Convex Block-Cholesky Approach to Risk-constrained Low-thrust Trajectory Design under Operational Uncertainty[*]

Kenshiro Oguri [†]
Purdue University, West Lafayette, Indiana, 47907

Gregory Lantoine [‡]
Jet Propulsion Laboratory, California Institute of Technology, Pasadena, California 91109

Designing robust trajectories under uncertainties is an emerging technology that may represent a key paradigm shift in space mission design. As we pursue more ambitious scientific goals (e.g., multi-moon tours, missions with extensive components of autonomy), it becomes more crucial that missions are designed with navigation (Nav) and flight-path control (FPC) processes in mind. This mission design paradigm calls for techniques that appropriately quantify statistical effects of Nav and FPC, evaluate associated risks, and design missions that ensure sufficiently low risk while minimizing a statistical performance metric; one example would be to minimize statistical Delta-V to a 99% confidence level (DV99). In response to the need, this paper develops an algorithm for risk-constrained trajectory optimization under operational uncertainties due to initial state dispersion, navigation error, maneuver execution error, and imperfect dynamics modeling. We formulate it as a nonlinear stochastic optimal control problem and develop a computationally tractable algorithm that combines optimal covariance steering and sequential convex programming (SCP). Specifically, the proposed algorithm takes a so-called block-Cholesky approach for convex formulation of optimal covariance steering, and leverages a recent SCP algorithm `SCvx*` for reliable numerical convergence. We apply the developed algorithm to risk-constrained, statistical trajectory optimization for exploration of dwarf planet Ceres with a Mars gravity assist, and demonstrate the robustness of the statistically-optimal trajectory and FPC policies via nonlinear Monte Carlo simulation.

I. Introduction

Designing robust trajectories under uncertainties is an emerging technology that may represent a key paradigm shift in space mission design. As the complexity of space missions grows to pursue more ambitious scientific goals (e.g., multi-moon tours, missions with extensive components of autonomy), it becomes increasingly crucial that missions are

---

designed with navigation (Nav) and flight-path control (FPC) processes in mind. These processes (Nav and FPC) are inherently statistical, as a result of statistical orbit determination (OD) and trajectory correction maneuvers (TCMs). Any risks (e.g., collision with a planetary body during a close flyby) under these statistical effects must be appropriately quantified and mitigated in mission design processes to ensure the mission safety. Although these risks have been historically mitigated in a rather heuristic manner (e.g., heuristic margins to constraints, manually-tuned forced coasting arcs) [1, 2], there has been a growing number of studies on directly incorporating these statistical effects into mission design and thereby coupling the trajectory optimization with navigation and FPC processes within the design process.

Similar to conventional deterministic problems, existing approaches to statistical trajectory optimization under uncertainties can be categorized into: indirect methods [3–5], direct methods [6–11] and differential dynamic programming (DDP) [12, 13]. This list of papers are certainly not exhaustive, but it already highlights the growing interest on this topic in the space mission design community. See [11] and its earlier version [14] for further discussion that compares these different methods. Key enablers of these techniques lie in the recent advances in stochastic optimal control and optimization communities, particularly in *optimal covariance steering* [15–21] and *computational control* based on convex programming [22–26], and their application to aerospace problems [27–32]. Unlike traditional optimal control, optimal covariance steering consider controlling the mean and covariance of the system's state while minimizing a statistical cost, which depends on both the state mean and covariance; it often also involves statistical constraints known as *chance constraints* (or equivalently, *risk constraints*), which imposes constraints on probability that constraints be satisfied under state uncertainties. Either explicitly or implicitly, many existing techniques for trajectory optimization under uncertainties build on the concept of optimal covariance steering, solved by indirect methods, direct methods, or DDP.

A particularly important aspect to consider when developing an algorithm for trajectory optimization under uncertainty is, *what specific approach we choose to formulate the optimal covariance steering problem*. First introduced in 1980s by Hotz and Skelton [33], the theory of optimal covariance steering has seen rapid advancement in the stochastic optimal control community in recent years, including continuous-time covariance steering [16] and discrete-time covariance steering with chance constraints [15]. Discrete-time covariance steering may find more relevant applications in aerospace, considering the nature of discrete opportunities for control commands and measurements. It is therefore beneficial to further discuss some variants of discrete-time covariance steering, such as one with input constraints [34], one without the need for history feedback [17], one with output feedback [18], and output-feedback covariance steering without history feedback [19]. In particular, this *output-feedback* covariance steering holds the key to unlocking the potential of incorporating navigation process (i.e., taking measurements and applying filtering to obtain orbital state estimates) into the framework of trajectory optimization under uncertainties. Notably, the discrete-time covariance steering studies mentioned in this paragraph formulate the problem in convex form by forming a large block-matrix for the system equation and utilizing the Cholesky factor of covariance matrices. This class of approaches are called

block-Cholesky formulation in this paper.

An important caveat of the *block-Cholesky formulation* lies in its computational complexity. As discussed in [8, 11, 21], their computational complexity is roughly proportional to $(n_x N^2)$, where n_x is the size of the state vector and N is the number of discretized trajectory segments. Hence, recent studies have developed a more computationally-efficient formulation of discrete-time covariance steering [20] and its output-feedback variant [21], which have been successfully applied to robust space trajectory optimization under uncertainties [9, 11, 14]. These approaches propagate the full covariance (rather than its Cholesky factor) and do not rely on the block-matrix formulation, and hence called *full-covariance formulation* in this paper. The main innovation of these studies lies in its elegant lossless convexification of originally non-convex covariance propagation equations by using the Schur complement and the Karush-Kuhn-Tucker (KKT) conditions. As a result, the *full-covariance formulation* enables us to formulate the discrete-time covariance steering problem in convex form, with its computational complexity roughly proportional to $N(n_x + n_u)$, where n_u is the size of the control vector. This implies that the benefit of using full-covariance formulations is greater when considering larger trajectory optimization problem (i.e., greater N).

Although full-covariance approaches have a critical advantage over block-Cholesky approaches in terms of the computational complexity, they also have key drawbacks that warrant careful considerations for space trajectory optimization. The drawbacks are: (1) numerical ill-conditioning, (2) non-convex chance constraints, and (3) inflexible cost functions. The first point, numerical ill-conditioning, is related to the large scale differences that are inherent to space trajectories, especially in interplanetary trajectories. To illustrate this aspect, let us consider spacecraft on a trajectory from Mars to a main-belt asteroid with state uncertainty of 10 km in position and 1 m/s in velocity (standard deviation). While the magnitudes of these position/velocity uncertainty variances are of $\{100 \text{ km}^2, 1 \times 10^{-6} \text{ km}^2/\text{s}^2\}$, the magnitude of position vector measured from Sun is in the order of $\sim 2-4 \times 10^8$ km; this implies that the numerical solver would need to deal with a problem of condition number as large as $\sim 1 \times 10^{14}$ and may not be as reliable as desired*. In fact, this type of numerical ill-conditioning is the main motivation for square-root formulations of OD filters (e.g., square-root unscented Kalman filter [35] and square-root information filter [36]), which can recover up to a half of the digits of precision that would have been lost otherwise.

The second drawback, non-convex chance constraints, is due to the use of full covariance matrices as decision variables. As demonstrated in existing studies on chance-constrained control [15, 17–19, 30, 31], we can formulate many chance constraints in convex form *if* the Cholesky factor of covariance matrices are decision variables. On the other hand, full-covariance approaches directly handle covariance matrices, which is a nonlinear function of the Cholesky factor, hence making those chance constraints non-convex, including control magnitude chance constraints and half-plane state chance constraints, to name a few. This fact requires us to approximate chance constraints at each

---

*If a canonical unit such that makes both 1 AU and 1 μ_{sun} equal to 1 is used to normalize the problem, the condition number might become even worse. In the case of the example above, the variances of position and velocity become approximately $\{4.4 \times 10^{-15}, 1.1 \times 10^{-17}\}$ against 2-3 AU.

iteration via either linearization [9, 11, 14] or convex-concave procedure [21].

Lastly, the issue of inflexible cost functions arises from the lossless convexification process in the formulation of full-covariance approaches. The lossless property, which is crucial for convexifying the full covariance propagation equation, relies on the KKT conditions, in particular their complementary slackness condition. To meet the KKT conditions, [20, 21] assume a specific form of objective function, and it is shown [37] that their full-covariance approaches cannot consider the Delta-V99 (DV99) cost metric (statistical Delta-V to 99% confidence level), which is a typical cost in space mission design. [11] (preprint version [38]) proposes a middle-ground solution that relaxes the inflexibility and enables approximate formulation of the DV99 cost.

Given these observations, this paper develops an approach based on block-Cholesky formulation for robust trajectory optimization under uncertainties.[†] Main contributions of the present paper are threefold. First, this paper leverages a recent formulation of block-Cholesky output-feedback covariance steering [30] and a sequential convex programming (SCP) algorithm SCvx* [26] to develop an optimization algorithm for risk-constrained low-thrust trajectory design under uncertainties. The algorithm incorporates various operational uncertainties due to navigation errors, maneuver execution errors, initial state dispersion, and imperfect dynamics modeling. Second, we extend the developed algorithm to incorporate planetary gravity assists (GAs), which is a must-have option in any modern interplanetary mission design. In particular, we utilize the patched-conic GA model in this study. Third, the developed algorithm introduces some improvements in its mathematical formulation, including a more general class of augmented Lagrangian functions than introduced in the original SCvx* paper [26], which can smoothly approximate the l_1-penalty function for better convergence; and a formulation of three-dimensional control magnitude chance constraints that provide tighter approximation than existing studies [7, 9, 13, 39] and are rigorously conservative compared to a heuristic approach taken in [12].

Fig. 1 presents the fundamental idea behind the proposed formulation for risk-constrained trajectory optimization under uncertainties. The formulation directly incorporates operational uncertainty models (e.g., measurement function, measurement noise, OD frequency) as well as statistical risk constraints (e.g., safety with 99.9% confidence) while minimizing the typical statistical cost, DV99. Considering other statistical costs is straightforward due to the flexibility of block-Cholesky approaches as opposed to full-covariance approaches. The developed algorithm then solves for statistically optimal low-thrust trajectories under the modeled uncertainties by utilizing the mean and covariance information derived from the Kalman filter process and optimizing the reference trajectory and FPC policies simultaneously via SCP.

The rest of the paper is structured as follows. After Section II introduces the system equation, Section III details the stochastic optimal control formulation that incorporates operational uncertainties and statistical risk constraints.

[†]A recent paper of the first author of this paper develops an algorithm based on a full-covariance approach [11] (preprint version [38]), if the readers are interested in the other formulation too.
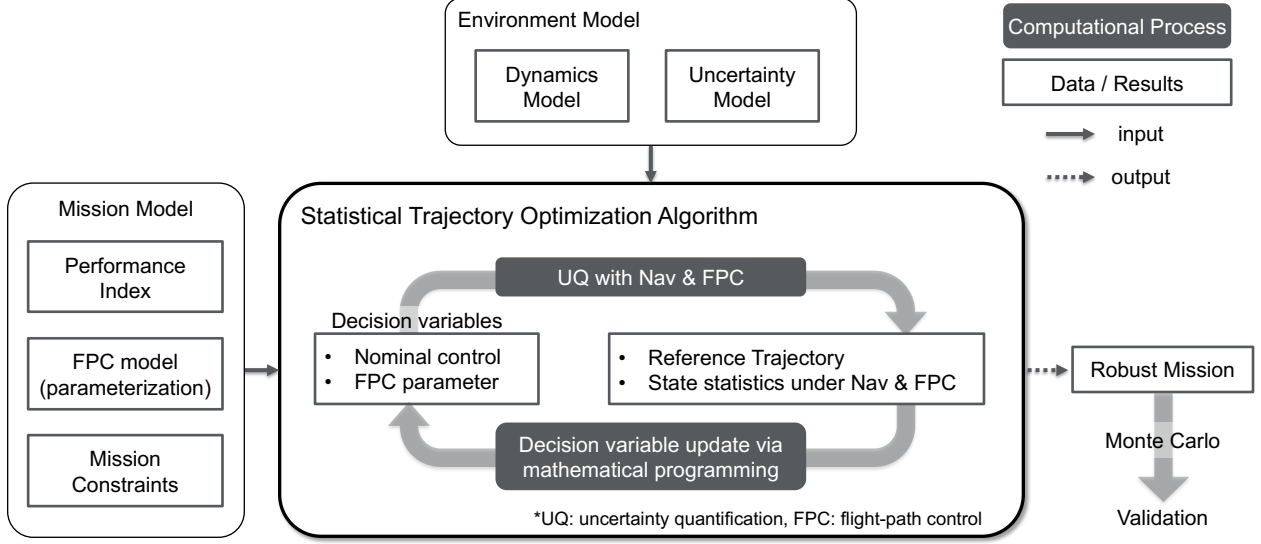
Figure 1   Fundamental idea of statistical trajectory optimization under uncertainty

Section IV presents a tractable convex approximation of the stochastic optimal control problem based on a block-Cholesky approach, which optimizes the reference trajectory and FPC policies simultaneously under the Kalman filter process and linear FPC policies. Section V develops a SCP algorithm that leverages the algorithmic framework of SCvx* and iteratively solves the convex subproblem to march toward an optimal solution. Section VI applies the developed algorithm to robust trajectory optimization for exploration of dwarf planet Ceres with a Mars gravity assist, and demonstrates the robustness of the obtained statistically-optimal trajectory and associated FPC policies via nonlinear Monte Carlo simulation.

II. Equations of Motion

This paper considers low-thrust spacecraft orbital motions with a simplified low-thrust model that ignores the mass decrease due to thrusting. This assumption is in line with most of the existing approaches to robust space trajectory optimization under uncertainty, including [4, 7, 9, 12]. While some studies, such as [8, 10, 40], consider the mass flow equation and associated uncertainty, they also point out that the resulting distribution of the mass uncertainty becomes highly-non-Gaussian and in fact take the form of generalized chi-squared distribution. It is well-known that the generalized chi-squared distribution does not have closed-form expression for the probability density function, making rigorous characterization of mass uncertainty less straightforward. Developing an approach to properly incorporating the effect of mass uncertainty is an important direction of future work.

Thus, letting $x \in \mathbb{R}^{n_x}$ denote the orbital state (e.g., position/velocity, osculating orbital elements, etc) and $u \in \mathbb{R}^3$ be the control acceleration due to the low-thrust engine, we can express the equations of motion in a generic form as

follows:

$$\dot{x} = f(x, u, t) = f_0(x, t) + F(x)u \tag{1}$$

where $f_0(\cdot, \cdot) : \mathbb{R}^{n_x} \times \mathbb{R} \mapsto \mathbb{R}^{n_x}$ represents natural orbital dynamics; $F(\cdot) : \mathbb{R}^{n_x} \mapsto \mathbb{R}^{n_x \times 3}$ returns a matrix that maps the acceleration to the rate of the state change. Eq. (1) encompasses equations of motion describing low-thrust orbital dynamics in various coordinate systems.

