^*-\xi}$ and $\pi^*_{\eta^*+\xi}$; the approximate η^* values found for the settings in Figures 4(a) and 4(b) are 5 and 19, respectively, and ξ is set to 0.2. As it can be seen from the figures, the resulting policy transmits less as the average cost constraint becomes more limiting, i.e., as η increases. We also note that, although the policies $\pi^*_{\eta^*-\xi}$ and $\pi^*_{\eta^*+\xi}$ are obtained for similar η^* values, and hence, have similar average number of transmissions, they may act quite differently especially for large C_{max} values.

Figure 5 illustrates the performance of the proposed randomized HARQ policy with respect to C_{max} for different p_0 values when λ is set to 0.5. We also include the performance of the optimal deterministic and randomized threshold policies with ARQ, derived in Section V, for $p_0 = 0.5$. For baseline, we use a simple no-feedback policy that periodically transmits a fresh status update with a period of $\lceil 1/C_{max} \rceil$, ensuring that the constraint on the average number of transmissions holds. The effect of feedback on the performance can be seen immediately: a single-bit ACK/NACK feedback, even with the ARQ protocol, decreases the average AoI considerably, although receiving feedback might be costly for some status update systems. The two curves for the ARQ policies demonstrate the effect of randomization: the curve corresponding



(a) $C_{max} = 0.4$, $p_0 = 0.3$



(b) $C_{max} = 0.2$, $p_0 = 0.4$

Figure 4. Deterministic policies $\pi_{\eta^*+\xi}$ (top) and $\pi_{\eta^*-\xi}$ (bottom) when $\lambda = 0.5$ and $r_{max} = 9$. (Blue circles, red stars, and green diamonds represent actions $\pi_\eta(\delta, r) =$ i, n and x, respectively.)

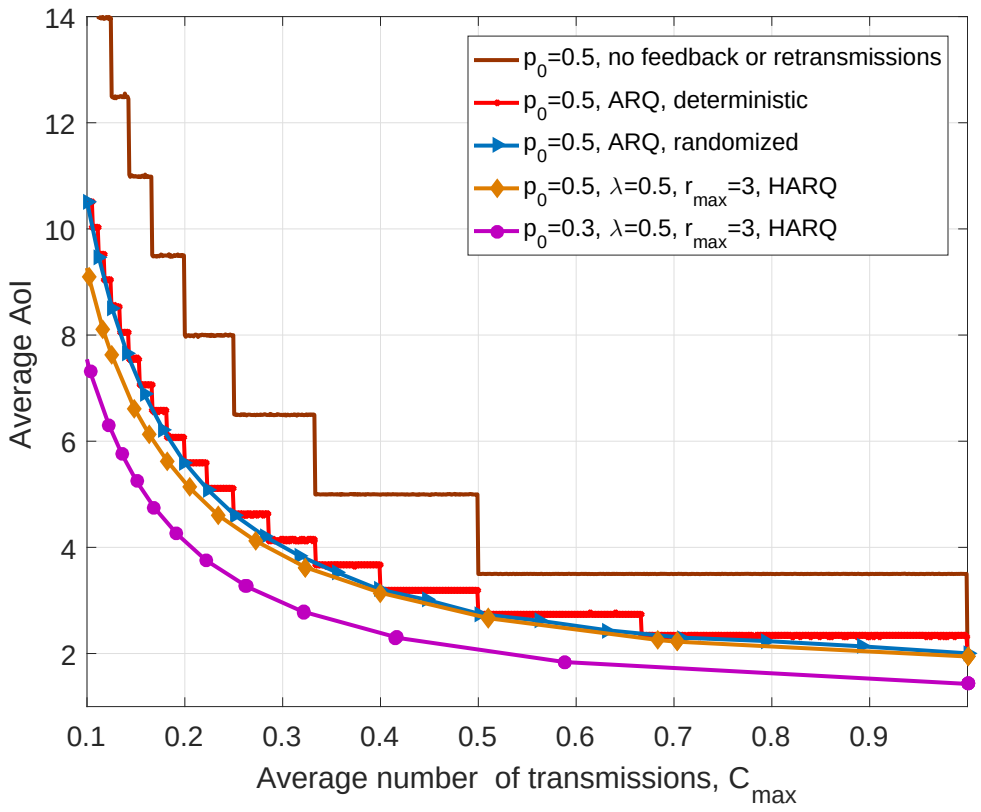


Figure 5. Expected average AoI as a function of C_{max} for ARQ and HARQ protocols for different p_0 values. Time horizon is set to $T = 10000$, and the results are averaged over 1000 runs.

