(Preprint) AAS 22-825

MAXIMIZING OBSERVATION THROUGHPUT VIA MULTI-STAGE SATELLITE CONSTELLATION RECONFIGURATION

Hang Woon Lee[*], Hao Chen[†], and Koki Ho[‡]

We examine the problem of multi-stage satellite constellation reconfiguration in the domain of Earth observations. The goal of the problem is to maximize the total system observation throughput by actively manipulating the orbits and the relative phasing of the constituent satellites. We propose a novel integer linear programming formulation of the problem that is constructed based on the concept of time-expanded networks. To tackle the computational intractability arising due to the combinatorial explosion of the solution space, we propose two decomposition-based algorithmic frameworks based on the principles of the myopic policy and the rolling horizon procedure. We empirically present that these heuristics produce high-quality solutions relative to optimal solutions. We conduct computational experiments to demonstrate the value of the proposed work.

INTRODUCTION

Satellite constellation reconfiguration provides a space-borne system with flexibility and responsiveness in response to dynamic changes in mission requirements and environments. The concept of constellation reconfiguration has been explored in different application domains including the Earth observations (EO),[1–3] telecommunications,[4] and navigation and positioning systems[5] for various reasons, spanning from the staged deployment to disaster monitoring. In this paper, in the domain of Earth observations, we investigate the problem of reconfiguring a fleet of (potentially) heterogeneous satellites through multiple stages to maximize the observation rewards by achieving the coverage on targets of interest requested by clients.

The ultimate goal of Earth observations satellite systems is to maximize the system observational throughput. Prior studies have investigated the EO satellite scheduling problems (EOSSP) whose goal is to maximize the observation profit during a specified mission planning horizon while satisfying the complex operational constraints (e.g., solar panel charging, downlinking raw images).[6,7] In the classical EOSSP context, one of the underyling assumptions is that satellites point to their nadir directions without any attitude or maneuver controllability. Due to this assumption, the visible time windows (VTWs), which define the periods of satellite-to-target visibility, are considered fixed parameters to the scheduling problems. To improve the observational throughput, recent studies have explored the concept of "agile satellites" with attitude control capability.[8,9] The agile satellites can control the orientation of their spacecraft body and directly manipulate the VTWs within an EO scheduler. Therefore, the longer duration of the VTWs can be obtained, which in turn has the potential to enhance the overall observational throughput and scheduling efficiency.

[*]Ph.D. Candidate, Daniel Guggenheim School of Aerospace Engineering, Georgia Institute of Technology.
[†]Assistant Professor, School of Systems and Enterprises, Stevens Institute of Technology
[‡]Assistant Professor, Daniel Guggenheim School of Aerospace Engineering, Georgia Institute of Technology.

We draw on the concept of satellite orbital transfer maneuverability as one of the most prominent notions of system flexibility along with satellite agility. Existing literature on satellite constellation reconfiguration has extensively focused on single-stage reconfiguration problem[10, 11] or without the consideration of optimizing a long mission planning horizon. However, constellation systems often face a series of reconfiguration opportunities arising due to satellite failures or change in mission objectives. Moreover, with the recent developments in on-orbit servicing, there is a greater potential to equip satellites with enhanced mobility for active orbital maneuvers.[12] Albeit these opportunities, several challenges also emerge. In practicing multi-stage reconfiguration, one of the main challenges that we confront is the excessive fuel consumption, particularly for high-thrust maneuvers in the low Earth orbit (LEO) regime. Therefore, it is critical to optimally lay out a set of orbital transfer paths of satellites through stages to maximize the profit of reconfiguration over a long-term mission planning horizon considering the fuel constraints.

In response to this background, we propose a novel integer linear (ILP) programming formulation for the multi-stage constellation reconfiguration problem (MCRP). MCRP is an extension to our prior work on single-stage reconfiguration problem.[13] Therefore, the formulation inherently features the *heterogeneity* in satellite hardware specifications and orbital characteristics, and *asymmetry* in satellite distribution. The consideration of the heterogeneity is especially useful in modeling a cooperative EO missions such as disaster monitoring.[14, 15] The asymmetric satellite distribution can lead to efficient constellation pattern sets for EO applications as demonstrated in Reference 16. The problem can be solved using a state-of-the-art branch-and-bound algorithm for provably-optimal solutions.

The contribution of this paper is two-fold:

1. **Multi-stage constellation reconfiguration problem.** We extend our prior work on single-stage constellation reconfiguration problems using the basis of a time-expanded network. This modeling allows us to better understand the hidden design space that is otherwise overlooked with one or zero reconfiguration stages. The proposed model aims to concurrently optimize the design and transfer aspects of multiple reconfigurations over the entire mission planning horizon.
2. **Heuristic solution methods and empirical analysis.** We propose two divide-and-conquer heuristic solution methods based on myopic policy and rolling horizon procedure to address the issue of computational intractability in solving large-scale problems. We empirically show that the myopic policy heuristic can be beneficial for instances with uniform observation rewards, and the rolling horizon procedure can be efficient for instances with dynamic environments.

The remainder of this paper is organized as follows. In the second section, we provide the formal description of the problem of multi-stage constellation reconfiguration and propose a novel ILP formulation. The third section discusses two heuristic solution methods. The fourth section then conducts computational experiments to compare the performances of the proposed methods on two sets of randomly-generated test instances. Lastly, we provide several interesting future work directions to enhance the applicability of the proposed work and conclude this paper.

MULTI-STAGE CONSTELLATION RECONFIGURATION PROBLEM

In this section, we describe and propose a mathematical optimization formulation of MCRP.

Problem Description

Given a finite discrete-time mission planning horizon $\mathcal{T} = \{1, 2, \dots, T\}$, the objective of MCRP is to find a set of orbital transfer maneuver sequences of K heterogeneous satellites that maximizes the total obtained observation reward imposed on a set of target points $\mathcal{P}$. Each satellite $k \in \mathcal{K}$ is associated with different hardware specifications (e.g., sensor field-of-view and propellant capacities) and orbital characteristics. In addition, each target point $p \in \mathcal{P}$ is associated with the non-negative time-dependent observation reward $\boldsymbol{\pi}_p = (\pi_{tp} \geq 0 : t \in \mathcal{T})$[*] and the minimum elevation angle threshold $\varepsilon_{\min}$. The observation reward π_{tp} is earned if at least r_{tp} number of satellites simultaneously cover target p at time step t. The concept of time-dependent observation reward models the "value" of the sensory data taken at a different time of day. One motivating example is the remote sensing application with visible spectral sensors; images taken under the Sun illumination may possess greater value than otherwise.

Time-Expanded Network During the specified mission planning horizon, there are N *stages* at which a constellation can undergo reconfiguration processes. Denoting $\mathcal{S} = \{0, 1, \dots, N\}$ by the set of stages (we let $s = 0$ indicate the initial state), we associate stage $s \in \mathcal{S}$ with the time-stamp $t_s \in \mathcal{T}$ and the stage planning horizon $\mathcal{T}_s = \{t : t_s \leq t < t_{s+1}, t \in \mathcal{T}\}$. We say that new stage forms at stage s. Without loss of generality, we assume that stages evenly distribute the mission planning horizon and satellites simultaneously arrive at their new destination orbital slots at t_s.

The flows of satellites through stages are defined by a set of directed graphs $\{\mathcal{G}^1, \dots, \mathcal{G}^K\}$ where we associate each satellite k with its own time-expanded graph (TEG) $\mathcal{G}^k = (\mathcal{J}^k, \mathcal{A}^k)$ as shown in the top part of Figure 1. Here, $\mathcal{J}^k = \{\mathcal{J}_0^k, \mathcal{J}_1^k, \dots, \mathcal{J}_N^k\}$ is the set of the source node $\mathcal{J}_0^k$ and the time-expanded nodes $\{\mathcal{J}_1^k, \dots, \mathcal{J}_N^k\}$ and $\mathcal{A}^k = \{\mathcal{A}_1^k, \dots, \mathcal{A}_N^k\}$ is the set of arcs that connect the nodes of two adjacent stages. Here, each node set $\mathcal{J}_s^k, s \geq 1$ is a copy of $\mathcal{J}_1^k$ and has the cardinality J. The concept of TEGs allows us to model the time evolution of satellites through stages over a set of identical nodes and arcs by associating each node with time-stamps. At each stage s, satellite k has options to either stay in its orbit or perform an active orbital maneuver to transfer from a prior stage's orbit $i \in \mathcal{J}_{s-1}^k$ to a new orbit $j \in \mathcal{J}_s^k$ with the non-negative cost of transfer $c_{ij} \geq 0$, which is deducted from the available resource $c_{s,\max}^k$. The source node of satellite k is included as an element of $\mathcal{J}_s^k, \forall s \in \mathcal{S}$ to enable the option to stay in orbit; consequently, $\exists c_{ij} = 0$ for $(i, j) \in \mathcal{A}_s^k, \forall s \in \mathcal{S} \setminus \{0\}, \forall k \in \mathcal{K}$.