Specifically, this study uses the Cartesian coordinate system to express the system state and considers perturbed two-body dynamics. Then, the orbital state and the dynamics functions are given by:

$$x = \begin{bmatrix} r \\ v \end{bmatrix}, \quad f_0(x, t) = \begin{bmatrix} v \\ -\frac{\mu}{\|r\|_2^3} r + a_{\text{pert}}(x, t) \end{bmatrix}, \quad F = \begin{bmatrix} 0_{3\times3} \\ I_3 \end{bmatrix} \tag{2}$$

where $r \in \mathbb{R}^3$ and $v \in \mathbb{R}^3$ are the spacecraft position and velocity in Cartesian coordinates, respectively; μ is the gravitational parameter of the central body; $a_{\text{pert}}(x, t)$ is the perturbing acceleration.

III. Risk-Constrained Nonlinear Stochastic Optimal Control Problem

Let us formulate the problem of risk-constrained low-thrust trajectory optimization under uncertainty in a form of chance-constrained optimal control. Our formulation incorporates the influence of major uncertain errors in space missions (initial state dispersion, maneuver execution error, and OD error) as well as the corrective effect of FPC efforts (i.e., performing trajectory corrections based on OD results).

A. Nonlinear Stochastic System

1. Initial State Dispersion

Any space trajectories begin with initial state dispersion, such as due to launch dispersion and orbit insertion errors. We model such initial state dispersion via a Gaussian distribution as:

$$x_0 \sim \mathcal{N}(\bar{x}_0, P_0) \tag{3}$$

2. Maneuver Execution Error

Any maneuvers involve some errors that are uncertain by nature, due to thruster misalignment, over/under burn, among other causes. A common approach to modeling such errors is the Gates model [41], which captures the error that is proportional to the applied thrust commands as well as that is fixed irrespective of the commands. Using the

Gates model, for a given commanded control $\boldsymbol{u}$, a maneuver execution error, denoted by $\widetilde{\boldsymbol{u}}$, is given as [29]:

$$\widetilde{\boldsymbol{u}} = G_{\text{exe}}(\boldsymbol{u})\boldsymbol{w}_{\text{exe}}, \quad G_{\text{exe}}(\boldsymbol{u}) = T(\boldsymbol{u})P_{\text{gates}}^{1/2}(\boldsymbol{u}), \quad \boldsymbol{w}_{\text{exe}} \sim \mathcal{N}(0, I_3), \tag{4}$$

where $\boldsymbol{w}_{\text{exe}}$ are independent and identically distributed (i.i.d.) standard Gaussian vectors, i.e., $\mathbb{E}\left[\boldsymbol{w}_{\text{exe}}\right] = 0$ and $\mathbb{E}\left[\boldsymbol{w}_{\text{exe}}(t_i)\boldsymbol{w}_{\text{exe}}(t_j)^{\top}\right] = \delta_{i,j}I$, where $\delta_{i,j}$ is the Kronecker delta. $P_{\text{gates}}(\cdot)$ denotes the execution error covariance in the local coordinates that are defined with respect to the thrust direction while $T(\cdot)$ rotates the direction to the coordinates we propagate our system, given by

$$P_{\text{gates}}(\cdot) = \text{diag}(\sigma_p^2, \sigma_p^2, \sigma_m^2), \quad T(\cdot) = \begin{bmatrix} \hat{\boldsymbol{S}} & \hat{\boldsymbol{E}} & \hat{\boldsymbol{Z}} \end{bmatrix}, \quad \hat{\boldsymbol{Z}} = \frac{\boldsymbol{u}}{\|\boldsymbol{u}\|_2}, \quad \hat{\boldsymbol{E}} = \frac{[0, 0, 1]^{\top} \times \hat{\boldsymbol{Z}}}{\left\|[0, 0, 1]^{\top} \times \hat{\boldsymbol{Z}}\right\|_2}, \quad \hat{\boldsymbol{S}} = \hat{\boldsymbol{E}} \times \hat{\boldsymbol{Z}} \tag{5}$$

Here, $\sigma_p^2 = \sigma_3^2 + \sigma_4^2 \|\boldsymbol{u}\|_2^2$ and $\sigma_m^2 = \sigma_1^2 + \sigma_2^2 \|\boldsymbol{u}\|_2^2$, where $\{\sigma_1, \sigma_2\}$ are {fixed, proportional} magnitude errors while $\{\sigma_3, \sigma_4\}$ are {fixed, proportional} pointing errors. It is clear that the terms $\sigma_2^2 \|\boldsymbol{u}\|_2^2$ and $\sigma_4^2 \|\boldsymbol{u}\|_2^2$ take greater values for larger control command, and hence represent proportional errors.

3. Stochastic Differential Equations

To incorporate uncertain errors and FPCs in our mission design process, one of the most straightforward approaches is to express our system via nonlinear stochastic differential equations (SDEs). Based on the equations of motion Eq. (1), our stochastic system can be formally expressed via a set of nonlinear SDEs as:

$$d\boldsymbol{x} = [\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t) + F(\boldsymbol{x})\widetilde{\boldsymbol{u}}]dt + G(\boldsymbol{x})d\boldsymbol{w}(t), \tag{6}$$

where $G(\cdot) : \mathbb{R}^{n_x} \mapsto \mathbb{R}^{n_x \times n_w}$ represents the intensity of disturbances, mapping $d\boldsymbol{w}(t)$ to the random variation of the state; $d\boldsymbol{w}(t) : \mathbb{R} \mapsto \mathbb{R}^{n_w}$ is a standard Brownian motion vector, i.e., $\mathbb{E}[d\boldsymbol{w}] = 0$ and $\mathbb{E}[d\boldsymbol{w}(t)d\boldsymbol{w}^{\top}(t)] = Idt$. The form $G(\cdot)d\boldsymbol{w}$ encompasses various types of disturbances, including unmodeled external perturbations (e.g., solar radiation pressure, momentum de-saturation operations, etc). The term $F(\boldsymbol{x})\widetilde{\boldsymbol{u}}$ captures the effect of maneuver execution errors to the variation of state uncertainty.

B. Nonlinear Stochastic System with Navigation Uncertainty and Flight-Path Control

Let us then model the navigation process and FPCs. We model the FPC policy in a way that calculates trajectory corrections based on imperfect state knowledge given by orbit determination (OD) processes.

1. Navigation Model

We model the OD process as a sequence of filtering processes with discrete-time observations. Assuming that a set of spacecraft tracking data becomes available at time t_k, the observation process is given by:

$$\boldsymbol{y}_k = \boldsymbol{f}_{\text{obs}}(\boldsymbol{x}_k, t_k) + G_{\text{obs}}(\boldsymbol{x}_k)\boldsymbol{w}_{\text{obs},k}, \quad k = 0, 1, ..., N \tag{7}$$

where $\boldsymbol{y}_k \in \mathbb{R}^{n_y}$ are the (collection of) measurements; $\boldsymbol{f}_{\text{obs}}(\cdot, \cdot) : \mathbb{R}^{n_x} \times \mathbb{R} \mapsto \mathbb{R}^{n_y}$ and $G_{\text{obs}}(\cdot) : \mathbb{R}^{n_x} \mapsto \mathbb{R}^{n_y \times n_y}$ model the observation process. $\boldsymbol{w}_{\text{obs},k} \in \mathbb{R}^{n_y}$ is measurement noise, modeled as a sequence of i.i.d. standard Gaussian random vectors.

At time t_k, the navigation solution, $\hat{\boldsymbol{x}}_k$, is obtained by a filtering process with the past observations $\boldsymbol{y}_i (i = 0, 1, ..., k)$ as follows:

$$\hat{\boldsymbol{x}}_k = \mathcal{F}_k(\hat{\boldsymbol{x}}_0^-, \boldsymbol{y}_i : i = 0, 1, ..., k), \tag{8}$$

where the "−" superscript indicates a quantity right before the measurement update; $\hat{\boldsymbol{x}}_0^-$ denotes the initial state estimate. Filtering typically utilizes the innovation process $\widetilde{\boldsymbol{y}}_k^-$, defined as:

$$\widetilde{\boldsymbol{y}}_k^- = \boldsymbol{y}_k - \boldsymbol{f}_{\text{obs}}(\hat{\boldsymbol{x}}_k^-). \tag{9}$$

2. Flight-path Control Model

Flight-path controls are a collection of efforts that drive the orbital state deviation back to the nominal trajectory, planed based on imperfect navigation solutions. Hence, this study models FPCs over the course of a trajectory by a sequence of feedback policies that compute necessary control modifications based on the current state estimate given by Eq. (8). The feedback policies are thus expressed as:

$$\boldsymbol{u}_k = \pi_k(\hat{\boldsymbol{x}}_k, \Omega_k), \tag{10}$$

where Ω_k denotes a set of parameters used to compute the corrective controls $\boldsymbol{u}_k$ according to the policy π_k.

C. Original Problem

1. Statistical Cost Metric

A common cost function in space trajectory optimization is the total control effort. However, we may not directly minimize the control effort in stochastic setting because it is not well-defined due to the stochasticity of $\boldsymbol{u}$ under the

feedback policy Eq. (10). Thus, this study considers minimizing the p-quantile of the closed-loop control effort, i.e.,

$$J = \int_{t_0}^{t_f} Q_{X \sim \|\boldsymbol{u}\|_2}(p) \tag{11}$$

where $Q_X(p)$ is the quantile function of a random variable X evaluated at probability p, formally defined as:

$$Q_X(p) = \min\{x \in \mathbb{R} \mid \mathbb{P}[X \le x] \ge p\}. \tag{12}$$

Note that using $p = 0.99$ corresponds to minimizing 99%-quantile of control cost, i.e., DV99.

2. Risk Constraints

The constraints considered in this study include path constraints and terminal constraints. As our state and control are subject to uncertainty, those constraints are not deterministic anymore and need to be treated stochastically.

In particular, we consider a class of probabilistic constraints called *chance constraints* to formulate our path constraints, as they naturally represent stochastic analogs of deterministic constraints classically considered in mission design. Intuitively, a chance constraint imposes a bound on the probability that a particular constraint be satisfied (i.e., no collision with an obstacle with a probability greater than 99.9%); see, for instance, [5, 14, 29, 42, 43] and references therein, for the use of chance constraints in the context of space trajectory optimization.

Mathematically, path chance constraints imposed on discrete-time state and control are expressed as:

$$\mathbb{P}\left[c_j(\boldsymbol{x}, \boldsymbol{u}, t) \le 0\right] \ge 1 - \varepsilon_j, \tag{13}$$

where $\mathbb{P}[\cdot]$ denotes the probability operator, $c_j(\cdot) : \mathbb{R}^{n_x} \times \mathbb{R}^3 \times \mathbb{R} \mapsto \mathbb{R}$ represents the j-th nonlinear constraint function, and ε_j is a risk bound associated with the j-th constraint (e.g., $\varepsilon_j = 0.001$ for 99.9% confidence). This imposes the constraint $c_j \le 0$ to be satisfied with probability greater than or equal to $1 - \varepsilon_j$.

Similarly, we model the terminal constraints as distributional constraints that ensure that the final state is within a prescribed region. Specifically, we impose terminal constraints on the final state, $\boldsymbol{x}_N$, to achieve the target $\boldsymbol{x}_f$ on average with prescribed accuracy represented by the final covariance P_f:

$$\mathbb{E}[\boldsymbol{x}_N] - \boldsymbol{x}_f = 0, \tag{14a}$$

$$\mathrm{Cov}[\boldsymbol{x}_N] \preceq P_f \tag{14b}$$

where $\mathbb{E}[\cdot]$ and $\mathrm{Cov}[\cdot]$ denotes the expectation and covariance operators, respectively. Mathematically, $\mathbb{E}[\boldsymbol{x}] = \int \boldsymbol{x} p(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}$ where $p(\boldsymbol{x})$ is a probability density function (pdf) of $\boldsymbol{x}$, and $\mathrm{Cov}[\boldsymbol{x}] = \mathbb{E}[(\boldsymbol{x} - \mathbb{E}[\boldsymbol{x}])(\boldsymbol{x} - \mathbb{E}[\boldsymbol{x}])^\top]$.

Thus, letting $\boldsymbol{\Theta}$ represent a vector of relevant parameters to optimize (which take different forms depending on specific problems), our original problem can be expressed as in Problem 1.

Problem 1 (Original problem). *Find $\boldsymbol{x}^*(t)$ $\pi_k, \forall k$, and $\boldsymbol{\Theta}$ that minimize the cost Eq. (11) subject to the dynamical constraint Eq. (6), path chance constraints Eq. (13), and terminal constraint Eq. (14) under a sequence of FPCs determined by Eq. (10) with the state estimates Eq. (8) based on observations Eq. (7).*

IV. Tractable Formulation via block-Cholesky Covariance Steering

In general, Problem 1 is intractable to solve. A major bottleneck of the difficulty lies in the process of *uncertainty quantification* (UQ) to evaluate the statistics of the state, cost, and constraints under uncertainty, which is computationally expensive in general, except for some special cases. In nonlinear trajectory optimization, it is most likely the case that we must iteratively perform such expensive UQ processes to evaluate the functions and their gradients with respect to decision variables. This requires an efficient and reliable solution method.

Among multiple possible approaches, this study proposes a solution method based on optimal covariance steering and sequential convex programming (SCP) to solve our problem. The proposed approach approximately formulates the originally intractable stochastic problem in a convex, deterministic form at each successive iteration; in particular, we employ a block-Cholesky formulation as discussed in Section I. At each iteration, the original problem is approximated as a convex subproblem, which is solved with guaranteed convergence to the global optimum of the approximated problem due to the convex property. Once the convergence of a convex subproblem is achieved, then the subproblem solution is used as a new reference trajectory for the next iteration. Leveraging the reliability and efficiency of convex programming, this solution method provides us with a vehicle for overcoming the major bottleneck of numerical solutions to nonlinear stochastic optimal control problems, at the cost of approximate dynamical evolution of the stochastic state deviations from the reference trajectory. Note that the dynamical feasibility of the reference trajectory is not compromised in our solution method, and the designed reference trajectory satisfies the nonlinear dynamics up to the user-specified tolerance.

Let us first approximately reformulate Problem 1 in a deterministic, convex form in the following.

A. Linear State Statistics Dynamics

To obtain the convex subproblem, Eq. (6) is first linearized and discretized assuming fixed-time problems (i.e., t_0 and t_f not part of optimization variables). It is possible to extend the formulation presented in this article to variable-time problems by incorporating the adaptive-mesh mechanism [44]. Such an extension is left as exciting future work, as it would allow the FPC policies (which are being optimized) to adjust maneuver epochs based on navigation results.