to the randomized policy is the lower convex hull of the piecewise constant AoI curve for deterministic policies. For the same $p_0 = 0.5$, HARQ with $\lambda = 0.5$ improves only slightly over ARQ. Smaller p_0 results in a decrease in the average AoI as expected, and the gap between the AoIs for different p_0 values is almost constant for different C_{max} values.

More significant gains can be achieved from HARQ when the error probability decreases faster with retransmissions (i.e., small λ), or more retransmissions are allowed. This is shown in Figure 6. On the other hand, the effect of retransmissions on the average AoI (with respect to ARQ) is more pronounced when p_0 is high and λ is low.

Figure 7 shows the average AoI achieved by the HARQ protocol with respect to different p_0 and λ values for $r_{max} = 3$. Similarly to Figure 5, the gap between the average AoI values is higher for unreliable environments with higher error probability, and the performance gap due to different λ values are not observable for relatively reliable environments, for example, when

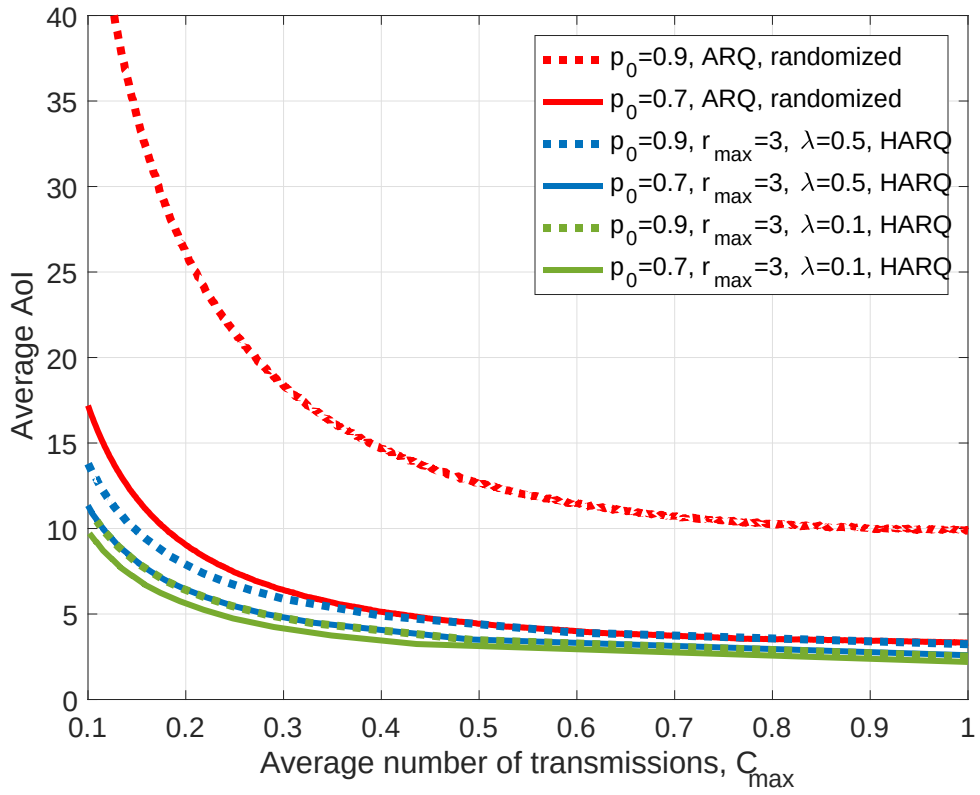


Figure 6. Expected average AoI with respect to C_{max} for ARQ and HARQ protocols for different p_0 and r_{max} values. Time horizon is set to $T = 10000$, and the results are averaged over 1000 runs.

