What's My Age of Information Again? The Role of Feedback in AoI Optimization Under Limited Transmission Opportunities

Andrea Munari, *Senior Member, IEEE,* and Leonardo Badia, *Senior Member, IEEE*

***Abstract*—Real-time applications in the Internet of things (IoT) commonly require to schedule status updates from remote sensors to minimize age of information (AoI), a metric that captures the freshness of received data. Oftentimes, this problem is tackled assuming that sensors operate over an indefinite time horizon and can decide when to transmit data leveraging the knowledge of the current AoI level at the receiver, even when the communication channel is unreliable. Such a modeling approach, however, neglects some key aspects of most practical IoT systems, where the frequency of status reporting is limited due to resource constraints, such as energy limitations, and tracking the outcome of the updates would require additional consumption of resources to acquire a feedback. In this paper, instead, we investigate the optimal schedule of updates over a finite time horizon for a resource-constrained sensor that is allowed to perform a limited number of updates. We discuss the role of the feedback from the receiver, and whether it is convenient to ask for it whenever this causes additional energy consumption and consequently allows the transmission of a lower number of updates. We analytically identify regions for the feedback cost and the reliability of the channel where making use of feedback may or may not be beneficial. Our study covers both the *generate-at-will* case, in which a sensor can produce a fresh reading whenever it wants to communicate with the receiver, and an *exogenous* setting, where the transmitter cannot decide when new status updates are produced. The results highlight some interesting trade-offs, providing useful design hints for the protocol operation of IoT remote sensing systems.**

***Index Terms*—Age of Information; Internet of things; Data acquisition; Feedback; Sensor networks.**

I. Introduction

The problem of transmitting status updates so as to minimize age of information (AoI) at a receiver has gained considerable momentum in the recent literature [1]–[3]. In fact, this issue is relevant for real-time applications in the Internet of things (IoT), where timely tracking of a process is required to gain up-to-date perception and possibly take prompt intervention whenever needed [4], [5]. Using AoI as a performance metric retains a mathematical character in the analysis, while precisely describing the aspect of timeliness in system monitoring and actuation, and has been shown to provide profound insight in many settings [6].

However, most studies that focus on AoI minimization just consider AoI or related quantities through a long-term analysis, from the perspective of, e.g., average values, violation probabilities, or steady-state distributions [7]–[10]. We argue that practical systems often suffer from resource limitations and operate over definite and limited horizons. These aspects can be related to concrete limitations in terms of energy consumption or computational complexity, as well as properties of the application, which may for example require a task-oriented approach, or at the same time impose that the constraints are not just met in the long run, but over specific time windows. Just to give an example, technical and legislative limitations of sensor applications based, e.g., on the LoRa standard impose a constraint on the duty cycle of device activity, and said constraint must be met not just in the long run, as often considered in the literature, but specifically with a maximum number of temporal activity over a definite window [11].

Another aspect where the most common approach found in the literature may fall short concerns the presence of feedback and its exploitation. In fact, if transmissions are possibly subject to failures and the sender receives feedback about it, the scheduling of status updates can be optimized through proper online procedures. However, implementing feedback over a return channel comes at a cost for IoT devices, which is often neglected in spite of a potentially significant impact. For instance, the reception of an acknowledgment forces a terminal to listen to the channel and to process the incoming message, consuming energy instead of, e.g., going back to sleep mode. As found by early influential investigations on this matter [12], and confirmed by commercial chipset implementations, e.g., [13], receiving consumes comparable power to transmitting for many sensor network platforms. Given a finite energy budget, this may translate into a reduced number of transmissions that can be performed to deliver status updates.

To address these aspects, we consider a monitoring task of limited duration, which translates into finite-horizon optimization of the scheduling. Within this interval, the transmitter is allowed to send only a limited number of updates, due to hardware and cost constraints. For this scenario, we present multiple analytical formulations depending on the availability at the transmitter's side of feedback about the success of updates. We first consider an agnostic optimization that schedules the instants for transmitting an update so as to minimize the expected average AoI across the finite horizon; this is done

A. Munari acknowledges the financial support by the Federal Ministry of Research, Technology and Space of Germany in the programme of "Souverän. Digital. Vernetzt." Joint project 6G-RIC, project identification number: 16KISK022.

A. Munari is with the Institute of Communications and Navigation, German Aerospace Center (DLR), Weßling, Germany. email: andrea.munari@dlr.de

L. Badia is with the Dept. of Information Engineering, University of Padova, 35131 Padova, Italy. email: leonardo.badia@unipd.it

Part of this work [1] was presented at IEEE Globecom, 2022.

without exploiting the online information about the actual AoI, which is not available. Afterwards, we consider an AoI-aware optimization of the scheduling instants with an online approach, based on dynamic programming, assuming that the sender receives feedback on the outcome of each performed transmission [14]–[16]. This ultimately results in an analytical characterization of the regions where using the feedback is advantageous (or not), depending on the time horizon, the number of opportunities, the transmission success probability, and the feedback cost. Finally, this analysis is expanded from the case where status updates are "generated at will," which is a typical assumption in the literature [17], [18], to a condition of exogenous generations due to an external source, which covers a broader umbrella of scenarios.

Our main contributions are thus summarized as follows:

i) focusing on a generate-at-will setting, we provide an exact analysis of the average AoI under agnostic scheduling for a finite horizon and unreliable channel, which can be solved to obtain the optimal transmission pattern;
ii) we further derive the optimal transmission policies via dynamic programming for an AoI-aware scheduling over a finite horizon in the presence of feedback, accounting for a reduction in the number of transmission opportunities for the IoT device;
iii) we then move to the more challenging setup of exogenous, non-persistent updates, where AoI does not reset to zero, and extend the results above;
iv) we discuss in depth changes in both scheduled transmission patterns depending on the system parameters, which eventually leads to a comparison between an agnostic scheduler and an AoI-aware one with fewer transmission opportunities due to increased energy consumption.

All of these contributions result in useful guidelines for practical IoT systems for the transmission of timely updates under constrained scenarios.

To illustrate our study, we start in Section II by reviewing related works. Section III presents our model and notation. Section IV discusses first the case where the source can generate data at will, whereas Section V expands to the case with exogenous and non-persistent generation of data. In both these sections, we present numerical results. Finally, we draw the conclusions in Section VI.

II. Related Work

Since its proposal in the seminal work [19], many papers have addressed AoI as a performance evaluation metric, especially in the context of remote sensing for the IoT. This leads to different formal approaches, related to various degrees with the present paper [20]–[22].

The first investigations on this subject generally make use of queueing theory, to track AoI for different medium access and/or queueing disciplines [23], [24]. This is somehow orthogonal to our analysis, since we are instead interested in planning transmission at specific instants and evaluating their impact. Albeit queueing theory sheds light on the presence of preemption or processor sharing, the role of scheduling is usually absent. We remark that some papers use the term "scheduling" to denote a choice of one among multiple sources sharing the same channel [7], [10], but this is a problem inherently different from the one tackled in the present paper, where we instead consider a single source and the allocation of its transmission over time.

Some other works [4], [8], [16], [25]–[28] take this view, framing the choice of transmission instants as a linear program or searching for practical policies based on greedy or consecutive scheduling. However, in most of them, AoI is included in the optimization but is not the main direct objective. For example, it is included as a constraint (e.g., information should not become too obsolete [16]) or part of the objective function, in combination with other metrics especially when relating to energy expenditure [9], [29].

Even when the optimization goal directly relates to minimizing information staleness, most of the time it considers a long-term average AoI [17], which leads to a steady-state analysis, where a stationary policy is defined for a binary choice between transmission or idling, depending on the instantaneous age value, as well as other additional parameters, such as battery level when energy harvesting is also present [5], [10], [30], feedback delay [31], or channel state [28].

However, in a finite horizon, the scheduling problem is inherently non-stationary and made difficult by the principles of optimal stopping, further expanded to the overall number of transmission opportunities. In other words, the need for a timely update in a finite horizon is at odds with the limitations in the allowed rate of activity, so that transmissions must be properly distributed over the entire window. It is known that for this problem a Markovian approach valid at the steady-state is essentially different from a martingale solution, which in a finite horizon and discrete time can employ backward induction [32].

This paper specifically addresses the problem of AoI minimization for a finite time horizon, where the transmission of updates happens over an erasure channel. Our aim is to show how, under these conditions, receiving feedback about the state of the update (specifically, whether it was successful or subject to an erasure) improves the scheduling [15]. However, we also argue how a costly feedback may actually decrease the number of available transmissions. For these reasons, our analysis is connected to that of [14], where it was shown that the AoI computations can be related to geometric considerations. However, in that reference, the authors consider a different scenario, with an infinite horizon and each update consisting of multiple packets, each of which potentially subject to erasures, so that the choice boils down to whether reset the transmission of a new update in the presence of an erasure, or keep going with the current one. Another related work is [33], where a finite-horizon AoI minimization is performed but considering multiple sources, so that the scheduling problem in that context refers to the choice of the source that is allowed to transmit and feedback is necessary to guarantee network control.

The role of feedback is further explored for an energy harvesting source in [34], but still for an infinite horizon. Finally, [35] and [31], extend the analysis to the case of the feedback being present but possibly erroneous, or delayed feedback, respectively.

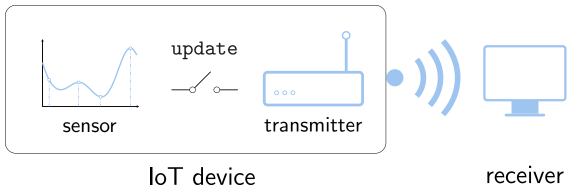


Fig. 1. In the considered setting, an IoT device sends samples of an observed process to a receiver over a wireless channel. Within the IoT device, samples are produced by a sensor and forwarded to a transmitter. The two entities are logically separate, and the latter may (generate at will) or may not (exogenous traffic) control when a new sample is produced by the sensor.

Our approach focuses not only on the finiteness of time horizon, but also on the impact of feedback [21], seen as a way to obtain an AoI-aware scheduling as opposed to a pre-planned pattern of status updates instants, but also as a further cause of energy expenditure that may be unavailable in massive IoT access [36], [37]. A preliminary version of this analysis was already presented by the authors in [1] but here we further expand it with theoretical results and also we relax the assumption of "generate at will."

III. System Model

Throughout our study we focus on the system illustrated in Fig. 1, where an IoT device communicates with a receiver (monitor) over a wireless channel. The device embeds two independent components: a *sensor* and a *transmitter*. The former produces updates, containing, e.g., readings of a physical process of interest, and forwards them to the transmitter. This, in turn, stores the latest incoming reading in a local buffer, and may decide to send it to the receiver at an appropriate time. In spite of its simplicity, the considered model is apt to capture the behavior of many IoT settings of practical relevance, in which the wireless communication components are fed with exogenous data, e.g., generated by external sensing blocks, and cannot control when new information is produced [2], [38].