1. Linear, Discrete-time System

The initial *a priori* state estimate $\hat{\boldsymbol{x}}_0^-$ and its error $\widetilde{\boldsymbol{x}}_0^-$ (e.g., state dispersion estimate immediately after deployment from a launcher) are assumed to be Gaussian-distributed, i.e., $\hat{\boldsymbol{x}}_0^- \sim \mathcal{N}(\bar{\boldsymbol{x}}_0, \hat{P}_0^-)$ and $\widetilde{\boldsymbol{x}}_0^- \sim \mathcal{N}(0, \widetilde{P}_0^-)$, where $\overline{(\cdot)}$ indicates the mean of a random variable. It implies that the initial state $\boldsymbol{x}_0^-$ is distributed as $\boldsymbol{x}_0 = \hat{\boldsymbol{x}}_0^- + \widetilde{\boldsymbol{x}}_0^- \sim \mathcal{N}(\bar{\boldsymbol{x}}_0, \hat{P}_0^- + \widetilde{P}_0^-)$. Linearizing the system Eq. (6) and discretizing the time into intervals $t_0 < t_1 < t_2 < ... < t_N = t_f$ with zeroth-order-hold (ZoH) control, i.e., $\boldsymbol{u}(t) = \boldsymbol{u}_k$, $\forall t \in [t_k, t_{k+1})$, yields [30, 45]:

$$\boldsymbol{x}_{k+1} = A_k \boldsymbol{x}_k + B_k \boldsymbol{u}_k + \boldsymbol{c}_k + G_{\text{exe},k} \boldsymbol{w}_{\text{exe},k} + G_k \boldsymbol{w}_k, \quad k = 0, 1, ..., N-1 \tag{15}$$

where $A_k, B_k, \boldsymbol{c}_k$, and G_k represent linear system matrices that are obtained by linearizing about the reference state $\boldsymbol{x}^{\text{ref}}(t)$ and control $\boldsymbol{u}^{\text{ref}}(t)$ and discretizing the system in time (see Ref. [30] for the definition); $\boldsymbol{w}_k$ is an i.i.d. standard Gaussian random vector.

2. Filtered State Dynamics

Since the state is initially Gaussian-distributed and obeys the linear dynamics in a convex subproblem, the optimal, unbiased state estimate can be obtained by the Kalman filter. Thus, the filtering process Eq. (8) is given as:

$$\begin{cases} \hat{\boldsymbol{x}}_k^- = A_{k-1}\hat{\boldsymbol{x}}_{k-1} + B_{k-1}\boldsymbol{u}_{k-1} + \boldsymbol{c}_{k-1} & \text{(time update)} \\ \widetilde{P}_k^- = A_{k-1}\widetilde{P}_{k-1}A_{k-1}^\top + G_{\text{exe},k-1}G_{\text{exe},k-1}^\top + G_{k-1}G_{k-1}^\top \\ \hat{\boldsymbol{x}}_k = \hat{\boldsymbol{x}}_k^- + L_k(\boldsymbol{y}_k - C_k\hat{\boldsymbol{x}}_k^-) & \text{(measurement update)} \\ \widetilde{P}_k = (I - L_k C_k)\widetilde{P}_k^-(I - L_k C_k)^\top + L_k D_k D_k^\top L_k^\top, \end{cases} \tag{16}$$

where $\widetilde{P}_k$ denotes the estimate error covariance; L_k is the Kalman gain; C_k and D_k are the linearized measurement equation about the reference, given by

$$\widetilde{P}_k \triangleq \text{Cov}\left[\widetilde{\boldsymbol{x}}_k\right], \quad L_k \triangleq \widetilde{P}_k^- C_k^\top (C_k \widetilde{P}_k^- C_k^\top + D_k D_k^\top)^{-1}, \quad C_k \triangleq \frac{\partial \boldsymbol{f}_{\text{obs}}}{\partial \boldsymbol{x}}(\boldsymbol{x}_k^{\text{ref}}), \quad D_k = G_{\text{obs}}(\boldsymbol{x}_k^{\text{ref}}) \tag{17}$$

As classically known in estimation theory, $\widetilde{P}_k$ and L_k can be computed *a priori* for linear systems. Combining the two equations in Eq. (16), the filtered state process under stochastic error and Kalman filtering is written as:

$$\hat{\boldsymbol{x}}_{k+1} = A_k \hat{\boldsymbol{x}}_k + B_k \boldsymbol{u}_k + \boldsymbol{c}_k + L_{k+1}\widetilde{\boldsymbol{y}}_{k+1}^- \tag{18}$$

where $\widetilde{\boldsymbol{y}}_k^-$ is the innovation process Eq. (9), whose linear form and its covariance, denoted by $P_{\widetilde{\boldsymbol{y}}_k^-}$, are given as [30]

$$\widetilde{\boldsymbol{y}}_k^- = C_k \widetilde{\boldsymbol{x}}_k + D_k \boldsymbol{w}_{\text{obs},k}, \quad P_{\widetilde{\boldsymbol{y}}_k^-} \triangleq \text{Cov}\left[\widetilde{\boldsymbol{y}}_k^-\right] = C_k \widetilde{P}_k^- C_k^\top + D_k D_k^\top. \tag{19}$$

3. Linear Flight-path Control Model

We consider a linear FPC policy to model Eq. (10) for convex formulation. In particular, we use the following form of the control policy:

$$\boldsymbol{u}_k = \bar{\boldsymbol{u}}_k + K_k z_k, \tag{20}$$

where $\bar{\boldsymbol{u}}_k$ is a nominal control input, K_k is a feedback gain matrix, and z_k is a stochastic process given by

$$z_{k+1} = A_k z_k + L_{k+1} \widetilde{\boldsymbol{y}}_{k+1}^-, \quad z_0 = \hat{\boldsymbol{x}}_0 - \bar{\boldsymbol{x}}_0. \tag{21}$$

4. State and Control Statistics

We are now ready to analytically express the statistics of the state and control—in particular, their first and second statistical moments, i.e., mean and covariance. As derived in Ref. [30, 46], it is convenient to express Eq. (18) in a block-matrix form as:

$$
\begin{bmatrix} \hat{\boldsymbol{x}}_0 \\ \hat{\boldsymbol{x}}_1 \\ \hat{\boldsymbol{x}}_2 \\ \vdots \end{bmatrix} = \begin{bmatrix} I_{n_x} \\ A_0 \\ A_1 A_0 \\ \vdots \end{bmatrix} \hat{\boldsymbol{x}}_0^- + \begin{bmatrix} 0 & 0 & \\ B_0 & 0 & \\ A_1 B_0 & B_1 & \\ & & \ddots \end{bmatrix} \begin{bmatrix} \boldsymbol{u}_0 \\ \boldsymbol{u}_1 \\ \boldsymbol{u}_2 \\ \vdots \end{bmatrix} + \begin{bmatrix} 0 & 0 & \\ I_{n_x} & 0 & \\ A_1 & I_{n_x} & \\ & & \ddots \end{bmatrix} \begin{bmatrix} \boldsymbol{c}_0 \\ \boldsymbol{c}_1 \\ \boldsymbol{c}_2 \\ \vdots \end{bmatrix} + \begin{bmatrix} L_0 & 0 & \\ A_0 L_0 & L_1 & \\ A_1 A_0 L_0 & A_1 L_1 & \\ & & \ddots \end{bmatrix} \begin{bmatrix} \widetilde{\boldsymbol{y}}_0^- \\ \widetilde{\boldsymbol{y}}_1^- \\ \widetilde{\boldsymbol{y}}_2^- \\ \vdots \end{bmatrix} \tag{22}
$$

which can be expressed in a compact form as:

$$\hat{X} = \mathbf{A}\hat{\boldsymbol{x}}_0^- + \mathbf{B}U + \mathbf{C} + \mathbf{L}Y, \tag{23}$$

where $\hat{X} = [\hat{\boldsymbol{x}}_0^\top, \hat{\boldsymbol{x}}_1^\top, ..., \hat{\boldsymbol{x}}_N^\top]^\top$, $U = [\boldsymbol{u}_0^\top, \boldsymbol{u}_1^\top, ..., \boldsymbol{u}_{N-1}^\top]^\top$, $Y = [\widetilde{\boldsymbol{y}}_0^{-1^\top}, \widetilde{\boldsymbol{y}}_1^{-1^\top}, ..., \widetilde{\boldsymbol{y}}_N^{-1^\top}]^\top$, and $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{L}$ are defined accordingly. Similarly, define $\mathbf{K}$ to contain K_k in a block matrix form as in Ref. [30]. Let us also define matrices E_{x_k} and E_{u_k} to extract $\boldsymbol{x}_k$ and $\boldsymbol{u}_k$ from X and U, respectively, as $\boldsymbol{x}_k = E_{x_k} X$ and $\boldsymbol{u}_k = E_{u_k} U$. Then, the mean and

covariance of the state are derived in Ref. [30], summarized as follows:

$$\bar{x}_k = E_{x_k}(\mathbf{A}\bar{x}_0 + \mathbf{B}\bar{U} + \mathbf{C}), \qquad \text{(mean)}$$

$$P_k = \hat{P}_k + \widetilde{P}_k, \qquad \text{(total covariance)}$$

$$\hat{P}_k = E_{x_k}(I + \mathbf{BK})\mathbf{S}(I + \mathbf{BK})^\top E_{x_k}^\top, \qquad \text{(state dispersion covariance)}$$

$$P_{u_k} = E_{u_k}\mathbf{KSK}^\top E_{u_k}^\top \qquad \text{(control covariance)}$$

(24)

where recall that $\widetilde{P}_k$ is the navigation covariance and recursively calculated according to the Kalman filter as in Eq. (16). $\mathbf{S}$ is a term that is independent from the control variables, given by

$$\mathbf{S} = \mathbf{A}\hat{P}_{0^-}\mathbf{A}^\top + \mathbf{LP}_Y\mathbf{L}^\top. \qquad (25)$$

In our formulation, a Cholesky square-root form of covariance matrices is convenient due to its convex property. Such square-root forms of covariance matrices are given as follows:

$$P_k^{1/2} = \begin{bmatrix} \hat{P}_k^{1/2} & \widetilde{P}_k^{1/2} \end{bmatrix}, \quad \hat{P}_k^{1/2} = E_{x_k}(I + \mathbf{BK})\mathbf{S}^{1/2}, \quad P_{u_k}^{1/2} = E_{u_k}\mathbf{KS}^{1/2}, \quad \mathbf{S}^{1/2} = \begin{bmatrix} \mathbf{A}\hat{P}_{0^-}^{1/2} & \mathbf{LP}_Y^{1/2} \end{bmatrix} \qquad (26)$$

Remark 1. *As clear from Eqs. (24) and (26), $\bar{x}_k$, $P_k^{1/2}$, $\hat{P}_k^{1/2}$, and $P_{u_k}^{1/2}$ are affine in the FPC variables $\bar{u}_k$ and K_k (see Proposition 1 of Ref. [30]), which plays a key role in our convex formulation.*

B. Cost Function

Under ZoH control input $\boldsymbol{u}(t) = \boldsymbol{u}_k$, $\forall t \in [t_k, t_{k+1})$, Eq. (11) is equivalent to

$$J = \sum_{k=0}^{N-1} Q_{X \sim \|\boldsymbol{u}_k\|_2}(p)\Delta t_k, \quad \Delta t_k = t_{k+1} - t_k \qquad (27)$$

Applying Lemma 4 of Ref. [30], we can calculate the upper bound of Eq. (27) as follows:

$$J \leq J_{\text{ub}} = \sum_k \left[\|\bar{\boldsymbol{u}}_k\|_2 + m_{\chi^2}(1 - p, n_u)\left\|P_{u_k}^{1/2}\right\|_2 \right]\Delta t_k \qquad (28)$$

where $m_{\chi^2}(\varepsilon, n_u)$ is the square root of the quantile function of the chi-squared distribution with n_u degrees of freedom, evaluated at probability $(1 - \varepsilon)$, and is mathematically expressed as:

$$m_{\chi^2}(\varepsilon, n_u) = \sqrt{Q_{X \sim \chi^2(n_u)}(1 - \varepsilon)} \qquad (29)$$

Importantly, Eq. (28) is convex in $\bar{U}$ and $\mathbf{K}$ due to Remark 1. We minimize the upper bound J_{ub} instead of J directly. $m_{\chi^2}(\varepsilon, n_u)$ is straightforward to calculate in modern programming languages. We can calculate $m_{\chi^2}(\varepsilon, n_u)$ via $\sqrt{\texttt{chi2inv}(1 - \varepsilon, n_u)}$ in Matlab and $\sqrt{\texttt{chi2.cdf}(1 - \varepsilon, n_u)}$ in $\texttt{scipy}$.

C. Constraint Function

Tractable formulation of Eq. (13) depends on the specific form of constraints of interest. In this article, we consider three important constraints that are common in interplanetary low-thrust trajectory design: *thrust magnitude constraint*, *terminal constraints*, and *planetary gravity assist constraint*.

1. Thrust Magnitude Chance Constraints:

With a risk tolerance ε_u and maximum thrust acceleration $u_{\max} > 0$, the thrust magnitude constraint is expressed as a chance constraint as:

$$\mathbb{P}\left[\|\boldsymbol{u}_k\|_2 \leq u_{\max}\right] \geq 1 - \varepsilon_u, \ k = 0, 1, ..., N - 1 \tag{30}$$

Since Eq. (30) is a special form of Eq. 53 in Ref. [30], we can directly apply Lemma 3 of Ref. [30] to Eq. (30), yielding

$$\|\bar{\boldsymbol{u}}_k\|_2 + m_{\chi^2}\left(\varepsilon_u, n_u\right)\left\|P_{u_k}^{1/2}\right\|_2 \leq u_{\max}, \ k = 0, 1, ..., N - 1 \tag{31}$$

which is again convex in $\bar{U}$ and $\mathbf{K}$ due to Remark 1. Note that convex formulation of thrust magnitude chance constraints with mass uncertainty requires more analytical effort; see Ref. [8] for instance.

Remark 2. *Control chance constraints given by Eq.* (31) *provide a better approximation than the one proposed in Ridderhof et.al.[7] (and used in Benedikter et.al.[40]). The key difference lies in the definition of $m_{\chi^2}\left(\varepsilon_u, n_u\right)$ that appear in Eqs.* (28) *and* (31)*, which is defined as in Eq.* (29) *in this study. On the other hand, the equivalent of m_{χ^2} proposed in previous studies [7, 40] takes the following form (denoting it as $m_{\chi^2(\text{old})}$):*

$$m_{\chi^2(\text{old})} = \begin{cases} \sqrt{2 \ln \frac{1}{\varepsilon}} + \sqrt{n_u} & (n_u > 2), \\ \sqrt{2 \ln \frac{1}{\varepsilon}} & (n_u = 1, 2) \end{cases} \tag{32}$$

The values of m_{χ^2} given by Eq. (29) *(proposed one) and by Eq.* (32) *(previous one [7, 40]) coincide when $n_u = 2$, whereas those by Eq.* (29) *are smaller for $n_u > 2$, leading to tighter (hence better) constraints. This tighter bound is particularly beneficial in space applications since we typically deal with three-dimensional control input (e.g., thrust, acceleration, etc). Table 1 numerically compares the two approaches, confirming the tightness of our approximation. The benefit of this tighter bound is not limited to control magnitude constraints but broadly applicable to any magnitude-related*

Table 1   Magnitude constraint tightness comparison

Approach	$\{\varepsilon, n_u\} =$	$\{10^{-2}, 3\}$	$\{10^{-3}, 3\}$	$\{10^{-2}, 4\}$	$\{10^{-3}, 4\}$
Proposed [Eq. (29)]	$m_{\chi^2} =$	3.3682	4.0331	3.6437	4.2973
Previous [7] [Eq. (32)]	$m_{\chi^2 (\text{old})} =$	4.7669	5.4490	5.0349	5.7169

chance constraints, including state deviation constraints (see Ref. [30]).

