Scheduling Agile Earth Observation Satellites with Onboard Processing and Real-Time Monitoring

Antonio M. Mercado-Martínez, Beatriz Soret *Senior Member, IEEE*, Antonio Jurado-Navas *Member, IEEE*

***Abstract*—The emergence of Agile Earth Observation Satellites (AEOSs) has marked a significant turning point in the field of Earth Observation (EO), offering enhanced flexibility in data acquisition. Concurrently, advancements in onboard satellite computing and communication technologies have greatly enhanced data compression efficiency, reducing network latency and congestion while supporting near real-time information delivery. In this paper, we propose a framework for addressing the Agile Earth Observation Satellite Scheduling Problem (AEOSSP) in a real-time remote monitoring scenario integrating onboard data processing. The AEOSSP consists of determining the optimal sequence of target observations to maximize overall observation profit, and, to this end, we define a set of priority indicators and develop a constructive heuristic method, further enhanced with a Local Search (LS) strategy. The results show that the proposed algorithm provides high-quality information by increasing the spatial resolution of the collected frames by up to 18.6% on average, while reducing the variance in the monitoring frequency of the targets within the instance by up to 84%, ensuring more up-to-date information across the entire set compared to First-In First-Out (FIFO) and genetic algorithm (GA)-based methods.**

I. Introduction

Earth Observation (EO) is a rapidly evolving and highly relevant application in satellite communications, providing critical insights for climate monitoring, disaster management, maritime surveillance, or vehicle tracking. Traditionally, EO has involved the acquisition and transmission of vast volumes of data, posing challenges for both communication and processing. However, the enhanced onboard computing and communication capabilities of modern satellites enable efficient data compression, reducing network congestion and latency while supporting near real-time information delivery [1].

These advancements are further reinforced by the emergence of Agile Earth Observation Satellites (AEOSs) [2]. While Conventional Earth Observation Satellites (CEOSs) can only adjust their attitude along the roll axis, AEOSs benefit from full three-axis attitude control (roll, pitch, and yaw), significantly increasing the flexibility of data acquisition by extending the Visible Time Window (VTW) – the time interval during which a specific target can be observed. Consequently, the selection of the actual observation time for a given observation target, referred to as the Observation Time Window (OTW), becomes more flexible. This flexibility expands feasible observation schedules but introduces complexity when selecting OTWs across multiple observation targets within the same Scheduling Time Horizon (STH). This is known as the Agile Earth Observation Satellite Scheduling Problem (AEOSSP), which aims to maximize the total observation profit for a given set of targets and satellites over the STH, while satisfying all temporal, energy, and storage constraints.

The observation profit depends on the context and is commonly defined by target priority and frame quality. The later depends on the relative position between satellite and target during the OTW, leading to the concept of time-dependent profit. A widely used indicator of frame quality in EO is the spatial resolution, typically measured by the ground sample distance (GSD) [3], which quantifies the actual ground area represented by a single pixel in the frame. As the target deviates from the satellite's nadir point, the GSD increases, leading to reduced spatial resolution.

Early research on AEOSSP primarily focused on single-satellite scenarios, which are already challenging. Extending the problem to multi-satellite configurations introduces additional complexity, as each target may be observed by multiple satellites. Solutions are classified into distributed and centralized approaches. In distributed schemes, each satellite operates as an autonomous agent, independently scheduling its own list of targets while coordinating with the rest of satellites in the constellation. Centralized methods, in contrast, rely on a central node –controller– to assign observation tasks to all satellites in the constellation, sending each satellite a predefined set of instructions to execute [4]. While distributed strategies are often more scalable and time-efficient, they may yield suboptimal global performance and require constant inter-satellite communication. In contrast, centralized approaches offer globally optimized solutions and are particularly well-suited for small satellite constellations.

In this paper, we present a centralized framework for addressing the AEOSSP tailored to continuous remote monitoring on ground, ensuring high-quality and fresh information of a given set of targets while leveraging the onboard processing capabilities of satellites. To achieve this, we integrate data processing into the AEOSSP formulation and

A. M. Mercado-Martínez, B. Soret, and A. Jurado-Navas are with the Telecommunications Research Institute (TELMA), Universidad de Málaga, 29071, Málaga, Spain. The work of A.M. Mercado-Martínez, B. Soret, and A. Jurado-Navas is partially supported by the Spanish Ministerio de Ciencia e Innovación under grant PID2022-136269OB-I00 funded by MCIN/AEI/10.13039/501100011033 and "ERDF A way of making Europe". The work of A.M. Mercado-Martínez is also partially supported by Grant DGP_PRED_2024_01603, funded by the Consejería de Universidad, Investigación e Innovación of Junta de Andalucía and the European Union. The authors thankfully acknowledge the computer resources, technical expertise and assistance provided by the SCBI (Supercomputing and Bioinformatics) center of the University of Málaga.

define the observation profit as a function of both spatial resolution and data freshness. Several priority indicators are introduced, and a constructive heuristic algorithm is designed to efficiently solve the problem, using GSD and Age of Information (AoI) as key performance indicators. The rest of the paper is organized as follows. We first provide an overview of related works (Section II) and system model (Section III). Then, the timing metrics, optimization problem, priority indicators and algorithm are described in Sections IV-VII. Results and conclusions are presented in Sections VIII and IX, respectively.