For convenience, we assume the system to operate over slots of equal duration, set to be unitary. In other words, a slot is the atomic operational unit for communication protocols that will be studied.[1] At the beginning of each slot, the sensor produces a new update with probability u, and remains inactive otherwise. In the first case, the update is stored in a one-reading sized buffer at the transmitter, and contains a time stamp indicating the slot at which it was obtained. Accordingly, the transmitter always has only one message to potentially send, containing the latest reading generated by the sensor.[2] At any slot the IoT device may decide to attempt delivery of the available reading over the wireless channel by sending a packet, whose transmission is completed within the same time unit.[3] We tackle the operation of the system over a finite time horizon of n slots, and assume that the IoT device can perform at most $\mathsf{m} \ll \mathsf{n}$ transmissions within the considered timespan. The limitation is consistent with practical IoT settings, where the time spent in transmission may be capped to preserve battery, or due to normative arguments, e.g., to a maximum of 1% for LoRaWAN operating in the ISM band [36].

Each time a packet is sent, it is successfully decoded at the receiver with probability p, or lost due to channel impairments with probability $1-p$.[4] We model the outcomes of transmissions over different slots as independent and identically distributed, noting that extensions of the present analysis accounting for correlation or more advanced retransmission techniques are also possible, e.g., considering approaches such as [9], [39]. Whenever a transmission is performed, the receiver may provide *feedback* to the device, informing it of whether the message was successfully retrieved or not. When implemented, we will assume the feedback to be instantaneous and always successful.[5] On the other hand, the use of feedback entails a cost for the IoT device. For example, the reception of a message requires the terminal to remain active (as opposed to entering sleep mode), and decode the incoming packet, thus consuming energy. This aspect is often key in IoT systems: many commercial solutions foresee operations without acknowledgments, as done for instance in LoRaWAN by Class A devices sending unconfirmed messages [36], or disable the feedback channel completely to prolong battery life [42].[6]

To capture the impact of feedback, we resort to a simple yet significant model, introducing a *feedback cost coefficient* $\zeta \geq 0$. Accordingly, when feedback is used, the number of transmission opportunities the IoT device can employ is

$$\mathsf{m} = \left\lfloor \frac{\eta\, \mathsf{n}}{1+\zeta} \right\rfloor. \tag{1}$$

Within (1), $\eta < 1$ is the *duty-cycle*, which we define as the fraction of time the device can spend at most in transmission within the considered time horizon. In turn, the floor operator is taken to obtain an integer number of attempts. When no cost is undertaken ($\zeta = 0$), the maximum number of opportunities ($\lfloor \eta \mathsf{n} \rfloor$) can be used. Conversely, as ζ grows, the amount

[1] Setting a unitary value corresponds to normalizing both the slot time and the overall horizon to the basic protocol operations, and offers the advantage of presenting results which are not directly dependent on the specific implementation or transmission parameters (e.g., air-time of messages).

[2] The choice of a one-update size buffer implements a pre-emption policy with replacement in waiting [2], [37]. Its relevance becomes clear in view of the considered metric, i.e., age of information. Note indeed that, aiming at information freshness, transmitting an older update when newer data is available would only degrade performance.

[3] Note that a reading remains in the buffer even after having being sent, and is only replaced upon generation of a new one. This also means that the same reading may be transmitted multiple times.

[4] The channel statistics are assumed to be known at the device. In typical IoT systems, these can be estimated through feedback, or can be communicated by the receiver during logon procedures or with sporadic downlink transmissions.

[5] Incidentally, we note that this is a common assumption in the literature, as the feedback packets are usually limited in size and possibly sent over a different channel. If, similar to the direct channel, errors on the feedback result in independent erasures where the return message is lost, one can adopt a timeout policy and account for this loss by increasing the success probability [40]. Even when they are the result of collisions over a common feedback channel, they are still tractable [41].

[6] While turning off feedback may be dictated by energy saving reasons, this is not the only motivation. For example, there may be scheduling of other tasks involved [7], [18]. Energy saving may also lead to reducing the frequency of sampling [26], which would cause fresh information not to be available at will, which is the reason why in the following we consider both cases of AoI being reset to 0 or to a value greater than 0.

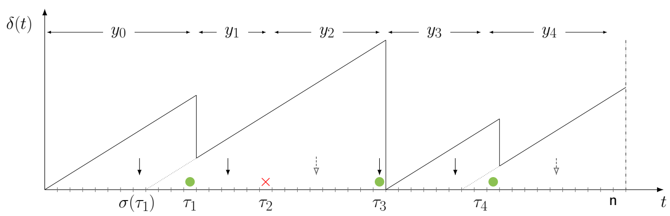


Fig. 2. Example of time evolution of $\delta(t)$ over a time horizon of n slots. A total of $\mathsf{m} = 4$ transmission opportunities is assumed, performed at time instants τ_i, $i \in \{1, \ldots, 4\}$. Green dots (●) denote successful transmissions, resetting the AoI value to $\tau_i - \sigma(\tau_i)$, whereas the red cross (×) indicates a sent message that was not delivered. Vertical arrows report instants at which the sensor produces new updates and makes them available to the transmitter. In particular, the two dashed arrows show updates which are generated but not transmitted. In the first (leftmost) case, the update is superseded by a newer one, generated right before the transmission at τ_3. Instead, the final generated reading is not sent as all the available transmissions have already been consumed.

of transmissions the IoT node can perform is reduced: for example, when receiving a message has the same cost of sending one ($\zeta = 1$), only $\lfloor \eta \mathsf{n}/2 \rfloor$ updates can be sent.

In this setting, we are interested in understanding the ability of different transmission strategies towards maintaining an up-to-date perception at the receiver of the process monitored by the device. To gauge this, we resort to the notion of age of information (AoI) [43]. Specifically, at any time slot t in $\{0, \ldots, \mathsf{n}\}$, we define the instantaneous (current) AoI as

$$\delta(t) := t - \sigma(t) \tag{2}$$

where $\sigma(t)$ is the time-stamp of the last message received by the monitor from the IoT device as of time t. Without loss of generality, we set $\delta(0) = 0$, and consider that the instantaneous AoI is reset to 0 whenever a new update is delivered in the same time slot it was generated. An example of the time evolution of $\delta(t)$ is reported and discussed in Fig. 2. Leaning on the definition in (2), we are interested in the average AoI

$$\Delta := \frac{1}{\mathsf{n}} \sum_{t=0}^{\mathsf{n}-1} \delta(t). \tag{3}$$

The expression in (3) depends not only on the times at which transmissions are performed, but also on the realizations of the r.v.s describing the readings generation and the packet delivery probability. In the remainder we will then target the *expected average AoI*

$$\bar{\Delta} := \mathbb{E}[\Delta]$$

where the expectation of the r.v. Δ is taken over all the aforementioned components.

A. Transmission strategies

The problem of scheduling packet transmissions [17] over the time horizon n so as to minimize $\bar{\Delta}$ is in general not trivial, as the limited number of attempts may be prone to losses, and the transmitter may not be aware of when and if new (fresh) readings will become available from the sensor. To tackle this setting, we consider different approaches, inspired by configurations and capabilities that are typical of IoT systems.

TABLE I
MAIN SYSTEM PARAMETERS

PARAMETER	MEANING
n	time horizon duration [slot]
m	number of transmission opportunities
η	duty-cycle
ζ	feedback cost coefficient
p	transmission success probability
τ_i	slot index of i-th transmission attempt
y_i	time between $(i-1)$-th and i-th transmission attempt

- *Offline scheduling (OFF)*: in this case, the transmitter relies on a pre-computed schedule, e.g., sending packets at regular intervals. As such, it is completely oblivious of whether its attempts are successful or not (no feedback is required), as well as of when the readings from the sensor are produced. While suboptimal, the approach can be appealing in view of its simplicity, as it requires no intelligence at the device, and represents a relevant baseline benchmark.
- *Online scheduling, zero-feedback (ZF)*: for this policy, we assume that no feedback is implemented, yet the transmitter is aware of the time elapsed since the generation of the sensor reading available in its queue, and can decide at run-time when to attempt delivery. For example, sending a message as soon as it is generated may be more convenient with respect to attempting transmission of an already stale reading, which, if successful, may reset AoI to a higher value. This solution epitomizes settings in which feedback is not implemented, e.g., unconfirmed messages in LoRaWAN [36].
- *Online scheduling, with feedback (FB)*: finally, we consider the possibility to rely not only on the knowledge of when updates were produced, but also on the outcome of transmission attempts through receiver feedback. In this configuration, the transmitter is aware at all times of the current level of AoI experienced at the monitor, as well as of how this would change in case of a successful packet delivery. Thus, it performs online decisions on whether and when to access the wireless channel. The strategy is characterized by a trade-off between the stronger ability to adapt and the fewer transmission attempts that may be available in view of the feedback cost ζ.

For the online approaches, we assume that the statistics of update generation rate, u, are known to the transmitter.

B. Notation

In the remainder, we denote as $\tau_\ell \in [0, \mathsf{n})$, $\ell \in \{1, \ldots, \mathsf{m}\}$ the instants at which the IoT device performs its transmission attempts. Accordingly, the time horizon is partitioned into $\mathsf{m} + 1$ intervals y_ℓ, $\ell \in \{0, \ldots, \mathsf{m}\}$, such that $y_\ell = \tau_{\ell+1} - \tau_\ell$, with $\tau_0 = 0$ and $\tau_{\mathsf{m}+1} = \mathsf{n}$ for consistency. Following this notation, an instance of transmission schedule implemented by the IoT device is completely specified by the vector $\boldsymbol{y} = [y_0\, y_1 \ldots y_{\mathsf{m}}]$. An example is illustrated in Fig. 2. The relevant notation is also summarized in Tab. I.

IV. The Generate-at-Will Case

As a starting point for our discussion, we consider the case $u = 1$, aiming to identify some fundamental trade-offs in a simpler setup. The configuration corresponds to the sensor producing a new reading at each slot, or, equivalently, to the transmitter having always a fresh update to transmit, i.e., a message that would reset the current AoI to its minimum value if decoded. The setting, often referred to as *generate-at-will*, is representative of situations in which the IoT device can drive the operation of the sensor, and has received a large deal of attention in AoI studies in the literature, e.g. [43] and references therein.

A. Offline scheduling (OFF)

We begin by providing the following result, which holds in the absence of feedback.

