JOURNAL OF AEROSPACE INFORMATION SYSTEMS
Vol. 15, No. 11, November 2018

Check for updates

Optimization-Based Scheduling Method for Agile Earth-Observing Satellite Constellation

Doo-Hyun Cho,* Jun-Hong Kim,† Han-Lim Choi,‡ and Jaemyung Ahn§
Korea Advanced Institute of Science and Technology, Daejeon 34141, Republic of Korea

DOI: 10.2514/1.I010620

This Paper proposes a two-step binary linear programming formulation for task scheduling of a constellation of low-Earth-orbit satellites and demonstrates its applicability and scalability to obtain high-quality solutions using a standard mixed-integer linear programming solver. In this instance, the goal of satellite constellation task scheduling is to allocate each task for the satellites and to determine the task starting times in order to maximize the overall mission performance metric. The scheduling problem is formulated to find the solution by first finding a set of candidate communication time intervals for each satellite/ground-station pair as one of the key constraints and time tabling the observation task to acquire the user-requested data, with the incorporation of key constraints for satellite constellation operation. Numerical experiments are designed for investigating the trends, sensitivity, and characteristics of scheduling outputs based on multiple representative instances. The performance of the scheduling solutions by the proposed two-step binary linear programming method exhibits significant improvement of up to 35% in the number of assignments and the sum of profits over the general greedy algorithm.

Nomenclature

$\mathbb{C}_{j,s}$	=	satellite-task capability indicator
d_{i_s}, e_{i_s}	=	amount of data and energy at end of i_s
$d_s^{\min}$, $e_s^{\min}$, $d_s^{\max}$, $e_s^{\max}$	=	minimum and maximum of data and energy in satellite s
d_s^{start}, e_s^{start}	=	initial amount of data and energy in satellite s
I_s, i_s	=	set of time intervals and its element that belong to satellite s, $i_s \in I_s$
J, S, G	=	set of observation tasks, satellites, and ground stations, respectively
j, s, g	=	indices of J, S, and G, respectively
$\mathbb{P}_{j_1,j_2}$	=	task precedence indicator
$p_{j,s}^O$, $p_{s,g,k}^G$	=	processing time for each satellite s to observe j and download in $\mathrm{TW}_{s,g,k}^G$, respectively
$r_{j,s,k}^O$, $r_{s,g,k}^G$, $r_{s,k}^D$, $r_{s,k}^E$	=	release times of $\mathrm{TW}_{j,s,k}^O$, $\mathrm{TW}_{s,g,k}^G$, $\mathrm{TW}_{s,k}^D$, and $\mathrm{TW}_{s,k}^E$, respectively
$\mathrm{TW}_{j,s}^O$, $\mathrm{TW}_{j,s,k}^O$	=	set of time windows of satellite s related to observation j and its kth element
TW_s^D, $\mathrm{TW}_{s,k}^D$	=	set of time windows of satellite s for downloading data and its kth element
TW_s^E, $\mathrm{TW}_{s,k}^E$	=	set of time windows of satellite s when sunlight is available and its kth element
TW_s^G, $\mathrm{TW}_{s,g,k}^G$	=	set of time windows of satellite s related to ground station (GS) and its kth element
t_{i_s}, t_{i_s+1}	=	start time and end time of interval i_s
t_j^O, $t_{s,g,k}^G$	=	start time of observation j and download in $\mathrm{TW}_{s,g,k}^G$, respectively
$w_{j,s}$	=	weight (or importance) of observation task j conducted by satellite s
w_1, w_2	=	coefficients of download interval and constellation mission scheduling problem, respectively
$x_{j,s,k}^O$, $x_{s,g,k}^G$	=	indicators of assignment on $\mathrm{TW}_{j,s,k}^O$ and $\mathrm{TW}_{s,g,k}^G$, respectively
$\Delta d_{i_s}^O$, $\Delta e_{i_s}^O$	=	amount of data and energy to be changed if observation is completed in i_s
$\Delta e_{i_s}^{\text{sun}}$	=	amount of energy to be increased if sunlight is available in i_s
$\delta_{j,s,k}^O$, $\delta_{s,g,k}^G$, $\delta_{s,k}^D$, $\delta_{s,k}^E$	=	due times of $\mathrm{TW}_{j,s,k}^O$, $\mathrm{TW}_{s,g,k}^G$, $\mathrm{TW}_{s,k}^D$, and $\mathrm{TW}_{s,k}^E$, respectively
$\theta_{g,(s_1,k_1),(s_2,k_2)}^{G2S}$	=	support variable to force ground station g to communicate with only single satellite
$\theta_{j,s,k}^{wj}$, $\theta_{j,s,k}^{jw}$	=	support variable to determine whether observation is assigned in $\mathrm{TW}_{j,s,k}^O$
$\theta_{j_1,j_2,s}^O$	=	support variable to prevent overlap between observations
$\theta_{s,(g_1,k_1),(g_2,k_2)}^{S2G}$	=	support variable to force satellite s to communicate with only single ground station
$\tau_{g,(s_1,k_1),(s_2,k_2)}^{G2S}$	=	setup time of antenna of ground station g from $\mathrm{TW}_{s_1,g,k_1}^G$ to $\mathrm{TW}_{s_2,g,k_2}^G$
$\tau_{s,(g_1,k_1),(g_2,k_2)}^{S2G}$	=	setup time of satellite s from $\mathrm{TW}_{s,g_1,k_1}^G$ to $\mathrm{TW}_{s,g_2,k_2}^G$
$\tau_{s,(j_1,k_1),(j_2,k_2)}^{S2O}$	=	setup time of satellite s from $\mathrm{TW}_{j_1,s,k_1}^O$ to $\mathrm{TW}_{j_2,s,k_2}^O$

Presented as Paper 2016-2107 at the AIAA Guidance, Navigation, and Control Conference, San Diego, CA, 4–8 January 2016; received 18 November 2017; revision received 5 July 2018; accepted for publication 8 July 2018; published online Open Access 28 September 2018.

*Ph.D. Candidate, Department of Aerospace Engineering; dhcho@lics.kaist.ac.kr.
†Ph.D. Candidate, Department of Aerospace Engineering; jun92uk@kaist.ac.kr.
‡Associate Professor, Department of Aerospace Engineering; hanlimc@kaist.ac.kr. Member AIAA.
§Associate Professor, Department of Aerospace Engineering; jaemyung.ahn@kaist.ac.kr.

I. Introduction

AN EARTH observation satellite (EOS) is designed to gather and store data that are to be sent to a set of ground-station networks as it observes specific areas while orbiting the Earth. With advantages such as wide coverage, high observation frequency, and the ability to travel above the border, the EOS has been employed for important science, surveillance, and technology missions across academic, government, and private sectors. Because of demand, the number of satellites has continuously increased. Despite decades of development, typical Earth observation satellite systems are still expensive. A new approach using constellations of small satellites can improve temporal and geographic coverage at a lower unit cost [1–3]. The challenge for these constellations is to develop methods to manage system resources and the capacity to direct large amounts of observation data to the user groups as quickly and reliably as possible.

There are several ongoing projects to design the next generation of Earth observation satellite constellations that will replace the currently operating satellites [4–6]. The next-generation satellites will improve the efficacy of the entire satellite system in terms of system operability, observation quality, capacity, and a delay between a user request and the arrival of observation data. In this Paper, we address a challenge with satellite scheduling (or mission planning) [7]. Although the term *satellite scheduling problem* is applied to different types of operations such as launch control, orbit planning, and lifecycle management, this Paper focuses on mission operations scheduling. Like other scheduling problems, satellite scheduling is a problem of mapping tasks such as observation, communication, and attitude control to satellite system resources [7].

The scheduling problem in satellite communications systems deals with when and how much data to communicate between satellites and a set of ground-station networks. The observation scheduling problem fixes the schedule for all the satellites for when to gather data from specific locations given by user requests. These problems are strongly connected to each other in terms of the resource usage (e.g., time and energy), and their solutions are interdependent. The simplest version of either problem belongs to the NP-hard (non-deterministic polynomial-time hard) class [8,9], and it is extremely difficult to obtain the optimal solution as the size of the problem increases. In addition, there are many characteristics that make scheduling problems more complex, such as visibility time windows, satellite availability for given user requests, and task precedence relationships. Increasing the number of satellites and tasks makes the problem even more difficult.

Because of the difficulties described previously, most studies have concentrated on finding suboptimal solutions with heuristic algorithms [7,10–14], although satellite applications require highly efficient schedule solutions to obtain more data. This Paper develops a two-step binary linear programming (TSBLP) model of the satellite scheduling problem. The suggested model includes key constraints such as data and the energy level, which are considered essential for actual operation. Numerical simulations based on a currently operating satellite system are performed for various instances obtained by changing the key system parameters. We investigate the extent to which the scheduling problem can be solved using the proposed model and an off-the-shelf mixed-integer linear programming (MILP) solver and how efficient the solutions are compared to heuristic solutions. In the scheduling problem, the goal is to maximize the amount of data gathered by a constellation and downloaded to ground stations. This Paper is an extension of our previous research [15], and we used more precise notations and descriptions. Also, the mathematical formulation is extended to cover additional key factors that are critical to the actual operation.

The rest of the Paper is organized as follows. Related studies are briefly surveyed in Sec. II. Section III describes the problem in detail. In Sec. IV, the problem is divided into two separate subproblems and formally stated. Computational results are provided and discussed in Sec. V. Finally, Sec. VI discusses the conclusions of this study.

II. Literature Review

In this Paper, we address the formulation of satellite operation scheduling for a satellite constellation in a continuous domain to obtain the solution. Although much work has focused on the scheduling problem, we are unaware of any studies explicitly studying the problem as formulated in this Paper. Because of the extensiveness of the literature, we first organized papers according to their approaches to solving EOS scheduling problems. We then determined the imperfections of existing studies and described the relevance of our approach.

Numerous studies have been developed on scheduling EOSs, and most of them concentrated on the single-satellite problem. This problem can be generalized or converted to more common problems, such as a single-machine scheduling problem [16,17], a network flow problem [18], the knapsack problem [19–22], dynamic programming [10,23], or constraint programming [7,10,24,25]. Many previous studies use suboptimal algorithms. The most common approach is a greedy algorithm with task priorities [7,10–14]. Cordeau and Laporte [14] described the EOS scheduling problem as a selection of requests to obtain maximal profit for a given orbit, which is called a satellite orbit problem and presented a tabu search algorithm. Lin et al. [26–28] employed the Lagrangian relaxation (LR) technique to integrate their approach with other heuristic approaches. Bianchessi and Righini [29] presented a procedure to guide an existing heuristic algorithm to find a better solution with the linear programming (LP) technique that provides upper bounds. Van den Akker et al. [30] showed a solution for the LP relaxation can be adjusted in polynomial time for certain special cases and proposed a column generation method.

A genetic algorithm has also been adopted by many researchers [12,31–36]. Parish [31] used a genetic algorithm approach to find the best priority ordering of requests and then used a schedule-builder program to build schedules. Globus et al. [33,37] compared several meta-heuristic algorithms and iterated sampling on realistically sized model EOS scheduling problems. Baek et al. [35] compared genetic algorithms and tabu search algorithms to optimize satellite mission schedules. Wolfe and Sorensen [13] used the window constrained packing problem to model the Earth observation scheduling problem.

In the field of operations research, single-agent scheduling with constraints such as the release time, deadline, or precedence condition for optimal solutions has been studied by several researchers. The problem of minimizing the sum of weighted tardiness when release times are given was analyzed and approached by the successive sublimation dynamic programming (SSDP) method [38] and the branch-and-bound (BB) algorithm [39–41]. To solve the precedence constrained scheduling problem while minimizing the sum of weighted completion times, the developed SSDP [42] and the BB algorithm [43] are proposed in a way similar to [38] and [39–41], respectively. The problem of minimizing the sum of the weighted completion time with deadlines or both release times and deadlines was analyzed in [44,45]. Furthermore, the BB algorithm was suggested in [46] to handle the problem minimizing a sum of tardiness that takes the release times, deadlines, and precedence conditions into account. However, consideration of the precedence condition and multiple release times and deadlines pairs for each task simultaneously in the scheduling problem has not yet been handled in the literature even for a single agent.