II. Related work

In recent years, the AEOSSP has attracted significant attention from the research community. Existing approaches can be broadly categorized into exact, heuristic, metaheuristic, and machine learning methods. Among these, heuristic and metaheuristic techniques have gained prominence over exact methods due to their ability to provide high-quality solutions with lower computational cost. For example, [5] propose several priority-based constructive heuristic methods with promising results, while [6] addresses the AEOSSP under oversubscribed target scenarios using a feedback structured heuristic, a scenario closely related to our work.

Parallel to advances in scheduling techniques, edge computing has emerged as a key enabler for reducing latency and congestion in satellite network communications. In [1], the authors propose an edge computing framework for real-time, very-high-resolution EO imaging, while [7] focus on resource allocation strategies in satellite constellations for EO. In addition, [8] explores satellite edge computing in a real-world application, aiming to design and size a constellation that balances frame quality with information freshness, using AoI as a key performance metric. AoI is a widely used metric to quantify the timeliness of information, measuring the time elapsed since the last received update [9]. If no updates occur, the AoI increases linearly; upon receiving a new update, it resets to reflect the time elapsed since the generation of that update. In the context of EO, AoI represents the time since a particular target was last observed. Over time, various extensions of AoI have been proposed, including the Peak Age of Information (PAoI), which is particularly suited for worst-case optimization scenarios, as it captures the AoI value just before an update is received.

III. System Model

Fig. 1a shows the considered scenario, which involves a set of AEOSs S, deployed in a constellation comprising N_{OP} orbital planes and N_{sat} satellites per plane, operating at an altitude h with an orbital inclination ι. The satellites are arranged according to a Walker-Delta or Walker-Star topology [10]. These satellites are responsible for observing a set of targets T within a designated observation area, processing the acquired data –i.e. a frame of the target area– and transmitting the resulting information to the ground segment, represented by a set G of ground stations, when within coverage following a store-and-forward approach (Fig. 1b). All the targets in the instance must be observed as regularly as possible over a STH defined as $STH = [sth_a, sth_b]$, where sth_a and sth_b denote the start and end times of the horizon, respectively. To achieve this, the STH is divided into N_{STP} Scheduling Time Periods (STPs), where the n-th STP is given by $STP^{(n)} = [stp_a^{(n)}, stp_b^{(n)}]$, with $stp_a^{(n)}$ and $stp_b^{(n)}$ representing its start and end times. For each STP, every satellite in the deployment receives a new schedule, allowing previous schedules to be taken into account when constructing the current one. Each target $t \in T$ requires an observation duration of $\tau_t^{(obs)}$ seconds and may be observed by a satellite $s \in S$ during multiple orbital passes. Let $O_{s,t}$ denote the set of orbits in which satellite s can observe target t. The corresponding VTW for each satellite s and target t during orbit $o \in O_{s,t}$ triplet is defined as $VTW_{s,t,o}$ spanning the interval $[sw_{s,t,o}, ew_{s,t,o}]$, where $sw_{s,t,o}$ and $ew_{s,t,o}$ are the start and end times of the VTW, respectively. Each $VTW_{s,t,o}$ is further discretized into a set of OTWs with a fixed time step of prc seconds. Thus, $OTW_{s,t,o,w}$ represents the OTW for satellite s and target t during orbit o within $VTW_{s,t,o}$, with index $w \in W_{s,t,o}$ where $W_{s,t,o}$ denotes the set of OTWs contained in $VTW_{s,t,o}$. An $OTW_{s,t,o,w}$ is defined by $[so_{s,t,o,w}, eo_{s,t,o,w}]$, and an associated observation profit $\rho_{s,t,o,w}$, where $so_{s,t,o,w}$ and $eo_{s,t,o,w}$ represent the start and end times of the OTW according to $\tau_t^{(obs)}$, respectively.

The attitude required for observing $OTW_{s,t,o,w}$ is defined by its roll ($\theta_{s,t,o,w}$), pitch ($\phi_{s,t,o,w}$), and yaw ($\psi_{s,t,o,w}$) angles. Each satellite's maneuvering capabilities are restricted by its maximum roll, pitch, and yaw angles (θ_{max}, ϕ_{max}, and ψ_{max}, respectively), as well as by its attitude transition time. These factors dictate which targets can be observed and scheduled at any given time. The attitude transition time is described by a piecewise linear function that depends on the speed of the camera and the angular difference between two consecutive observations. Specifically, we consider the following attitude transition time between two consecutive observations $OTW_{s,t,o,w}$ and $OTW_{s,t',o,w'}$ [11]:

$$\Delta t_{t,w-t',w'} = \begin{cases} 11.66, & \alpha_{t,w-t',w'} \leq 10^\circ \\ 5 + \alpha_{t,w-t',w'}/1.5, & 10^\circ < \alpha_{t,w-t',w'} \leq 30^\circ \\ 10 + \alpha_{t,w-t',w'}/2, & 30^\circ < \alpha_{t,w-t',w'} \leq 60^\circ \\ 16 + \alpha_{t,w-t',w'}/2.5, & 60^\circ < \alpha_{t,w-t',w'} \leq 90^\circ \\ 22 + \alpha_{t,w-t',w'}/3, & \alpha_{t,w-t',w'} > 90^\circ \end{cases}, \tag{1}$$

$$\begin{aligned} \alpha_{t,w-t',w'} =& |\theta_{s,t,o,w} - \theta_{s,t',o,w'}| + |\phi_{s,t,o,w} - \phi_{s,t',o,w'}| \\ &+ |\psi_{s,t,o,w} - \psi_{s,t',o,w'}|, \end{aligned} \tag{2}$$

where $\alpha_{t,w-t',w'}$ is the total attitude transition angle between $OTW_{s,t,o,w}$ and $OTW_{s,t',o,w'}$.