Proposition 1. *In the generate-at-will case ($u = 1$), the expected average AoI of an offline schedule $\boldsymbol{y}$ is given by*

$$\bar{\Delta}(\boldsymbol{y}) = \frac{1}{\mathsf{n}} \sum_{\ell=0}^{\mathsf{m}} \left[\frac{y_\ell(y_\ell+1)}{2} + \sum_{i=0}^{\ell-1} y_\ell y_i (1-p)^{\ell-i} \right]. \quad (4)$$

Proof. See Appendix A. □

Leaning on this result, the problem of finding the optimal offline schedule can be directly formalized as

$$\min_{\boldsymbol{y}} \ \bar{\Delta}(\boldsymbol{y})$$
$$\text{s.t.} \ \sum_{\ell=0}^{\mathsf{m}} y_\ell = \mathsf{n}$$

Applying a continuous relaxation, the minimum AoI can be found by nulling the gradient $\nabla\bar{\Delta}(\boldsymbol{y})$, which, in the specific coordinates y_is, corresponds to setting the first-order partial derivatives of (4) to 0, i.e.,

$$\frac{\partial \bar{\Delta}(\boldsymbol{y})}{\partial y_\ell} = 0 \qquad \forall \ell. \quad (5)$$

Observing that, by definition, $y_{\mathsf{m}} = \mathsf{n} - \sum_{\ell=0}^{\mathsf{m}-1} y_\ell$, (5) leads after simple manipulations to

$$\Big(2y_\ell - \mathsf{n} + \sum_{i\neq\ell} y_i\Big) \left[1 - (1-p)^{\mathsf{m}-\ell}\right] + \sum_{i\neq\ell} y_i \left[(1-p)^{|\ell-i|} - (1-p)^{\mathsf{m}-i}\right] = 0$$

for all $\ell \in \{0, \dots, \mathsf{m}\}$. Thus, we obtain a full-rank system of m linear equations in m unknowns, whose solution offers the optimal transmission times sought in the absence of feedback. This can be easily obtained with standard tools, whose computational complexity, even without leveraging sparsity properties, is at most $\mathcal{O}(\mathsf{m}^3)$. Note that the corresponding slot indexes τ_ℓ are then obtained by rounding the solution to the closest integer, leading to some approximations, which become negligible for the typical case of large n and $\mathsf{m} \ll \mathsf{n}$.

Remark: In the generate-at-will case, the online, zero-feedback (ZF) transmission scheme collapses into the offline (OFF) approach, as no additional benefit can be leveraged by knowing at run-time the generation time of the updates. Hence, ZF will not be further discussed in this section.

B. Online scheduling, with feedback (FB)

Even though the number of transmission opportunities may decrease, the presence of feedback can offer an advantage to the optimization. Indeed, the IoT device can achieve a more efficient schedule of the status updates, dynamically adapting the transmission instants based on the outcome of already performed attempts, and thus on the current AoI. The optimal solution in such conditions can be found following a dynamic programming approach. The problem is classically cast on defining a state, a control vector, and a noise component [44].

To this aim, we describe the state of the system at the start of slot $\ell \in \{0, \dots, \mathsf{n}\}$ as $x(\ell) = \big(\delta(\ell), m(\ell)\big)$, where $\delta(\ell)$ is the instantaneous AoI at time ℓ, whereas $m(\ell) \in \{0, \dots \mathsf{m}\}$ is the number of transmission opportunities still available at time ℓ. The state is initialized as $x(0) = \{0, \mathsf{m}\}$. In turn, the control of the system $c(\ell)$ results in a binary choice on whether to transmit over the ℓ-the slot, while the noise component is completely captured in the generate-at-will case by the success probability p. With these conventions, the system evolves from $x(\ell)$ as

- $x(\ell+1) = \big(\delta(\ell)+1, m(\ell)\big)$ if no update is attempted over slot ℓ (i.e., $c(\ell)$=0). In the absence of transmission, the AoI increases by a slot duration, and the same number of transmissions remain available to the IoT device;
- $x(\ell+1) = \big(\delta(\ell)+1, m(\ell)-1\big)$ if $\mathsf{m}(\ell) > 0$, the device sends an update during slot ℓ (i.e., $c(\ell)$=1), but it is unsuccessful, which happens with probability $1-p$;
- $x(\ell+1) = \big(0, m(\ell)-1\big)$ if $m(\ell) > 0$ and the sensor sends instead a successful update at time ℓ that resets the AoI, which happens with probability p.

Leaning on these definitions, the FB strategy is defined by finding the optimal control policy $c(\ell) = \mu_\ell(x(\ell), p)$ to apply at any state $x(\ell)$, i.e. the strategy that minimizes the expectation over the time horizon n of an instantaneous cost $g_\ell\big(x(\ell), c(\ell), p\big) = \delta(\ell)$. This is proven by the following proposition.

Proposition 2. *For the generate-at-will case, the FB scheme operates using in each $\ell \in \{0, \dots, \mathsf{n}-1\}$ the optimal policy*

$$\mu_\ell\big(x(\ell), p\big) = \mathbb{1}\big[(1-p)\, R_{\ell+1}\big(\delta(\ell)+1, m(\ell)-1\big) + pR_{\ell+1}\left(0, m(\ell)-1\right) < R_{\ell+1}\big(\delta(\ell)+1, m(\ell)\big)\big]$$

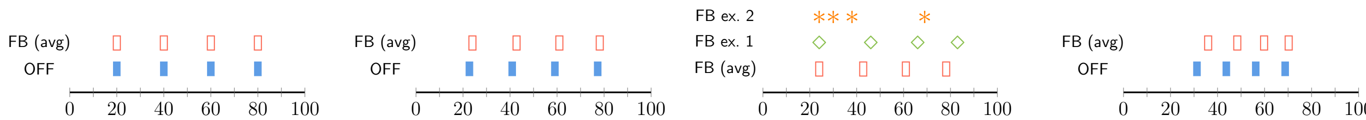
where

$$R_\ell\big(\delta(\ell), m(\ell)\big) = \sum_{i=\ell}^{\mathsf{n}} g_\ell\big(x(i), \mu_i\big(x(i)\big), p\big) \quad (6)$$

and $\mathbb{1}[\cdot]$ is the indicator function.

Proof. The result follows by exploiting Bellman's optimality condition [44]. This implies that the optimal choice at time ℓ when the system state is $x(\ell)$ is made by comparing the two alternatives (to transmit or not to transmit), based on the current reward and the expected future evolution of the system. Depending on this expected reward being higher when transmitting or not, $\mu_\ell\big(x(\ell), p\big)$ is set to 1 or 0, respectively.

Since the horizon is finite, the instantaneous choice is optimal at the end of the horizon, and also at every previous instant if the instantaneous best decision is made and then



(a) $p = 1$ (b) $p = 0.8$ (c) $p = 0.4$

Fig. 3. Example of transmission schedules for $\mathsf{n} = 100$ slots, $\mathsf{m} = 4$, for different values of the success probability p, generate-at-will case. Filled rectangles represent the transmission instants with offline optimization (OFF), whereas empty red rectangles denote the *average* instants obtained with the online scheme with feedback (FB). The right plot within subfigure (b) also reports two instances of online schedule based on the outcomes, obtained respectively with 4 consecutive successful transmissions (◇) or 2 failures followed by 2 successes (∗).

the following ones are according to the previously computed optimal policy. Indeed, it holds that, if the optimal policy is described by $\mu_0, \mu_1, \ldots, \mu_{\mathsf{n}-1}$, then for any value of an intermediate state $x(\ell)$ at time $\ell \in \{0, \ldots, \mathsf{n}-1\}$ occurring with positive probability, the minimizing policy for the residual cost from ℓ till n is $\mu_\ell, \ldots, \mu_{\mathsf{n}-1}$. □

As clarified by Prop. 2, the optimal control at time ℓ is achieved by making the decision that minimizes an expected total cost equal to the AoI, assuming future decisions are optimally made and averaging over channel errors. Remarkably, the only actions for the border cases $x(\mathsf{n}-1)$ with any $\mathsf{m}(\mathsf{n}-1) > 0$ and for any state $x(\ell)$ with $\mathsf{m}(\ell) = 0$ are to transmit and not to transmit, respectively, so one can start by defining μ for these cases and proceed backwards to find the optimal online scheduling for all reachable states at every ℓ.

In terms of computational complexity, it is worth noting that finding the optimal policy has a space and time complexity of $\mathcal{O}(\mathsf{mn}^2)$, since the state $x(\ell)$ contains n entries, each with $\mathsf{m} \cdot \mathcal{O}(\mathsf{n})$ possible values and is sequentially filled. This value can be lowered if high AoI values in $\delta(\ell)$ are capped. Also, this computation is just computed preliminarily, not at run-time.

C. Preliminary remarks

Initial insights on the schedules obtained with the OFF and FB policies can be obtained by looking at the examples reported in Fig. 3. The plot shows time instants at which transmissions are performed under different system settings and for specific realizations of the involved random outcomes. In all cases, we assume that $\mathsf{m} = 4$ updates can be scheduled over a time horizon of $\mathsf{n} = 100$ slots. In the presence of a perfectly reliable channel ($p = 1$), Fig. 3a shows that the optimal update instants are uniformly spread over the time horizon [34], and, as expected, there is no difference between a stateful and a stateless optimization, since there is anyway no need for feedback. In Fig. 3b, the probability of success is set to $p = 0.8$ and a difference appears between the offline and online scheduling. In particular, the scheduling instants for the OFF policy shift towards the center of the window of interest. For what concerns the case with feedback, the figure shows the average positions of the scheduling instants, since they clearly depend on the specific realization of the channel. The average position of the updates across the time window is slightly postponed for the online optimization, since the availability of the feedback can be better exploited and this allows to intervene even at a later stage. The same trend is confirmed in Fig. 3c for a lower success probability $p = 0.4$.

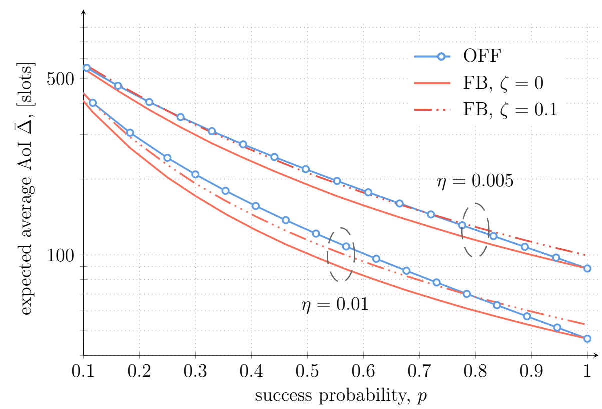


Fig. 4. Minimum $\bar{\Delta}$ vs. success probability for different values of m, corresponding to a duty cycle η of 1% and 0.5%. Circle-marked lines represent the performance of the OFF strategy, whereas plain solid lines the results of the FB scheme with cost-free feedback. The behavior of FB with costly feedback $\zeta = 0.1$ is shown by dash-dotted lines. In all cases, n=1600.

For $p = 0.8$, an extra plot in Fig. 3b shows also a comparison of practical realizations of the schedule in the presence of feedback, depending on whether the updates succeed or fail. In more detail, we compare a case where all updates are successful (example 1, marked as ◇) with one where the first two transmissions fail (example 2, marked as ∗). As visible from the plot, in the latter case, subsequent transmissions are scheduled much earlier to counteract the missing updates due to the undergone channel failures.

D. Performance evaluation

To gauge the role of feedback in the generate-at-will setting, we report and discuss some key trends of interest. Unless otherwise specified, all numerical results have been obtained considering a time horizon of $\mathsf{n} = 1600$ slots, and assuming $\mathsf{m} = 16$ transmissions available to the device when operating without resorting to feedback. Such a configuration is inspired by practical IoT systems such as LoRaWAN, where duty cycles in the order of $\eta = 0.01$ are typical for operations in the ISM band [36].