The number of papers dealing with the multiple-EOSs observation problem has rapidly increased recently [25,47–52]. Similar to the single-EOS case, several heuristic algorithms have been proposed: priority-based algorithms [25,51], rule-based algorithms [47], genetic algorithms [34], and tabu search algorithms [48]. Wu et al. [50] designed an acyclic directed graph model for the multisatellite scheduling problem oriented to emergency jobs and common jobs and proposed a hybrid ant colony optimization algorithm mixed with an iteration local search. Wang et al. [49] proposed a multi-objective mathematical programming model for real-time scheduling of EOSs with a job dynamic merging strategy. Wang et al. [52] suggested a sample approximation method that transforms the chance constraint programming model into an integer linear programming model to obtain scheduling results for multiple nonagile EOSs under the uncertainties of clouds. Wang et al. [25] introduced a nonlinear model of the multiple agile satellites scheduling problem, taking into account both observations and downloads, and proposed a priority-based heuristic and a decision support system.

In a different way, recent studies have tended to expand the existing scheduling problem considering data routing or energy constraints. [2,9,53–58]. Spangelo et al. [55] considered the problem with a single satellite and multiple ground stations and formulated the problem with an mixed integer programming (MIP) model to find the optimal solution. Monmousseau [56] provided both the MIP and simulated annealing approach to deal with observation and communications scheduling for a constellation. Castaing [57] extended the download problem by adding energy constraints with a greedy scheduling heuristic to exploit multiple small satellites, given the capacity-constrained multiple-ground-stations network. Kennedy and Cahoy [2] solved the observation and download scheduling problem for a large-scale satellite constellation, which can crosslink between nearby satellites. They used an algorithm combining heuristic and MILP-based formulation to perform onboard planning in which MILP formulation is restricted to a single satellite [2] or a hierarchical heuristic that divides the problem into onboard and ground-based algorithms to enable scalability to large (dozens to hundreds of satellites) constellations [58].

Generally, many of the formulations and approaches in the reviewed papers dealing with satellite operation scheduling problems ignored important constraints, such as energy and data. To construct the problem realistically, more constraints and practical inputs are needed, including parameters for operating the system of the satellite constellation. Although many studies exist to solve different kinds of scheduling problems, we are unaware of any studies that considered the agile satellite constellation scheduling problem with download and energy constraints in a continuous domain. The following sections describe the details of the problem handled in this Paper.

III. Satellite-Task Scheduling

A. Mission Overview

The overall mission architecture of the satellite constellation system considered for the scheduling problem in this Paper is given in Fig. 1. The mission center collects user requests for Earth observation data/images and determines the overall observation and download plan of a satellite constellation. A user request contains information defining the parameters of the scheduling problem: location and size with the geometric configuration, resolution, priority, and due time of the measured data. Two observation modes can be specified by the user request: single-spot observation mode, for which a single image over a small region is obtained, or wide-area coverage mode, which requires multiple scans of images over a certain designed area. For the wide-area coverage mode, it is assumed that the mission center appropriately divides the region of interest into several strips. A strip represents a single observation where its length and width depend on the duration and the observation mode, respectively. User requests may be weighted differently depending on the profits or scientific values of the request observations; for the request consisting of multiple image strips, it is assumed that each strip is weighted proportionally to the expected image acquisition time, also known as the processing time. Obtained data are sent to the user who requested them upon the download. A network of ground stations communicates with the mission center and satellite constellation. For each ground station, we assume that data are downloaded from a single satellite as it passes over the station.

The Walker-Delta Pattern, symmetric geometry often used for Earth observation missions, is used for orbits of a low-Earth-orbit (LEO) satellite constellation. This type of constellation is defined by four parameters, i: $t/p/f$, where i is the inclination, t is the total number of satellites, p is the number of equally spaced planes, and f is the relative spacing between satellites in adjacent planes [59]. Each satellite is equipped with devices for communication from/to the ground station, electric energy from solar panel and batteries, three-axis attitude control system for agile maneuver, and Earth observation optical instruments.

B. Key Notions in Scheduling

Two types of tasks are involved in the aforementioned mission concept: observation tasks and download tasks. An observation task indicates image acquisition of a certain specified location on Earth. To avoid image distortion, the observation task must be performed when the satellite passes within some allowable pitch and roll angles from the nadir such as Fig. 2. Likewise, a download task is to be performed only when the satellite is within a certain range of the elevation angle relative to the ground station. Each task yields a certain reward (or profit); the goal of scheduling is to maximize the sum of these rewards. In this Paper, tasks are nonpreemptive; in other words, once a satellite begins to execute a certain task, it cannot be interrupted by another task until completion of the first task.

Each task is associated with visibility time windows that consist of the release time and the due time; the task needs to start and finish within this time window. In this Paper, tasks that can be completed strictly within this window are scheduled, since the satellite cannot measure the desired

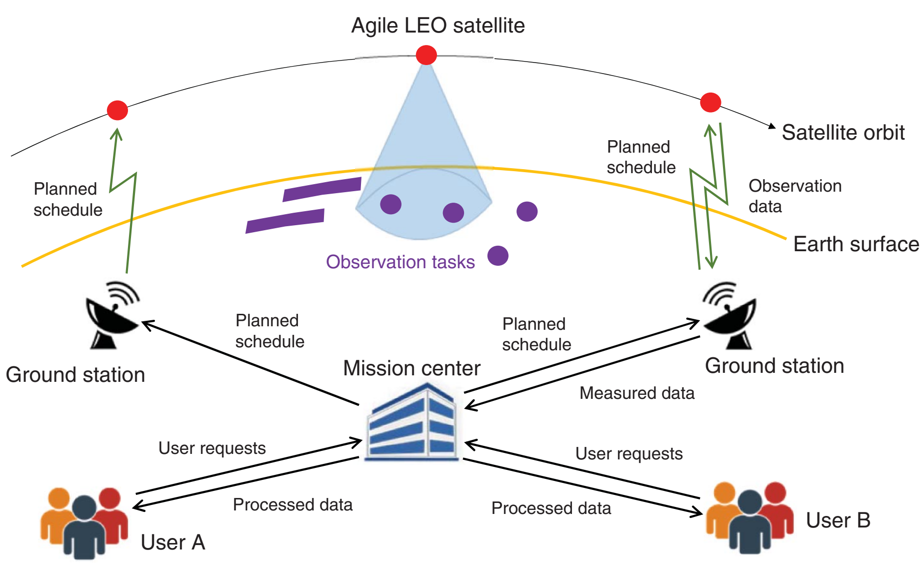

Fig. 1 Overview of the system related to the scheduling problem.

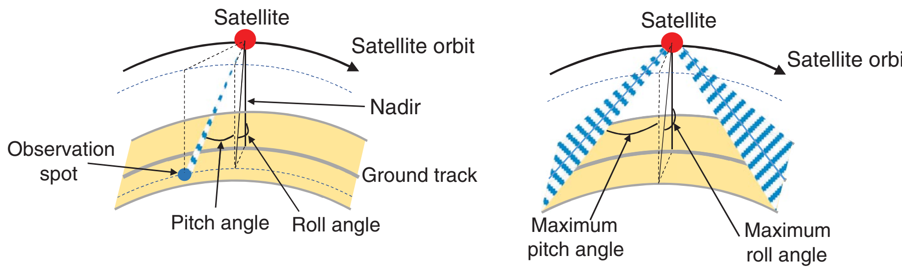

Fig. 2 Illustration of relevant angles (left) and maximum roll and pitch (right).

area or communicate with the ground station if it is out of range. Depending on the type of task, the visibility time window is divided into the observation and download time window. Given the orbits, the maximum rotation angle or the available communication range, and locations of the tasks or ground stations, the time windows for each satellite can be obtained using software such as Satellite Tool Kit (STK). The number of simultaneously visible satellites from the ground station is checked using STK. For a constellation of up to 50 satellites, only a single satellite can be seen from the ground-control center with the orbit altitude set as 500 km and the minimum elevation angle at which the ground station can communicate with the satellite greater than 5 deg.

To perform several tasks sequentially, the satellite must control the attitude of its body or the angle of an antenna toward a set of specific roll and pitch angles. The setup time for a satellite represents the time required for this attitude transition. While the detailed quantification of the setup time involves knowledge of full attitude dynamics of the satellite and the associated controllers, this Paper assumes that the setup time between two observation tasks is represented in terms of the differences in the roll angles. When the roll angles for two consecutive tasks i and j are denoted as ϕ_i and ϕ_j, the setup time between these two asks is represented as $\tau_{ij}^S = c_1|\phi_j - \phi_i| + c_2$ with positive constants c_1 (in seconds/degree) and c_2 (in seconds) depending on the satellite. Similar to the satellite, the ground station needs to steer its antenna pointing from one satellite to the next satellite to perform download tasks; this defines the setup time for the download task. It is primarily dependent on the differences in the roll angles of the antenna on the ground station. Given the roll angles ϕ_i and ϕ_j for two consecutive satellites i and j, the setup time between two download tasks can be represented as $\tau_{ij}^G = c_3|\phi_j - \phi_i| + c_4$ with positive constants c_3 (in seconds/degree) and c_4 (in seconds) depending on the antenna.

In terms of data, the data collected by a satellite from observation tasks are stored in the onboard storage before being downloaded to the ground station. The storage has a certain positive minimum and maximum allowable data limit denoted as $d_{\min}$ and $d_{\max}$, where $d_{\min}$ is assumed to be 0. To ensure the data stored are within these bounds, each satellite needs to keep track of the amount of data stored onboard. The data stored in the next time interval start from the amount of data stored in the current interval, add an amount collected from observation tasks, and subtract an amount discharged by downloading tasks. Likewise, the satellite battery energy is bounded by positive constants $e_{\min}$ and $e_{\max}$. Similar to the data storage level, the satellite's energy at the next interval starts from the amount of energy at the current interval, subtracts energy used for observation tasks and download tasks, and adds energy generated by a solar panel.

IV. Problem Formulation

To reduce the complexity while maintaining the quality of the output from the scheduling problem, the scheduling problem is divided into two subproblems:

1) In download interval scheduling, given satellite orbits and ground-station locations, the data download time intervals between each satellite and ground station are allocated through this subproblem. This allocation result is a constraint in the following subproblem.

2) In constellation mission scheduling, the detailed schedules of the observation and download tasks are obtained through this subproblem. For the observation tasks, each satellite gets a result for when to start and finish a task. For download tasks, each satellite gets a result for how much data to transmit over the time interval obtained from the first subproblem. The scheduling results satisfy the various parameters and constraints.

This separation of the download scheduling and mission scheduling assumes that there exists an appropriate mechanism to determine the ground-control center for each download task of a satellite. This assumption is not restrictive for two practical reasons:

1) For a constellation of tens of satellites in the Walker-Delta pattern, it is not very likely that multiple satellites are in the communication range of a ground station.

2) In case there are multiple ground stations within the communication range of a satellite, a simple heuristic to choose the ground station would not incur significant performance degradation. As such, this Paper focuses on the case in which this assumption is valid; for a larger-scale problem, a tighter coupling between the two subproblems needs to be incorporated.

To solve the TSBLP problem, all related parameters for task scheduling including the satellite orbits are assumed to be known in advance. The big M method is used to add the conditional constraints in which big M refers to a large number associated with the artificial variables, represented by the letter M. In notations, several words are abbreviated for simplicity: O for observation, G for ground station, D for download, and E for energy. The superscript refers to the name or attribute, and the subscript(s) refer(s) to the input variable(s).