2. Terminal Constraints

A convex formulation of Eq. (14) is readily available in existing studies (e.g., Refs. [18, 30]), yielding:

$$E_{x_N} (\mathbf{A}\bar{x}_0 + \mathbf{B}\bar{U} + \mathbf{C}) - x_f = 0, \tag{33a}$$

$$\left\| (P_f - \widetilde{P}_N)^{-1/2} E_{x_N} (I + \mathbf{B}\mathbf{K})\mathbf{S}^{1/2} \right\|_2 - 1 \leq 0. \tag{33b}$$

which are convex in $\bar{U}$ and $\mathbf{K}$.

D. Incorporating Planetary Gravity Assists

As commonly done in interplanetary mission design, we use the patched-conic model [47] to model the effect of GA on the state trajectory under uncertainty and formulate (chance) constraints accordingly. In this model, a GA rotates the spacecraft velocity relative to the planet while the periapsis radius about the planet, r_{peri} is constrained to be above a planetary impact radius, $r_{p\,\text{min}}$. Note that a higher-fidelity GA modeling is also possible to incorporate in the proposed framework. In that case, extending the formulation proposed in Ref. [48] to stochastic setting would be a straightforward approach, where the state statistics would be propagated in the planet-centric frame while imposing the impact constraint.

Let a GA occur at t_k, and assign k and $k + 1$ to the nodes that are immediately before and after the GA, respectively. Denoting the spacecraft velocity relative to the planet by $v_k^\infty = v_k - v_p(t_k)$, where $v_p(t_k)$ is the planet's velocity at t_k, under the patched-conic model, the GA effect is instantaneous and rotates v_k^∞ by a turn angle θ [47], i.e.,

$$t_k = t_{k+1}, \quad r_k = r_{k+1}, \quad \left\| v_{k+1}^\infty \right\|_2 = \left\| v_k^\infty \right\|_2, \quad v_{k+1}^\infty \cdot v_k^\infty = \left\| v_k^\infty \right\|_2^2 \cos \theta \tag{34}$$

The periapsis radius with respect to the planet during the flyby, $r_{\text{peri}}(\cdot)$, is a function of θ and v_k^∞ (hence of v_k and t_k), and must be greater than a minimum altitude, $r_{p\,\text{min}}$, as follows [47]:

$$r_{\text{peri}}(v_k, \theta, t_k) \geq r_{p\,\text{min}}, \quad r_{\text{peri}}(v_k, \theta, t_k) = \frac{\mu_p}{\left\| v_k^\infty \right\|_2^2} \left(\frac{1}{\sin \frac{\theta}{2}} - 1 \right), \tag{35}$$

where μ_p is the planet's gravity constant. To ease the formulation in Section IV.D.3, we equivalently express Eq. (35) as:

$$\left\| v_k^\infty \right\|_2 - \sqrt{\frac{\mu_p}{r_{p\,\mathrm{min}}} \left(\frac{1}{\sin \frac{\theta}{2}} - 1 \right)} \leq 0 \tag{36}$$

1. State Statistics Mapping across a Gravity Assist

While Eq. (34) is simple to apply for deterministic problems, it is not suited for stochastic problems because it does not clarify how the state statistics should be mapped across a GA. Thus, this study derives another form of Eq. (34) that furnishes an explicit formula that maps P_k to P_{k+1} across a GA event. To this end, we express the rotation of v_∞ via an orthogonal matrix $R_{\mathrm{GA}} \in \mathbb{R}^{3\times3}$ as $v_{k+1}^\infty = R_{\mathrm{GA}} v_k^\infty$, where R_{GA} is assumed to be deterministic (hence θ too) and may be considered as an optimization variable. However, using R_{GA} as an optimization variable introduces another issue; imposing the orthogonality condition is non-convex. To address this issue, this study utilizes *Cayley transform*[‡] to express the orthogonal matrix R_{GA} in terms of a skew-symmetric matrix $V \in \mathbb{R}^{3\times3}$ (hence $V^\top = -V$) as $R_{\mathrm{GA}} = (I_3 + V)^{-1}(I_3 - V)$, yielding

$$v_{k+1}^\infty = (I_3 + V)^{-1}(I_3 - V)v_k^\infty \quad \Leftrightarrow \quad v_{k+1} - v_p(t_k) = (I_3 + V)^{-1}(I_3 - V)[v_k - v_p(t_k)] \tag{37}$$

where $I_n \in \mathbb{R}^{n\times n}$ is an identity matrix. It is clear that, because $R_{\mathrm{GA}} = (I_3 + V)^{-1}(I_3 - V)$ is a rotation matrix, Eq. (37) automatically satisfies the third constraint of Eq. (34).

We can parameterize the skew-symmetric matrix V by only three scalars and express it as $V = [u_k]_\times$ with $u_k \in \mathbb{R}^3$, where $[\cdot]_\times$ denotes the matrix cross product operator. Here, we can see u_k as a control input at t_k that characterizes the effect of the GA at this epoch. This viewpoint, combined with the Cayley transformation, allows us to describe the mapping of state under the effect of a GA as a nonlinear function, which then provides a linear mapping equation. This idea is formally stated in Lemma 1.

Lemma 1. *Using Cayley transformation, the mapping of orbital states x_k to x_{k+1} due to a gravity assist is expressed as a nonlinear function of x_k, u_k, t_k, as follows:*

$$x_{k+1} = f_{\mathrm{GA}}(x_k, u_k, t_k) = \begin{bmatrix} r_k \\ v_p(t_k) + R_{\mathrm{GA}}(u_k) \cdot [v_k - v_p(t_k)] \end{bmatrix}, \quad R_{\mathrm{GA}}(u_k) = (I_3 + [u_k]_\times)^{-1}(I_3 - [u_k]_\times) \tag{38}$$

where $[\cdot]_\times$ denotes the matrix cross product operator, and $(I_3 + [u_k]_\times)^{-1}$ always exists. Also, its linearized equation

---

[‡]For any skew-symmetric matrix A, a matrix Q that satisfies $Q = (I + A)^{-1}(I - A)$ is an orthogonal matrix with its determinant +1, i.e., $QQ^\top = I$ and $|Q| = 1$.

can be expressed as $x_{k+1} = A_k x_k + B_k u_k + c_k$, where the linear system matrices A_k, B_k, c_k are given by

$$A_k = \left.\frac{\partial f_{GA}}{\partial x}\right|^{\text{ref}} = \begin{bmatrix} I_3 & 0_{3\times3} \\ 0_{3\times3} & R_{GA}(u_k^{\text{ref}}) \end{bmatrix}, \quad B_k = \left.\frac{\partial f_{GA}}{\partial u}\right|^{\text{ref}} = \begin{bmatrix} 0_{3\times3} \\ (I_3 + [u_k^{\text{ref}}]_\times)^{-1}([v_{k+1}^{\text{ref}}]_\times + [v_k^{\text{ref}}]_\times - 2[v_p]_\times) \end{bmatrix}, \quad (39)$$

$$c_k = f_{GA}(x_k^{\text{ref}}, u_k^{\text{ref}}) - A_k x_k^{\text{ref}} - B_k u_k^{\text{ref}}$$

Proof. See Appendix .A. $\qquad\qquad\square$

Using Lemma 1, we can integrate the effect of GAs into the form of Eq. (18) with $L_{k+1} = 0$, assuming that we do not have the opportunity to perform measurement updates between t_k and t_{k+1} since $t_k = t_{k+1}$ at a GA. This way, we can capture the statistical effect of GAs in a unified form, and hence the state statistics mapping due to GA are characterized by Eqs. (24) and (26) with A_k, B_k, c_k given as in Eq. (39). Note here that, since the turn angle θ (and hence V) is assumed deterministic, u_k at the GA epoch is also deterministic, and therefore $\bar{u}_k = u_k$ and $K_k = 0$.

The development above takes care of the first three equations of Eq. (34) in the stochastic setting. Sections IV.D.2 and IV.D.3 discuss the reformulation of the last equation of Eq. (34) and Eq. (36).

2. Turn angle constraint

The turn angle constraint given by the last equation of Eq. (34) is a deterministic constraint in the current formulation due to the assumption that the turn angle θ is deterministic. Hence, the turn angle constraint needs to be satisfied at the mean values of v_{k+1}^∞ and v_k^∞, i.e.,

$$g_{GA-\text{turn}}(\bar{x}_{k+1}, \bar{x}_k, \theta) = \left\|\bar{v}_k^\infty\right\|_2^2 \cos\theta - \bar{v}_{k+1}^\infty \cdot \bar{v}_k^\infty = 0, \quad \bar{v}_k^\infty = E_v x_k - v_p(t_k) \qquad (40)$$

where $E_v = [0_{3\times3} \ I_3]$ extracts the velocity from x_k. Eq. (40) is clearly non-convex in $\bar{x}_{k+1}, \bar{x}_k, \theta$ and thus linearized about $\bar{x}_k^{\text{ref}}, \theta^{\text{ref}}$ to form convex subproblem at each iteration, where $(\cdot)^{\text{ref}}$ indicates evaluation at the solution of the previously accepted iteration. Indicating the linearized equation with $\widetilde{(\cdot)}$, linearizing Eq. (40) about $\bar{x}_{k+1}^{\text{ref}}, \bar{x}_k^{\text{ref}}, \theta^{\text{ref}}$ leads to

$$\widetilde{g}_{GA-\text{turn}}(\bar{x}_{k+1}, \bar{x}_k, \theta) = g_{GA-\text{turn}}^{\text{ref}} + \left.\frac{\partial g_{GA-\text{turn}}}{\partial \bar{x}_{k+1}}\right|^{\text{ref}}(\bar{x}_{k+1} - \bar{x}_{k+1}^{\text{ref}}) + \left.\frac{\partial g_{GA-\text{turn}}}{\partial \bar{x}_k}\right|^{\text{ref}}(\bar{x}_k - \bar{x}_k^{\text{ref}}) + \left.\frac{\partial g_{GA-\text{turn}}}{\partial \theta}\right|^{\text{ref}}(\theta - \theta^{\text{ref}}),$$

$$\frac{\partial g_{GA-\text{turn}}}{\partial \bar{x}_k} = 2\bar{v}_k^{\infty\top} E_v \cos\theta - \bar{v}_{k+1}^{\infty\top} E_v, \quad \frac{\partial g_{GA-\text{turn}}}{\partial \bar{x}_{k+1}} = -\bar{v}_k^{\infty\top} E_v, \quad \frac{\partial g_{GA-\text{turn}}}{\partial \theta} = -\left\|\bar{v}_k^\infty\right\|_2^2 \sin\theta$$

$$(41)$$

3. Impact chance constraint

Due to the stochasticity in $\boldsymbol{v}_k$ and hence in $\boldsymbol{v}_k^\infty$, the impact constraint, Eq. (36), is ill-defined in its current form, and needs to be expressed as a chance constraint with a risk level $\varepsilon_{\mathrm{GA}}$:

$$\mathbb{P}\left[\left\|\boldsymbol{v}_k^\infty\right\|_2 - \sqrt{\frac{\mu_p}{r_{p\,\min}}\left(\frac{1}{\sin\frac{\theta}{2}} - 1\right)} \leq 0\right] \geq 1 - \varepsilon_{\mathrm{GA}} \tag{42}$$

Noting that $\boldsymbol{v}_k^\infty = \boldsymbol{v}_k - \boldsymbol{v}_p(t_k) = E_v \boldsymbol{x}_k - \boldsymbol{v}_p(t_k) \sim \mathcal{N}(E_v \bar{\boldsymbol{x}}_k - \boldsymbol{v}_p,\ E_v P_k E_v^\top)$, we can take the same approach as for Eq. (30) and directly apply Lemma 3 of Ref. [30] to Eq. (42), leading to

$$g_{\mathrm{GA-impact}}(\bar{\boldsymbol{x}}_k, P_k^{1/2}, \theta) = \left\|E_v \bar{\boldsymbol{x}}_k - \boldsymbol{v}_p\right\|_2 + m_{\chi^2}(\varepsilon_{\mathrm{GA}}, 3)\left\|E_v P_k^{1/2}\right\|_2 - \sqrt{\frac{\mu_p}{r_{p\,\min}}\left(\frac{1}{\sin\frac{\theta}{2}} - 1\right)} \leq 0 \tag{43}$$

which is convex in $\bar{\boldsymbol{u}}_k, K_k, \forall k$ due to Remark 1, but not convex in θ. Thus, $g_{\mathrm{GA-impact}}(\cdot)$ is linearized about θ^{ref}, yielding

$$\widetilde{g}_{\mathrm{GA-impact}}(\bar{\boldsymbol{x}}_k, P_k^{1/2}, \theta) = g_{\mathrm{GA-impact}}(\bar{\boldsymbol{x}}_k, P_k^{1/2}, \theta^{\mathrm{ref}}) - \sqrt{\frac{\mu_p}{r_{p\,\min}}}\ \frac{\partial}{\partial\theta}\left(\sqrt{\frac{1}{\sin\frac{\theta}{2}} - 1}\right)_{\theta=\theta^{\mathrm{ref}}}(\theta - \theta^{\mathrm{ref}}) \tag{44}$$

V. Solution Method via Sequential Convex Programming

A. Penalized Convex Subproblem with Trust Region

Like typical SCP approaches for nonlinear trajectory optimization, we introduce *virtual buffers* and *trust region* and penalize the constraint buffers to facilitate the constraint satisfaction; see [24, 26, 49] for more detail.

