AIAA Aviation Forum and ASCEND co-located Conference Proceedings
29 July - 2 August 2024, Las Vegas, Nevada
AIAA AVIATION FORUM AND ASCEND 2024
10.2514/6.2024-4816

Integrated Space Mission Planning Under Uncertainty via Stochastic and Decomposition-Based Optimization

Masafumi Isaji* and Koki Ho†
Georgia Institute of Technology, Atlanta, GA, 30332, USA

As we strive to establish a long-term presence in space, it is crucial to plan large-scale space missions and campaigns with future uncertainties in mind. However, integrated space mission planning, which simultaneously considers mission planning and spacecraft design, faces significant challenges when dealing with uncertainties; this problem is formulated as a stochastic mixed integer nonlinear program (MINLP), and solving it using the conventional method would be computationally prohibitive for realistic applications. Extending a deterministic decomposition method from our previous work, we propose a novel and computationally efficient approach for integrated space mission planning under uncertainty. The proposed method effectively combines the Alternating Direction Method of Multipliers (ADMM)-based decomposition framework from our previous work, robust optimization, and two-stage stochastic programming (TSSP). This hybrid approach first solves the integrated problem deterministically, assuming the worst scenario, to precompute the robust spacecraft design. Subsequently, the two-stage stochastic program is solved for mission planning, effectively transforming the problem into a more manageable mixed-integer linear program (MILP). This approach significantly reduces computational costs compared to the exact method, but may potentially miss solutions that the exact method might find. We examine this balance through a case study of staged infrastructure deployment on the lunar surface under future demand uncertainty. When comparing the proposed method with a fully coupled benchmark, the results indicate that our approach can achieve nearly identical objective values (no worse than 1% in solved problems) while drastically reducing computational costs.

Nomenclature

$\mathcal{A}$	=	Set of arcs
$\boldsymbol{c}_{vijt}$	=	Commodity cost coefficient vector of
c'_{vijt}	=	Spacecraft Cost coefficient
C_c	=	Set of continuous commodity flow variables
C_d	=	Set of discrete commodity flow variables
$\boldsymbol{d}_{it}^{\tilde{\omega}}$	=	Demand vector
$\mathbb{E}$	=	Expectation
$\mathcal{F}$	=	Spacecraft sizing function
H_{vij}	=	Concurrency matrix
m_d	=	Spacecraft dry mass
Γ_f	=	Spacecraft propellant capacity
Γ_p	=	Spacecraft payload capacity
$\mathcal{N}$	=	Set of nodes
$\mathbb{P}$	=	Probability
T_{vijt}	=	Commodity transformation matrix
$\mathcal{T}$	=	Set of time steps
Δt_{ij}	=	Time of Flight (ToF)

*Ph.D. Student, Daniel Guggenheim School of Aerospace Engineering, Atlanta, GA, AIAA Student Member.
†Associate Professor, Daniel Guggenheim School of Aerospace Engineering, Atlanta, GA, AIAA Senior Member.

$u^{\tilde{\omega}}_{vijt}$ = Spacecraft flow variable
$\mathcal{V}$ = Set of spacecraft
W_{ij} = Launch time window
$\boldsymbol{x}^{\tilde{\omega}}_{vijt}$ = Commodity flow variable
$\boldsymbol{w}$ = Penalty weight
γ = Weight update parameter
$\boldsymbol{\delta}$ = Discrepancy
η = Number of all commodity types
$\boldsymbol{\lambda}$ = Lagrange multiplier estimate
ϕ = Penalty function
$\boldsymbol{\omega}$ = Random parameter
$\tilde{\omega}$ = Random parameter realization
Ω = Support of random parameter

Indices
i = Departure Node index
j = Arrival node index
k = Vector element index
ℓ = Subsystem index
s = Scenario or realization index
t = Time index
v = Vehicle index
ζ = Iteration count

I. Introduction

As we strive to establish a long-term presence in space, it is essential to be prepared for future uncertainties and to plan large-scale space missions and campaigns accordingly. At the same time, since mission planning, spacecraft design, and potentially other elements are tightly coupled, they should be considered simultaneously [1]; we refer to this problem as integrated space mission planning. Whereas more effective decisions can be made by combining these two imperative aspects, it requires solving the integrated space mission planning under uncertainty. Our previous work [2] has established an effective decomposition method for the deterministic variant of the same problem. As such, this paper focuses on addressing the stochastic nature of the problem and extends the approach to solve integrated space mission planning under uncertainty.

One common way to handle uncertainties, especially in the space industry, is to assume the worst-case scenario; however, it can lead to prohibitive costs if other, less severe scenarios unfold [3]. Hence, a better strategy is to make flexible and uncertainty-aware decisions now and adapt to the realization in the future, while still ensuring the worst-case scenario can be managed. In the field of stochastic programming, such a way of optimizing current decisions is known as the Two-Stage Stochastic Program (TSSP) [4]. However, naively incorporating TSSP into the existing framework results in a stochastic Mixed-Integer NonLinear Program (MINLP), which is a highly costly or impossible problem to solve.

There exist studies tackling decision-making under uncertainty in various subfields of space logistics: on-orbit servicing [5, 6], satellite constellation deployment [7], and campaign scheduling under stochastic launch delay [8], to name a few. Although they successfully handle uncertainties in their respective ways, they rely on Mixed-Integer Linear Programming (MILP) formulations and thus are not applicable to our problem. Work in Ref. [9] addresses a similar problem from a policy optimization perspective using hierarchical reinforcement learning. However, as with any learning-based method, it suffers from the irreproducibility of solutions, the need for extensive hyperparameter tuning, and high computational expense for training.

To solve the problem of integrated space mission planning under uncertainty, we propose a novel and computationally scalable approach that combines the decomposition method from our previous work, robust (i.e., worst-case) optimization, and TSSP for MILP. These "building blocks" are detailed in Section III. Based on the concepts of *hard constraints* in robust optimization [10], we hypothesize that solving the problem of integrated space mission planning under uncertainty will produce the same or similar spacecraft design as in the worst case. Assuming the hypothesis holds, we first solve

the deterministic robust optimization, which can be solved efficiently by our decomposition method. Then, the TSSP for mission planning is solved with robust spacecraft design; that is, the mission planning and spacecraft design are no longer coupled. Exploiting a unique structure of the integrated mission planning problem, where nonlinearity only exists in sizing constraints for the spacecraft (or other components), we can formulate the TSSP as a MILP. The proposed approach reduces computational cost in exchange for a partial coupling between space mission planning and spacecraft design. To study the trade-off, we compared the proposed approach against a fully coupled benchmark method, which is a straightforward integration of TSSP with the decomposition framework. The results show that the objective values of the proposed and benchmark methods are closely matched or identical for every case study problem while achieving marginally lower computational expense.

The remainder of this paper is structured as follows. In Section II, we define the problem of interest, including detailed definitions of the objective, constraints, variables, and parameters of the problem. Then Section III describes the main challenges in solving the defined problem and introduces the optimization concepts and methods used to address them. The proposed approach is presented and discussed in Section IV, along with its advantages, limitations, and underlying assumptions. The approach is then tested through case studies in Section V. A staged deployment of infrastructure on the lunar surface under future demand uncertainty is considered. We define a benchmark method and compare the results of the two in several case study settings. Finally, the conclusion is stated in Section VI.

II. Problem Definition: Integrated Space Mission Planning Under Uncertainty

We consider simultaneous optimization of two-mission space campaign planning and spacecraft design under uncertain future demand. The optimization formulation and its definitions of variables and parameters are detailed in this section. Table 1 presents definitions for variables, parameters, and sets used in the formulation in (1). The integrated space mission planning problem aims to optimize commodity (e.g., crew, propellant) and spacecraft flow over a space logistics network composed of nodes (e.g., lunar surface) and arcs (e.g., trajectory from low-Earth orbit to low-lunar orbit) to minimize the total campaign cost. In addition, the spacecraft to be used in the campaign is designed concurrently. To consider uncertainty in this already complex problem, we make several important assumptions. First, if we let $\boldsymbol{\omega}$ be the vector of uncertain parameters, we assume that its support $\Omega = \{\omega_1, \omega_2, \dots\}$ is known a priori and finite. Namely, we know a finite number of possible scenarios that can happen in the future, while their realization is probabilistic. If the uncertainty follows a continuous distribution, it must be approximated as a discrete distribution using methods such as the average sample approximation [11]. Secondly, we only consider uncertainties in future demand; other types of uncertainty, such as model uncertainties, are beyond the scope of this work. In a more specific context, we assume that the mission requirements are known for the first (current) mission but not for the second (future) mission. Finally, we assume that the spacecraft can be reused but cannot be modified (e.g., expanding capacities) once it is built for the first mission. Based on these assumptions and definitions in Table 1, we can formulate the optimization problem as a TSSP as follows.

