Active Continuous-Time Simultaneous Localization & Mapping for Powered Descent Guidance Maneuvers

Samuel C. Buckner* and Behçet Açıkmeşe†
University of Washington, Seattle, WA 98195, USA

John M. Carson III‡ and Breanna J. Johnson§
NASA Johnson Space Center, Houston, TX 77058, USA

The problem of guidance-navigation co-design for autonomous aerospace systems concerns the simultaneous satisfaction of guidance and navigation requirements in mission & trajectory design, and is of keen interest for the realization of robust autonomy in the aerospace industry. In this paper, a general-purpose technique–Active Continuous-Time Simultaneous Localization & Mapping (ACT-SLAM)–is proposed to systematically and flexibly solve for various elements of guidance-navigation co-design and is specialized for scenarios involving spatial perception objectives. The resulting architecture makes use of advances in successive convexification for nonconvex trajectory optimization to solve the resulting problem at near-real-time speeds. ACT-SLAM is consequently demonstrated on the highly-constrained powered descent guidance (PDG) problem in a lunar environment with LiDAR-based measurements, showing promising results for joint reduction in vehicle and mapping uncertainty. Furthermore, rigorous satisfaction of both guidance and navigation requirements is shown through 3σ Monte Carlo analysis, and benchmark comparisons to several recent methods are made, demonstrating considerable improvements in terms of information-theoretic objectives.

I. Nomenclature

$\mathcal{I}$	Inertially-fixed frame of reference (inertial frame)
$\mathcal{B}$	Body-fixed frame of reference (body frame)
$\mathcal{F}$	Arbitrary frame of reference
$\mathcal{R}_{\mathcal{F}_1\to\mathcal{F}_2}$	Rotation matrix from frame $\mathcal{F}_1$ to frame $\mathcal{F}_2$
$\{e_1,\ldots,e_n\}$	Standard basis in n-dimensional space
SO(3)	Special orthogonal group in three-dimensional space
$\mathcal{S}(3)$	Unit 3-sphere manifold in $\mathbb{R}^4$.
$\partial\mathcal{S}$	Defines the boundary of a set $\mathcal{S}$
$[n]$	Shorthand notation, equivalent to $1,\ldots,n$
$\|\cdot\|$	Norm operation, assumed to be Euclidean (two) norm by default
$\text{vec}(\cdot)$	Vectorization operation, such that $b=\text{vec}(A)$ maps a matrix $A\in\mathbb{R}^{n\times m}$ to a vector $b\in\mathbb{R}^{nm}$
$(x_1,\ldots,x_n)$	Vector concatenation operation, equivalent to $[x_1^\top,\ldots,x_n^\top]^\top\in\mathbb{R}^{n_{x_1}\times\cdots\times n_{x_n}}$
6-DOF	Six Degree-of-Freedom point-mass dynamics (translation and rotation)
CT-OCP	Continuous-Time Optimal Control Problem
EKBF	Extended Kalman-Bucy Filter
SLAM	Simultaneous Localization And Mapping

*NSF Fellow, William E. Boeing Dept. of Aeronautics & Astronautics, AIAA Student Member; sbuckne1@uw.edu.
†Professor, William E. Boeing Dept. of Aeronautics & Astronautics, AIAA Fellow; behcet@uw.edu.
‡Technical Integration Manager – Precision Landing, NASA STMD, AIAA Associate Fellow; john.m.carson@nasa.gov.
§Aerospace Engineer, Flight Mechanics and Trajectory Design Branch, AIAA Atmospheric Flight Mechanics Technical Committee Member.

II. Introduction

WITHIN the aerospace industry exists an ever-increasing interest in full-scale autonomy for aerial vehicles and spacecraft. This autonomy is often expressed in terms of three principle components: guidance, navigation & control (GN&C). These disciplines typically experience overlap in terms of requirements for constructing an autonomous system, however their integrated design at a high level of fidelity has been left relatively unexplored in the literature. The problem of **co-design** concerns efforts to jointly optimize objectives, or satisfy requirements, for several of these elements *simultaneously*. In this paper, our interest will be focused towards the problem of co-designing for guidance and navigation objectives. Specifically, we seek to design (plan) a trajectory that obeys physical and operational constraints (guidance) such that uncertainty metrics associated with vehicle estimation are optimized for (navigation).

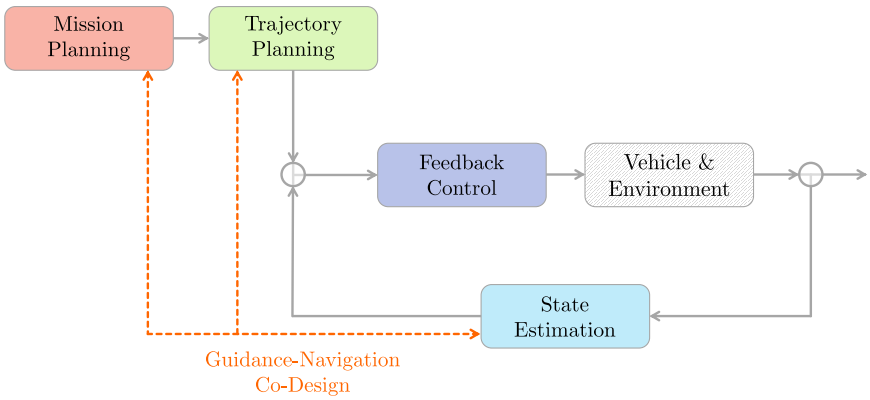


Fig. 1 High-level diagram of the GN&C pipeline and how guidance-navigation co-design relates mission & trajectory planning (*guidance*) to state estimation (*navigation*).

A. Related Literature

The problem of guidance-navigation co-design is applicable to many aerospace mission contexts; we specifically concern ourselves with spacecraft powered descent guidance (PDG) maneuvers, the final downstream phase of planetary entry, descent & landing (EDL) [1], as the focus of this work. The PDG problem is concerned with the algorithmic design of pinpoint landing methodologies which can safely achieve soft landing on a planetary body while obeying physical and operational constraints, known in the optimization parlance as *path constraints*. This problem has been explored in the literature through implementation of optimal control techniques [2], including lossless convexification [3–8] and successive convexification [9, 10]. Existing approaches have been developed to solve the guidance-navigation co-design problem for EDL maneuvers, such as Robust Trajectory Optimization (RTO) [11]. This technique makes use of a genetic algorithm as the (sampling-based) optimizer, and optimizes over covariance terms from a forward propagation (simulation) of linear covariance analysis [12, 13]. Although this method provides a capable co-design approach for upstream phases of EDL maneuvers, the sampling-based nature of the algorithm limits it to optimization over only a few variables (in the case of Calkins et. al. [11], two variables). For the downstream PDG maneuvering phase, substantially more resolution in terms of optimization variables is required to find a solution which can satisfy all path constraints when using numerical optimal control techniques such as those aforementioned*. This would be prohibitively expensive or even intractable to optimize with sampling-based methods, motivating a new approach for co-design under highly-constrained mission scenarios. Ticozzi et. al. [14] present a factor graph-based technique for guidance-navigation co-design in the context of perceptive spacecraft maneuvers, however are still restricted to the same limitations of zeroth-order (sampling-based) optimization.

Other techniques have been developed that attempt to embed guidance-navigation co-design directly within gradient-based trajectory optimization templates. This includes a map covariance minimization technique for PDG scenarios [15] and discrete-time active mapping without path constraints applied to robotic locomotion

*For example, a trajectory optimization with 10 discrete knot points and 6 control elements (thrust and torque terms), parameterized with a first-order hold, corresponds to $10 \times 6 = 60$ variables that must be optimized for at a minimum.

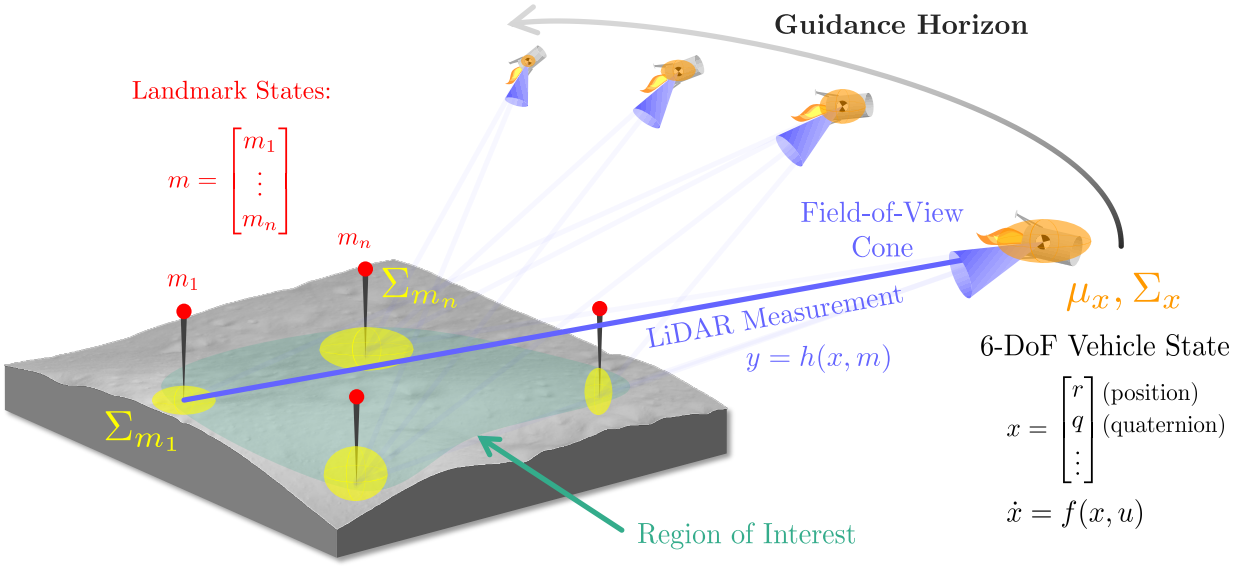


Fig. 2 Illustration demonstrating the modeling approach behind the ACT-SLAM technique applied to the active sensing guidance problem.

in closed environments [16, 17]. In this work, we attempt to extend these existing developments to a point of further generality that makes the approach amenable to the canonical continuous-time optimal control problem (CT-OCP) template. We focus our efforts specifically on application to the PDG problem with a perceptive range-based sensor (e.g. LiDAR) to form the measurement model for estimation, where the system relies on these measurements to produce an accurate navigational solution in terms of its own state and the terrain environment. Guidance-navigation co-design for perceptive maneuvers is notably difficult due to the tight positional and attitude restrictions necessary to perform pointing maneuvers while landing, making them a suitable stress test for the proposed approach. We refer to this problem as the *active sensing guidance* problem going forwards.