Each satellite is equipped with an onboard Central Processing Unit (CPU) that processes the collected data in real time. The processed data, D', is determined by the compression factor σ, defined as:

$$D' = \frac{D}{\sigma}, \tag{3}$$

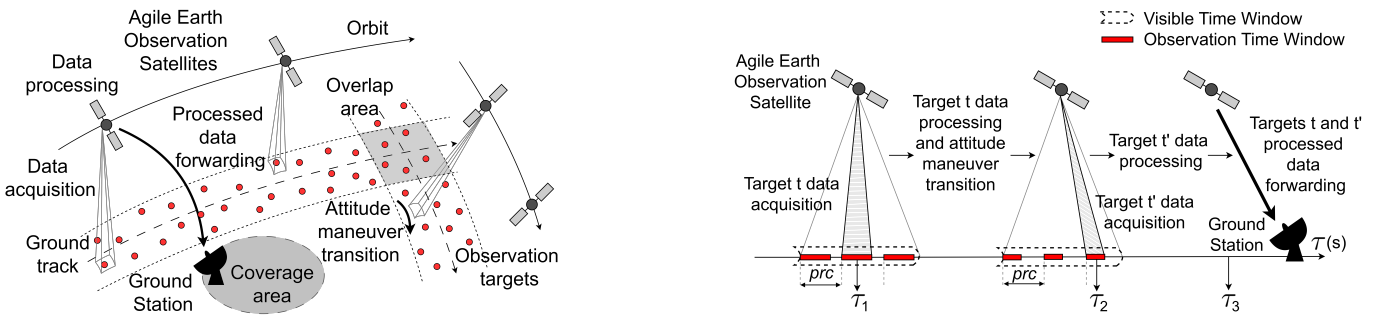

Fig. 1: (a) Scenario with multiple AEOSs tasked with observing multiple targets, processing and transmitting the acquired data. (b) Sketch of the task sequence: a satellite observes target t at τ_1 for $\tau_t^{(obs)}$ seconds, processes the acquired data, executes the corresponding attitude maneuvers, observes target t' at τ_2 for $\tau_{t'}^{(obs)}$ seconds, and processes the newly acquired data. The overall processed data is transmitted to ground once a ground station is reachable at τ_3.

where D is the amount of data to be processed. The compression factor can range from a few units or tens (traditional compression algorithms) [1]; to thousands when using semantic information extraction methods in the context of semantic or goal-oriented communications [8]. For the case of a CPU (and similarly for a Graphics Processing Unit (GPU)), the processing time is calculated as:

$$\tau_{s,t,o,w}^{(proc)} = \frac{D_{s,t,o,w} C}{N_{cores} f_{CPU}}, \tag{4}$$

where $D_{s,t,o,w}$ is the number of bits to be processed for $OTW_{s,t,o,w}$, C the complexity of the compression algorithm expressed in CPU cycles per bit, and N_{cores} and f_{CPU} the number of CPU cores and work frequency, respectively. $D_{s,t,o,w}$ is defined as:

$$D_{s,t,o,w} = \frac{\xi_{s,t,o,w}(\xi_{s,t,o,w} + \tau_t^{(obs)} R_E \omega)}{GSD_{s,t,o,w}^2} q_{px}, \tag{5}$$

where $\xi_{s,t,o,w}$ is the swath width, i.e, the width of the observable area on the surface of the Earth, determined by the field of view (FoV) of the sensor and the relative position between target t and satellite s during $OTW_{s,t,o,w}$; $GSD_{s,t,o,w}$ is the GSD for $OTW_{s,t,o,w}$; R_E is the Earth's radius; ω is the satellite angular velocity; and q_{px} is the bit depth of a pixel in the frame.

Energy consumption for observation, transition, and processing is modeled via E_{obs}, E_{tran}, and E_{proc}, respectively, and all representing energy per time unit. The energy budget for each satellite s during a STP is limited by E_{max}.

Satellites communicate with ground stations when they are within coverage, taking into account for both propagation and transmission delays. Thus, the communication time is calculated as follows:

$$\tau_{comm} = \frac{D_l}{R_l} + \frac{d_l}{c}, \tag{6}$$

where D_l is the number of bits to transmit through a given link l, R_l the bit rate of such link, d_l the distance between two nodes of the path, and c the speed of light. The time elapsed between data processing and its transmission is denoted as τ_{store}.

