MITSUBISHI ELECTRIC RESEARCH LABORATORIES
https://www.merl.com

Powered Descent Decision Making: A Reachability-Steering Approach

Kento, Tomita; Elango, Purnanand; Vinod, Abraham P.; Di Cairano, Stefano; Weiss, Avishai

TR2026-014 January 14, 2026

Abstract

Classical powered descent guidance (PDG) algorithms focus on low-level trajectory op- timization, such as minimizing propellant consumption subject to soft-landing constraints. However, in future missions to unexplored bodies, the dominant challenge is high-level decision making: during descent, the lander must continuously trade off gathering information about terrain hazards, preserving divert options, and committing to a final landing site. In this paper we formalize such a powered descent decision-making (PDDM) problem as a belief Markov decision process whose objective is to maximize the probability of a safe landing. Directly solving the resulting belief- and set-valued optimal control problem is intractable, so we propose a reachability-steering guidance algorithm with a one-step utility function with precomputed constrained controllable sets. The utility function balances exploitation of the current safest candidate site against the preservation of backup divert options, while the containment constraint in the visibility-safe controllable set guarantees recursive feasibility along the descent. Offline, the controllable sets are computed efficiently using constrained zonotopes combined with a lossless convexification of the PDG problem. Numerical simulations with stochastic safety belief evolution demonstrate that the proposed approach improves the probability of a safe landing relative to a greedy retargeting baseline. Moderate weighting of divert options yields trajectories that remain resilient to the stochastic safety belief process without incurring excessive conservatism in fuel usage.

AIAA SciTech Forum 2026

Mitsubishi Electric Research Laboratories, Inc.
201 Broadway, Cambridge, Massachusetts 02139

Powered Descent Decision Making: A Reachability-Steering Approach

Kento Tomita *, Purnanand Elango †, Abraham P. Vinod ‡, Stefano Di Cairano §, and Avishai Weiss¶
Mitsubishi Electric Research Laboratories, Cambridge, MA, 02139, USA

Classical powered descent guidance (PDG) algorithms focus on low-level trajectory optimization, such as minimizing propellant consumption subject to soft-landing constraints. However, in future missions to unexplored bodies, the dominant challenge is high-level decision making: during descent, the lander must continuously trade off gathering information about terrain hazards, preserving divert options, and committing to a final landing site. In this paper we formalize such a powered descent decision-making (PDDM) problem as a belief Markov decision process whose objective is to maximize the probability of a safe landing. Directly solving the resulting belief- and set-valued optimal control problem is intractable, so we propose a reachability-steering guidance algorithm with a one-step utility function with precomputed constrained controllable sets. The utility function balances exploitation of the current safest candidate site against the preservation of backup divert options, while the containment constraint in the visibility-safe controllable set guarantees recursive feasibility along the descent. Offline, the controllable sets are computed efficiently using constrained zonotopes combined with a lossless convexification of the PDG problem. Numerical simulations with stochastic safety belief evolution demonstrate that the proposed approach improves the probability of a safe landing relative to a greedy retargeting baseline. Moderate weighting of divert options yields trajectories that remain resilient to the stochastic safety belief process without incurring excessive conservatism in fuel usage.

I. Introduction

Autonomous safe landing on unexplored planets or small bodies inherently poses a decision-making problem. In these scenarios, the locations of safe landing sites are not known a priori. Instead, the spacecraft must sense the terrain during descent and, based on incrementally revealed information about the target surface, iteratively steer toward safer-appearing regions. This process must be repeated under strict resource constraints, such as limited fuel and actuator capability. In practice, both the quality of terrain sensing and the maneuverability of a lander are limited. Consequently, it is impractical to exhaustively evaluate landing safety across the terrain and simply select the safest landing target. As the lander descends, it faces a trade-off between the range of observable and reachable terrain and the quality of safety information; more precise information becomes available at lower altitudes, but the set of reachable and visible landing sites diminishes correspondingly. Managing this trade-off constitutes a decision-making problem, which we refer to as the Powered Descent Decision-Making (PDDM) problem [1]. In this work, we focus on solving PDDM with the objective of maximizing the *probability of a safe landing (PSL)*, defined as the probability that the terminal landing site is free of hazards.

Modeling and solving the PDDM problem is nontrivial. Conventional powered descent guidance (PDG) problems are typically posed as optimal control problems, where the objectives are directly linked to control commands and lander states (e.g., minimizing fuel consumption while being close to a predetermined landing site), which enables relatively straightforward formulations [2]. In contrast, PDDM under a safety-maximization objective must account for terrain topography, which is an external and uncertain state. The intermediate lander state and control inputs interact with this topography through visibility, reachability, and sensing quality. As a result, formulating a suitable optimization problem that incorporates these interdependencies is not straightforward. Moreover, even if such a formulation is established, it inherently involves belief states about landing safety and reachable and visible sets of the terrain for each lander

*Research Scientist
†Research Scientist
‡Principal Research Scientist
§Distinguished Research Scientist
¶Senior Principal Research Scientist

state. Solving such a set-based optimal control problem with a nonlinear safety belief field is most likely going to be computationally intractable for onboard computational resources.

The present paper develops a formal decision-making framework for maximizing the PSL. First, we present a problem formulation that models PDDM as a belief Markov decision process (belief MDP), capturing belief evolution in landing safety and its coupling with lander dynamics, sensing opportunities, and terrain visibility. Second, we introduce a reachability-steering method where a heuristic objective function is proposed with recursive feasibility based on the surface points visible and reachable from next-observation state. Third, we present a tractable solution approach built on controllable sets represented as *constrained zonotopes* (CZs). We decompose the original nonlinear and belief-dependent trajectory optimization problem into two computationally efficient components: (i) a small-scale nonlinear optimization problem for selecting a target region, and (ii) a convex optimization problem for set-based powered-descent guidance. Finally, we validate the proposed method through extensive simulations using a representative simulated test environment, demonstrating its performance, robustness, and statistical behavior.

In summary, this paper contributes: *(i)* a PDDM problem formulation in a belief MDP framework, capturing uncertainty in terrain safety and its interaction with lander dynamics; *(ii)* a reachability-steering solution based on a one-step heuristic function with recursive feasibility; *(iii)* a principled decomposition of the PDDM optimization problem into a nonlinear target-region selection step and a convex set-based guidance step with an outer-loop iteration for the recursive feasibility; and *(iv)* extensive statistical evaluation on representative scenarios that demonstrate the effectiveness of the proposed approach.

A. Background and Related Works

1. Hazard Detection and Avoidance for Planetary Landing

Early planetary landers lacked onboard hazard-detection and avoidance (HDA) systems, so risk had to be mitigated before launch. Engineers developed the vehicles so that they can tolerate moderate rocks and slopes [3], and mission planners restricted landing ellipses to exceptionally smooth terrain, insisting that roughly 95–99% of each ellipse be obstacle-free [4, 5]. Because these safety predictions were drawn from coarse orbital imagery and statistical investigations, they often gave incomplete estimates of the true surface hazards, resulting sometimes in unsafe touchdowns, which has driven the development of online HDA technologies [6, 7].

Two technical capabilities are essential for autonomous landing: autonomous Precision Landing (PL) and autonomous Hazard Detection and Avoidance (HDA) [8]. Central to autonomous PL is the need for detailed reconnaissance data that provides preferred landing sites and precise navigation relative to the selected sites. Meanwhile, autonomous HDA hinges on the capabilities of hazard detection sensors and algorithms tailored to assess and mitigate landing risks in real time. The demand for autonomous PL and HDA technologies is influenced by the amount of prior knowledge about the terrain and the available sensor systems on the spacecraft. The technical capabilities of autonomous PL and HDA have been demonstrated in real missions; autonomous PL was first demonstrated by NASA's Mars 2020 mission, while autonomous HDA was first demonstrated by CNSA's Chang'e-3 lunar lander [9, 10]. As the feasibility of autonomous spacecraft landing is increasingly demonstrated, the demand for even more advanced capabilities grows, with the aim of achieving anytime and anywhere global and safe landing capabilities across the solar system—a milestone proposed by NASA's Precision Landing and Hazard Avoidance (PL&HA) program [11, 12]. In this work we focus on developing a guidance algorithm for autonomous HDA.

We define an HDA guidance algorithm as a method that adjusts the descent trajectory of a planetary lander equipped with hazard detection capabilities to maximize the probability of a safe landing, which is a new class of guidance problems. We briefly review conventional PDG algorithms for planetary landers first, followed by a review of recent studies related to the HDA guidance problem.

2. Conventional PDG Algorithms

Conventional PDG algorithms for planetary landing aim at providing the optimal command profile that minimizes propellant consumption while satisfying specified landing conditions and constraints. Such guidance algorithms can be categorized into two broad classes: closed-form guidance and computational guidance.

Closed-form PDG algorithms compute state-feedback control commands using an analytical thrust acceleration vector, achieving strong efficiency without significantly compromising performance. A representative example is the classical optimal feedback guidance based on zero-effort-miss (ZEM) and zero-effort-velocity (ZEV) [13]. Due to its simple structure and tractability, ZEM and ZEV approaches have been widely exploited in mission planning, system

design, and landing site selection. Recent works in this category have achieved improved performance and robustness by utilizing analytic gravity-turn trajectories [14] or by modeling the guidance sequence in multiple subphases [15].

Computational PDG algorithms, in contrast, transform the optimal control problem into a trajectory optimization problem to be solved numerically. These include indirect and direct methods. Indirect methods often achieve lower cost but require careful initialization and are sensitive to modeling errors. A recent example is the G-POLAR algorithm [16], which solves an eight-variable zero-finding problem with a smoothed optimization landscape. Direct methods using convex optimization [17–19] have become especially prominent thanks to lossless convexification techniques. These methods reformulate the PDG problem into a sequence of second-order cone programs (SOCPs), with guaranteed convergence within a finite number of iterations and computational efficiency compatible with real-time implementation.

3. HDA Guidance Algorithms

HDA guidance seeks to optimize a lander's descent trajectory so as to maximize the probability of a safe landing when the locations of hazard-free sites are uncertain. The core difficulty stems from limited sensor resolution and accuracy: acquiring submeter-scale topography from altitudes of 1–2 km is challenging, and sensors small and light enough for planetary missions often can only offer coarse data early in the descent. Consequently, the terrain topography, and therefore the safety map, remains uncertain until sufficiently close to the surface. HDA guidance is therefore a sequential decision-making problem: control actions must simultaneously gather information, preserve reachability of promising sites, and maximize the probability of a safe landing.

Greedy retargeting represents the simplest strategy. At every update, the lander selects the best site still reachable with the available fuel. Rogata et al. [20] implemented such a scheme using the E-Guidance law; Zhao et al. [21] extended the idea with low-resolution imagery and terminal sliding-mode control. While computationally light, these approaches can be overly myopic when early estimates of terrain safety are unreliable.

Deferred decision trajectory optimization (DDTO) is another related approach [22, 23]. It maintains reachability to a set of candidate targets/landing sites for as long as possible to improve resilience to unmodeled disturbances and to provide time to gather information about target viability. DDTO has been extended to adaptively update the target collection [24] and has been integrated into a closed-loop, set-based optimal control framework [25]. In related work, Srinivas *et al.* [26] use precomputed controllable sets to enable rapid retargeting. However, DDTO approaches optimize for selected landing-site candidates, rather than concurrently optimizing for both the trajectory and the selection of landing sites to maximize the probability of safe landing.

Learning-based methods have also been proposed. Iiyama *et al.* [27] treated HDA as a POMDP and trained a deep reinforcement learning agent to jointly select targets and tune guidance parameters. While capable of handling high-dimensional beliefs, DRL-based methods require extensive simulations in environments representative of all potential terrains—an impractical requirement for risk-averse landing missions.