B. Core Contribution

We introduce a novel, computationally-tractable technique towards solving the general-form active sensing guidance problem. This technique provides a synthesis of optimal estimation and optimal control properties while being embedded directly into a CT-OCP problem template, making it an appropriate candidate for modern PDG maneuver design relative to pre-existing approaches in the literature. This modeling approach considers a vehicle state $x(t) \in \mathbb{R}^{n_x}$ and corresponding control input $u(t) \in \mathbb{R}^{n_u}$, where the vehicle state must necessarily contain *at least* a six degree-of-freedom (6-DoF) point mass representation [18] to model LiDAR measurements, typically parameterized as an inertial position $r(t) \in \mathbb{R}^3$ and inertial-to-body quaternion $q \in \mathcal{S}^3$. Additionally, a *region of interest* is identified with respect to the ground terrain, and a set of positional landmarks are characterized within this region as $m_j(t) \in \mathbb{R}^3$, $j = 1, \ldots, n_\ell$, with the full map state represented as $m(t) \triangleq (m_1(t), \ldots, m_{n_\ell}(t))$. The vehicle and map states are consequently modeled as multivariate Gaussian random variables such that $x(t) \sim \mathcal{N}(\mu_x(t), \Sigma_x(t))$ and $m(t) \sim \mathcal{N}(\mu_m(t), \Sigma_m(t))$, respectively, with cross-correlation terms $\Sigma_{xm}(t)$. LiDAR measurements are used to directly correlate and reduce both vehicle uncertainty $\Sigma_x(t)$ and mapping uncertainty $\Sigma_m(t)$. A free-final-time trajectory optimization problem is modeled under decision variables $d(t) \triangleq (\mu_x(t), \mu_m(t), \Sigma_x(t), \Sigma_m(t), \Sigma_{xm}(t), u(t))$ over time span $t \in [0, t_f]$, where $\mu_x(t), u(t)$ must satisfy traditional PDG path constraints as part of the guidance requirements[†], and $\Sigma_x(t), \Sigma_m(t)$ must satisfy 3σ error budget constraints as part of the navigation requirements. The optimization output $\mu_x(t), u(t)$ can then serve as the nominal trajectory, which can be computed either offline (in a trajectory design/shaping context), or online (recursively revising the guidance

[†]Examples of path constraints typical to PDG problems include bounds on thrust and torque control terms and glideslope constraints.

solution based on new priors or a shifting region of interest). This explicit characterization of covariance propagation in trajectory optimization relates our technique to the well known class of methods known as *covariance steering* [19–21], with the additional characterization of measurement covariance contributions (as opposed to purely process covariance) as will be explored in the following sections.

The proposed approach is labeled **Active Continuous-Time Simultaneous Localization and Mapping** (ACT-SLAM), which consists of the following key components that form its core contribution to the literature:

1) **Actively** optimizes over a Gaussian trajectory distribution over the forward time horizon through explicit mean & covariance representation in a continuous-time optimal control problem (CT-OCP). Specifically, the continuous-time successive convexification (CT-SCVX) [22] technique is chosen for real-time nonconvex trajectory optimization.
2) Uses a **continuous-time** optimal estimation model–the *Kalman-Bucy filter* [23]–to embed the aforementioned Gaussian trajectory distribution as a set of differential equations within a CT-OCP template.
3) Models the joint relationships between uncertainty in the vehicle's position and uncertainty in the mapped terrain environment through the use of **simultaneous localization and mapping (SLAM)** [24] modeling approaches.

A graphical illustration of the ACT-SLAM approach is provided in figure 2. The remainder of this paper is structured as follows: section III presents the candidate Gaussian filtering model, section IV uses this filtering model to propose the ACT-SLAM problem formulation, section V presents the CT-SCVX algorithm to solve the ACT-SLAM problem, section VI provides high-fidelity simulation, model validation and literature comparison results, and section VII presents concluding remarks.

III. The Filtering Model

Before we can progress towards realization of the full ACT-SLAM trajectory optimization problem, we must first select an appropriate measurement filtering strategy which will enable us to embed a Gaussian trajectory distribution in the optimization dynamics. This effort corresponds to the state estimation (*navigation*) component of the co-design process, and is expanded on within this section.

A. Continuous-time Gaussian Filtering

We start by defining a continuous-time stochastic state-space system, parameterized with a system state vector $z(t) \in \mathbb{R}^{n_z}$ and observation vector $y(t) \in \mathbb{R}^{n_y}$:

$$\dot{z}(t) = f(z(t), u(t)) + w(t) \tag{1}$$
$$y(t) = h(z(t)) + v(t) \tag{2}$$

Where $f : \mathbb{R}^{n_z} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_z}$ corresponds to the process dynamics, $h : \mathbb{R}^{n_z} \to \mathbb{R}^{n_y}$ corresponds to the observation model, and $w(t) \in \mathbb{R}^{n_z}, v(t) \in \mathbb{R}^{n_y}$ correspond to zero-mean temporally-uncorrelated Gaussian processes for the process and observation models, with covariances $W(t) \in \mathbb{R}^{n_z \times n_z}$ and $V(t) \in \mathbb{R}^{n_y \times n_y}$ defined as follows for a fixed $t \in \mathbb{R}$ and for any $\tau \in \mathbb{R}$:

$$\mathbb{E}[w(t)w(\tau)^\top] = W(t)\delta(t-\tau) \tag{3}$$
$$\mathbb{E}[v(t)v(\tau)^\top] = V(t)\delta(t-\tau) \tag{4}$$

Where δ corresponds to the Dirac delta function. We note that, under this definition, the system state will also define a Gaussian process such that, for each fixed time $t \in \mathbb{R}$, its marginal distribution is characterized with a mean $\mu(t) \in \mathbb{R}^z$ and covariance $\Sigma(t) \in \mathbb{R}^{n_z \times n_z}$ such that:

$$\mu(t) = \mathbb{E}[z(t)] \tag{5}$$
$$\Sigma(t) = \mathbb{E}[(z(t) - \mu(t))(z(t) - \mu(t))^\top] \tag{6}$$

We remark that this notation to describe stochastic systems avoids the formalized use of stochastic differential equations, however is a common and accepted shorthand [12, 25, 26], and all following results apply without

loss of generality to the formalized expression. Traditional spaceflight-grade navigation filters make use of the hybrid extended Kalman filter (hybrid-EKF), which splits the mean and covariance propagation into prediction and update steps over a time interval $t \in [t_{k-1}, t_k]$. Note that we use the shorthand $\phi_k = \phi(t_k)$ for a variable $\phi(t)$ to denote evaluation of a continuous-time variable at discrete time t_k. The hybrid-EKF is summarized with the following system of equations:

$$
\begin{aligned}
(\textbf{predict})\ \dot{\mu}(t) &= f(\mu(t), u(t)), && (7)\\
\dot{\Sigma}(t) &= F(t)\Sigma(t) + \Sigma(t)F(t)^\top + W(t), && (8)\\
(\textbf{update})\ \mu_k &= \mu_k^- + K_k(y_k - H_k\mu_k^-), && (9)\\
\Sigma_k &= (I - K_k H_k)\Sigma_k^-, && (10)\\
K_k &= \Sigma_k^- H_k^\top (H_k \Sigma_k^- H_k^\top + V_k)^{-1} && (11)
\end{aligned}
$$

Where $F(t), H(t)$ define measurement partials about a given reference $(\bar{z}(t), \bar{u}(t))$:

$$
F(t) = \left.\frac{\partial f(z(t), u(t))}{\partial z(t)}\right|_{(\bar{z}(t), \bar{u}(t))} \tag{12}
$$

$$
H(t) = \left.\frac{\partial h(z(t))}{\partial z(t)}\right|_{\bar{z}(t)} \tag{13}
$$

This technique involves integrating the prediction-side system of differential equations, then performing discrete updates at a frequency that is notionally aligned with a measurement acquisition rate. Specifically, equations (7)-(8) are integrated across $t \in [t_{k-1}, t_k)$ from μ_{k-1}, Σ_{k-1} to produce μ_k^-, Σ_k^-, and a discrete measurement update at time t_k is applied to μ_k^-, Σ_k^- to produce μ_k, Σ_k. This approach has been explored for both online, embedded implementations [27] and offline linear covariance analysis [28, 29]. Consider a concatenated filtering state $\zeta(t) \triangleq (\mu(t), \text{vec}(\Sigma(t))) \in \mathbb{R}^{n_\zeta}$, where $n_\zeta = n_z + n_z^2$ and define mapping $\tilde{g} : \mathbb{R}^{n_\zeta \times n_y \times n_u} \to \mathbb{R}^{n_\zeta}$ such that $\dot{\zeta} = \tilde{g}(\zeta, y, u)$ satisfies equations (7)-(11). Due to the *discrete* update equations (9)-(11), this mapping is defined as a *hybrid dynamical system* [30]. While this system is considered sound within the context of state estimation, hybrid dynamical systems present substantial numerical complexity when embedded within a CT-OCP problem formulation [31], motivating an alternative formulation which removes the discrete-time update step.

To address this concern, we can instead consider an alternative formulation for the Kalman filter which is expressed entirely as a set of continuous-time differential equations, known as the *extended Kalman-Bucy filter* (EKBF) [23]. Moreover, this filter contains the following system of differential equations to couple the aforementioned predict and update steps:

$$
\begin{aligned}
\dot{\mu}(t) &= f(\mu(t), u(t)) + K(t)(y(t) - h(\mu(t))) && (14)\\
\dot{\Sigma}(t) &= F(t)\Sigma(t) + \Sigma(t)F(t)^\top + W(t) - K(t)H(t)\Sigma(t) && (15)\\
K(t) &= \Sigma(t)H(t)^\top V(t)^{-1} && (16)
\end{aligned}
$$

Considering the same filtering state $\zeta(t)$, we can define mapping $g : \mathbb{R}^{n_\zeta \times n_y \times n_u} \to \mathbb{R}^{n_\zeta}$ such that $\dot{\zeta} = g(\zeta, y, u)$ satisfies equations (14)-(16), and note that this new dynamical equation is fully continuous, thus is readily amenable to CT-OCP formulations.

B. Simultaneous Localization and Mapping (SLAM)

In this work, we aim to solve the active sensing guidance problem, which pertains to modeling the joint correlations between the vehicle state and landmark states to concurrently reduce both. The simultaneous localization and mapping (SLAM) problem [24] provides an assortment of well-established approaches towards directly parameterizing these vehicle-map relationships, and has been explored for powered descent guidance problems in a strictly estimation sense [27, 31]. One such approach that is particularly well-suited towards trajectory optimization is the EKF-SLAM approach, wherein the vehicle and landmark states are directly concatenated to form the system state, and filtered through an EKF. In this work, we extend EKF-SLAM

towards a continuous-time variant, EKBF-SLAM, and assert certain formulation nuances to recover a model consistent with perceptual estimation yet tractable with numerical trajectory optimization methods.

We start by parameterizing a vehicle state $x(t) \in \mathbb{R}^{n_x}$ with a corresponding control $u(t) \in \mathbb{R}^{n_u}$, and additionally parameterize the environment with a discrete set of position landmarks $m_j(t) \in \mathbb{R}^3$, $j = 1, \ldots, n_\ell$ such that the full map state is represented as $m(t) = (m_j(t))_{j=1}^{n_\ell} \in \mathbb{R}^{3n_\ell}$. Then, the system state defined in section III.A is set as the concatenation of the vehicle and map states, such that $z(t) = (x(t), m(t))$, which can be directly fed into the system of EKBF equations defined in (14)-(16). Notably, this design leads to the following partitions on the stochastic state-space model terms, where subscripts x, m index out the vehicle and mapping components of each term, respectively:

$$\mu = \begin{bmatrix} \mu_x \\ \mu_m \end{bmatrix}, \quad \Sigma = \begin{bmatrix} \Sigma_x & \Sigma_{xm} \\ \Sigma_{xm}^\top & \Sigma_m \end{bmatrix}, \quad f = \begin{bmatrix} f_x \\ f_m \end{bmatrix}, \quad W = \begin{bmatrix} W_x & 0 \\ 0 & W_m \end{bmatrix}$$

We show all partitions for generality, however note that for a static terrain-based environment, the map dynamics will be taken to be $f_m(\cdot) = 0$. At this point, we must assert some system assumptions and modifications to the EKBF model to relate our formulation to the SLAM problem. Notably, we must relate the vehicle and map state uncertainties through the concept of perceptive measurements in a restricted field of view. To do this, we must assert several assumptions about our vehicle model and configuration. We start by defining an inertial (world) frame $\mathcal{I}$ and six degree-of-freedom (6-DOF) point mass body frame representation $\mathcal{B}$ affixed to the vehicle. Then, the following assumptions are made:

Assumption 1. *The vehicle state $x(t)$ contains a 6-DOF pose representation parameterized by an inertial position $r_\mathcal{I}$ and inertial-to-body transformation matrix $\mathcal{R}_{\mathcal{I}\to\mathcal{B}}(t)$. We could also consider a reduced-order attitude representation, such as a quaternion, however we must be able to recover a transformation matrix via some closed-form relation. In the remainder of this subsection, we will drop the subscripts for these terms for notational convenience.*