Observation Reward Mechanism Each node is an orbital slot with the fixed coordinate in the Earth-centered inertial frame and is associated with the visibility profile per target point. Letting $x_j = 1$ to indicate the occupancy of orbital slot $j \in \mathcal{J}_s^k$ by satellite k at stage s ($x_j = 0$, otherwise), a set of newly occupied orbital slots forms a new constellation configuration $\mathcal{C}_s := \{j : x_j = 1, j \in \mathcal{J}_s^k, k \in \mathcal{K}\}$ that is valid for the time interval $[t_s, t_{s+1})$ (see the middle part of Figure 1). We denote $\boldsymbol{y}_p = (y_{tp} \in \{0, 1\} : t \in \mathcal{T})$ by the VTW of target p where $y_{tp} = 1$ if target p is covered simultaneously by at least r_{tp} satellites ($y_{tp} = 0$, otherwise). The VTWs are the function of constellation configuration $\mathcal{C}_s$ during the stage planning horizon $\mathcal{T}_s$. Observation rewards are realized when VTWs are aligned with the periods of the non-negative observation rewards (see the bottom part of Figure 1). To obtain the maximal sum $\sum_{p \in \mathcal{P}} \sum_{t \in \mathcal{T}_s} \pi_{tp} y_{tp}$, the alignments of VTWs and the periods of high observation rewards need to be maximized.

[*] Note that our problem differs from the conventional problem settings considered in the EOSSP literature. In EOSSP, the images requested by the clients only need to be acquired once. The problem we are considering is the imaging of the same target points for a longer duration.

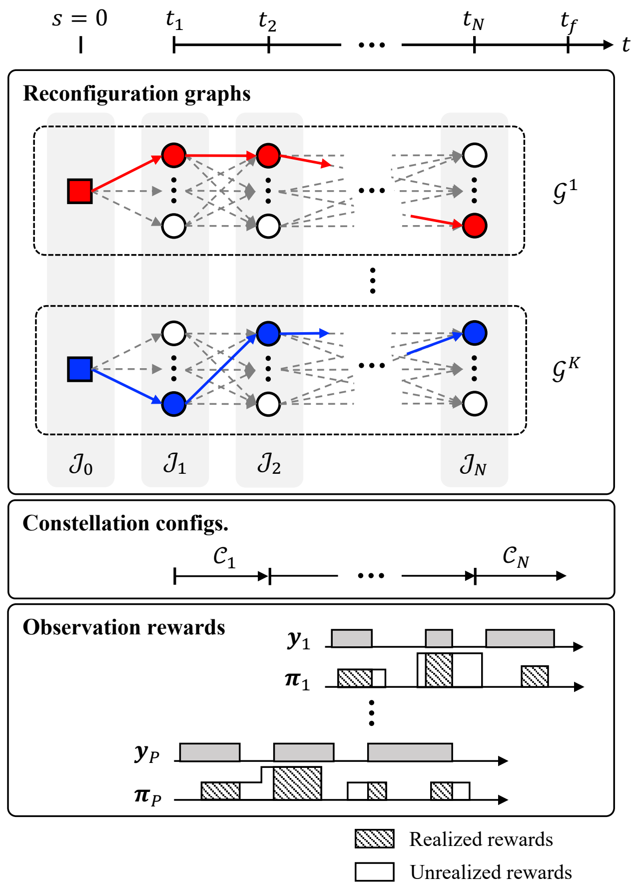


Figure 1: Mission planning horizon, reconfiguration graphs, constellation configurations, and observation rewards.

Remarks Each stage involves the optimization of (i) the *design* of a maximal-reward destination configuration and (ii) the minimum-cost *transfer* of satellites from one configuration to another. MCRP is an extension to the single-stage design-transfer problem explored in Reference 13 by expanding it in the time dimension. In MCRP, all stages are coupled through the resource budget constraints. Therefore, no stage can be individually solved to obtain the true optimal solution to MCRP. If configuration designs in early stages are aggressive in terms of the resource consumption, then there will be no (or low) flexibility in designing good configurations in later stages. MCRP is about determining the optimal balance between the cost and the performance over the entire mission planning horizon. By adding more degrees of freedom for reconfiguration throughout the mission planning horizon, the goal is to infuse more flexibility into the system and identify an optimal reconfiguration process that is otherwise overlooked with one or zero reconfiguration stages.

Mathematical Formulation

MCRP is a deterministic multi-period decision-making problem with the basis of network flows. However, the problem is not fully defined in a graph-theoretic setting because the reward set on a node is not a scalar value but involves conditional evaluation due to the linking between the constellation configuration and its coverage state [Constraints (2d)]. Therefore, the use of efficient algorithms for network flow problems such as the longest path problem cannot be applied.

To this end, we formulate MCRP as an ILP optimization problem. First, we define sets, parameters, and decision variables. Then, we introduce the mathematical formulation of MCRP.

Sets and indices

$\mathcal{S}$	Set of stage indices (index s; cardinality $N+1$)
$\mathcal{K}$	Set of satellite indices (index k; cardinality K)
$\mathcal{J}_s^k$	Set of orbital slot indices of stage s for satellite k (indices i, j; cardinality J)
$\mathcal{P}$	Set of target point indices (index p; cardinality P)
$\mathcal{T}_s$	Planning horizon for stage s (index t)
$\mathcal{T}$	Mission planning horizon (index t; cardinality T)

Parameters

c_{ij}	Cost of transferring satellite k from orbital slot $i \in \mathcal{J}_{s-1}^k$ to orbital slot $j \in \mathcal{J}_s^k$ ($c_{ij} \in \mathbb{R}_{\geq 0}$)
$c_{\max}^k$	Resource availability for satellite k
π_{tp}	Coverage reward for target point p at time step t ($\pi_{tp} \in \mathbb{R}_{\geq 0}$)
r_{tp}	Minimum coverage threshold to receive the reward of target point p at time step t ($r_{tp} \in \mathbb{Z}_{\geq 1}$)
V_{tjp}	$\begin{cases} 1, & \text{if orbital slot } j \text{ is visible from target point } p \text{ at time step } t \\ 0, & \text{otherwise} \end{cases}$

Decision variables

φ_{ij}	$\begin{cases} 1, & \text{if satellite } k \text{ transfers from orbital slot } i \in \mathcal{J}_{s-1}^k \text{ to orbital slot } j \in \mathcal{J}_s^k \\ 0, & \text{otherwise} \end{cases}$
y_{tp}	$\begin{cases} 1, & \text{if target point } p \text{ is covered at time step } t \\ 0, & \text{otherwise} \end{cases}$

We denote $\mathbb{R}_{\geq 0}$ by the set of non-negative real numbers and $\mathbb{Z}_{\geq 1}$ by the set of integer numbers greater than or equal to one. We can relate the relationship between a flow on (i, j) with the destination node j:

$$x_j = \sum_{i \in \mathcal{J}_{s-1}^k} \varphi_{ij}, \quad \forall j \in \mathcal{J}_s^k, \forall s \in \mathcal{S} \setminus \{0\}, \forall k \in \mathcal{K} \tag{1}$$

