Optimizing Peak Age under Intermittent Satellite Connectivity and Store-and-Forward

Çağrı Arı*, Özkan Tuğberk Kartal*, Andrea Munari†, Leonardo Badia‡, Elif Uysal*, Onur Kaya§
* Communication Networks Research Group, Dept. of EEE, METU, Ankara, Türkiye
† Institute of Communications and Navigation, German Aerospace Center (DLR), Weßling, Germany
‡ Dept. of Information Engineering, University of Padova, Padova, Italy
§ Electrical-Electronics Engineering, Işık University, Ankara, Türkiye

***Abstract*—We consider a real-time *task-oriented* application operating over an *intermittently available* satellite-based communication network, aiming to collect status updates generated by a remote sensing device. The system is modeled as a scheduling problem over a finite horizon, corresponding to the duration of the task, to minimize the *peak Age of Information* at the destination. The number of updates that can be transmitted is constrained by a transmission budget. Moreover, the status updates are subject to delays caused by the store-and-forward operation of the satellites, which may vastly vary depending on the network conditions. We investigate three levels of awareness regarding the connectivity conditions of the satellite network: (i) scheduling without any information about connectivity conditions, (ii) scheduling based solely on the current conditions, and (iii) scheduling based on full connectivity knowledge. The first case admits a relatively simple structure, for which a periodic transmission strategy is adopted. The latter two cases are formulated as semi-Markov decision processes and solved to obtain the optimal transmission scheduling policy. Simulation results demonstrate the impact of connectivity awareness on the application performance at the destination. Through a simple modeling approach, we provide first insights into the practically relevant setting of store-and-forward satellite architectures.**

***Index Terms*—Age of Information; Satellite Communications; Intermittent Availability; Store-and-Forward.**

I. Introduction

Future network scenarios, where intelligent terminals operate collaboratively in dynamic and uncertain environments, can envision a growth of task-oriented real-time applications, such as autonomous vehicle coordination, remote drone surveillance, and emergency robotics during natural disasters [1]. These applications require timely information to make decisions that impact physical actions in the correct way.

A promising solution to sustain task-oriented real-time applications effectively by providing wide-area, infrastructure-independent connectivity is represented by satellite communications, which are experiencing a revived interest in view of recent technological and standardization developments, with the introduction of non-terrestrial networks in the 3GPP ecosystem from Rel. 17. Satellite links can be crucial in remote, mobile, or disaster-affected environments where terrestrial networks are unavailable or unreliable. Low Earth orbit (LEO) constellations offer reduced latencies compared to more traditional geostationary satellites, enabling near real-time data exchange, which makes them suitable for the aforementioned applications [2]. However, data collection via LEO links poses peculiar challenges, as satellites remain within the visibility of a ground user for limited time windows and may not always have a direct connection to a ground station to enable delivery of information to the final destination. To tackle these issues, the concept of *store-and-forward* has recently attracted significant attention. In this approach, a satellite receiving data from a ground user can store them in its buffer and transmit them to the gateway once a direct link becomes available, or forward them to other nearby satellites in the constellation via inter-satellite links, which subsequently route them to the gateway. Store-and-forward represents a practical solution for satellite operators providing IoT services, and will be part of the NB-IoT standard from the upcoming 3GPP Rel. 19.

From a quantitative standpoint, one suitable performance indicator of information freshness for a task-oriented application is Peak Age of Information (PAoI) [3], i.e., the maximum age that information reaches over a monitoring window, which can be used as a strong assessment of the worst-case staleness used by the controller. However, investigations of scheduling policies that aim to reduce PAoI typically rely on consistent and predictable communication channels. This assumption breaks down in an environment with intermittent connectivity, e.g., when relying on LEO satellites. This can cause PAoI to spike, which complicates the task of determining when to report the status of the system. The scheduling must then be connectivity-aware, possibly adapting in real-time to channel state information [4].

In this paper, we consider a task-oriented real-time application where a sensing unit that reports information about the system under monitoring is connected through a satellite network, whose nodes use a store-and-forward policy. The monitoring task has a predefined duration, and the sensing unit is constrained to report a limited number of updates within this window, e.g., due to energy constraints or duty-cycle regulations. Hence, the problem is modeled as a finite-horizon scheduling of the transmission opportunities.