We start by considering Fig. 4, which shows the average AoI achieved by optimizing the transmission times over the time horizon, reported against the probability of success. The behavior in the absence of feedback (OFF) is represented by circle-marked lines, whereas plain solid lines were obtained considering the FB policy under a cost-free feedback (i.e., $\zeta = 0$). Finally, dash-dotted lines refer to the FB strategy when a costly feedback is implemented ($\zeta = 0.1$), resulting in a reduction of the number of available transmissions from the device to the monitor. For the OFF case, the average AoI was directly computed through the minimization of (4). Conversely, in the presence of feedback, the dynamic optimization process described in Section IV-B was applied to derive the optimal, exact, schedule. To compute the average AoI, which depends on the channel realization and the process of arrivals, we resorted as customary to a Montecarlo methodology.

First, consider the case $\eta = 0.01$, corresponding to $\mathsf{m} = 16$ transmission opportunities, and focus on operations without feedback. As expected, AoI decreases as the success probability increases, thanks to the more frequent delivery of status updates, reaching the minimum value $\mathsf{n}/[2(\mathsf{m}+1)]$ when $p = 1$. In turn, when feedback is available at no cost (FB, $\zeta = 0$), an improvement emerges. In this case, the possibility to adapt the upcoming transmission times based on the outcome of the current attempt is beneficial, enabling a reduction of the AoI of up to 20%. We note that the achievable gain is larger for moderately low success probability values, whereas the two policies behave similarly (and eventually coincide) when p is either very high or very low.

Notably, things change significantly when the cost entailed by feedback is accounted for. In fact, while for lower success rates the use of feedback continues to be beneficial, there exist values of p (e.g., $p > 0.8$ in the considered example) for which an offline optimization of the transmission times emerges as the policy of choice. In such conditions, the availability of fewer delivery attempts – induced by employing feedback procedures – more than counterbalances the positive effects of dynamically adapting the transmission times, rendering the no-feedback approach more effective in terms of AoI. The result offers a first important and non-trivial insight, pinpointing how the use of a return channel shall be carefully considered in practical IoT systems.

Similar trends also emerge when the device is allowed to access the channel less often, e.g. to save energy (curves with duty cycle $\eta = 0.005$ in the plot). In this case, higher values of AoI are attained by all the strategies considered, as a consequence of the less frequent transmissions. Interestingly, when a cost is to be undergone ($\zeta = 0.1$), the OFF strategy already starts to outperform the FB scheme for values of $p > 0.7$. This effect reveals how the reduction in the number of available attempts becomes critical as transmissions are sporadic and may be lost due to channel impairments.

The impact of the feedback cost is further explored in Fig. 5, where we report the ratio of the expected average AoI achieved with the FB policy to that of the optimal OFF schedule. Values larger than 1 denote thus better performance *without* feedback, and trends are shown against the feedback cost ζ. Three different values of success probability are considered, namely $p \in \{0.75, 0.9, 0.99\}$, as identified by different markers, and in all cases the duty cycle is set to $\eta = 0.01$. The trends highlight how the use of a costly feedback quickly becomes detrimental for the practical values of success probability reported in the figure. From this standpoint, for instance, worse AoI performance is attained already for $\zeta = 0.1$ when $p = 0.9$. Furthermore, AoI quickly deteriorates with the feedback cost.

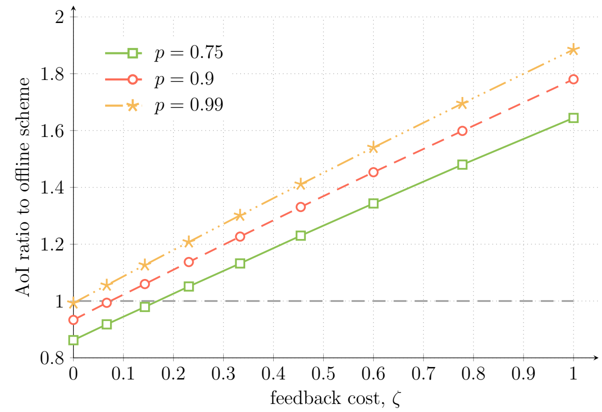


Fig. 5. Ratio of the expected average AoI with FB to the one obtained with optimized OFF vs feedback cost ζ for different success probability p. Values above 1 denote better performance *without* feedback. In all cases, $\mathsf{n} = 1600$, $\eta = 0.01$.

Interestingly, for $\zeta = 1$, corresponding to a practically relevant condition in which the reception of feedback may entail an energy cost similar to the one of a transmission, the expected average AoI almost doubles in comparison to the simpler no-feedback approach when $\zeta = 0.9$, as a result of the much tighter restrictions on the transmission opportunities (equivalent to operating at a duty cycle $\eta = 0.005$).

These remarks trigger a fundamental system design question, highlighting the need to understand when the implementation of feedback can lead to better performance. We tackle this aspect in Fig. 6, which identifies in the (p, ζ) parameter plane the region where feedback shall or shall not be used from an expected average AoI minimization standpoint. The diagram offers a simple yet useful tool, quickly identifying the most suitable strategy to be followed under any operating condition. The importance of carefully considering the cost of feedback clearly emerges. In particular, the implementation of a return channel leads to an AoI reduction under harsh channel conditions (i.e., low probability of success), even at the expense of the availability of fewer update delivery attempts. Conversely, a simpler offline optimization is to be preferred when more reliable transmissions can be performed.

We remark that these trends hold when the IoT device always has fresh data to send, i.e., it can control the production of sensor reading. We will see in the next section how the situation changes under exogenous traffic. Incidentally, we also note that the staircase shape of Fig. 6 stems from the fact that a change in performance is only observed when the cost leads to a reduction of the number of available transmissions, as per (1). In this sense, increasing the frequency of transmission or the window length would naturally smooth the curve, without however altering the fundamental trends that were presented.

To conclude our discussion, we study the impact of the time horizon over which the IoT device has to operate. To this aim, we report in Fig. 7 the expected average AoI as a function of n for the OFF and FB policy (with $\zeta = 0$). Results were obtained for $p = 0.8$ and with a duty-cycle of $\eta = 0.01$, i.e., proportionally increasing the number of available transmission

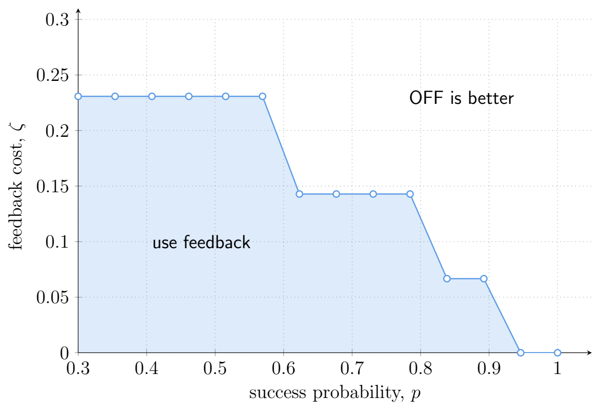


Fig. 6. (p, ζ) pairs for which the use of feedback is beneficial (shaded area) or not (blank area) in terms of attainable expected average AoI for the generate-at-will setting. In all cases, $\mathsf{n} = 1600$, $\eta = 0.01$.

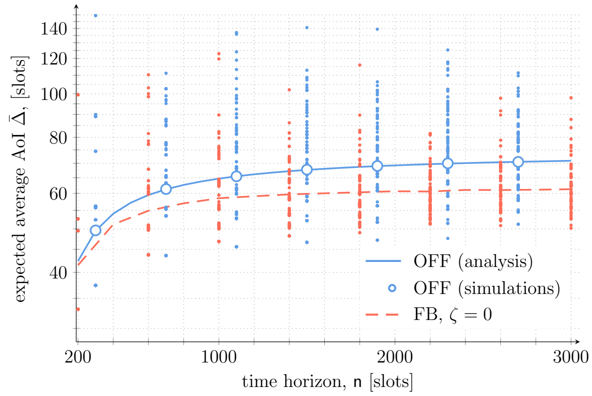


Fig. 7. Average AoI vs horizon length, n, in the generate-at-will setting. Lines report analytical results, whereas circle markers show the average AoI of the offline policy obtained by means of detailed Montecarlo simulations. The clouds of points show the average AoI value obtained for the offline (blue) and ideal feedback (red) policies for different realizations of the channel (i.e., packet success/failures). In all cases, $p = 0.8$, $\eta = 0.01$.

opportunities as n grows. In the plot, the solid blue and dashed red lines denote the value of $\bar{\Delta}$ attained with OFF and FB, respectively. Following the same color-code, the figure also reports clouds of points obtained for some specific values of n. Each point, in turn, represents the average AoI value obtained over a specific realization of the channel conditions (i.e., of packet losses) when applying the corresponding transmission policy. In other words, the obtained scattered bars are an indication of the dispersion of the actual values of Δ around its statistical average $\bar{\Delta}$. Two main take-aways emerge from the plot. First, the expected average AoI grows initially with n, and later stabilizes. The effect stems from having $\delta(0)=0$, which reflects in small AoI values at the beginning of the horizon. This has a stronger effect on the mean value when the overall time window is short, and its impact vanishes for larger n. More interestingly, the fact that the metric tends to settle confirms the broad applicability of the more detailed results we presented so far for the case $\mathsf{n} = 1600$. As a second remark, the dispersion of the average AoI is more pronounced in proportion for low n, so that larger fluctuations around the predicted value shall be expected at the receiver when the IoT device has to operate over short horizons. Indeed, for a given duty-cycle, a lower n corresponds to fewer transmission opportunities, and thus to fewer occasions to maintain a lower AoI in case of an uplink failure. For instance, the leftmost cloud of points for the OFF scheme in Fig. 7 refers to the setting $\mathsf{n} = 300$, $\mathsf{m} = 3$. In this case, when all three messages are lost, the AoI is never reset, leading to an average value of 150 (highest marker in the plot). Finally, Fig. 7 also shows circle markers. These refer to the mean of the average AoI obtained for the OFF policy, obtained by means of detailed Montecarlo simulations. The perfect match exhibited with the blue line (obtained analytically via (4)) validate the correctness of the mathematical derivations.

V. The Exogenous Traffic Case

We complement our study by considering the more general case $u < 1$. This setup in terms of update generation is often referred to as *exogenous traffic*, as the transmitter cannot directly control when new readings are produced [2]. Such conditions render the scheduling problem at hand significantly more challenging, as the device has to decide whether to send a reading that may already be stale - potentially with a small reduction of AoI at the receiver - or refrain from doing so and await for a fresh update - yet allowing AoI to grow in the meantime. As done for the generate-at-will setting, we first characterize the expected average AoI for the OFF policy, and later tackle the online strategies.