A closely related line of work by some of the authors introduced the first predictive HDA guidance formulation under stochastic safety-map uncertainty [28]. The HDA problem was posed as a belief-dependent optimization problem, and two predictive guidance strategies, greedy and reachability-steering, were developed using Gaussian random fields and neural-network–based reachable-set evaluation. Reference [28] highlighted the importance of forecasting sensing quality and reachability structure, but relied on coarse reachable set approximations and lacked real-time capability. The subsequent study [1] introduced an efficient reachability-steering algorithm based on the decomposition with the target region selection and the constrained trajectory optimization, but lacked formal guarantees. The present paper builds on both of these preliminary efforts, replacing the earlier approximations with a formal decision-making framework, a set-based feasibility analysis using constrained zonotopes, and a recursively feasible integration with powered-descent guidance.

II. Preliminaries

This section summarizes the PDG models and computational tools used throughout the paper. Section II.A reviews the 3-DOF powered-descent dynamics and state and input constraints. Section II.B recalls the standard lossless convexification framework for PDG. Section II.C introduces a recently proposed dynamic-programming-based perspective with an augmented fuel-to-go state, and Section II.D reviews constrained zonotopes and set operations used for tractable controllable set computations.

A. Powered Descent Dynamics and Constraints

The 3-DOF motion of a planetary lander in terminal descent is often described in a topocentric Cartesian coordinate frame by the system of equations

$$\dot{r} = v, \quad \dot{v} = T/m + g, \quad \dot{m} = -\alpha_{\text{mf}}\|T\|, \tag{1}$$

where $r = [r_x, r_y, r_z] \in \mathbb{R}^3$ is the position, $v = [v_x, v_y, v_z] \in \mathbb{R}^3$ is the velocity, $m \in \mathbb{R}$ is the mass, $T = [T_x, T_y, T_z] \in \mathbb{R}^3$ is the thrust force, $g \in \mathbb{R}^3$ is the constant gravitational acceleration, and $\alpha_{\text{mf}} > 0$ is a mass-flow parameter. The thrust vector is constrained by

$$0 < T_{\min} \le \|T\| \le T_{\max}, \quad \|T\| \cos\theta_p \le T_z, \tag{2}$$

where $T_{\min}$ and $T_{\max}$ are the minimum and maximum thrust magnitudes, and θ_p is the maximum thrust pointing angle. The terminal conditions for soft landing are defined by $m_{\text{dry}} \le m(t_f)$, $r_z(t_f) = 0$, and $v(t_f) = 0$, where m_{dry} is the dry mass and $r_z(t_f)$ is the touchdown altitude. To avoid the lander approaching too close to the surface before landing, we enforce the following affine glide slope constraint [2]

$$H_{\text{gs}}\left(r(t) - r(t_f)\right) \le \mathbf{0}, \tag{3}$$

where $\{x \in \mathbb{R}^3 \mid \text{row}_j(H_{\text{gs}})\ x \le 0, \forall j\}$ represents intersection of half-spaces with a maximum glide slope angle θ_{gs} from the x-y plane. For example, a four-sided (i.e., upside-down pyramid) glide slope constraint is given by $\text{row}_j(H_{\text{gs}}) = [\cos\theta_j \sin\theta_{\text{gs}},\ \sin\theta_j \sin\theta_{\text{gs}},\ -\cos\theta_{\text{gs}}]$ where $\theta_j = (\pi/4)\ j$ for $j \in \{0, 1, 2, 3\}$. Alternatively, one can enforce a conic glide slope constraint, as is done in [29].

B. Lossless Convexification via Pontryagin Maximum Principle

The following lossless convexification (LCvx) technique is often used to obtain a convex PDG problem [2]. Define a new lander state $x = [r; v; z]$ with the change of variables $u := T/m$, $\sigma := \|T\|/m$, and $z := \ln m$ to obtain the linear dynamics

$$\dot{r} = v, \qquad \dot{v} = u + g, \qquad \dot{z} = -\alpha_{\text{mf}}\,\sigma. \tag{4}$$

With the log-mass transformation, we obtain the thrust bounds $\sigma \in [\sigma_{\min}(z), \sigma_{\max}(z)]$, where $\sigma_{\min}(z) = T_{\min}e^{-z}$ and $\sigma_{\max}(z) = T_{\max}e^{-z}$. We approximates these bounds as constants $\tilde{\sigma}_{\min} = T_{\min}/m_{\text{dry}}$ and $\tilde{\sigma}_{\max} = T_{\max}/m_{\text{wet}}$, which does not incur a significant penalty when realistic numerical parameters are used [25]. We can also construct z-dependent convex bounds via quadratic approximation at the price of an increased complexity [17].

The remaining nonconvex constraint $\|u\| = \sigma$ is relaxed to $\|u\| \le \sigma$, resulting in the convex state and control constraints for $t \in [t_0, t_f)$:

$$\|u\| \le \sigma \tag{5a}$$
$$\sigma\cos\theta_p \le u_z \tag{5b}$$
$$\tilde{\sigma}_{\min} \le \sigma \le \tilde{\sigma}_{\max} \tag{5c}$$
$$H_{\text{gs}}\left(r(t) - r(t_f)\right) \le \mathbf{0} \tag{5d}$$

with the terminal set for soft landing

$$\mathcal{X}_{\text{soft}} := \left\{x_f \mid r_{z,f} = 0,\ v_f = 0_3,\ \ln m_{\text{dry}} \le z_f\right\}. \tag{6}$$

We can show that minimizing fuel consumption $\int_{t_0}^{t_f} \sigma\ dt$ in the relaxed system drives the relaxed constraint $\|u\| \le \sigma$ to be tight, and recovers the optimal solution for the original problem with the approximated thrust bounds, using Pontryagin's Maximum Principle [29].

C. Lossless Convexification via Dynamic Programming

Similar lossless convexification results can be obtained using Dynamic Programming [25, 30], with a slightly increased conservativeness. Consider a discrete time model for the linear dynamics (4),

$$x^+_{i+1} = A\,x^+_i + B\,u^+_i + d, \tag{7}$$

where $u^+ = [u; \sigma]$ is the input, and the additional state c of the augmented state $x^+ = [r; v; z; c]$ representing the *cost-to-go* (fuel-to-go) state. With zero-order-hold discretization, the matrices A, B and vector d are obtained as

$$A = \begin{bmatrix} I_3 & \Delta t\, I_3 & 0 & 0 \\ 0 & I_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} \frac{1}{2}\Delta t^2 I_3 & 0 \\ \Delta t\, I_3 & 0 \\ 0 & -\alpha_{\mathrm{mf}}\Delta t \\ 0 & -\alpha_{\mathrm{mf}}\Delta t \end{bmatrix}, \quad d = \begin{bmatrix} \frac{1}{2}\Delta t^2 g \\ \Delta t\, g \\ 0 \\ 0 \end{bmatrix}, \tag{8}$$

which gives $c_{i+1} = c_i - \alpha_{\mathrm{mf}}\Delta t\, \sigma_i$. Then, we replace the control input lower bound of $\tilde{\sigma}_{\min} \le \sigma_i$ with the conservative approximation

$$\tilde{\sigma}_{\min} \le u_{i,z}. \tag{9}$$

Next, we recursively construct a *controllable set*, the set of all initial states from which a given terminal state set is reachable with all state and control constraints (see Chapter 10 of [31]). Here, we consider the terminal set for pinpoint landing, where the terminal state is at the origin with zero velocity and $m_f \ge m_{\mathrm{dry}}$. At termination, the fuel-to-go satisfies $c_f \ge 0$.

$$C_f^+ = \mathcal{X}^+_{\mathrm{pinpoint}} := \left\{ x_f \mid r_f = 0_3,\ v_f = 0_3,\ z_f \ge \ln m_{\mathrm{dry}},\ c_f \ge 0 \right\} \tag{10a}$$

$$C_i^+ = \mathrm{Proj}_{x^+}\left\{ [x^+; u^+] \in \mathcal{X}^+ \times \mathcal{U}^+ \;\middle|\; Ax^+ + Bu^+ + d \in C_{i+1}^+ \right\}, \quad i = N_f - 1, \ldots, 0, \tag{10b}$$

where $x_i^+ \in \mathcal{X}^+$ and $u_i^+ \in \mathcal{U}^+$ denote the state and control constraints (3), (5a), (5c), (9) for all $i \in \{0, \ldots N_f - 1\}$. The subscript f represents the discrete time step $i = N_f$, corresponding to $t = t_f$.

The minimum fuel guidance optimization, $\min_{u^+} \Sigma_i \sigma_i$, is equivalent to minimizing the initial cost-to-go variable c_0 with the state containment constraints with controllable sets:

$$\min_{u^+}\ c_0 \quad \text{s.t.} \quad x_{i+1}^+ = A\, x_i^+ + B\, u_i^+ + d, \quad x_{i+1}^+ \in C_{i+1}, \quad u_i^+ \in \mathcal{U}^+, \quad \forall i = 0, \ldots, N_f - 1 \tag{11}$$

and with dynamic programming we can show that the optimal solution satisfies the lossless property, i.e., $\|u_i\| = \sigma_i$ [25, 30].

Informally, the tightness of $\|u_i\| \le \sigma_i$ is enforced by minimizing c_i because (i) $c_i = c_{i+1} + \alpha_{\mathrm{mf}}\Delta t \sigma_i$ is a monotonically increasing function of σ_i and (ii) there is no additional lower bound on σ_i. This is why $\tilde{\sigma}_{\min} \le \sigma_i$ is replaced with $\tilde{\sigma}_{\min} \le u_{i,z}$. Note that c_i's role cannot be replaced by z_i because it is restricted by dynamics and boundary conditions, i.e., $z_0 = \ln m_{\mathrm{wet}}$ and $z_i \ge \ln m_{\mathrm{dry}}$ prohibit unconstrained minimization of z_i. It is possible to remove z_i to make the augmented state seven dimensional, but a boundary-condition feasibility check is required every step (see Remark 7 of [25]).

D. Constrained Zonotope for Controllable Set Recursion

The controllable-set recursion (10) requires repeatedly propagating sets backward through Minkowski sum with inputs, affine dynamics, and intersecting them with state constraints,

$$C_f^+ = \mathcal{X}^+_{\mathrm{pinpoint}}, \quad C_i^+ = \mathcal{X}^+ \cap A^{-1}\left(C_{i+1}^+ \oplus \left(-B\,\mathcal{U}^+ - d\right)\right), \quad \forall i = N_f - 1, \ldots, 0. \tag{12}$$

We can make the recursion (12) tractable by using *constrained zonotopes* (CZs), which provide an efficient polytope description compatible with the required set operations [25, 30].

A constrained zonotope is defined as

$$C\mathcal{Z} = \{\, x = G\xi + c_{\mathrm{cz}} \;\mid\; \|\xi\|_\infty \le 1,\ A_{\mathrm{eq}}\xi = b_{\mathrm{eq}} \,\}, \tag{13}$$

where G and c_{cz} specify the geometric shape and location, and the linear equalities restrict the latent parameter ξ so that (13) represents a full-dimensional polytope. Constrained zonotopes are equivalent to bounded polytopes, but admit closed-form expressions for affine maps, Minkowski sums, and intersections [32], which enable an exact and computationally efficient implementation of the controllable-set recursion (12). Additionally, we need appropriate box bounds on position r_i and velocity v_i to ensure bounded $\mathcal{X}^+$. The implementation is based on the recently released Python package, *pycvxset* [33].

III. Powered Descent Decision Making Problem

This section formulates the powered descent decision-making (PDDM) problem addressed in this paper. Section III.A introduces the terrain-safety belief model and the probabilistic objective based on sensor observations. Section III.B then formulates the resulting sequential decision-making problem.