Assumption 2. *The vehicle contains a restricted field-of-view (FOV) perceptive sensor that acquires range-based measurements (such as a LiDAR sensor), with a body-fixed boresight pointing vector b and FOV half-angle $\beta_{\max}$.*

We must further modify the definition of our measurement noise term to be state-dependent, such that $v(t) = v(z(t)) \implies V(t) = V(z(t))$. Let us first define a range-based measurement model to each landmark as the following:

$$h_j(z(t)) = \|m_j(t) - r(t)\|_2 \tag{17}$$

We consequently have $n_y = n_\ell$ measurements corresponding to each landmark, with the full measurement function defined as $h(z(t)) = (h_j(z(t)))_{j=1}^{n_\ell}$. We can also define a range-based standard deviation of the measurement noise to each landmark as the following:

$$\tilde{\sigma}_j(z(t)) = \gamma_1 \exp(\gamma_2 h_j(z(t))) \tag{18}$$

Where γ_1, γ_2 represent hardware constants. This type of exponential noise model for range-based measurements is common practice in the modeling of LiDAR sensors, and has been validated for accuracy in the context of both spacecraft [32] and autonomous driving [33]. Because our range-based measurements require line-of-sight to the landmark, we must further define the line-of-sight set $\mathcal{F}_j$ such that $z(t) \in \mathcal{F}_j$ implies that landmark m_j is in line-of-sight of the sensor. More rigorously, we define angle $\beta(t)$ as the angle between relative distance $\Delta r_j(t) = m_j(t) - r(t)$ and $\mathcal{R}(t)^\top b$ (where application of $\mathcal{R}(t)$ is necessary to express the boresight vector in the inertial frame). Then, the objective is to find $\mathcal{F}_j$ as the set such that $\beta(t) \leq \beta_{\max}$. This leads to the following definition:

$$\mathcal{F}_j = \left\{ r, \mathcal{R}, m^j \in z \;\middle|\; \frac{(\mathcal{R}^\top b)^\top (m^j - r)}{\|m^j - r\|_2} \geq \cos\beta_{\max} \right\} \tag{19}$$

Because of the line-of-sight requirement, we must revise our noise model to be *state-dependent*, such that the noise standard deviation is (effectively) infinite when the landmark is outside the FOV of the vehicle, since no measurement can be made under this condition. This can be done with the following model building on

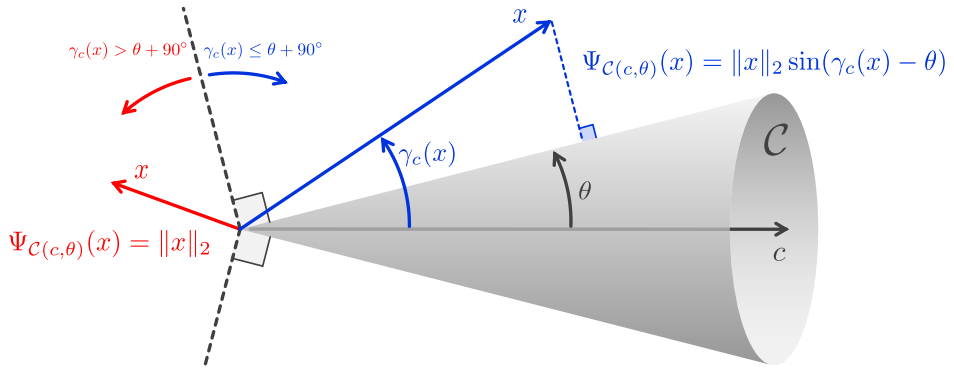


Fig. 3 A visual interpretation of the set $\Psi_{\mathcal{C}(c,\theta)}$.

the noise definition in equation (18):

$$\sigma_j(z(t)) = \begin{cases} \tilde{\sigma}_j(z(t)), & z(t) \in \mathcal{F}_j \\ \infty, & \text{else} \end{cases} \tag{20}$$

Moreover, we are interested in the reciprocal of the noise:

$$\sigma_j^{-1}(z(t)) = \begin{cases} \tilde{\sigma}_j^{-1}(z(t)), & z(t) \in \mathcal{F}_j \\ 0, & \text{else} \end{cases} \tag{21}$$

This definition notably contains a *discontinuity* when crossing the boundary of set $\mathcal{F}_j$. As will be seen in the next section, gradient-based trajectory optimization techniques (typically) have the requirement that the dynamics model be expressed as $\mathcal{C}^1-$differentiable. To address the discontinuity in equation (21), we must first establish several important definitions.

Definition 1 ***(Signed distance function).*** *The signed distance function (SDF) of a set $\mathcal{S}$ is defined as:*

$$\Psi_{\mathcal{S}}(x) \triangleq \begin{cases} -\min_{x^\star \in \partial\mathcal{S}} \|x - x^\star\|_2, & x \in \mathcal{S} \\ \min_{x^\star \in \partial\mathcal{S}} \|x - x^\star\|_2, & x \notin \mathcal{S} \end{cases} \tag{22}$$

Definition 2 ***(Dot-product second order cone).*** *The class of dot-product second order cones $\mathcal{C}$ can be parameterized with a pointing vector $c \in \mathcal{R}^n$ and half-angle $\theta \in \mathcal{R}$ using the trigonometric dot product relation such that:*

$$\mathcal{C}(c,\theta) \triangleq \left\{ x \in \mathcal{R}^n \mid \frac{c^\top x}{\|c\|_2 \|x\|_2} \geq \cos\theta \right\} \tag{23}$$

By applying definitions 1 and 2, we can derive the SDF for a second-order cone $\mathcal{C}$ as the following:

$$\Psi_{\mathcal{C}(c,\theta)}(x) = \begin{cases} \|x\|_2 \sin(\gamma_c(x) - \theta), & \gamma_c(x) \leq \theta + 90° \\ \|x\|_2, & \gamma_c(x) > \theta + 90° \end{cases} \tag{24}$$

Where γ_c represents the angle between x and c:

$$\gamma_c(x) = \cos^{-1}\left(\frac{x^\top c}{\|x\|_2 \|c\|_2} \right) \tag{25}$$

Using this definition, the SDF for a set $\mathcal{F}_j$ can be defined as the following:

$$\Psi_{\mathcal{F}_j}(z(t)) = \Psi_{\mathcal{C}(b,\beta_{\max})}(\mathcal{R}(t)(m_j(t) - r(t))) \tag{26}$$

A visualization of this set is provided in figure 3. The geometric intuition behind the SDF function (22) is that it defines the magnitude of the orthogonal projection of any vector input to the boundary of a set, with a sign to indicate whether this point belongs inside or outside the set, giving it favorable numerical properties when applying smoothing operations about the set boundary. With this definition in place, we still need to deal with the discontinuity existing in the aforementioned noise model.

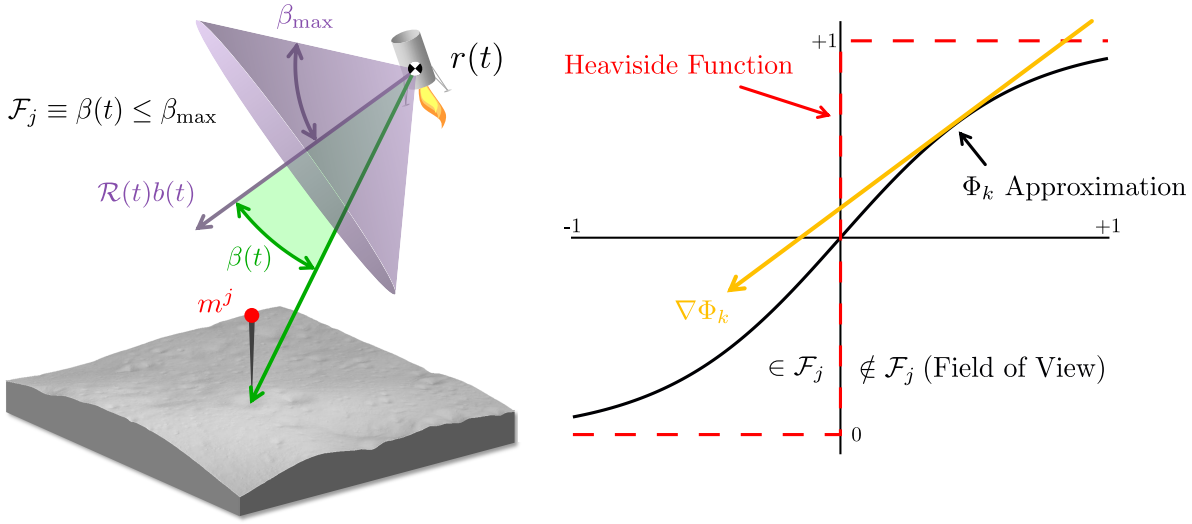


Fig. 4 Left: Visual interpretation of the geometry underlying the FOV set $\mathcal{F}_j$. Right: Visual interpretation of the sigmoid smoothing approximation Φ_κ and what it provides for gradient-based trajectory optimization. The gradient of this function, $\nabla\Phi_\kappa$, is non-zero at all points in its domain, and when performing numerical gradient descent to minimize uncertainty associated with the measurement noise σ_j (or some proximal objective that implicitly seeks to minimize this uncertainty), the optimizer "pushes" in the direction towards perceiving landmarks in the FOV of the vehicle's range-based sensor.

Definition 3 ***(Sigmoid Function).*** *The sigmoid function $\Phi_\kappa : \mathbb{R} \to \mathbb{R}$ [34] provides a $\mathcal{C}^1$-differentiable expression which tends towards 0 as $x \to -\infty$ and towards 1 as $x \to +\infty$, with a sharpening parameter $\kappa \in \mathbb{R}$, where $\kappa \to \infty$ recovers the Heaviside (step) function, and is defined as:*

$$\Phi_\kappa(x) \triangleq 1 - (1 + \exp(\kappa x))^{-1} \tag{27}$$

Using definition 3, we can develop a smooth approximation to the Heaviside (step) function, and compose this approximation with the numerically well-conditioned SDF in equation (26) to obtain the following approximation to the range-based noise standard deviation, which we denote $\sigma^{-1}_{\kappa,j}(z)$:

$$\sigma^{-1}_{\kappa,j}(z) = \tilde{\sigma}_j^{-1}(z)\left(1 - \Phi_\kappa(\Psi_{\mathcal{F}_j}(z))\right)$$

Then, the measurement noise covariance matrix is defined as a diagonal matrix, where the jth diagonal element of the matrix inverse is defined as follows:

$$\left(V_\kappa(z)^{-1}\right)_{j,j} = \left(\sigma^{-1}_{\kappa,j}(z)\right)^2 \tag{28}$$

We now revisit the EKBF system of equations (14)-(16), and modify the covariance update component with the smoothed approximation to the Kalman gain as follows:

$$\dot{\Sigma}(t) = F(t)\Sigma(t) + \Sigma(t)F(t)^\top + W(t) - K_\kappa(\mu(t))H(t)\Sigma(t) \tag{29}$$

$$K_\kappa(\mu(t)) = \Sigma(t)H(t)^\top V_\kappa(\mu(t))^{-1} \tag{30}$$

While the covariance dynamics above are well-defined in a forward-horizon (predictive control) sense, we must now confront a separate issue, in that the observation vector $y(t)$ is not well-defined before any actual measurements can be made. To address this issue, we must leverage one further assumption:

Assumption 3. *In the absence of online measurements, we assume that the observation vector matches the predictive measurement model, such that:*

$$y(t) = h(z(t)) \quad \forall z(t) \in \mathbb{R}^{n_z} \tag{31}$$

Using assumption 3, we see that the EKBF mean update equation (14) has the Kalman update term canceled out, such that it simplifies to:

$$\dot{\mu}(t) = f(\mu(t), u(t)) \tag{32}$$

Using the new mean and covariance system of differential equations (29),(30),(32), we can then define the smoothed approximation to the dynamical mapping $g_\kappa : \mathbb{R}^{n_\zeta \times n_u} \to \mathbb{R}^{n_\zeta}$ with no explicit measurements such that the following relationship satisfies the aforementioned system of equations:

$$\dot{\zeta}(t) = g_\kappa(\zeta(t), u(t)) \tag{33}$$

In the following section, equation (33) will be the primary dynamical relationship used to express a Gaussian trajectory distribution as a system of smoothed differential equations. Figure 4 provides visual intuition for the FOV set and why the sigmoid function approximation based around this set is valuable for gradient-based trajectory optimization. In section V, we will discuss a technique which allows us to systematically update the sharpening term κ to recover the true noise model (20) while maintaining numerical stability.