This work was supported in part by grant no. 101122990-GO SPACE-ERC-2023-ADG. Çağrı Arı was also supported in part by Turk Telekom within the framework of the 5G and Beyond Joint Graduate Support Programme coordinated by the Information and Communication Technologies Authority.

The decision of scheduling a transmission is also affected by the delay experienced on satellite links [5]. From a modeling standpoint, we adopt a discrete (slotted) time so that these delays follow a discrete random distribution. Moreover, we derive different scheduling policies based on different levels of awareness of such delays.

II. Related work

In [6], the satellite uplink is considered for multi-access, with a specific focus on how packet collisions are affected by the different positions of the sending nodes in relation to the satellite and how this reflects on the Age of Information (AoI). However, from the perspective of scheduling, this is a plain slotted ALOHA, whose AoI can be analyzed as in [7]–[9], and in this regard, the authors extend the coordination of multiple users, but not their scheduling.

Reference [10] makes two other different contributions in the context of satellite communications and AoI. Specifically, it proposes a closed-form expression for AoI derived from the block error rate in a mixed satellite-terrestrial scenario, using either orthogonal or non-orthogonal multiple access. This specific analytical advancement actually relates to the multiple access part only; as the authors themselves claim, the analytical model is interchangeable and just applied to the design of the scenario considered there. A similar investigation is presented by [11] with the additional element of energy awareness, but again, the connectivity of the satellite is included in all the nodes involved in the non-orthogonal multiple access, being at the same time in line of sight with the satellite. Moreover, another contribution in [10] is in the analysis of coalition formation as a potential game based on the analytical characterization previously derived. As such, there is also no direct scheduling of the transmission.

A more direct relationship with our analysis is present in [5], where the overall LEO constellation and its intermittent connectivity are modeled through a variable delay arising from multiple satellites storing and forwarding the content before ultimately delivering it to the destination. A similar modeling approach is adopted in [12] for a more general setting involving tributary flows with different priorities, with the primary case study again focusing on a LEO constellation. Reference [13] also investigates significantly varying delay conditions in remote inference applications served through non-terrestrial networks. It considers a two-way delay between the sensing unit and the destination, where the network delay state evolves according to a finite-state ergodic Markov chain.

Finally, our previous work [4] considers a dynamic programming approach for an AoI-minimal scheduling. However, there are considerable differences with the present contribution, since that paper involves a minimization of the average AoI, as opposed to peak AoI, and also only investigates a single satellite link. As a result, the connection can only be available or not, but always results in a constant delay when available (which cannot further be optimized), an issue that we also investigated in [14]. In this contribution, we adopt not only another optimization objective, but also a different multi-dimensional model of the connection quality that expands the binary state of an intermittent connectivity to a variable delay.

III. Modeling approach and Contributions

To characterize the system under study, we consider a ground user that generates time-stamped status updates addressed to a monitoring gateway. End-to-end connectivity is achieved in two stages: (i) the user transmits the status updates to a LEO satellite through an uplink channel, and (ii) the LEO satellite forwards the updates to the gateway over a downlink network. The user operates over a finite horizon of T time slots and is allowed to make at most m transmission attempts.

The uplink channel is modeled to capture intermittent connectivity, alternating between periods of satellite coverage and intervals during which no link is available, as commonly observed in LEO constellations [15]. To this end, we assume that the uplink channel state U_n in any time slot $n \in \{1, 2, \ldots, T\}$ can take values in $\{1, 2\}$, representing the absence and presence of connectivity, respectively. The evolution of the process U_n is assumed to follow a two-state Markov chain that makes a single transition at the end of each time slot, thus determining the uplink channel state for the subsequent slot. When the uplink channel is available in the nth time slot, i.e., $U_n = 2$, the ground user can successfully deliver a status update to the LEO satellite over the slot.

Upon receiving a status update, the LEO satellite forwards it to the gateway through a downlink network. The transmission is always successful, but experiences a varying delay. Specifically, we assume that the downlink network state D_n in any time slot $n \in \{1, 2, \ldots, T\}$ can take N possible values, i.e. $D_n \in \{1, 2, \ldots, N\}$, each corresponding to a distinct constant transmission delay, denoted by $\{d_1, d_2, \ldots, d_N\}$ time slots, respectively. The process D_n evolves according to an N-state Markov chain that makes a single transition at the end of each time slot. If the LEO satellite forwards a status update in time slot n when the downlink network is in state $D_n = i$, the gateway receives the update d_i time slots later.