In the following we consider two sampling periods for time discretization, one for guidance and the other for terrain sensing. We denote guidance time step sequence by $\mathcal{T}_{\text{guidance}} := \{t_i\}_{i\in\mathcal{I}}$ and terrain sensing time step sequence by $\mathcal{T}_{\text{sensing}} := \{t_k\}_{k\in\mathcal{K}}$. The subscript i and k are also used for variables corresponding to the guidance time or terrain sensing time sequences. We denote the time at touchdown by t_f and the final sensing time by t_N, and use the subscripts f and N for variables at these time instants. We assume $\mathcal{T}_{\text{sensing}} \subset \mathcal{T}_{\text{guidance}}$ and $t_N < t_f$. Also, we define $\mathcal{T}^k_{\text{guidance}}$ as the guidance time discretizations between two sensing time instants

$$\mathcal{T}^k_{\text{guidance}} = \begin{cases} \mathcal{T}_{\text{guidance}} \cap [t_k, t_{k+1}), & \text{if } k \neq N \\ \mathcal{T}_{\text{guidance}} \cap [t_N, t_f), & \text{otherwise.} \end{cases} \tag{14}$$

A. Terrain Sensing and Landing Safety

At the terrain sensing steps $t \in \mathcal{T}_{\text{sensing}}$, the estimated landing safety $\hat{s}_k$ over the terrain is updated. Let $s(\gamma) : \mathbb{R}^2 \to \{0, 1\}$ be the true landing safety for terrain's local horizontal coordinate $\gamma \in \mathbb{R}^2$, where 1 and 0 denote safe and unsafe, respectively. Terrain sensing provides estimated landing safety probability $\hat{p}_k(\gamma) := \mathbb{E}[s(\gamma)|\mathcal{G}_k]$ for any surface point within the sensor field of view (FOV), which is defined as a function of the lander state, $\mathcal{F}_k(x_k)$. Here, $\mathcal{G}_k = \sigma(\hat{s}_{0^-}, y_{[0:k]})$ is the filtration and $\hat{p}_k$ takes into account all the past measurements $y_{[0:k]}$. The $\hat{s}_k$ is updated as

$$\hat{s}_k(\gamma) = \begin{cases} \hat{p}_k(\gamma) & \text{if} \quad \gamma \in \mathcal{F}_k \\ \hat{s}_{k-1}(\gamma) & \text{otherwise.} \end{cases} \tag{15}$$

We assume that the reliability of the landing safety estimate is inversely proportional to the distance from the terrain.

Given the best-effort estimate of landing safety after all terrain sensing operations, $\hat{s}_N$, the optimal landing site is the safest reachable site at t_N. In other words, the landing site deterministically depends on $\hat{s}_N$, therefore our objective is reduced to the maximization of the reachable safest landing site at t_N. Formally, we define the *reachable surface* $\mathcal{R}_k$ as the set of landing sites reachable from x_k while satisfying all physical and operational constraints for landing. Then, our objective is

$$\max_{u} \mathbb{E}[J_{\text{safe}}], \quad \text{where} \quad J_{\text{safe}} := \begin{cases} \max_{\gamma\in\mathcal{R}_N} \hat{s}_N(\gamma) & \text{if } \mathcal{R}_N \neq \varnothing \\ 0 & \text{otherwise.} \end{cases} \tag{16}$$

where $\mathcal{R}_N$ is the set of reachable landing sites from x_N and J_{safe} is a random variable due to $\hat{s}_N$.

B. Optimal Hazard Detection and Avoidance Guidance Problem

The objective of HDA guidance is to maximize the probability of a safe landing, (16), which is a function of the final terrain-safety estimate $\hat{s}_N$. The lander does not have access to $\hat{s}_N$ until time t_N, but control actions must be selected sequentially as new sensing information becomes available. To make the connection between guidance and sensing clearer, let π_k denote a policy at time t_k that selects an ideal next sensing state set $\mathcal{X}^{k+1}_{\text{sensing}}$ based on the current lander state x_k and terrain safety estimate $\hat{s}_k$,

$$\mathcal{X}^{k+1}_{\text{sensing}} \leftarrow \pi_k(x_k, \hat{s}_k). \tag{17}$$

For a given sensing state set $\mathcal{X}^{k+1}_{\text{sensing}}$, the lander computes a dynamically feasible, minimum-fuel trajectory by solving a guidance optimization problem over the time grid $\mathcal{T}^k_{\text{guidance}}$. We define the corresponding *solution mapping*

$$\mathcal{S}_k \ : \ \left(x_k, \mathcal{X}^{k+1}_{\text{sensing}}\right) \to \left(u^\star_{[t_k,t_{k+1})}, \gamma'^\star_{k+1}, x_{k+1}\right), \tag{18}$$

where $u^{\star}_{[t_k,t_{k+1})}$, $\gamma'^{\star}_{k+1}$, and x_{k+1} are given by the solution of

$$\min_{\{u_i\},\,\gamma'} \quad \sum_{t_i \in \mathcal{T}^k_{\text{guidance}}} \|u_i\| \tag{19a}$$

$$\text{subject to} \quad x_{i+1} = f(x_i, u_i), \qquad \forall t_i \in \mathcal{T}^k_{\text{guidance}}, \tag{19b}$$

$$x_{i+1} \in \mathcal{X}(\gamma'), \;\; u_i \in \mathcal{U}, \qquad \forall t_i \in \mathcal{T}^k_{\text{guidance}}, \tag{19c}$$

$$x(t_{k+1}) \in \mathcal{X}^{k+1}_{\text{sensing}}. \tag{19d}$$

Here, $\gamma' \in \mathbb{R}^2$ denotes a virtual landing site, and the state constraint set $\mathcal{X}(\gamma')$ depends on γ' through the glide-slope constraint. Using $\mathcal{S}_k$ (18), the HDA guidance problem can be written as the following bilevel optimization problem,

$$\max_{\{\pi_k\}} \quad \mathbb{E}^{\pi}[J_{\text{safe}}] \tag{20a}$$

$$\text{subject to} \quad \mathcal{X}^{k+1}_{\text{sensing}} = \pi_k(x_k, \hat{s}_k), \qquad k = 0, \ldots, N-1, \tag{20b}$$

$$\left(u^{\star}_{[t_k,t_{k+1})}, \gamma'^{\star}_{k+1}, x_{k+1}\right) = \mathcal{S}_k\left(x_k, \mathcal{X}^{k+1}_{\text{sensing}}\right), \qquad k = 0, \ldots, N-1, \tag{20c}$$

$$\hat{s}_{k+1} \sim \mathscr{T}_k\left(\,\cdot\,|\hat{s}_k, x_{k+1}\right), \qquad k = 0, \ldots, N-1, \tag{20d}$$

where $\hat{s}_k$ is the landing safety belief state with the transition $\mathscr{T}_k$.

In (20), the upper-level problem selects the ideal next sensing state set through the policy π_k to maximize the terminal landing safety probability, while the lower-level problem computes a dynamically feasible, minimum-fuel trajectory that steers the lander to the selected sensing state set. Feasibility beyond the final sensing time t_N is implicitly enforced through the objective (16). Poorly designed policies π_k may result in a terminal state x_N from which no feasible landing trajectory exists, in which case the reachable safe landing set satisfies $\mathcal{R}_N = \varnothing$ and $J_{\text{safe}} = 0$.

IV. Reachability-Steering Approach

Problem (20) is modeled as a *belief Markov decision process (belief MDP)* where $\hat{s}$ is the belief state. In theory, the optimal policy π^*_k is obtained by solving the following Bellman recursion,

$$\begin{aligned} V^{\pi^*}_N(x_N, \hat{s}_N) &= \max_{\gamma \in \mathcal{R}_N} \hat{s}_N(\gamma) \\ V^{\pi^*}_k(x_k, \hat{s}_k) &= \max_{\mathcal{X}^{k+1}_{\text{sensing}}} \mathbb{E}_{\hat{s}_{k+1}}\left[V^{\pi^*}_{k+1}(x_{k+1}, \hat{s}_{k+1}) \,\middle|\, x_k, \hat{s}_k, \mathcal{X}^{k+1}_{\text{sensing}}\right] \\ &= \max_{\mathcal{X}^{k+1}_{\text{sensing}}} \int_{\hat{s}_{k+1}} \mathbb{P}(\hat{s}_{k+1} \mid x_k, \hat{s}_k, \mathcal{X}^{k+1}_{\text{sensing}}) V^{\pi^*}_{k+1}(x_{k+1}, \hat{s}_{k+1})\, d\hat{s}_{k+1}, \end{aligned} \tag{21}$$

however, solving (21) is intractable. Unlike typical POMDPs where the main issue is the infinite realizations of the continuous belief state, the challenge for HDA is the difficulty of modeling the safety map belief dynamics, $\mathbb{P}(\hat{s}_{k+1} \mid x_k, \hat{s}_k, \mathcal{X}^{k+1}_{\text{sensing}})$. For environments requiring online HDA, prior terrain knowledge is insufficient to construct or validate an accurate safety belief dynamics model, as doing so would require detailed information about terrain topography realizations and their associated sensing outcomes. Indeed, if the terrain were sufficiently well known to support a validated belief dynamics model a priori, online HDA would no longer be required. In the absence of such information, any belief dynamics model is necessarily assumption-driven. Accordingly, the HDA guidance solution is primarily shaped by the choice of these assumptions, and our approach seeks to minimize reliance on belief dynamics modeling while remaining resilient to uncertainty.

A. Utility Functions for Divert-Aware Resilient Guidance

A simple approach is to keep targeting the safest landing site. However, such a greedy policy is equivalent to the constant safety belief assumption (see Appendix VII.A), and there is a risk of steering toward an isolated site that later turns out unsafe. For instance, suppose there exists an isolated landing site γ_A with $\hat{s}_k(\gamma_A) = 0.8$, while an entire region $\{\gamma_{B_i}\}_i$ has moderate but consistent safety $\hat{s}_k(\gamma_{B_i}) = 0.7$. Greedy HDA would always steer toward γ_A as long as it is

reachable. However, if γ_A is later revealed unsafe, there is no alternate and the mission fails. By contrast, region B provides multiple divert options and could be less risky overall. This motivates an approach that balances the best available candidate and the number and quality of backup options.

1. One-Step Utility Function with Belief Propagation

To formalize the aforementioned trade-off, let $s_L \in [0, 1)$ be a minimum safety threshold for a site to qualify as a divert candidate. At step $k + 1$, one heuristic that approximates the value function $V^{\pi^*}(x_{k+1}, \hat{s}_{k+1})$ and also captures the balance between the safest site and other divert options is

$$J'_{k+1}(\mathcal{R}_{k+1}, \hat{s}_{k+1}) := J'_{\text{best}}(\mathcal{R}_{k+1}, \hat{s}_{k+1}) + \lambda_k J'_{\text{divert}}(\mathcal{R}_{k+1}, \hat{s}_{k+1}), \tag{22}$$

where $\lambda_k > 0$ balances the two terms

$$J'_{\text{best}}(\mathcal{R}_{k+1}, \hat{s}_{k+1}) = \max_{\gamma \in \mathcal{R}_{k+1}} \hat{s}_{k+1}(\gamma), \tag{23a}$$

$$J'_{\text{divert}}(\mathcal{R}_{k+1}, \hat{s}_{k+1}) = \int_{\mathcal{M}_{k+1}} \hat{s}_{k+1}(\gamma)\, d\gamma, \quad \mathcal{M}_{k+1} := \{\gamma \in \mathcal{R}_{k+1} : \hat{s}_{k+1}(\gamma) \geq s_L\}. \tag{23b}$$

The first term favors the most promising site, while the second rewards the total "safe probability mass" of divert options above threshold. However, $J'_{k+1}(\mathcal{R}_{k+1}, \hat{s}_{k+1})$ requires propagating the belief to obtain $\hat{s}_{k+1}$ and we propose an approximation for tractability.