Remark 1 ***(State-dependent noise).*** *While the focus of this work is on the construction of perceptive noise models, we note that many other common aerospace sensors may exhibit these discontinuous state-dependent limitations, such as altimeters (with a corresponding maximum altitude limit) or star trackers (with both FOV restrictions and sun incidence angle avoidance). To this end, both the discontinuous noise modeling approach and the consequent SDF-composed sigmoid smoothing methodology can be readily applied to all such sensor types.*

Remark 2 ***(Noise parameter terms).*** *This approach can also readily support inclusion of noise parameter terms, such as biases, scaling factors and misalignment errors, at the cost of additional states to model.*

C. Vehicle Dynamics Model

Noting the requirement of assumption 1, we make use of the point-mass cartesian 6-DOF vehicle model as described in [35] and commonly considered for the PDG problem. Moreover, the vehicle state $x(t)$ consists of a rigid-body mass $m_x(t) \in \mathbb{R}$ (not to be confused with the map state $m(t)$), inertial position $r_\mathcal{I}(t) \in \mathbb{R}^3$, inertial velocity $v_\mathcal{I}(t) \in \mathbb{R}^3$, body-relative unit quaternion attitude $q_{\mathcal{I}\to\mathcal{B}}(t) \in \mathbb{R}^4$ and body-fixed angular velocity $\omega_\mathcal{B}(t) \in \mathbb{R}^3$, such that $x = (m_x, r_\mathcal{I}, v_\mathcal{I}, q_{\mathcal{I}\to\mathcal{B}}, \omega_\mathcal{B})$. The vehicle control input $u(t)$ consists of a single body-fixed thrust term $T_\mathcal{B}(t) \in \mathbb{R}^3$ and torque term $M_\mathcal{B}(t) \in \mathbb{R}^3$, such that $u = (T, M)$. This leads to the following system of differential equations defining the vehicle dynamics:

$$\textbf{Mass:} \qquad \dot{m}_x(t) = -\alpha \|T_\mathcal{B}(t)\|_2 \tag{34}$$

$$\textbf{Position:} \qquad \dot{r}_\mathcal{I}(t) = v_\mathcal{I}(t) \tag{35}$$

$$\textbf{Velocity:} \qquad \dot{v}_\mathcal{I}(t) = \frac{1}{m(t)} R_{\mathcal{I}\to\mathcal{B}}(t)^\top T_\mathcal{B}(t) + g_\mathcal{I} \tag{36}$$

$$\textbf{Attitude:} \qquad \dot{q}_{\mathcal{I}\to\mathcal{B}}(t) = \frac{1}{2}\Omega\left(\omega_\mathcal{B}(t)\right) q_{\mathcal{I}\to\mathcal{B}}(t) \tag{37}$$

$$\textbf{Angular Velocity:} \qquad \dot{\omega}_\mathcal{B}(t) = J_\mathcal{B}^{-1}\left(M_\mathcal{B}(t) - [\omega_\mathcal{B}(t)\times] J_\mathcal{B} \omega_\mathcal{B}(t)\right) \tag{38}$$

Where $\alpha \in \mathbb{R}$ denotes the vehicle engine's mass flow constant in terms of vacuum-specific impulse $I_{\text{sp}} \in \mathbb{R}$ and Earth's standard gravity magnitude $g_0 \in \mathbb{R}$, $g_\mathcal{I} \in \mathbb{R}^3$ denotes the local gravitational acceleration expressed in the inertial frame, and $J_\mathcal{B} \in \mathbb{R}^{3\times3}$ denotes the vehicle's moment of inertia matrix in body frame coordinates. The terms $[a\times], \Omega(a)$ denote skew-symmetric matrix operations on a vector $a \in \mathbb{R}^3$ and are defined as follows:

$$[a\times] \triangleq \begin{bmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{bmatrix} \qquad \Omega(a) \triangleq \begin{bmatrix} 0 & -a_x & -a_y & -a_z \\ a_x & 0 & a_z & -a_y \\ a_y & -a_z & 0 & a_x \\ a_z & a_y & -a_x & 0 \end{bmatrix}$$

Additionally, as was mentioned in assumption 1, we require an expression of the vehicle's attitude in terms of a direction cosine matrix (DCM) $R_{\mathcal{I}\to\mathcal{B}}(t) \in \mathrm{SO}(3)$. The following transformation can be made from a unit quaternion $q_{\mathcal{I}\to\mathcal{B}}(t) = [q_0, q_1, q_2, q_3]^\top$:

$$R_{\mathcal{I}\to\mathcal{B}}(t) = \begin{bmatrix} 1-2(q_2^2+q_3^2) & 2(q_1q_2+q_0q_3) & 2(q_1q_3-q_0q_2) \\ 2(q_1q_2-q_0q_3) & 1-2(q_1^2+q_3^2) & 2(q_2q_3+q_0q_1) \\ 2(q_1q_3+q_0q_2) & 2(q_2q_3-q_0q_1) & 1-2(q_1^2+q_2^2) \end{bmatrix}$$

IV. ACT-SLAM Problem Formulation

In this section, we make use of the well-posed, $\mathcal{C}^1$-differentiable filtering dynamics model derived in the previous section in equation 33, and propose a general-purpose continuous-time optimal control problem (CT-OCP) formulation (ACT-SLAM) to solve the active sensing guidance problem. This process connects the state estimation (*navigation*) component with solution to a trajectory optimization problem (*guidance*), and serves as a solidified instance of guidance-navigation co-design. Consider the following free-final-time CT-OCP defined over a time interval $t \in [t_0, t_f]$:

ACT-SLAM Problem (39)

$$\begin{aligned}
\min_{\zeta(\cdot),u(\cdot)} \quad & \mathcal{J}(\zeta(\cdot), u(\cdot)) && \text{(39a)} \\
\text{subject to} \quad & \dot{\zeta}(t) = g_\kappa(\zeta(t), u(t)) && \text{(39b)} \\
& \mu(t) \in \mathcal{X}(t) && \text{(39c)} \\
& u(t) \in \mathcal{U}(t) && \text{(39d)} \\
& \zeta(t_0) \in \Lambda_0 && \text{(39e)} \\
& \zeta(t_f) \in \Lambda_f && \text{(39f)} \\
& \forall t \in [t_0, t_f]
\end{aligned}$$

Problem (39) formulates the ACT-SLAM problem over the full filtering state $\zeta(t) = (\mu(t), \mathrm{vec}(\Sigma(t)))$ and control input $u(t)$. The formulation contains a cost function $\mathcal{J} : \mathbb{R}^{n_\zeta} \times \mathbb{R}^{n_u} \to \mathbb{R}$ that is optimized over in (39a), smoothed filtering dynamics in equation (39b) originally defined in equation (33), path constraints defined over the mean state and input by sets $\mathcal{X}(t)$ and $\mathcal{U}(t)$, respectively, in equations (39c)-(39d), and boundary conditions defined on the initial and terminal filtering state by sets Λ_0 and Λ_f, respectively, in equations (39e)-(39f). Notably, these boundary conditions can be considered probabilistic *priors* and *posteriors* in the context of the guidance horizon.

A. Co-design Configurations

The ACT-SLAM formulation presented in problem (39) enables a variety of candidate co-design modeling approaches for practical mission scenarios–we discuss several possible approaches here. One such approach is a direct specification of navigational requirements at the terminal point in time. Within aerospace contexts, these are typically modeled as 3σ error specifications. Consider, for example, a requirement that the terminal position in the inertial $+z$ element must fall within a 3σ uncertainty level ρ_{r_z}, and denote index k to extract this position element from the covariance matrix $\Sigma(t)$. Then, we could specify the following as part of constraint (39f):

$$\sqrt{\Sigma_{k,k}(t_f)} \leq \frac{1}{3}\rho_{r_z} \tag{40}$$

Perhaps this approach may be used while satisfying some baseline mission-level objective, such as a minimum-fuel-usage objective as is typically seen for PDG scenarios:

$$\mathcal{J}_{\text{minfuel}} = \int_{t_0}^{t_f} \|T(t)\|_2 \, \mathrm{d}t \tag{41}$$

If, however, there are no strict navigational requirements associated with the terminal uncertainty, one may instead wish to perform an *uncertainty-minimizing* maneuver. One technique to accomplish this is to apply the concept of differential entropy [17], which seeks to minimize the collective "surprise" associated with a probability distribution. It has been shown that, for Gaussian distributions, this term is proportional to the logarithmic determinant of the covariance of the distribution. As such, we can formulate the following objective:

$$\mathcal{J}_{\text{entropy}} = \int_{t_0}^{t_f} \log\det \Sigma(t) \, \mathrm{d}t \tag{42}$$

Many mission design scenarios may also desire to combine these two types of configurations. For example, there may be certain navigational requirements which are hard-set and can be encoded as constraints, whereas other elements of the navigational solution must be optimized over to search for the best possible solution, as in [13]. This is not an exhaustive list of examples, either, but is meant to provide several highlighted options for the practitioner of these techniques. Going forward, we will use this differential entropy objective to produce results in section VI.

V. Successive Convexification

In this section, we present implementation and adaption of the continuous-time successive convexification (CT-SCvx) algorithm to solve general-purpose nonconvex CT-OCP problems with embedded homotopic transformations [36]. We take strong inspiration from the approach in [22], which proposes a method to model and solve free-final-time CT-OCPs with guaranteeable constraint satisfaction at all points in time, rather than just at discrete knot points. This method falls under a class of techniques known as sequential convex programming (SCP) [9, 10, 37], which iteratively solve convex approximations (subproblems) of the original nonconvex problem to converge towards an optimized solution, and have been shown to provide strong properties in terms of real-time tractability and systematic convergence.