$p_0 = 0.3$. The performance difference for different λ values (with a fixed p_0) is more pronounced when the average number of transmissions, C_{max}, is low, since then less resources are available to correct an unsuccessful transmission.

Figure 8 shows the evolution of the average AoI over time when the average-cost SARSA learning algorithm is employed. It can be observed that the average AoI achieved by Algorithm 2, denoted by *RL* in the figure, converges to the one obtained from the RVI algorithm which has *a priori* knowledge of $g(r)$. We can observe from Figure 8 that the performance of SARSA achieves that of RVI in about 10000 iterations. Figure 9 shows the performance of the two algorithms (with again 10000 iterations in SARSA) as a function of C_{max} in two different setups. We can see that SARSA performs very close to RVI with a gap that is more or less constant for the whole range of C_{max} values. We can also observe that the variance of the average AoI achieved by SARSA is much larger when the number of transmissions is limited, which also limits the

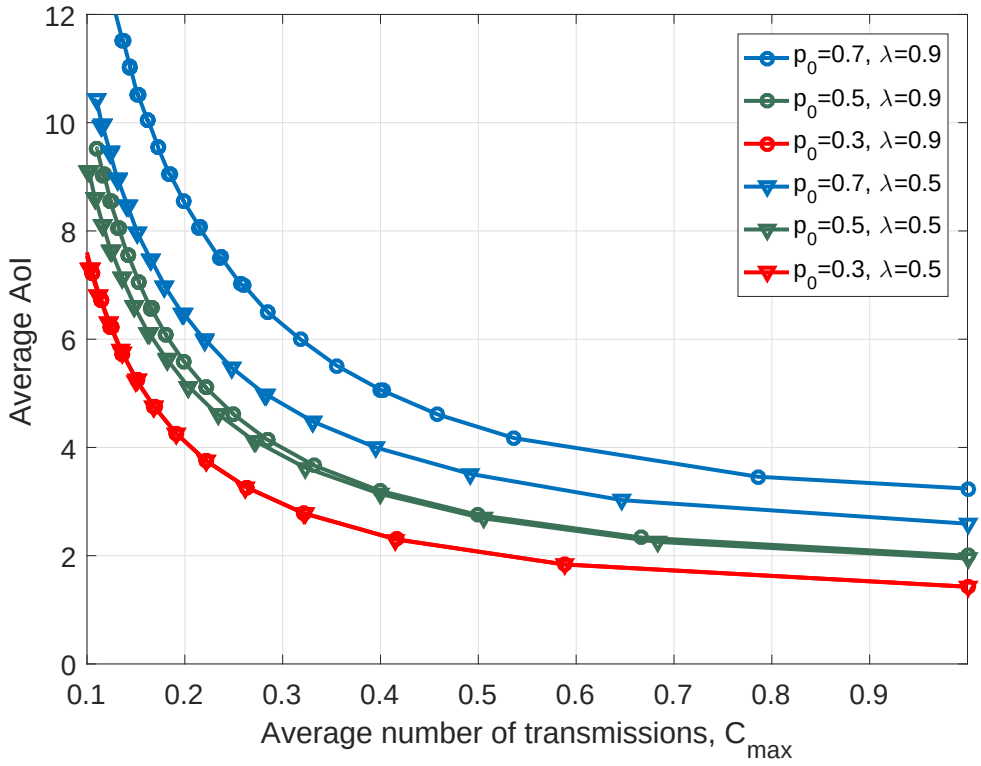


Figure 7. Expected average AoI with respect to C_{max} for HARQ protocols with different $g(r) = p_0\lambda^r$ values corresponding to different p_0 and λ values with $r_{max} = 3$. The time horizon is set to $T = 10000$, and the results are averaged over 1000 runs.

algorithm's learning capability.