The mathematical formulation of MCRP for maximal observation throughput is as follows:

$$
\begin{aligned}
(\text{MCRP}) \quad \max \quad & \sum_{p\in\mathcal{P}}\sum_{t\in\mathcal{T}} \pi_{tp} y_{tp} && \text{(2a)}\\
\text{s.t.} \quad & \sum_{j\in\mathcal{J}_1^k} \varphi_{ij} = 1, \quad \forall i \in \mathcal{J}_0^k, \forall k \in \mathcal{K} && \text{(2b)}\\
& \sum_{j\in\mathcal{J}_{s+1}^k} \varphi_{ij} - \sum_{q\in\mathcal{J}_{s-1}^k} \varphi_{qi} = 0, \quad \forall i \in \mathcal{J}_s^k, \forall s \in \mathcal{S}\setminus\{0,N\}, \forall k \in \mathcal{K} && \text{(2c)}\\
& \sum_{k\in\mathcal{K}}\sum_{j\in\mathcal{J}_s^k}\sum_{i\in\mathcal{J}_{s-1}^k} V_{tjp}\varphi_{ij} \geq r_{tp} y_{tp}, \quad \forall t \in \mathcal{T}_s, \forall s \in \mathcal{S}\setminus\{0\}, \forall p \in \mathcal{P} && \text{(2d)}\\
& \sum_{s\in\mathcal{S}\setminus\{0\}}\sum_{j\in\mathcal{J}_s^k}\sum_{i\in\mathcal{J}_{s-1}^k} c_{ij}\varphi_{ij} \leq c_{\max}^k, \quad \forall k \in \mathcal{K}' \subseteq \mathcal{K} && \text{(2e)}\\
& \varphi_{ij} = \{0,1\}, \quad \forall i \in \mathcal{J}_{s-1}^k, \forall j \in \mathcal{J}_s^k, \forall s \in \mathcal{S}\setminus\{0\}, \forall k \in \mathcal{K} && \text{(2f)}\\
& y_{tp} = \{0,1\}, \quad \forall t \in \mathcal{T}, \forall p \in \mathcal{P} && \text{(2g)}
\end{aligned}
$$

The objective function (2a) maximizes the total reward (i.e., the total observation throughput) obtained by covering a set of target points of interest. Constraints (2b) are the initial stage outflow constraints. Constraints (2c) balances the outflow (the first term) and inflow (the second term) of the nodes of intermediate stages. Constraints (2d) are the configuration-coverage linking constraints that ensure that target point p is covered at time step t only if there exists at least r_{tp} satellite(s) in view. Constraints (2d) couples the flow of satellites at every stage. Constraints (2e) are the resource availability constraints that restrict the maximum allowable Δv of satellite k to $c_{\max}^k$. The set $\mathcal{K}' \subseteq \mathcal{K}$ is used to denote the subset of satellites that impose such resource availability constraints. Constraints (2f)–(2g) define the domain of decision variables.

HEURISTIC METHODS

In the previous section, we formulated MCRP as an integer linear program. Consequently, we can utilize generic mixed-integer linear programming (MILP) methods such as the branch-and-bound algorithm to solve the problem. However, MCRP is a combinatorial optimization problem that suffers from the curse of dimensionality as the total number of potentially feasible plans grows exponentially with the linear increase in J, N, and K (e.g., there are at most J^{NK} plans to consider). For example, an instance I of MCRP with 3 reconfiguration stages, 5 satellites, and 50 candidate orbital slots per satellite has up to 3.05×10^{25} potentially feasible plans. The enumeration of these plans can be computationally prohibitive.

To address the computational intractability in solving MCRP, we construct two sequential decision-making heuristic methods based on the principles of myopic policy and the rolling horizon procedure.[17] Feasible solutions obtained by the heuristic methods are feasible solutions to MCRP.

Myopic Policy Heuristic

To circumvent the challenge of combinatorial explosion, we develop a divide-and-conquer sequential decision-making framework called the *Myopic Policy Heuristic* (MPH). The principal idea is to partition MCRP by stages into N smaller subproblems with manageable sizes and solve subproblems in a successive manner. With the knowledge of the satellite states from the precedent stage, which we denote with $\tilde{\mathcal{J}}_{s-1}$, the number of potentially feasible plans effectively reduces to J^K per subproblem. Considering the same instance I of MCRP, we can partition the problem into 3 subproblems; each subproblem has up to 3.13×10^8 potentially feasible plans. Small subproblems can be efficiently solved using a commercial software package. Nevertheless, additional algorithmic efforts can be applied on the basis of MPH to further improve the performance such as the solution quality and the time complexity. It is important to note that MPH is an algorithmic framework with a myopic policy. That is, no impacts on the future stages are considered in the current-stage decision-making. Figure 2 illustrate the scope of a subproblem.

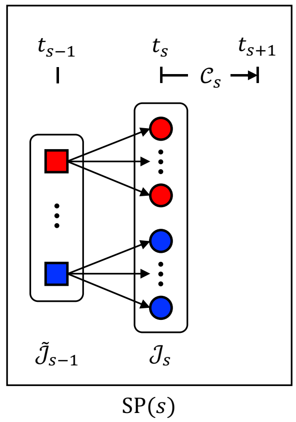


Figure 2: MPH subproblem for stage s.

Subproblem Formulation A subproblem is parameterized with the stage index s and is denoted with SP(s). The mathematical formulation of SP(s) is as follows:

$$(\mathsf{SP}(s)) \quad z_s = \max \sum_{p \in \mathcal{P}} \sum_{t \in \mathcal{T}_s} \pi_{tp} y_{tp} \tag{3a}$$

$$\text{s.t.} \quad \sum_{j \in \mathcal{J}_s^k} \varphi_{ij} = 1, \quad \forall i \in \tilde{\mathcal{J}}_{s-1}^k, \forall k \in \mathcal{K} \tag{3b}$$

$$\sum_{i \in \tilde{\mathcal{J}}_{s-1}^k} \varphi_{ij} \leq 1, \quad \forall j \in \mathcal{J}_s^k, \forall k \in \mathcal{K} \tag{3c}$$

$$\sum_{k \in \mathcal{K}} \sum_{j \in \mathcal{J}_s^k} \sum_{i \in \tilde{\mathcal{J}}_{s-1}^k} V_{tjp} \varphi_{ij} \geq r_{tp} y_{tp}, \quad \forall t \in \mathcal{T}_s, \forall p \in \mathcal{P} \tag{3d}$$

$$\sum_{j \in \mathcal{J}_s^k} \sum_{i \in \tilde{\mathcal{J}}_{s-1}^k} c_{ij} \varphi_{ij} \leq c_{s,\max}^k, \quad \forall k \in \mathcal{K}' \subseteq \mathcal{K} \tag{3e}$$

$$\varphi_{ij} = \{0, 1\}, \quad \forall i \in \tilde{\mathcal{J}}_{s-1}^k, \forall j \in \mathcal{J}_s^k, \forall k \in \mathcal{K} \tag{3f}$$

$$y_{tp} = \{0, 1\}, \quad \forall t \in \mathcal{T}_s, \forall p \in \mathcal{P} \tag{3g}$$

The objective function (3a) maximizes the total observation reward for stage s. Constraints (3b) and (3c) are the usual assignment problem constraints. Constraints (3d) are the configuration-coverage linking constraints for stage s. Constraints (3e) are the individual resource availability constraints; unlike Constraints (2e), the resource availability constraints reflect the resource consumptions from the prior stages $\{1, \ldots, s-1\}$ and parameterize them; it is computed as follows:

$$c_{s,\max}^{k} = c_{\max}^{k} - \sum_{q=1}^{s-1} \sum_{j \in \mathcal{J}_q^k} \sum_{i \in \tilde{\mathcal{J}}_{q-1}^k} c_{ij} \varphi_{ij} \tag{4}$$

Constraints (3f) and (3g) define the domain of decision variables.

Algorithm 1 outlines the overall solution procedure of MPH. SP(s) outputs the optimum z_s and the optimal assignment solution $\boldsymbol{\varphi}_s^* = (\varphi_{ij}^* = \{0,1\}, i \in \tilde{\mathcal{J}}_{s-1}^k, j \in \mathcal{J}_s^k, k \in \mathcal{K})$ and the optimal coverage state solution $\boldsymbol{y}_s^* = (y_{tp}^* \in \{0,1\} : t \in \mathcal{T}_s, p \in \mathcal{P})$. (Notice that the term optimality is with respect to SP(s)). The algorithm stores the results from each stage and returns the heuristic solution objective value z_{mph}, which is the sum of all z_s, and a feasible solution $(\boldsymbol{\varphi}^*, \boldsymbol{y}^*)$, which is the collection of all $(\boldsymbol{\varphi}_s^*, \boldsymbol{y}_s^*)$, to MCRP.