A. Model Augmentations

We now consider two augmentations to our state-space model (and the consequent dynamics) to accommodate additional modeling objectives. The first augmentation enables us to solve the CT-OCP problem such that t_f may implicitly become an optimization variable (as well as all other times $t \in [t_0, t_f]$), making the problem *free-final-time*. We consider the time dilation formulation introduced in [22, 35], where we define wall-clock time as a smooth, strictly-monotonic mapping $t : [0,1] \to \mathbb{R}_+$ such that $t(0) = t_0$ and $t(1) = t_f$. We will consequently define a normalized time $\tau \in [0,1]$ and time derivative $s(\tau) = \partial t(\tau)/\partial\tau > 0$. Through the chain rule, we can now define a dynamical relationship on the normalized time grid as follows:

$$\mathring{\zeta}(t(\tau)) = \frac{\partial \zeta(t(\tau))}{\partial \tau} = s(\tau) g(\zeta(t(\tau)), u(t(\tau))) \tag{43}$$

Where $s(\tau)$ becomes an explicit optimization variable. The second augmentation concerns guaranteeable satisfaction of path constraints at all points in time. Consider a reformulation of constraints (39c)-(39d) to explicit functions as follows:

$$p(\zeta(t), u(t)) \leq 0 \tag{44}$$
$$h(\zeta(t), u(t)) = 0 \tag{45}$$

It has been shown in Lemma 2 of [22] that equations (44)-(45) are equivalently solved by satisfaction of the following integral:

$$\int_{t_0}^{t_f} h(\zeta(t), u(t))^2 + \max(0, p(\zeta(t), u(t)))^2 \, \mathrm{d}t = 0 \tag{46}$$

Moreover, we can define an additional explicit optimization variable $\theta \in \mathbb{R}$ that satisfies the following system of equations:

$$\dot{\theta}(t) = \Lambda(\zeta(t), u(t)) \triangleq h(\zeta(t), u(t))^2 + \max(0, p(\zeta(t), u(t)))^2 \tag{47}$$
$$\theta(t_0) = \theta(t_f) \tag{48}$$

Which equivalently satisfies equation (46) and the path constraints (44)-(45) by extension. Finally, we can combine these two model augmentations together, by considering the following modifications to the state-space model used in problem (39). We start by redefining the optimization state and input variables through concatenation:

$$\tilde{\zeta}(\tau) = (\zeta(\tau), \theta(\tau), t(\tau)) \tag{49}$$
$$\tilde{u}(\tau) = (u(\tau), s(\tau)) \tag{50}$$

Then, the filter dynamics can be redefined as follows:

$$G_\kappa(\tilde{\zeta}, \tilde{u}) \triangleq s \begin{bmatrix} g_\kappa(\zeta, u) \\ \Lambda(\zeta, u) \\ 1 \end{bmatrix} \tag{51}$$

B. Linearization & Discretization

One of the key details behind the CT-SCvx algorithm is the direct linearization of the nonconvex (filtering) dynamics, to obtain a convex constraint representation in the CT-SCvx subproblem formulation. We consider the following jacobian terms associated with the dynamics:

$$A_\kappa(\tau) = \nabla_{\tilde{\zeta}} G_\kappa(\tilde{\zeta}(\tau), \tilde{u}(\tau)) \tag{52}$$
$$B_\kappa(\tau) = \nabla_{\tilde{u}} G_\kappa(\tilde{\zeta}(\tau), \tilde{u}(\tau)) \tag{53}$$

Then, the continuous-time linear time varying (CT-LTV) dynamics are described through a first-order Taylor series expansion:

$$\mathring{\tilde{\zeta}}(\tau) = A_\kappa(\tau)\tilde{\zeta}(\tau) + B_\kappa(\tau)\tilde{u}(\tau) + w_\kappa(\tau) \tag{54}$$

Where $w_\kappa(\tau)$ corresponds to the remaining Taylor series terms:

$$w_\kappa(\tau) = G_\kappa(\bar{\zeta}(\tau), \bar{u}(\tau)) - A_\kappa(\tau)\bar{\zeta}(\tau) - B_\kappa(\tau)\bar{u}(\tau) \tag{55}$$

And $(\bar{\zeta}(\tau), \bar{u}(\tau)), \tau \in [0, 1]$ defines a reference trajectory. Next, to perform discretization, the time interval is discretized into a discrete set of knot points $\tau_k, k = 1, \ldots, N$, and a finite-dimensional control parameterization is chosen. In the interest of brevity, we present the formulation as it pertains to a first-order hold (FOH) parameterization, however, in practice, any finite tractable parameterization can be used. We can consequently form a state transition matrix and solve the initial value problem over each knot point interval to arrive at the following discrete-time linear time varying (DT-LTV) dynamical relationship:

$$\tilde{\zeta}_{k+1} = A_k\tilde{\zeta}_k + B_k^- \tilde{u}_k + B_k^+ \tilde{u}_{k+1} + w_k \tag{56}$$

The details on derivation of the terms A_k, B_k^-, B_k^+, w_k can be found in [15, 22] for the interested reader. We drop the κ subscript in expression of these terms for notational convenience, however we note their implicit dependence on this term nonetheless.

C. Forming the Convex Subproblem

With the aforementioned details in place, we are now ready to form the convex subproblem, as part of the CT-SCvx framework, through implementation of the *prox-linear method*. This technique involves the embedding of a trust region term on the optimization variables directly, as well as an ℓ_1-norm penalization term on slack variables applied to convexified constraints (which, for this problem formulation, refer only to

the nonlinear filter dynamics), to the main objective function. These terms help alleviate concerns of *artificial infeasibility* and *unboundedness* [37], and are important for numerical implementation. We recall that the original objective is denoted as $\mathcal{J}$, and define the following additional components of the objective function:

$$\mathcal{J}_\eta(\tilde{\zeta}_K, \tilde{u}_K) \triangleq \sum_{k=1}^{N} \delta\tilde{\zeta}_k^\top \delta\tilde{\zeta}_k + \delta\tilde{u}_k^\top \delta\tilde{u}_k \tag{57}$$

$$\mathcal{J}_\nu(\nu_K) \triangleq \|\nu_K\|_1 \tag{58}$$

Where:

$$\delta\tilde{\zeta}_k \triangleq \tilde{\zeta}_k - \bar{\zeta}_k \tag{59}$$

$$\delta\tilde{u}_k \triangleq \tilde{u}_k - \bar{u}_k \tag{60}$$

$$\tilde{\zeta}_K \triangleq (\tilde{\zeta}_1, \ldots, \tilde{\zeta}_N) \tag{61}$$

$$\tilde{u}_K \triangleq (\tilde{u}_1, \ldots, \tilde{u}_N) \tag{62}$$

$$\tilde{\nu}_K \triangleq (\tilde{\nu}_1, \ldots, \tilde{\nu}_N) \tag{63}$$

Here, we define $\nu_k \in \mathbb{R}^{n_z}$ as slack variables applied to the filtering dynamics to allow dynamic violation during convergence. The reference trajectory is defined by $(\bar{\zeta}_k, \bar{u}_k)_{k=1}^N$, which either corresponds to the solution to the previous subproblem iteration, or to an initial guess on the first iterate. We note that, in the case that $\mathcal{J}$ is nonconvex, the jacobian of this cost must be used in its place to ensure a convex problem definition. Finally, the following prox-linear subproblem is defined:

ℓ_1-Penalized Prox-Linear Subproblem (64)

$$\min_{\tilde{\zeta}_K, \tilde{u}_K, \nu_K} \quad \mathcal{J}(\tilde{\zeta}_K, \tilde{u}_K) + w_\eta \mathcal{J}_\eta(\tilde{\zeta}_K, \tilde{u}_K) + w_\nu \mathcal{J}_\nu(\nu_K) \tag{64a}$$

$$\text{subject to} \quad \tilde{\zeta}_{k+1} = A_k\tilde{\zeta}_k + B_k^- \tilde{u}_k + B_k^+ \tilde{u}_{k+1} + w_k + \nu_k \tag{64b}$$

$$\theta_{k+1} - \theta_k \leq \epsilon_{\text{CTCS}} \tag{64c}$$

$$\tilde{\zeta}_1 \in \Lambda_0 \tag{64d}$$

$$\tilde{\zeta}_N \in \Lambda_f \tag{64e}$$

$$\forall k = 1, \ldots, N$$

Where w_η, $w_\nu \in \mathbb{R}_+$ correspond to relative weighting terms for $\mathcal{J}_\eta$ and $\mathcal{J}_\nu$, respectively. Problem (64) contains the expanded objective weighted with trust region and dynamic violation penalization terms in (64a), the DT-LTV filter dynamics with the slack violation term in constraint (64b), and the boundary conditions in constraints (64d)-(64e). Notably, an additional (convex) constraint (64c) is applied to ensure satisfaction of equation (48) and consequent satisfaction of the path constraints, with a small slack term $\epsilon_{\text{CTCS}} \sim 1e\text{-}4$ to ensure satisfaction of the linear independence constraint qualification (LICQ) condition [22]. This subproblem is solved in an iterative manner until the following convergence criteria are met:

$$\mathcal{J}_\eta(\tilde{\zeta}_K, \tilde{u}_K) \leq \epsilon_\eta \tag{65}$$

$$\mathcal{J}_\nu(\nu_K) \leq \epsilon_\nu \tag{66}$$

Additionally, now that we have an actionable formulation in the form of problem (64), we can reassess the impact of the sharpening parameter κ embedded within the filtering dynamics in equation (33), and consequently in each prox-linear method subproblem as noted in section V.B. Moreover, we would like to make use of an update law for this parameter, in tandem with the subproblem iteration, such that the

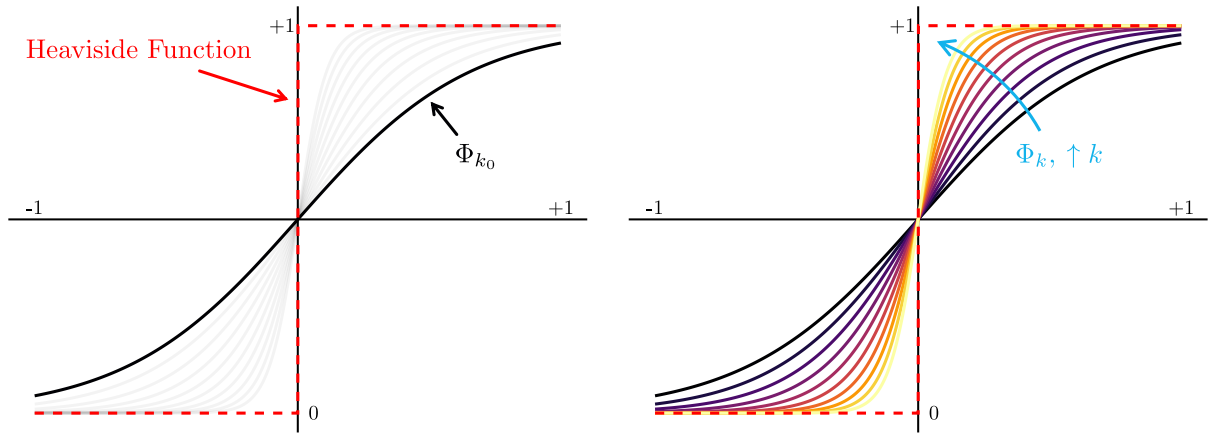


Fig. 5 Illustrative diagram of the embedded numerical continuation process to provide an update law for the iterate-varying smoothing parameter κ_k.

parameter can be systematically increased, and the corresponding step function approximation in the noise model can be tightened. We take direct inspiration from the *embedded numerical continuation* methodology presented in [36], where we conditionally update κ_k, indexed for the kth CT-SCVX iteration, according to a geometric progression:

$$\kappa_k = \begin{cases} \max(\alpha\kappa_{k-1}, \kappa_{\min}), & \Delta\mathcal{J}_k \in [\beta_L, \beta_U] \\ \kappa_{k-1}, & \text{else} \end{cases} \tag{67}$$

Where $\Delta\mathcal{J}_k = \mathcal{J}_k - \mathcal{J}_{k-1}$ and κ_0 represents the initial tightening parameter. We then apply an additional convergence criterion ontop of criteria (65)-(66) which states that κ_k has converged to its maximum allowable tightness:

$$\kappa_k = \kappa_{\min} \tag{68}$$

An illustration of this technique is provided in figure 5. We refer the user to [36] for more details, and for derivation of parameters $\alpha, \kappa_0, \kappa_{\min}, \beta_L, \beta_U$.[‡]

VI. Results & Analysis

In this section, we consider a full-scale analysis of the ACT-SLAM algorithm applied to a practical PDG scenario inspired by [35]. Specifically, we consider a descent scenario with three modeled landmarks located on a realistic digital elevation map (DEM), specifically corresponding to a designated Lunar south pole site of interest [38]. This is considered a reasonable number of landmarks, consistent with modern techniques [28], for the purpose of online SLAM in the context of lunar landing.

A. Scenario Parameters

The baseline scenario parameters (including vehicle configuration, path constraints and mean boundary conditions), as well as the (x, y)-components of the landmark positions (defined in a landing-site-relative *East-North-Up* (ENU) reference frame), can be found in [35]. The z-components of the landmarks are consequently taken from the Lunar site #1 DEM [38] at the referenced (x, y) coordinates. In terms of uncertainty priors, the covariance prior is set such that the vehicle position diagonal elements are equal to $(100)^2$ m, and landmark position diagonal elements are equal to $(30)^2$ m. For the purpose of this analysis, the process noise covariance matrix $W(t)$ is set to be zero. Additional model and algorithm parameters are provided in table 2.