2. One-Step Utility Function without Belief Propagation

Because $\hat{s}_{k+1}$ is unknown at step k, we approximate it using the current map $\hat{s}_k$ but only over the next sensor footprint $\mathcal{F}_{k+1}(x_{k+1})$, reflecting that (i) only points in $\mathcal{F}_{k+1}$ will be updated and (ii) their estimates will be more reliable at lower altitude. This yields the surrogate

$$\tilde{J}_{k+1}(\mathcal{R}_{k+1}, \mathcal{F}_{k+1}, \hat{s}_k) := \tilde{J}_{\text{best}}(\mathcal{R}_{k+1}, \mathcal{F}_{k+1}, \hat{s}_k) + \lambda_k \tilde{J}_{\text{divert}}(\mathcal{R}_{k+1}, \mathcal{F}_{k+1}, \hat{s}_k), \tag{24}$$

with

$$\tilde{J}_{\text{best}}(\mathcal{R}_{k+1}, \mathcal{F}_{k+1}, \hat{s}_k) = \max_{\gamma \in \mathcal{R}_{k+1} \cap \mathcal{F}_{k+1}} \hat{s}_k(\gamma) \tag{25a}$$

$$\tilde{J}_{\text{divert}}(\mathcal{R}_{k+1}, \mathcal{F}_{k+1}, \hat{s}_k) = \int_{\tilde{\mathcal{M}}_{k+1}} \hat{s}_k(\gamma)\, d\gamma, \quad \tilde{\mathcal{M}}_{k+1} = \{\gamma \in \mathcal{R}_{k+1} \cap \mathcal{F}_{k+1} : \hat{s}_k(\gamma) \geq s_L\}. \tag{25b}$$

However, the score $\tilde{J}_{k+1}$ is well-defined only if $\mathcal{R}_{k+1} \cap \mathcal{F}_{k+1} \neq \emptyset$. For example, the lander may have such a large horizonal velocity that no reachable landing site lies inside the nadir sensor footprint. Even if the current state x_k is not ill-posed, there is no guarantee that from x_{k+1} onward $\mathcal{R}_{j+1} \cap \mathcal{F}_{j+1}$ remains nonempty. Therefore, without additional safeguards, recursive optimization could fail at a later step.

3. One-Step Utility Function with Recursive Feasibility

To enforce feasibility guarantee over subsequent optimization steps, we embed visibility into the operational constraints. Consider a nadir-pointing sensor FOV of a pyramid shape with the footprint:

$$\mathcal{F}_k = \{\gamma \in \mathbb{R}^2 \mid H_{\text{fov}} \left(r_k - r_f(\gamma)\right) \leq \mathbf{0}\} \tag{26}$$

where $r_f(\gamma)^\top = [\gamma_x, \gamma_y, 0]$ and $\text{row}_j(H_{\text{fov}}) = [\cos\theta_j \cos(\theta_{\text{fov}}/2),\ \sin\theta_j \cos(\theta_{\text{fov}}/2),\ -\sin(\theta_{\text{fov}}/2)]$ and $\theta_j = (\pi/4)j$ for $j \in \{0, 1, 2, 3\}$. Equation (26) shows that restricting the landing site to be within sensor footprint is mathematically equivalent to the glide slope constraint (3) with θ_{gs} replaced by $(\pi - \theta_{\text{fov}})/2$. Therefore, by tightening the glide slope angle to $\theta'_{\text{gs}} := \max\left(\theta_{\text{gs}},\ \pi/2 - \theta_{\text{fov}}/2\right)$, we obtain the following *visibility-safe reachable surface*, which is a set of reachable terminal landing sites that are visible at all times onward.

$$\bar{\mathcal{R}}_k := \left(\mathcal{R}_k \text{ with glide slope angle } \theta'_{\text{gs}}\right) \subseteq \mathcal{F}_k. \tag{27}$$

Since $\bar{\mathcal{R}}_{k+1} \subseteq \mathcal{F}_{k+1}$, the explicit FOV mask in (24) is redundant once the tightened glide slope constraint is in force. This leads to the final surrogate used by the Reachability–Steering HDA:

$$J_{k+1}(\bar{\mathcal{R}}_{k+1}, \hat{s}_k) := J_{\text{best}}(\bar{\mathcal{R}}_{k+1}, \hat{s}_k) + \lambda_k J_{\text{divert}}(\bar{\mathcal{R}}_{k+1}, \hat{s}_k), \tag{28}$$

with

$$J_{\text{best}}(\bar{\mathcal{R}}_{k+1}, \hat{s}_k) = \max_{\gamma \in \bar{\mathcal{R}}_{k+1}} \hat{s}_k(\gamma) \tag{29a}$$

$$J_{\text{divert}}(\bar{\mathcal{R}}_{k+1}, \hat{s}_k) = \int_{\bar{\mathcal{M}}_{k+1}} \hat{s}_k(\gamma)\, d\gamma, \quad \bar{\mathcal{M}}_{k+1} = \{\gamma \in \bar{\mathcal{R}}_{k+1} : \hat{s}_k(\gamma) \geq s_L\}. \tag{29b}$$

Remark IV.1 (Adaptive choice of s_L) *The threshold s_L may be chosen adaptively from the current belief map, e.g., as the q-quantile of $\{\hat{s}_k(\gamma) \mid \gamma \in \bar{\mathcal{R}}_k\}$. This ensures that $\bar{\mathcal{M}}_{k+1}$ is nonempty, while still emphasizing sites with relatively high safety probability.*

B. Reachability steering

Using the utility function $J_{k+1}(\bar{\mathcal{R}}_{k+1}, \hat{s}_k)$ (28), we formalize a *reachability-steering* approach for the HDA guidance problem (20). At each sensing time t_k, we recursively select the next sensing state x_{k+1} that maximizes $J_{k+1}(\bar{\mathcal{R}}_{k+1}, \hat{s}_k)$, which depends on x_{k+1} through $\bar{\mathcal{R}}_{k+1}$. With the solution mapping (18), we propose the following reachability-steering guidance

$$\max_{x_{k+1}} \quad J_{k+1}\big(\bar{\mathcal{R}}_{k+1}, \hat{s}_k\big) \tag{30a}$$

$$\text{subject to} \quad \Big(u^{\star}_{[t_k, t_{k+1})}, \gamma'^{\star}_{k+1}, x_{k+1}\Big) = \mathcal{S}_k(x_k, \{x_{k+1}\})\,, \tag{30b}$$

for $k = 0, \ldots, N-1$. The constraint (30b) restricts the next sensing state x_{k+1} to be reachable from x_k.

V. Real-Time Reachability-Steering Algorithm

The reachability steering (30) is a challenging noncovex optimization problem, and we propose a tractable solution approach based on decomposition and iteration. Namely, we decompose the master problem (30) into two computationally tractable subproblems: (i) a target region selection (TRS) problem (§V.A) and (ii) a constrained minimum fuel guidance problem (§V.B), with an outer-loop iteration for the recursive feasibility (§V.C).

A. Target Region Selection

We parameterize a candidate reachable surface, which we call a *target region*, by a convex polygon circumscribing a circle of diameter p_d located at (p_x, p_y). Formally, we consider the parameter $p = (p_x, p_y, p_d)$ for the target region $\mathcal{R}_{\text{TRS}}(p)$:

$$\mathcal{R}_{\text{TRS}}(p) = \left\{ (\gamma_x, \gamma_y) \,\middle|\, H_n \begin{bmatrix} \gamma_x - p_x \\ \gamma_y - p_y \end{bmatrix} \leq \frac{p_d}{2} \mathbf{1}_n \right\}, \tag{31}$$

where $\text{row}_j(H_n) = [\cos\theta_j, \sin\theta_j]$ with $\theta_j = 2\pi j/n$ for $j = 0, \ldots, n-1$.

Then, the target region selection (TRS) problem is defined as

$$\max_{p} \quad J_{k+1}\left(\mathcal{R}_{\text{TRS}}(p), \hat{s}_k\right) \tag{32a}$$

$$\text{subject to} \quad \mathcal{R}_{\text{TRS}}(p) \subseteq \mathscr{R}_{k+1}, \quad 0 \leq p_d \leq \beta D_k, \tag{32b}$$

where $D_k = 2r_{z,k}\tan(\theta_{\text{FOV}}/2)$ is a projected FOV diameter at time t_k. Here, β and $\mathscr{R}_{k+1} \subset \mathbb{R}^2$ are given parameters, which are regulated in relation to the constrained minimum fuel guidance problem (§V.B) to enforce HDA guidance feasibility. See §$V.C$ for more details.

For the given parameters, an optimal p_d is determined to be its upper bound as the objective is non-decreasing in p_d due to $\hat{s}_k \geq 0$. The remaining parameters (p_x, p_y) are then efficiently estimated using greedy search with image-based filtering algorithms.

B. Constrained Minimum Fuel Guidance

Given the current state x_k and the selected target region $\mathcal{R}_{\text{TRS}}$, we solve for minimum fuel guidance such that *any surface point* on selected $\mathcal{R}_{\text{TRS}}$ is visible and reachable from the next observation state x_{k+1}. Such a constraint is well described with controllable sets (§II.C). Here, we define the *visibility-safe controllable set* similarly to the reachable surface (27):

$$\bar{C}_i^+ := \left(C_i^+ \text{ (10) with glide slope angle } \theta'_{\text{gs}} \right). \tag{33}$$

Next, we define a *regional controllable set*, which is a set of initial state from which *any point* within a target region is visible and reachable. Such a set can be defined with Pontryagin difference by exploiting the translational invariance of $\bar{C}_i^+$. With a slight abuse of notation, we define the regional controllable set for the selected target region as

$$\begin{aligned} \bar{C}_i^+(\mathcal{R}_{\text{TRS}}) := \bar{C}_i^+ \ominus \mathcal{R}_{\text{TRS}} &= \cap_{\gamma \in \mathcal{R}_{\text{TRS}}} \bar{C}_i^+ - \iota(\gamma) \\ &= \cap_{\gamma \in V_{\text{TRS}}} \bar{C}_i^+ - \iota(\gamma), \end{aligned} \tag{34}$$

where $\iota(\gamma) := E_2 \gamma'$ injects γ into the full-dimension space with $E_2 := [I_{2\times 2} |\ 0_{2\times 6}]^\top$, and V_{TRS} denotes the target region's vertices. The second equality follows the definition of Pontryagin difference, and the last equality is known for a convex and bounded polyhedron (see Theorem 2.1(xiii) of Reference [34]).

Finally, we define the constrained minimum-fuel guidance problem as a following convex optimization probelm utilizing the set-based lossless convexification (§II.C)

$$\min_{u_i^+} \quad c(t_k) \tag{35a}$$

$$\text{subject to} \quad x_{i+1}^+ = A x_i^+ + B u_i^+ + d, \qquad \forall t_i \in \mathcal{T}_{\text{guidance}}^k \tag{35b}$$

$$x_{i+1}^+ \in \mathcal{X}^+(\gamma_p), \quad u_i^+ \in \mathcal{U}^+, \qquad \forall t_i \in \mathcal{T}_{\text{guidance}}^k \tag{35c}$$

$$x(t_{k+1}) \in \bar{C}_{k+1}^+ - \iota(\gamma_j), \qquad \forall \gamma_j \in V_{\text{TRS}}, \tag{35d}$$

where the temporal landing site is set $\gamma_p := (p_x, p_y)$ from the TRS problem output (32). Problem (35) can be solved with SOTA real-time-capable convex solvers on the order of milliseconds [35, 36], which is orders of magnitude faster than the time scale of the terrain sensing updates [37].