$$\min \quad \sum_{t\in\mathcal{T}_1}\sum_{(v,i,j)\in\mathcal{A}} \left(\boldsymbol{c}_{vijt}{}^{\top} \boldsymbol{x}^{\omega_1}_{vijt} + c'_{vijt} m_{d_v} u^{\omega_1}_{vijt} \right) + \mathbb{E}\left[\sum_{t\in\mathcal{T}_2}\sum_{(v,i,j)\in\mathcal{A}} \left(\boldsymbol{c}_{vijt}{}^{\top} \boldsymbol{x}^{\omega}_{vijt} + c'_{vijt} m_{d_v} u^{\omega}_{vijt} \right) \right] \tag{1a}$$

s.t.

$$\sum_{(v,j):(v,i,j)\in\mathcal{A}} \begin{bmatrix} \boldsymbol{x}^{\tilde{\omega}}_{vijt} \\ m_{d_v} u^{\tilde{\omega}}_{vijt} \end{bmatrix} - \sum_{(v,j):(v,i,j)\in\mathcal{A}} T_{vjit} \begin{bmatrix} \boldsymbol{x}^{\tilde{\omega}}_{vji(t-\Delta t_{ji})} \\ m_{d_v} u^{\tilde{\omega}}_{vji(t-\Delta t_{ji})} \end{bmatrix} \le \boldsymbol{d}^{\tilde{\omega}}_{it} \qquad \forall i \in \mathcal{N},\ \forall t \in \mathcal{T},\ \forall \tilde{\omega} \in \Omega \tag{1b}$$

$$H_{vij} \boldsymbol{x}^{\tilde{\omega}}_{vijt} \le \begin{bmatrix} \Gamma_{p_v} \\ \Gamma_{f_v} \end{bmatrix} u^{\tilde{\omega}}_{vijt} \qquad \forall (v,i,j) \in \mathcal{A},\ \forall t \in \mathcal{T},\ \forall \tilde{\omega} \in \Omega \tag{1c}$$

$$\begin{cases} \boldsymbol{x}^{\tilde{\omega}}_{vijt} \ge \boldsymbol{0} & \text{if } t \in W_{ij} \\ \boldsymbol{x}^{\tilde{\omega}}_{vijt} = \boldsymbol{0} & \text{otherwise} \end{cases} \qquad \forall (v,i,j) \in \mathcal{A},\ \forall t \in \mathcal{T},\ \forall \tilde{\omega} \in \Omega \tag{1d}$$

$$m_{d_v} = \mathcal{F}(\Gamma_{p_v}, \Gamma_{f_v}) \qquad \forall v \in \mathcal{V} \tag{1e}$$

$$\begin{aligned} \boldsymbol{x}^{\omega_1}_{vijt} &= \cdots = \boldsymbol{x}^{\omega_{|\Omega|}}_{vijt} \\ u^{\omega_1}_{vijt} &= \cdots = u^{\omega_{|\Omega|}}_{vijt} \end{aligned} \qquad \forall t \in \mathcal{T}_1 \tag{1f}$$

Table 1 Variables, parameters, and sets definitions

Name	Description	Domain
Variables		
$\boldsymbol{x}_{vijt}^{\tilde{\omega}}$	Commodity flow variable, or the quantity of the commodity delivered from node i to j at time t by spacecraft v for scenario realization $\tilde{\omega}$. Each commodity is in the continuous commodity set (C_c) or the discrete commodity set (C_d).	$\mathbb{R}_{\geq 0}^{\lvert C_c\rvert} \times \mathbb{Z}_{\geq 0}^{\lvert C_d\rvert}$
$u_{vijt}^{\tilde{\omega}}$	Spacecraft flow variable, which indicates the number of spacecraft type v flying from node i to j at time t for scenario realization $\tilde{\omega}$.	$\mathbb{Z}_{\geq 0}$
Γ_{p_v}	Payload capacity of spacecraft v.	$\mathbb{R}_{\geq 0}$
Γ_{f_v}	Propellant capacity of spacecraft v.	$\mathbb{R}_{\geq 0}$
m_{d_v}	Dry mass of spacecraft v.	$\mathbb{R}_{\geq 0}$
Parameters		
$\boldsymbol{c}_{vijt}$	Commodity cost coefficient where the number of the commodities is $\eta = \lvert C_c\rvert + \lvert C_d\rvert$.	$\mathbb{R}_{\geq 0}^{\eta}$
c'_{vijt}	Spacecraft cost coefficient.	$\mathbb{R}_{\geq 0}$
$\boldsymbol{d}_{it}^{\tilde{\omega}}$	Demand/supply vector of different commodities and spacecraft at node i at time t for scenario realization $\tilde{\omega}$.	$\mathbb{R}^{\eta+1}$
T_{vijt}	Commodity transformation matrix.	$\mathbb{R}^{(\eta+1)\times(\eta+1)}$
H_{vij}	Concurrency constraint matrix.	$\mathbb{R}^{2\times\eta}$
W_{ij}	Launch window vector including the allowed time steps from node i to j.	$\mathbb{R}^{\lvert\mathcal{T}\rvert}$
Δt_{ij}	Time of Flight (ToF) from node i to j.	$\mathbb{R}_{\geq 0}$
Sets		
$\mathcal{A}$	Set of arcs realized by spacecraft.	
C_c	Set of continuous commodity flow variables.	
C_d	Set of discrete commodity flow variables.	
$\mathcal{N}$	Set of nodes.	
$\mathcal{T}$	Set of time steps for all missions.	
$\mathcal{T}_1, \mathcal{T}_2$	Set of time steps for the first mission ($\mathcal{T}_1$) and the second mission ($\mathcal{T}_2$).	
$\mathcal{V}$	Set of spacecraft (vehicles).	
Ω	Support of random parameter $\boldsymbol{\omega}$.	

The objective of the problem is defined in (1a). The design variables are scaled linearly with cost coefficients; this type of objective is known as Initial Mass at Low-Earth Orbit (IMLEO) and is widely adopted in Space Logistics literature (e.g., Refs [2, 12–14]). The first term is the mass launched to LEO for the first mission (note that it is summed over the first mission time steps $\mathcal{T}_1$), and the second term is the *expectation* of the second mission launch mass. The second term differs from IMLEO in the mentioned literature and is unique to the TSSP formulation. Such an objective definition aims to optimize here-and-now (first mission) decisions while accounting for the uncertainties in the future (second mission). The solution of the TSSP formulation allows for more flexibility and adaptability to future scenario realizations, and the second-stage cost inside the expectation is termed the recourse cost (or function) because it measures the cost to adjust to a certain realization [4]. As we assume finite support Ω, the following is the equivalent expression:

$$\min \quad \sum_{t\in\mathcal{T}_1} \sum_{(v,i,j)\in\mathcal{A}} \left(\boldsymbol{c}_{vijt}{}^{\top} \boldsymbol{x}_{vijt}^{\omega_1} + c'_{vijt} m_{d_v} u_{vijt}^{\omega_1} \right) + \sum_{\tilde{\omega}\in\Omega} \mathbb{P}\{\boldsymbol{\omega} = \tilde{\omega}\} \left[\sum_{t\in\mathcal{T}_2} \sum_{(v,i,j)\in\mathcal{A}} \left(\boldsymbol{c}_{vijt}{}^{\top} \boldsymbol{x}_{vijt}^{\tilde{\omega}} + c'_{vijt} m_{d_v} u_{vijt}^{\tilde{\omega}} \right) \right] \tag{2}$$

where $\tilde{\omega}$ is an arbitrary realization of $\boldsymbol{\omega}$. If we additionally assume the equal probability for all scenarios, it simplifies to

$$\min \quad \sum_{t\in\mathcal{T}_1} \sum_{(v,i,j)\in\mathcal{A}} \left(\boldsymbol{c}_{vijt}{}^{\top} \boldsymbol{x}_{vijt}^{\omega_1} + c'_{vijt} m_{d_v} u_{vijt}^{\omega_1} \right) + \sum_{\tilde{\omega}\in\Omega} \sum_{t\in\mathcal{T}_2} \sum_{(v,i,j)\in\mathcal{A}} \frac{1}{\lvert\Omega\rvert} \left(\boldsymbol{c}_{vijt}{}^{\top} \boldsymbol{x}_{vijt}^{\tilde{\omega}} + c'_{vijt} m_{d_v} u_{vijt}^{\tilde{\omega}} \right) \tag{3}$$

The set of constraints in (1b) to (1f) formulates the flows of commodities and spacecraft over a space logistics network. Inequality (1b) is the mass balance or conservation constraint that ensures that the demand or supply for commodities is satisfied for all nodes, time steps, and scenarios. It also captures the consumption of commodities, such as propellant usage or crews' consumption. In addition to the said transformation, the conservation of commodities and spacecraft is expressed via the commodity transportation matrix, denoted by T_{vjit}. The order of the nodes (that is, ij vs. ji) is reversed here to represent the inflow and outflow of commodities. Furthermore, to guarantee the capacity limit of the spacecraft, we impose the so-called concurrency constraint expressed in inequality (1c). The concurrency constraint matrix H_{vij} is used, for example, to sum up the mass of different types of payload to compare with the payload capacity of the spacecraft Γ_{p_v}. The expression in (1d) is the time window constraint, stating that commodities can only flow over arc (i, j) in the allowed time steps. Equation (1e) is the spacecraft sizing constraint, where the dry mass of spacecraft v is a nonlinear and potentially black-box function of its payload capacity, propellant capacity, and some other parameters that are not variables in optimization. See Eq. (15) in the appendix for details. Lastly, Eq. (1f) is called the nonanticipativity constraint in stochastic programming, stating that decisions should only depend on the information available at the time of decision-making, not future uncertainty realizations. In this problem, we force all the first mission variables to be identical (notice $\mathcal{T}_1$ is the set of the first mission time steps). It does not apply to variables that are independent of realization $\tilde{\omega}$, such as spacecraft capacities. The resulting optimization problem cannot be solved in a straightforward manner. As such, we introduce and review some useful concepts and methods that we employ to tackle the difficulties in the next section.