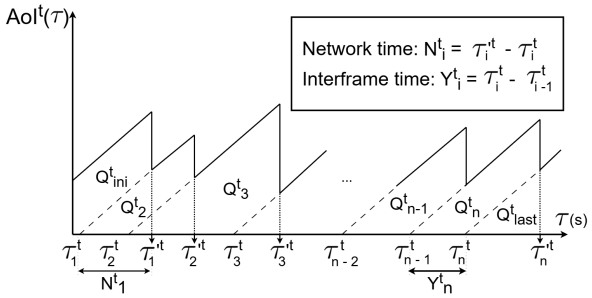

Fig. 2: Evolution of the AoI of target t.

IV. Timing Metrics

Each observation target t is associated with an AoI value denoted as AoI^t, which accumulates over time until updated, including processing, store, and communication times. In our application, AoI is defined as the time elapsed at the ground segment since the last received compressed image of target t was captured by the corresponding satellite [8]. Mathematically, this is expressed as follows. Fig. 2 shows the time evolution of the AoI of a target t. It is assumed that the system is first observed at time $\tau = 0$ and index i represents the number of the frame for target t. A frame i of target t is captured at $\tau = \tau_i^t$ and the corresponding compressed information is received at the ground segment at $\tau = \tau_i'^t$.

N_i^t is defined as the total network time of the system (processing + store + communication times), $N_i^t = \tau_i'^t - \tau_i^t$; and Y_i^t as the interframe time, the time between the capture of two frames for the same target t, $Y_i^t = \tau_i^t - \tau_{i-1}^t$.

The average AoI of target t can be calculated as follows:

$$\overline{AoI^t} = \frac{1}{sth_b - sth_a} \left(Q_{ini}^t + Q_{last}^t + \sum_{i=2}^{N(STH)} Q_i^t \right); \tag{7}$$

where $N(STH)$ is the number of arrivals by the end of the STH. Each Q_i^t for $1 < i$ is a trapezoid whose area can be

calculated as:

$$Q_i^t = \frac{1}{2}\left(N_i^t + Y_i^t\right)^2 - \frac{1}{2}\left(N_i^t\right)^2 = Y_i^t N_i^t + \frac{\left(Y_i^t\right)^2}{2}. \quad (8)$$

Assuming an initial observation of the system at $\tau = 0$, the average PAoI can be calculated as:

$$\overline{PAoI^t} = \frac{1}{N(STH)}\left(\tau_1'^t + \sum_{i=2}^{N(STH)} (\tau_i'^t - \tau_{i-1}^t)\right). \quad (9)$$

Furthermore, for our scenario, we define δ_t as the number of STPs that have elapsed since target t was last included in a schedule. This metric is directly related to AoI, since targets that have not been observed for a longer time are associated with higher AoI values.

V. Agile Earth Observation Satellite Scheduling Problem

The AEOSSP aims to maximize the total collected observation profit while satisfying all temporal, energy, and storage constraints. In the considered scenario, since data is processed dynamically as it is acquired and transmitted to the ground, storage constraints are neglected, and processing is incorporated into the optimization problem. The most relevant parameters of the optimization problem are summarized in Table I. The mathematical formulation is as follows:

$$\max \quad \sum_{s\in S}\sum_{t\in T}\sum_{o\in O_{s,t}}\sum_{w\in W_{s,t,o}} \rho_{s,t,o,w}\cdot x_{s,t,o,w}, \quad (10a)$$

subject to

$$sw_{s,t,o} \le so_{s,t,o,w} \le ew_{s,t,o} - \tau_t^{(obs)}, \quad (10b)$$

$$so_{s,t,o,w} + \tau_t^{(obs)} + \max\left(\Delta\tau_{t,w-t',w'}, \tau_{s,t,o,w}^{(proc)}\right) \le so_{s,t',o,w'} \quad (10c)$$

$$\sum_{t\in T}\sum_{w\in W_{s,t,o}} x_{t,s,o,w}\cdot\left(E_{obs}\cdot\tau_t^{(obs)} + E_{proc}\cdot\tau_{s,t,o,w}^{(proc)}\right) + \sum_{t,t'\in T}\sum_{w,w'\in W_{s,t,o}} y_{t,w,t',w'}\cdot\left(E_{tran}\cdot\Delta\tau_{t,w-t',w'}\right) \le E_{max}, \quad (10d)$$

$$\sum_{s\in S}\sum_{o\in O_{s,t}}\sum_{w\in W_{s,t,o}} x_{s,t,o,w} \le 1, \quad (10e)$$

$$x_{s,t,o,w}, y_{t,w,t',w'} \in \{0,1\}, \quad (10f)$$

where $x_{s,t,o,w}$ and $y_{t,w,t',w'}$ are binary decision variables, $x_{s,t,o,w} = 1$ denotes that $OTW_{s,t,o,w}$ is scheduled, otherwise $x_{s,t,o,w} = 0$, and $y_{t,w,t',w'} = 1$ that $OTW_{s,t,o,w}$ is scheduled right before $OTW_{s,t',o,w'}$, otherwise $y_{t,w,t',w'} = 0$. Equation (10a) is the optimization objective function, which is to maximize the sum of collected profits; Equation (10b) represents the VTW constraint, which ensures that the observation of target t occurs within one of its VTW; Equation (10c) represents the attitude transition and processing time constraint, ensuring that $OTW_{s,t',o,w'}$ can only be scheduled after $OTW_{s,t,o,w}$ if the information associated with $OTW_{s,t,o,w}$ has been fully processed and the attitude maneuver is feasible; Equation (10d) represents the energy constraint, which guarantees that the total energy consumption does not exceed the maximum allowed energy; Equation (10e) denotes that a target can exist at most once in a single schedule; and Equation (10f) represents the value of the decision variables.