A. Offline scheduling (OFF)

Proposition 3. *In the exogenous case ($u < 1$), the expected average AoI of an offline schedule $\boldsymbol{y}$ is given by* (7)*, reported at the top of next page.*

Proof. See Appendix B. ☐

Clearly, the obtained formulation of $\bar{\Delta}$ falls back to the generate-at-will case by setting $u = 1$. More interestingly, (7) captures the effect of exogenous generation, highlighted by the terms under brace (a), and pinpoints how their effect is not independent but rather intertwined with the probability of successfully delivering an update. The exact result provided by Prop. 3 can readily be computed offline for any schedule $\boldsymbol{y}$. Leaning on this, a minimization problem for the expected average AoI can be cast following the same approach Sec. IV-A, leading to a system of equations, which, albeit more involved and not reported here, can be solved via standard numerical tools to obtain the optimal schedule for any given (p, u) pair.

B. Online scheduling, with feedback (FB)

Let us first consider the situation in which feedback on the outcome of a transmission is available. In this case, it is once more possible to enact an online scheduling, along the same lines of Sec. IV-B, i.e., through a dynamic programming approach, applying a binary control $c(\ell)$.

However, the state of the system in slot $\ell \in \{0, \ldots, \mathsf{n}\}$ is enriched to $x(\ell) = \big(\delta(\ell), m(\ell), w(\ell)\big)$, where $\delta(\ell)$ and $m(\ell)$ have the same meaning (and evolution) as previously defined,

$$\bar{\Delta}(\boldsymbol{y}) = \frac{1}{\mathsf{n}}\sum_{\ell=0}^{\mathsf{m}} y_\ell \left\{ \frac{y_\ell+1}{2} + \sum_{i=0}^{\ell-1} y_i (1-p)^{\ell-i} + \overbrace{\frac{p(1-u)}{u}\sum_{i=0}^{\ell-1}(1-p)^i \sum_{j=0}^{\ell-i-1}\left[1-(1-u)^{y_j}\right]\prod_{w=j+1}^{\ell-i-1}(1-u)^{y_w}}^{(a)} \right\} \tag{7}$$

and $w(\ell)$ represents the value that AoI will reset to, if a successful update is performed. This last component of the state is also subject to further noise, not only due to packet losses, but also in view of the exogenous generation process.

The evolution of $x(\ell)$ can be tracked as follows. First, $w(\ell+1)$ can take value

- 0 if new data is generated, which happens with probability u. Indeed, with fresh data, a successful transmission would reset the AoI to its minimum value, as was the case in the generate-at-will setting;
- $w(\ell)+1$ otherwise, i.e., with probability $1-u$.

Then, $\delta(\ell+1)$ is set to

- $w(\ell+1)$ if an update is attempted in slot ℓ, thus requiring $c(\ell) = 1$, and such update is successful, which happens with probability p
- or $\delta(\ell)+1$ if no update is delivered over slot ℓ, i.e, if no transmission is attempted or a transmission is attempted but it is not successful.

Finally, $m(\ell+1)$ is set to $[m(\ell) - c(\ell)]^+$, with $[x]^+$ denoting $\max(x, 0)$. That is, the number of transmission opportunities left is decreased by 1 with respect to $m(\ell)$ if the control action $c(\ell) = 1$, which in turn requires $m(\ell) > 0$, i.e., some transmission opportunities are still available. Otherwise, if $m(\ell) = 0$, then $c(\ell)$ must be 0 and the value of $m(\ell+1)$ also stays zero.

Similarly to the generate-at-will case, also within this framework the optimal control policy $\mu_\ell\big(x(\ell), p, u\big)$ can be found by dynamic programming, i.e., obtained through backward induction from the last stage. It is also immediate to prove further properties such as threshold behaviors for $\delta(\ell)$ and $w(\ell)$, along the lines of [45], as we clarify in the following.

Proposition 4. *In the exogenous case ($u < 1$), when feedback is available, it holds:*

(a) *The optimal control satisfies a lower-threshold condition on $\delta(\ell)$, i.e., if $\mu_\ell\big(\delta(\ell), m(\ell), w(\ell)\big) = 1$, then $\mu_\ell\big(x, m(\ell), w(\ell)\big) = 1$ for every $x > \delta(\ell)$.*

(b) *The optimal control satisfies an upper-threshold condition on $w(\ell)$, i.e., if $\mu_\ell\big(\delta(\ell), m(\ell), w(\ell)\big) = 1$, then $\mu_\ell\big(\delta(\ell), m(\ell), x\big) = 1$ for every x s.t. $0 \le x < w(\ell)$.*

(c) *The optimal control policy $\mu_\ell\big(x(\ell), p, u\big)$ has a further dependence on u and is computed as*

$$\mu_\ell\big(x(\ell), p, u\big) = \mathbb{1}\big[\mathcal{Y}_{\ell+1}\big(\delta(\ell), m(\ell), w(\ell)\big) > p\mathcal{X}_{\ell+1}\big(m(\ell)-1, w(\ell)\big) + (1-p)\mathcal{Y}_{\ell+1}\big(\delta(\ell), m(\ell)-1, w(\ell)\big)\big] \tag{8}$$

where:

$$\mathcal{X}_\ell\big(m, w\big) = uR_\ell\big(0, m, 0\big) + (1-u)R_\ell\big(w+1, m, w+1\big)$$
$$\mathcal{Y}_\ell\big(\delta, m, w\big) = uR_\ell\big(\delta+1, m, 0\big) + (1-u)R_\ell\big(\delta+1, m, w+1\big)$$

and $R_\ell\big(x(\ell)\big)$ is defined as in (6).

Proof: Statements (a) and (b) are immediate consequences of Bellman condition, following from monotonic properties of $\delta(\ell)$. For (c), the optimality of the policy in (8) can be proven analogously to Prop. 2. ■

We remark that the optimal policy found in Prop. 4, similar to the one found in Prop. 2, is *nonstationary*, which implies that approaches of long-term average AoI minimization with steady-state conditions, neglecting slot index ℓ, would necessarily be suboptimal.

For what concerns computational complexity, the same considerations of Sec. IV-B apply, in that the optimal policy is computed once and just works as a look-up during run-time. This time, since the state has an extra element w, whose size is $\mathcal{O}(\mathsf{n})$, the complexity is $\mathcal{O}(\mathsf{mn}^3)$, but in reality neither $\delta(\ell)$ nor $w(\ell)$ reach value n if not rarely.

C. Online scheduling, zero feedback (ZF)

Finally, consider the case of exogenous generation without feedback. Here, the transmitter is unaware of the transmission outcomes, but still knows whether new data have been generated or not. This means being fully aware of the value of $w(\ell)$ but not of $\delta(\ell)$. However, since past transmission instants are known, the expected value of $\delta(\ell)$ can be computed through the success probability by considering all possible cases of past successes/failures [31].

Thus, the optimal control action $\mu_\ell\big(x(\ell), p, u\big)$, as derived through (8), would still be valid, but the uncertainty over the instantaneous AoI $\delta(\ell)$ requires it to be replaced with its estimate $A(\ell) = \mathbb{E}[\delta(\ell)]$, which is the expected AoI in slot ℓ.

One can compute $A(\ell)$ through the following inductive procedure. Recalling that the transmission attempt times are denoted as $\tau_1, \tau_2, \ldots, \tau_{\mathsf{m}}$, if $\ell < \tau_1$, then $A(\ell) = \delta(\ell) = \ell$. Moreover, it holds:

$$A(\tau_j) = p \cdot w(\tau_j) + (1-p) \cdot \big(A(\tau_j-1)+1\big) \tag{9}$$
$$A(\ell) = A(\tau_j) + (\ell - \tau_j) \quad \text{for } \tau_j < \ell < \tau_{j+1} \text{ or } \ell = \mathsf{n}$$

In other words, $A(\ell)$ grows linearly in between transmission attempts because of the linearity of the expectation. In each transmission attempt, the options bifurcate between a successful transmission with probability p, which would reset AoI to $w(\ell)$, and a packet loss with probability $1-p$, which causes AoI to continue growing like before.

All the values of $w(\ell)$, τ_j, and p are known to the transmitter even in the ZF case, making the use of $A(\ell)$ feasible. However, since $A(\ell)$ is in general fractional, we take the optimal control action as dictated by $\big[A(\ell)\big]$, where $\big[\,\cdot\,\big]$ denotes rounding to the closest integer. This only introduces a quantization error, that is actually negligible if updates are very sporadic. Instead, replacing $\delta(\ell)$ with $A(\ell)$ is still optimal for the case where $\delta(\ell)$ is not available but $A(\ell)$ is, as proven below.

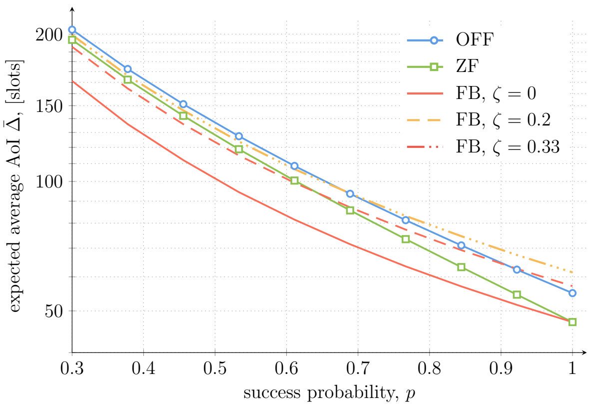


Fig. 8. Expected average AoI vs success probability under exogenous traffic, with $u = 0.1$. Circle markers denote the offline policy, square markers the online, zero-feedback solution, whereas the lines without markers report the trends for the online solution relying on feedback, with different cost factors. In all cases, $\mathsf{n} = 1600$, $\eta = 0.01$.

Proposition 5. *Control $\mu_\ell\big(x(\ell), p, u\big)$, as specified in (8), is optimal up to the quantization error in the absence of feedback, i.e., if $x(\ell) = \big([A(\ell)], m(\ell), w(\ell)\big)$ is used.*

Proof: The proof immediately follows from applying Bellman's condition (8) to choose the optimal action between the two available ones (transmit or not) as the one with lower expectation of the cumulated penalty $g\big(x(\ell), c(\ell), p\big)$, which, as per (6), is the AoI. Thus, the comparison boils down to evaluating whether the cumulated sum of $\mathbb{E}[A(t)] = A(t) = \mathbb{E}[\delta(t)]$ is lower when transmitting or not. As a side note, using $[A(\ell)]$ instead of $A(\ell)$ (as we are forced to do, since $\mu_\ell\big(x(\ell), p, u\big)$ only takes discrete values for $x(\ell)$) implies a quantization error, which is however kept limited by the threshold nature of the optimal policy, as argued in Prop. 4. ■

Compared to the case with feedback (FB), where $\delta(\ell)$ is precisely known as time unfolds, not only in expectation, the scheduling of this resulting ZF approach may be different, depending on the value of p. This uncertainty over past transmissions is minimized when $p \to 1$ (but also when it approaches 0). Overall, knowing only $w(\ell)$ and $A(\ell) = \mathbb{E}[\delta(\ell)]$, we obtain an intermediate AoI between an agnostic scheduling that knows nothing, and the FB case where also $\delta(\ell)$ is known.