III. Optimization Background

The major difficulties in problem formulation (1) are twofold: the nonlinear and nonconvex spacecraft sizing constraint in Eq. (1e) and the computational expense caused by the introduction of variables for each realization $\tilde{\omega}$. We cover how to address them in their respective subsections in the following.

A. Deterministic ADMM-Based Decomposition Method

The deterministic simultaneous optimization of space mission planning and spacecraft sizing has been studied in our previous work [2]. We only give a brief overview of the method here; see [2] for details. To define the deterministic formulation, we can simply consider a single scenario, say $\boldsymbol{\omega}_1$, with probability 1 (i.e., $\Omega = \{\boldsymbol{\omega}_1\}$). Consequently,

$$\begin{aligned} \min \quad & \sum_{t\in\mathcal{T}} \sum_{(v,i,j)\in\mathcal{A}} \left(\boldsymbol{c}_{vijt}{}^{\top} \boldsymbol{x}_{vijt}^{\boldsymbol{\omega}_1} + c'_{vijt} m_{d_v} u_{vijt}^{\boldsymbol{\omega}_1} \right) \\ \text{s.t.} \quad & \text{(1b) to (1e)} \\ & \mathbb{P}\{\boldsymbol{\omega} = \boldsymbol{\omega}_1\} = 1 \end{aligned} \tag{4}$$

is the deterministic formulation. Although there is no stochasticity here, it is still a highly challenging problem due to the co-existence of discrete variables and nonconvex constraint in Eq. (1e), which results in an MINLP. This problem cannot be solved as is even by the state-of-the-art MINLP solvers, such as Baron [15], because of the black-box or implicit nonconvex constraint in Eq. (1e). Our approach avoids directly solving the nonconvex MINLP by decomposing the problem to a convex mixed-ineger quadratic program (MIQP) and nonconvex nonlinear programs (NLP) and solving them iteratively to find the solution to the original problem. Specifically, our previous work decomposes the sizing constraint from the rest using a multidisciprinary design optimization (MDO) method based on augmented Lagrangian and alternating direction method of multipliers (ADMM) [16–18]. We chose the ADMM-based method over other decomposition schemes due to the extensive and established knowledge in augmented Lagrangian and ADMM, its compatibility with parallel computing, and its capability to handle complex subproblem interactions (e.g., hierarchical subsystem design). While it was originally developed for MDO in continuous domain, we employed it to separate discreteness and nonconvexity. Namely, we relax the sizing constraint in (4) and move it to another *subproblem*. To ensure the solution after decoupling is still feasible, the so-called target values are generated, and each subproblem is penalized as its solution deviates from the target. To this end, the augmented Lagrangian penalty function in the following is introduced for each subproblem ℓ

$$\phi(\boldsymbol{\delta}^\ell) = (\boldsymbol{\lambda}^\ell)^\top \boldsymbol{\delta}^\ell + \left\| \boldsymbol{w}^\ell \circ \boldsymbol{\delta}^\ell \right\|_2^2 \tag{5}$$

where $\boldsymbol{\lambda}$ is the Lagrangian multiplier estimates, $\boldsymbol{w}$ is the penalty weights, $\boldsymbol{\delta}$ is the discrepancy from the target values, and $\circ$ represents the Hadamard product. The target values are only generated for the variables shared among subprolbems.

Since the sizing constraint involves three variables (payload capacity Γ_{p_v}, propellant capacity Γ_{f_v}, dry mass m_{d_v}), the discrepancy for subproblem ℓ is

$$\boldsymbol{\delta}^\ell = \begin{bmatrix} \hat{\Gamma}_{p_v} - \Gamma^\ell_{p_v} \\ \hat{\Gamma}_{f_v} - \Gamma^\ell_{f_v} \\ \hat{m}_{d_v} - m^\ell_{d_v} \end{bmatrix} \tag{6}$$

where $\hat{\cdot}$ denotes the target values. To compute the target values, we can solve the following master problem which minimizes the sum of penalty functions for all subproblems

$$\min \quad \sum_\ell \phi(\boldsymbol{\delta}^\ell) \tag{7}$$

Once the target values are computed, the augmented Lagrangian penalty function in Eq. (5) and subproblems can be defined. Notice that a set of target variables (i.e., initial guess) must be provided to initiate the algorithm. To this end, we use the multi-dimensional piecewise linear (PWL) approximation method in [19] for the sizing constraint. See [2, 20] for its usage in space mission planning. To apply the ADMM-based decomposition to (4), we need to define $1 + |\mathcal{V}|$ subproblems; one for space mission planning, and one for each spacecraft $v \in \mathcal{V}$. Let ℓ be $\ell \in \{0, \ldots, |\mathcal{V}|\}$ so that subproblem 0 is for space mission planning, and subproblem v is for spacecraft v. Then, the space mission planning subproblem can be defined as:

$$\begin{aligned} \min \quad & \sum_{t\in\mathcal{T}} \sum_{(v,i,j)\in\mathcal{A}} \left(\boldsymbol{c}_{vijt}{}^\top \boldsymbol{x}^{\omega_1}_{vijt} + c'_{vijt} m_{d_v} u^{\omega_1}_{vijt} \right) + \phi(\boldsymbol{\delta}^0) \\ \text{s.t.} \quad & \text{(1b) to (1d)} \\ & \mathbb{P}\{\boldsymbol{\omega} = \boldsymbol{\omega}_1\} = 1 \end{aligned} \tag{8}$$

Compared to (4), the sizing constraint Eq. (1e) is removed and the augmented Lagrangian penalty function is added to the objective. As ϕ is a convex quadratic function, the subproblem is a convex MIQP. The sizing constraint is moved to its corresponding spacecraft sizing subprolbem as follows:

$$\begin{aligned} \min \quad & \phi(\boldsymbol{\delta}^v) \\ \text{s.t.} \quad & m_{d_v} = \mathcal{F}(\Gamma_{p_v}, \Gamma_{f_v}) \end{aligned} \tag{9}$$

Since it does not have its own objective to minimize (e.g., IMLEO for the space mission planning subproblem), only the augmented Lagrangian penalty is minimized here. Notice it is a nonconvex NLP due to the sizing constraint. Once all subproblems are solved, the Lagrange multiplier estimates and penalty weights are updated for the next iteration. If we let ζ be the iteration number of the algorithm, the Lagrange multiplier estimates are updated by the following rule.

$$\boldsymbol{\lambda}^\ell_{\zeta+1} = \boldsymbol{\lambda}^\ell_\zeta + 2\boldsymbol{w}^\ell_\zeta \circ \boldsymbol{w}^\ell_\zeta \circ \boldsymbol{\delta}^\ell_\zeta \tag{10}$$

Let $w^\ell_{\zeta,k}$ be the k-th element of the penalty weight for subproblem ℓ at iteration ζ. Then, the update rule is

$$w^\ell_{\zeta+1,k} = \begin{cases} w^\ell_{\zeta,k} & \text{if} \quad \left|\delta^\ell_{\zeta,k}\right| \le \gamma_2 \left|\delta^\ell_{\zeta-1,k}\right| \\ \gamma_1 w^\ell_{\zeta,k} & \text{otherwise} \end{cases} \tag{11}$$

where $\gamma_1 > 0$ and $0 < \gamma_2 < 1$ are weight update parameters. In this paper, $\boldsymbol{\lambda}_0 = \mathbf{0}, \boldsymbol{w}_0 = \mathbf{1}, \gamma_1 = 2$, and $\gamma = 0.5$ are used. The algorithm terminates when the discrepancy for all subproblems is within a predefined tolerance; see [17] for the specific termination conditions.