TABLE I: Parameter description.

Parameter	Description
$\rho_{s,t,o,w}$	Observation profit for $OTW_{s,t,o,w}$, $s \in S$, $t \in T$, $o \in O_{s,t}$, $w \in W_{s,t,o}$
$sw_{s,t,o}$, $ew_{s,t,o}$	Start and end time for $VTW_{s,t,o}$, $s \in S$, $t \in T$, $o \in O_{s,t}$
$ow_{s,t,o}$, $ow_{s,t,o}$	Start and end time for $OTW_{s,t,o,w}$, $s \in S$, $t, t' \in T$, $o \in O_{s,t}$, $w, w' \in W_{s,t,o}$
E_{obs}, E_{proc}, E_{tran}	Energy consumption per unit of observation, processing, and attitude transition time of AEOS s, $s \in S$
E_{max}	Energy capacity during a STP of AEOS s, $s \in S$
$\tau_t^{(obs)}$	Duration of the observation for target t, $t \in T$
$\tau_{s,t,o,w}^{(proc)}$	Processing time for $OTW_{s,t,o,w}$, $s \in S$, $t \in T$, $o \in O_{s,t}$, $w \in W_{s,t,o}$
$\Delta\tau_{t,w-t',w'}$	Attitude transition transition time between $OTW_{s,t,o,w}$ and $OTW_{s,t',o,w'}$, $s \in S$, $t, t' \in T$, $o \in O_{s,t}$, $w, w' \in W_{s,t,o}$
$x_{s,t,o,w}$	Binary decision variable; $x_{s,t,o,w} = 1$ denotes that $OTW_{s,t,o,w}$ is scheduled, otherwise $x_{s,t,o,w} = 0$, $s \in S$, $t \in T$, $o \in O_{s,t}$, $w \in W_{s,t,o}$
$y_{t,w-t',w'}$	Binary decision variable; $y_{t,w,t',w'} = 1$ denotes that $OTW_{s,t,o,w}$ is scheduled right before $OTW_{s,t',o,w'}$, otherwise $y_{t,w,t',w'} = 0$, $s \in S$, $t, t' \in T$, $o \in O_{s,t}$, $w, w' \in W_{s,t,o}$

We consider both the quality of the collected frames, represented by the GSD, and the freshness of the last received information, quantified by the AoI, when defining the observation profit. The observation profit $\rho_{s,t,o,w}$ ranges from 0 to 1, where 1 indicates that $OTW_{s,t,o,w}$ has the highest possible priority and quality. We formulate it as:

$$\rho_{s,t,o,w} = \frac{GSD_{nadir}}{GSD_{s,t,o,w}}\frac{\delta_t}{\delta_{max}}, \quad (11)$$

where GSD_{nadir} is the GSD at nadir, and δ_{max} the δ_t value of the target that has remained unobserved for the longest time in the instance. As mentioned previously, the GSD increases as the target moves away from the nadir reaching its minimum at the nadir itself. A small GSD value leads to a better resolution. Therefore, the first term of the observation profit accounts for spatial resolution, while the second term ensures that targets with higher AoI values yield higher profit.

VI. Priority Indicators

When scheduling multiple AEOS, it is common practice to design priority indicators to ensure that the selected targets in the final schedule yield the highest possible profit.

In our scenario, the primary priority indicator is δ_t, as it directly correlates with the AoI. A higher δ_t value indicates higher priority.

Within the same STP, not all targets have the same number of scheduling opportunities, as the number of OTWs varies for each target in the instance. We define the number of available OTW for target t as its assignment flexibility [12], denoted as FL_t. A higher FL_t indicates more opportunities for inclusion in the final schedule, whereas a lower FL_t signifies higher priority among targets with the same δ_t value.

When selecting the OTW for target t, not only its associated observation profit should be considered, but also the observation profit of conflicting OTWs that would be excluded from the schedule. Two OTWs are considered to be in conflict when, disregarding the energy constraint (Equation 10d), scheduling both would render the optimization problem infeasible. This concept is well captured by the Opportunity Cost (OC) [5], which is mathematically defined as the sum of the observation profits of the conflicting OTWs:

$$OC_{s,t,o,w} = \sum_{t',w' \in COTW_{s,t,o,w}} \rho_{s,t',o,w'}, \tag{12}$$

where $COTW_{s,t,o,w}$ represents the set of OTWs that conflict with $OTW_{s,t,o,w}$. A lower $OC_{s,t,o,w}$ value signifies higher priority among the OTWs for the same target.