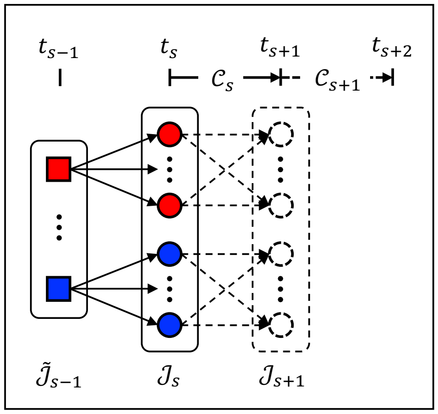
Algorithm 1: Myopic policy heuristic

Input: $\boldsymbol{c}, \boldsymbol{\pi}, \boldsymbol{V}, \boldsymbol{r}$
Output: z_{mph}, $(\boldsymbol{\varphi}^*, \boldsymbol{y}^*)$
Initialize $s \leftarrow 1$
Compute SP(s) and store: z_1 and $(\boldsymbol{\varphi}_1^*, \boldsymbol{y}_1^*)$
$s \leftarrow s+1$
while $s \leq N$ **do**
    Update $\tilde{\mathcal{J}}_{s-1}^k \leftarrow \{j : \varphi_{ij} = 1, i \in \mathcal{J}_{s-2}^k, j \in \mathcal{J}_{s-1}^k\}$
    Compute SP(s) and store: z_s and $(\boldsymbol{\varphi}_s^*, \boldsymbol{y}_s^*)$
    $s \leftarrow s+1$
$z_{\text{mph}} \leftarrow \sum_{s \in \mathcal{S} \setminus \{0\}} z_s$
$(\boldsymbol{\varphi}^*, \boldsymbol{y}^*) \leftarrow \big((\boldsymbol{\varphi}_s^*, \boldsymbol{y}_s^*) : s \in \mathcal{S} \setminus \{0\}\big)$

The formulation of SP(s) is identical to the regional constellation reconfiguration problem with individual resource constraints (RCRP-IRC) as outlined in Reference 13. The problem embeds a budgeted assignment problem and a maximal covering location problem. As discussed previously, any dedicated algorithm can be applied to solve SP(s). In our case, we can exploit the established Lagrangian-relaxation based heuristic method for RCRP-IRC.

Rolling Horizon Procedure

The *Rolling Horizon Procedure* (RHP) uses the impact of the current-stage decisions on future stages to make informed decisions at the current stage.[18] In this paper, we use the deterministic 1-stage lookahead policy. This allows us to partition MCRP into $N-1$ smaller subproblems; the last iteration at $s = N-1$ deterministically optimizes the entire remaining mission planning horizon. Due to the lookahead policy, each subproblem is larger than the subproblem SP(s) of MPH. Considering the same instance I of MCRP, we can partition the problem into 2 subproblems; each subproblem has up to 9.76×10^{16} potentially feasible plans. Figure 3 illustrates the scope of a subproblem with the 1-stage lookahead policy.



1-SP(s)

Figure 3: RHP subproblem for stage s with the 1-stage lookahead policy.

Subproblem Formulation We denote 1-SP(s) by a subproblem parameterized with the stage index s and the 1-stage lookahead policy. The mathematical formulation of 1-SP(s) is as follows:

$$\text{(1-SP}(s)) \quad z_s + z_{s+1} = \max \sum_{p\in\mathcal{P}} \sum_{t\in\{\mathcal{T}_s,\mathcal{T}_{s+1}\}} \pi_{tp} y_{tp} \tag{5a}$$

$$\text{s.t.} \quad \sum_{j\in\mathcal{J}_s^k} \varphi_{ij} = 1, \quad \forall i \in \tilde{\mathcal{J}}_{s-1}^k, \forall k \in \mathcal{K} \tag{5b}$$

$$\sum_{j\in\mathcal{J}_{s+1}^k} \varphi_{ij} - \sum_{q\in\tilde{\mathcal{J}}_{s-1}^k} \varphi_{qi} = 0, \quad \forall i \in \mathcal{J}_s^k, \forall k \in \mathcal{K} \tag{5c}$$

$$\sum_{k\in\mathcal{K}} \sum_{j\in\mathcal{J}_s^k} \sum_{i\in\tilde{\mathcal{J}}_{s-1}^k} V_{tjp}\varphi_{ij} \geq r_{tp} y_{tp}, \quad \forall t \in \mathcal{T}_s, \forall p \in \mathcal{P} \tag{5d}$$

$$\sum_{k\in\mathcal{K}} \sum_{j\in\mathcal{J}_{s+1}^k} \sum_{i\in\mathcal{J}_s^k} V_{tjp}\varphi_{ij} \geq r_{tp} y_{tp}, \quad \forall t \in \mathcal{T}_{s+1}, \forall p \in \mathcal{P} \tag{5e}$$

$$\sum_{j\in\mathcal{J}_s^k} \sum_{i\in\tilde{\mathcal{J}}_{s-1}^k} c_{ij}\varphi_{ij} + \sum_{j\in\mathcal{J}_{s+1}^k} \sum_{i\in\mathcal{J}_s^k} c_{ij}\varphi_{ij} \leq c_{s,\max}^k, \quad \forall k \in \mathcal{K}' \subseteq \mathcal{K} \tag{5f}$$

$$\varphi_{qi}, \varphi_{ij} = \{0,1\}, \quad \forall q \in \tilde{\mathcal{J}}_{s-1}^k, \forall i \in \mathcal{J}_s^k, \forall j \in \mathcal{J}_{s+1}^k, \forall k \in \mathcal{K} \tag{5g}$$

$$y_{tp} = \{0,1\}, \quad \forall t \in \{\mathcal{T}_s, \mathcal{T}_{s+1}\}, \forall p \in \mathcal{P} \tag{5h}$$

The objective function (5a) maximizes the sum of observation rewards for stages s and $s+1$. Constraints (5b) and (5c) are the usual network flow conservation constraints. Constraints (5d) and (5e) are the configuration-coverage linking constraints for stages s and $s+1$, respectively. Constraints (5f) are the individual resource availability constraints; $c_{s,\max}^k$ is defined in Eq. (4). Constraints (5g) and (5h) define the domain of decision variables.

Algorithm 2 overviews RHP. In essence, the structure of this algorithm is very similar to that of Algorithm 1 but it possesses one distinct characteristics. The arguments of 1-SP(s) are $(\boldsymbol{\varphi}_s^*, \boldsymbol{y}_s^*)$

and $(\boldsymbol{\varphi}^*_{s+1}, \boldsymbol{y}^*_{s+1})$. However, we do not skip stage $+1$ and proceed directly to stage $s+2$. Instead, we only make decisions for the stage s using the deterministic forecast of the next immediate stage and proceed to stage $s+1$. As discussed previously, no decisions are made for stage $s+1$ except if the current stage is at $N-1$.