VIII. Conclusions

We have considered a communication system transmitting time-sensitive data over an imperfect channel with the average AoI as the performance measure, which quantifies the timeliness of the data available at the receiver. Considering both the classical ARQ and the HARQ protocols, preemptive scheduling policies have been proposed by taking into account retransmissions under a resource constraint. In addition to identifying a randomized threshold structure for the optimal policy when the error probabilities are known, an efficient RL algorithm is also presented for practical applications when the system characteristics may not be known in advance. The effects of feedback and the HARQ structure on the average AoI are demonstrated through numerical simulations. The algorithms adopted in this paper are also relevant to different systems concerning the timeliness of information, and the proposed methodology can be used in other CMDP

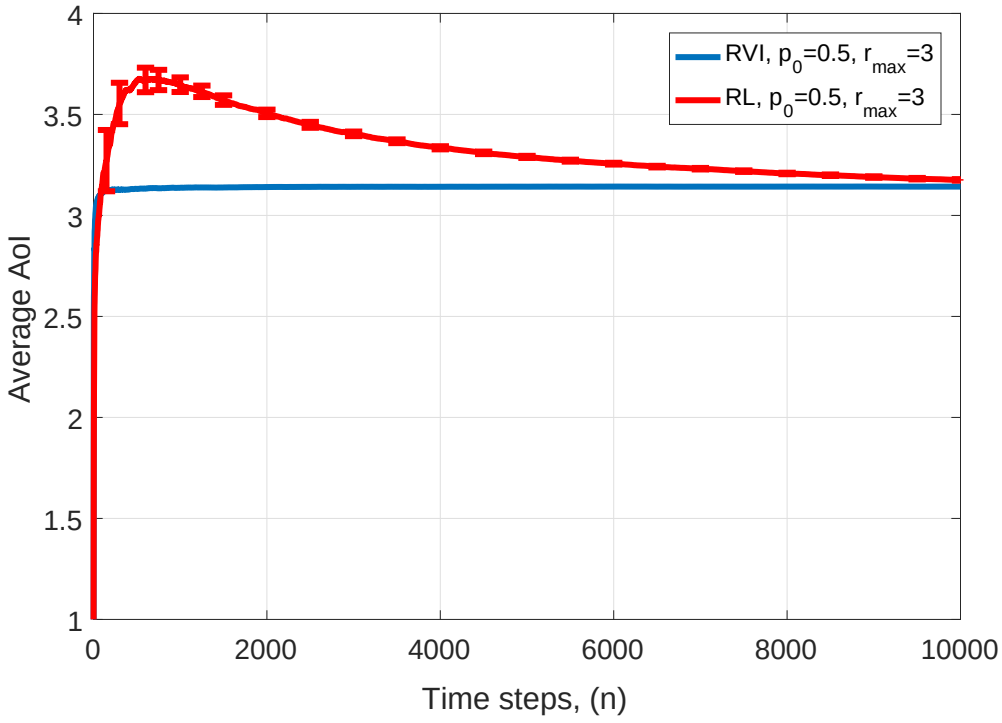


Figure 8. Performance of the average-cost SARSA for $r_{max} = 3$, $p_0 = 0.5$, $\lambda = 0.5$, $C_{max} = 0.4$ and $n = 10000$, averaged over 1000 runs (both the mean and the variance are shown).

problems. As future work, the problem will be extended to time-correlated channel statistics in a multi-user setting.

APPENDIX

A. *Proof of Lemma 1*

We are going to show that the decision to transmit ($a = \mathrm{n}$) is monotone with respect to the age δ, that is if $a^*(\delta^1) = \mathrm{n}$, then $a^*(\delta^2) = \mathrm{n}$ for all $\delta^2 \geq \delta^1$. By (8), this holds if $Q_\eta(\delta, a)$ has a *sub-modular* structure [26]: that is, when the difference between the Q functions is monotone with respect to the state-action pair (δ, a). We have

$$Q_\eta(\delta^1, \mathrm{n}) - Q_\eta(\delta^1, \mathrm{i}) \geq Q_\eta(\delta^2, \mathrm{n}) - Q_\eta(\delta^2, \mathrm{i}), \tag{18}$$

for any $\delta^2 \geq \delta^1$. From (12) and (13), for any $\delta > 0$, we have

$$Q_\eta(\delta, \mathrm{n}) - Q_\eta(\delta, \mathrm{i}) = \eta + (1-p)h_\eta(1) - ph_\eta(\delta + 1). \tag{19}$$