Virtual buffers are typically introduced to relax constraints that are originally non-convex and hence need to be approximated for forming a convex subproblem. On the other hand, relaxing all the approximated constraints can lead to too loose problem formulations. In general, the choice of which constraints to relax highly depends on the problem and reflects the mission designers' experience and insights. Denote the equality and inequality constraints to relax as $\boldsymbol{g}^{\mathrm{eq,relax}}(=0)$ and $\boldsymbol{g}^{\mathrm{ineq,relax}}(\leq 0)$, respectively, and their approximated convex forms as $\widetilde{\boldsymbol{g}}^{\mathrm{eq,relax}}$ and $\widetilde{\boldsymbol{g}}^{\mathrm{ineq,relax}}$. Then, the approximated constraints with relaxation via virtual buffers are expressed as:

$$\widetilde{\boldsymbol{g}}^{\mathrm{eq,relax}}(\bar{X}, \bar{U}, \mathbf{K}, \boldsymbol{\Theta}) = \boldsymbol{\xi}, \quad \widetilde{\boldsymbol{g}}^{\mathrm{ineq,relax}}(\bar{X}, \bar{U}, \mathbf{K}, \boldsymbol{\Theta}) \leq \boldsymbol{\zeta}, \tag{45}$$

where $\boldsymbol{\xi}$ and $\boldsymbol{\zeta} \geq 0$ are virtual buffers for equality and inequality constraints, respectively. Recall that $\boldsymbol{\Theta}$ denotes a vector of relevant parameters to optimize (e.g., turn angle θ at each GA).

Specifically, in this work, we relax the final mean state constraint (equality) and the gravity-assist impact chance constraint (inequality), which are both non-convex in variables in the original form. Hence, $\widetilde{\boldsymbol{g}}^{\mathrm{eq,relax}}$ consists of Eq. (33a)

and $\widetilde{g}^{\text{ineq,relax}}$ of Eq. (44).

We formulate the penalty function based on an SCP algorithm called SCvx* (*SCvx-star*) [26], which leverages the augmented Lagrangian framework to extend its predecessor SCvx [24, 50] to provide a theoretical convergence guarantee to a feasible optimal solution. SCvx* introduces Lagrange multipliers λ and $\mu \geq 0$, which have the same dimensions as the virtual buffers ξ and ζ, respectively, and defines the penalty function as:

$$P(\xi, \zeta) = \sum_i \lambda_i \xi_i + \frac{1}{w} \phi(w\xi_i) + \mu_i \zeta_i + \frac{1}{w} \phi(w[\zeta_i]_+) \tag{46}$$

where $w > 0$ is a penalty weight, $\phi(\cdot) : \mathbb{R} \mapsto \mathbb{R}_+$ is strictly convex and smooth (once continuously differentiable), and $[\zeta_i]_+ = \max(\zeta_i, 0)$. The multipliers λ_i and μ_i are updated according to the augmented Lagrangian framework (in its generalized form; see Chapter 5, [51]):

$$\lambda_i \leftarrow \lambda_i + \nabla\phi(wg_i^{\text{eq,relax}}), \quad \mu_i \leftarrow \left[\mu_i + \nabla\phi(wg_i^{\text{ineq,relax}})\right]_+, \tag{47}$$

where $g_i^{\text{eq,relax}}$ and $g_i^{\text{ineq,relax}}$ represent the i-th elements of the original functions for the relax equality and inequality constraints, respectively. If we use $\phi(z) = z^2/2$, then $\nabla\phi = z$ (where $z \in \mathbb{R}$ is a dummy variable), which reduces Eq. (47) to the conventional multiplier updates: $\lambda_i \leftarrow \lambda_i + wg_i^{\text{eq,relax}}$ and $\mu_i \leftarrow [\mu_i + wg_i^{\text{ineq,relax}}]_+$ (see, for instance, [26, 52]).

On the other hand, this study utilizes the following form of $\phi(\cdot)$ to define the penalty function:

$$\phi(z) = \frac{1}{\tau}|z|^\tau + \frac{1}{2}z^2, \quad \nabla\phi(z) = \text{sign}(z)|z|^{\tau-1} + z, \quad \tau \in (1, 2) \tag{48}$$

which is convex in z and closely approximates l_1-norm penalty ($\phi(z) = |z|$) near the origin ($z \approx 0$) when τ is close to unity. Eq. (48) is chosen to take advantage of the desirable property of l_1-norm penalty (known as a class of *exact* penalty functions in constrained optimization literature [53, 54]) while retaining the smoothness; the smoothness (continuous differentiability) is required in the augmented Lagrangian framework (Chapter 5, [51]), while the vanilla l_1-norm penalty $\phi(z) = |z|$ is not smooth at the origin.

Fig. 2 shows the values of the l_1-like penalty function and its gradient with different τ given by Eq. (48) and compares them against the l_1 exact penalty $\phi(z) = |z|$ and quadratic penalty $\phi(z) = z^2$. This comparison visually verifies that Eq. (48) approaches the l_1 penalty near the origin as $\tau \to 1$ while retaining the smoothness, i.e., continuous $\nabla\phi(z)$, for any z including the origin, as clear from Fig. 2(b). Based on these observations, the numerical examples in this paper employ $\tau = 1.1$.
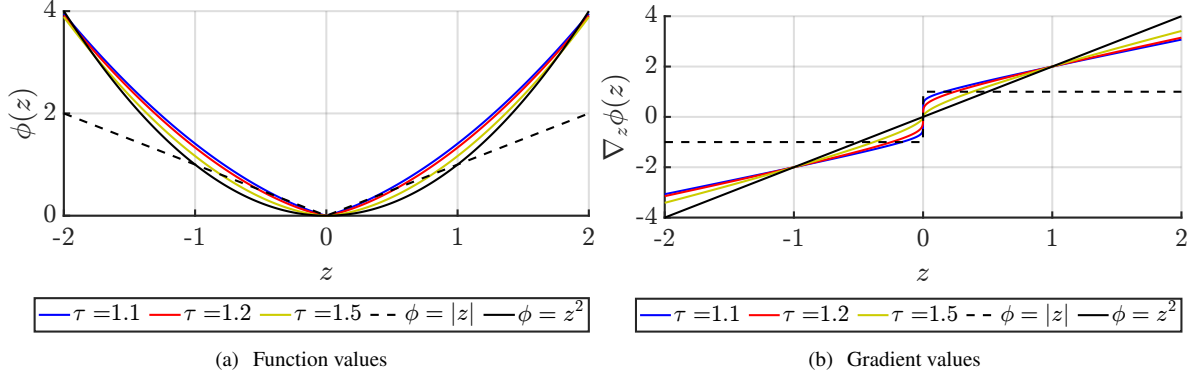
(a) Function values   (b) Gradient values

Figure 2 l_1**-like smooth penalty function** $\phi(z)$ **given in Eq. (48) and its gradient for different** τ **for** $z \in [-2, 2]$**, compared with** l_1 **exact penalty function** $|z|$ **and quadratic penalty** z^2

We then argument the cost function (DV99 upper bound) Eq. (28) by the augmented-Lagrangian penalty Eq. (46) as:

$$J_{\text{aug}}(\overline{X}, \overline{U}, \mathbf{K}, \mathbf{\Theta}, \boldsymbol{\xi}, \boldsymbol{\zeta}) = J_{\text{ub}}(\overline{U}, \mathbf{K}) + P(\boldsymbol{\xi}, \boldsymbol{\zeta}) \tag{49}$$

We also impose trust region constraints on the decision variables to avoid artificial unboundedness:

$$\left\| D_x(\overline{X} - \overline{X}^{\text{ref}}) \right\|_{\infty} \le \Delta_{\text{TR}}, \quad \left\| D_u(\overline{U} - \overline{U}^{\text{ref}}) \right\|_{\infty} \le \Delta_{\text{TR}}, \quad \left\| D_\theta(\mathbf{\Theta} - \mathbf{\Theta}^{\text{ref}}) \right\|_{\infty} \le \Delta_{\text{TR}}, \tag{50}$$

where $\Delta_{\text{TR}} > 0$ is the trust region radius, D_x, D_u, and D_θ are scaling (typically diagonal) matrices for the state, control, and parameter updates, respectively; intuitively, greater D_x, D_u leads to a stricter trust region for a given Δ_{TR}. We do not impose a trust region on $\mathbf{K}$, as $\mathbf{K}$ only linearly affects the trajectory of (Cholesky factor of) covariance matrices in our formulation.

Remark 3. *Both Eqs. (49) and (50) are convex in variables.* J_{aug} *in Eq. (49) is convex in* $\overline{U}$, $\mathbf{K}$ *as clear from Eq. (28) and Remark 1.* $P(\boldsymbol{\xi}, \boldsymbol{\zeta})$ *in Eq. (49), defined in Eq. (46) with Eq. (48), is convex in* $\boldsymbol{\xi}$ *and* $\boldsymbol{\zeta}$ *because Eq. (48) is convex (which proves* $\phi(w\xi_i)$ *being convex in* $\boldsymbol{\xi}$*), and the fact that* $[\zeta_i]_+$ *is non-negative and* $\tau > 1$ *implies the convexity of* $\phi(w[\zeta_i]_+)$ *in* $\boldsymbol{\zeta}$ *(recall that, if g is convex and nonnegative, then* $g(x)^p$ *is convex for* $p \ge 0$ *[55]). Eq. (50) is clearly convex in* $\overline{X}$*.*

Combing the discussion thus far, the convex subproblem is formulated as in Problem 2.

Problem 2 (Convex subproblem). *Find* $\overline{X}^*$, $\overline{U}^*$, $\mathbf{K}^*$, $\mathbf{\Theta}^*$, $\boldsymbol{\xi}^*$, *and* $\boldsymbol{\zeta}^*$ *that minimize the augmented cost Eq. (49) while satisfying the filtered state dynamics constraints Eq. (26) including the GA effect Eq. (39), control magnitude constraints Eq. (31), GA turn-angle constraints Eq. (41), terminal covariance constraints Eq. (33b), relaxed constraints Eq. (45) with Eqs. (33a) and (44), and trust region constraints Eq. (50).*

Algorithm 1 Robust Trajectory Optimization under Uncertainty via SCvx*

 1: Generate the initial reference trajectory $\{\overline{X}^{\mathrm{ref}}, \overline{U}^{\mathrm{ref}}, \mathbf{K}^{\mathrm{ref}}(=0), \boldsymbol{\Theta}^{\mathrm{ref}}\}$
 2: Initialize SCvx* parameters   ▷ *Line 1 of Ref. [26]*
 3: **while** the convergence criteria not met **do**   ▷ *Line 2 of Ref. [26]*
 4: Obtain linearized block system matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{L}, \mathbf{Y}$ ▷ *based on the current reference* $\overline{X}^{\mathrm{ref}}, \overline{U}^{\mathrm{ref}}$
 5: $\{\overline{X}^*, \overline{U}^*, \mathbf{K}^*, \boldsymbol{\Theta}^*, \boldsymbol{\xi}^*, \boldsymbol{\zeta}^*\} \leftarrow$ solve Problem 2
 6: **if** acceptance conditions met **then**   ▷ *Eq. (54)*
 7:   $\{\overline{X}^{\mathrm{ref}}, \overline{U}^{\mathrm{ref}}, \mathbf{K}^{\mathrm{ref}}, \boldsymbol{\Theta}^{\mathrm{ref}}\} \leftarrow \{\overline{X}^*, \overline{U}^*, \mathbf{K}^*, \boldsymbol{\Theta}^*\}$ ▷ *solution update*
 8:   Update Lagrange multipliers   ▷ *Line 13-15, Algorithm 1 of Ref. [26], with Eq. (47)*
 9: Update trust region   ▷ *Eq. (55)*
10: **return** $\{\overline{X}^{\mathrm{ref}}, \overline{U}^{\mathrm{ref}}, \mathbf{K}^{\mathrm{ref}}, \boldsymbol{\Theta}^{\mathrm{ref}}\}$

Remark 4. *Problem 2 is convex in variables due to Remark 1, the discussion in Section IV.C, and Remark 3.*

B. Sequential Convex Programming via SCvx* Algorithm

With Problem 1 reformulated in a deterministic, convex form as in Problem 2, we are ready to efficiently solve the problem via sequential convex programming (SCP). In particular, this study utilizes a recent SCP algorithm called SCvx* [26]. The SCvx* algorithm extends a successful SCP algorithm known as SCvx [24, 50] to provide a theoretical guarantee for convergence to a feasible optimal solution by leveraging the augmented Lagrangian framework [51].

Algorithm 1 presents the solution algorithm based on SCvx*. For the detail of the algorithm and its convergence property, see Ref. [26] (and Ref. [56], which corrects a few minor typographical errors). The returned quantities at the convergence $\{\overline{X}^*, \overline{U}^*, \mathbf{K}^*, \boldsymbol{\Theta}^*\}$ approximately solves Problem 1, where the propagation of state uncertainty is approximated as the linear propagation of state mean and covariance. Extending the solution method to incorporate non-Gaussian uncertainty quantification while retaining the computational tractability is an important direction of ongoing research. There have been promising studies developed to address the problem of chance-constrained control under non-Gaussian uncertainty, such as those via Gaussian mixture model [57, 58] and polynomial chaos expansion [6, 59], which have been successfully demonstrated in simpler systems, single-maneuver planning, or without FPC policy optimization (i.e., open-loop control).

The SCP algorithm developed in this study is nearly identical to the original SCvx*, although there are a few aspects that warrant some discussion, including some algorithmic modifications to extend the algorithm to be applicable to trajectory optimization under uncertainty. These modifications have been also employed in our recent paper [11] (preprint version [38]). These aspects are highlighted in the following.

1. Objective Improvement Measure

A key in Algorithm 1 is the *solution acceptance* criterion in line 6. Similar to [24, 26], the solution acceptance relies on the ratio of the nonlinear cost reduction ΔJ to the approximated one ΔL:

$$\rho = \frac{\Delta J}{\Delta L} = \frac{J_{\text{aug}}^{\text{NL}}(\overline{X}^{\text{ref}}, \overline{U}^{\text{ref}}, \mathbf{K}^{\text{ref}}, \mathbf{\Theta}^{\text{ref}}) - J_{\text{aug}}^{\text{NL}}(\overline{X}^*, \overline{U}^*, \mathbf{K}^*, \mathbf{\Theta}^*)}{J_{\text{aug}}^{\text{NL}}(\overline{X}^{\text{ref}}, \overline{U}^{\text{ref}}, \mathbf{K}^{\text{ref}}, \mathbf{\Theta}^{\text{ref}}) - J_{\text{aug}}(\overline{X}^*, \overline{U}^*, \mathbf{K}^*, \mathbf{\Theta}^*, \boldsymbol{\xi}^*, \boldsymbol{\zeta}^*)} \tag{51}$$

where J_{aug} is given in Eq. (49) while $J_{\text{aug}}^{\text{NL}}$ represents the cost of the original non-convex problem with augmented Lagrangian. Note that one must use the same values of ω, λ, μ when calculating all the $J_{\text{aug}}^{\text{NL}}$ and J_{aug} in Eq. (51).