Algorithm 2: Rolling horizon procedure

Input: $\boldsymbol{c}$, $\boldsymbol{\pi}$, $\boldsymbol{V}$, $\boldsymbol{r}$
Output: z_{rhp}, $(\boldsymbol{\varphi}^*, \boldsymbol{y}^*)$

Initialize $s \leftarrow 1$
Compute 1-SP(s) and store: z_1 and $(\boldsymbol{\varphi}^*_1, \boldsymbol{y}^*_1)$
$s \leftarrow s+1$
while $s \leq N-1$ **do**
  Update $\tilde{\mathcal{J}}^k_{s-1} \leftarrow \{j : \varphi_{ij} = 1, i \in \mathcal{J}^k_{s-2}, j \in \mathcal{J}^k_{s-1}\}$
  if $s < N-1$ **then**
    Compute 1-SP(s) and store: z_s and $(\boldsymbol{\varphi}^*_s, \boldsymbol{y}^*_s)$
  else
    Compute 1-SP(s) and store: $\{z_s, z_{s+1}\}$ and $\big((\boldsymbol{\varphi}^*_s, \boldsymbol{y}^*_s), (\boldsymbol{\varphi}^*_{s+1}, \boldsymbol{y}^*_{s+1})\big)$
  $s \leftarrow s+1$
$z_{\text{rhp}} \leftarrow \sum_{s \in \mathcal{S} \setminus \{0\}} z_s$
$(\boldsymbol{\varphi}^*, \boldsymbol{y}^*) \leftarrow \big((\boldsymbol{\varphi}^*_s, \boldsymbol{y}^*_s) : s \in \mathcal{S} \setminus \{0\}\big)$

Upper Bound of MCRP for Heuristic Solution Gap Analysis

The principal motivation for us to consider MPH and RHP is to address the issue of the curse of dimensionality in solving MCRP. While these heuristic methods can compute feasible solutions to MCRP relatively faster than concurrently solving for the entire mission planning horizon, there is no guarantee on the optimality of the obtained solutions.

To analyze the quality of a generic heuristic solution z_{h} without the knowledge of the optimal solution z, we identify an upper bound $\hat{z}$ to MCRP that can be used to compute the duality gap (DG), that is, the difference between the upper bound of MCRP and the heuristic solution. Ensuring that the optimal solution is always bounded between the upper bound and the heuristic solution, we can use such information to infer the quality of the heuristic solution with respect to the unknown optimal solution. One typical upper bound for this purpose is the solution to the LP relaxation problem of MCRP. However, obtaining the LP relaxation bound can also be computationally challenging in large-scale instances. In what follows, we describe one upper bound metric that can be computed with a given set of parameters.

To find an upper bound $\hat{z}$ of MCRP, we begin by relaxing Constraints (2d). This decouples the coupling between stages. Also, this proves that any upper bound metric derived beyond this point will always satisfy $\hat{z} \geq z$, and hence proves the boundness of the optimum.

Solving the resource availability-relaxed MCRP can be still challenging because the problem now consists of N maximum coverage problems, which are shown to be NP-hard.[13,19] In what follows, we present a computationally-efficient upper bound metric.

We begin by examining Constraints (2d) for stage s and aggregating $t \in \mathcal{T}_s$ and $p \in \mathcal{P}$:

$$\sum_{p\in\mathcal{P}}\sum_{t\in\mathcal{T}_s}\frac{\pi_{tp}}{r_{tp}}\sum_{k\in\mathcal{K}}\sum_{j\in\mathcal{J}_s^k}\sum_{i\in\tilde{\mathcal{J}}_{s-1}^k} V_{tjp}\varphi_{ij} \geq \sum_{p\in\mathcal{P}}\sum_{t\in\mathcal{T}_s}\pi_{tp}y_{tp}$$

With this, we wish to maximize the left-hand side. We can do so by casting it as a maximization problem:

$$\hat{z}_s = \max_{x_j\in\{0,1\}, j\in\mathcal{J}_s^k, k\in\mathcal{K}} \left\{ \sum_{p\in\mathcal{P}}\sum_{t\in\mathcal{T}_s}\frac{\pi_{tp}}{r_{tp}}\sum_{k\in\mathcal{K}}\sum_{j\in\mathcal{J}_s^k} V_{tjp}x_j : \sum_{j\in\mathcal{J}_s^k} x_j = 1, k\in\mathcal{K} \right\}$$

where we use Eq. (1) to change variables. This problem can be further decomposed into K subproblems, each with the satellite index $k \in \mathcal{K}$ as a parameter:

$$\hat{z}_s^k = \max_{x_j\in\{0,1\}, j\in\mathcal{J}_s^k} \left\{ \sum_{p\in\mathcal{P}}\sum_{t\in\mathcal{T}_s}\frac{\pi_{tp}}{r_{tp}}\sum_{j\in\mathcal{J}_s^k} V_{tjp}x_j : \sum_{j\in\mathcal{J}_s^k} x_j = 1 \right\}$$

By aggregating $\hat{z}_s$ for all stages, we obtain an upper bound of MCRP:

$$\hat{z} = \sum_{s\in\mathcal{S}\setminus\{0\}} \hat{z}_s$$

COMPUTATIONAL EXPERIMENTS

In the first part of this section, we conduct computational experiments to evaluate the impact of the problem size on the quality of the heuristic solutions obtained by MPH and RHP compared to the baseline MCRP. All problems are solved using a commercial MILP solver. The second part of this section conducts a detailed analysis to investigate the impact of having multiple stages on the system observational throughput.

Comparative Analysis

Experimental Setup We have two sets of uniquely-generated test instances. Each set consists of 12 test instances, and each test instance draws one parameter value from each of the following sets: $N \in \{3, 4, 5\}$, $J \in \{50, 150\}$, and $K \in \{3, 5\}$. The smallest instance has at most 1.96×10^{15} feasible plans, and the largest instance has at most 2.53×10^{54} feasible plans. The goal is to investigate the impact of the problem size on the solution quality and the computational runtime. For each set of test instances, we apply a different parameter generation rule; no two instances with the same problem dimension have identical parameters.

For the first set, we set following parameter values. We assume a group of K homogeneous satellites in inclined circular orbits following the Walker-δ constellation pattern rule of $80° : K/K/0$. This indicates that each satellite occupies its own orbital plane and the relative phasing between satellites in adjacent orbital planes is zero. The altitude of the constellation system is randomly selected in the range $[700\,\text{km}, 2000\,\text{km}]$. The set of $P = 10$ spot targets are randomly generated in the latitude interval $[-80°, 80°]$ and no restriction is set on the longitudes; all spot targets are set with $\varepsilon_{\min} = 5°$. We assume $\pi_{tp} = 1, \forall t \in \mathcal{T}, p \in \mathcal{P}$ and $r_{tp} = 1, \forall t \in \mathcal{T}, p \in \mathcal{P}$. This models that all targets have identical weights for imaging. We let the resource availability to $c_{\max}^k = 600\,\mathrm{m/s}, \forall k \in \mathcal{K}$. The considered mission planning horizon is for $5\,\mathrm{days}$ and it is discretized with the time step size of $t_{\text{step}} = 100\,\mathrm{s}$; consequently, we have $T = 4320$. Each $\mathcal{J}_s^k = \{1, \ldots, J\}$ is generated such that each orbital slot is associated with identical orbital elements but different true anomaly; the true anomalies of orbital slots within the orbital plane of satellite k is uniformly spaced.

For the second set, we focus on analyzing the impacts of the spatiotemporally-varying observation rewards on the quality of solutions. We keep every parameter generation rules the same as the first set, but we vary the following parameter values. We assume that there are different sets of targets $\{\mathcal{P}_1, \ldots, \mathcal{P}_N\}$ that are valid during the period of a particular stage, and each target $p \in \mathcal{P}_s$ is associated with the temporally-varying reward following the rule:

$$\pi_{tp} = \begin{cases} \sim U(0,1), & \text{if } t \in \mathcal{T}_s \\ 0, & \text{otherwise} \end{cases}$$

where $U(0,1)$ is the uniform distribution in $[0,1]$. We set no observation rewards on targets beyond or before the periods of the assigned stages. We intend to simulate dynamically-changing environments such that a constellation configuration optimized for one stage would be drastically unfit for another. This setting is in contrast to that of the first set because all targets have uniform rewards throughout the entire mission planning horizon and no targets are dynamically generated or removed.

An arc (i,j) is associated with the cost of transfer c_{ij}. In our problem domain, the cost is the Δv required to transfer a satellite from one orbital slot to another. Given the many-maneuver opportunistic nature of multi-stage reconfiguration, it is logical to assume only co-planar maneuvers in this paper. More specifically, we restrict the co-planar maneuvers to phasings only. Out-of-plane impulsive maneuvers are especially costly in the LEO regime; therefore, such maneuvers are not ideal for a series of orbital transfers. We approximate the cost of transferring satellite k from orbital slot $i \in \mathcal{J}_{s-1}^k$ to orbital slot $j \in \mathcal{J}_s^k$ at stage s by taking these two nodes as the boundary conditions of a circular co-planar phasing problem as outlined in Reference 20.