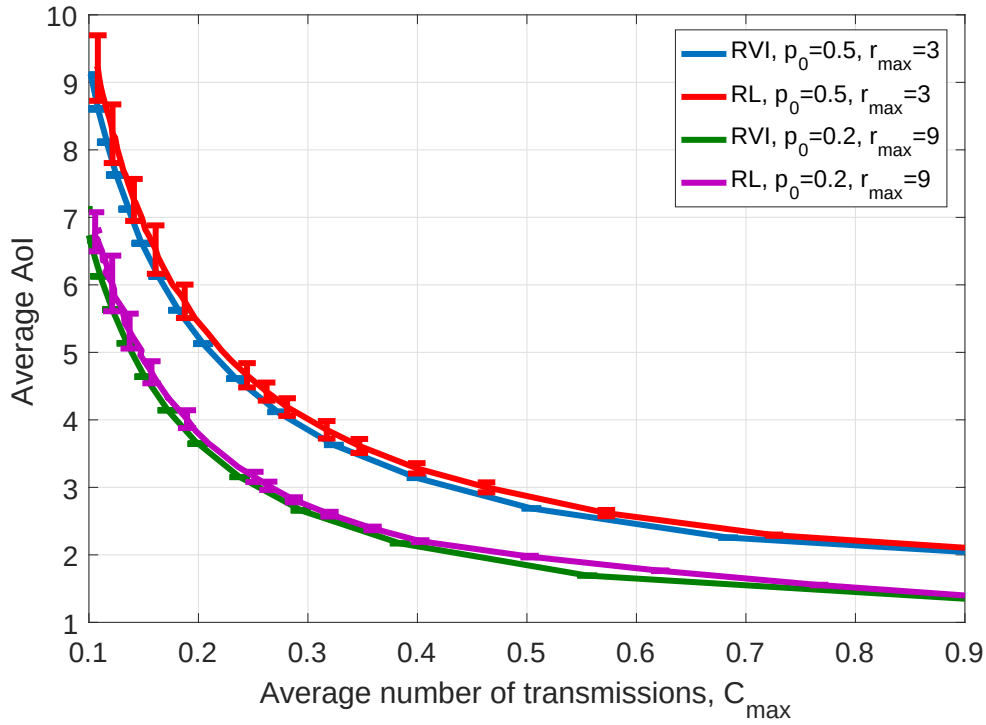


Figure 9. Performance of the proposed RL algorithm (average-cost SARSA) and its comparison with the RVI algorithm for $n = 10000$ iterations, and values are averaged over 1000 runs for different p_0 and r_{max} values when $\lambda = 0.5$ (both the mean and the variance are shown).

We can see that (18) holds if and only if $h_\eta(\delta)$ is a non-decreasing function of the age.

We compare the costs incurred by the systems starting in states δ^1 and δ^2 via coupling the stochastic processes governing the behavior of the system; that is, we assume that the realization of the channel behavior is the same for both systems over the time horizon (this is valid since channel states/errors are independent of the ages and the actions). Assume a sequence of actions $\{a_t^2\}_{t=1}^{\infty}$ corresponds to the optimal policy starting from age δ^2 for a particular realization of channel errors, and let $\{\delta_t^i\}$ denote the sequence of states obtained after following actions $\{a_t^2\}$ starting from state $\delta_1 = \delta^i$, $i = 1, 2$. Then, if $\delta^1 \leq \delta^2$, clearly $\delta_t^1 \leq \delta_2^t$ for all t. Furthermore, by