A. Mathematical Model: Download Interval Scheduling

A toy problem for the download interval scheduling is described in Figs. 3 and 4. In this example, two satellites are assumed to share the same orbit. Each downloadable region is shown as a transparent blue cone, and the time interval when a satellite sweeps through the ground-control center becomes a visibility time window. The communication range of the satellites for each time in Fig. 3 is depicted in Fig. 4. Each download visibility time window is shared from the relevant satellite and the control station. The download time should be inside the time window, and the whole download interval schedule is obtained by solving the problem of maximizing the objective function while satisfying all of the constraints.

The mathematical formulation of this problem is

$$\max \left[w_1 \sum_{s\in S} \sum_{g\in G} \sum_{k\in\{1,\ldots,|\mathrm{TW}_{s,g}^G|\}} p_{s,g,k}^G + (1 - w_1)Z \right] \tag{1}$$

subject to

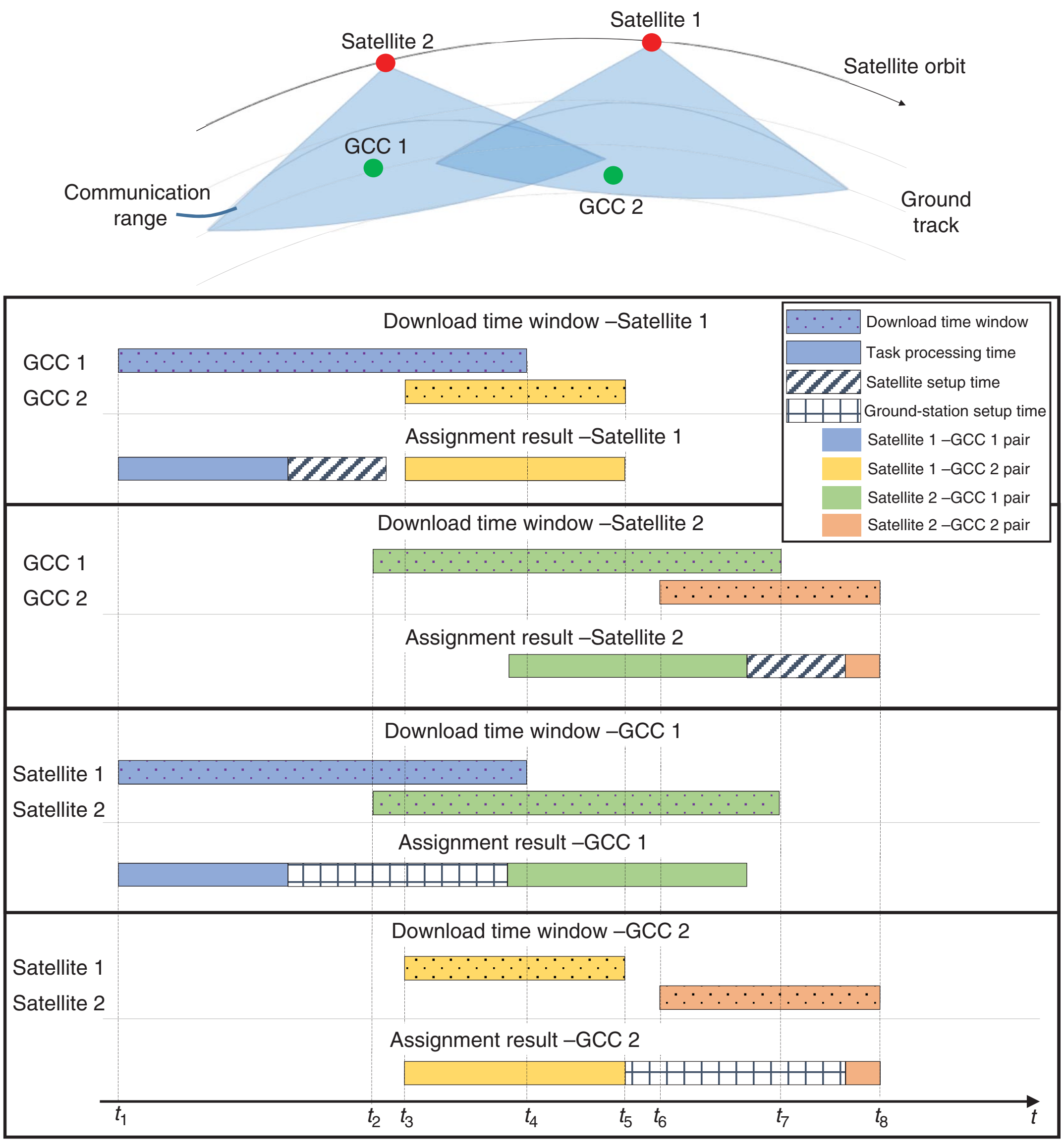

Fig. 3 A toy version of download task scheduling with two satellites and two ground control centers (GCC). The visibility time windows and the assignment results are shown for each satellite and GCC pair, where the assignment results represent when to communicate.

$$Z \leq \sum_{g \in G} \sum_{k \in \{1,\ldots,|\mathrm{TW}^G_{s,g}|\}} p^G_{s,g,k} \quad \forall\ s \in S \tag{2}$$

$$(1 - x^G_{s,g,k})M + p^G_{s,g,k} \leq M \quad \forall\ s \in S, \qquad g \in G, \qquad k \in \{1,\ \ldots,\ |\mathrm{TW}^G_{s,g}|\} \tag{3a}$$

$$0 \leq p^G_{s,g,k} \leq \delta^G_{s,g,k} - t^G_{s,g,k} \quad \forall\ s \in S, \qquad g \in G, \qquad k \in \{1,\ \ldots,\ |\mathrm{TW}^G_{s,g}|\} \tag{3b}$$

$$(1 - x^G_{s,g,k})M + t^G_{s,g,k} \leq M \quad \forall\ s \in S, \qquad g \in G, \qquad k \in \{1,\ \ldots,\ |\mathrm{TW}^G_{s,g}|\} \tag{4a}$$

$$x^G_{s,g,k}M + (r^G_{s,g,k} - t^G_{s,g,k}) \leq M \quad \forall\ s \in S, \qquad g \in G, \qquad k \in \{1,\ \ldots,\ |\mathrm{TW}^G_{s,g}|\} \tag{4b}$$

$$x^G_{s,g,k}M + (t^G_{s,g,k} - \delta^G_{s,g,k}) \leq M \quad \forall\ s \in S, \qquad g \in G, \qquad k \in \{1,\ \ldots,\ |\mathrm{TW}^G_{s,g}|\} \tag{4c}$$

$$\theta^{S2G}_{s,(g_1,k_1),(g_2,k_2)}M + (t^G_{s,g_1,k_1} + p^G_{s,g_1,k_1} + \tau^{S2G}_{s,(g_1,k_1),(g_2,k_2)} - t^G_{s,g_2,k_2}) \leq M \;\; \forall\, s \in S, \quad g_1, g_2 \in G, \quad g_1 \neq g_2,\;\; k_1 \in \{1,\ldots,|\mathrm{TW}^G_{s,g_1}|\}, k_2 \in \{1,\ldots,|\mathrm{TW}^G_{s,g_2}|\} \tag{5}$$

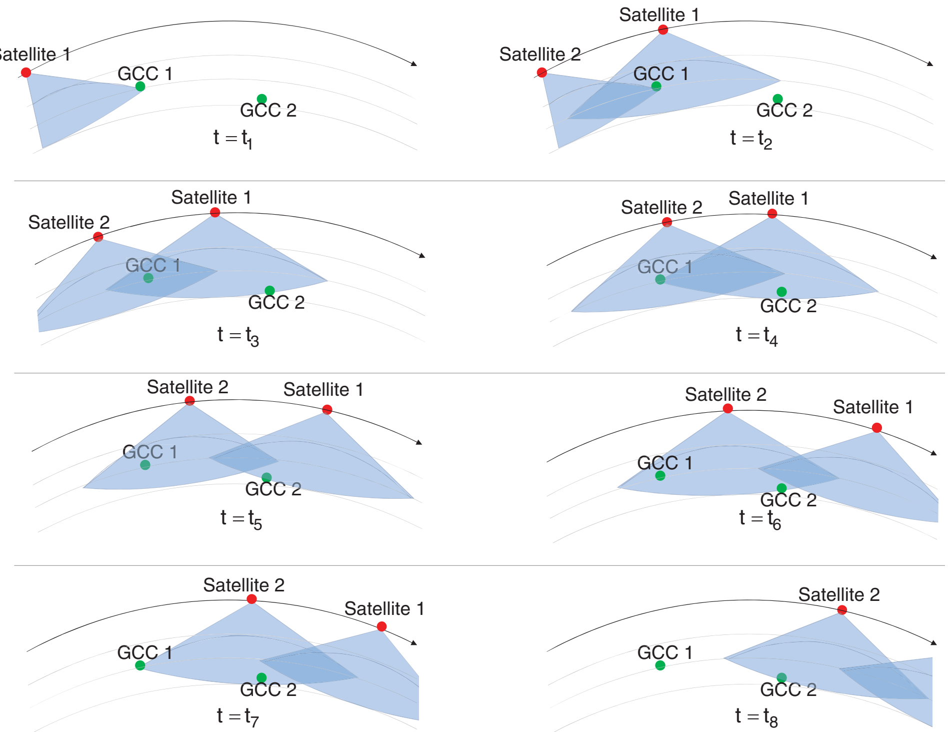

Fig. 4 Download task scheduling, a detailed description for each time step in the situation of Fig. 3; sat 1/GCC 1 start meeting at t_1, sat 2/GCC 1 start meeting at t_2, sat 1/GCC 2 start meeting at t_3, sat 1/GCC 1 end meeting at t_4, sat 1/GCC 2 end meeting at t_5, sat 2/GCC 2 start meeting at t_6, sat 2/GCC 1 end meeting at t_7, and sat 2/GCC 2 end meeting at t_8.

$$\theta^{S2G}_{s,(g_1,k_1),(g_2,k_2)} + \theta^{S2G}_{s,(g_2,k_2),(g_1,k_1)} = 1 \quad \forall\, s \in S, \quad g_1, g_2 \in G, \quad g_1 \neq g_2, \quad k_1 \in \{1, \ldots, |\mathrm{TW}^G_{s,g_1}|\}, \quad k_2 \in \{1, \ldots, |\mathrm{TW}^G_{s,g_2}|\} \tag{6}$$

$$\theta^{G2S}_{g,(s_1,k_1),(s_2,k_2)} M + (t^G_{s_1,g,k_1} + p^G_{s_1,g,k_1} + \tau^{G2S}_{g,(s_1,k_1),(s_2,k_2)} - t^G_{s_2,g,k_2}) \leq M \;\; \forall\, s_1, s_2 \in S, \;\; s_1 \neq s_2, \;\; g \in G, \;\; k_1 \in \{1, \ldots, |\mathrm{TW}^G_{s_1,g}|\}, \;\; k_2 \in \{1, \ldots, |\mathrm{TW}^G_{s_2,g}|\} \tag{7}$$

$$\theta^{G2S}_{g,(s_1,k_1),(s_2,k_2)} + \theta^{G2S}_{g,(s_2,k_2),(s_1,k_1)} = 1 \quad \forall\, s_1, s_2 \in S, \quad s_1 \neq s_2, \quad g \in G, \quad k_1 \in \{1, \ldots, |\mathrm{TW}^G_{s_1,g}|\}, \quad k_2 \in \{1, \ldots, |\mathrm{TW}^G_{s_2,g}|\} \tag{8}$$