We utilize a commercial software package, the Gurobi optimizer (version 9.1.1.), to solve MCRP and the subproblems of MPH and RHP. All computational experiments are coded and conducted on a platform with the Intel Core i-9700 3.00 GHz CPU processor (8 cores and 8 threads) and 32 GB of memory. In all cases, we allow the Gurobi optimizer to utilize all available cores. We use the default parameters of the Gurobi optimizer except that we impose the runtime limit of $3600\,\mathrm{s}$. No early termination is enforced on the heuristic methods even if the runtime aggregated thus far exceeds $3600\,\mathrm{s}$.

To gauge the quality of heuristic solutions relative to the MCRP solution obtained by the Gurobi optimizer, we define the *relative performance metric* (RP): $(z_{\text{h}} - z)/z_{\text{h}}$ unrestricted in sign where z_{h} denotes the generic heuristic solution objective function value. The positive sign of RP indicates the outperformance of a heuristic method relative to the Gurobi optimizer for MCRP. In cases where computing the optimal solutions of MCRP is computationally prohibitive, we can infer the quality of the heuristic solutions by computing the duality gap that bounds the optimal solution. We define DG as follows: $|\hat{z} - z_{\text{h}}|/z_{\text{h}}$. Note that the LP relaxation solution of MCRP can be used in place of $\hat{z}$, but quantifying it can be computationally challenging.

Numerical Results The top part of Table 1 (instances 1–12) reports the results of the computational experiments on the first set of test instances. Out of 12 MCRP instances, the Gurobi optimizer triggers the time limit of $3600\,\mathrm{s}$ for 9 instances and optimally solves 3 instances. For instance 11, the Gurobi optimizer forcefully terminates due to the out-of-memory issue. For the same instance, however, both MPH and RHP manage to obtain feasible solutions. In particular, MPH solves instance 11 with the duality gap of $2.98\,\%$ in $370.26\,\mathrm{s}$ whereas RHP solves the same instance with the duality gap of $11.27\,\%$ in $4770.89\,\mathrm{s}$ (the first stage subproblem triggers the time limit). Overall, solving MCRP, MPH, and RHP using the Gurobi optimizer led to finding 4, 4, and 6 best solutions, respectively (see boldface entries). In instances in which MPH underperform than MCRP, the relative under-performance is at most $0.76\,\%$. A similar analysis extends to RHP with the relative underperformance with at most $0.31\,\%$. RHP found more high-quality solutions than others, however, RHP comes at the price of additional computational runtime. For instances 11 and 12, which represent "large-scale" problems, RHP achieves solutions with relatively larger DG than other instances, and we can observe that MPH performs better than RHP with much lower DGs.

The bottom part of Table 1 (instances 13–24) reports the results of the computational experiments on the second set of test instances. Out of 12 instances, MCRP formulation led to finding the best solutions for 8 instances. As described previously, the test instances of the second set have drastic changes in the targets of interest that vary stage by stage. Therefore, the ability to concurrently optimize the entire stage is highly desired. It was already expected that MCRP would outperform others; we wish to see how MPH and RHP would perform in such scenarios by comparing the metrics such as RPs and DGs. MPH and RHP found 0 and 6 best solutions, respectively. In general, the duality gaps of heuristic methods are poorer than those we computed in the first set. Examining the relative performance metrics, we can see that the worst underperformance of MPH and RHP relative to MCRP are $4.68\,\%$ and $1.37\,\%$, respectively. These values are poorer than what we found in the first set of instances, however, we believe that these numbers indicate the high-quality solutions of the heuristic methods, especially considering the computational runtime required to produce such results. In cases we cannot quantify z, we need to resort to duality gaps to infer the solution quality. We observe that the metric $\hat{z}$ provides a good upper bound estimate of the optimum to MCRP.

Table 1: Results from the computational experiments. The boldface fonts indicate the maximum of z, z_{mph}, and z_{rhp}.

Instance				MCRP				MPH				RHP				No recon.
ID	J	N	K	z	Runtime,* s	DG†, %	$\hat{z}$	z_{mph}	Runtime, s	RP, %	DG, %	z_{rhp}	Runtime, s	RP, %	DG, %	
1	50	3	3	**12,018**	1,835.34	0.00	12,159	**12,018**	2.46	0.00	1.17	11,981	52.08	-0.31	1.49	10,213
2	50	3	5	13,582	-	20.00	16,473	15,871	64.53	14.42	3.79	**15,947**	3,608.29‡	14.83	3.30	12,117
3	50	4	3	**7,328**	200.54	0.00	7,440	7,273	0.36	-0.76	2.30	7,307	16.21	-0.29	1.82	5,745
4	50	5	3	**12,477**	-	0.27	12,591	12,444	1.62	-0.27	1.18	12,476	32.13	-0.01	0.92	8,852
5	50	4	5	13,473	-	18.04	16,108	15,239	107.59	11.59	5.70	**15,249**	3,754.84‡	11.65	5.63	12,224
6	150	3	3	**5,408**	1,103.81	0.00	5,447	5,371	3.78	-0.69	1.42	**5,408**	86.47	0.00	0.72	4,713
7	50	5	5	16,043	-	26.47	20,643	19,802	64.42	18.98	4.25	**19,803**	3,625.95‡	18.99	4.24	12,198
8	150	3	5	7,730	-	2.04	8,003	**7,832**	27.73	1.30	2.18	7,746	3,634.44‡	0.21	3.32	6,597
9	150	4	3	9,391	-	5.44	9,991	9,836	20.00	4.52	1.58	**9,840**	466.63	4.56	1.53	8,066
10	150	5	3	10,775	-	0.87	11,015	10,808	17.66	0.31	1.92	**10,823**	328.39	0.44	1.77	8,331
11§	150	4	5	N/A	-	N/A	24,050	**23,354**	370.26	N/A	2.98	21,615	4,770.89‡	N/A	11.27	13,878
12	150	5	5	16,118	-	167.96	20,526	**19,659**	1,573.23	18.01	4.41	18,187	3,667.17‡	11.38	12.86	12,989
13	50	3	3	**2,458.23**	693.75	0.00	2,539.16	2,428.22	1.56	-1.24	4.57	**2,458.23**	25.41	0.00	3.29	1,758.90
14	50	3	5	3,357.83	-	7.63	3,744.99	3,343.33	5.94	-0.43	12.01	**3,456.53**	3,732.53‡	2.86	8.35	2,378.22
15	50	4	3	**2,172.80**	-	1.09	2,266.15	2,162.64	0.55	-0.47	4.79	2,157.82	19.36	-0.69	5.02	1,620.92
16	50	5	3	**582.51**	7.68	0.00	607.38	556.49	0.12	-4.68	9.14	**582.51**	2.56	0.00	4.27	490.52
17	50	4	5	**1,918.26**	-	0.61	2,024.75	1,852.82	1.43	-3.53	9.28	1,915.98	18.34	-0.12	5.68	1,202.84
18	150	3	3	**2,059.32**	3,341.73	0.00	2,126.02	2,022.12	2.74	-1.84	5.14	2,044.69	274.94	-0.72	3.98	1,636.00
19	50	5	5	**1,229.16**	-	2.71	1,342.20	1,188.32	0.30	-3.44	12.95	1,212.52	13.19	-1.37	10.69	1,037.60
20	150	3	5	2,888.41	-	150.33	3,902.66	3,579.20	31.13	19.30	9.04	**3,676.89**	5,001.70‡	21.44	6.14	2,492.79
21	150	4	3	**1,388.89**	1,915.90	0.00	1,425.62	1,365.13	1.59	-1.74	4.43	1,384.16	117.55	-0.34	3.00	1,113.82
22	150	5	3	**1,446.07**	-	0.50	1,508.76	1,421.57	0.92	-1.72	6.13	1,435.62	113.91	-0.73	5.09	1,040.36
23	150	4	5	2,334.46	-	130.72	3,424.50	3,096.22	5.89	24.60	10.60	**3,116.20**	1,609.56	25.09	9.89	1,671.40
24	150	5	5	1,531.05	-	181.38	1,945.27	1,699.07	2.24	9.89	14.49	**1,744.20**	2,104.79	12.22	11.53	1,382.86

* Hyphen (-) indicates the trigger of the time limit of 3600 s.
† Duality gap (MIPGap) computed internally by the Gurobi optimizer; the default optimality tolerance is 0.01 %.
‡ Trigger of the time limit during the first-stage subproblem.
§ Gurobi optimizer runs out of memory and terminates (Gurobi error 10001: Out of memory).