This modeling approach offers a simple, yet insightful way to capture the store-and-forward paradigm. Downlink network states with small delays correspond to scenarios where a direct link is available between the LEO satellite and the gateway, enabling immediate forwarding. Conversely, states with large delays represent cases where the satellite must store the update until its trajectory brings it into visibility of the gateway. Finally, intermediate delay states can model the latency introduced when updates pass through inter-satellite routes within the LEO constellation before reaching the gateway.

In this setup, we consider the AoI at the gateway, defined in any time slot $n \in \{1, 2, \ldots, T\}$ as [16]

$$\delta_n = n - \sigma_n, \tag{1}$$

where σ_n denotes the time slot in which the most recently delivered update by time slot n was generated at the ground terminal. Also note that we set $\sigma_n = 0$ if no updates have yet been delivered to the gateway by time slot n.

As our performance indicator, we focus on the peak AoI over the entire time horizon, defined as

$$\hat{\Delta} := \max_{n \in \{1,2,\ldots,T\}} \delta_n. \quad (2)$$

We study the problem of determining when the ground user should transmit, i.e., when it should send freshly generated status updates over the available time slots, to minimize the experienced peak AoI.

IV. Transmission Scheduling Techniques

In this section, we provide transmission schedulers for scenarios that capture a broad range of practical settings. Across these scenarios, the scheduler is assumed to possess progressively higher levels of connectivity awareness regarding the uplink channel and the downlink network.

A. No Connectivity Information

We begin with the simplest case, in which the ground user does not have information about the processes U_n and D_n. This setting is relevant for basic IoT terminals that can only transmit and are not provided with location information, such that the current availability of the satellite cannot be exploited.

For this case, we adopt a periodic schedule whose transmission happens in time slots

$$k_i = \left\lfloor \frac{iT}{m+1} \right\rceil, \, i \in \{1, \ldots, m\}$$

where $\lfloor\cdot\rceil$ denotes rounding to the nearest integer.

When the uplink channel is not available at any of the scheduled transmission time slots, the transmission fails and is not delivered to the gateway. Although this design is simple, it leverages the benefits of periodic updates to reduce the PAoI.

B. Transmission Scheduling Based on the Knowledge of Current Connectivity Conditions

A slightly more advanced solution foresees the possibility for the ground user to observe the availability of the uplink channel U_n, and the LEO satellite to be aware of state D_n. These conditions are inspired by practical settings, as terminals typically detect the presence of a downlink signal prior to attempting transmissions, and the current constellation and routing conditions are known at the satellite. In this setting, we assume that, in the presence of connectivity, the satellite transmits the current downlink state information D_n to the ground user at the beginning of the same time slot. As a result, the ground user can rely on knowledge of uplink availability and delay to be expected for a delivery when deciding whether to transmit an update. We further assume that the ground user cannot initiate a new transmission until the previous one has been successfully delivered to the gateway. This is primarily adopted for modeling convenience, yet stands reasonable, as the problem of interest typically involves a moderate number of transmissions m that must be efficiently allocated over a large time horizon. Although there is no end-to-end feedback from the gateway, the ground user can still determine the exact delivery time of each transmission, since each D_n corresponds to a deterministic downlink delay. For the same reason, the user can also track the AoI process δ_n at the gateway. However, the ground user cannot know when the uplink channel will become available again after the previous transmission, as the evolution of U_n is a stochastic process.

The optimal strategy in this case can be found resorting to a finite-horizon Semi-Markov Decision Process (SMDP), characterized by the following components.

Decision Time: Any time slot $n \in \{1, 2, \ldots, T\}$ is a decision time slot if the uplink channel is available, i.e., $U_n = 2$, and there is no ongoing transmission that has not yet been delivered to the gateway. Let $k_1, k_2, \ldots, k_M$ denote the finite number of decision times occurring over the time horizon of T slots.