In this formulation, the objective function is a sum of all download time in the scenario and a minimum of satellites' total download processing time with appropriate weight values. A variable Z in the constraints (2) is a lower bound among the total download time of each satellite. Constraints (3) and (4) represent the lower and upper bounds of the start and download processing times for each time window if the corresponding binary decision variable $x^G_{s,g,k}$ equals 1. Briefly, $0 \leq p^G_{s,g,k} \leq \delta^G_{s,g,k} - t^G_{s,g,k}$ and $r^G_{s,g,k} \leq t^G_{s,g,k} \leq \delta^G_{s,g,k}$ if $x^G_{s,g,k} = 1$; otherwise, $p^G_{s,g,k}$ and $t^G_{s,g,k}$ become 0. Constraints (5) ensure that each satellite cannot perform multiple downloads at the same time. For a satellite s, $\theta^{S2G}_{s,(g_1,k_1),(g_2,k_2)}$ is 1 if a download in time window $\mathrm{TW}^G_{s,g_2,k_2}$ is scheduled to start after completing a download in $\mathrm{TW}^G_{s,g_1,k_1}$ with a setup time for the preparation of starting in $\mathrm{TW}^G_{s,g_2,k_2}$ and 0 otherwise. In short, $t^G_{s,g_1,k_1} + p^G_{s,g_1,k_1} + \tau^{S2G}_{(s,g_1,k_1),(g_2,k_2)} \leq t^G_{s,g_2,k_2}$ if $\theta^{S2G}_{s,(g_1,k_1),(g_2,k_2)} = 1$. Because only one of the two schedules can precede the other, constraints (6) enforce this restriction. Similar to the situation for satellites, constraints (7) and (8) ensure that each ground station cannot perform multiple downloads at the same time. For a ground station g, $\theta^{G2S}_{g,(s_1,k_1),(s_2,k_2)}$ is 1 if a download in time window TW^G_{s2,g,k_2} is scheduled to start after completing a download in TW^G_{s1,g,k_1} with a setup time for the preparation of starting in TW^G_{s2,g,k_2} and 0 otherwise. In other words, $t^G_{s_1,g,k_1} + p^G_{s_1,g,k_1} + \tau^{G2S}_{g,(s_1,k_1),(s_2,k_2)} \leq t^G_{s_2,g,k_2}$ if $\theta^{G2S}_{g,(s_1,k_1),(s_2,k_2)} = 1$.

The key feature of the download-task scheduling model is that processing a task in time window $\mathrm{TW}^G_{s,g,k}$ requires a ground station g and a satellite s to work simultaneously.

B. Mathematical Model: Constellation Mission Scheduling

Before describing the entire mathematical model of the constellation mission scheduling problem, we illustrate an observation scheduling model from [15], which excludes download and energy constraints.

1. Observation Task Scheduling: Unrestricted Data and Energy Constraints from Constellation Mission Scheduling

A toy version of the observation-task scheduling problem is shown in Fig. 5. Given a satellite orbit and locations to be observed, time windows for each observation task can be calculated with the maximum measurable angle. Each task has a different strip length and strip width, which is given from a user request. A resulting observation schedule is given at the bottom of the figure.

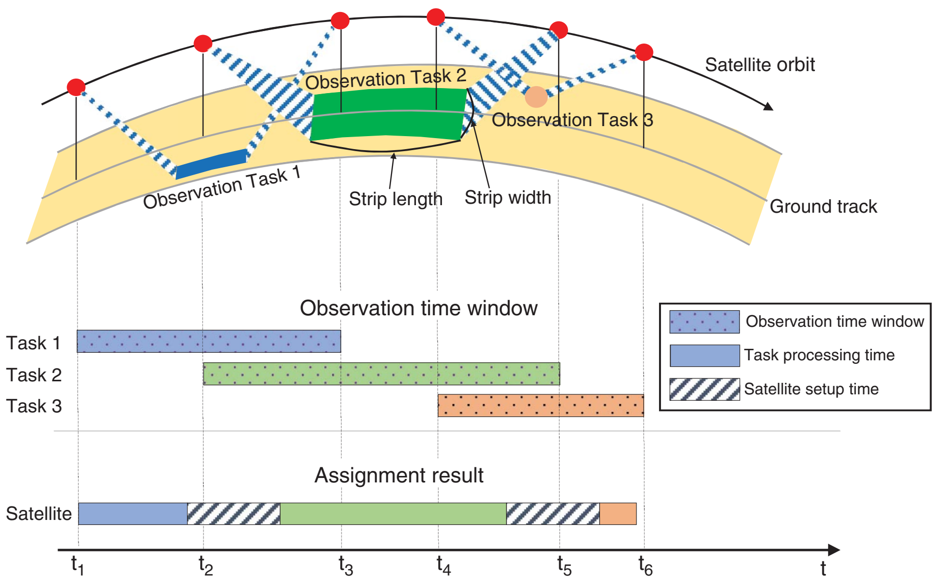

Fig. 5 A simple example of an observation scheduling problem.

The observation task scheduling model is formulated as

$$\max \sum_{j\in J}\sum_{s\in S}\sum_{k\in\{1,\ldots,|\mathrm{TW}_{j,s}^{O}|\}} w_j^O x_{j,s,k}^O \tag{9}$$

subject to

$$\mathbb{P}_{j_1,j_2}M + \left(t_{j_1}^O + \sum_{k_1\in\{1,\ldots,|\mathrm{TW}_{j_1,s}|\}} x_{j_1,s,k_1}\cdot p_{j_1,s}^O - t_{j_2}^O\right) \le M \quad \forall\ j_1,j_2\in J,\qquad j_1\ne j_2,\qquad s\in S \tag{10}$$

$$(1-\mathbb{C}_{j,s})M + x_{j,s,k} \le M \quad \forall\ j\in J,\qquad s\in S,\qquad k\in\{1,\ldots,|\mathrm{TW}_{j,s}^O|\} \tag{11}$$

$$\theta_{j_1,j_2,s}^O M + \left(t_{j_1}^O + \left(\sum_{k_1\in\{1,\ldots,|\mathrm{TW}_{j_1,s}|\}}\sum_{k_2\in\{1,\ldots,|\mathrm{TW}_{j_2,s}|\}} x_{j_1,s,k_1}\cdot\left(p_{j_1,s}^O + \tau_{s,(j_1,k_1),(j_2,k_2)}^{S2O}\right)\right) - t_{j_2}^O\right) \le M \ \forall j_1,j_2\in J,\quad j_1\ne j_2,\quad s\in S \tag{12}$$

$$\theta_{j_1,j_2,s}^O + \theta_{j_2,j_1,s}^O = 1 \quad \forall\ j_1,j_2\in J,\qquad j_1\ne j_2,\qquad s\in S \tag{13}$$

$$\theta_{j,s,k}^{wj}M + (r_{j,s,k}^O - t_j^O) \le M \quad \forall\ j\in J,\qquad s\in S,\qquad k\in\{1,\ldots,|\mathrm{TW}_{j,s}^O|\} \tag{14}$$

$$\theta_{j,s,k}^{jw}M + (t_j^O + p_{j,s}^O\cdot x_{j,s,k}^O - \delta_{j,s,k}^O) \le M \quad \forall\ j\in J,\qquad s\in S,\qquad k\in\{1,\ldots,|\mathrm{TW}_{j,s}^O|\} \tag{15}$$

$$x_{j,s,k}^O M + (\theta_{j,s,k}^{wj} + \theta_{j,s,k}^{jw} - 2) \le M \quad \forall\ j\in J,\qquad s\in S,\qquad k\in\{1,\ldots,|\mathrm{TW}_{j,s}^O|\} \tag{16a}$$

$$M + (\theta_{j,s,k}^{wj} + \theta_{j,s,k}^{jw} - 2) \ge x_{j,s,k}^O M \quad \forall\ j\in J,\qquad s\in S,\qquad k\in\{1,\ldots,|\mathrm{TW}_{j,s}^O|\} \tag{16b}$$

The objective is to maximize the sum of profits from observations that are completed on time. The profit w_j^O is multiplied by $x_{j,s,k}^O$ to distinguish the importance of each observation task. Constraints (10) state that task j_2 should be assigned after task j_1 is finished if the precedence indicator $\mathbb{P}_{j_1,j_2}$ is 1. Constraints (11) state that task j cannot be assigned to satellite s if the capability indicator designates a 0 value. Constraints (12) ensure that each satellite performs only one observation task at a time. For a satellite s, $\theta_{j_1,j_2,s}^O$ is 1 if observation task j_2 is scheduled to start after completing j_1 with a setup time for the preparation of starting j_2 and 0 otherwise. Briefly, $t_{j_1}^O + \sum_{k_1\in\{1,\ldots,|\mathrm{TW}_{j_1,s}|\}}\sum_{k_2\in\{1,\ldots,|\mathrm{TW}_{j_2,s}|\}} x_{j_1,s,k_1}\cdot \left(t_{j_1,s}^{\mathrm{proc}} + \tau_{s,(j_1,k_1),(j_2,k_2)}^{S2O}\right) \le t_{j_2}^{\mathrm{start}}$ if $\theta_{j_1,j_2,s}^O$ is 1. Constraints (13) enforce the restriction that only one of the two schedules can precede the other. Constraints (14) represent the lower bound of a start time for each observation task. If $\theta_{j,s,k}^{wj}$ is 1, the task is scheduled to start after the start time of a time window and 0 otherwise. Constraints (15) represent the upper bound of a completion time for each observation task. If $\theta_{j,s,k}^{jw}$ is 1, the task is scheduled to be completed before the end time of a time window and 0 otherwise. Constraints (16) ensure that if a decision variable $x^O_{j,s,k}$ equals 1 both support variables $\theta^{wj}_{j,s,k}$ and $\theta^{jw}_{j,s,k}$ should be true. Figure 5 shows a simple example of a situation with three observation tasks having different processing times for the model described previously.

2. Constellation Mission Scheduling

The time intervals i_s are defined and divided based on the end time of time windows $\mathrm{TW}^O_{j,s}$, TW^D_s, and TW^E_s. For simplicity, every $i_s \in I_s$ is sorted in a time-ascending order, i.e., $t_{i_s} \le t_{i_s+1}$, $\forall i_s$, $s \in S$. There is a one-to-one correspondence between the end time of i_s and one of the due times $\delta^O_{j,s,k}$, $\delta^D_{s,k}$, or $\delta^E_{s,k}$. The constellation mission scheduling can be structured based on the observation scheduling model and adding the constraints (10–16),

$$\max\left[w_2\sum_{j\in J}\sum_{s\in S}\sum_{k\in\{1,\ldots,|\mathrm{TW}^O_{j,s}|\}} w^O_j x^O_{j,s,k} + (1-w_2)\sum_{s\in S}\sum_{i_s\in I_s}\Delta d^D_{i_s}\right] \tag{17}$$

subject to

$$\text{Equations (10)} - \text{(16)}$$

$$d_{0_s} = d^{\mathrm{start}}_s \quad \forall\, s \in S \tag{18}$$

$$d^{\min}_s \le d_{i_s} \le d^{\max}_s \quad \forall\, s \in S, \qquad i_s \in I_s \tag{19}$$

$$d_{i_s+1} = d_{i_s} + \Delta d^O_{i_s}\cdot x^O_{j,s,k} - \Delta d^D_{i_s} \quad \forall\, s \in S, \quad i_s \in I_s \tag{20}$$

$$e_{0_s} = e^{\mathrm{start}}_s \quad \forall\, s \in S \tag{21}$$

$$e^{\min}_s \le e_{i_s} \le e^{\max}_s \quad \forall\, s \in S, \qquad i_s \in I_s \tag{22}$$

$$e_{i_s+1} = \min(e^{\max}_s, e_{i_s} - \Delta e^O_{i_s}\cdot x^O_{j,s,k} - \Delta e^D_{i_s} + \Delta e^{\mathrm{sun}}_{i_s}) \quad \forall\, s \in S, \qquad i_s \in I_s \tag{23}$$

In this model, the constraints (18–23) are added to the observation task scheduling model. Constraints (18) initialize the amount of stored data. Constraints (19) show that the amount of data stored at the beginning of each interval is inside the lower and upper limits. Constraints (20) represent the amount of data stored at the beginning of each interval equal to the amount stored at the start of the preceding interval plus the amount acquired by scheduled observation tasks minus the amount downloaded to the ground stations. Constraints (21) initialize the amount of stored energy. Constraints (22) show that the amount of energy stored at the beginning of each interval is inside the lower and upper limits. Constraints (23) represent the amount of energy stored at the beginning of each interval equal to the amount stored at the start of the preceding interval minus the amount consumed for observing and downloading operations plus the amount acquired.