Evaluation of $J_{\text{aug}}^{\text{NL}}$ requires approximation. Exact evaluation of $J_{\text{aug}}^{\text{NL}}$ based on Problem 1 is intractable since it necessitates evaluating the original DV99 cost Eq. (11) and penalty costs associated with chance constraints Eq. (13) and terminal constraint Eq. (14) under the nonlinear filtering process Eq. (8) driven by the nonlinear SDE Eq. (6) and measurements Eq. (7). To approximately compute $J_{\text{aug}}^{\text{NL}}$ in a tractable manner, this work takes full advantage of the linear covariance-based formulation developed in Section IV. Specifically, we define $J_{\text{aug}}^{\text{NL}}$ as:

$$J_{\text{aug}}^{\text{NL}}(\overline{X}, \overline{U}, \mathbf{K}, \mathbf{\Theta}) = J_{\text{aug}}[\overline{X}, \overline{U}, \mathbf{K}, \boldsymbol{g}^{\text{eq,relaxed}}(\overline{X}, \overline{U}, \mathbf{K}, \mathbf{\Theta}), \boldsymbol{g}^{\text{ineq,relaxed}}(\overline{X}, \overline{U}, \mathbf{K}, \mathbf{\Theta})], \tag{52}$$

where $\boldsymbol{g}^{\text{eq,relaxed}}$ and $\boldsymbol{g}^{\text{ineq,relaxed}}$ encompass the constraints that are relaxed and appended to $P(\cdot)$ in Problem 2:

$$\boldsymbol{g}^{\text{eq,relaxed}}(\overline{X}, \overline{U}, \mathbf{K}, \mathbf{\Theta}) = \bar{x}_0 + \sum_{k=0}^{N-1} \int_{t_k}^{t_{k+1}} \boldsymbol{f}(\bar{x}, \bar{u}_k, t) \mathrm{d}t - \boldsymbol{x}_f,$$

$$\boldsymbol{g}^{\text{ineq,relaxed}}(\overline{X}, \overline{U}, \mathbf{K}, \mathbf{\Theta}) = \left\{ g_{\text{GA-impact}}(\bar{x}_{k_{\text{GA}}}, P_{k_{\text{GA}}}^{1/2}, \theta_{k_{\text{GA}}}) \right\}_{\forall k_{\text{GA}}} \quad \text{(Eq. (43))} \tag{53}$$

This formulation implies that we utilize Eq. (28) for DV99 evaluation while penalizing the constraint violation of Eqs. (14a) and (42), where we replace Eq. (14a) with the nonlinear deterministic propagation of the state under the mean control $\bar{u}$ and Eq. (42) with the deterministic yet non-convex impact chance constraint given by Eq. (43). Note here that the other constraints in Problem 1 are considered to be automatically satisfied as a result of solving Problem 2.

2. Step Acceptance and Trust Region Update

The algorithm then uses ρ to determine acceptance of the current iteration and the update of Δ_{TR}. In this work, we employ a different scheme than the one proposed in Ref. [26]. Similar to [11] (preprint version [38]), we employ a different scheme than the one proposed in Ref. [26], as follows:

$$\begin{cases} \text{accept} & \text{if } \rho \in [1 - \eta_0, 1 + \eta_0] \\ \text{reject} & \text{else} \end{cases} \tag{54}$$

Table 2 Parameters used for the SCP algorithm

Symbol	ϵ_{opt}	ϵ_{feas}	$\{\eta_0, \eta_1, \eta_2\}$	$\{\alpha_1, \alpha_2, \beta, \gamma\}$	$w^{(1)}$	w_{max}	$\Delta_{\text{TR}}^{(1)}$	$\Delta_{\text{TR,min}}$	$\Delta_{\text{TR,max}}$	τ
Value	10^{-6}	10^{-6}	$\{1.0, 0.5, 0.1\}$	$\{2.0, 3.0, 2, 0.95\}$	10^2	10^{10}	0.1	10^{-8}	1.0	1.1

where $1 \geq \eta_0 > \eta_1 > \eta_2 > 0$. The trust region is updated as follows:

$$\Delta_{\text{TR}} \leftarrow \begin{cases} \max\{\alpha_2 \Delta_{\text{TR}}, \Delta_{\text{TR,min}}\} & \text{if } \rho \in [1 - \eta_2, 1 + \eta_2] \\ \Delta_{\text{TR}} & \text{elseif } \rho \in [1 - \eta_1, 1 + \eta_1] \\ \min\{\Delta_{\text{TR}}/\alpha_1, \Delta_{\text{TR,max}}\} & \text{else} \end{cases} \tag{55}$$

where $\alpha_1 > 1$ and $\alpha_2 > 1$, and $0 < \Delta_{\text{TR,min}} < \Delta_{\text{TR,max}}$ are the lower and upper bounds of the trust region radius.

These algorithmic modifications of SCvx* are motivated by the fact that the convex subproblem Problem 2 is based on *inexact linearization* of the state covariance propagation. See our recent work [11] (preprint version [38]) for further discussion on this aspect.

C. Implementation

1. Sequential Convex Programming

Table 2 lists the parameters used for the presented SCP algorithm. For trust region scaling, $D_x = D_u = D_\theta = 1$ are used. In each iteration, we use CVX [60] with MOSEK [61] in Matlab to solve Problem 2. We normalize all the physical quantities by unit length l_{unit} and time t_{unit}, which are defined as $l_{\text{unit}} = 1$ AU and $t_{\text{unit}} = \sqrt{\mu_{\text{sun}}/l_{\text{unit}}^3}$.

2. Safeguard

As discussed in Ref. [49], we sometimes encounter an undesirable situation where numerically ill-conditioned problems prevent a convex solver from finding the solution to Problem 2. It is observed that such numerical issues are often due to large penalty weights, $w^{(i)}$. Thus, as a safeguard, we apply the following remedy when the SCP algorithm encounters such numerical issues: reduce $w^{(i)}$ by a factor $\beta > 1$, i.e., $w^{(i+1)} = w^{(i)}/\beta$, and re-run the convex programming without changing any other parameters or reference trajectory.

3. Nonlinear Monte Carlo Simulation for Solution Verification

After obtaining a solution, it is insightful to perform nonlinear Monte Carlo simulations to verify that the approximations made for the tractable formulation are reasonable. To that end, we utilize extended Kalman filter (EKF) in place of KF given in Eq. (16) for filtering, where the time update is replaced with the nonlinear mean propagation, the linear system matrices $A_k, B_k, c_k, G_{\text{exe},k}, G_k$ are propagated about the nonlinear mean state, and the innovation process

utilizes the nonlinear model given in Eq. (9). The Brownian motion in the SDE Eq. (6) is approximated in the same manner as in Eq. 64 of Ref. [5].

We seek a way to convert the policy defined in Eq. (20), which takes the stochastic process z_k defined in Eq. (21) to calculate statistical TCMs, to a policy that is based on $\hat{x}_k$, without affecting the optimization result. Such conversion is desirable when applying optimized FPC policies in nonlinear Monte Carlo because it is less straightforward to calculate the nonlinear equivalent of the stochastic process z_k. Hence, we utilize the result stated in Lemma 2 to equivalently convert Eq. (20) (the z_k-based FPC policy) to Eq. (56), which feeds the deviation of $\hat{x}_k$ from $\bar{x}_k$ to calculate statistical TCMs. With this $\hat{x}_k$-based FPC policy, we can directly use the state estimate $\hat{x}_k$ from EKF to calculate u_k in nonlinear Monte Carlo.

Lemma 2. *The output feedback control policy* $u_k = \bar{u}_k + K_k z_k$ *given in Eq. (20) is equivalently expressed as the following state-estimate history feedback control policy:*

$$u_k = \bar{u}_k + \sum_{i=0}^{k} \hat{K}_{k,i}(\hat{x}_i - \bar{x}_i) \tag{56}$$

where $\hat{K}_{k,i}$ *correspond to sub-matrices of* $\hat{\mathbf{K}}$ *as follows:*

$$\hat{\mathbf{K}} = \begin{bmatrix} \hat{K}_{0,0} & 0 & 0 & \cdots & 0 \\ \hat{K}_{1,0} & \hat{K}_{1,1} & \ddots & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 & 0 \\ \hat{K}_{N-1,0} & \hat{K}_{N-1,1} & \cdots & \hat{K}_{N-1,N-1} & 0 \end{bmatrix}, \quad \hat{\mathbf{K}} = \mathbf{K}(I + \mathbf{BK})^{-1} \tag{57}$$

where $\mathbf{K}$ *solves Problem 2, and* $(I + \mathbf{BK})$ *is invertible.*

Proof. See Appendix .B. □

Fig. 3 illustrates the flow of nonlinear MC simulations implemented in this study. We propagate two sets of trajectories for each sample: true trajectory $x(t)$ in the world and estimate trajectory $\hat{x}(t)$ in the brief space. For $x(t)$, we incorporate noise acting on the acceleration due to dynamics mis-modeling (while approximating the Brownian motion in SDE in a similar manner to [5]) in the nonlinear propagation as well as execution error based on Gates model. In belief space, we calculate planned control u_k that incorporates statistical corrections based on the optimized FPC, propagate the estimate $\hat{x}(t)$ nonlinearly, and perform OD to obtain $\hat{x}_{k+1}$ via extended Kalman filter (EKF) by fusing measurements. Again, calculation of u_k is based on the state-estimate history feedback FPC policy given by Eq. (56); it is observed that the state-estimate feedback FPC policy Eq. (56) leads to MC trajectories that are more consistent with the linearly predicted covariance evolution, compared to the policy based on Eq. (20). Although these two policies are
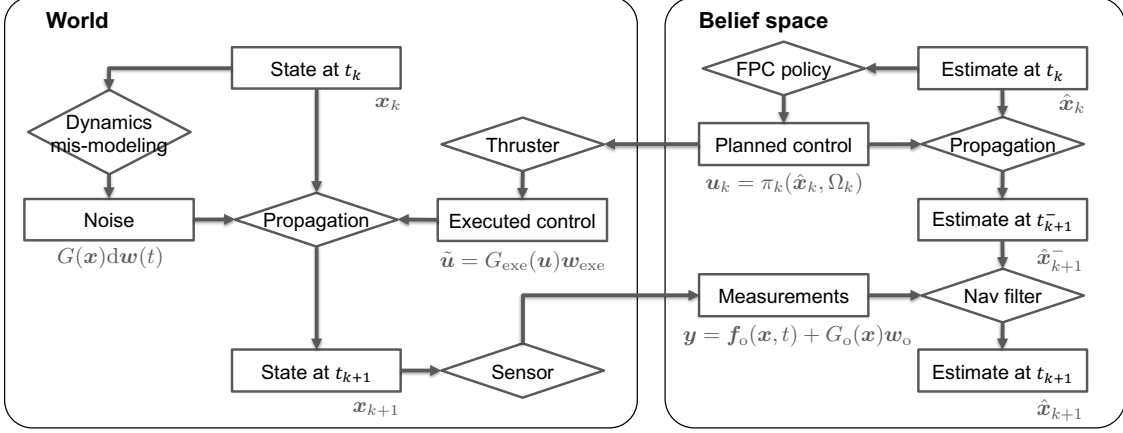
Figure 3   Nonlinear Monte Carlo flow chart, illustrating a simulation flow from t_k to t_{k+1}.

equivalent in linear sense, their performance difference in nonlinear MC makes sense, since Eq. (20) uses z, which needs to be linearly propagated via Eq. (21), whereas Eq. (56) utilizes $\hat{x}$, which may be estimated nonlinearly.

VI. Numerical Example

A. Scenario: Earth-Mars-Ceres transfer

We consider an Earth-Mars-Ceres transfer with a GA at Mars (MGA) to demonstrate the proposed method. The spacecraft is launched from Earth on 2030 Dec 18, performs the MGA on 2031 Aug 15, and arrives at Ceres on 2035 Aug 14, with the total time-of-flight (ToF) being 1700 days ($\approx$ 4.66 years). The position and velocity of the celestial bodies are defined based on the DE430 planetary ephemeris model provided by JPL. The trajectory is discretized in time with intervals about $\Delta t_k = 45$ days, resulting in 7 nodes for the Earth-Mars leg and 34 nodes for the Mars-Ceres leg. We assume $T_{\max} = 0.35$ N, $m = 3000$ kg, and $u_{\max} = T/m$, and the dynamics to be Keplerian about Sun, with instantaneous change in velocity at Mars due to the MGA.

The Earth launch is modeled as:

$$\boldsymbol{r}_0 - \boldsymbol{r}_{\text{Earth}}(t_0) = 0, \quad \|\boldsymbol{v}_0 - \boldsymbol{v}_{\text{Earth}}(t_0)\|_2 \leq v_{\infty,\max}, \tag{58}$$

that is, the spacecraft position is the same as that of Earth at the launch epoch t_0, while the velocity relative to Earth must be within the launcher's capability denoted by $v_{\infty,\max}$. We assume $v_{\infty,\max} = 3.5$ km/s, which corresponds to $C_3 = v_{\infty,\max}^2 = 12.25$ km^2/s^2.

The MGA is modeled by using the patched-conic model (see Section IV.D). We assume the Mars gravitational parameter to be $\mu_p = 42828$ km^3/s^2 and the minimum periapsis radius for Mars GA to be $r_{p\,\min} = 3689.5$ km, about 300 km altitude above the Mars surface.

(a) Deterministic optimal trajectory   (b) Time history of deterministic optimal control
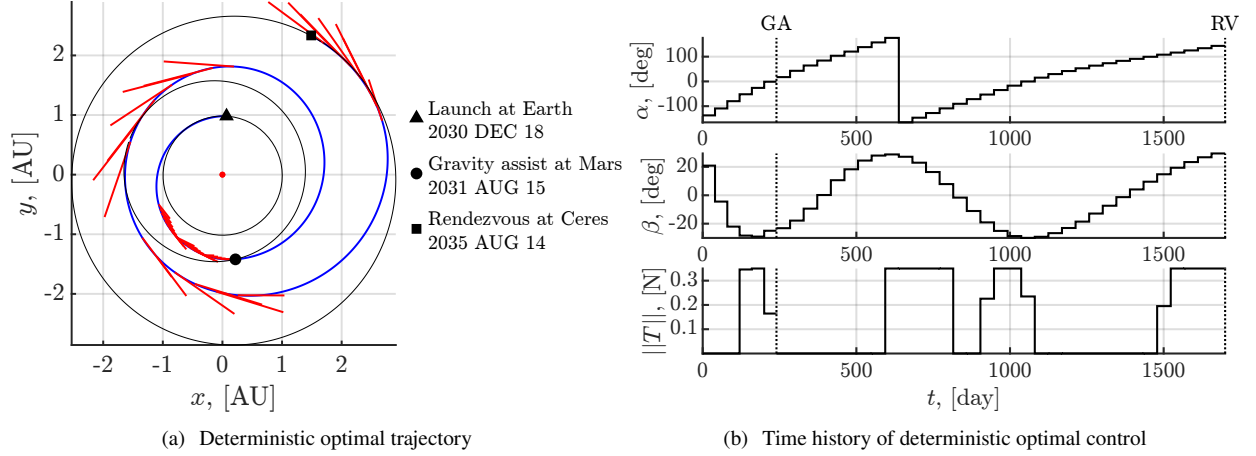
Figure 4   Deterministic optimal Earth-Mars-Ceres low-thrust transfer

Table 3   Comparison of deterministic-optimal and statistically-optimal solutions

	Launch: $\{v_{\infty,0}, \text{RA}, \text{DEC}\}$	MGA altitude: {nominal, min}	Total DV: {nominal, DV99}
Deterministic	{3.5 km/s, -145.3 deg, 37.3 deg}	{300 km, N/A}	{6.74 km/s, N/A}
Statistical	{3.5 km/s, -144.7, 36.0 deg}	{417 km, 368 km}	{6.81 km/s, 7.09 km/s}

The Mars arrival is modeled as rendezvous, i.e., the spacecraft position and velocity must match those of Mars at the arrival epoch.