C. Outer-Loop for Recursive Feasibility

The constrained minimum-fuel guidance (35) can be infeasible depending on the proposed target region $\mathcal{R}_{\text{TRS}}$; unlike the original form of reachability-steering guidance (30), $\mathcal{R}_{\text{TRS}}$ is not strictly tied to x_k, and some landing sites within $\mathcal{R}_{\text{TRS}}$ could be not reachable. To this end, we restrict the target region domain within the region selected in the previous step, and shrink the admissible diameter by updating β upon infeasibility,

$$\mathscr{R}_{k+1} \leftarrow \mathcal{R}_{\text{TRS}} \text{ for } \mathcal{X}_{\text{sensing}}^k, \quad k = 1, \ldots, N-1 \tag{36a}$$

$$\beta \leftarrow \kappa\beta \quad \text{if Subproblem (35) is infeasible,} \tag{36b}$$

where $\kappa \in (0, 1)$. Equation (36a) is an iteration over sensing opportunities $k = 1, \ldots, N-1$, but Eq. (36b) occurs for each proposed target region $\mathcal{R}_{\text{TRS}}$ and subsequent guidance subproblem (35) within each sensing opportunity k. Therefore, β is initialized to be 1 for each k. The initial admissible region $\mathscr{R}_1$ should be an appropriate approximation of the reachable surface, and can be iteratively shrank upon observing infeasibility. In summary, the resulting algorithm alternates between target-region selection and convex optimal guidance, adaptively shrinking the region only when necessary to recover feasibility. The overall flow is shown in the flowchart of Fig. 1.

VI. Numerical Demonstration

A. Test Scenarios

We evaluate the proposed reachability-steering guidance law in a simplified hazard environment that captures the key trade-offs between exploitation and exploration. The test terrain contains one isolated candidate landing site A and three closely clustered sites B1–B3, as depicted in Fig. 2. Only these four regions are assigned nonzero probability of being safe; the remainder of the surface is treated as unsafe for landing.

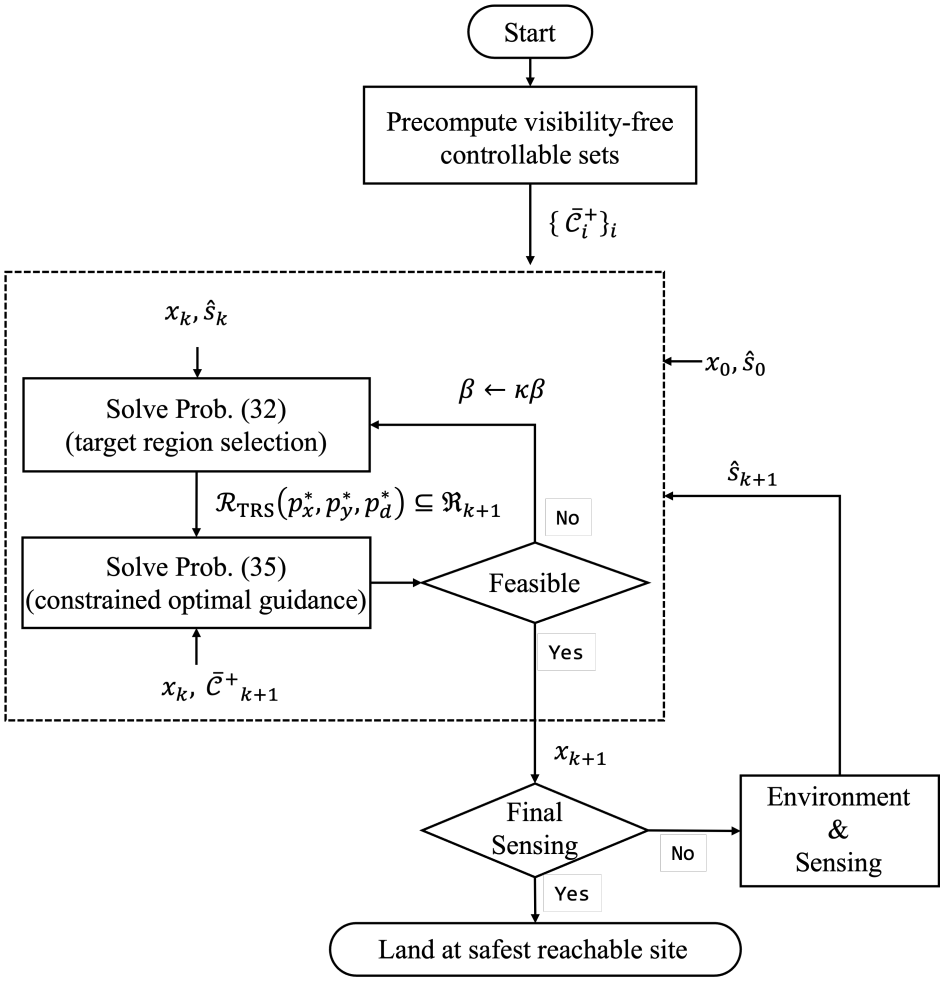

Fig. 1 Flowchart of reachability-steering HDA guidance algorithm.

The lander descends from a high altitude with limited field of view, and its belief about the safety of each site evolves as new information becomes available. To model the evolution of perceived landing safety during descent, each site is assigned a probability of safety $\hat{s}_\ell \in [0, 1]$ at each discrete altitude level indexed by ℓ. The probability is updated according to a bounded random walk with state-dependent drift and diffusion,

$$\hat{s}_{\ell+1} = \hat{s}_\ell + \mu(\hat{s}_\ell)\ \Delta h + \sigma(\hat{s}_\ell)\ \sqrt{\Delta h}\,\xi_\ell, \tag{37a}$$

$$\mu(s) = \epsilon\ \ln\left(\frac{s}{1-s}\right), \tag{37b}$$

$$\sigma(s) = \sigma_0 + \sigma_1 H(s), \qquad H(s) = -s\log_2 s - (1-s)\log_2(1-s), \tag{37c}$$

where $\Delta h > 0$ is the altitude discretization step, and $\xi_\ell \sim \mathcal{N}(0, 1)$ are independent Gaussian increments. The drift term $\mu(s)$, scaled by a small gain $\epsilon > 0$, biases the evolution toward more decisive beliefs ($s \approx 0$ or $s \approx 1$) as the altitude decreases. The diffusion term $\sigma(s)$ combines a baseline noise level $\sigma_0 > 0$ with an entropy-dependent component $\sigma_1 H(s)$ that increases uncertainty when the belief is ambiguous ($s \approx 0.5$) and decreases it as s approaches the extremes. After each update, the probability is clipped to the interval $[0.05,\ 0.95]$ to ensure physically meaningful values. In our implementation, we take $\epsilon = 10^{-3}$, $\sigma_0 = 0.05$, and $\sigma_1 = 0.10$.

Each landing site belief evolves independently according to (37), but the sites begin with different initial probabilities of safety. We consider three representative scenarios—A09, A08, and A07—which differ only in the initial value assigned to the isolated site `A`. Specifically, $\hat{s}_0^A$ is set to 0.9, 0.8, and 0.7 in scenarios A09, A08, and A07, respectively. The clustered sites `B1`, `B2`, and `B3` all begin with the same nominal initial value $\hat{s}_0^{B_i} = 0.6$, but their stochastic evolutions

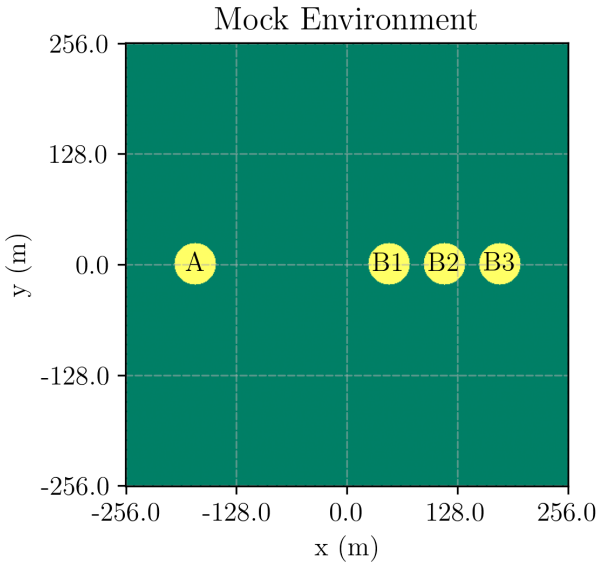

Fig. 2 Mock landing environment with one isolated site `A` and three clustered sites `B1–B3`. Only these regions carry nonzero safety probability.

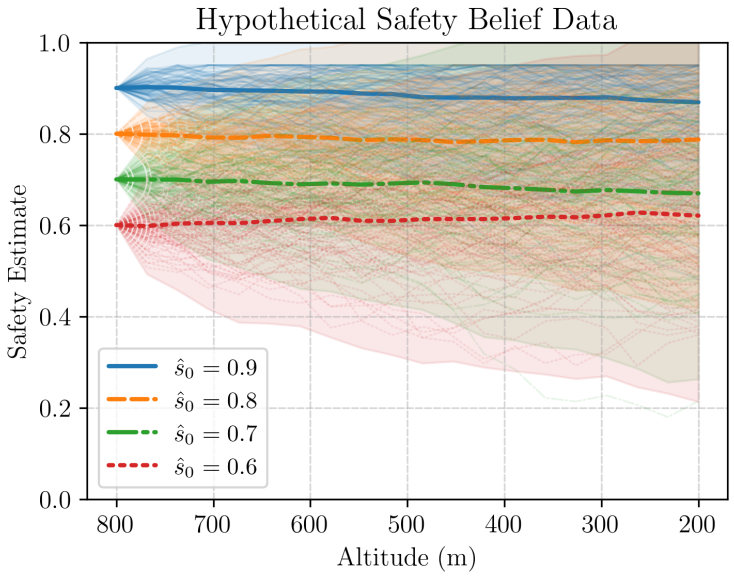

Fig. 3 Sample realizations of the stochastic safety model (37) **for different initial probabilities.**

Table 1 Initial probability of safety for each landing site in the three test scenarios. All sites evolve independently according to the stochastic model (37)**.**

Scenario	$\hat{s}_0^A$	$\hat{s}_0^{B1}$	$\hat{s}_0^{B2}$	$\hat{s}_0^{B3}$
A09	0.9	0.6	0.6	0.6
A08	0.8	0.6	0.6	0.6
A07	0.7	0.6	0.6	0.6

Parameter	Value
r_0	(0, 0, 800) m
v_0	(-10, 0, -30) ms^{-1}
m_{dry}	1905 kg
$t_f - t_0$	80 s
θ_{p}	50 deg
θ_{gs}	10 deg
θ_{fov}	30 deg
g	1.625 ms^{-2}
T_{max}	8400 N
T_{min}	2100 N
α_{mf}	0.00115

Table 2 Simulation Parameters

are independent. The initial conditions for all cases are summarized in Table 1. Figure 3 visualizes sample realizations of the stochastic model (37) for each scenario, showing how safety beliefs sharpen or fluctuate as the lander descends.

We run 100 Monte Carlo simulations using the parameters given in Table 2. For time discretizations, we use $\Delta t = 4$ s for guidance times t_i and $\Delta t = 20$ s for sensing times t_k. The first terrain sensing is made at t_0, and the final sensing is made at $t_N = 60$ sec, totaling four sensing opportunities.

B. Analysis

For each scenario, we compare the proposed reachability-steering guidance policy with a greedy baseline that, at every sensing step, selects as its target the currently safest site within the reachable set. The probability of safe landing is evaluated for the safest reachable target in the final sensing opportunity (16), and Table 3 reports the mean and standard deviation of this probability over repeated stochastic realizations of the safety-belief process. The proposed method is tested for several values of the normalized hyperparameter

$$\rho = \frac{\lambda_k}{(\beta D_k)^2}, \tag{38}$$

which defines the one-step utility in (28) and effectively controls the trade-off between exploiting the currently best site J_{best} and preserving divert options J_{divert}.