The energy level of a feasible solution from the previous model can be thought to be negative if a download or observation task and energy generation via a solar panel together lead to a sharply nonlinear change of energy value in a certain time interval. However, all of the following conditions must be satisfied in order for this phenomenon to occur:

1) The energy level of the battery should be close to zero.
2) The energy consumption per unit time should be higher than the amount generated by the solar panel.
3) The energy generated must change rapidly.
4) The processing time must be shorter than the given time interval.

In the simulation, most time intervals used to set the data and energy constraints are set to be similar to or shorter than the processing time, which can be an answer for condition 4. As for condition 3, it is known that the power-generation level of the solar panel increases linearly over several tens of seconds [60,61]. Although not shown in Sec. V, we verified for all the scheduling outputs that the energy level remained nonnegative.

The objective is to maximize the sum of observation profits assigned in the schedule and the total amount of download data, which was excluded in Eq. (9). By adjusting the weight w_2, it is possible to decide which term is to be more emphasized[¶] [1].

V. Numerical Experiments

In this section, the computational results on multiple test instances of the satellite scheduling problem described in Sec. IV are reported. The main purposes are to check the effect on scheduling outputs varying of various input parameters and the scalability of the optimization. Each instance is solved using an Intel Core i7-6700K 4.0 GHz processor with 16 GB of memory using the Gurobi optimizer. Figure 6 shows the overall organization of the simulation procedure.

[¶]No matter which values were assigned in the weights, we confirmed that an appropriate scheduling solution was obtained even for the extreme case in which one of the values was 0. This is due to the data storage constraint for the satellites. To perform as many observation tasks as possible, the data space must be secured through the download. In the same manner, the download task can be performed if the data are accumulated by performing the observation tasks.

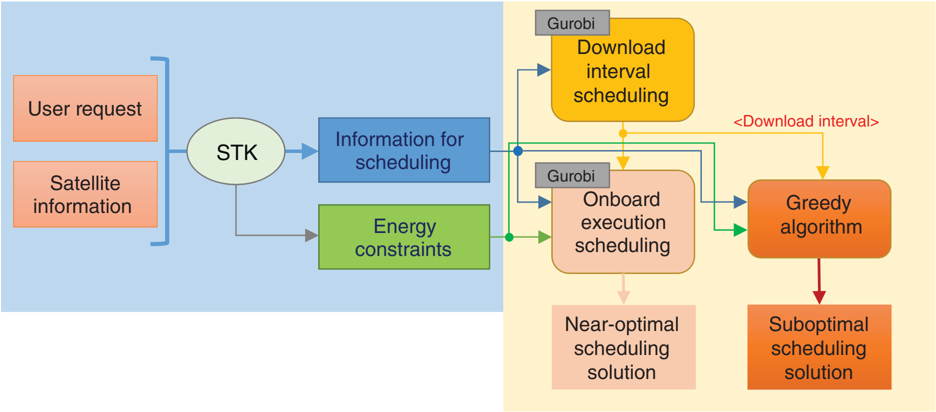

Fig. 6 Overview of the satellite-task scheduling.

The first subsection describes the choice of parameters for the simulation in detail. The second subsection shows the models for performance analysis and comparison of the proposed algorithm. The remaining subsections discuss the computational results varying the parameters in Table 1 to check the effect on outputs.

A. Scenarios

The parameters used for computational experiments are primarily based on a preexisting satellite mission observing the surface of the Earth. The parameters for orbit, data, and energy constraints are largely referred from the satellites called Terrasar-X and KOMMPSAT-3A, which successfully finished or is still conducting the Earth-observing mission in a low-Earth-orbit area. The units of data and energy are simplified so that they are not limited to specific platforms. The number of satellites is set as one or three, and the Walker-Delta Pattern is used to construct a constellation. A network of one to four ground stations receives data from satellites during the simulation. Table 2 lists the global positions, mean percentage of the visible time from satellites, and mean length of visibility time windows for each station.

In the simulation, the number of the observation tasks ranges from 10 to 700 for a single satellite and 10 to 500 for three satellites. The importance weights for each task are set as the same or vary randomly with a uniform distribution. Task locations are randomly chosen from the set of cities that have a population of at least 15,000[**] [2]. The visibility time windows are generated for each satellite based on the orbit and sensor coverage, and the number of time windows varies from 3 to 3258 depending on the number of satellites, length of a planning horizon, and number of the observation tasks. To check the effect of the length of a planning horizon, five planning horizons are used, ranging from a day to a week (1, 2, 3, 5, and 7 days). Each planning horizon starts from 1 April 2017 0:00:00. Details are given in Table 1.

B. Models for Comparison

We compare the results against two different algorithms. The first eliminates data and energy constraints providing an upper bound on the objective function value, which is shown in Sec. IV.B.1. The second uses a greedy algorithm based on the first-in/first-out (FIFO) strategy, which assigns as many download and observation tasks as possible during every opportunity. The second algorithm is partly based on the method previously developed by De Florio [62], and the details are implemented to obey every constraint in Sec. IV. The sketch of the algorithm is provided in Appendix A, Algorithm 1.

The objective value differences between the proposed model, the unrestricted model, and the greedy scheduling algorithm are explored. In general, the proposed model demonstrates a huge improvement compared to the greedy algorithm when the problem instance becomes complicated. The complicated instance refers to the case in which many overlaps for download or an observation visibility time window exist due to a large number of ground-control centers or observation tasks, or the data and energy constraints of the satellites are strong and it is difficult to handle many tasks in a short time.

Before exploring the results under a variety of different instances, the relationship between the calculation time and solution quality is briefly described. Because the download interval scheduling problem took only a few seconds even on the largest instance, the computation time will be addressed only in the constellation mission scheduling problem. We also excluded the computation time of the FIFO algorithm because the largest instance was solved in under 1 s.

C. Computation Time

The trend of the objective value and the upper bound vs the computation time is shown as a log scale in Fig. 7. Here, the upper bound is a value obtained from the linear programming problem, which is a relaxed version of the original TSBLP problem. For the simplicity of the plot, 12 randomly generated instances in terms of the number of observation tasks, satellites, and ground stations are described and calculated with the computation time limit to be 10,000 s. The upper bound and best solution value over time can be obtained from the solver's branch and bound algorithm for solving a problem instance. Based on the 100% line (red dashed line on the middle of the plots), the upper lines represent the upper bound of each instance over time, and the lower lines show the best-known solution values for each instance over time. In the bottom plot of Fig. 7, the thick vertical lines crossing the red horizontal 100% line indicate that the calculation of the instance ends at the time at which it is located. In the beginning, the objective values rapidly approach the optimal value until t is 10^2. After that, only the upper bounds from LP relaxation are tightened as the time increases and the best-known solution values have little change. We found that most instances had a similar tendency on a change of the upper bounds and solution values. Based on this fact, the maximum computation time of the simulations performed in this study is set to 10,000 s with an optimality gap of 0.1%.

Among the parameters in Table 1, the number of observation tasks has the greatest influence on computation time. Results of the computation time and optimality gap for instances with random numbers of observation tasks satellites are shown in Fig. 8 to emphasize the trends. The ranges of each parameter are set from 100 to 1000 and 3 to 12, respectively, for each instance, and each was randomly generated ten times and averaged for

**Data available online at http://download.geonames.org/export/dump/ [retrieved 1 April 2017].

Table 1 Parameters for simulation (bold values are for the representative instance)

Satellite parameters	Value
Altitude, km	**500**
Walker-Delta Pattern notation	97.3°: {**1**,2,3,4}/{1,2,3}/1
Planning horizon, days	{**1**, **2**,3,5,7}
Number of observations	{100,200,300,500,700}
Observation-task processing time, s	**30**
Weight of observation task	$P(20) = P(40) = 0.5$
Min., max. roll angles, deg	−30, **30**
Min., max. pitch angles, deg	−30, **30**
$d_s^{\min}$, $d_s^{\max}$, unit data	**0**, **500**
d_s^{start}, unit data	**0**
Data gain rate (observation task), data/s	**1**
Data transfer rate (download task), data/s	**1**
$e_s^{\min}$, $e_s^{\max}$, unit energy	**0**, **500**
e_s^{start}, unit energy	**0**
Energy gain rate (under sunlight), energy/s	**0.1**
Energy consumption rate (observation task), energy/s	**1**
Energy consumption rate (download task), energy/s	**0.1**

Table 2 Locations of ground stations

Station	Latitude, °	Longitude, °	Visibility, %	Mean duration, s
Yoshida, Kasuga cho	37.68	138.88	2.131	439.6
Tha Muang	13.96	99.64	1.580	421.6
Bassar	9.25	0.78	1.556	434.4
Baturaja	−4.13	104.17	1.501	432.3

the stability. A clear trend depending on the number of tasks and satellites is shown in the upper plot of Fig. 8. Through both plots, it can be seen that the optimal solution is found within a reasonable time until the number of tasks is approximately 400, and up to 500 tasks, the optimal solution is found within the time limit up to three satellites. When the number of tasks is more than 500, the calculation is completed without finding the optimal solution in most cases.

D. Number of Observation Tasks, Ground Stations, and Satellites

The results for different numbers of observation tasks from 100 to 700 and ground stations from 1 to 4 are shown in Fig. 9 with fixing the number of satellites as 1 and the length of the planning horizon at 2 days.

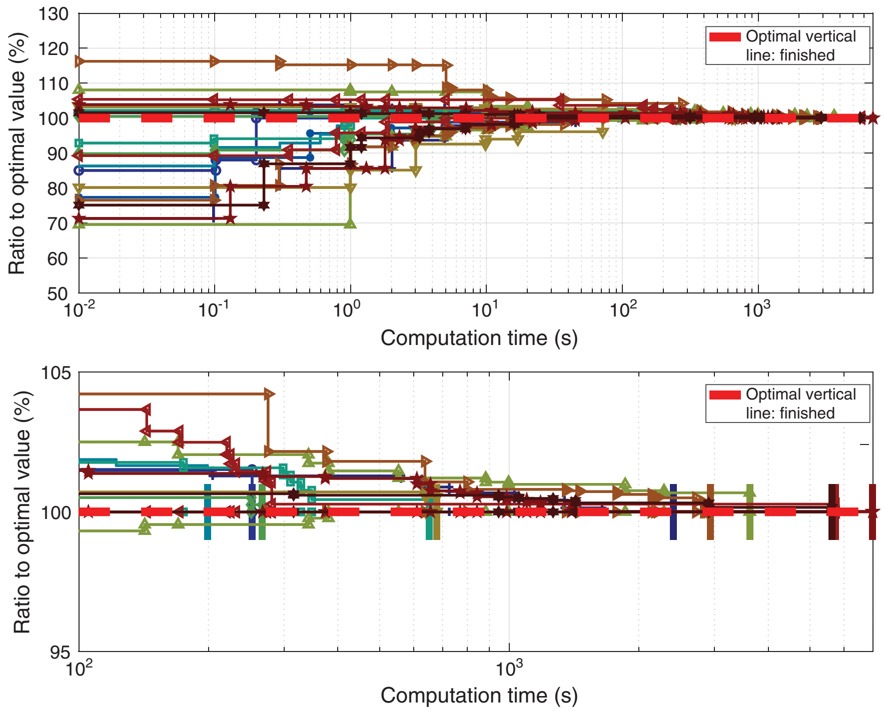

Fig. 7 Trend of upper bounds and best solution values over time. The plot below is an enlargement of the one above.