B. Two-Stage Stochastic Programming and Robust Optimization for Mixed-Integer Linear Programs

Another challenge in solving the TSSP problem (1) is the high computational cost introduced by duplicates of the design variables for each scenario. When the number of possible realizations grows, the problem formulation in the form of (2) (i.e., minimizing the first-stage cost plus the second-stage cost expectation) becomes computationally prohibitive even without nonlinear constraints. Two-stage stochastic integer programming has been an active research area to handle such computational cost, and there exist specialized decomposition, cut generation, and branch-and-bound algorithms

[21–24]. Their major drawback is that they often require strict assumptions, such as pure integer second-stage cost or mixed-binary decision variables. However, our problem involves general mixed-integer variables; for example, the number of crew flying over an arc can take values of 0, 1, 2, and so on. There exists a recently published method that applies to general mixed-integer cases [25], but our implementation suggests that the computational expense to generate the proposed cutting planes is still too expensive. As such, we restrict the size of stochastic (sub)problems to be manageable by the generic branch-and-bound algorithm [26].

It is crucial to clarify and discuss why TSSP is needed despite its potentially prohibitive cost. Specifically, one may wonder why it is necessary to consider variables for all scenarios simultaneously. This question can be answered by a concept in stochastic programming called the expected value of perfect information [4]. For a generic TSSP, the following inequality holds

$$\inf_{x\in X} \mathbb{E}[f(\boldsymbol{x},\boldsymbol{\omega})] \geq \mathbb{E}\left[\inf_{x\in X} f(\boldsymbol{x},\boldsymbol{\omega})\right]$$

where f is the objective function, $\boldsymbol{x}$ is the (first- and second-stage) design variable, X is the feasible space for the design variable $\boldsymbol{x}$, and $\boldsymbol{\omega}$ is the random parameter vector; see Chapter 2.4.4 of [4] for the proof. Under the finite support assumption, we can express the right-hand side in terms of the optimal objective values for realizations $\tilde{\boldsymbol{\omega}}$:

$$\inf_{\boldsymbol{x}\in X} \mathbb{E}[f(\boldsymbol{x},\boldsymbol{\omega})] \geq \mathbb{E}\left[\inf_{\boldsymbol{x}\in X} f(\boldsymbol{x},\boldsymbol{\omega})\right] = \sum_{\tilde{\boldsymbol{\omega}}\in\Omega} \mathbb{P}\left\{\boldsymbol{\omega}=\tilde{\boldsymbol{\omega}}\right\}\left[\inf_{\boldsymbol{x}\in X} f(\boldsymbol{x},\tilde{\boldsymbol{\omega}})\right]$$

The difference between the two quantities is known as the expected value of perfect information (EVPI)

$$\text{EVPI} = \inf_{\boldsymbol{x}\in X} \mathbb{E}[f(\boldsymbol{x},\boldsymbol{\omega})] - \mathbb{E}\left[\inf_{\boldsymbol{x}\in X} f(\boldsymbol{x},\boldsymbol{\omega})\right]$$

and it is always nonnegative. EVPI = 0 if and only if the minimizer of the objective f is independent of realization $\tilde{\boldsymbol{\omega}}$, which is rarely the case [4]. An important implication here is that we, in general, cannot run minimization for each scenario and then take the expectation to recover the TSSP objective. As such, TSSP formulations and their computational cost are justified and needed for flexible decision-making under uncertainty.

One way to tackle the computational expense of a stochastic program is robust optimization. It can be described as the optimization of the worst-case objective, and its general expression is

$$\min_{x\in X} \max_{\tilde{\boldsymbol{\omega}}\in\Omega} f(\boldsymbol{x},\tilde{\boldsymbol{\omega}})$$

It is particularly useful for *hard constraints*, namely the constraints that must be satisfied for all realizations $\tilde{\boldsymbol{\omega}} \in \Omega$ [10]. In our problem, the concurrency constraint in inequality (1c) is an example of a hard constraint because the spacecraft cannot physically carry more commodities than its capacity, regardless of the realization. It is known that, under some mild assumptions, an equivalent deterministic formulation can be found for robust linear programs [10] and MILP [27, 28]. Note that it is not a trivial task, especially when uncertainties exist on both sides of (in)equalities to define the feasible space. If a deterministic formulation can be found, a robust optimization problem can be solved efficiently. Although it is not a replacement for solving TSSP, we can still use it to facilitate the solution process of (1).

IV. Proposed Approach

To efficiently solve the problem of integrated space mission planning under uncertainty in (1), we propose a novel approach that combines the ADMM-based decomposition, robust optimization, and TSSP for MILP. It first solves a deterministic robust formulation using the ADMM-based decomposition to find robust spacecraft sizes that meet nonlinear and nonconvex constraint Eq. (1e). Then, it solves the TSSP with the robust spacecraft design, which is a MILP as the sizing constraint is removed. The pseudo-algorithm for this method is presented in Algorithm 1. Even though the proposed approach still accounts for uncertainties in the future, space mission planning and spacecraft sizing are no longer coupled when solving TSSP. That is, it is based on the hypothesis that the solutions found by the original TSSP problem in (1) and its robust variation in (12) have the same or similar spacecraft design.

$$\begin{aligned} \min \quad & \sum_{t\in\mathcal{T}_1} \sum_{(v,i,j)\in\mathcal{A}} \left(\boldsymbol{c}_{vijt}{}^{\top}\boldsymbol{x}^{\omega_1}_{vijt} + c'_{vijt} m_{d_v} u^{\omega_1}_{vijt}\right) + \max_{\omega} \left\{ \sum_{t\in\mathcal{T}_2} \sum_{(v,i,j)\in\mathcal{A}} \left(\boldsymbol{c}_{vijt}{}^{\top}\boldsymbol{x}^{\omega}_{vijt} + c'_{vijt} m_{d_v} u^{\omega}_{vijt}\right) \right\} \qquad (12) \\ \text{s.t.} \quad & \text{(1b) to (1f)} \end{aligned}$$

The justification of this approach can be explained as follows. Recall that we assume that the spacecraft cannot be modified once built for the first mission. In addition, the concurrency constraint in inequality (1c) states that the spacecraft load must be within its capacity for all realizations. As such, the spacecraft design by the original formulation (1) must be at least *robust feasible*; that is, the spacecraft must be able to carry out the campaign for all scenarios, including the worst case. We need to emphasize that it by no means guarantees that the optimal spacecraft design found in the original TSSP problem is *robust optimal* (i.e., the optimal design for the robust formulation (12)), especially when multiple spacecraft are considered. This is because robust feasible spacecraft designs are non-unique, and the sizing constraint is nonconvex.

The first step of our approach is to compute the robust spacecraft design. Let $\boldsymbol{\omega}_R$ be the realization corresponding to the worst case. The deterministic robust optimization is expressed as:

$$\begin{aligned} \min \quad & \sum_{t\in\mathcal{T}} \sum_{(v,i,j)\in\mathcal{A}} \left(\boldsymbol{c}_{vijt}{}^{\top} \boldsymbol{x}_{vijt}^{\omega_R} + c'_{vijt} m_{d_v} u_{vijt}^{\omega_R} \right) \\ \text{s.t.} \quad & \text{(1b) to (1f)} \\ & \mathbb{P}\{\boldsymbol{\omega} = \boldsymbol{\omega}_R\} = 1 \end{aligned} \tag{DRO}$$

Note that identifying such a worst scenario or realization $\boldsymbol{\omega}_R$ can be challenging. If so, the previously mentioned methods [10, 27] can be employed to derive the deterministic robust problem. For simplicity, we assume that $\boldsymbol{\omega}_R$ can be found easily in this paper. (DRO) is a deterministic integrated space mission planning problem; hence, it can be solved using the ADMM-based decomposition method. Recall that the initial guess of (DRO) can be computed by applying the PWL approximation to the sizing constraint. The robust space mission planning subproblem can be formulated as follows:

$$\begin{aligned} \min \quad & \sum_{t\in\mathcal{T}} \sum_{(v,i,j)\in\mathcal{A}} \left(\boldsymbol{c}_{vijt}{}^{\top} \boldsymbol{x}_{vijt}^{\omega_R} + c'_{vijt} m_{d_v} u_{vijt}^{\omega_R} \right) + \phi(\boldsymbol{\delta}^0) \\ \text{s.t.} \quad & \text{(1b) to (1d) and Eq. (1f)} \\ & \mathbb{P}\{\boldsymbol{\omega} = \boldsymbol{\omega}_R\} = 1 \end{aligned} \tag{13}$$