[‡]In [36], the parameters $\alpha, \kappa_0, \kappa_{\min}$ are represented in expanded form with parameters $N_h, \epsilon_k, \delta_0, \delta_1$.

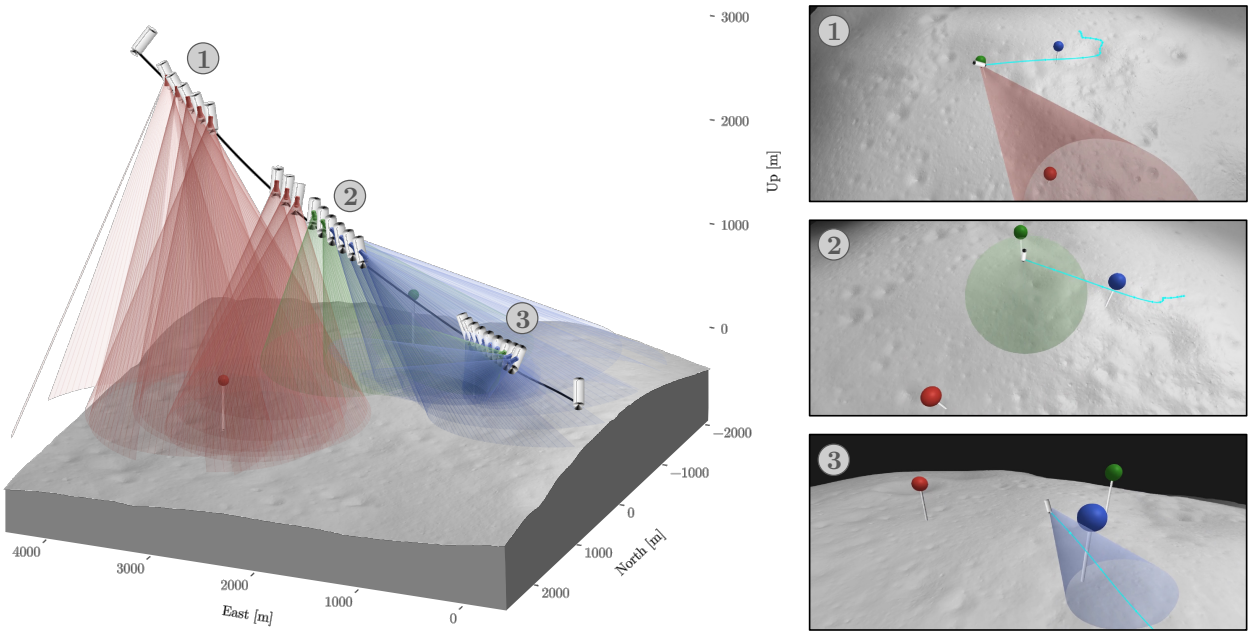


Fig. 6 Left: Trajectory timelapse for the baseline PDG scenario using ACT-SLAM, where perceptive scans to each landmark are displayed with a 3 second delay between successive scans, and color-coded according to the first, second and third landmarks. Right: Snapshots of the baseline scenario in a high-fidelity visualization environment at different points in time.

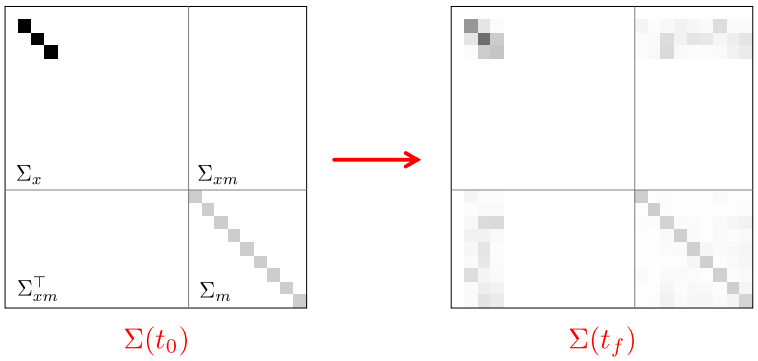


Fig. 7 Example covariance evolution for the baseline scenario across the timeframe $t \in [0, t_f]$, where relative magnitudes between elements are represented by grayscale color intensity *(darker = higher magnitude)* under an element-wise square root mapping for visual clarity. Matrix partitions are demarcated by lines.

B. Baseline Scenario

Figure 6 illustrates an ACT-SLAM-generated solution for the baseline PDG scenario, with scans to each landmark displayed. As can be seen, the algorithm is able to obtain substantial coverage of each landmark, which aids in reducing vehicle uncertainty as well as landmark uncertainties over the full guidance horizon. The relative uncertainty changes can be visualized in the elements of the covariance matrix $\Sigma(t)$ as seen in figure 7, where diagonal elements correspond to direct navigational objectives or requirements. For this particular scenario, the landmark uncertainty priors were much smaller than the positional uncertainty priors, and so uncertainty reduction was mainly consolidated to the vehicle's position. This is consistent with expectations, and is a common observation made for terrain-relative navigation scenarios in contexts where terrain mapping can be performed *a-priori*, such that the vehicle localization portion of the SLAM problem is the primary focus for performance improvements. Figure 8 provides additional results for the trajectory of

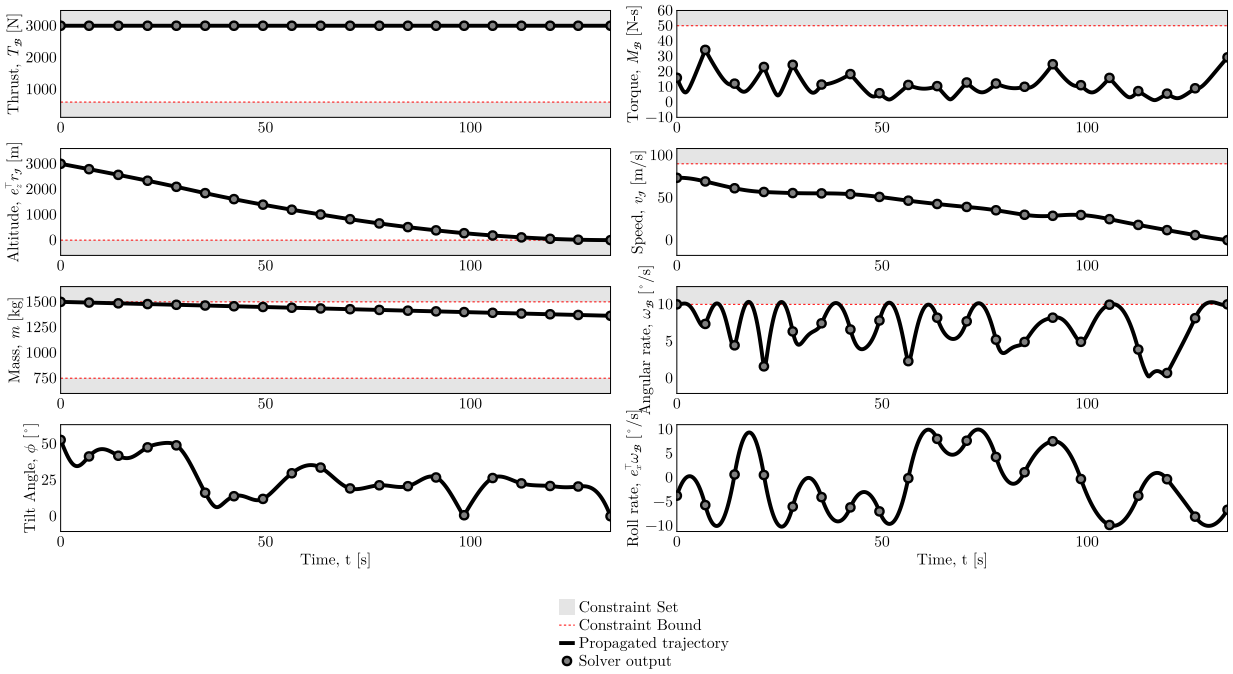


Fig. 8 Vehicle trajectory signals and corresponding constraint sets associated with simulation of the baseline scenario.

the lander in terms of the vehicle state and control input.

C. Model Validation

In light of the sigmoid smoothing approximation and measurement prediction assumption made in section III, as well as inherent error associated with the EKBF's linearization-based approximation of the covariance propagation, we desire to quantify the impacts of these approximations through statistical analysis. To facilitate this analysis, we can consider specification of two separate states: the *truth* state x_{truth} and the *navigational* state x_{nav}. We begin by sampling an initial truth state $x_{\text{truth},0}$ about the prior Gaussian distribution (μ_0, Σ_0), and forward-simulate (integrate) this state according to the stochastic process model in equation (1). Simultaneously, we instantiate an EKBF filter based around equations (14)-(16). For validation purposes, we sample perceptive measurements $y(t)$ (lifting assumption 3), with FOV triggered as a step function based on the truth state $x_{\text{truth}}(t)$ at the current point in time (lifting the sigmoid approximation), to propagate and forward-simulate the EKBF model. Consequently, the navigational state is recorded as the mean of the filter $x_{\text{nav}}(t) = \mu(t)$. For analysis purposes, we consider the navigation-truth dispersion:

$$\delta x \triangleq x_{\text{nav}} - x_{\text{truth}} \tag{69}$$

Statistical analysis is performed through a Monte Carlo assessment over $n = 1000$ trials, distinguished by the sampling of state $x_{\text{truth},0}$ and model noise $w(t), v(t)$ according to their previously-defined distributions. Figure 9 provides the navigation-truth dispersion error histories across all trials for the vehicle position components, and compares them against the 3σ level sets of the Gaussian trajectory distribution provided by ACT-SLAM in the form of $\Sigma(t)$. Notably, the 3σ level set boundaries are only crossed in 1.1% of trials, corresponding to a 98.9% success rate–even in the cases where the boundaries are crossed, the magnitude of error is generally very small, suggesting high-precision approximation performance.

D. Literature Comparison

While sections VI.B and VI.C establish the viability of ACT-SLAM in isolation, we ultimately seek to understand how this technique compares to established methods in terms of active sensing guidance objectives.

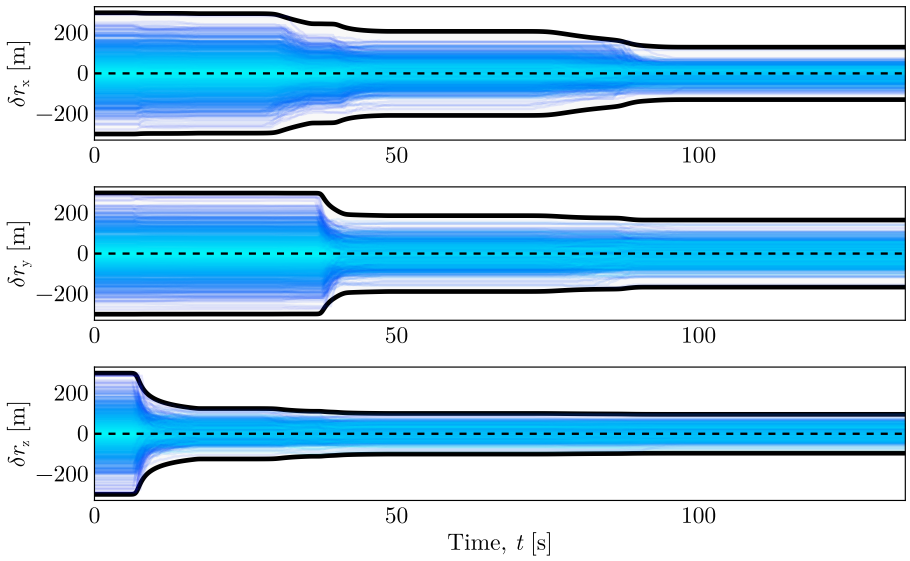


Fig. 9 Monte Carlo analysis of ACT-SLAM model validation simulation over $n = 1000$ trials. Plots displaying navigation-truth dispersions are shown for each element of the vehicle position, where trajectories are color-coded according to the magnitude of the dispersion at $t = 0$. The 3σ level sets from the ACT-SLAM-output covariance are displayed by symmetric black curves.