When the isolated site A is only moderately more promising than the cluster (scenario A07), the greedy policy often commits early to a single site based on noisy, high-altitude information and therefore exhibits a noticeably lower final probability of safety. In contrast, the reachability-steering policy consistently improves the final outcome for $\rho \in [0.5, 10]$ by explicitly valuing future divert options and information gain. In higher-safety scenarios (A08–A09), where site A is already strongly favored a priori, both strategies achieve similar performance, although overly large values of ρ (e.g., $\rho = 100$) can make the guidance excessively conservative and slightly reduce the achieved probability of safety.

Figure 4 provides additional insight by plotting the x–z projections of all simulated trajectories for each scenario and each value of ρ. Under the greedy baseline (left column), trajectories tend to steer strongly toward the site currently

Table 3 Final probability of safety for the selected landing site under the greedy baseline and the proposed method with varying λ. Values denote mean ± standard deviation.

Scenario	Greedy	$\rho = 0.5$	5	10	100
A09	0.873 ± 0.005	0.873 ± 0.006	**0.875 ± 0.005**	0.874 ± 0.005	0.829 ± 0.009
A08	0.789 ± 0.013	0.774 ± 0.024	**0.793 ± 0.011**	0.780 ± 0.013	0.720 ± 0.009
A07	0.698 ± 0.016	0.698 ± 0.030	**0.723 ± 0.009**	0.721 ± 0.009	0.720 ± 0.009

estimated safest, leading to early commitment with limited opportunities for sensing other sites and divert maneuvers. As ρ increases from 0.5 to 5 and 10, the proposed policy keeps the lander closer to the center of the admissible corridor for more of the descent, delaying commitment until the safety beliefs become more informative, which explains the improved statistics in Table 3.

The interaction between target-region selection and sensing is illustrated in Figs. 5 and 6 for scenario A07. Each figure shows the nadir sensor footprint and the selected target region over four consecutive sensing steps for several representative Monte-Carlo runs. In Fig. 5, with $\rho = 5.0$, the reachability-steering guidance initially chooses target regions that simultaneously contain multiple candidate sites within the field of view. This design enables the lander to refine the safety estimates of several sites in parallel while remaining in states from which divert maneuvers to different sites are still feasible. As the beliefs converge and one of the clustered sites B1–B3 emerges as clearly preferable, the target region shrinks and shifts to enclose that site, and the FOV footprint correspondingly contracts around the selected landing region.

Figure 6 shows the same visualization for the greedy baseline. Here, the target region is typically placed directly over the site with the highest instantaneous safety estimate. As a consequence, the FOV footprint often excludes alternative candidates after only one or two sensing steps, and the admissible divert options narrow rapidly. When the early estimates are overly optimistic for the isolated site A or one particular clustered site, the greedy policy has little opportunity to collect additional evidence elsewhere, which can lead to a lower final probability of safety.

Finally, Fig. 7 aggregates the footprints and target regions over all scenarios and values of ρ. The reachability-steering guidance generates a family of nested footprints that gradually contract while repeatedly covering the clustered region, indicating that several safe divert paths are intentionally kept open until late in the descent. In contrast, the greedy strategy produces footprint trails that quickly commit to a single site. This qualitative difference in how the two policies trade off exploitation and exploration is consistent with the quantitative trends in Table 3: moderate values of ρ lead to trajectories that are more resilient to unfavorable realizations of the safety-belief process, while excessively large ρ values yield diminishing returns and can even be slightly detrimental due to overconservative motion.

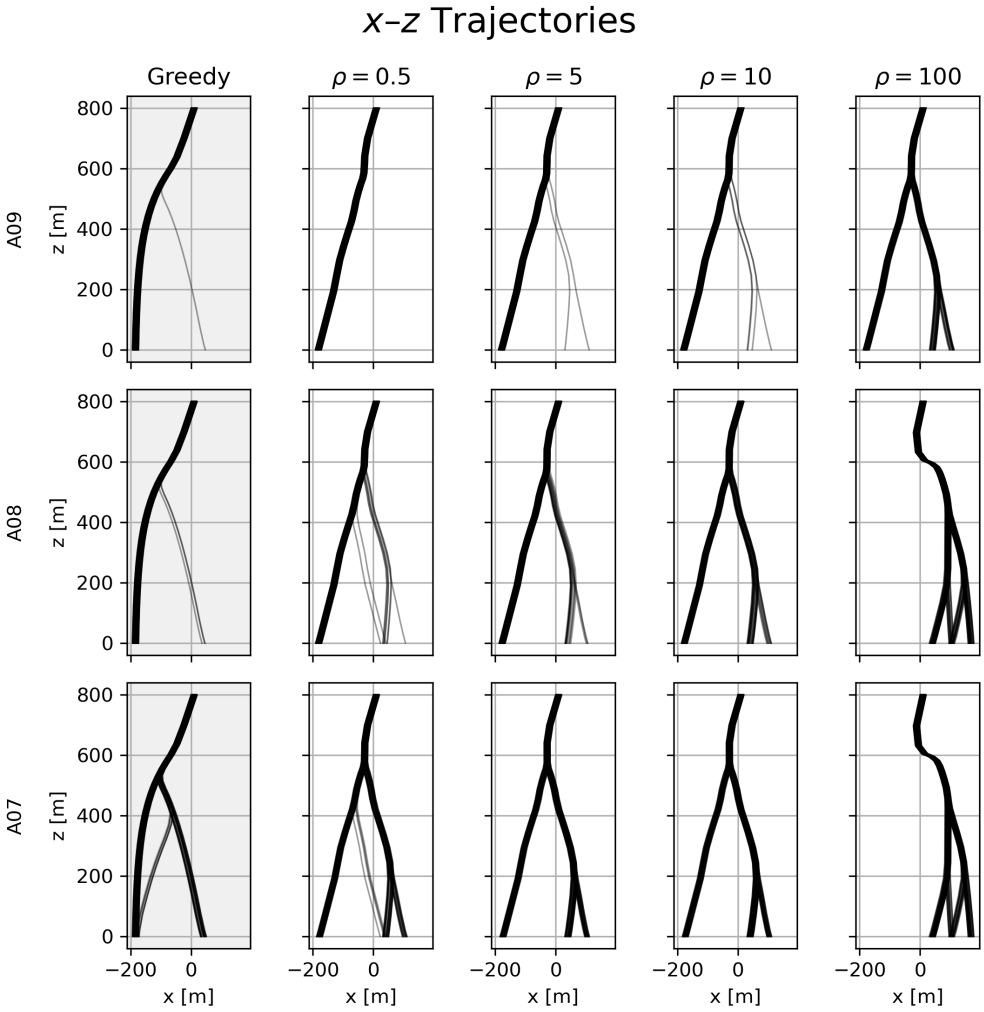

Fig. 4 All trajectories for each scenario with 100 Monte Carlo realizations of safety probability.

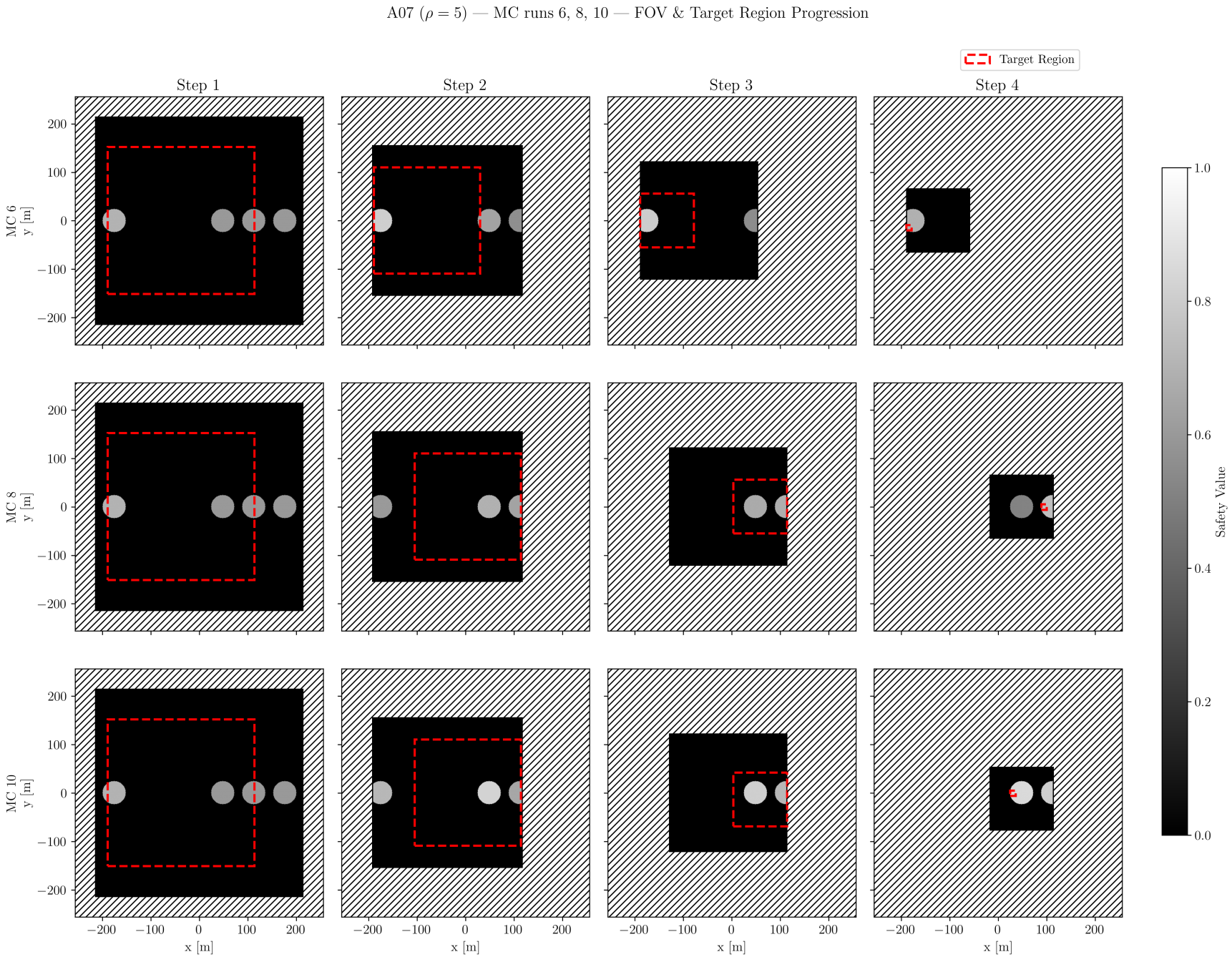

Fig. 5 FOV and selected target regions for selected Monte-Carlo runs for proposed approach in A07 scenario with ρ = 5.0.