The spacecraft sizing subproblems remain the same as (9). Once the robust optimal solution (denoted by $^{\mathrm{RO}}$) is obtained, the two-stage stochastic MILP can be formulated as follows

$$\begin{aligned} \min \quad & \sum_{t\in\mathcal{T}_1} \sum_{(v,i,j)\in\mathcal{A}} \left(\boldsymbol{c}_{vijt}{}^{\top} \boldsymbol{x}_{vijt}^{\omega_1} + c'_{vijt} m_{d_v} u_{vijt}^{\omega_1} \right) + \mathbb{E}\left[\sum_{t\in\mathcal{T}_2} \sum_{(v,i,j)\in\mathcal{A}} \left(\boldsymbol{c}_{vijt}{}^{\top} \boldsymbol{x}_{vijt}^{\omega} + c'_{vijt} m_{d_v} u_{vijt}^{\omega} \right) \right] \\ \text{s.t.} \quad & \text{(1b) to (1d) and Eq. (1f)}, \quad m_{d_v} = m_{d_v}^{\mathrm{RO}}, \quad \Gamma_{p_v} = \Gamma_{p_v}^{\mathrm{RO}}, \quad \Gamma_{f_v} = \Gamma_{f_v}^{\mathrm{RO}} \end{aligned} \tag{TSSMILP}$$

Since it is a MILP without spacecraft sizing, the computational expense is greatly reduced compared to the original formulation. The deterministic robust optimization (DRO) is also more manageable than the original as there is only one scenario. From the perspective of computational cost, the proposed approach attempts to solve a sequence of tractable (sub)problems to find the solution for the original computationally expensive problem. However, as mentioned, the robust optimal spacecraft design might or might not be optimal for the TSSP. Therefore, the proposed approach risks the optimality for computational cost reduction. We numerically test the trade-off in Section V through case studies.

Algorithm 1 Proposed approach for the integrated space mission planning under uncertainty

Require: Parameters and sets in Table 1, piecewise linear (PWL) approximation breakpoint increment
Require: $\boldsymbol{\lambda}, \boldsymbol{w}, \gamma_1, \gamma_2$, robust realization $\boldsymbol{\omega}_R \in \Omega$
  $\hat{\Gamma}_{p_v}, \hat{\Gamma}_{f_v} \leftarrow$ Solve (DRO) with PWL approximation for $m_{d_v} = \mathcal{F}(\Gamma_{p_v}, \Gamma_{f_v})$
  $\hat{m}_{d_v} \leftarrow \mathcal{F}(\hat{\Gamma}_{p_v}, \hat{\Gamma}_{f_v})$
  while ADMM termination conditions are not met **do** ▷ Solving deterministic robust optimization (DRO)
    for $\ell \in \{0, \ldots, |\mathcal{V}|\}$ **do**
      if $\ell = 0$ **then**
        Solve (13) ▷ Space mission planning subproblem
      else
        Solve (9) ▷ Spacecraft sizing subproblem
      end if
    end for
    $\boldsymbol{\lambda} \leftarrow$ (10) ▷ Lagrange multiplier estimate update
    $\boldsymbol{w} \leftarrow$ (11) ▷ Penalty weight update
    $\hat{\Gamma}_{p_v}, \hat{\Gamma}_{f_v}, \hat{m}_{d_v} \leftarrow$ Solve (7) ▷ Target variable update
  end while
  $\Gamma^{\mathrm{RO}}_{p_v}, \Gamma^{\mathrm{RO}}_{f_v}, m^{\mathrm{RO}}_{d_v} \leftarrow \hat{\Gamma}_{p_v}, \hat{\Gamma}_{f_v}, \hat{m}_{d_v}$
  Solve (TSSMILP) ▷ Two-stage stochastic mixed-integer linear program

V. Case Study: Staged Infrastructure Deployment on the Lunar Surface

A staged deployment of infrastructure elements on the lunar surface under demand uncertainty is considered as a case study to examine the performance of the proposed approach and test the hypothesis mentioned. It is a two-mission crewed campaign to the lunar surface where infrastructure is deployed over the two missions. Whereas the amount of infrastructure elements to be delivered and deployed is fixed for the first mission, it is uncertain for the second mission. When this campaign is seen as the preparation phase for further space exploration in the future, it is natural to assume that the demand for the infrastructure is uncertain, and a flexible deployment is crucial. In addition, we assume that the sample return mass from the lunar surface is uncertain with a smaller deviation.

A. Case Study Settings

The detailed case study settings are given in Fig. 1 and Tables 2 to 4. First, Fig. 1 defines the lunar campaign network nodes and arcs along with their Δv cost and time of flight. The commodities and spacecraft are not allowed to fly over arcs not defined in Fig. 1 (e.g., flying directly from Earth to LLO is not possible).

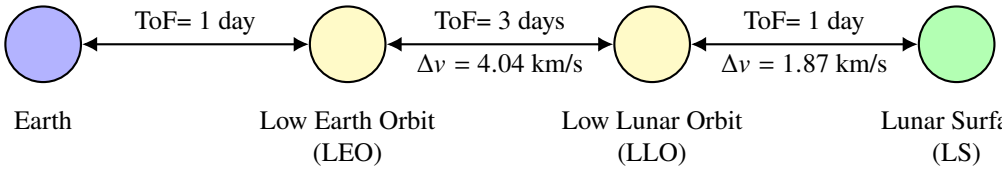

Fig. 1 Lunar campaign network (modified from [20]).

The description and values of the parameters used in this case study are summarized in Table 2. It includes some parameters for the spacecraft sizing model; see Eq. (15) in the appendix for the model. The demand and supply of commodities are specified in Table 3. The k-th element of the uncertain parameter vector for scenario s is denoted by $\boldsymbol{\omega}_{s,k}$. Since there are two uncertain parameters for each scenario (i.e., infrastructure demand at the lunar surface (LS) node and sample return demand at the Earth node), we have $\boldsymbol{\omega}_s \in \mathbb{R}^2$. The realization values of the infrastructure elements and returned sample mass are the first and second elements of $\boldsymbol{\omega}_s$, respectively, and they are specified in Table 4. Note that the number of scenarios, $|\Omega|$, is at least 2 and at most 4. The support Ω is defined so that it includes realizations from the first $|\Omega|$ scenario indices (i.e., $\Omega = \{\boldsymbol{\omega}_1, \boldsymbol{\omega}_2, \ldots\}$) and does not skip an index (e.g., $\Omega = \{\boldsymbol{\omega}_1, \boldsymbol{\omega}_3\}$

is not allowed). It is also important to note that these realizations are defined so that the robust scenario is easy to identify. For example, if we consider three scenarios, the support is $\Omega = \{\omega_1, \omega_2, \omega_3\}$, and ω_3 is the robust scenario due to its highest demand for both elements. See Section IV for the justification of this rather strong assumption.

Table 2 Parameters used in the case study problem

Parameters	Values
Spacecraft propellant type	LH2/LOX
Propellant I_{sp}, s	420
Propellant density ρ_f, kg/m^3	360
Spacecraft miscellaneous mass fraction c_{misc}	0.05
Spacecraft payload capacity range, kg	[500, 10,000]
Spacecraft propellant capacity range, kg	[1,000, 100,000]
Increment for piecewise linear approximation breakpoints, kg	2,500
Spacecraft maintenance supply mass per dry mass per flight	0.01
Crewed mission duration on lunar surface t_{surf}, days	3
Crew mass (including space suit), kg/person	100
Crew consumption, kg/(day·person)	8.655
Time interval between mission start dates, days	365

Table 3 Lunar campaign commodity demand and supply

Payload Type	Node	Date	Supply/Demand
Outbound to the Moon			
Crew	Earth	0, 365	4
Infra. elements, consumables, and propellant, kg	Earth	0, 365	∞
Crew	LS	5, 370	-4
Infrastructure elements, kg	LS	5	-2000
		370	$-\omega_{s,1}$
Inbound to the Earth			
Crew	LS	8, 373	4
Returned sample mass, kg	LS	8, 373	∞
Crew	Earth	13, 378	-4
Returned sample mass, kg	Earth	13	-1000
		378	$-\omega_{s,2}$

Table 4 Uncertainty realization for each scenario

Scenario index s	1	2	3	4
ω_s	$\begin{bmatrix}2500\\900\end{bmatrix}$	$\begin{bmatrix}3500\\1100\end{bmatrix}$	$\begin{bmatrix}4000\\1200\end{bmatrix}$	$\begin{bmatrix}2000\\800\end{bmatrix}$

The results of the numerical experiment presented in this paper were produced by a machine with a Ryzen™9 7950X CPU (16 core 32 threads at 4.5 to 5.7 GHz) and 64 GB of RAM. Gurobi 10 [29] and IPOPT [30] were used for convex mixed-integer (sub)problems and nonconvex NLP subproblems, respectively.