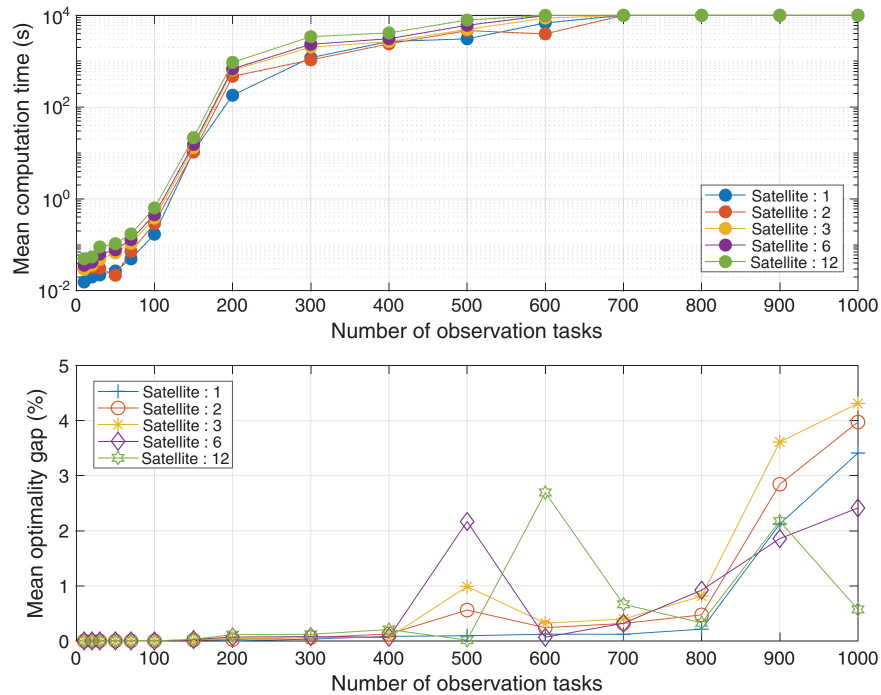

Fig. 8 Average performance of instances under various numbers of observation tasks and satellites. The upper plot shows the mean computation time, and the plot below shows the mean optimality gap.

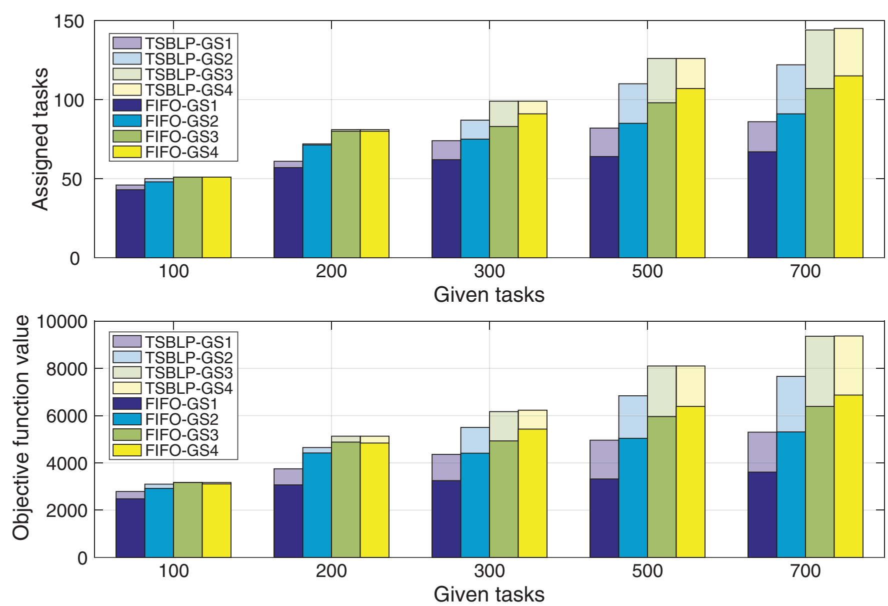

Fig. 9 Comparison of the results on the number of observation tasks; the planning horizon is 2 days.

Figure 9 reveals that the number of observation tasks assigned to satellite scheduling increased as the numbers of given tasks and ground stations increased. The values increased as the number of ground stations increased because the opportunity to secure data storage space increased through data transmission. Little difference was found between the cases of three and four ground stations in each instance because the position of the fourth ground station was intentionally positioned near to the second station, and only a few observation tasks existed between them. But what is important here is the gap between the results of the proposed method and the greedy algorithm. There was little difference between the results of the two methods until the number of the given tasks was 200, but the difference increased gradually up to 30% for the number of assigned tasks and 35% for the objective value. This occurred because, from the moment when the number of given observation tasks increased, there were many tasks in which the positions were close to each other and the time window overlapped. To check how much time windows overlap, we use the following metrics:

$$\text{mean order of overlap} = \frac{\text{sum of time lengths of the entire time windows}}{\text{sum of time lengths in which atleast one time window exists}} \tag{24}$$

$$\text{standard of overlap} = \text{standard deviation of overlapping time windows for each second} \tag{25}$$

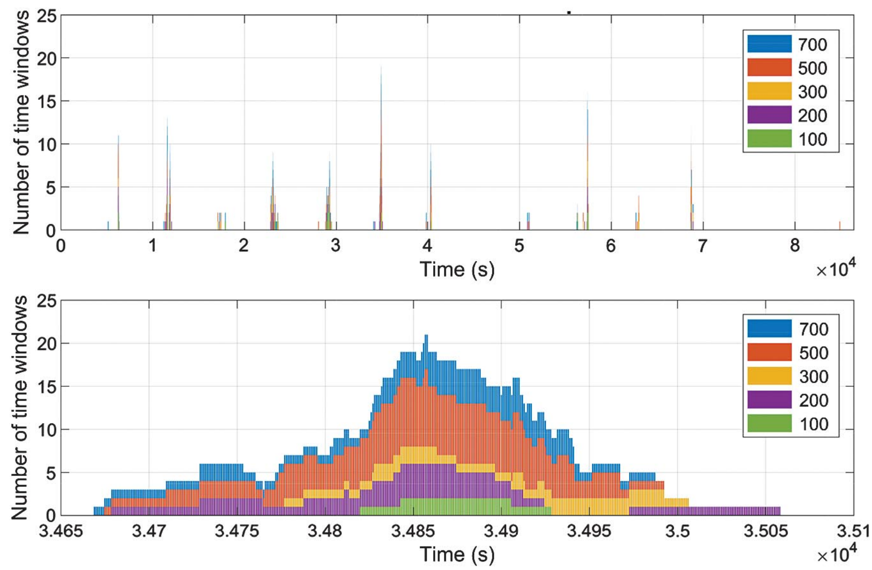

Fig. 10 Numbers of time windows over time with different numbers of observation tasks for a single satellite; the planning horizon is 1 day.

Using the given metric, if the time windows do not overlap, then the mean order of overlap is 1. On the other hand, if there are ten time windows with identical time periods, the order becomes 10. According to the given number of observation tasks, the mean order of overlap and the standard of overlap are approximately as follows (number-mean, std): (100-1.53, 0.23), (200-1.93, 0.44), (300-2.26, 0.58), (500-2.94, 0.91), (700-3.44, 1.18). In general, however, the given values are not representative because the tasks are heavily crowded on a specific part of the Earth, so a time window does not exist on most of the planning horizon (the upper plot of Fig. 10). The bottom plot in Fig. 10 shows the number of overlapping time windows for different given numbers of observation tasks as the satellite passed through the areas where tasks were most closely located.

An extreme instance causing the different results from the greedy algorithm and the TSBLP method is shown in Fig. 11. The setup time of each satellite was the main cause of the difference in objective values. For the sake of clarity, the pitch angle of the satellite was unchanged, and the initial pitch and roll angle were assumed to be 0 for the satellite to point to the nadir. There were three tasks on the left side and six tasks on the opposite side based on the direction of the satellite, and the weights of all tasks were the same. In the case of the greedy algorithm, task 1, which was the first to be encountered by a satellite, was included in the schedule at first, and the satellite controlled its attitude to the left side. After task 1, task 4 with a small setup time was scheduled instead of task 3, which was inexecutable because of the huge roll angle difference. Likewise, after task 4, task 7 was scheduled instead of task 6. On the contrary, in the case of the TSBLP method, the satellite turned to the right to schedule more tasks than the greedy algorithm.

Although it is not shown in Fig. 9, we confirmed that the complexity of the problem instance increased and the performance of the greedy algorithm is degraded as the overlaps increased as opposed to when only the number of tasks increased. To increase the number of tasks to be assigned or the objective function in the same length of the planning horizon, assigning the weight value used in Eq. (9) uniformly for each task is needed. It is confirmed that the performance of the result was as poor as the result of the greedy algorithm when the values were biased to specific tasks.

The results for different numbers of satellites with fixing the number of observation tasks at 700, the number of ground stations at 4, and the planning horizon at 1 day is shown in Fig. 12. The result values were increased as the number of satellites increased, and the average gap between each method was around 30%.

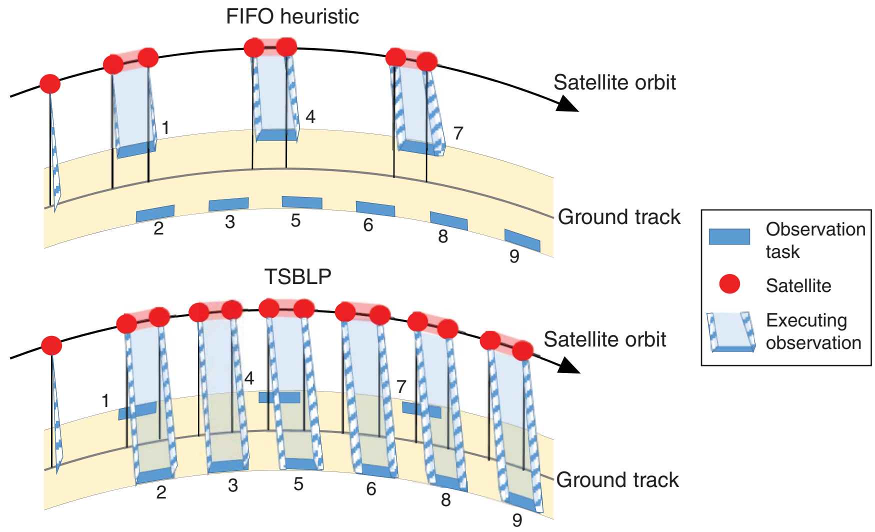

Fig. 11 An instance incurring different results from the greedy algorithm and the TSBLP method.

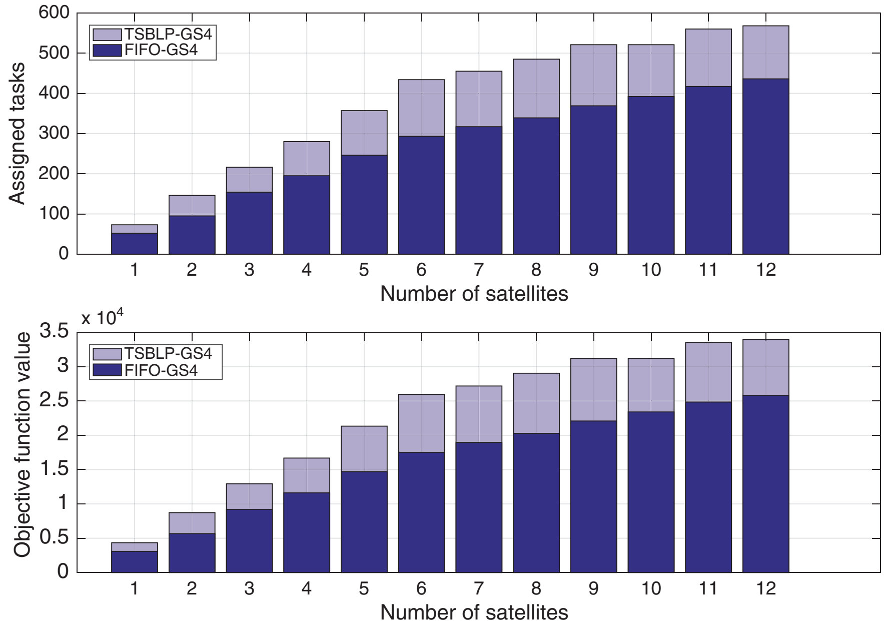

Fig. 12 Comparison of the results with different numbers of satellites.