State Definition: For any k_i, the system state is represented by the tuple $(\delta_{k_i}, D_{k_i}, r_{k_i}, \Delta_{k_i})$. Here, δ_{k_i} denotes the AoI at decision time $n = k_i$, as defined in (1), D_{k_i} is the current downlink network state, r_{k_i} indicates the remaining number of allowed transmissions, and Δ_{k_i} corresponds to the peak AoI incurred by decision time k_i.

Action: At time k_i, the ground user selects action a_{k_i}, which is either to initiate a new transmission or to remain idle. A new transmission can be initiated provided that it will be delivered no later than the final time slot T, i.e., $k_i + d_{D_{k_i}} \leq T$, and that at least one additional transmission is permitted, i.e., $r_{k_i} \geq 1$.

State Transitions: The state transitions depend on the action taken by the user and are described as follows:

- *Idle action:* If the ground user remains idle at decision time k_i with state $(\delta_{k_i}, D_{k_i}, r_{k_i}, \Delta_{k_i})$, the next decision time k_{i+1} is the smallest integer j such that $k_i + 1 \leq j \leq T$ and $U_j = 2$. The state at k_{i+1} becomes $(\delta_{k_{i+1}}, D_{k_{i+1}}, r_{k_{i+1}}, \Delta_{k_{i+1}})$, where
  - $\delta_{k_{i+1}} = \delta_{k_i} + k_{i+1} - k_i$
  - the downlink network state evolves according to its corresponding Markov chain
  - $r_{k_{i+1}} = r_{k_i}$
  - $\Delta_{k_{i+1}} = \max(\delta_{k_i} + k_{i+1} - k_i, \, \Delta_{k_i})$.
- *Transmit action:* If the ground user transmits at decision time k_i with state $(\delta_{k_i}, D_{k_i}, r_{k_i}, \Delta_{k_i})$, the next decision time k_{i+1} is the smallest integer j such that $k_i + d_{D_i} \leq j \leq T$ and $U_j = 2$. The state at k_{i+1} becomes $(\delta_{k_{i+1}}, D_{k_{i+1}}, r_{k_{i+1}}, \Delta_{k_{i+1}})$, where
  - $\delta_{k_{i+1}} = k_{i+1} - k_i$
  - the downlink network state evolves according to its corresponding Markov chain
  - $r_{k_{i+1}} = r_{k_i} - 1$
  - $\Delta_{k_{i+1}} = \max(k_{i+1} - k_i, \, \delta_{k_i} + d_{D_i} - 1, \, \Delta_{k_i})$.

We assume that the uplink channel is available at time slot $n = T$, i.e., $U_T = 2$. This assumption does not affect the AoI evolution, since the ground user cannot initiate a new transmission at time slot $n = T$, regardless of the uplink channel's availability. The purpose of this assumption is to ensure that, when there are remaining time slots in which the ground user cannot start a transmission, it keeps selecting the idle action, and the process terminates at the decision time

$k_M = T$ with state $(\delta_{k_M}, D_{k_M}, r_{k_M}, \Delta_{k_M})$. Here, Δ_{k_M} is equal to the peak AoI $\hat{\Delta}$ defined in (2).

Reward: The reward function $\mathcal{R}_n(\delta_n, D_n, r_n, \Delta_n)$ is nonzero only at the terminal time slot $n = k_M = T$ with state $(\delta_{k_M}, D_{k_M}, r_{k_M}, \Delta_{k_M})$, and is defined as

$$\mathcal{R}_{k_M}(\delta_{k_M}, D_{k_M}, r_{k_M}, \Delta_{k_M}) = -\Delta_{k_M}. \tag{3}$$

We can obtain the optimal transmission scheduling policy through dynamic programming via the backward recursion algorithm, which solves the SMDP defined above by maximizing the expected reward, i.e., by minimizing the expected peak AoI.