B. Fully Coupled Benchmark Formulation

To analyze the accuracy and computational efficiency of the proposed approach, we define and solve a benchmark formulation for the same problem. It is a more straightforward extension of the deterministic ADMM-based method in our previous work [2]. Instead of precomputing the spacecraft design, we can let the space campaign planning

subproblem handle the uncertainty directly. As a result, space mission planning and spacecraft design are fully coupled. The subproblem can then be expressed as follows where the nonconvex spacecraft sizing constraint Eq. (1e) is removed from the original problem (1), and the augmented Lagrangian penalty function is added to the objective:

$$\min \quad \sum_{t\in\mathcal{T}_1}\sum_{(v,i,j)\in\mathcal{A}}\left(\boldsymbol{c}_{vijt}{}^{\top}\boldsymbol{x}^{\omega_1}_{vijt}+c'_{vijt}m_{d_v}u^{\omega_1}_{vijt}\right)+\mathbb{E}\left[\sum_{t\in\mathcal{T}_2}\sum_{(v,i,j)\in\mathcal{A}}\left(\boldsymbol{c}_{vijt}{}^{\top}\boldsymbol{x}^{\omega}_{vijt}+c'_{vijt}m_{d_v}u^{\omega}_{vijt}\right)\right]+\phi(\boldsymbol{\delta}^0) \tag{14}$$

s.t. (1b) to (1d) and Eq. (1f)

The spacecraft sizing subproblems remain the same as (9), and an initial guess can be produced by solving (1) with PWL approximation for the sizing constraint Eq. (1e). As the coupling between space campaign planning and spacecraft design is fully considered here, it can yield a better solution than the proposed approach.

C. Performance Comparison

The optimization results of the proposed and benchmark methods are compared to study the trade-off between optimality and computational cost. Hence, the quantities of particular interest are the objective value (i.e., IMLEO) and total computational time. Various numbers of scenarios are considered for four problem instances, and it is assumed that all scenarios are equally probable. Let n_{des} be the number of different spacecraft designs and n_{copy} be the number of spacecraft per design such that the total of $n_{\text{des}}n_{\text{copy}}$ spacecraft is used in the campaign. Then, pairs of values $(n_{\text{des}}, n_{\text{copy}})$ are defined as $(2,3)$, $(3,2)$, $(4,1)$, and $(5,1)$ for problem instances 1, 2, 3, and 4, respectively. The comparison of the results of the proposed and benchmark methods is summarized in Table 5. The maximum computational time for each (sub)problem, not to be confused with the total computational time, was set to 3 hours (i.e., 10,800 s). The results where the time limit was reached are denoted by *.

Table 5 Performance comparison of the proposed approach and fully coupled benchmark formulation

Problem instance	$\lvert\Omega\rvert$	Method	IMLEO, kg	Comp. time, s	Rel. diff. in IMLEO, %	Reduction in comp. time, %
1 ($n_{\text{des}}=2, n_{\text{copy}}=3$)	2	Proposed	438,696	17	0	87.5
		Fully coupled	438,696	136		
	3	Proposed	451,703	20	0	98.9
		Fully coupled	451,703	1,828		
	4	Proposed	449,844	20	-0.63	99.8
		Fully coupled	452,700*	11,223*		
2 ($n_{\text{des}}=3, n_{\text{copy}}=2$)	2	Proposed	438,640	26	0.45	97.7
		Fully coupled	436,668	1,117		
	3	Proposed	451,717	20	0.48	99.8
		Fully coupled	449,567*	10,839*		
	4	Proposed	449,857	45	0.66	99.6
		Fully coupled	446,919*	10,903*		
3 ($n_{\text{des}}=4, n_{\text{copy}}=1$)	2	Proposed	438,696	10	-0.002	71.4
		Fully coupled	438,705	35		
	3	Proposed	451,753	9	0	69.0
		Fully coupled	451,753	29		
	4	Proposed	449,844	9	0	84.8
		Fully coupled	449,844	59		
4 ($n_{\text{des}}=5, n_{\text{copy}}=1$)	2	Proposed	421,434	71	-2.00	58.7
		Fully coupled	430,030	172		
	3	Proposed	436,327	49	-0.49	86.3
		Fully coupled	438,480	358		
	4	Proposed	432,560	49	-0.33	99.0
		Fully coupled	433,984	5,101		

One of the most important observations is that the objective values of the proposed and benchmark methods are closely matched or identical for every problem instance and the number of scenarios. It aligns with expectations that the two methods have slightly different objective values due to the nonconvex nature of the problem. Note that, even though the fully coupled benchmark case explores the entire design space, it might converge to locally optimal solutions, which is why in some cases it shows a worse result than the proposed method. Another observation is that whether the proposed solution can find a close or identical solution to the fully coupled problem appears to depend more on the problem instance (number of spacecraft designed, etc.) than the number of scenarios. For instances 1 and 3, we see smaller relative differences in the objective than in the other instances (note that the benchmark method for instance 1 with four scenarios is terminated prematurely due to the three-hour time limit). On the other hand, the proposed method seems to perform better and worse for instances 4 and 2, respectively. Nevertheless, the results demonstrate that the proposed method can achieve nearly identical or the same objectives as the fully coupled benchmark method.

The most significant advantage of the proposed approach is its low computational cost. The comparison of computational costs between the two methods is visualized in Fig. 2 for every combination of problem instances and numbers of scenarios. We observe that the computational time is reduced for every case, with a reduction exceeding 90% in half of the cases; see Table 5. Whereas the benchmark method suffers from high or prohibitive computational costs in several cases, the proposed approach has more consistent computational times. The savings are particularly significant for instances 1, 2, and 4 as well as for the four-scenario cases. Notice that instance 3 exhibits much lower computational times than the other instances, regardless of the methods used. It is not surprising to see a specific problem setting that is considerably easier to solve than the others, although such a setting is difficult to predict. The proposed method still finds (almost) identical objective values, but the cost reduction might not be as beneficial. Yet, given more consistent performance and the fact that it is not possible to know in advance whether a specific problem is manageable by the fully coupled method or not, the proposed method can work as an effective and computationally efficient approach for solving integrated space mission planning under uncertainty.

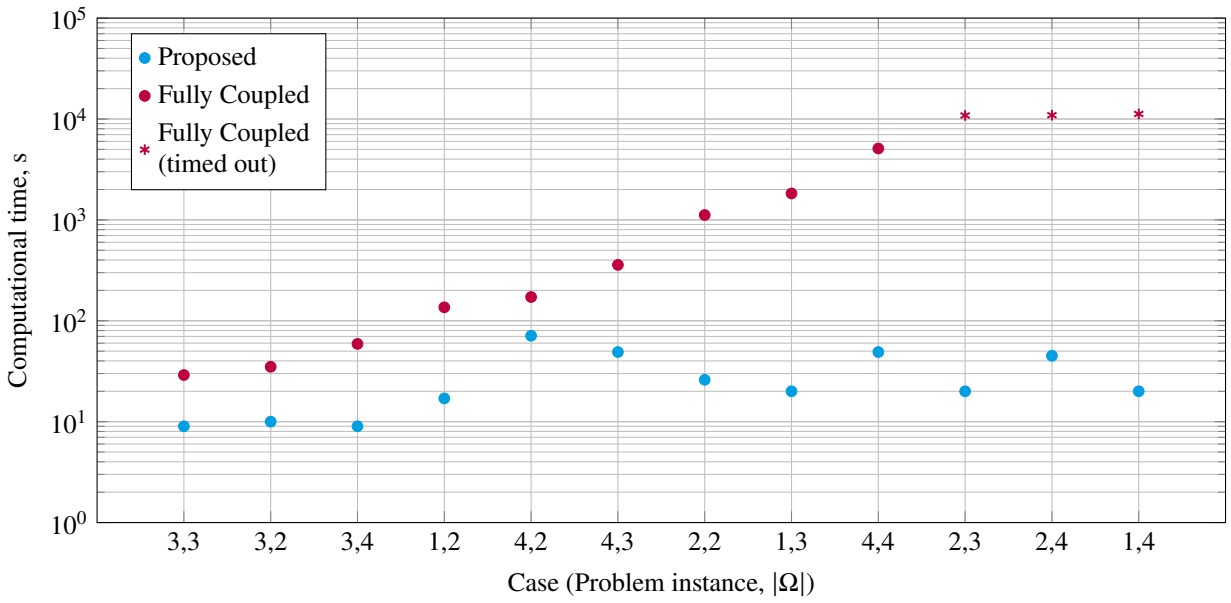

Fig. 2 Comparison of computational times for all problem instances and numbers of scenarios. The cases are sorted in ascending order based on the computational time of the fully coupled benchmark method.