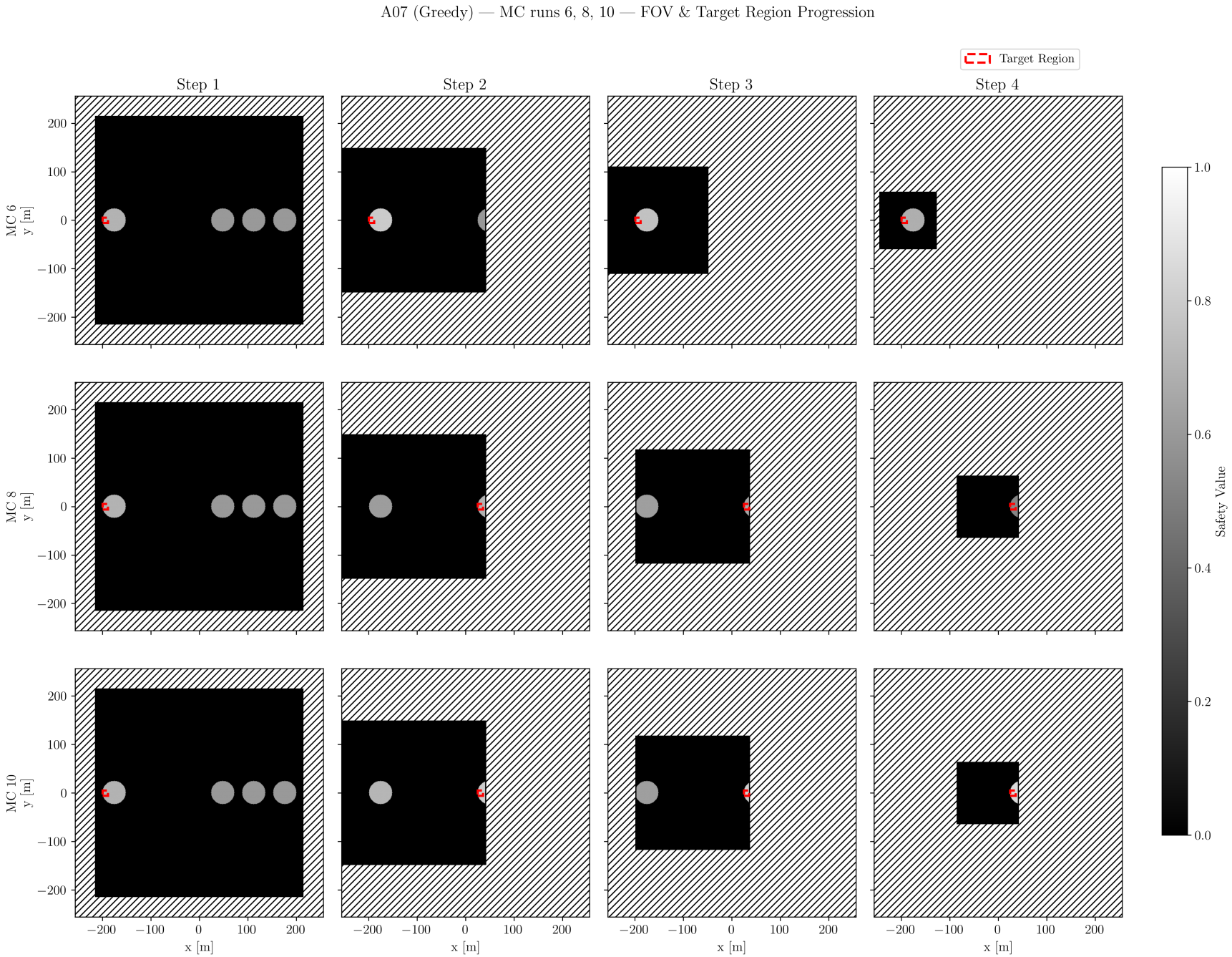

Fig. 6 FOV and selected target regions for selected Monte-Carlo runs for greedy approach.

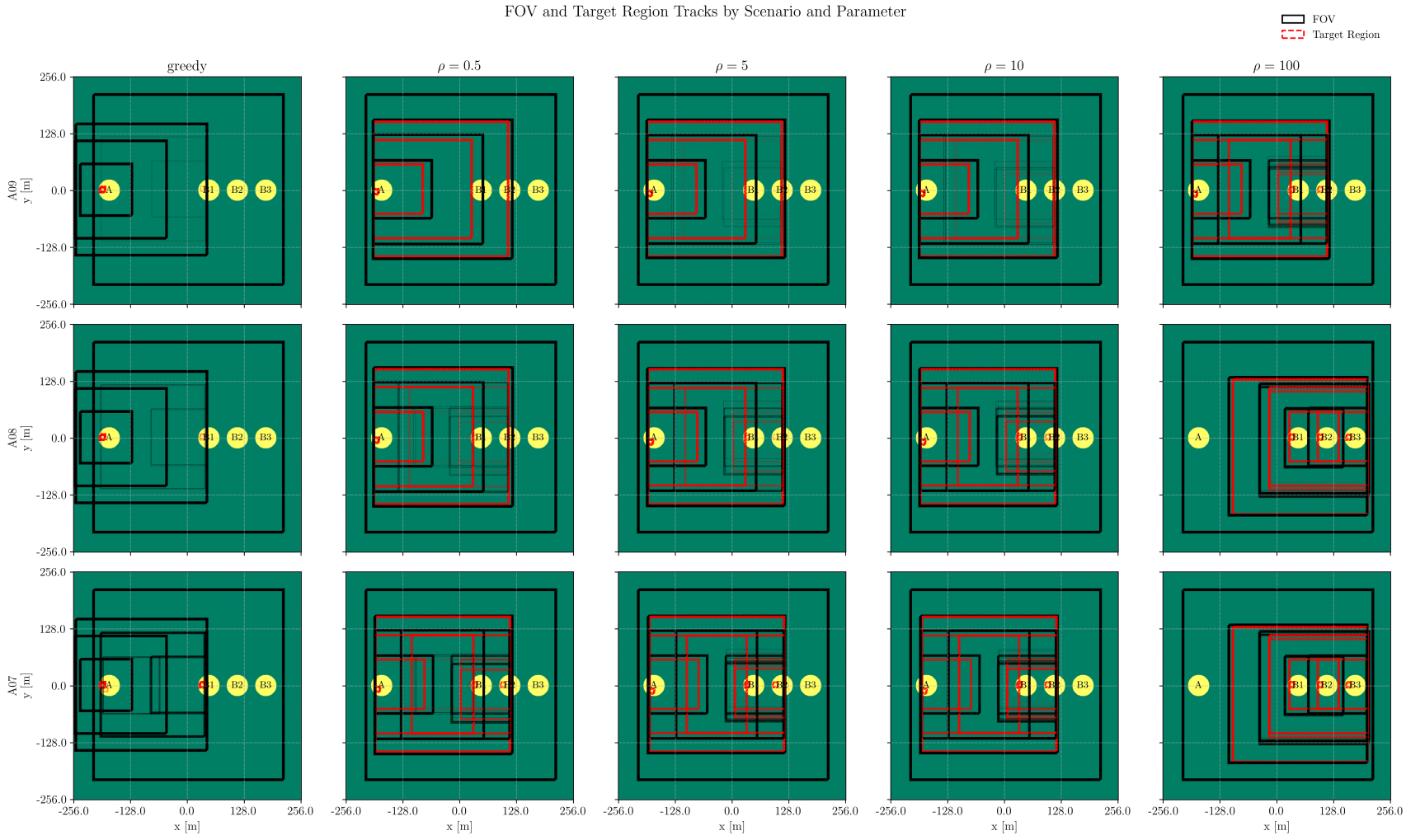

Fig. 7 All records of FOV and target regions for all cases.

VII. Conclusion

This paper introduced the powered descent decision-making (PDDM) problem, in which a lander must select a terminal landing site and descent trajectory so as to maximize the probability of a safe landing under uncertain terrain safety. We modeled PDDM as a partially observable Markov decision process, where the lander's control actions jointly influence reachability of candidate sites, field-of-view visibility, and the evolution of a stochastic safety-belief map. Direct dynamic programming over this belief- and set-valued state is computationally intractable, motivating a structured approximation.

To address this challenge, we proposed a reachability-steering hazard detection and avoidance (HDA) guidance algorithm. At each sensing step, the algorithm selects a convex target region on the surface using a one-step utility function that balances the currently safest candidate site against the divert options. The utility is evaluated over a visibility-safe reachable surface obtained by tightening the glide-slope constraint, which guarantees recursive feasibility by ensuring that all candidate landing sites remain within the sensor footprint throughout the descent. Given the selected target region, a convex optimal guidance problem based on lossless convexification is solved using precomputed controllable sets, enabling fast onboard trajectory optimization with guaranteed satisfaction of soft-landing constraints.

The offline computation of controllable sets is made tractable by representing them as constrained zonotopes, which are closed under the affine dynamics, Minkowski difference, and intersection operations needed for the backward recursion. This set-based structure links PDDM to recent advances in robust and resilient control, and provides a reusable library of controllable sets that can be indexed online by the desired sensing state and target-region parameters.

Numerical simulations in a simplified hazard environment with one isolated site and a cluster of nearby sites illustrate the behavior of the proposed guidance strategy. When early safety-belief information is ambiguous, the greedy baseline often commits prematurely to the isolated site based on noisy high-altitude data, leading to a lower final probability of safety. In contrast, the reachability-steering algorithm with moderate divert weighting maintains multiple safe divert options, repeatedly revisits the clustered region, and achieves a higher probability that the final landing site is safe. For cases in which one site is clearly superior from the outset, both methods perform similarly, while excessively large divert weights can introduce mild overconservatism. These results support the interpretation of the proposed algorithm as a principled interpolation between pure exploitation and explicit preservation of divert options.

Appendix

A. Greedy HDA

A simple approach is to keep updating the landing target as the safest landing site within the reachable surface. We call this approach *greedy HDA*, and show that it is optimal under the assumption that

$$\hat{s}_k = \hat{s}_{k+1} = \cdots = \hat{s}_N. \tag{39}$$

Proposition 1 (Open-Loop Optimality under Static Belief) *If the safety-belief map remains unchanged after step k, then for all $j \in \{k, \ldots, N\}$*

$$V^{\pi^*}(x_j, \hat{s}_j) = \max_{\gamma \in \mathcal{R}_j} \hat{s}_j(\gamma) \tag{40}$$

and the optimal policy is the greedy HDA: at each step j, select U_j so that the next state preserves reachability of a maximizer of $\hat{s}_j$ over $\mathcal{R}_j$. Equivalently, the optimal control profile can be chosen in open loop by first selecting

$$\gamma^* \in \arg\max_{\gamma \in \mathcal{R}_k} \hat{c}_k(\gamma), \tag{41}$$

and then executing any admissible control sequence that lands at γ^* while satisfying all constraints.

Proof. With static belief, $\hat{s}_{j+1} = \hat{s}_j$, the Bellman recursion reduces to $V^{\pi^*}(x_j, \hat{s}_j) = \max_{U_j} V^{\pi^*}(x_{j+1}, \hat{s}_j)$. At $j = N$, $V^{\pi^*}(x_N, \hat{s}_N) = \max_{\gamma \in \mathcal{R}_N} \hat{c}_N(\gamma)$. By induction, suppose the claim holds at $j+1$. Then

$$\begin{aligned} V^{\pi^*}(x_j, \hat{s}_j) &\overset{\hat{s}_{j+1}=\hat{s}_j}{=} \max_{U_j} \left(\max_{\gamma \in \mathcal{R}_{j+1}} \hat{s}_j(\gamma) \right) \\ &= \max_{\gamma \in \cup_{U_j} \mathcal{R}_{j+1}} \hat{s}_j(\gamma) = \max_{\gamma \in \mathcal{R}_j} \hat{s}_j(\gamma). \end{aligned} \tag{42}$$

At each step j, an optimal U_j is one that maintains reachability of a safest site. Because the maximizer set does not change across j ($\hat{s}_j = \hat{s}_k$ and $\mathcal{R}_j \subseteq \mathcal{R}_k$ for $j \in \{k, \ldots, N\}$), the initial choice γ^* at step k determines the entire optimal trajectory. Any admissible open-loop sequence that lands at γ^* is therefore optimal, which proves both claims. □

Remark VII.1 *When the safety map will not improve with future sensing, exploration has no value. The HDA problem collapses to a one-shot target selection followed by open-loop steering, and feedback only matters when future sensing updates the belief map.*

References

[1] Tomita, K., and Ho, K., *Reachability-Steering Approach for Autonomous Safe Planetary Landing*, 2025. https://doi.org/10.2514/6.2025-1898, URL https://arc.aiaa.org/doi/abs/10.2514/6.2025-1898.

[2] Malyuta, D., Reynolds, T. P., Szmuk, M., Lew, T., Bonalli, R., Pavone, M., and Açıkmeşe, B., "Convex Optimization for Trajectory Generation: A Tutorial on Generating Dynamically Feasible Trajectories Reliably and Efficiently," *IEEE Control Systems Magazine*, Vol. 42, No. 5, 2022, pp. 40–113. https://doi.org/10.1109/MCS.2022.3187542.