The optimal policy $\pi^*_{k_i}(\delta_{k_i}, D_{k_i}, r_{k_i}, \Delta_{k_i})$ specifies the optimal action, either *idle* or *transmit*, in any possible decision time slot $k_i \in \{1, 2, \ldots, T-1\}$ and for every state $(\delta_{k_i}, D_{k_i}, r_{k_i}, \Delta_{k_i})$. This policy is obtained in two parts:

Initialization: We initialize the value function in the terminal time slot $k_M = T$ for every state as

$$V_T(\delta_T, D_T, r_T, \Delta_T) = -\Delta_T. \tag{4}$$

Backward Recursion: We *sequentially* compute the value function $V_{k_i}(\delta_{k_i}, D_{k_i}, r_{k_i}, \Delta_{k_i})$, and thus the optimal action $\pi^*_{k_i}(\delta_{k_i}, D_{k_i}, r_{k_i}, \Delta_{k_i})$, for all possible decision time slots $k_i = T-1, T-2, \ldots, 1$ and for every state $(\delta_{k_i}, D_{k_i}, r_{k_i}, \Delta_{k_i})$ through the backward recursion algorithm that consists of the following steps:

1) The action-value function $Q_{k_i}(\delta_{k_i}, D_{k_i}, r_{k_i}, \Delta_{k_i}; a_{k_i})$ is evaluated for every state and for each action a_{k_i} at the possible decision time $k_i = T-1$:

$$\begin{aligned} &Q_{k_i}(\delta_{k_i}, D_{k_i}, r_{k_i}, \Delta_{k_i}; a_{k_i}) \\ &\quad = \mathbb{E}\left[V_{k_{i+1}}(\delta_{k_{i+1}}, D_{k_{i+1}}, r_{k_{i+1}}, \Delta_{k_{i+1}}) \,\middle|\, a_{k_i}\right]. \end{aligned} \tag{5}$$

2) The value function $V_{k_i}(\delta_{k_i}, D_{k_i}, r_{k_i}, \Delta_{k_i})$ and the optimal action $\pi^*_{k_i}(\delta_{k_i}, D_{k_i}, r_{k_i}, \Delta_{k_i})$ for every state at the decision time $k_i = T-1$ are computed as

$$\begin{aligned} &V_{k_i}(\delta_{k_i}, D_{k_i}, r_{k_i}, \Delta_{k_i}) \\ &\quad = \max_{a_{k_i} \in \{\text{IDLE,TRANSMIT}\}} Q_{k_i}(\delta_{k_i}, D_{k_i}, r_{k_i}, \Delta_{k_i}; a_{k_i}), \end{aligned} \tag{6}$$

$$\begin{aligned} &\pi^*_{k_i}(\delta_{k_i}, D_{k_i}, r_{k_i}, \Delta_{k_i}) \\ &\quad = \arg\max_{a_{k_i} \in \{\text{IDLE,TRANSMIT}\}} Q_{k_i}(\delta_{k_i}, D_{k_i}, r_{k_i}, \Delta_{k_i}; a_{k_i}). \end{aligned} \tag{7}$$

3-) Steps 1 and 2 are then repeated sequentially for all remaining possible decision times $k_i = T-2, T-3, \ldots, 1$.

Note that, as long as the value function $V_n(\delta_n, D_n, r_n, \Delta_n)$ evaluated in time slots $n > k_i$ is available, the conditional expectation in (5) can be computed. This condition is ensured at each iteration of the backward recursion algorithm by the initialization part and (6).

C. Transmission Scheduling Based on the Knowledge of Full Connectivity Conditions

In the last case, we consider a setting in which the transmission scheduler has access to the entire uplink channel availability sequence $\{U_1, U_2, \ldots, U_T\}$ and the downlink network delay state sequence $\{D_1, D_2, \ldots, D_T\}$ when making decisions. This setting may correspond to some real-time applications in which the ground user precisely knows the satellite's visibility duration, for instance, by leveraging location information and pre-loaded satellite ephemerides.

For any given sequences $\{U_1, \ldots, U_T\}$ and $\{D_1, \ldots, D_T\}$, this transmission scheduling problem can be modeled as a deterministic finite-horizon SMDP, similar to the one defined in Section IV-B. The only difference is that the state transitions occur according to the pre-known sequences, rather than stochastically as in the Markov chain case. The optimal transmission scheduling policy can still be obtained using the backward recursion algorithm. Equations (4)–(7) remain valid; whereas the expectation in (5) is no longer required.

V. Simulation Results

In this section, we compare the average PAoI achieved by the three transmission scheduling policies obtained throughout the paper. We consider a finite horizon of $T = 100$ time slots. The Markov chain D_n, representing the downlink network, has $N = 2$ states with transition probability matrix

$$P_D = \begin{bmatrix} 0.75 & 0.25 \\ 0.25 & 0.75 \end{bmatrix}.$$

The downlink network states $D_n \in \{1, 2\}$ correspond to store-and-forward delays of 2 and 5 time slots, respectively.