In the following subsection, we review two rivaling techniques and show how ACT-SLAM compares to them in terms of various performance benchmarks.

1. Passive Perception Baseline

The first reference method concerns a passive perception algorithm which will serve as a baseline going forward. Borrowing from the modeling approach described in [14], we define a passive perception approach as one which does not opportunistically optimize for an information-theoretic objective, but independently computes a feasible trajectory for the scenario (in this case, under a separate objective), and modifies the trajectory to satisfy information-theoretic objectives *post hoc*.

We first compute a deterministic 6-DoF solution to the powered descent guidance problem using the CT-SCVX algorithm across only the vehicle mean state μ_x (neglecting mapping elements and covariance dynamics) using the minimum-fuel objective described in equation (41) and all original constraints imposed on the vehicle mean state in the baseline scenario given by section VI.B. Next, we restrict our modeling considerations to a *non-gimbaled* lander configuration such that the thrust vector is fixed in the body frame along the longitudinal x-axis with $\|T_{\mathcal{B}}(t)\|_2 = e_1 = [1, 0, 0]^\top$. Under this condition, we remark that the system can be considered *roll-decoupled*, such that the lander can freely rotate (roll) about the $\mathcal{B}$-frame x-axis without affecting translational degrees of freedom. Using this property, we modify the prior 6-DoF solution's rotation profile $\mathcal{R}(t)$ with roll adjustments that prioritize pointing towards the closest landmark at any time in the trajectory's history. To do this, we make use of the pointing direction projection technique described in [35]. Let us denote the provided minimum-fuel trajectory as $\{x^-(t), u^-(t)\}$, $t \in [t_0, t_f]$ using superscript $\square^-$. For each landmark, we denote the desired pointing direction to landmark j, expressed in the body frame, as $p_j(t) = \mathcal{R}^-(t)(m_j^-(t) - r^-(t))/\|m_j^-(t) - r^-(t)\|_2$, and the current boresight pointing direction as b. We can also define a projection matrix onto the $\mathcal{B}$-frame y-z (roll) plane as:

$$P \triangleq \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{70}$$

First, we must obtain the rotation matrix about the $\mathcal{B}$-frame roll axis, $\mathcal{R}_j(t)$, which most-closely points $\mathcal{R}_j(t)b$

Table 2 Supplementary scenario parameters.

Parameter	Value	Parameter	Value
ϵ_η	1e−4	N	20
ϵ_ν	1e−4	w_η	1e0
ϵ_{CTCS}	1e−4	w_ν	1e2
N_h	10	γ_1	100 m
ϵ_κ	0.01	γ_2	1e−5 m^{-1}
δ_0	1		
δ_1	0.1		
β_L	−1e−3		
β_U	1e−1		

in the direction of $p_j(t)$. This can be done by rotational alignment of the pointing directions projected to the roll plane. Let us denote the signed angle between two (unit norm) vectors x, y about a rotation axis n as:

$$\angle(x, y; n) \triangleq \text{atan2}((x \times y)^\top n, x^\top y) \tag{71}$$

Then, we can define $\mathcal{R}_j(t)$ by an axis-angle pair [39], such that $n(t) = \mathcal{R}(t)^\top e_1$ and $\theta(t) = \angle(Pb, Pp_j(t); n(t))$. Next, we can denote the unsigned (shortest-arc) angle between two (unit norm) vectors as:

$$\angle_{\text{u}}(x, y) \triangleq \text{acos}(x^\top y) \tag{72}$$

To determine the landmark we should point towards, we can select, for all times $t \in [t_0, t_f]$:

$$j^\star = \underset{j \in [n_\ell]}{\text{argmin}} \angle_{\text{u}}(\mathcal{R}_j(t)b, p_j(t)) \tag{73}$$

Then, the provided trajectory is fitted with a new orientation profile $\mathcal{R}^+(t) = \mathcal{R}_{j^\star}(t)\mathcal{R}^-(t)$, which represents pointing towards the closest landmark in angular deviation space. Consequently, if $\angle_{\text{u}}(\mathcal{R}_{j^\star}(t)b, p_j(t)) \leq \beta_{\max}$, then we have that landmark $j^\star$ is in field of view. We remark that this passive perception technique provides two salient benefits relative to ACT-SLAM: 1. computational efficiency due to only having to perform trajectory optimization on the reduced-order vehicle mean state space, and 2. preservation of the original minimum fuel objective in terms of optimality. Due to its computational efficiency, this solution is also used as an initial guess for problem (39). We also note that this technique can result in arbitrarily-large roll rates which may exceed the body rate constraints imposed on the scenario–for example, $\mathcal{R}_{j^\star}(t)$ is discontinuous in the general case when when the tracked landmark $j^\star$ changes. For the purpose of comparison, we consider this an acceptable expression of the *upper bound* on landmark tracking performance (and consequently, in covariance reduction) that a passive perception technique would be capable of obtaining, such that any method that must project the roll rate onto the feasible body rate constraint space would logically exhibit worse performance.

2. Information-Aware Powered Descent Guidance

The second reference method considered is the information-aware powered descent guidance technique recently explored in [15], denoted as INFO-PDG. Consider the covariance matrix for the jth landmark:

$$\Sigma^j = \begin{bmatrix} \Sigma^j_x & \Sigma^j_{xy} & \Sigma^j_{xz} \\ * & \Sigma^j_y & \Sigma^j_{yz} \\ * & * & \Sigma^j_z \end{bmatrix} \tag{74}$$

INFO-PDG constructs a covariance dynamics law which is restricted to only the z-component covariance element of each landmark, $\Sigma^j_z(t)$, and models the dynamics as a smooth Gaussian conditioned on the FOV

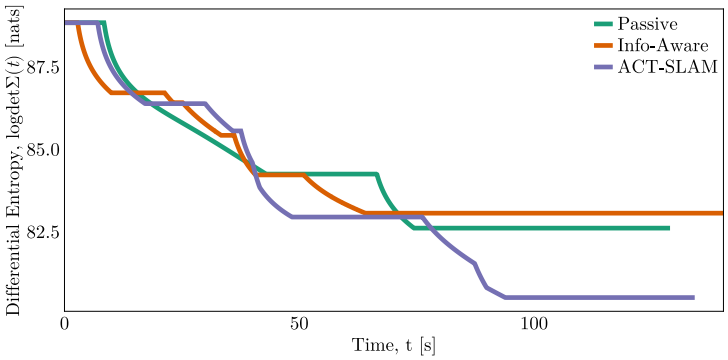


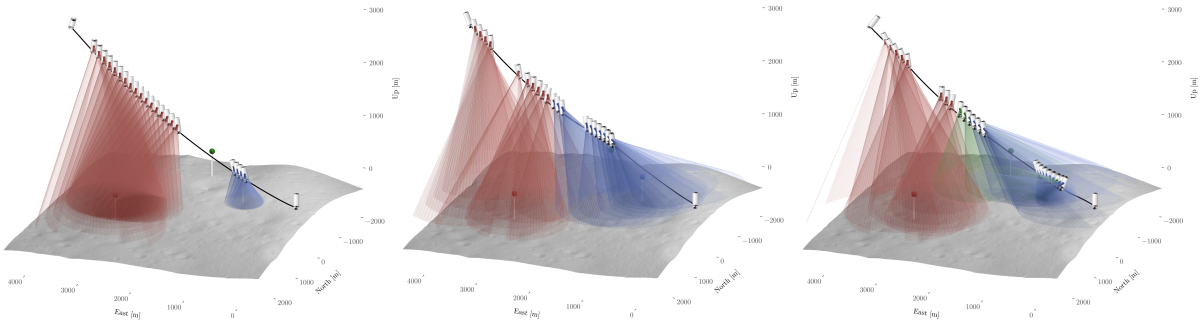
Fig. 10 Differential entropy history over time between the passive perception, Info-PDG and ACT-SLAM methods.

containment set $\mathcal{F}_j$, with an additional saturation condition to prevent the covariance elements from becoming negative. Using the conventions of this paper, this dynamics model can be summarized as:

$$\dot{\Sigma}_z^j(t) = \begin{cases} -\delta(\Phi_{-\kappa}(\max(0, F_j(\mu_x(t)))^2) - 1), & \Sigma_z^j(t) > 0 \\ 0, & \Sigma_z^j(t) \leq 0 \end{cases} \tag{75}$$

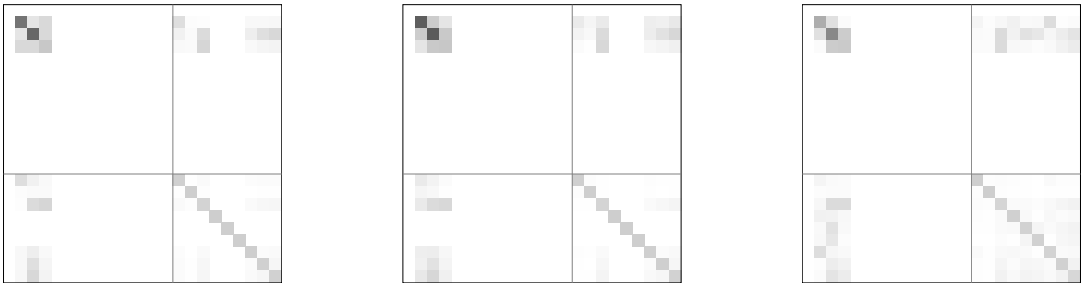
Where $F_j(\mu_x(t)) = \cos\beta_{\max} - (\mathcal{R}(t)^\top b)^\top (r(t) - m_j(t))/\|r(t) - m_j(t)\|_2$ denotes the FOV constraint set evaluation such that $F_j(\mu_x(t)) \leq 0 \iff \mu_x(t) \in \mathcal{F}_j$, and the composition $\max(0, F(\mu_x(t)))^2$ ensures that the covariance strictly decreases only when $\mu_x(t) \in \mathcal{F}_j$, driven by a rate of descent $\delta \in \mathbb{R}_+$. Consequently, the INFO-PDG objective is chosen as:

$$J_{\text{info}} = \int_{t_0}^{t_f} \sum_{j=1}^{n_\ell} \Sigma_z^j(t)\, \mathrm{d}t \tag{76}$$



(a) Passive Perception **(b) Info-PDG** **(c) ACT-SLAM**

Fig. 11 Trajectory timelapses for the passive perception, Info-PDG and ACT-SLAM methods, using the same visualization conventions as figure 6.



(a) Passive Perception **(b) Info-PDG** **(c) ACT-SLAM**

Fig. 12 Visualization of the terminal covariance matrix, $\Sigma(t_f)$, for the passive perception, Info-PDG and ACT-SLAM methods, using the same visualization conventions as figure 7.

When drawing modeling comparisons to ACT-SLAM in terms of the covariance dynamics model in equation (29), we can make several important observations. First, ACT-SLAM considers a much larger state space encompassing all covariance elements and their cross-correlations in accordance with an EKBF model, enabling a complete and rigorous expression of state estimation error at the cost of increased computational burden. This property also prevents the need for a switch statement as in equation (75), ensuring smoothness of the resulting dynamics function. Finally, ACT-SLAM does not require parameter tuning of δ and κ (the latter due to embedded numerical continuation) – this enables ACT-SLAM to be *tuning-free* by comparison and guarantees that sensor measurements are approximated to high numerical precision. In the case of INFO-PDG, these parameters are chosen as $\delta = 11$ and $\kappa = 1$ for the purpose of comparison.