VII. Algorithm

Based on the priority indicators introduced in Section VI, we propose a constructive heuristic algorithm to solve the AEOSSP for continuous monitoring. For each STP, the targets are grouped based on δ_t, with groups arranged in descending order of δ_t. Within each group, targets are ranked in ascending order of FL_t, and their OTWs are sorted in ascending order of OC. The procedure is carried out as follows: for each target t, the first available $OTW_{s,t,o,w}$ is checked against the scheduling constraints. If feasible, it is added to the schedule, and the next target t' is considered; otherwise, the next OTW for the same target is evaluated until either one is scheduled or all available OTWs are exhausted. This process is repeated for every group until all targets have been considered.

While constructive heuristics provide fast solutions, they can also be suboptimal, as they rely on greedy decision-making without exploring alternative configurations that could yield better overall results. For this reason, Local Search (LS) [5] or feedback [6] strategies are applied to refine the initial solution. We propose a LS strategy in which new solutions are generated following an insertion and removal policy, i.e., attempting to schedule previously unscheduled targets by removing some of the scheduled ones. The procedure unfolds as follows:

Step 1: Unscheduled targets and their corresponding OTWs are listed. The set of unscheduled targets is denoted as U. The OC for every listed OTW is computed accounting only for the conflicting missions in the current schedule. Both targets and OTWs are then arranged following the same priority policies used in the constructive heuristic.

Step 2: Let u and $OTW_{s,u,o,w}$ be the current unscheduled target and OTW, respectively. If temporal constraints are met, proceed to Step 3; otherwise, check wether $OC_{s,u,o,w}$ is lower than $\rho_{s,u,o,w}$. If so, provisionally remove the conflicting scheduled missions and move to Step 3; otherwise consider the next OTW for u or, if no more OTWs are available for target u, the next unscheduled target u' and repeat Step 2.

Step 3: Check energy constraints. If the schedule is feasible, add $OTW_{s,u,o,w}$ to the schedule and consider the next unscheduled target u'; otherwise, remove the scheduled OTWs with lower observation profit from the schedule of satellite s until the optimization problem is feasible and add provisionally $OTW_{s,u,o,w}$. If the new solution improves upon the current one, update the schedule; otherwise, consider the next OTW for u or, if no more OTWs are available for target u, the next unscheduled target u' and go to Step 2.

This process is repeated until all unscheduled targets have been considered. The pseudocode of the proposed algorithm is deployed in Algorithm 1.

This approach is designed to produce high-quality solutions at low computational cost, which is crucial in the proposed scenario, where schedules must be continuously generated and delivered to the corresponding AEOS. This makes the proposed method particularly advantageous in terms of computational efficiency and response time when compared to more resource-intensive alternatives, such as genetic algorithms (GAs).

VIII. Results

The simulation parameters are as follows. The space segment is composed of $N_{op} = 4$ orbital planes and $N_{sat} = 2$ satellites per orbital plane, deployed at $h = 600$ kilometers, with an orbital inclination $\iota = 53°$, distributed according to a walker delta topology. Each satellite owns a CPU with $N_{cores} = 8$ and $f_{CPU} = 1.8$ GHz, and a camera whose attitude maneuvers are limited by $\theta_{\max} = 45°$, $\phi_{\max} = 45°$, and $\psi_{\max} = 90°$ that allows a $GSD_{nadir} = 0.5$ m/pixel and a swath from $\xi = 5$ to $\xi = 7.29$ kilometers. The bit depth of the captured frames is set to $q_{px} = 11$ bits/pixel. E_{obs}, E_{proc}, and E_{tran} values are set to 2 units/second, while $E_{max} = 5000$ units. We consider a complexity of the compression algorithm of $C = 100$ CPU cycles per bit and a compression factor of $\sigma = 10$ is assumed. The KSAT Global Ground Station network is regarded as the ground segment. For the communication environment, the parameters defined in [10] are adopted. We consider a STH of 10 orbital periods and $N_{STP} = 10$, so that each orbital period corresponds to one STP, and instances of $\{1000, 1200, 1400, 1600, 1800\}$ targets uniformly distributed across the coverage area of the satellites. Each target must be observed for a duration $\tau_t^{(obs)} = Uniform(1, 5)$ seconds, and a discretization step of $prc = 10$ seconds is assumed.

The performance of the proposed algorithm is compared against a First-In First-Out (FIFO) constructive heuristic [13], which schedules feasible targets in ascending VTW order, capturing them at the first feasible OTW, and a GA initialized with random solutions. Following [12], the crossover and mutation operators, as well as the procedure for generating individuals in the initial population, have been adapted to