For convenience, we use the abbreviation TSw/CC to denote the periodic transmission scheduling policy obtained for the case in which the ground user has no information about the connectivity conditions. Likewise, we use TSwCCC and TSwFCC to denote, respectively, the policies obtained for the cases in which the user has access to the current connectivity conditions or full connectivity conditions for scheduling.

Fig. 1 shows the average PAoI performance achieved by the policies for the number of allowed transmissions $m \in \{3, 4, \ldots, 15\}$. Here, we set the transition probability matrix P_U of the Markov chain U_n, representing the uplink channel availability, equal to P_D.[1] As seen in Fig. 1, not having information about the connectivity conditions results in a significant performance degradation: TSw/CC achieves approximately twice the PAoI of the other policies. Moreover, the other two policies perform similarly. The key distinction is that, although TSwFCC achieves a slightly lower PAoI, it is computationally more expensive than TSwCCC, since TSwCCC can be reused across multiple tasks as long as the statistics of the connectivity conditions remain the same, i.e., the Markov chains U_n and D_n do not change across tasks. In contrast, TSwFCC must be recomputed for each specific pair

[1] In the simulations, U_1 and D_1 are randomly initialized according to the steady-state distributions of their corresponding Markov chains.

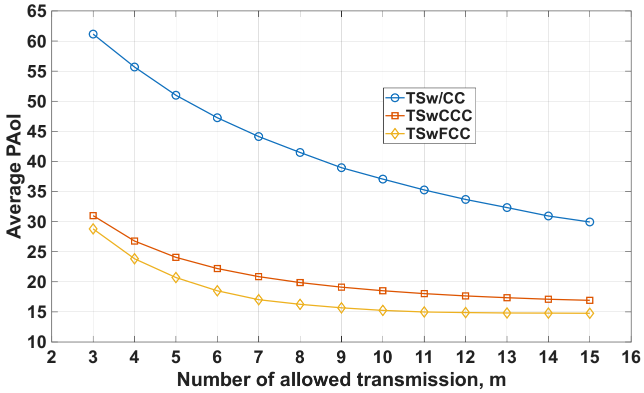

Fig. 1. Average PAoI vs number of allowed transmissions m

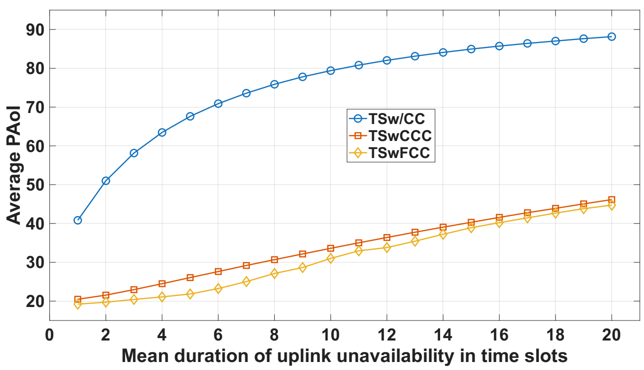

Fig. 2. Average PAoI vs mean duration of uplink unavailability in time slots

of sequences $\{U_1, \ldots, U_T\}$ and $\{D_1, \ldots, D_T\}$. The results offer an interesting insight, showing a simple knowledge of the current connectivity conditions is enough to reap most of the improvements.

Fig. 2, on the other hand, reports the average PAoI for different mean durations of uplink unavailability (in time slots). Here, we set the number of allowed transmissions to $m = 5$, and the transition probability matrix P_U of the Markov chain U_n is given by

$$P_U = \begin{bmatrix} 1-p & p \\ 0.5 & 0.5 \end{bmatrix}.$$

Note that the mean duration of uplink unavailability is $1/p$, i.e., the expected number of steps for the Markov chain U_n to leave state 1. As expected, the performance degrades for all policies as the mean duration of uplink unavailability increases, due to the lower frequency at which updates can be delivered. On the other hand, substantial improvements are attained with a simple scheduling approach that leverages current connectivity conditions, whereas minimal further gains are triggered by the knowledge of future network conditions.