D. Performance Evaluation

We focus again on the reference setting with $\mathsf{n} = 1600$ slots, and with a duty cycle $\eta = 0.01$. Initial insights are offered in Fig. 8, which reports the expected average AoI against the success probability for $u = 0.1$. In the plot, the performance of the optimized OFF policy is shown with circle markers, whereas square markers denote the results of the ZF solution. Lines without markers refer to the FB policy, considering both the ideal case $\zeta = 0$ (solid line) and settings in which feedback entails a cost (dashed, dash-dotted lines).

Let us focus first on the OFF and zero-cost FB ($\zeta = 0$) strategies. Under exogenous traffic, both solutions suffer a performance degradation compared to the generate-at-will case. For instance, for a success probability $p = 0.7$, the expected average AoI of the OFF scheme increases by 8% compared to what shown in Fig. 4, whereas the loss is contained to just 1.4% with ideal feedback. Moreover, the two schemes no longer offer the same performance under perfect channel conditions $p = 1$ as discussed for $u = 1$, and the optimal offline schedule leads to a value of $\bar{\Delta}$ almost 20% worse than what achieved by the FB solution. Both remarks stem from the intrinsic advantage of online approaches to adapt to the traffic generation pattern and to the benefit that sending a packet might have on the current level of AoI at the receiver. In this perspective, it is also interesting to notice how the region of p values where the use of feedback is convenient even when a cost has to be undergone is increased. For instance, for $\zeta = 0.2$, the expected average AoI of the FB policy improves over the optimal offline schedule as soon as the packet loss rate is higher than 7% ($p = 0.93$), whereas in such conditions a performance loss of more than 10% was undergone with feedback in the generate at will case. Similarly, it is now possible to find p values for which FB outperforms OFF even for $\zeta = 0.33$, in contrast to the $u = 1$ case (see Fig. 6).

To better isolate the role played by the knowledge of the current AoI at the receiver and the knowledge of the update generation times, consider now the simpler ZF solution. The scheme requires no feedback (i.e., it operates agnostically with respect to $\delta(t)$), and only adapts the schedule dynamically based on the time-stamp of the currently available reading at the transmitter. As expected, the scheme always outperforms an offline approach. Notably, the gain grows with p, and thus becomes relevant especially for operating conditions of practical relevance (e.g., $p > 0.8$). Note in fact that, when there is little uncertainty on the successful delivery of an update, the decisions made by the ZF approach are almost optimal, and the scheme actually converges to the full-blown ideal FB case for $p = 1$. From this standpoint, the use of a simple online scheme emerges as a good solution for reasonably reliable communication links, as it suffices to attain remarkable and close-to-optimal improvements over an offline solution without the need to implement a return channel from receiver to IoT device. In fact, ZF outperforms FB when a feedback cost has to be undergone. For instance, in the setting under study, an AoI reduction of ~15% is attained with respect to FB with a limited cost of $\zeta = 0.2$ for $p = 0.9$.

From the above discussion, two aspects can be stressed: the importance of relying on a dynamic schedule, and the fact that the trade-off between AoI reduction and feedback cost changes significantly in the case of exogenous traffic when compared to a generate-at-will setting. In turn, both aspects crucially depend on how frequently new readings are made available for transmissions. To further explore this aspect, we consider the impact of the generation rate u in Fig. 9. The plot is akin to Fig. 5, reporting the ratio of the expected average AoI attained with ZF or FB to that of the optimal offline schedule derived via (7). Dashed and dotted lines indicate the performance with a costly feedback for different values of ζ, whereas the solid, circle marked line that of a cost-free feedback. Finally, the solid, square-marked curve denotes the behavior of the ZF solution. The results were obtained considering $p = 0.7$, and

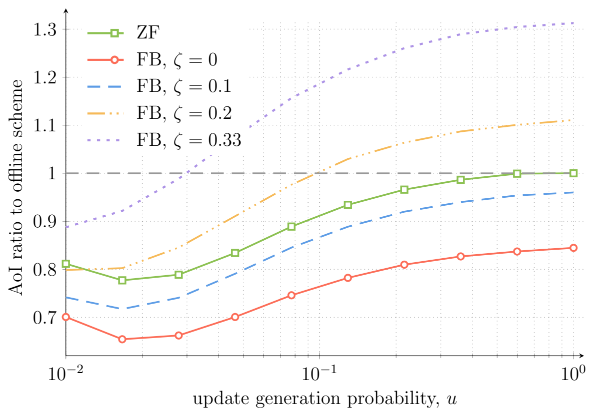


Fig. 9. Ratio of the expected average AoI attained with different schemes to the expected average AoI of the optimized offline approach, reported against the probability of generating a new updated at each slot, u. In all cases, $\mathsf{n} = 1600$, $\eta = 0.01$, $p = 0.7$.

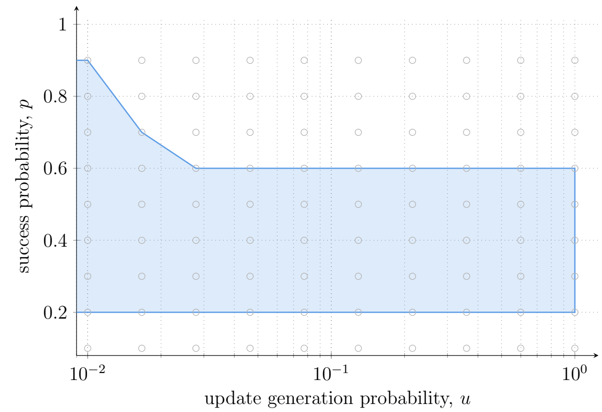


Fig. 10. The blue-shaded region reports the (u, p) pairs for which the use of a costly feedback is convenient over the simpler online, zero-feedback approach. Conversely, the white region indicates where the zero-feedback approach offers lower expected average AoI. The circles report all the (u, p) pairs that were considered. Results were obtained for $\mathsf{n} = 1600$, $\eta = 0.01$ and $\zeta = 0.2$.

we recall that values below 1 pinpoint conditions where the OFF strategy is outperformed.

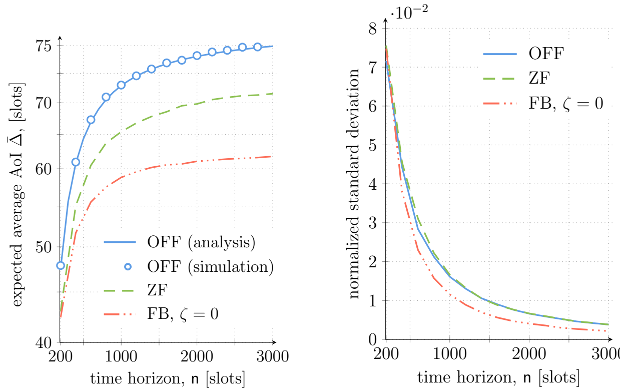
An interesting trend emerges, as the improvement offered by all the online approaches peaks for a certain value of u. Notably, such values are comparable to the duty cycle ($\eta = 0.01$), so that the maximum benefit is attained when the average number of updates produced over the time horizon is similar to the number of available transmission opportunities. In such conditions, the expected average AoI is reduced by more than 35% with a cost-free feedback, or by roughly 22% with the simpler ZF scheme. This behavior can be explained observing how, when readings are seldom produced (i.e., very low values of u), all online schemes tend to converge towards transmitting as soon as a new update is available and have thus little room for adaptation. Eventually, for $u \to 0$, the difference with the offline strategy vanishes, as AoI can never be reset. Conversely, as u increases (rightmost part of the plot), fresh messages are available more often - and, on average, more frequently compared to the duty cycle - so that a dynamic adaptation to the traffic generation loses leverage compared to the OFF solution.

Fig. 9 also highlights that, for a relatively low feedback cost ($\zeta = 0.1$), the FB scheme outperforms the offline schedule regardless of the generation rate. Moreover, even for larger values of ζ the availability of feedback can be convenient in terms of AoI reduction for sufficiently low u. In this perspective, the cost undergone for the implementation of a return channel may be valuable in applications that can only produce readings sporadically, e.g., due to sensor limitations, processing times, or energy expenditure. In the reported setting, for instance, the possibility to adapt transmissions to the time-stamp of the available reading as well as to the current level of AoI at the receiver is convenient also for $\zeta = 0.3$ as long as $u < 0.03$. On the other hand, severe performance degradation can be incurred with a FB strategy under certain conditions. Considering again the not too high cost $\zeta = 0.3$, $\bar{\Delta}$ can be up to 30% worse than the basic OFF case when updates are generated frequently enough. In addition, we observe that the simple ZF solution can outperform the full-blown FB approach when the presence of feedback decreases the number of transmission opportunities. In the setting of Fig. 9, this is the case for $\zeta > 0.2$.

The final remarks clearly point out the importance of understanding which strategy shall be preferred under which conditions. From this standpoint, an ideal (and impractical) cost-free feedback always performs best, and an offline schedule is always outperformed by the ZF scheme. The more interesting question of whether a costly feedback shall be considered in place of the ZF approach is instead tackled in Fig. 10. In the plot, a cost $\zeta = 0.2$ was considered, and the shaded area denotes the (p, u) pairs in which the FB scheme provides a lower expected average AoI compared to ZF. A total of 90 different configurations were tested, with $p \in [0.1, 0.9]$ and $u \in [0.01, 1]$, and are shown by circle markers.

In very harsh channel conditions, e.g., values of $p < 0.2$ for the setting under study, the ZF scheme offers lower AoI. In this case, having more transmission attempts is preferable in terms of dynamic schedule adaptation compared to knowing their (already likely failed) outcome. Conversely, the FB approach becomes dominant for intermediate values of p. Here, the ability to adapt and react to potential packet losses becomes paramount, and outweighs the availability of fewer transmissions. This is especially true for lower values of u, where feedback remains convenient for the tested values of p up to 0.9. In the generate-at-will case ($u = 1$), as already discussed, the ZF solution coincides with the basic offline approach, and may beneficial compared to the use of feedback with a reduced number of transmission opportunities depending on p (see Fig. 6). In general, the framework we presented provides a useful tool, as diagrams like the one of Fig. 10 can easily be generated for any value of ζ, determining the non-trivial conditions in terms of of both application specific generation rates and channel reliability of the channel that shall drive the scheduling approach of choice.

Finally, we study the impact of the time horizon duration on the considered scheduling policies also under exogenous traffic



(a) (b)

Fig. 11. (a): expected average AoI vs horizon length. For the offline case, markers report the outcome of simulation results, verifying the analysis in (7) (solid line). (b): standard deviation of the average AoI, normalized to the duration of the time horizon. All results for $p = 0.8$, $\eta = 0.01$, $u = 0.01$.

conditions. Fig. 11a reports the expected average AoI against n for a fixed duty cycle $\eta = 0.01$, $p = 0.8$ and $u = 0.01$, for the OFF (solid blue line), ZF (dashed green line) and zero-cost feedback (dash-dotted, red line). For all the policies, Fig. 11b shows the standard deviation of the average AoI obtained over different realizations of the channel conditions, normalized to the duration of the time horizon. The fundamental remarks highlighted in the generate-at-will case (Fig. 7) still hold, supported by same reasoning: as n grows, $\bar{\Delta}$ tends to stabilize, and the standard deviation of the average AoI value becomes proportionally smaller. Interestingly, Fig. 11b also pinpoints a consistently smaller normalized standard deviation for the FB policy with respect to its competitors, hinting at another potential benefit of the fully-aware scheduling enabled in the presence of feedback.