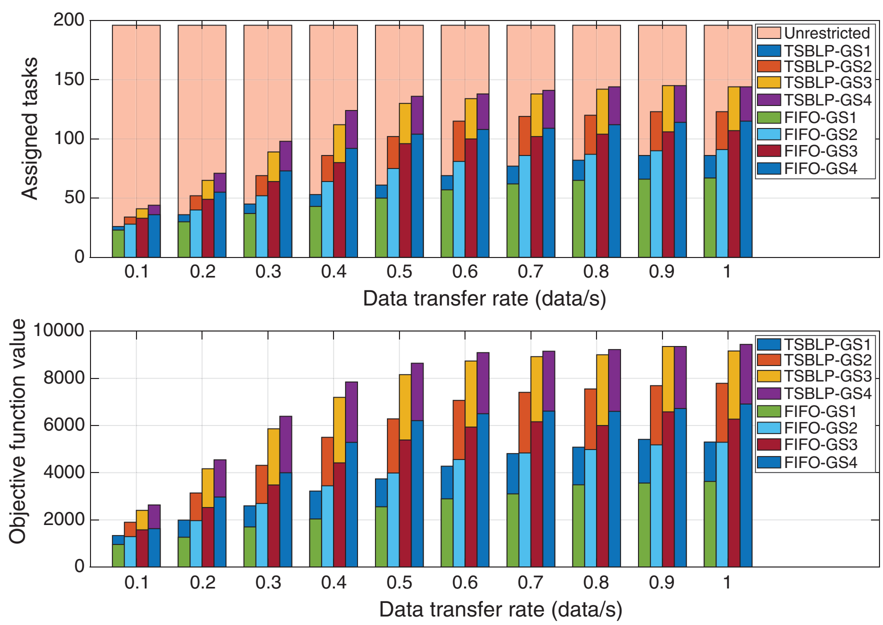

Fig. 13 Comparison of the results with different data transfer rates.

E. Data Transfer Rate

The data transfer rate could be set differently for each download time interval, but the simulation was performed by fixing the rate as the same value for each instance for convenience. The results for different data transfer rates are shown in Fig. 13 by fixing the number of observation tasks at 700 and the length of the planning horizon at 2 days. The value of the unrestricted model is drawn as red bars on the upper plot of Fig. 12 for comparison. Because of the data and energy constraints, the rewards did not substantially increase when the data transfer rate exceeded 0.6, and the gap with the unrestricted model was maintained around 30–35%. In addition, the results show a linearity depending on the number of ground stations when the data transmission rate was low due to a bottleneck.

VI. Conclusions

This Paper described the two-step binary linear programming formulation for the satellite constellation task scheduling problem that can be applied easily to obtain a high-quality solution. The first step addresses the download scheduling subproblem, in which the communication time window candidate of the satellite-Ground control center (GCC) pair was obtained in the continuous time domain by performing optimization, which maximizes the total communication time while equalizing the communication time allocated to each satellite through the maximum-minimum operator. The second step addresses the constellation mission scheduling problem, in which the scheduling output is obtained by converting the results of the first subproblem, the eclipse time obtained from STK, and user requests into problem constraints. Numerical studies showed an improvement up to 35% over traditional greedy methods. This Paper's method scaled up to 12 satellites and 700 observation tasks. Future work will explore the handling of uncertainties and stochastic nature in the problem parameters and also on the development of the effective approximate algorithm for larger-scale problems.

Appendix: Pseudocode of FIFO Greedy Algorithm

Algorithm 1 FIFO Greedy Algorithm

```
Input: Download candidate intervals, information about satellites, and observation tasks
Output: Constellation mission schedulings for each satellite
procedure SCHEDULEFIFO
  for each satellite s, do
    Initialize data and energy value
    Plan_s ← {}    ▷Plan_s: schedule of satellite s
  end for
  TW^O ← ∪_{j∈J,s∈S} TW^O_{j,s}    ▷TW^O: universal set of observation time windows
  Sort TW^O in time ascending order
  t ← 0
  while TW^O not empty do
    j, s ← observation task and satellite indices in the first element of TW^O
    t_prev ← t
    t ← start time of the first element of TW^O
    PlanD ← download schedule of satellite s from t_prev to t
    if PlanD exists, then
        Plan_s ← Plan_s ∪ PlanD
        Update data and energy status of satellite s
    end if
    PlanO ← observation schedule of assigning task j
    if (time conflict) and (data conflict) and (energy conflict), then
        Plan_s ← Plan_s ∪ PlanO
        Update data and energy status of satellite s
        Delete all time window elements related to task j from TW^O
    else
        Delete first time window element from TW^O
    end if
  end while
  for each satellite s, do    ▷add a download task after the last observation task
    PlanD ← download schedule of satellite s after the last assigned observation task
    if PlanD exists, then
        Plan_s ← Plan_s ∪ PlanD
        Update data and energy status of satellite s
    end if
  end for
  Return Plan
end procedure
```

Acknowledgments

The first and the third authors would like to acknowledge financial support by the research grant from the National Research Foundation of Korea (grant number NRF-2016M1A3A3A02017919).

References

[1] Baker, D. N., and Worden, S. P., "The Large Benefits of Small-Satellite Missions," *Transactions of the American Geophysical Union*, Vol. 89, No. 33, 2008, pp. 301–302.
doi:10.1029/2008EO330001

[2] Kennedy, A. K., and Cahoy, K. L., "Performance Analysis of Algorithms for Coordination of Earth Observation by CubeSat Constellations," *Journal of Aerospace Information Systems*, Vol. 14, No. 8, 2017, pp. 451–471.
doi:10.2514/1.I010426

[3] Selva, D., Golkar, A., Korobova, O., Cruz, I. L. I., Collopy, P., and de Weck, O. L., "Distributed Earth Satellite Systems: What Is Needed to Move Forward?" *Journal of Aerospace Information Systems*, Vol. 14, No. 8, 2017, pp. 412–438.
doi:10.2514/1.I010497

[4] Maillard, A., Verfaillie, G., Pralet, C., Jaubert, J., Sebbag, I., Fontanari, F., and Lhermitte, J., "Adaptable Data Download Schedules for Agile Earth-Observing Satellites," *Journal of Aerospace Information Systems*, Vol. 13, No. 8, 2016, pp. 280–300.
doi:10.2514/1.I010383

[5] Nagendra, N. P., "Diversification of the Indian Space Programme in the Past Decade: Perspectives on Implications and Challenges," *Space Policy*, Vol. 36, May 2016, pp. 38–45.
doi:10.1016/j.spacepol.2016.05.003

[6] Séguin, G., and Ahmed, S., "RADARSAT Constellation, Project Objectives and Status," *2009 IEEE International Geoscience and Remote Sensing Symposium, IGARSS 2009*, Vol. 2, IEEE Publ., Piscataway, NJ, 2009, pp. II–894.

[7] Pemberton, J. C., and Galiber, F., "A Constraint-Based Approach to Satellite Scheduling," *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, Vol. 57, DIMACS, Piscataway, NJ, 2001, pp. 101–114.
doi:10.1090/dimacs/057

[8] Garey, M. R., and Johnson, D. S., *Computers and Intractability: A Guide to the Theory of NP-Completeness*, Macmillan Higher Education, 1979.