VI. Conclusion

In this paper, we study the scheduling of status updates within a finite time horizon for a real-time, task-oriented application operating over an intermittently available satellite-based network that employs a store-and-forward policy. We consider three progressively more complex transmission scheduling problems in terms of connectivity awareness. In the simplest case, where the ground user has no information about the uplink channel availability or the downlink network state, we adopt a periodic transmission strategy. The other two scheduling problems—corresponding to settings in which the ground user has access to either the current or full information about the uplink channel and downlink network states—are formulated as Semi-Markov Decision Processes (SMDPs) and solved using a dynamic programming algorithm, namely backward induction. Our results demonstrate that awareness of satellite connectivity has a significant impact on scheduling decisions and peak AoI performance.

References

[1] A. Shamsoshoara, F. Afghah, A. Razi, S. Mousavi, J. Ashdown, and K. Turk, "An autonomous spectrum management scheme for unmanned aerial vehicle networks in disaster relief operations," *IEEE Access*, vol. 8, pp. 58 064–58 079, 2020.

[2] T. Darwish, G. K. Kurt, H. Yanikomeroglu, M. Bellemare, and G. Lamontagne, "LEO satellites in 5G and beyond networks: A review from a standardization perspective," *IEEE Access*, vol. 10, pp. 35 040–35 060, 2022.

[3] M. A. Abd-Elmagid and H. S. Dhillon, "Average peak age-of-information minimization in UAV-assisted IoT networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 2, pp. 2003–2008, 2018.

[4] L. Badia and A. Munari, "Satellite intermittent connectivity and its impact on age of information for finite horizon scheduling," in *Proc. IEEE Adv. Sat. Multimedia Syst. Conf. Signal Proc. Space Commun. (ASMS/SPSC)*, 2025, pp. 1–8.

[5] Y. Li, Y. Xu, Q. Zhang, and Z. Yang, "Age-optimized multihop information update mechanism on the LEO satellite constellation via continuous time-varying graphs," *IEEE Internet Things J.*, vol. 10, no. 8, pp. 7189–7203, 2022.

[6] D. Zhang, Y. Peng, Y. Ke, J. Song, S. Wang, G. Pan, and J. An, "Age of information variation of LEO satellite-terrestrial uplink transmissions," *IEEE Trans. Veh. Technol.*, 2025.

[7] X. Chen, K. Gatsis, H. Hassani, and S. S. Bidokhti, "Age of information in random access channels," *IEEE Trans. Inf. Theory*, vol. 68, no. 10, pp. 6548–6568, 2022.

[8] L. Badia, "Impact of transmission cost on age of information at nash equilibrium in slotted ALOHA," *IEEE Netw. Lett.*, vol. 4, no. 1, pp. 30–33, 2021.

[9] 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.

[10] Z. Gao, A. Liu, C. Han, and X. Liang, "Non-orthogonal multiple access-based average age of information minimization in LEO satellite-terrestrial integrated networks," *IEEE Trans. Green Commun. Netw.*, vol. 6, no. 3, pp. 1793–1805, 2022.

[11] C.-Y. Lin and W. Liao, "Energy-aware age of information (AoI) minimization for Internet of things in NOMA-based LEO satellite networks," in *Proc. IEEE VTC Spring*, 2024, pp. 1–5.

[12] F. Chiariotti, O. Vikhrova, B. Soret, and P. Popovski, "Age of information in multihop connections with tributary traffic and no preemption," *IEEE Trans. Commun.*, vol. 70, no. 10, pp. 6718–6733, 2022.

[13] C. Ari, M. K. C. Shisher, Y. Sun, and E. Uysal, "Goal-oriented status updating for real-time remote inference over networks with two-way delay," *arXiv preprint arXiv:2410.08706*, 2024.

[14] L. Badia, A. Zancanaro, G. Cisotto, and A. Munari, "Status update scheduling in remote sensing under variable activation and propagation delays," *Ad Hoc Networks*, vol. 163, p. 103583, 2024.

[15] S. Kozhaya, H. Kanj, and Z. M. Kassas, "Multi-constellation blind beacon estimation, Doppler tracking, and opportunistic positioning with OneWeb, Starlink, Iridium NEXT, and Orbcomm LEO satellites," in *Proc. IEEE/ION Pos. Location Navig. Symp. (PLANS)*, 2023.

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