Another advantage of the proposed approach is its scalability with respect to an increasing number of scenarios. As seen in the plots for instances 1, 2, 4 in Fig. 3, the cost of the benchmark method rapidly increases as the number of scenarios grows. On the other hand, the computational time for the proposed method either does not change much or grows moderately with the number of scenarios. It suggests the potential of the proposed method to handle greater numbers of scenarios, which is crucial when considering campaigns with more uncertainties, and hence more scenarios. While we make simplified assumptions about realizations in this paper, it is often unavoidable to consider a greater number of scenarios. For example, assume the two uncertain parameters we consider (i.e., infrastructure and return sample mass demands) are independent and can take four realizations each. Then, the number of possible combinations is now 16, and it only increases as we take other uncertainties into account. Based on the trend that we see in the computational cost, it is likely that such a problem can only be handled by the proposed approach. The analysis of the growing number of scenarios is outside the scope of this paper, and we leave it to future work.

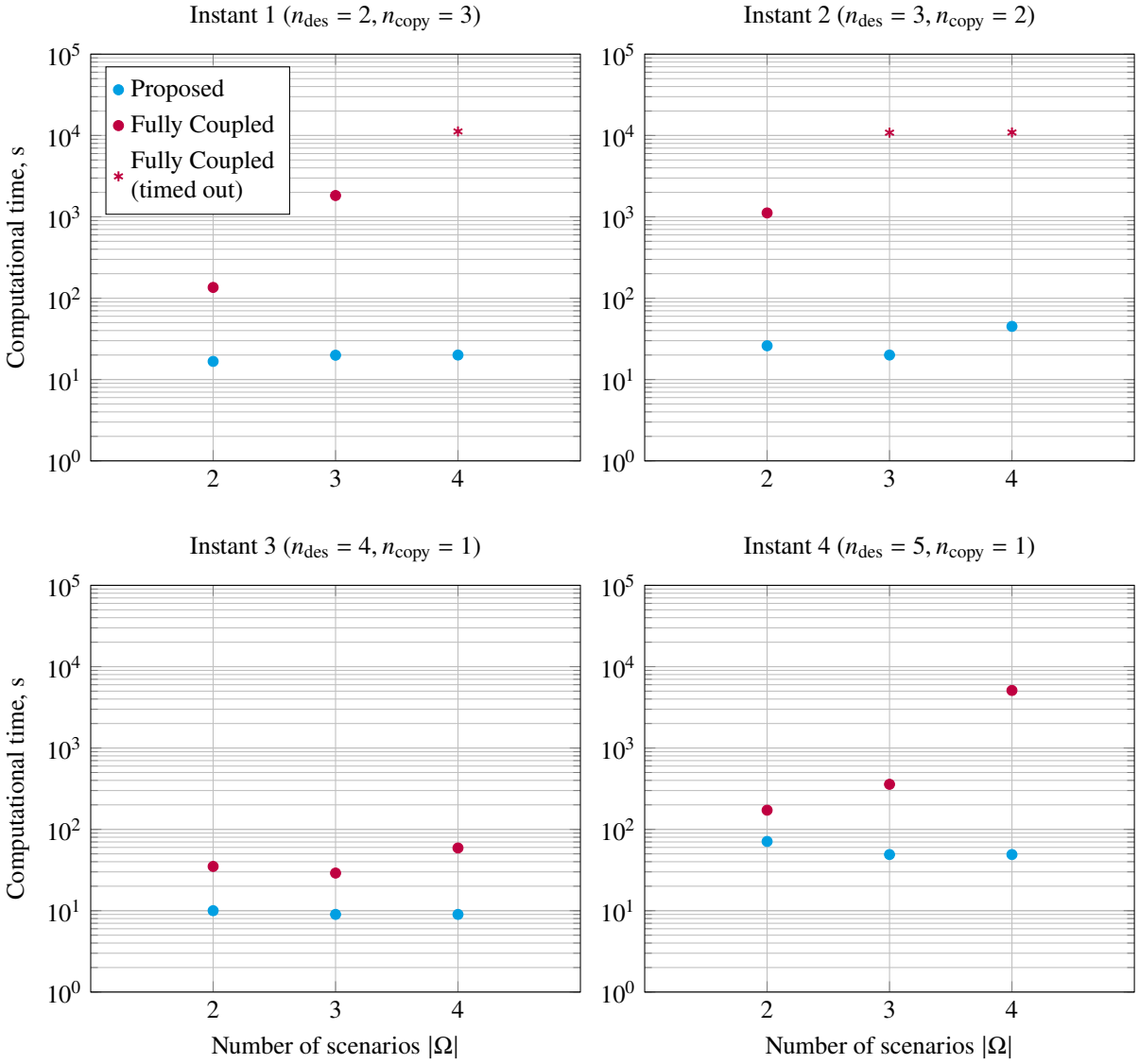

Fig. 3 Comparison of computational time between the proposed and fully coupled benchmark methods for each case study instance.

VI. Conclusion

This work has proposed a computationally efficient method for the problem of integrated space mission planning under uncertainty, which is a stochastic MINLP and can be intractable as is. The proposed approach effectively combines the ADMM-based decomposition framework from our previous work, robust optimization, and two-stage stochastic mixed-integer linear programming. It first solves the deterministic concurrent optimization of space mission planning and spacecraft design for the worst case (i.e., robust optimization). Then, the two-stage stochastic program is solved with the precomputed robust spacecraft design. Since the nonlinearity in spacecraft sizing constraint is removed, the problem becomes a MILP, which is significantly more manageable. Despite its lower computational cost, it does not entirely capture the coupling between mission planning and spacecraft design. That is, whereas the proposed approach can reduce the computational cost, it can potentially miss a better solution that could be found if the problem is fully coupled. This balance between computational cost and optimality is examined through a case study of staged infrastructure deployment on the lunar surface under future demand uncertainty. For different instances and uncertainty realizations, the proposed method is compared with the fully coupled benchmark method. The results of the case study suggest that the proposed approach can achieve an objective value closely matched or identical to the benchmark method while drastically reducing the computational cost. In all the problem cases, the worst relative difference in the objective value was less than 1%, whereas the computational time is reduced for every case, with a reduction exceeding 90% in half of the cases. Although there is no guarantee that the method can always find a closely matched solution, it is a practical and computationally efficient approach when the original stochastic MINLP formulation is too expensive or impossible to solve. Future work could explore more general forms of uncertainties and develop a systematic method to derive a robust optimization formulation when the robust scenario is not easily identifiable.

Appendix

The spacecraft sizing constraint $m_d = \mathcal{F}(\Gamma_p, \Gamma_f)$ is defined in Eq. (15). It is a parametric model for lunar landers, originally proposed in our previous work [2] using the data and method developed in [31, 32]. Assuming that the size of each spacecraft subsystem depends on the total dry mass of the spacecraft m_d, mass estimation relations can be developed for all subsystems in terms of m_d and other parameters. Since the dry mass is merely the summation of all subsystem mass, an implicit equation can be written as shown in Eq. (15). Given all other parameters, we can numerically solve for m_d.

$$m_d = \sum m_{\text{sub}} = m_{\text{str}} + m_{\text{prop}} + m_{\text{power}} + m_{\text{avi}} + m_{\text{ECLSS}} + m_{\text{misc}}$$

where

$$\begin{aligned}
m_{\text{str}} &= {n_{\text{stg}}}^{-0.6705} \left(0.3238\, m_d + 693.7\, {\Gamma_p}^{0.04590}\right) \\
m_{\text{prop}} &= 0.1648 \left(m_d + \Gamma_p\right) + 20.26\, \frac{\Gamma_f}{\rho_f} \\
m_{\text{power}} &= 7.277 \cdot 10^{-8}\, {m_d}^{2.443} + 137.0 \\
m_{\text{avi}} &= 1.014\, {m_{\text{power}}}^{0.8423} + 22.33\, t_{\text{surf}} \\
m_{\text{ECLSS}} &= 0.004190\, n_{\text{crew}}\, t_{\text{surf}}\, {m_d}^{0.9061}\, {n_{\text{stg}}}^{0.7359} + 434.7 \\
m_{\text{misc}} &= c_{\text{misc}}\, m_d
\end{aligned} \tag{15}$$

Note that n_{stg} is the number of stages (either 1 or 2), ρ_f is the propellant density in kg/m^3, n_{crew} is the number of crew, t_{surf} is the lunar surface mission duration in days, and c_{misc} is the miscellaneous mass fraction. m_{str}, m_{prop}, m_{avi}, m_{ECLSS}, and m_{misc} are the mass for the structure and thermal protection system, the propulsion system (e.g., engines, tanks), the power system, the avionics, the environmental control and life support system, and miscellaneous subsystems, respectively. $n_{\text{stg}} = 1$ is used in this paper, and other parameters can be found in Table 2. See [2] for more details on the model.

Funding Sources

This work was conducted with support from the National Science Foundation under Grant No. 1942559.

References

[1] Taylor, C., "Integrated Transportation System Design Optimization," Ph.D. thesis, Dept. of Aeronautics and Astronautics, MIT, Cambridge, MA, 2007.