Algorithm 1 Constructive Heuristic Algorithm

Input: S, T, G, STH, N_{STP}
Output: Schedule
Initialize Schedule $\leftarrow \varnothing$
Calculate OTWs.
for STP in STH **do**
  Calculate priority indicators δ, FL, and OC.
  Sort targets according to the proposed priority policies.
  for t in T **do**
    Sort OTWs for t and current STP, $OTW_{t,STP}$, in ascending OC order.
    while t is not scheduled **and** $OTW_{t,STP}$ is not empty **do**
      $OTW_{s,t,o,w} \leftarrow OTW_{t,STP}[0]$
      if scheduling $OTW_{s,t,o,w}$ is feasible **then**
        Add $OTW_{s,t,o,w}$ to Schedule.
      else
        Remove $OTW_{s,t,o,w}$ from $OTW_{t,STP}$.
      end if
    end while
  end for
  List unscheduled targets U.
  for u in U **do**
    Sort OTWs for u and current STP, $OTW_{u,STP}$, in ascending OC order accounting only for the conflicting missions in Schedule.
    while u is not scheduled **and** $OTW_{u,STP}$ is not empty **do**
      $OTW_{s,u,o,w} \leftarrow OTW_{u,STP}[0]$.
      Construct feasible solution, Schedule', scheduling $OTW_{s,u,o,w}$ following the proposed insertion and removal policy.
      if Schedule' improves Schedule **then**
        Schedule $\leftarrow$ Schedule'
      else
        Remove $OTW_{s,u,o,w}$ from $OTW_{u,STP}$.
      end if
    end while
  end for
  Update δ_t for every target $t \in T$
end for

ensure that all obtained solutions are feasible. The parameter settings for the GA are as follows:

- Population size: 10.
- Crossover probability: 0.2.
- Mutation probability: 0.2.
- Maximum generations: 100.
- Maximum execution time: 1000 s.

All algorithms are implemented in Python, employing four AMD EPYC 7H12 cores at 2.6 GHz and 16 GB of RAM as computational resources.

Fig. 3 shows the total collected observation profit and the percentage of missed targets at the end of the STH. The proposed constructive heuristic algorithm consistently outperforms both FIFO and GA across all considered instances. Moreover, while FIFO and GA fail to observe the entire set of targets – showing an increasing proportion of missed targets as the problem size grows – the proposed method successfully monitors all targets in each instance.

Furthermore, Fig. 4 presents the boxplot of the GSD of the captured frames, highlighting that the proposed method achieves higher-quality frames compared to the other evaluated approaches, being surpassed only by GA in the 1000-target instance. Specifically, it achieves an average GSD reduction of up to 8% compared to GA, and up to 18.6% with respect to FIFO.

Additionally, Fig. 5 displays the boxplot of the average AoI of the monitored targets by the end of the STH. The proposed method exhibits lower variance, indicating that most targets are observed with similar regularity. Specifically, a variance reduction of up to 84% compared to the other methods is achieved. This trend is also observed in Fig. 6, which presents the boxplot of the average PAoI for the monitored targets. The 99th percentile values range from 7.33 to 9.34 orbital periods for FIFO, and from 9.12 to 9.85 orbital periods for GA, whereas the proposed heuristic achieves lower values, ranging from 4.46 to 8.51 orbital periods. Since FIFO and GA do not account for information freshness, they often schedule recently observed targets, while the heuristic approach prioritizes those that have remained unobserved for longer periods, leading to a more balanced monitoring.

Finally, Table II reports the average execution time per STP for each instance size and algorithm. GA reaches the maximum execution time in all cases due to the complexity of the optimization problem, while still yielding significantly worse results than the proposed approach. FIFO is the fastest algorithm but, as previously discussed, its performance is significantly worse than that of the proposed heuristic method, which maintains a reasonable execution time of just a few tens of seconds even in the most complex cases, thereby offering the best trade-off between performance and efficiency among the evaluated algorithms.

Thus, we can conclude that the proposed algorithm for solving the AEOSSP is able to efficiently provide high-quality and up-to-date information of all the targets in the instance within a multi-satellite, continuous, real-time monitoring scenario.

IX. Conclusions

In this paper, we propose a multi-satellite, continuous, and real-time monitoring scenario to address the AEOSSP, together with a constructive heuristic algorithm further enhanced with a LS strategy. Onboard data processing is explicitly integrated into the optimization problem, and the observation profit is defined to account for both image quality and information freshness, using GSD and AoI as key performance indicators. The obtained results demonstrate that the proposed scheduling method is capable of collecting high-quality frames while ensuring regular monitoring of most targets in the instance. Future work will explore a more

TABLE II: Average execution time per STP for each algorithm.

Number of Targets	Average Execution Time [s]		
	FIFO	GA	Heuristic
1000	2.7243	1000	9.2947
1200	4.9166	1000	13.6843
1400	5.9012	1000	17.6615
1600	6.7584	1000	26.9870
1800	9.4630	1000	41.1620

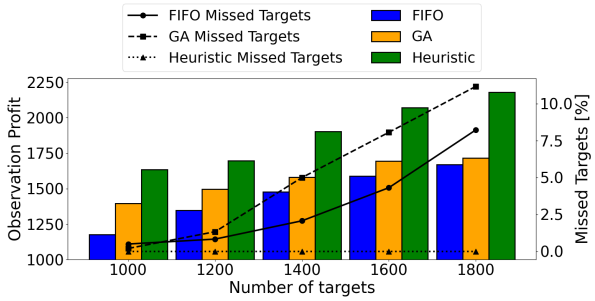

Fig. 3: Total collected observation profit by the end of the STH for all the instances.