B. Deterministically-optimal Solution

The transfer problem discussed above is first solved without considering uncertainties to generate the initial guess for the proposed method. We solve the deterministic trajectory optimization problem using the SCP method presented in Ref. [44] with $s_k = t_f - t_0 \ \forall k$, i.e., no time discretization mesh adaptation.

Fig. 4 summarizes the deterministic optimal solution. Fig. 4(a) shows the transfer trajectory projected onto the x-y plane, where the origin is the Sun; Fig. 4(b) shows the optimal control profile, where α, β are the inertial longitude and latitude of the thrust direction, calculated as: $\alpha = \arctan 2(u_2/u_1)$ and $\beta = \arcsin(u_3)$; $\arctan 2(\cdot)$ is the four-quadrant inverse tangent function. $\|T\|$ is the thrust magnitude in Newton. Some statistics of the deterministic optimal solution are summarized in Table 3, which indicates that the optimal trajectories fully leverages MGA (because $r_p = r_{p\ \min}$) and the launch capability ($v_{\infty,0} = \|v_0 - v_{\text{Earth}}(t_0)\|_2 = v_{\infty,\max}$), with DV requirement 6.74 km/s; RA and DEC in the table are the right ascension and declination of the launch in the inertial frame. This table also includes the data for statistically-optimal solution, which is discussed in the next subsection.

C. Statistically-optimal Solution

Now, we apply the proposed method to the same scenario while considering operational uncertainties and ensuring the robustness, including the statistical GA impact constraint and arrival dispersion constraint, minimizing DV99.

1. Operational Uncertainty Modeling

We consider various operational uncertainties, including launch dispersion, navigation error, maneuver execution error, and stochastic force. These uncertainties are modeled as:

$$P_0 = \texttt{blkdiag}(\sigma_{r0}^2 I_3, \sigma_{v0}^2 I_3), \quad G_{\text{obs,phase}} = \texttt{blkdiag}(\sigma_{r,\text{phase}}^{\text{nav}} I_3, \sigma_{v,\text{phase}}^{\text{nav}} I_3), \quad G = \sigma_{\text{acc}} \sqrt{\Delta t_{\text{WN}}} \cdot I_3 \qquad (59)$$

where $\texttt{blkdiag}(\cdot)$ forms a block diagonal matrices. Table 4 summarizes specific values of the standard deviations of these uncertain quantities assumed for this numerical example. Δt_{WN} denotes the time discretization step for approximation of SDE with white noise (see, for instance, Eq. 64 of [5]); this roughly corresponds to experiencing zero-mean acceleration with standard deviation σ_{acc} with update interval Δt_{WN}. We take $\Delta t_{\text{WN}} = 1$ hour. Note that $(\sigma_{\text{acc}} =)1.0 \ \mu\text{m/s}^2$ is slightly greater than the solar radiation pressure acceleration Psyche spacecraft might experience at 1 AU (based on the spacecraft property in [62]).

In Eq. (59), the observation is modeled as a result of orbit determination processes over Δt_k and hence G_{obs} takes the form of full-state measurements, where the navigation solutions are delivered about every 45 days ($\approx \Delta t_k$). However, the formulation given in Sections III and IV can accommodate any partial-state measurements, such as range, range-rate, Doppler shift, optical navigation (OpNav), among others; in such cases, we can consider having some nodes where measurements are taken but not control is applied, to have more frequent measurements for realism. For instance, [30] considers horizon-based OpNav measurements, where OpNav measurements are taken every 0.8 days while (impulsive) control is applied every 2.3 days.

Also note that "phase" in the subscript of G_{obs} corresponds to one of the phases: Launch, Cruise, MGA, and Arrival, to model different levels of navigation uncertainties we may expect at different phases of the mission. These values are summarized in Table 5, where the navigation error immediately after launch is greater than that of the cruise phase, while those immediately before MGA and Ceres arrival are smaller since we can expect more frequent orbit determination campaigns before these critical events; also, OpNav information would likely become available as the spacecraft approaches Mars or Ceres.

27

Table 4   Operational uncertainty model. Values represent the standard deviations.

	Launch dispersion		Navigation error (cruise)		Execution error (Gates)		Stoch. accel.
	position	velocity	position	velocity	magnitude	pointing	
Symbol	σ_{r0}	σ_{v0}	$\sigma_{r,\text{cruise}}^{\text{nav}}$	$\sigma_{v,\text{cruise}}^{\text{nav}}$	σ_2	σ_4	σ_{acc}
Value	1.0×10^5 km	10 m/s	30 km	0.3 m/s	0.5%	0.5 deg	1.0 μm/s^2

Table 5   Navigation error standard deviations at different phases (position and velocity)

Cruise	Launch	MGA	Arrival
$\{\sigma_{r,\text{cruise}}^{\text{nav}}, \sigma_{v,\text{cruise}}^{\text{nav}}\}$	$\{3\sigma_{r,\text{cruise}}^{\text{nav}}, 3\sigma_{v,\text{cruise}}^{\text{nav}}\}$	$\{1/3 \cdot \sigma_{r,\text{cruise}}^{\text{nav}}, 1/3 \cdot \sigma_{v,\text{cruise}}^{\text{nav}}\}$	$\{0.1\sigma_{r,\text{cruise}}^{\text{nav}}, 0.1\sigma_{v,\text{cruise}}^{\text{nav}}\}$

2. Statistical Constraints

We impose chance constraints on the control magnitude and on the GA pariapsis radius and terminal distribution constraint. The following parameters for these statistical constraints are assumed in the numerical example:

$$P_f = \texttt{blkdiag}(\sigma_{rf}^2 I_3, \sigma_{vf}^2 I_3), \quad \sigma_{rf} = 10^5 \text{ km}, \quad \sigma_{vf} = 100 \text{ m/s}, \quad \varepsilon_u = \varepsilon_{\text{GA}} = 10^{-3} \tag{60}$$

3. Optimization Result

The developed solution method is applied to solve the problem, which converged in 69 iterations of convex programming and took around 1.5 hours. As pointed out in Section I, a main bottleneck of the presented approach is the computational complexity, and some recent studies based on full covariance matrix [11, 14] are found more computationally efficient at the cost of some minor drawbacks (see Section I for more discussion).

Fig. 5 shows the optimization result, with Fig. 6 illustrating the nominal trajectory and control directions projected on the x-y plane, while Fig. 5(b) depicting the nominal control history (solid line) with the bottom plot overlaying the 99.9% control magnitude bound under statistical FPC (dotted line). First, it is evident that the control profile in Fig. 5(b) is largely different from Fig. 4(b). In particular, we can see that the nominal thrust level is reduced at immediate before the MGA and arrival at Ceres, implying that margins are automatically introduced to the nominal control to accommodate necessary statistical FPC activities. Conceptually, these margins are similar to heuristic margins quantity called *forced coasting period* and *duty cycle* [1], which are introduced manually at sensitive locations on low-thrust trajectories. The significance here is that this kind of margin quantities are discovered by the solution method with no specific inputs from mission designers, enabled by the stochastic optimal control approach that integrates navigation and FPC processes into mission design.
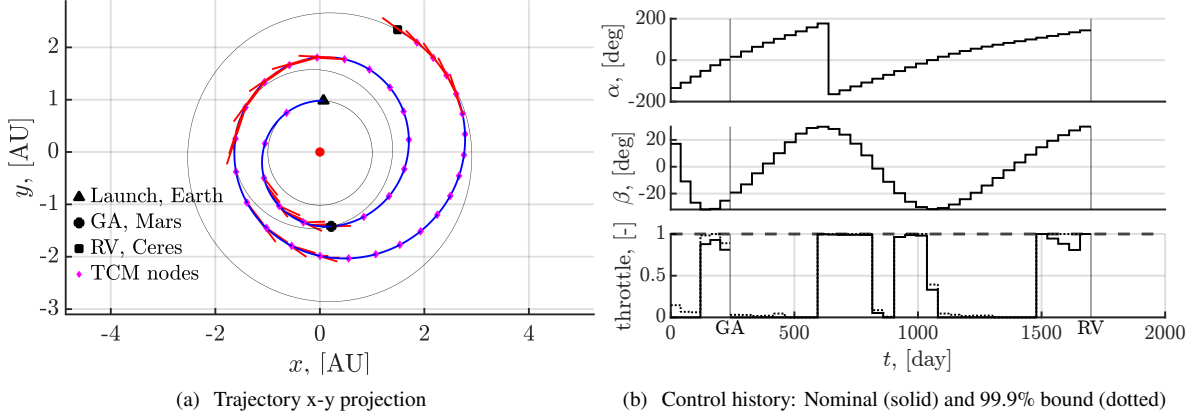
(a) Trajectory x-y projection

(b) Control history: Nominal (solid) and 99.9% bound (dotted)

Figure 5   Statistically-optimal Earth-Mars-Ceres low-thrust transfer solution

4. Monte Carlo Simulation

To confirm the robustness of the designed reference trajectory and associated FPC policies, nonlinear Monte Carlo (MC) simulations are performed with 100 samples. Recall Fig. 3 for the flow of nonlinear MC simulations.

Fig. 6 summarizes the MC trajectory results. Fig. 6(a) shows the MC trajectories, projected on the x-y plane, where the deviation from the reference trajectory is amplified by 20 times for visualization purpose. The red ellipses overlaid represent the covariance computed inside the proposed solution method (and used for calculation of DV99 and statistical constraints), highlighting the overall accuracy of UQ results performed within the optimization. The black ellipse in the zoomed view represents the covariance of the distributional constraint in Eq. (14), which confirms the successful arrival at Ceres with the prescribed accuracy, albeit some samples are not within the ellipse, indicating the effect of nonlinear dynamics. Fig. 6(b) includes the same trajectories but projected in x-y, y-z, and x-z planes, further demonstrating the accuracy of LinCov UQ in this case. Fig. 6(c) shows the state component time profiles of the MC simulations, which illustrates: 1) state dispersion is kept relatively small until MGA, 2) state dispersion significantly grows around the middle of the trajectory, which, as clear from Fig. 5(b), corresponds to the period where the optimized FPC does not apply much statistical corrections, 3) position dispersion rapidly decreases as the spacecraft approaches Ceres.

Fig. 7 display MC results in terms of the executed control profiles, which incorporates FPC and Gates execution error [Fig. 7(a)], and the navigation error time profiles based on EKF [Fig. 7(b)]. Fig. 7(a) clearly demonstrates that the maximum thrust constraint is satisfied under the action of statistical FPC, as expected from the imposed chance constraint. Fig. 7(b) shows the navigation errors and three-sigma covariance bounds over time, with the thrusting arcs overlaid as gray regions, highlighting that all the navigation solutions are successfully within the three-sigma bounds. The change in the navigation error levels reflects the different navigation uncertainties defined in Table 5; also, the velocity navigation error is greater on thrusting arcs as expected.

Fig. 8 complies some important statistics of the MC results. Fig. 8(a) shows the distribution of the periapsis radius

29

(a) MC trajectories projected on x-y plane.

(b) MC trajectories in three projections (not to scale).

(c) State deviation: MC samples (gray) and 99.9% confidence (black)
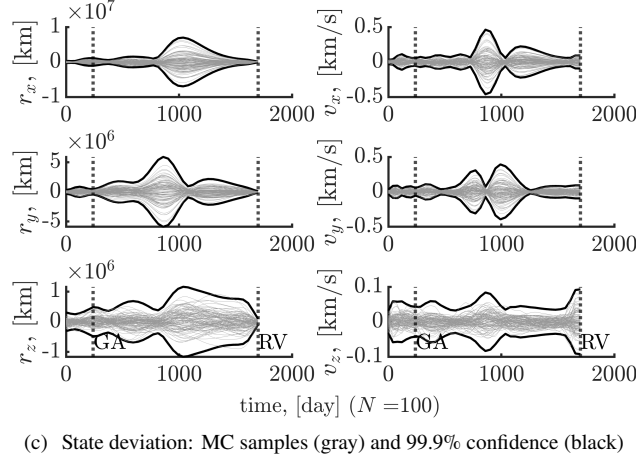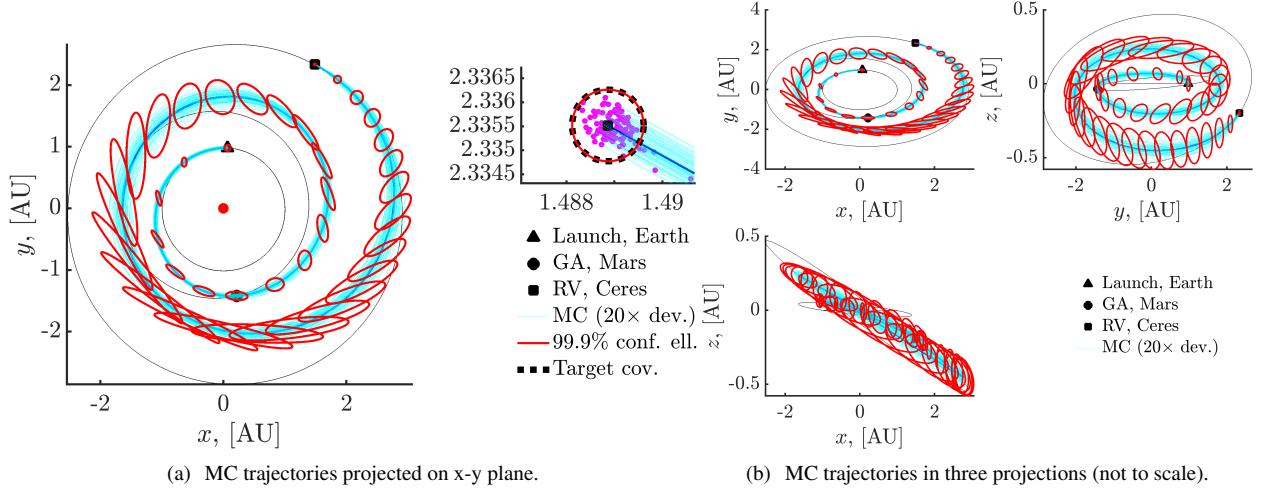
Figure 6   Monte Carlo simulation results. Curves in (a) and (b) represent: nominal (blue), MC samples (light blue), and 99.9% confidence ellipses (red).

at the Mars GA and demonstrates the satisfaction of the minimum MGA periapsis constraint under uncertainties thanks to the robust statistical FPC policies. Fig. 8(b) shows the distribution of the total DV, with the nominal DV, DV99 calculated from MC samples, and the upper bound of DV99, i.e., J_{ub} in Eq. (28), which is minimized within the presented algorithm. This figure shows that this bound is indeed upper-bounding the DV99 cost, albeit with some conservativeness; exploring different formulations that have smaller conservativeness is an interesting direction of future work.