3. Comparative Analysis

With both reference techniques established, we can now engage in a comparison between both of these methods and ACT-SLAM. Table 3 provides a summary of performance metrics used to compare between passive perception, INFO-PDG and ACT-SLAM. This analysis includes an additional metric introduced by [14] labeled *information gain* and characterized as the net difference in differential entropy over the maneuver duration:

$$\Delta I \triangleq \frac{1}{2}\log\det\Sigma(t_0) - \frac{1}{2}\log\det\Sigma(t_f) = \frac{1}{2}\log\frac{\det\Sigma(t_0)}{\det\Sigma(t_f)} \tag{77}$$

Under this metric, we observe that ACT-SLAM achieves a +29.1% and +36.3% improvement over passive perception and INFO-PDG, respectively. We can also consult the original objective in equation (42) and observe +1.3% and +1.1% improvements, respectively (noting the difference in improvement magnitudes due to logarithmic compression). Notably, we normalize the objective by the time of maneuver, t_f, to improve the comparison – we seek to characterize how fast the differential entropy metric decreases *in relation to* maneuver duration, such that we don't arbitrarily penalize a maneuver for having a longer duration. Figure 10 shows a time signal comparison between these methods in terms of differential entropy. In figure 11, we can see a visual comparison between the trajectories of each method and the sensor measurement scans taken for each landmark. Of particular note is the fact that, between all three methods, only ACT-SLAM manages to achieve visibility of all three landmarks. We contend that this is due to the fact that: 1. passive perception *greedily* prioritizes the closest landmark (in terms of angular deviation), and 2. INFO-PDG does not prioritize any landmark over another in its objective, provided the covariance associated with a landmark does not saturate to zero. ACT-SLAM, on the other hand, encodes an objective and dynamics law which results in up-weighting landmarks with higher covariances, prioritizing more mapping coverage and improving overall estimation performance. We note this as an important additional benefit of the ACT-SLAM method relative to its predecessors. Figure 12 displays the terminal covariance matrices associated with each method, further displaying the improvements in covariance reduction with respect to both the vehicle positional state and landmark states, in addition to more stable and uniform cross-correlations.

Referring back to table 3, we note that ACT-SLAM does exhibit worse performance in terms of the minimum fuel objective (expressed in terms of mass difference over the maneuver, $\Delta m = m(t_0) - m(t_f)$) and solve

Table 3 Literature Comparison Summary.

Method	$\uparrow \Delta I$ (nats)	$\downarrow \int \log\det \Sigma(t)\mathrm{d}t/t_f$ (nats/s)	$\downarrow \Delta m$ (kg)	$\downarrow$ Solve Time (s)
Passive [14]	3.11	84.45	**128.03**	**1.05**
INFO-PDG [15]	2.89	84.36	136.22	91.67
ACT-SLAM	**4.17**	**83.40**	136.82	96.12

time, relative to the baseline passive perception method. This is consistent with expectations and highlights an important tradeoff for improved information acquisition. In light of these tradeoffs, we contend that these results present ACT-SLAM as a promising approach for optimal information-theoretic trajectory generation, suitable for detailed offline analysis and nominal trajectory generation in the context of active sensing guidance. Future efforts to improve the solve time performance towards online (real-time) trajectory optimization may benefit from investigations of information filters and their sparsity properties [40].

VII. Conclusions

In this paper, the novel ACT-SLAM formulation, and consequent algorithm, is presented as a general-purpose solution to the active sensing guidance problem. This formulation also introduces the first established approach to solving guidance-navigation co-design under the template of continuous-time optimal control problems, making ACT-SLAM amenable to highly-constrained maneuvers in autonomous aerospace contexts, such as powered descent guidance. Future research directions include rigorous full-scale analysis of entry, descent & landing maneuvers with a full sensor suite model, and computational improvements to the algorithm through investigation of reduced-order modeling approaches.

References

[1] Steltzner, A., Kipp, D., Chen, A., Burkhart, D., Guernsey, C., Mendeck, G., Mitcheltree, R., Powell, R., Rivellini, T., San Martin, M., et al., "Mars Science Laboratory entry, descent, and landing system," *2006 IEEE Aerospace Conference*, IEEE, 2006, pp. 15–pp.

[2] Lewis, F. L., Vrabie, D., and Syrmos, V. L., *Optimal control*, John Wiley & Sons, 2012.

[3] 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.

[4] Açıkmeşe, B., and Blackmore, L., "Lossless convexification of a class of optimal control problems with non-convex control constraints," *Automatica*, Vol. 47, No. 2, 2011, pp. 341–347.

[5] Blackmore, L., Açıkmeşe, B., and Carson III, J. M., "Lossless convexification of control constraints for a class of nonlinear optimal control problems," *Systems & Control Letters*, Vol. 61, No. 8, 2012, pp. 863–870.

[6] 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.

[7] Harris, M. W., and Açıkmeşe, B., "Lossless convexification for a class of optimal control problems with quadratic state constraints," *2013 American Control Conference*, IEEE, 2013, pp. 3415–3420.

[8] Harris, M. W., and Açıkmeşe, B., "Lossless convexification of non-convex optimal control problems for state constrained linear systems," *Automatica*, Vol. 50, No. 9, 2014, pp. 2304–2311.

[9] Mao, Y., Szmuk, M., and Açıkmeşe, B., "Successive convexification of non-convex optimal control problems and its convergence properties," *2016 IEEE 55th Conference on Decision and Control (CDC)*, IEEE, 2016, pp. 3636–3641.

[10] Mao, Y., Szmuk, M., Xu, X., and Açikmese, B., "Successive convexification: A superlinearly convergent algorithm for non-convex optimal control problems," *arXiv preprint arXiv:1804.06539*, 2018.

[11] Calkins, G. E., Woffinden, D. C., and Putnam, Z. R., "Robust Trajectory Optimization for Guided Powered Descent and Landing," *AAS/AIAA Astrodynamics Specialists Conference*, 2022.

[12] Geller, D. K., "Linear covariance techniques for orbital rendezvous analysis and autonomous onboard mission planning," *Journal of Guidance, Control, and Dynamics*, Vol. 29, No. 6, 2006, pp. 1404–1414.

[13] Geller, D. K., and Christensen, D. P., "Linear covariance analysis for powered lunar descent and landing," *Journal of Spacecraft and Rockets*, Vol. 46, No. 6, 2009, pp. 1231–1248.

[14] Ticozzi, L., and Tsiotras, P., "Factor Graph-Based Active SLAM for Spacecraft Proximity Operations," *arXiv preprint arXiv:2501.10950*, 2025.

[15] Hayner, C. R., Pavlasek, N., Leung, K., Acikmese, B., and Carson, J. M., "Information-Aware Powered Descent Guidance for Entry, Descent and Landing," *AIAA SCITECH 2025 Forum*, 2025, p. 1896.

[16] Koga, S., Asgharivaskasi, A., and Atanasov, N., "Active exploration and mapping via iterative covariance regulation over continuous SE (3) trajectories," *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2021, pp. 2735–2741.

[17] Koga, S., Asgharivaskasi, A., and Atanasov, N., "Active SLAM over continuous trajectory and control: A covariance-feedback approach," *2022 American Control Conference (ACC)*, IEEE, 2022, pp. 5062–5068.

[18] Szmuk, M., Eren, U., and Acikmese, B., "Successive convexification for mars 6-dof powered descent landing guidance," *AIAA Guidance, Navigation, and Control Conference*, 2017, p. 1500.

[19] Okamoto, K., and Tsiotras, P., "Optimal stochastic vehicle path planning using covariance steering," *IEEE Robotics and Automation Letters*, Vol. 4, No. 3, 2019, pp. 2276–2281.

[20] Ridderhof, J., Okamoto, K., and Tsiotras, P., "Nonlinear uncertainty control with iterative covariance steering," *2019 IEEE 58th Conference on Decision and Control (CDC)*, IEEE, 2019, pp. 3484–3490.

[21] Echigo, K., Sheridan, O., Buckner, S., and Açıkmeşe, B., "Dispersion Sensitive Optimal Control: A Conditional Value-at-Risk-Based Tail Flattening Approach via Sequential Convex Programming," *IEEE Transactions on Control Systems Technology*, 2024.

[22] Elango, P., Luo, D., Kamath, A. G., Uzun, S., Kim, T., and Açıkmeşe, B., "Successive convexification for trajectory optimization with continuous-time constraint satisfaction," *arXiv preprint arXiv:2404.16826*, 2024.

[23] Ruymgaart, P. A., and Soong, T. T., *Mathematics of Kalman-Bucy Filtering*, Vol. 14, Springer Science & Business Media, 2013.

[24] Stachniss, C., Leonard, J. J., and Thrun, S., "Simultaneous localization and mapping," *Springer handbook of robotics*, 2016, pp. 1153–1176.

[25] Maybeck, P. S., *Stochastic models, estimation, and control*, Vol. 3, Academic press, 1982.

[26] Johansson, R., Verhaegen, M., and Chou, C. T., "Stochastic theory of continuous-time state-space identification," *IEEE Transactions on Signal Processing*, Vol. 47, No. 1, 1999, pp. 41–51.

[27] Fritz, M., Doll, J., Ward, K. C., Mendeck, G., Sostaric, R. R., Pedrotty, S., Kuhl, C., Acikmese, B., Bieniawski, S. R., Strohl, L., et al., "Post-flight performance analysis of navigation and advanced guidance algorithms on a terrestrial suborbital rocket flight," *AIAA SCITECH 2022 Forum*, 2022, p. 0765.

[28] Woffinden, D., Robinson, S., Williams, J., and Putnam, Z. R., "Linear covariance analysis techniques to generate navigation and sensor requirements for the safe and precise landing integrated capabilities evolution (SPLICE) project," *AIAA SciTech 2019 forum*, 2019, p. 0662.

[29] Williams, J., Woffinden, D., and Putnam, Z. R., "Mars Entry Guidance and Navigation Analysis Using Linear Covariance Techniques for the Safe and Precise Landing–Integrated Capabilities Evolution (SPLICE) Project," *AIAA Scitech 2020 Forum*, 2020, p. 0597.

[30] Goebel, R., Sanfelice, R. G., and Teel, A. R., "Hybrid dynamical systems," *IEEE control systems magazine*, Vol. 29, No. 2, 2009, pp. 28–93.

[31] Piccoli, B., "Hybrid systems and optimal control," *Proceedings of the 37th IEEE Conference on Decision and Control (Cat. No. 98CH36171)*, Vol. 1, IEEE, 1998, pp. 13–18.

[32] Tanaka, K., and Ishigami, G., "Modeling of lidar measurement uncertainty for rover path planning," *Proc. of The International Symposium on Artificial Intelligence, Robotics and Automation in Space*, 2016.

[33] Wang, K., Wang, Y., Liu, B., and Chen, J., "Quantification of uncertainty and its applications to complex domain for autonomous vehicles perception system," *IEEE Transactions on Instrumentation and Measurement*, Vol. 72, 2023, pp. 1–17.

[34] Kyurkchiev, N., and Markov, S., "Sigmoid functions: some approximation and modelling aspects," *LAP LAMBERT Academic Publishing, Saarbrucken*, Vol. 4, 2015, p. 34.

[35] Buckner, S. C., Shaffer, J., Carson, J. M., Johnson, B. J., Sostaric, R. R., and Acikmese, B., "Constrained Visibility Guidance for 6-DOF Powered Descent Maneuvers with Terrain Scanning using Sequential Convex Programming," *AIAA SCITECH 2024 Forum*, 2024, p. 1759.

[36] Malyuta, D., and Açıkmeşe, B., "Fast homotopy for spacecraft rendezvous trajectory optimization with discrete logic," *Journal of Guidance, Control, and Dynamics*, Vol. 46, No. 7, 2023, pp. 1262–1279.

[37] 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.

[38] "High-Resolution LOLA Topography for Lunar South Pole Sites," https://pgda.gsfc.nasa.gov/products/78, ???? Accessed: 2025-05-26.

[39] Kim, S., and Kim, M., "Rotation representations and their conversions," *Ieee Access*, Vol. 11, 2023, pp. 6682–6699.

[40] Thrun, S., Koller, D., Ghahramani, Z., Durrant-Whyte, H., and Ng, A. Y., "Simultaneous mapping and localization with sparse extended information filters: Theory and initial results," *Algorithmic Foundations of Robotics V*, Springer, 2004, pp. 363–380.