Case Study: Impacts of Stages and Solution Methods on System Observational Throughput

In this section, we conduct a case study to analyze the impact of having multiple stages on the system observation throughput. To do so, we vary $N \in \{1, 2, 3, 4, 5, 6\}$ on an identical problem with fixed parameters.

Case Study Setup Following are the fixed parameters:

- **Mission.** Mission planning horizon is referenced to the J2000 epoch, and the duration is 5 days. We set $t_{\text{step}} = 100\,\text{s}$ and $T = 4320$.
- **Satellites.** A fleet of 5 heterogeneous satellites with different orbits (but all are circular) and fuel availabilities. Each satellite has 50 candidate orbital slots that are uniformly distributed within the orbital plane. Table 2 shows the key specifications of satellites.

Table 2: Key satellite specification parameters.

k	Altitude, km	Inclination, deg.	RAAN, deg.	$c^k_{\max}$, m/s
1	926.16	85.02	196.82	687.40
2	787.89	71.28	159.59	587.85
3	724.69	78.64	12.98	796.70
4	846.24	77.28	296.24	696.53
5	733.40	73.04	98.39	701.13

- **Targets.** A set of 10 targets randomly distributed between the latitude interval $[-80°, 80°]$ and the longitude interval $[-180°, 180°]$. Figure 4 visualizes the coordinates of the targets. The rewards are target-dependent. Each target p has the time-dependent reward $\boldsymbol{\pi}_p$ where we assume $\pi_{tp} \in \{0, 1\}$; we randomly generate a random number of blocks of contiguous ones. Figure 5 visualizes the observation rewards imposed on each target. Also, we let $\varepsilon_{\min} = 5°$ and $r_{tp} = 1, \forall t \in \mathcal{T}, p \in \mathcal{P}$.

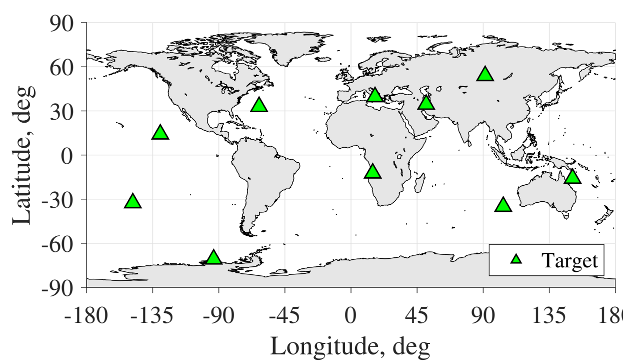


Figure 4: Coordinates of the randomly generated targets.

Numerical Results All instances are solved by MCRP, MPH (for $s \geq 2$), and RHP (for $s \geq 3$), and the Gurobi optimizer solves MCRP and the subproblems of MPH and RHP. We set the runtime limit of $3600\,\text{s}$. Figure 6 reports the results of the total observation rewards obtained by all methods per stage. The bottom dashed line shows the reference case without any reconfiguration, which has

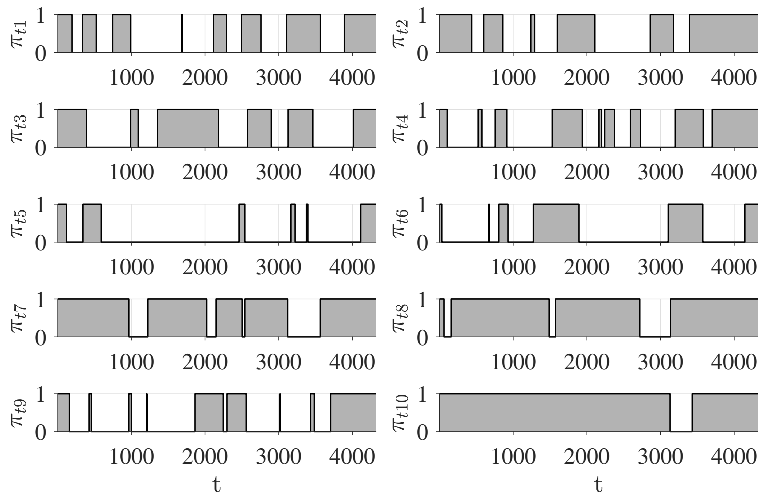


Figure 5: Observation rewards per target. The gray shaded areas refer to ones.

the score of 4054. By employing reconfiguration, even with only a single stage, the system enhances the observational throughput.

The results conform with our intuitive expectation that, in general, increasing the number of stages increases the total observation rewards by allowing more degrees of freedom for flexibility. (This is not always true because we have resource availability that complicates the problem of multi-stage reconfiguration.) However, in a case with $s = 3$, both RHP and MPH outperform MCRP with the scores of $z_{\text{rhp}} = 4329$, $z_{\text{mph}} = 4291$, and $z = 4278$, respectively. This result is primarily due to the fact that the Gurobi optimizer terminates MCRP early due to the trigger of the runtime limit of $3600\,\text{s}$ and returns the best feasible incumbent solution found thus far. Furthermore, this result implies that unless we solve MCRP to optimality (or with more runtime limit), the heuristic methods can achieve better solutions. To validate this remark, we test again case $s = 3$ of MCRP with a longer runtime limit of $7200\,\text{s}$. In this setting, we obtain $z = 4302$ with the duality gap of $2.09\,\%$ that triggers the new runtime limit (hence the solution is still suboptimal); however, the obtained value is greater than the previous one with the runtime limit of $3600\,\text{s}$ and those of the heuristic methods. The worst underperformances relative to MCRP are at most $3.62\,\%$ for MPH (case $s = 5$) and $2.26\,\%$ for RHP (case $s = 5$). These values attest to the high-quality solutions of the heuristic methods.

We also report the computational runtimes for each instance. We observe that MCRP triggers the runtime limit for $s \geq 2$ cases owing to the problem scales while no heuristic methods trigger the runtime limit. MPH maintains its runtime less than $14\,\text{s}$ (case $s = 2$) for all $s \geq 2$ cases, which is significantly faster than MCRP.

CONCLUSIONS

This paper proposes a novel mathematical model to solve the problem of reconfiguring a fleet of satellites to maximize the observation rewards obtained by covering a set of targets over the mission planning horizon while satisfying the individual resource availability constraints. To model stage transitions and the fuel consumption by satellites, we adopt the concept of time-expanded graphs by expanding the nodes (the orbital slots) forward in time and constructing weighted arcs

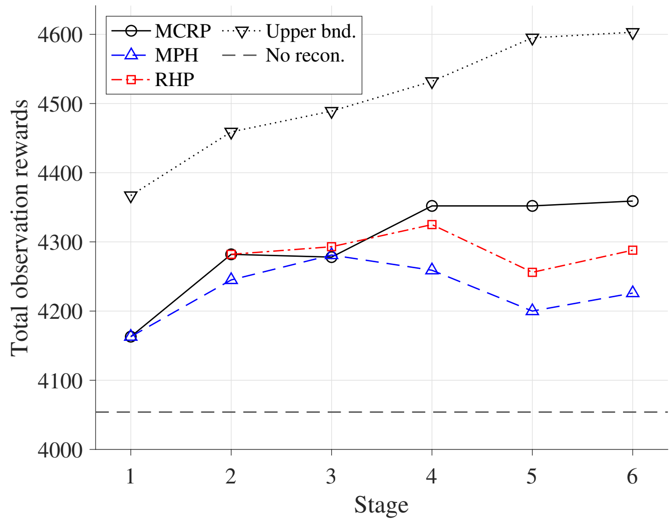


Figure 6: Total observation rewards by varying the number of stages.