[9] Barbulescu, L., Watson, J.-P., Whitley, L. D., and Howe, A. E., "Scheduling Space–Ground Communications for the Air Force Satellite Control Network," *Journal of Scheduling*, Vol. 7, No. 1, 2004, pp. 7–34.
doi:10.1023/B:JOSH.0000013053.32600.3c
[10] Lematre, M., Verfaillie, G., Jouhaud, F., Lachiver, J.-M., and Bataille, N., "Selecting and Scheduling Observations of Agile Satellites," *Aerospace Science and Technology*, Vol. 6, No. 5, 2002, pp. 367–381.
doi:10.1016/S1270-9638(02)01173-2
[11] Muraoka, H., Cohen, R. H., Ohno, T., and Doi, N., "Aster Observation Scheduling Algorithm," *Proceedings of the International Symposium Space Mission Operations and Ground Data Systems*, NASDA, JAXA, Tokyo, Japan, 1998, https://ntrs.nasa.gov/search.jsp?R=19950010764.
[12] Han, S.-M., Beak, S.-W., Cho, K.-R., Lee, D.-W., and Kim, H.-D., "Satellite Mission Scheduling Using Genetic Algorithm," *SICE Annual Conference*, IEEE Publ., Piscataway, NJ, 2008, pp. 1226–1230.
[13] Wolfe, W. J., and Sorensen, S. E., "Three Scheduling Algorithms Applied to the Earth Observing Systems Domain," *Management Science*, Vol. 46, No. 1, 2000, pp. 148–166.
doi:10.1287/mnsc.46.1.148.15134
[14] Cordeau, J.-F., and Laporte, G., "Maximizing the Value of an Earth Observation Satellite Orbit," *Journal of the Operational Research Society*, Vol. 56, No. 8, 2005, pp. 962–968.
doi:10.1057/palgrave.jors.2601926
[15] Cho, D.-H., Kim, H.-Y., and Choi, H.-L., "Optimal Continuous-Time Job Scheduling for Multiple Low Earth Orbit Satellites," *AIAA Guidance, Navigation, and Control Conference*, AIAA Paper 2016-2107, 2016.
[16] Sun, B., Mao, L., Wang, W., Xie, X., and Qin, Q., "Satellite Mission Scheduling Algorithm Based on GA," *2nd International Conference on Spatial Information Technology*, International Soc. for Optics and Photonics Paper 67950U, Wuhan, China, 2007.
[17] Qiu, D., He, C., Liu, J., and Ma, M., "A Dynamic Scheduling Method of Earth-Observing Satellites by Employing Rolling Horizon Strategy," *Scientific World Journal*, Vol. 2013, 2013, pp. 1–5.
doi:10.1155/2013/792484
[18] Martin, W., "Satellite Image Collection Optimization," *Optical Engineering*, Vol. 41, No. 9, 2002, pp. 2083–2087.
doi:10.1117/1.1495856
[19] Vasquez, M., and Hao, J.-K., "A "Logic-Constrained" Knapsack Formulation and a Tabu Algorithm for the Daily Photograph Scheduling of an Earth Observation Satellite," *Computational Optimization and Applications*, Vol. 20, No. 2, 2001, pp. 137–157.
doi:10.1023/A:1011203002719
[20] Bensana, E., Lemaitre, M., and Verfaillie, G., "Earth Observation Satellite Management," *Constraints*, Vol. 4, No. 3, 1999, pp. 293–299.
doi:10.1023/A:1026488509554
[21] Lee, J.-H., Wang, S. M., Chung, D., Ko, K. H., Choi, S., Ahn, H. S., and Jung, O., "Multi-Satellite Control System Architecture and Mission Scheduling Optimization," *2012 IEEE Aerospace Conference*, IEEE Publ., Piscataway, NJ, 2012, pp. 1–13.
[22] Baek, S.-W., Han, S.-M., Cho, K.-R., Lee, D.-W., Yang, J.-S., Bainum, P. M., and Kim, H.-D., "Development of a Scheduling Algorithm and GUI for Autonomous Satellite Missions," *Acta Astronautica*, Vol. 68, No. 7, 2011, pp. 1396–1402.
doi:10.1016/j.actaastro.2010.08.011
[23] Augenstein, S., Estanislao, A., Guere, E., and Blaes, S., "Optimal Scheduling of a Constellation of Earth-Imaging Satellites, for Maximal Data Throughput and Efficient Human Management," *Proceedings of the 26th International Conference on International Conference on Automated Planning and Scheduling*, Association for the Advancement of Artificial Intelligence, London, U.K., June 2016, pp. 345–352.
[24] Verfaillie, G., Pralet, C., and Lematre, M., "How to Model Planning and Scheduling Problems Using Constraint Networks on Timelines," *Knowledge Engineering Review*, Vol. 25, No. 03, 2010, pp. 319–336.
doi:10.1017/S0269888910000172
[25] Wang, P., Reinelt, G., Gao, P., and Tan, Y., "A Model, a Heuristic and a Decision Support System to Solve the Scheduling Problem of an Earth Observing Satellite Constellation," *Computers & Industrial Engineering*, Vol. 61, No. 2, 2011, pp. 322–335.
doi:10.1016/j.cie.2011.02.015
[26] Lin, W.-C., and Liao, D.-Y., "A Tabu Search Algorithm for Satellite Imaging Scheduling," *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*, Vol. 2, IEEE Publ., Piscataway, NJ, 2004, pp. 1601–1606.
[27] Lin, W.-C., and Chang, S.-C., "Hybrid Algorithms for Satellite Imaging Scheduling," *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*, Vol. 3, IEEE Publ., Piscataway, NJ, 2005, pp. 2518–2523.
[28] Lin, W.-C., Liao, D.-Y., Liu, C.-Y., and Lee, Y.-Y., "Daily Imaging Scheduling of an Earth Observation Satellite," *IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans*, Vol. 35, No. 2, 2005, pp. 213–223.
doi:10.1109/TSMCA.2005.843380
[29] Bianchessi, N., and Righini, G., "A Mathematical Programming Algorithm for Planning and Scheduling an Earth Observing SAR Constellation," *Proceeding of 5th International Workshop on Planning and Scheduling for Space*, 2006, http://www.stsci.edu/institute/conference/iwpss/program.
[30] Van den Akker, J., Diepen, G., and Hoogeveen, J., "Minimizing Total Weighted Tardiness on a Single Machine with Release Dates and Equal-Length Jobs," *Journal of Scheduling*, Vol. 13, No. 6, Dec. 2010, pp. 561–576.
doi:10.1007/s10951-010-0181-1
[31] Parish, D. A., "A Genetic Algorithm Approach to Automating Satellite Range Scheduling," Air Force Institute of Technology Rept. AFIT/GOR/ENS/94M-10, Wright-Patterson AFB OH, 1994.
[32] Hartmann, S., "A Competitive Genetic Algorithm for Resource-Constrained Project Scheduling," *Naval Research Logistics (NRL)*, Vol. 45, No. 7, 1998, pp. 733–750.
doi:10.1002/(ISSN)1520-6750
[33] Globus, A., Crawford, J., Lohn, J., and Pryor, A., "Scheduling Earth Observing Satellite with Evolutionary Algorithms," *Proceedings of International Conference on Space Mission Challenges for Information Technology*, 2003.
[34] Soma, P., Venkateswarlu, S., Santhalakshmi, S., Bagchi, T., and Kumar, S., "Multi-Satellite Scheduling Using Genetic Algorithms," *Space OPS 2004 Conference*, AIAA Paper 2004-515, 2004.
[35] Baek, S.-W., Cho, K.-R., Lee, D.-W., and Kim, H.-D., "A Comparison of Scheduling Optimization Algorithm for the Efficient Satellite Mission Scheduling Operation," *Journal of the Korean Society for Aeronautical & Space Sciences*, Vol. 38, No. 1, 2010, pp. 48–57.
doi:10.5139/JKSAS.2010.38.1.048
[36] Xhafa, F., Sun, J., Barolli, A., Biberaj, A., and Barolli, L., "Genetic Algorithms for Satellite Scheduling Problems," *Mobile Information Systems*, Vol. 8, No. 4, 2012, pp. 351–377.
doi:10.1155/2012/717658
[37] Globus, A., Crawford, J., Lohn, J., and Pryor, A., "A Comparison of Techniques for Scheduling Earth Observing Satellites," *Proceedings of the 19th National Conference on Artificial Intelligence*, Association for the Advancement of Artificial Intelligence, San Jose, CA, July 2004, pp. 836–843.
[38] Tanaka, S., and Fujikuma, S., "A Dynamic-Programming-Based Exact Algorithm for General Single-Machine Scheduling with Machine Idle Time," *Journal of Scheduling*, Vol. 15, No. 3, June 2012, pp. 347–361.
doi:10.1007/s10951-011-0242-0

[39] Akturk, M. S., and Ozdemir, D., "An Exact Approach to Minimizing Total Weighted Tardiness with Release Dates," *IIE Transactions*, Vol. 32, No. 11, Nov. 2000, pp. 1091–1101.
doi:10.1023/A:1013741325877

[40] Akturk, M. S., and Ozdemir, D., "A New Dominance Rule to Minimize Total Weighted Tardiness with Unequal Release Dates," *European Journal of Operational Research*, Vol. 135, No. 2, Dec. 2001, pp. 394–412.
doi:10.1016/S0377-2217(00)00319-2

[41] Jouglet, A., Baptiste, P., and Carlier, J., *Handbook of Scheduling: Algorithms, Models and Performance Analysis*, CRC Press, Boca Raton, FL, 2004, pp. 13-1–13-18.

[42] Tanaka, S., and Sato, S., "An Exact Algorithm for the Precedence-Constrained Single-Machine Scheduling Problem," *European Journal of Operational Research*, Vol. 229, No. 2, Sept. 2013, pp. 345–352.
doi:10.1016/j.ejor.2013.02.048

[43] Potts, C. N., "A Lagrangian Based Branch and Bound Algorithm for Single Machine Sequencing with Precedence Constraints to Minimize Total Weighted Completion Time," *Management Science*, Vol. 31, No. 10, Oct. 1985, pp. 1300–1311.
doi:10.1287/mnsc.31.10.1300

[44] Pan, Y., "An Improved Branch and Bound Algorithm for Single Machine Scheduling with Deadlines to Minimize Total Weighted Completion Time," *Operation Research Letters*, Vol. 31, No. 6, Nov. 2003, pp. 492–496.
doi:10.1016/S0167-6377(03)00048-8

[45] Pan, Y., and Shi, L., "Dual Constrained Single Machine Sequencing to Minimize Total Weighted Completion Time," *IEEE Transactions on Automation Science and Engineering*, Vol. 2, No. 4, Oct. 2005, pp. 344–357.
doi:10.1109/TASE.2005.853474

[46] Davari, M., Demeulemeester, E., Leus, R., and Nobibon, F. T., "Exact Algorithms for Single-Machine Scheduling with Time Windows and Precedence Constraints," *Journal of Scheduling*, Vol. 19, No. 3, 2016, pp. 309–334.
doi:10.1007/s10951-015-0428-y

[47] Wang, J.-M., Li, J.-F., and Tan, Y.-J., "Study on Heuristic Algorithm for Dynamic Scheduling Problem of Earth Observing Satellites," *Proceedings of the 8th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing*, International Association for Computer and Information Science, International Association for Computer and Information Science (IEEE/ACIS), July 2007, pp. 9–14.

[48] Bianchessi, N., Cordeau, J.-F., Desrosiers, J., Laporte, G., and Raymond, V., "A Heuristic for the Multi-Satellite Multi-Orbit and Multi-User Management of Earth Observation Satellites," *European Journal of Operational Research*, Vol. 177, No. 2, March 2007, pp. 750–762.
doi:10.1016/j.ejor.2005.12.026

[49] Wang, J., Zhu, X., Zhu, J., Ma, M., and Yang, L. T., "DMTRH: A Real-Time Scheduling Algorithm for Multiple Earth Observation Satellites," *2012 IEEE 14th International Conference on High Performance Computing and Communications*, IEEE Publ., Piscataway, NJ, 2012, pp. 673–680.

[50] Wu, G., Ma, M., Zhu, J., and Qiu, D., "Multi-Satellite Observation Integrated Scheduling Method Oriented to Emergency Tasks and Common Tasks," *Journal of Systems Engineering and Electronics*, Vol. 23, No. 5, Oct. 2012, pp. 723–733.
doi:10.1109/JSEE.2012.00089

[51] Qin, J., Liu, Y., Mao, X., and McNair, J., "Deadline Based Resource Balancing Task Allocation for Clustered Heterogeneous LEO Small Satellite Network," *2013 IEEE Military Communications Conference*, IEEE Publ., Piscataway, NJ, 2013, pp. 1825–1831.

[52] Wang, J., Demeulemeester, E., and Qiu, D., "A Pure Proactive Scheduling Algorithm for Multiple Earth Observation Satellites Under Uncertainties of Clouds," *Computers & Operations Research*, Vol. 74, Oct. 2016, pp. 1–13.
doi:10.1016/j.cor.2016.04.014

[53] Wei, Z., Xin, M., and He, H., "Genetic Algorithm for TDRS Communication Scheduling with Resource Constraints," *2008 International Conference on Computer Science and Software Engineering*, Vol. 3, IEEE Publ., Piscataway, NJ, 2008, pp. 893–897.

[54] Marinelli, F., Nocella, S., Rossi, F., and Smriglio, S., "A Lagrangian Heuristic for Satellite Range Scheduling with Resource Constraints," *Computers & Operations Research*, Vol. 38, No. 11, 2011, pp. 1572–1583.
doi:10.1016/j.cor.2011.01.016

[55] Spangelo, S., Cutler, J., Gilson, K., and Cohn, A., "Optimization-Based Scheduling for the Single-Satellite, Multi-Ground Station Communication Problem," *Computers & Operations Research*, Vol. 57, May 2015, pp. 1–16.
doi:10.1016/j.cor.2014.11.004

[56] Monmousseau, P., "Scheduling of a Constellation of Satellites: Improving a Simulated Annealing Model by Creating a Mixed-Integer Linear Model," KTH Royal Institute of Technology, Stockholm, Sweden, 2015.

[57] Castaing, J., "Scheduling Downloads for Multi-Satellite, Multi-Ground Station Missions," *Proceedings of the AIAA/USU Conference on Small Satellites, Frank J. Redd Student Competition*, SSC14-VIII-4, 2014, https://digitalcommons.usu.edu/smallsat/2014/FJRStudentComp/4/.

[58] Cahoy, K., and Kennedy, A. K., "Initial Results from ACCESS: An Autonomous CubeSat Constellation Scheduling System for Earth Observation," 2017, https://digitalcommons.usu.edu/smallsat/2017/all2017/98/.

[59] Walker, J. G., "Satellite Constellations," *Journal of the British Interplanetary Society*, Vol. 37, Dec. 1984, pp. 559–572.

[60] "Measuring Power Generation of Solar Panels on a Satellite," American Geological Inst., Pennsylvania, 2018, http://help.agi.com/stk/index.htm#training/SeasonalPower.htm.

[61] Tsiotras, P., Shen, H., and Hall, C., "Satellite Attitude Control and Power Tracking with Energy/Momentum Wheels," *Journal of Guidance, Control, and Dynamics*, Vol. 24, No. 1, 2001, pp. 23–34.
doi:10.2514/2.4705

[62] De Florio, S., "Performances Optimization of Remote Sensing Satellite Constellations: A Heuristic Method," *Proceedings of the 5th International Workshop on Planning and Scheduling for Space*, 2006, http://www.stsci.edu/institute/conference/iwpss/program.

M. J. Kochenderfer
Associate Editor