VI. Conclusions and Future Work

We investigated the role of feedback in AoI-optimal finite-horizon scheduling for IoT devices with limited transmission opportunities over an erasure channel. We compared an agnostic scheduler, where the status update reporting times over the finite horizon are predefined in advance, and an online scheduling, which instead allows for adjusting the transmission pattern at run time, depending on the outcome of channel transmissions. For both, we developed different optimization frameworks, and we further extended the analysis to different conditions concerning the frequency of update generation. In particular, we considered the most common scenario where updates are generated at will [10], [14], but also the case for exogenous non-persistent generation, where AoI is reset to an intermediate value. The latter setting actually results in three different schedulers, namely, one that is totally agnostic and planned in advance; one that is fully aware of the current AoI value, as well as the value of the freshest information available at the transmitter's side; and finally an intermediate zero-feedback case, where the scheduling can still be adjusted at runtime, but the current AoI is only known in expectation.

In all these situations, we performed comparative evaluations towards the evaluation of whether a feedback from the monitoring node is required if it comes at a cost, which is

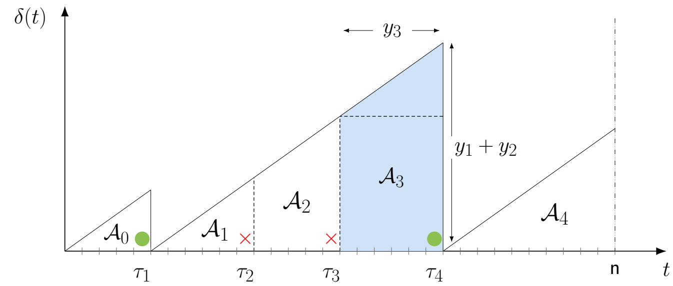


Fig. 12. Example of time evolution for $\delta(t)$ for the derivation of Prop. 1, generate-at-will case. $\mathsf{m} = 4$ transmissions are performed, out of which the central two are not successful ($\times$). The shaded area $\mathcal{A}_3$ is the area contribution of the transmission performed at time τ_3, corresponding to the area of an isosceles triangle of side y_3 and of a rectangle of sides y_3 and $y_1 + y_2$.

captured by a reduction in the number of available sensor transmissions. The takeaway message is that in many cases, especially if the feedback cost is heavy, a predefined agnostic scheduling may be sufficient to obtain good performance, and even better than waiting for an expensive feedback.

A possible direction for future work would be to expand the time horizon, keeping it finite while allowing for concatenation of different scheduling cycles, which can be a realistic way to represent operation of IoT nodes. The comparison of different techniques, and its implications on the utility of exploiting a feedback at the transmitter's side can be extended to more general scenarios, including multiple nodes [20], access protocol aspects [22], or energy harvesting [30]. In particular, the study of schemes that foresee the possibility to send aggregate feedback acknowledging multiple packets to save energy, or of solutions that jointly tackle minimization of AoI and performed transmissions may be of notable practical relevance.

Appendix A
Proof of Proposition 1

The proposition follows from geometric arguments. With reference to Fig. 12, the average AoI can be expressed as the sum of multiple components, so that

$$\bar{\Delta}(\boldsymbol{y}) = \mathbb{E}\left[\frac{1}{\mathsf{n}}\sum_{\ell=0}^{\mathsf{m}} \mathcal{A}_\ell\right]. \tag{10}$$

Here, $\mathcal{A}_\ell$ is the sum of the y_ℓ AoI values within the ℓ-th transmission interval, and the expectation is intended over the outcomes of all the transmissions performed within the time horizon. By the linearity of expectation, we tackle the generic term $\mathbb{E}[\mathcal{A}_\ell]$, and observe that $\mathcal{A}_\ell$ can be expressed as the sum of two components:

$$\mathcal{A}_\ell = \frac{y_\ell(y_\ell + 1)}{2} + y_\ell Z_\ell. \tag{11}$$

The first addend corresponds to the sum of AoI values within the isosceles triangle of side y_ℓ, capturing the unavoidable linear growth of AoI during the interval, whereas the second accounts for a rectangle of sides y_ℓ and Z_ℓ. In turn, the r.v. Z_ℓ represents the value of the current AoI at the start of the ℓ-th transmission period, i.e., $\delta(\tau_\ell)$. As exemplified in Fig. 12, Z_ℓ is zero if the ℓ-th transmission succeeds, or equals $Z_{\ell-1} + y_{\ell-1}$

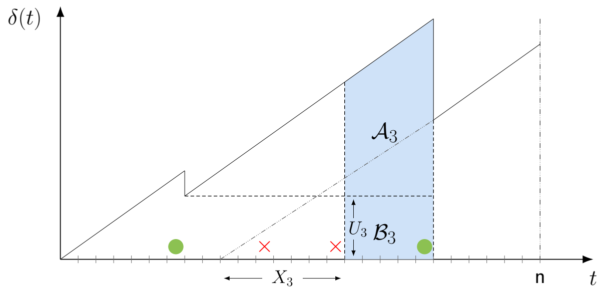


Fig. 13. Example of time evolution for $\delta(t)$ for the derivation of Prop. 3, exogenous case. $\mathsf{m}=4$ transmissions are performed, out of which the central two are not successful (×). The shaded area is the contribution of the transmission performed at time τ_3, expressed as the sum of $\mathcal{A}_3$ and $\mathcal{B}_3$.

otherwise, as the current AoI grows linearly over the previous interval. Leaning on this, and recalling that the outcome of successive attempts is i.i.d., we express the expectation of Z_ℓ in recursive form as

$$\mathbb{E}[Z_\ell] = (1-p)y_{\ell-1} + (1-p)\mathbb{E}[Z_{\ell-1}].$$

Moreover, we have $\mathbb{E}[Z_0]=0$, given the assumption $\delta(0)=0$. Simple manipulations lead to the compact expression

$$\mathbb{E}[Z_\ell] = \sum_{i=0}^{\ell-1} y_i (1-p)^{\ell-i} \tag{12}$$

for $\ell \geq 1$. Plugging (12) into (11) and taking the summation in (10) leads to the result.

Appendix B
Proof of Proposition 3

Following once more geometric arguments, the area below the AoI curve can be computed as

$$\bar{\Delta}(\boldsymbol{y}) = \mathbb{E}\left[\frac{1}{\mathsf{n}}\sum_{\ell=0}^{\mathsf{m}}(\mathcal{A}_\ell + \mathcal{B}_\ell)\right].$$

In this case, the contribution of the ℓ-th transmission interval is split in two terms. The former, $\mathcal{A}_\ell$, is the one already defined in (11). On the other hand, $\mathcal{B}_\ell$ corresponds to the sum of the AoI values within a rectangle of side y_ℓ and height equal to the last value at which the AoI was successfully reset. This component accounts for the propagating effect of delivering a non-fresh update, and is intrinsic to the exogenous traffic generation case. To characterize this aspect, let us introduce the r.v.s U_ℓ, $\ell = 0, \dots, \mathsf{m}$, denoting exactly the value of $\delta(t)$ after the last update delivery as of time $\tau_{\ell-1}$. By definition, $U_0 = 0$. Let us furthermore consider the r.v.s X_ℓ, $\ell = 0, \dots, \mathsf{m}$, indicating the value at which the AoI would be reset by the ℓ-th transmission if this was successful. In other words, the variable captures the number of slots elapsed since tha last update generation as of time τ_ℓ. Examples of both quantities are reported in Fig. 13.

Following these definitions, for any $\ell \in \{1, \dots, \mathsf{m}\}$:

$$U_\ell = \begin{cases} U_{\ell-1} & \text{w/ prob. } (1-p) \\ X_\ell & \text{w/ prob. } p. \end{cases}$$

Accordingly, the expectation of the r.v. can be conveniently expressed in recursive form as

$$\mathbb{E}[U_\ell] = (1-p)\mathbb{E}[U_{\ell-1}] + p\,\mathbb{E}[X_\ell].$$

Recalling that $\mathcal{B}_\ell = y_\ell\, U_\ell$, we also have after simple manipulations

$$\mathbb{E}[\mathcal{B}_\ell] = y_\ell\, p \sum_{i=0}^{\ell-1} (1-p)^i\, \mathbb{E}[X_{\ell-i}]. \tag{13}$$

The expression in (13) clarifies how a computation of the first order moment of the r.v.s X_ℓ suffices to obtain the sought expected average AoI. To this aim, we observe that X_ℓ can be written as

$$X_\ell = \begin{cases} x & \text{w/ prob. } u(1-u)^x \\ X_{\ell-1} + y_{\ell-1} & \text{w/ prob. } (1-u)^{y_{\ell-1}} \end{cases}$$

where $x \in 0, \dots, y_{\ell-1}-1$. Here, the first case corresponds to having a new reading generated by the sensor within the ℓ-th interval, specifically at the $(y_\ell - x)$-th slot of the period. Conversely, if no update is produced within the interval (probability $(1-u)^{y_\ell}$), the whole y_ℓ-slot duration is added to the time elapsed since the last generation. The expected value of the r.v. takes thus the form

$$\mathbb{E}[X_\ell] = (1-u)^{y_{\ell-1}}\left(y_{\ell-1} + \mathbb{E}[X_{\ell-1}]\right) + \sum_{x=0}^{y_{\ell-1}-1} x\, u (1-u)^x.$$

with $\mathbb{E}[X_0] = 0$. Simple algebraic steps applied to this recursion lead to the compact expression

$$\mathbb{E}[X_\ell] = \sum_{i=0}^{\ell-1} \frac{(1-u)[1-(1-u)^{y_i}]}{u} \prod_{j=i+1}^{\ell-1} (1-u)^{y_j}. \tag{14}$$

Plugging (14) into (13), and combining the result with the derivation of $\mathbb{E}[\mathcal{A}_\ell]$ presented in (11)-(12) finally leads to the complete formulation of $\bar{\Delta}$ in (7).

References

[1] A. Munari and L. Badia, "The role of feedback in AoI optimization under limited transmission opportunities," in *Proc. IEEE Global Commun. Conf.*, 2022, pp. 1972–1977.

[2] R. D. Yates, Y. Sun, D. R. Brown, S. K. Kaul, E. Modiano, and S. Ulukus, "Age of information: An introduction and survey," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 5, pp. 1183–1210, May 2021.

[3] A. Kosta, N. Pappas, V. Angelakis *et al.*, "Age of information: A new concept, metric, and tool," *Found. Trends Netw.*, vol. 12, no. 3, pp. 162–259, 2017.

[4] C. Kam, S. Kompella, G. D. Nguyen, J. E. Wieselthier, and A. Ephremides, "On the age of information with packet deadlines," *IEEE Trans. Inf. Theory*, vol. 64, no. 9, pp. 6419–6428, Sep. 2018.