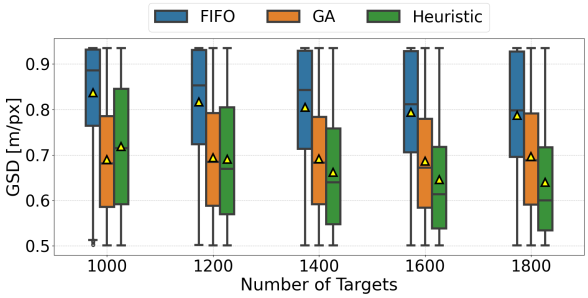

Fig. 4: Boxplot of the GSD of the captured frames by the end of the STH.

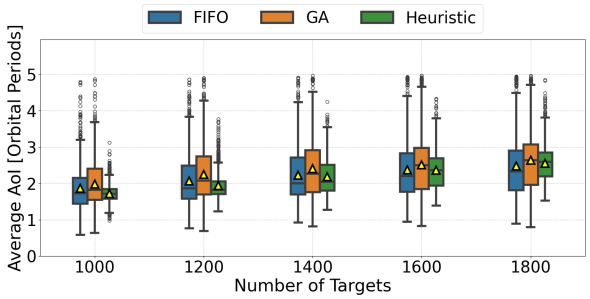

Fig. 5: Boxplot of the average AoI of in the targets in the instance that have been captured by the end of the STH.

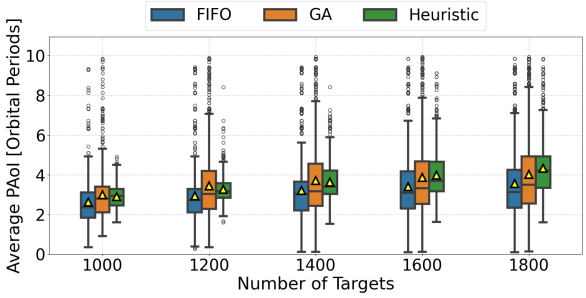

Fig. 6: Boxplot of the average PAoI of the targets in the instance that have been captured by the end of the STH.

dynamic formulation of priority indicators to incorporate sudden changes in mission urgency or weather conditions, as well as their refinement for the development of more advanced solution strategies or for deployment in a distributed scheme. Additionally, more comprehensive evaluations, considering realistic target distributions – such as clustered formations – and real-world EO satellite systems will be conducted.

References

[1] I. Leyva-Mayorga, M. Martinez-Gost, M. Moretti, A. Pérez-Neira, M. A. Vázquez, P. Popovski, and B. Soret, "Satellite edge computing for real-time and very-high resolution earth observation," *IEEE Transactions on Communications*, vol. 71, no. 10, pp. 6180–6194, 2023.

[2] X. Wang, G. Wu, L. Xing, and W. Pedrycz, "Agile Earth Observation Satellite Scheduling Over 20 Years: Formulations, Methods, and Future Directions," *IEEE Systems Journal*, vol. 15, no. 3, p. 3881–3892, Sep. 2021.

[3] W. Yi *et al.*, "Comprehensive Evaluation of the GF-4 Satellite Image Quality from 2015 to 2020," *ISPRS International Journal of Geo-Information*, vol. 10, no. 6, 2021.

[4] H. Chen, S. Peng, C. Du, and J. Li, *Earth observation satellites : task planning and scheduling*, 1st ed. Singapore: Springer Nature Singapore, 2023.

[5] R. Xu, H. Chen, X. Liang, and H. Wang, "Priority-based constructive algorithms for scheduling agile Earth observation satellites with total priority maximization," *Expert Systems with Applications*, vol. 51, pp. 195–206, 2016.

[6] X. Wang, C. Han, R. Zhang, and Y. Gu, "Scheduling multiple agile Earth observation satellites for oversubscribed targets using complex networks theory," *IEEE Access*, vol. 7, pp. 110 605–110 615, 2019.

[7] F. Valente, V. Eramo, and F. G. Lavacca, "Optimal bandwidth and computing resource allocation in low Earth orbit satellite constellation for earth observation applications," *Computer Networks*, vol. 232, p. 109849, 2023.

[8] A. Mercado-Martínez, B. Soret, and A. Jurado-Navas, "Goal-oriented vessel detection with distributed computing in a LEO satellite constellation," in *2024 6th International Conference on Communications, Signal Processing, and their Applications (ICCSPA)*, 2024, pp. 1–5.

[9] A. Kosta, N. Pappas, and V. Angelakis, "Age of information: A new concept, metric, and tool," *Foundations and Trends® in Networking*, vol. 12, no. 3, pp. 162–259, 2017.

[10] I. Leyva-Mayorga *et al.*, *Non-Geostationary Satellite Communications Systems*. Institution of Engineering and Technology, Dec. 2022.

[11] X. Liu, G. Laporte, Y. Chen, and R. He, "An adaptive large neighborhood search metaheuristic for agile satellite scheduling with time-dependent transition time," *Computers & Operations Research*, vol. 86, pp. 41–53, 2017.

[12] X. Chen, G. Reinelt, G. Dai, and M. Wang, "Priority-based and conflict-avoidance heuristics for multi-satellite scheduling," *Applied Soft Computing*, vol. 69, pp. 177–191, 2018.

[13] N. Bianchessi and G. Righini, "Planning and scheduling algorithms for the cosmo-skymed constellation," *Aerospace Science and Technology*, vol. 12, no. 7, pp. 535–544, 2008.