[2] Isaji, M., Takubo, Y., and Ho, K., "Multidisciplinary Design Optimization Approach to Integrated Space Mission Planning and Spacecraft Design," *Journal of Spacecraft and Rockets*, Vol. 59, No. 5, 2022, pp. 1660–1670. https://doi.org/10.2514/1.A35284.

[3] de Weck, O. L., de Neufville, R., and Chaize, M., "Staged Deployment of Communications Satellite Constellations in Low Earth Orbit," *Journal of Aerospace Computing, Information, and Communication*, Vol. 1, No. 3, 2004, pp. 119–136. https://doi.org/10.2514/1.6346.

[4] Shapiro, A., Dentcheva, D., and Ruszczynski, A., *Lectures on Stochastic Programming: Modeling and Theory, Third Edition*, 3rd ed., Society for Industrial and Applied Mathematics, Philadelphia, PA, 2021. https://doi.org/10.1137/1.9781611976595.

[5] Sarton du Jonchay, T., Chen, H., Gunasekara, O., and Ho, K., "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. https://doi.org/10.2514/1.A34978.

[6] Sarton du Jonchay, T., Chen, H., Isaji, M., Shimane, Y., and Ho, K., "On-Orbit Servicing Optimization Framework with High- and Low-Thrust Propulsion Tradeoff," *Journal of Spacecraft and Rockets*, Vol. 59, No. 1, 2022, pp. 33–48. https://doi.org/10.2514/1.A35094, URL https://doi.org/10.2514/1.A35094.

[7] Lee, H. W., Jakob, P. C., Ho, K., Shimizu, S., and Yoshikawa, S., "Optimization of Satellite Constellation Deployment Strategy Considering Uncertain Areas of Interest," *Acta Astronautica*, Vol. 153, 2018, pp. 213–228. https://doi.org/10.1016/j.actaastro.2018.03.054.

[8] Gollins, N. J., and Ho, K., "Optimization of Multi-Mission Space Exploration Campaign Schedules Subject to Stochastic Launch Delay," *AIAA SciTech 2024 Forum*, American Institute of Aeronautics and Astronautics, 2024. https://doi.org/10.2514/6.2024-2718.

[9] Takubo, Y., Chen, H., and Ho, K., "Hierarchical Reinforcement Learning Framework for Stochastic Spaceflight Campaign Design," *Journal of Spacecraft and Rockets*, Vol. 59, No. 2, 2022, pp. 421–433. https://doi.org/10.2514/1.A35122.

[10] Ben-Tal, A., and Nemirovski, A., "Robust Solutions of Uncertain Linear Programs," *Operations Research Letters*, Vol. 25, No. 1, 1999, pp. 1–13. https://doi.org/10.1016/S0167-6377(99)00016-4.

[11] Kleywegt, A. J., Shapiro, A., and Homem-de Mello, T., "The Sample Average Approximation Method for Stochastic Discrete Optimization," *SIAM Journal on Optimization*, Vol. 12, No. 2, 2002, pp. 479–502. https://doi.org/10.1137/S1052623499363220.

[12] Ho, K., De Weck, O. L., Hoffman, J. A., and Shishko, R., "Dynamic Modeling and Optimization for Space Logistics Using Time-Expanded Networks," *Acta Astronautica*, Vol. 105, No. 2, 2014, pp. 428–443. https://doi.org/10.1016/j.actaastro.2014.10.026.

[13] Ishimatsu, T., De Weck, O. L., Hoffman, J. A., and Ohkami, Y., "Generalized Multicommodity Network Flow Model for the Earth-Moon-Mars Logistics System," *Journal of Spacecraft and Rockets*, Vol. 53, No. 1, 2016, pp. 25–38. https://doi.org/10.2514/1.A33235.

[14] Chen, H., Lee, H. W., and Ho, K., "Space Transportation System and Mission Planning for Regular Interplanetary Missions," *Journal of Spacecraft and Rockets*, Vol. 56, No. 2, 2019, pp. 12–20. https://doi.org/10.2514/1.A34168.

[15] Sahinidis, N. V., "BARON: A general purpose global optimization software package," *Journal of Global Optimization*, Vol. 8, No. 2, 1996, pp. 201–205. https://doi.org/10.1007/BF00138693.

[16] Tosserams, S., Etman, L., Papalambros, P., and Rooda, J., "An augmented Lagrangian relaxation for analytical target cascading using the alternating direction method of multipliers," *Structural and Multidisciplinary Optimization*, Vol. 31, No. 3, 2006, pp. 176–189. https://doi.org/10.1007/s00158-005-0579-0.

[17] Tosserams, S., Etman, L., and Rooda, J., "An augmented Lagrangian decomposition method for quasi-separable problems in MDO," *Structural and Multidisciplinary Optimization*, Vol. 34, No. 3, 2007, pp. 211–217. https://doi.org/10.1007/s00158-006-0077-z.

[18] Tosserams, S., Etman, L. F. P., and Rooda, J. E., "Augmented Lagrangian coordination for distributed optimal design in MDO," *International Journal for Numerical Methods in Engineering*, Vol. 73, No. 13, 2008, pp. 1885–1910. https://doi.org/https://doi.org/10.1002/nme.2158.

[19] Vielma, J. P., Ahmed, S., and Nemhauser, G., "Mixed-Integer Models for Nonseparable Piecewise-Linear Optimization: Unifying Framework and Extensions," *Operations Research*, Vol. 58, No. 2, 2010, pp. 303–315. https://doi.org/10.1287/opre.1090.0721.

[20] Chen, H., and Ho, K., "Integrated Space Logistics Mission Planning and Spacecraft Design with Mixed-Integer Nonlinear Programming," *Journal of Spacecraft and Rockets*, Vol. 55, No. 2, 2018, pp. 365–381. https://doi.org/10.2514/1.A33905.

[21] Ahmed, S., Tawarmalani, M., and Sahinidis, N. V., "A Finite Branch-and-Bound Algorithm for Two-Stage Stochastic Integer Programs," *Mathematical Programming*, Vol. 100, No. 2, 2004, pp. 355–377. https://doi.org/10.1007/s10107-003-0475-6.

[22] Gade, D., Küçükyavuz, S., and Sen, S., "Decomposition algorithms with parametric Gomory cuts for two-stage stochastic integer programs," *Math. Program.*, Vol. 144, 2014, pp. 39–64. https://doi.org/https://doi.org/10.1007/s10107-012-0615-y.

[23] Laporte, G., and Louveaux, F. V., "The integer L-shaped method for stochastic integer programs with complete recourse," *Operations Research Letters*, Vol. 13, No. 3, 1993, pp. 133–142. https://doi.org/https://doi.org/10.1016/0167-6377(93)90002-X.

[24] Zhang, M., and Küçükyavuz, S., "Finitely Convergent Decomposition Algorithms for Two-Stage Stochastic Pure Integer Programs," *SIAM Journal on Optimization*, Vol. 24, No. 4, 2014, pp. 1933–1951. https://doi.org/10.1137/13092678X.

[25] van der Laan, N., and Romeijnders, W., "A Converging Benders' Decomposition Algorithm for Two-Stage Mixed-Integer Recourse Models," *Operations Research*, Vol. 0, No. 0, 2023. https://doi.org/10.1287/opre.2021.2223.

[26] Land, A. H., and Doig, A. G., "An Automatic Method of Solving Discrete Programming Problems," *Econometrica*, Vol. 28, No. 3, 1960, pp. 497–520.

[27] Lin, X., Janak, S. L., and Floudas, C. A., "A New Robust Optimization Approach for Scheduling under Uncertainty: I. Bounded Uncertainty," *Computers & Chemical Engineering*, Vol. 28, No. 6, 2004, pp. 1069–1085. https://doi.org/10.1016/j.compchemeng.2003.09.020.

[28] Janak, S. L., Lin, X., and Floudas, C. A., "A New Robust Optimization Approach for Scheduling under Uncertainty: II. Uncertainty with Known Probability Distribution," *Computers & Chemical Engineering*, Vol. 31, No. 3, 2007, pp. 171–195. https://doi.org/10.1016/j.compchemeng.2006.05.035.

[29] Gurobi Optimization, LLC, "Gurobi Optimizer Reference Manual," , 2021. URL https://www.gurobi.com.

[30] Wächter, A., and Biegler, L. T., "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Mathematical Programming*, Vol. 106, No. 1, 2006, pp. 25–57. https://doi.org/10.1007/s10107-004-0559-y.

[31] Isaji, M., Maynard, I., and Chudoba, B., "Surface Access Architecture Modeling: Trend Analysis and Classification from a Lander Database," *AIAA SPACE and Astronautics Forum and Exposition 2018*, AIAA, 2018. https://doi.org/10.2514/6.2018-5131.

[32] Isaji, M., Maynard, I., and Chudoba, B., "A New Sizing Methodology for Lunar Surface Access Systems," *AIAA SciTech 2020 Forum*, AIAA, 2020. https://doi.org/10.2514/6.2020-1775.