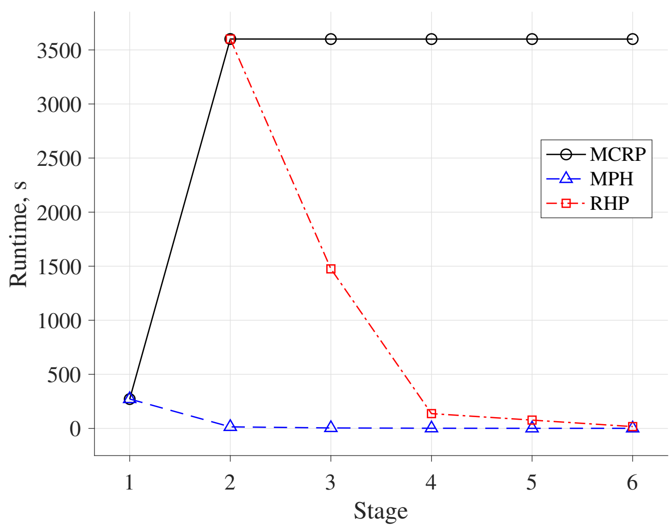


Figure 7: Computational runtimes by varying the number of stages.

between the nodes of any two adjacent stages. Based on this model, we propose an ILP formulation, which enables the use of commercial MILP solvers for convenience-handling and provably-optimal solutions.

To address the issue of the computational intractability in solving large-scale MCRP, we propose the use of two sequential decision-making approaches: the myopic policy and the rolling horizon procedure. Through the computational experiments, we empirically show that both MPH and RHP provide high-quality solutions in a reasonable computational runtime for instances with uniform observation rewards. In the case of spatiotemporally-varying observation rewards and dynamically-generated targets, RHP outperforms MPH by making informed decisions exploiting the deterministic forecast of the impact of current-stage decisions on an immediate subsequent stage. However, there is no guaranteed outperformance of RHP relative to MPH as the remaining periods are not fully taken into account in the decision-making. We show in the case study that having more stages can increase the total observation rewards, but the quality of actual solutions depends on the performance of an algorithm.

There are two fruitful directions for future research. The first is to improve the fidelity and the applicability of the model. This paper only considers an ideal case with orbital transfers as decision variables. To accurately assess the impact of multi-stage reconfiguration on the scheduling of Earth observation systems, the proposed problem ought to be integrated with a scheduler with various satellite tasks and operational constraints. Such integration would require modeling efforts in the mapping of complex interactions between various tasks and requirements. The second is to challenge the curse of dimensionality. As shown previously, SP(s) of MPH is RCRP-IRC of Reference 13, which itself is a combinatorial optimization problem that suffers from the explosion of a solution space in largely-sized instances. Several algorithmic efforts such as asymptotic analysis, node aggregation, approximate dynamic programming, and decomposition-based techniques can reduce the time complexity.

ACKNOWLEDGMENT

This material is based upon work supported by the National Science Foundation Graduate Research Fellowship Program under Grant No. DGE-2039655. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

[1] Y. Chen, V. Mahalec, Y. Chen, X. Liu, R. He, and K. Sun, “Reconfiguration of satellite orbit for cooperative observation using variable-size multi-objective differential evolution,” *European Journal of Operational Research*, Vol. 242, No. 1, 2015, pp. 10–20, 10.1016/j.ejor.2014.09.025.

[2] S. W. Paek, S. Kim, and O. d. Weck, “Optimization of Reconfigurable Satellite Constellations Using Simulated Annealing and Genetic Algorithm,” *Sensors*, Vol. 19, No. 4, 2019, 10.3390/s19040765.

[3] X. He, H. Li, L. Yang, and J. Zhao, “Reconfigurable Satellite Constellation Design for Disaster Monitoring Using Physical Programming,” *International Journal of Aerospace Engineering*, Vol. 2020, 2020, p. 8813685, 10.1155/2020/8813685.

[4] O. L. de Weck, U. Scialom, and A. Siddiqi, “Optimal reconfiguration of satellite constellations with the auction algorithm,” *Acta Astronautica*, Vol. 62, No. 2, 2008, pp. 112–130, 10.1016/j.actaastro.2007.02.008.

[5] M. P. Ferringer, D. B. Spencer, and P. Reed, “Many-objective reconfiguration of operational satellite constellations with the Large-Cluster Epsilon Non-dominated Sorting Genetic Algorithm-II,” *2009 IEEE Congress on Evolutionary Computation*, 2009, pp. 340–349, 10.1109/CEC.2009.4982967.

[6] P. Wang, G. Reinelt, P. Gao, and Y. Tan, "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. Combinatorial Optimizatiion in Industrial Engineering, https://doi.org/10.1016/j.cie.2011.02.015.

[7] W. J. Wolfe and S. E. Sorensen, "Three Scheduling Algorithms Applied to the Earth Observing Systems Domain," *Management Science*, Vol. 46, No. 1, 2000, pp. 148–166, 10.1287/mnsc.46.1.148.15134.

[8] 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, 2021, pp. 3881–3892, 10.1109/JSYST.2020.2997050.

[9] J. Kim and J. Ahn, "Integrated Framework for Task Scheduling and Attitude Control of Multiple Agile Satellites," *Journal of Aerospace Information Systems*, Vol. 18, No. 8, 2021, pp. 539–552, 10.2514/1.I010910.

[10] H. Lee and K. Ho, "Binary Integer Linear Programming Formulation for Optimal Satellite Constellation Reconfiguration," *AAS/AIAA Astrodynamics Specialist Conference*, 08 2020.

[11] H. Lee and K. Ho, "A Lagrangian Relaxation-Based Heuristic Approach to Regional Constellation Reconfiguration Problem," *AAS/AIAA Astrodynamics Specialist Conference*, 08 2021.

[12] T. S. d. Jonchay, H. Chen, O. Gunasekara, and K. Ho, "Framework for Modeling and Optimization of On-Orbit Servicing Operations Under Demand Uncertainties," *Journal of Spacecraft and Rockets*, Vol. 58, No. 4, 2021, pp. 1157–1173, 10.2514/1.A34978.

[13] H. Lee and K. Ho, "Regional Constellation Reconfiguration Problem: Integer Linear Programming Formulation and Lagrangian Heuristic Method," 2022, 10.48550/ARXIV.2206.02910.

[14] G. Denis, H. de Boissezon, S. Hosford, X. Pasco, B. Montfort, and F. Ranera, "The evolution of Earth Observation satellites in Europe and its impact on the performance of emergency response services," *Acta Astronautica*, Vol. 127, 2016, pp. 619–633, 10.1016/j.actaastro.2016.06.012.

[15] S. Voigt, F. Giulio-Tonolo, J. Lyons, J. Kučera, B. Jones, T. Schneiderhan, G. Platzeck, K. Kaku, M. K. Hazarika, L. Czaran, S. Li, W. Pedersen, G. K. James, C. Proy, D. M. Muthike, J. Bequignon, and D. Guha-Sapir, "Global trends in satellite-based emergency mapping," *Science*, Vol. 353, No. 6296, 2016, pp. 247–252, 10.1126/science.aad8728.

[16] H. Lee, S. Shimizu, S. Yoshikawa, and K. Ho, "Satellite Constellation Pattern Optimization for Complex Regional Coverage," *Journal of Spacecraft and Rockets*, Vol. 57, No. 6, 2020, pp. 1309–1327, 10.2514/1.A34657.

[17] W. B. Powell, "What you should know about approximate dynamic programming," *Naval Research Logistics (NRL)*, Vol. 56, No. 3, 2009, pp. 239–249, https://doi.org/10.1002/nav.20347.

[18] S. Sethi and G. Sorger, "A theory of rolling horizon decision making," *Annals of Operations Research*, Vol. 29, No. 1, 1991, pp. 387–415, 10.1007/BF02283607.

[19] N. Megiddo, E. Zemel, and S. L. Hakimi, "The Maximum Coverage Location Problem," *SIAM Journal on Algebraic Discrete Methods*, Vol. 4, No. 2, 1983, pp. 253–261, 10.1137/0604028.

[20] D. Vallado, *Fundamentals of Astrodynamics and Applications*. Space technology library, Microcosm Press, 2013.