[5] S. S. Vilni, A. Zakeri, M. Moltafet, and M. Codreanu, "Goal-oriented real-time tracking in a status update system under imperfect feedback," in *Proc. IEEE Asilomar Conf. Signals Syst. Comp.*, 2024, pp. 391–395.

[6] Y. Sun, Y. Polyanskiy, and E. Uysal, "Sampling of the Wiener process for remote estimation over a channel with random delay," *IEEE Trans. Inf. Theory*, vol. 66, no. 2, pp. 1118–1135, Feb. 2020.

[7] Y.-P. Hsu, E. Modiano, and L. Duan, "Scheduling algorithms for minimizing age of information in wireless broadcast networks with random arrivals," *IEEE Trans. Mobile Comput.*, vol. 19, no. 12, pp. 2903–2915, 2020.

[8] Q. Liu, C. Li, Y. T. Hou, W. Lou, and S. Kompella, "AION: A bandwidth optimized scheduler with AoI guarantee," in *Proc. IEEE Infocom*, 2021.

[9] E. T. Ceran, D. Gündüz, and A. György, "Average age of information with hybrid ARQ under a resource constraint," *IEEE Trans. Wireless Commun.*, vol. 18, no. 3, pp. 1900–1913, Mar. 2019.

[10] S. Sun, W. Wu, C. Fu, X. Qiu, J. Luo, and J. Wang, “AoI optimization in multi-source update network systems under stochastic energy harvesting model,” *IEEE J. Sel. Areas Commun.*, vol. 42, no. 11, pp. 3172–3187, 2024.

[11] A. Munari, “Modern random access: An age of information perspective on irregular repetition slotted ALOHA,” *IEEE Trans. Commun.*, vol. 69, no. 6, pp. 3572–3585, 2021.

[12] Q. Wang, M. Hempstead, and W. Yang, “A realistic power consumption model for wireless sensor network devices,” in *Proc. IEEE SECON*, vol. 1, 2006, pp. 286–295.

[13] Semtech, “SX1276 Spreadsheet,” www.semtech.com/products/wireless-rf/lora-connect/sx1276.

[14] A. Javani, M. Zorgui, and Z. Wang, “On the age of information in erasure channels with feedback,” in *Proc. IEEE Int. Conf. Commun. (ICC)*. IEEE, 2020.

[15] S. Farazi, A. G. Klein, and D. R. Brown, “Average age of information in update systems with active sources and packet delivery errors,” *IEEE Wireless Commun. Lett.*, vol. 9, no. 8, pp. 1164–1168, 2020.

[16] C. Li, Q. Liu, S. Li, Y. Chen, Y. T. Hou, W. Lou, and S. Kompella, “Scheduling with age of information guarantee,” *IEEE/ACM Trans. Netw.*, vol. 30, no. 5, pp. 2046–2059, 2022.

[17] I. Kadota, A. Sinha, E. Uysal-Biyikoglu, R. Singh, and E. Modiano, “Scheduling policies for minimizing age of information in broadcast wireless networks,” *IEEE/ACM Trans. Netw.*, vol. 26, no. 6, pp. 2637–2650, 2018.

[18] A. M. Bedewy, Y. Sun, S. Kompella, and N. B. Shroff, “Optimal sampling and scheduling for timely status updates in multi-source networks,” *IEEE Trans. Inf. Theory*, vol. 67, no. 6, pp. 4019–4034, 2021.

[19] S. Kaul, R. Yates, and M. Gruteser, “Real-time status: How often should one update?” in *Proc. IEEE Infocom*, 2012, pp. 2731–2735.

[20] B. Zhou and W. Saad, “Minimum age of information in the Internet of things with non-uniform status packet sizes,” *IEEE Trans. Wireless Commun.*, vol. 19, no. 3, pp. 1933–1947, Mar. 2019.

[21] ——, “Performance analysis of age of information in ultra-dense Internet of things (IoT) systems with noisy channels,” *IEEE Trans. Wireless Commun.*, vol. 21, no. 5, pp. 3493–3507, May 2022.

[22] J. Sun, Z. Jiang, B. Krishnamachari, S. Zhou, and Z. Niu, “Closed-form Whittle’s index-enabled random access for timely status update,” *IEEE Trans. Commun.*, vol. 68, no. 3, pp. 1538–1551, Mar. 2019.

[23] P. D. Mankar, M. A. Abd-Elmagid, and H. S. Dhillon, “Spatial distribution of the mean peak age of information in wireless networks,” *IEEE Trans. Wireless Commun.*, vol. 20, no. 7, pp. 4465–4479, Jul. 2021.

[24] J. P. Champati, H. Al-Zubaidy, and J. Gross, “Statistical guarantee optimization for AoI in single-hop and two-hop FCFS systems with periodic arrivals,” *IEEE Trans. Commun.*, vol. 69, no. 1, pp. 365–381, 2020.

[25] O. T. Yavascan and E. Uysal, “Analysis of slotted ALOHA with an age threshold,” *IEEE J. Sel. Areas Commun.*, vol. 39, no. 5, pp. 1456–1470, May 2021.

[26] E. Fountoulakis, N. Pappas, M. Codreanu, and A. Ephremides, “Optimal sampling cost in wireless networks with age of information constraints,” in *Proc. IEEE Infocom Wkshps*, 2020, pp. 918–923.

[27] H. Pan, J. Liang, S. C. Liew, V. C. Leung, and J. Li, “Timely information update with nonorthogonal multiple access,” *IEEE Trans. Ind. Informat.*, vol. 17, no. 6, pp. 4096–4106, Jun. 2020.

[28] H. Tang, J. Wang, L. Song, and J. Song, “Minimizing age of information with power constraints: Multi-user opportunistic scheduling in multi-state time-varying channels,” *IEEE J. Sel. Areas Commun.*, vol. 38, no. 5, pp. 854–868, May 2020.

[29] M. Xie, Q. Wang, J. Gong, and X. Ma, “Age and energy analysis for LDPC coded status update with and without ARQ,” *IEEE Internet Things J.*, vol. 7, no. 10, pp. 10 388–10 400, Oct. 2020.

[30] M. Hatami, M. Leinonen, and M. Codreanu, “AoI minimization in status update control with energy harvesting sensors,” *IEEE Trans. Commun.*, vol. 69, no. 12, pp. 8335–8351, Dec. 2021.

[31] Y. Ji, Y. Lu, X. Xu, and X. Huang, “Age-optimal packet scheduling with resource constraint and feedback delay,” *IEEE Trans. Commun.*, vol. 72, no. 7, pp. 4041–4054, 2024.

[32] G. Peskir and A. Shiryaev, *Optimal stopping and free-boundary problems*. Springer, 2006.

[33] O. Ayan, M. Vilgelm, M. Klügel, S. Hirche, and W. Kellerer, “Age-of-information vs. value-of-information scheduling for cellular networked control systems,” in *Proc. ACM/IEEE ICCPS*, 2019, pp. 109–117.

[34] S. Feng and J. Yang, “Age of information minimization for an energy harvesting source with updating erasures: Without and with feedback,” *IEEE Trans. Commun.*, vol. 69, no. 8, pp. 5091–5105, Aug. 2021.

[35] F. Pyttel, W. de Sombre, A. Ortiz, and A. Klein, “Age of information minimization in status update systems with imperfect feedback channel,” in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2024, pp. 201–206.

[36] LoRa Alliance Tech. Committee, “LoRaWAN 1.1 specification,” Oct. 2017.

[37] K.-H. Ngo, G. Durisi, A. Munari, F. Lázaro, and A. Graell i Amat, “Timely status updates in slotted ALOHA networks with energy harvesting,” *IEEE Trans. Commun.*, 2025, early access, doi: 10.1109/TCOMM.2025.3548035.

[38] R. Yates, “Lazy is timely: Status updates by an energy harvesting source,” in *Proc. IEEE ISIT*, 2015.

[39] L. Badia, “On the impact of correlated arrivals and errors on ARQ delay terms,” *IEEE Trans. Commun.*, vol. 57, no. 2, pp. 334–338, Feb. 2009.

[40] ——, “On the effect of feedback errors in Markov models for SR ARQ packet delays,” in *Proc. IEEE Globecom*, 2009.

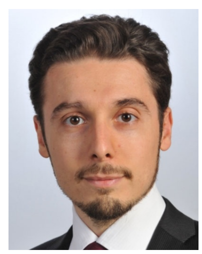
[41] F. Chiariotti, A. Munari, L. Badia, and P. Popovski, “Peak age of incorrect information of reactive ALOHA reporting under imperfect feedback,” in *Proc. IEEE ICC*, 2025.

[42] H. Jayakumar, K. Lee, W. S. Lee, A. Raha, Y. Kim, and V. Raghunathan, “Powering the Internet of things,” in *Proc. ACM Int. Symp. Low Power Elec. Design (ISLPED)*, 2014, pp. 375–380.

[43] R. D. Yates and S. K. Kaul, “Status updates over unreliable multiaccess channels,” in *Proc. IEEE ISIT*, 2017, pp. 331–335.

[44] D. Bertsekas, *Dynamic programming and optimal control: Volume I*. Athena scientific, 2012.

[45] M. A. Abd-Elmagid, H. S. Dhillon, and N. Pappas, “A reinforcement learning framework for optimizing age of information in RF-powered communication systems,” *IEEE Trans. Commun.*, vol. 68, no. 8, pp. 4747–4760, 2020.



Andrea Munari (Senior Member, IEEE) received the M.Sc. and Ph.D. degrees in telecommunications engineering from the University of Padova, Padua, Italy, in 2006 and 2010, respectively. From 2007 to 2010, he was with IBM Research in Zurich, Switzerland. In 2011, he joined the Corporation Research and Development Division, Qualcomm Inc., San Diego, CA, US. He is currently with the Institute of Communications and Navigation, German Aerospace Center (DLR), Wessling, Germany. From 2014 to 2018, he held a Senior Researcher and a Lecturer position at the RWTH Aachen University, Germany. His main research interests are in the area of wireless communications, with special focus on Internet of things and satellite communications. Dr. Munari received the 2018 ACM MobiCom Workshop on Millimeter Wave Networks and Sensing Systems Best Paper Award, and the IEEE GLOBECOM 2020 Communications Theory Symposium Best Paper Award. He serves as Associate Editor of IEEE Communications Letters.

Leonardo Badia (Senior Member, IEEE) received the Laurea degree (Hons.) in electrical engineering and the Ph.D. degree in information engineering from the University of Ferrara, Italy, in 2000 and 2004, respectively. From 2002 to 2003, he was visiting scholar at the Radio System Technology Labs (currently, Wireless@KTH), Royal Institute of Technology, Stockholm, Sweden. After having been with the Engineering Department, University of Ferrara, he joined in 2006 the IMT Institute for Advanced Studies, Lucca, Italy. In 2011, he moved to the University of Padua, Italy, where he is currently Associate Professor. Prof. Badia published more than 250 research papers. His scientific interests include protocol design for multihop networks, cross-layer optimization of wireless communication, transmission protocol modeling, and applications of game theory to radio resource management. He is an active referee of research articles for many scientific periodicals, as well as technical program committee chair for conferences in the broad areas of communications and networking.