the Bellman optimality equation (6),

$$\begin{aligned}
h_\eta(\delta^1) &\le \delta_1^1 + \eta \cdot \mathbb{1}[a_1^2 \neq \mathrm{i}] - J_\eta^* + \mathbb{E}\left[h_\eta(\delta_2^1)\right] \\
&\le \delta_1^1 + \eta \cdot \mathbb{1}[a_1^2 \neq \mathrm{i}] - J_\eta^* + \mathbb{E}\left[\delta_2^1 + \eta \cdot \mathbb{1}[a_2^2 \neq \mathrm{i}] - J_\eta^* + \mathbb{E}\left[h_\eta(\delta_3^1)\right]\right] \\
&\ \ \vdots \\
&\le \mathbb{E}\left[\sum_{t=1}^{\infty}(\delta_t^1 + \eta \cdot \mathbb{1}[a_t^2 \neq \mathrm{i}] - J_\eta^*)\middle|\delta_1^1 = \delta^1\right] \\
&\le \mathbb{E}\left[\sum_{t=1}^{\infty}(\delta_t^2 + \eta \cdot \mathbb{1}[a_t^2 \neq \mathrm{i}] - J_\eta^*)\middle|\delta_1^1 = \delta^2\right] \\
&= h_\eta(\delta^2)\ .
\end{aligned}$$

This completes the proof of the lemma. □

B. Proof of Lemma 2

First we compute the steady state probabilities p_δ of the age δ for a given integer threshold Δ, for all $\delta = 1, 2, \ldots, N$. We have

$$p_\delta = \begin{cases} p_1 & \text{if } 1 \le \delta \le \Delta \\ p_{\delta-1}p = p_1 p^{\delta-\Delta} & \text{if } \delta \ge \Delta + 1\ . \end{cases}$$

Since $\sum_{\delta=1}^{\infty} p_\delta = 1$, we can compute the p_δ in closed form when N goes to infinity:

$$p_\delta = \begin{cases} \frac{1}{\Delta + \frac{p}{1-p}} & \text{if } \delta \le \Delta; \\ \frac{p^{\delta-\Delta}}{\Delta + \frac{p}{1-p}} & \text{otherwise.} \end{cases} \tag{20}$$

Then, the closed form of the expected Lagrangian cost function can be computed as:

$$\begin{aligned}
J_\eta &= \sum_{\delta=1}^{\infty} p_\delta(\delta + \eta\mathbb{1}[\delta \ge \Delta]) = p_1\left(\sum_{\delta=1}^{\Delta-1}\delta + \sum_{\delta=\Delta}^{\infty} p^{\delta-\Delta}(\delta+\eta)\right) \\
&= p_1\left(\frac{(\Delta-1)\Delta}{2} + \frac{\eta+\Delta}{1-p} + \frac{p}{(1-p)^2}\right).
\end{aligned} \tag{21}$$

Substituting p_1 from (20) and minimizing over Δ (by setting the derivative $\partial J_\eta/\partial\Delta$ to zero) yields that the optimal non-integer value of Δ is given by

$$\hat{\Delta}_\eta = \frac{\sqrt{2\eta(1-p)+p} - p}{1-p}\ .$$

Using that J_η is a convex function of Δ by (21), the optimal integer threshold Δ_η^* is either

$$\left\lfloor \frac{\sqrt{2\eta(1-p)+p}-p}{1-p} \right\rfloor \text{ or } \left\lceil \frac{\sqrt{2\eta(1-p)+p}-p}{1-p} \right\rceil .$$

Computing just the cost term from (21), we obtain the formula for C^Δ for any integer threshold Δ. □

REFERENCES

[1] E. T. Ceran, D. Gündüz, and A. György, "Average age of information with hybrid ARQ under a resource constraint," in *IEEE Wireless Communications and Networking Conference (WCNC)*, 2018.

[2] E. Altman, R. E. Azouzi, D. S. Menasché, and Y. Xu, "Forever young: Aging control in DTNs," *CoRR, abs/1009.4733*, 2010.

[3] S. Kaul, M. Gruteser, V. Rai, and J. Kenney, "Minimizing age of information in vehicular networks," in *IEEE Coms. Society Conf. on Sensor, Mesh and Ad Hoc Coms. and Nets.*, June 2011, pp. 350–358.

[4] S. Kaul, R. Yates, and M. Gruteser, "Real-time status: How often should one update?" in *Proc. IEEE INFOCOM,*, March 2012, pp. 2731–2735.

[5] P. Frenger, S. Parkvall, and E. Dahlman, "Performance comparison of HARQ with chase combining and incremental redundancy for HSDPA," in *Proc. IEEE Vehicular Technology Conf.*, vol. 3, 2001, pp. 1829–1833.