[3] Golombek, M. P., Cook, R. A., Moore, H. J., and Parker, T. J., "Selection of the Mars Pathfinder landing site," *Journal of Geophysical Research: Planets*, Vol. 102, No. E2, 1997, pp. 3967–3988. https://doi.org/https://doi.org/10.1029/96JE03318, URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/96JE03318.

[4] Spencer, D. A., Adams, D. S., Bonfiglio, E., Golombek, M., Arvidson, R., and Seelos, K., "Phoenix Landing Site Hazard Assessment and Selection," *Journal of Spacecraft and Rockets*, Vol. 46, No. 6, 2009, pp. 1196–1201. https://doi.org/10.2514/1.43932, URL https://doi.org/10.2514/1.43932.

[5] Masursky, H., and Crabill, N. L., "The Viking Landing Sites: Selection and Certification," *Science*, Vol. 193, No. 4255, 1976, pp. 809–812. https://doi.org/10.1126/science.193.4255.809, URL https://www.science.org/doi/abs/10.1126/science.193.4255.809.

[6] Lorenz, R. D., "Planetary landings with terrain sensing and hazard avoidance: A review," *Advances in Space Research*, Vol. 71, No. 1, 2023, pp. 1–15. https://doi.org/https://doi.org/10.1016/j.asr.2022.11.024, URL https://www.sciencedirect.com/science/article/pii/S0273117722010468.

[7] Tomita, K., and Ho, K., "Real-Time Stochastic Terrain Mapping and Processing for Autonomous Safe Landing," *Journal of Spacecraft and Rockets*, Vol. 62, No. 5, 2025, pp. 1848–1868. https://doi.org/10.2514/1.A36253, URL https://doi.org/10.2514/1.A36253.

[8] Jr., P. T., Gleichman, K. W., Carmer, D. C., Morita, Y., Trichel, M., and Gilbert, R. K., "Passive and active sensors for autonomous space applications," *Surveillance Technologies*, Vol. 1479, edited by S. Gowrinathan, R. J. M. Sr., and S. J. Schwartz, International Society for Optics and Photonics, SPIE, 1991, pp. 164 – 182. https://doi.org/10.1117/12.44529, URL https://doi.org/10.1117/12.44529.

[9] Johnson, A. E., Cheng, Y., Trawny, N., Montgomery, J. F., Schroeder, S., Chang, J., Clouse, D., Aaron, S., and Mohan, S., "Implementation of a Map Relative Localization System for Planetary Landing," *Journal of Guidance, Control, and Dynamics*, Vol. 46, No. 4, 2023, pp. 618–637. https://doi.org/10.2514/1.G006780, URL https://doi.org/10.2514/1.G006780.

[10] Li, S., Jiang, X., and Tao, T., "Guidance Summary and Assessment of the Chang'e-3 Powered Descent and Landing," *Journal of Spacecraft and Rockets*, Vol. 53, No. 2, 2016, pp. 258–277. https://doi.org/10.2514/1.A33208, URL https://doi.org/10.2514/1.A33208.

[11] Carson, J. M., Munk, M. M., and Wright, M. J., *NASA Development Strategy for Navigation Technologies to Precisely Land Payloads and Avoid Landing Hazards*, 2022. https://doi.org/10.2514/6.2022-0355, URL https://arc.aiaa.org/doi/abs/10.2514/6.2022-0355.

[12] Directorate, N. S. T. M., "LAND: Precision Landing and Hazard Avoidance," https://spacetechpriorities.org/wp-content/uploads/2024/04/LAND-Precision_Landing_and_Hazard_Avoidance_50m-2023-08-17.pdf, 2024. Accessed: 2024-05-07.

[13] Guo, Y., Hawkins, M., and Wie, B., "Applications of Generalized Zero-Effort-Miss/Zero-Effort-Velocity Feedback Guidance Algorithm," *Journal of Guidance, Control, and Dynamics*, Vol. 36, No. 3, 2013, pp. 810–820. https://doi.org/10.2514/1.58099, URL https://doi.org/10.2514/1.58099.

[14] Han, S., Jo, B.-U., and Ho, K., "Terminal Soft Landing Guidance Law Using Analytic Gravity Turn Trajectory," *Journal of Guidance, Control, and Dynamics*, Vol. 47, No. 9, 2024, pp. 1808–1821. https://doi.org/10.2514/1.G007903, URL https://doi.org/10.2514/1.G007903.

[15] Ito, T., Ueda, S., Yokota, K., Sakai, S.-i., Sawai, S., Sugita, M., Shibasaki, Y., Mukumoto, Y., Watabe, D., and Shimizu, S., "Terminal Powered Descent Guidance for the Smart Lander for Investigating Moon," *Journal of Guidance, Control, and Dynamics*, Vol. 48, No. 6, 2025, pp. 1298–1313. https://doi.org/10.2514/1.G008746, URL https://doi.org/10.2514/1.G008746.

[16] Lu, P., and Callan, R., "Propellant-Optimal Powered Descent Guidance Revisited," *Journal of Guidance, Control, and Dynamics*, Vol. 46, No. 2, 2023, pp. 215–230. https://doi.org/10.2514/1.G007214, URL https://doi.org/10.2514/1.G007214.

[17] Acikmese, B., and Ploen, S. R., "Convex Programming Approach to Powered Descent Guidance for Mars Landing," *Journal of Guidance, Control, and Dynamics*, Vol. 30, No. 5, 2007, pp. 1353–1366. https://doi.org/10.2514/1.27553, URL https://doi.org/10.2514/1.27553.

[18] Blackmore, L., Açikmeşe, B., and Scharf, D. P., "Minimum-Landing-Error Powered-Descent Guidance for Mars Landing Using Convex Optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 33, No. 4, 2010, pp. 1161–1171. https://doi.org/10.2514/1.47202, URL https://doi.org/10.2514/1.47202.

[19] Blackmore, L., "Autonomous Precision Landing of Space Rockets," *Frontiers of Engineering: Reports on Leading-Edge Engineering from the 2016 Symposium*, Vol. 46, 2016, pp. 15–20.

[20] Rogata, P., Di Sotto, E., Câmara, F., Caramagno, A., Rebordão, J., Correia, B., Duarte, P., and Mancuso, S., "Design and performance assessment of hazard avoidance techniques for vision-based landing," *Acta Astronautica*, Vol. 61, No. 1, 2007, pp. 63–77. https://doi.org/https://doi.org/10.1016/j.actaastro.2007.01.030, URL https://www.sciencedirect.com/science/article/pii/S0094576507000239, bringing Space Closer to People, Selected Proceedings of the 57th IAF Congress, Valencia, Spain, 2-6 October, 2006.

[21] Zhao, H., Dai, H., and Dang, Z., "Optimal Guidance for Lunar Soft Landing with Dynamic Low-Resolution Image Sequences," Vol. 69, No. 11, 2022, pp. 4013–4025. https://doi.org/10.1016/j.asr.2022.03.006, URL https://linkinghub.elsevier.com/retrieve/pii/S0273117722001922.

[22] Elango, P., Sarsilmaz, S. B., and Acikmese, B., "Deferred-Decision Trajectory Optimization," , 2025. URL https://arxiv.org/abs/2502.06623.

[23] Elango, P., Sarsilmaz, S. B., and Acikmese, B., *Deferring Decision in Multi-target Trajectory Optimization*, AIAA SCITECH 2022 Forum, 2022. https://doi.org/10.2514/6.2022-1583, URL https://arc.aiaa.org/doi/abs/10.2514/6.2022-1583.

[24] Hayner, C. R., Buckner, S. C., Broyles, D., Madewell, E., Leung, K., and Açikmeşe, B., "HALO: Hazard-Aware Landing Optimization for Autonomous Systems," *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 3261–3267. https://doi.org/10.1109/ICRA48891.2023.10160655.

[25] Kamath, A. G., Vinod, A. P., Elango, P., Cairano, S. D., and Weiss, A., "Set-based Optimal, Robust, and Resilient Control with Applications to Autonomous Precision Landing," , 2025. URL https://arxiv.org/abs/2512.07043.

[26] Srinivas, N., Vinod, A. P., Cairano, S. D., and Weiss, A., *Lunar Landing with Feasible Divert using Controllable Sets*, AIAA SCITECH 2024 Forum, 2024. https://doi.org/10.2514/6.2024-0324, URL https://arc.aiaa.org/doi/abs/10.2514/6.2024-0324.

[27] Iiyama, K., Tomita, K., Jagatia, B. A., Nakagawa, T., and Ho, K., "Deep Reinforcement Learning for Safe Landing Site Selection with Concurrent Consideration of Divert Maneuvers," , 2021. URL https://arxiv.org/abs/2102.12432.

[28] Tomita, K., Shimane, Y., and Ho, K., *Optimal Predictive Guidance for Autonomous Hazard Detection and Avoidance*, 2024. https://doi.org/10.2514/6.2024-1585, URL https://arc.aiaa.org/doi/abs/10.2514/6.2024-1585.

[29] Açıkmeşe, B., Carson, J. M., and Blackmore, L., "Lossless Convexification of Nonconvex Control Bound and Pointing Constraints of the Soft Landing Optimal Control Problem," *IEEE Transactions on Control Systems Technology*, Vol. 21, No. 6, 2013, pp. 2104–2113. https://doi.org/10.1109/TCST.2012.2237346.

[30] Vinod, A. P., Kamath, A., Weiss, A., and Di Cairano, S., "Set-based lossless convexification for a class of robust nonlinear optimal control problems," *IEEE Conference on Decision and Control (CDC)*, 2025. URL https://www.merl.com/publications/TR2025-160.

[31] Borrelli, F., Bemporad, A., and Morari, M., *Predictive control for linear and hybrid systems*, Cambridge University Press, 2017.

[32] Scott, J. K., Raimondo, D. M., Marseglia, G. R., and Braatz, R. D., "Constrained zonotopes: A new tool for set-based estimation and fault detection," *Automatica*, Vol. 69, 2016, pp. 126–136. https://doi.org/https://doi.org/10.1016/j.automatica.2016.02.036, URL https://www.sciencedirect.com/science/article/pii/S0005109816300772.

[33] Vinod, A. P., "pycvxset: A Python package for convex set manipulation," , 2024. URL https://arxiv.org/abs/2410.11430.

[34] Baca, M. J., and Ferri, T. J., "Some results on operator commutator series and spectral distribution functions," *International Journal of Mathematics and Mathematical Sciences*, Vol. 21, No. 8, 1998, pp. 585–601. https://doi.org/10.1155/S1024123X98000866.

[35] Elango, P., Kamath, A. G., Yu, Y., Acikmese, B., Mesbahi, M., and Carson, J. M., *A Customized First-Order Solver for Real-Time Powered-Descent Guidance*, 2022. https://doi.org/10.2514/6.2022-0951, URL https://arc.aiaa.org/doi/abs/10.2514/6.2022-0951.

[36] Wang, Y., and Boyd, S., "Fast Model Predictive Control Using Online Optimization," *IEEE Transactions on Control Systems Technology*, Vol. 18, No. 2, 2010, pp. 267–278. https://doi.org/10.1109/TCST.2009.2017934.

[37] Carson, J. M., Munk, M. M., Sostaric, R. R., Estes, J. N., Amzajerdian, F., Blair, J. B., Rutishauser, D. K., Restrepo, C. I., Dwyer-Cianciolo, A. M., Chen, G., and Tse, T., *The SPLICE Project: Continuing NASA Development of GN&C Technologies for Safe and Precise Landing*, 2019. https://doi.org/10.2514/6.2019-0660, URL https://arc.aiaa.org/doi/abs/10.2514/6.2019-0660.