[6] V. Tripathi, E. Visotsky, R. Peterson, and M. Honig, "Reliability-based type ii hybrid ARQ schemes," in *IEEE Int'l Conf. on Communications,*, vol. 4, May 2003, pp. 2899–2903 vol.4.

[7] X. Lagrange, "Throughput of HARQ protocols on a block fading channel," *IEEE Communications Letters*, vol. 14, no. 3, pp. 257–259, March 2010.

[8] D. Gunduz, K. Stamatiou, N. Michelusi, and M. Zorzi, "Designing intelligent energy harvesting communication systems," *IEEE Communications Magazine*, vol. 52, pp. 210–216, 2014.

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

[10] I. Kadota, E. Uysal-Biyikoglu, R. Singh, and E. Modiano, "Minimizing age of information in broadcast wireless networks," in *Allerton Conf. On on Communication, Control, and Computing*, Sep. 2016.

[11] Y. P. Hsu, E. Modiano, and L. Duan, "Age of information: Design and analysis of optimal scheduling algorithms," in *IEEE Int'l Symposium on Information Theory (ISIT)*, June 2017, pp. 561–565.

[12] Y. Sun, E. Uysal-Biyikoglu, R. D. Yates, C. E. Koksal, and N. B. Shroff, "Update or wait: How to keep your data fresh," *IEEE Transactions on Information Theory*, vol. 63, no. 11, pp. 7492–7508, Nov 2017.

[13] K. Chen and L. Huang, "Age-of-information in the presence of error," in *IEEE Int'l Symposium on Information Theory (ISIT)*, July 2016, pp. 2579–2583.

[14] S. Mahadevan, "Average reward reinforcement learning: Foundations, algorithms, and empirical results," *Machine Learning*, vol. 22, no. 1, pp. 159–195, 1996.

[15] P. Parag, A. Taghavi, and J. F. Chamberland, "On real-time status updates over symbol erasure channels," in *IEEE Wireless Communications and Networking Conference (WCNC)*, March 2017, pp. 1–6.

[16] E. Najm, R. Yates, and E. Soljanin, "Status updates through M/G/1/1 queues with HARQ," in *IEEE International Symposium on Information Theory (ISIT)*, June 2017, pp. 131–135.

[17] R. D. Yates, E. Najm, E. Soljanin, and J. Zhong, "Timely updates over an erasure channel," in *IEEE Int'l Symposium on Information Theory (ISIT) (ISIT)*, June 2017, pp. 316–320.

[18] M. R. el Fenni, R. El-Azouzi, D. S. Menasche, and Y. Xu, "Optimal sensing policies for smartphones in hybrid networks: A POMDP approach," in *Int'l ICST Conf. on Performance Evaluation Methodologies and Tools*, Oct 2012, pp. 89–98.

[19] "Approved draft IEEE standard for local and metropolitan area networks corrigendum to IEEE standard for local and metropolitan area networks-part 16: Air interface for fixed broadband wireless access systems (incorporated into IEEE std 802.16e-2005 and IEEE std 802.16-2004/cor 1-2005 e)," *IEEE Std P802.16/Cor1/D5*, 2005.

[20] E. Altman, *Constrained Markov Decision Processes*, ser. Stochastic modeling. Boca Raton, London: Chapman & Hall/CRC, 1999.

[21] L. I. Sennott, "Constrained average cost Markov decision chains," *Probability in Eng. and Informational Sciences*, vol. 7, no. 1, p. 6983, 1993.

[22] ——, "Average cost optimal stationary policies in infinite state Markov decision processes with unbounded costs," *Operations Research*, vol. 37, no. 4, pp. 626–633, 1989.

[23] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. New York, NY, USA: John Wiley & Sons, 1994.

[24] D. P. Bertsekas, *Dynamic Programming and Optimal Control*, 2nd ed. Athena Scientific, 2000.

[25] R. S. Sutton and A. G. Barto, *Introduction to Reinforcement Learning*, 1st ed. Cambridge, MA, USA: MIT Press, 1998.

[26] D. M. Topkis, "Minimizing a submodular function on a lattice," *Oper. Res.*, vol. 26, no. 2, pp. 305–321, Apr. 1978.