D. Discussion

The numerical result demonstrates the effectiveness of the proposed method in designing statistically optimal, robust trajectories under state uncertainties due to launch dispersion, maneuver execution error, navigation error, and imperfect dynamics modeling. The designed trajectory satisfies critical constraints under uncertainties while minimizing the
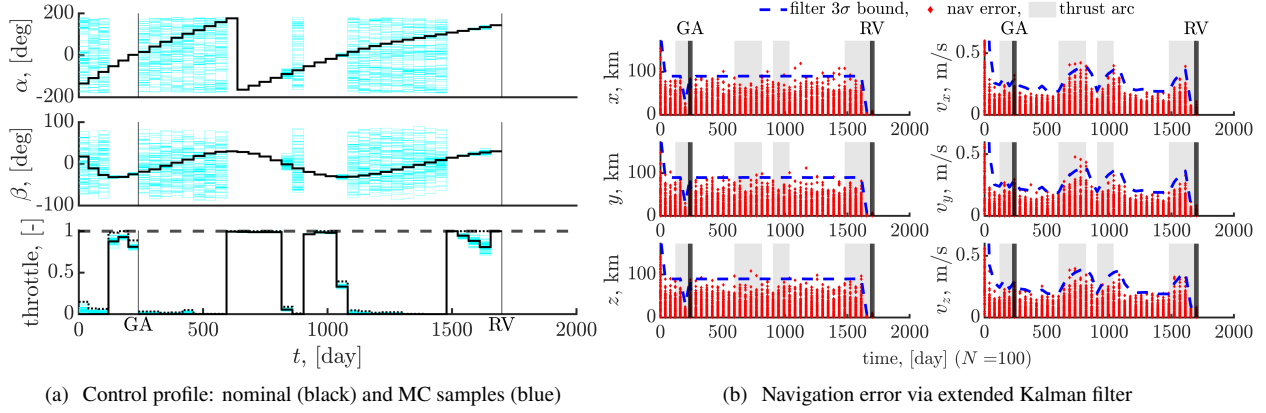
(a) Control profile: nominal (black) and MC samples (blue)

(b) Navigation error via extended Kalman filter

Figure 7   Monte Carlo simulation results: Control profile and navigation errors.

(a) MGA periapsis distribution and statistics
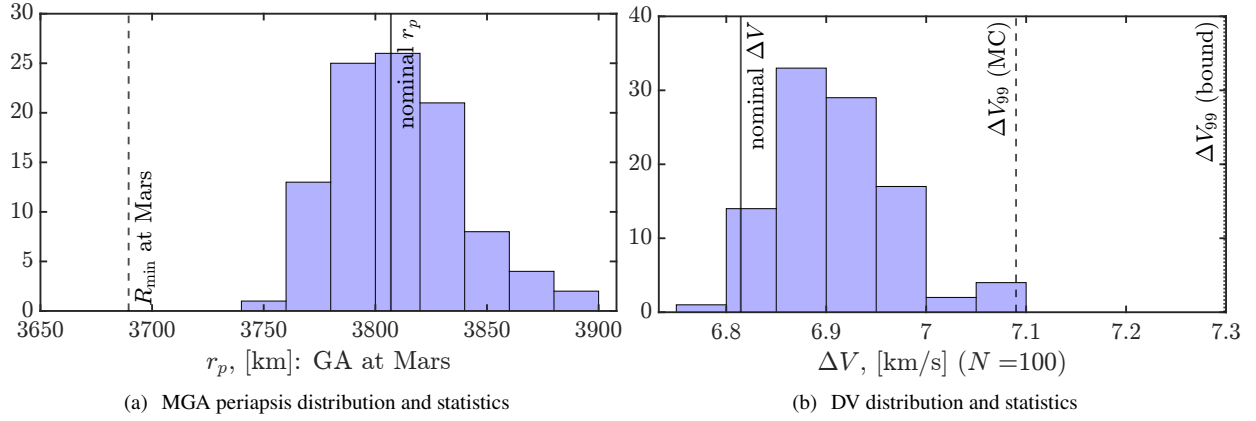
(b) DV distribution and statistics

Figure 8   Monte Carlo simulation results: Statistics (y-axis: number of samples)

statistical cost, DV99 (its upper bound, to be precise). The formulation can naturally incorporate gravity assists (GAs), which underlines the flexibility of the proposed framework.

Table 3 compares the statistics of the deterministically and statistically optimal trajectories, which provides a few important observations. First, the statistically optimal trajectory has a greater MGA altitude than the deterministic optimal counterpart, which represents a safe MGA margin automatically (and optimally) derived within the algorithmic framework of the proposed method. This shift in the nominal MGA altitude naturally introduces a less effective MGA, which is compensated by the additional nominal DV and largely altered thrust profiles (compare Figs. 4(b) and 5(b)). This change in the nominal trajectory is also accompanied by the slight change in the optimal launch direction, about 0.5-1 degree change in the right ascension and declination, suggesting explicit incorporation of state uncertainty into mission design may affect the optimal launch condition. In a similar vein, it is anticipated that the optimal epochs of critical events (e.g., launch, GA, arrival) would be also different for statistically and deterministically optimal trajectories; incorporating time-varying events into the proposed SCP framework is straightforward by utilizing an adaptive-mesh

SCP algorithm [44].

There is one important caveat about the GA formulation presented in this paper. Upon closer look at Fig. 6(c), acute readers may wonder why the position dispersion at MGA is not reduced as much as we may intuitively expect; one plausible reason for that is due to the simple (patched-conic) MGA modeling. As it can be seen in Eq. (38), the effect of a GA is modeled as a rotation of the velocity relative to the body, where the rotation is parameterized by $\boldsymbol{u}_k$. It is an important future work to consider a higher-fidelity GA model (e.g., treating a GA as a finite-duration phase centered around the body [48]) to capture the nonlinear effect more accurately, which may benefit from non-Gaussian UQ depending on the level of state uncertainty at the time of GA.

Another important extension of this line of research is its application to solar-sail trajectory design. Solar-sail missions are more likely to suffer from state uncertainties due to the unpredictable nature of the thrust forces caused by the interaction between the sunlight and deformed, possibly time-varying, sail surface [63]. Ref. [45] is one of the early studies that applies a statistical approach to robust solar-sail trajectory design, which however did not demonstrate the full capability of statistical approaches, since it had to impose artificial constraints on the admissible feedback control due to the nonlinear reliance of solar-sail acceleration on the attitude variables. An exciting avenue here is to combine a statistical approach with the recent work on a lossless control-convex formulation of solar-sail trajectory optimization problem [49] to design statistically optimal, robust solar-sail trajectories under severe uncertainties.

VII. Conclusions

A mission design framework for robust, statistically optimal trajectory design under uncertainty is presented and numerically demonstrated. The developed framework leverages recent advances in stochastic optimal control and sequential convex programming to formulate the otherwise intractable problem into a sequence of convex problems. The framework enables mission designers to concurrently design a reference trajectory and flight-path control plans such that minimize dV99 while ensuring the satisfaction of mission constraints (e.g., planetary rendezvous with certain accuracy, no collision with a planet during a flyby, and more) under uncertainty due to launch dispersion, navigation error, execution error, and stochastic disturbance. The proposed framework is successfully applied to an Earth-Mars-Ceres transfer with an intermediate gravity assist at Mars. Nonlinear Monte Carlo simulations demonstrate the robustness of the designed reference trajectory and flight-path control under the original, nonlinear stochastic dynamics and uncertain errors.

Acknowledgments

Appendix

A. Proof of Lemma 1: Nonlinear and Linear State Mapping under Gravity Assist

Proof. Deriving Eq. (38) is straightforward by combining Eqs. (34) and (37) and noting that $[\boldsymbol{u}_k]_\times = V$.

To derive Eq. (39), let us begin with expressing Eq. (37) as $\boldsymbol{v}_{k+1} - \boldsymbol{v}_p = (I_3 + [\boldsymbol{u}_k]_\times)^{-1}(I_3 - [\boldsymbol{u}_k]_\times)(\boldsymbol{v}_k - \boldsymbol{v}_p)$, which is equivalent to imposing $\boldsymbol{c}_{\text{GA},v}(\boldsymbol{v}_{k+1}, \boldsymbol{v}_k, \boldsymbol{u}_k) = 0$, where

$$\boldsymbol{c}_{\text{GA},v}(\boldsymbol{v}_{k+1}, \boldsymbol{v}_k, \boldsymbol{u}_k) = (I_3 + [\boldsymbol{u}_k]_\times)(\boldsymbol{v}_{k+1} - \boldsymbol{v}_p) - (I_3 - [\boldsymbol{u}_k]_\times)(\boldsymbol{v}_k - \boldsymbol{v}_p) \tag{61}$$

Using the fact that $[\boldsymbol{u}_k]_\times \boldsymbol{a} = \boldsymbol{u}_k \times \boldsymbol{a} = -\boldsymbol{a} \times \boldsymbol{u}_k = -[\boldsymbol{a}]_\times \boldsymbol{u}_k$ for any vector $\boldsymbol{a} \in \mathbb{R}^3$, Eq. (61) is equivalent to

$$\boldsymbol{c}_{\text{GA},v}(\boldsymbol{v}_{k+1}, \boldsymbol{v}_k, \boldsymbol{u}_k) = (\boldsymbol{v}_{k+1} - \boldsymbol{v}_k) - ([\boldsymbol{v}_{k+1}]_\times + [\boldsymbol{v}_k]_\times - 2[\boldsymbol{v}_p]_\times)\boldsymbol{u}_k \tag{62}$$

Thus, its Taylor series expansion about $\boldsymbol{v}_{k+1}^*, \boldsymbol{v}_k^*, \boldsymbol{u}_k^*$ is given by $[(\cdot)^*$ indicates evaluation at the reference point]:

$$\boldsymbol{c}_{\text{GA},v}(\boldsymbol{v}_{k+1}, \boldsymbol{v}_k, \boldsymbol{u}_k) = \boldsymbol{c}_{\text{GA},v}^* + \left.\frac{\partial \boldsymbol{c}_{\text{GA},v}}{\partial \boldsymbol{v}_{k+1}}\right|^* (\boldsymbol{v}_{k+1} - \boldsymbol{v}_{k+1}^*) + \left.\frac{\partial \boldsymbol{c}_{\text{GA},v}}{\partial \boldsymbol{v}_k}\right|^* (\boldsymbol{v}_k - \boldsymbol{v}_k^*) + \left.\frac{\partial \boldsymbol{c}_{\text{GA},v}}{\partial \boldsymbol{u}_k}\right|^* (\boldsymbol{u}_k - \boldsymbol{u}_k^*) + H.O.T. \tag{63}$$

where

$$\left.\frac{\partial \boldsymbol{c}_{\text{GA},v}}{\partial \boldsymbol{v}_{k+1}}\right|^* = I_3 + [\boldsymbol{u}_k^*]_\times, \quad \left.\frac{\partial \boldsymbol{c}_{\text{GA},v}}{\partial \boldsymbol{v}_k}\right|^* = -(I_3 - [\boldsymbol{u}_k^*]_\times), \quad \left.\frac{\partial \boldsymbol{c}_{\text{GA},v}}{\partial \boldsymbol{u}_k}\right|^* = -([\boldsymbol{v}_{k+1}^*]_\times + [\boldsymbol{v}_k^*]_\times - 2[\boldsymbol{v}_p]_\times) \tag{64}$$

Rearranging the term, we obtain

$$\boldsymbol{c}_{\text{GA},v}(\boldsymbol{v}_{k+1}, \boldsymbol{v}_k, \boldsymbol{u}_k) = (I_3 + [\boldsymbol{u}_k^*]_\times)\boldsymbol{v}_{k+1} - (I_3 - [\boldsymbol{u}_k^*]_\times)\boldsymbol{v}_k - ([\boldsymbol{v}_{k+1}^*]_\times + [\boldsymbol{v}_k^*]_\times - 2[\boldsymbol{v}_p]_\times)\boldsymbol{u}_k + H.O.T. \tag{65}$$

Thus, to the first order, $\boldsymbol{c}_{\text{GA},v}(\boldsymbol{v}_{k+1}, \boldsymbol{v}_k, \boldsymbol{u}_k) = 0$ is equivalent to

$$(I_3 + [\boldsymbol{u}_k^*]_\times)\boldsymbol{v}_{k+1} = (I_3 - [\boldsymbol{u}_k^*]_\times)\boldsymbol{v}_k + ([\boldsymbol{v}_{k+1}^*]_\times + [\boldsymbol{v}_k^*]_\times - 2[\boldsymbol{v}_p]_\times)\boldsymbol{u}_k \iff$$
$$\boldsymbol{v}_{k+1} = (I_3 + [\boldsymbol{u}_k^*]_\times)^{-1}(I_3 - [\boldsymbol{u}_k^*]_\times)\boldsymbol{v}_k + (I_3 + [\boldsymbol{u}_k^*]_\times)^{-1}([\boldsymbol{v}_{k+1}^*]_\times + [\boldsymbol{v}_k^*]_\times - 2[\boldsymbol{v}_p]_\times)\boldsymbol{u}_k \tag{66}$$

which, together with $\boldsymbol{r}_{k+1} = \boldsymbol{r}_k$, leads to $A_k, B_k, \boldsymbol{c}_k$ given above. $\qquad\square$

B. Proof of Lemma 2: State-estimate Feedback Control Policy Conversion

Proof. We show that the two control policies Eq. (20) and Eq. (56) give the same filtered state trajectory.

First, recall from Eq. (57) that $\hat{\mathbf{K}} = \mathbf{K}(I + \mathbf{BK})^{-1}$, where $(I + \mathbf{BK})$ is invertible since $\mathbf{B}$ is strictly block lower-triangular (see Eq. (22)) and $\mathbf{K}$ is block lower-triangular (see Ref. [30]). Using Eq. 41 to 43 of Ref. [18] implies that $\mathbf{K}$ and $\hat{\mathbf{K}}$

satisfy

$$(I - \mathbf{B}\hat{\mathbf{K}})^{-1} = (I + \mathbf{B}\mathbf{K}) \tag{67}$$

where note that $\mathbf{K}$ and $\hat{\mathbf{K}}$ here correspond to F and K in Ref. [18], respectively. Also, since $\mathbf{B}$ is strictly block lower-triangular and $\mathbf{K}$ is block lower-triangular (see Eq. (57)), $(I - \mathbf{B}\hat{\mathbf{K}})$ is also invertible.

Then, under the policy Eq. (20), the filtered state trajectory is given as (see Eq. 45 of Ref. [30]):

$$\hat{X} - \overline{X} = (I + \mathbf{B}\mathbf{K})[\mathbf{A}(\hat{x}_0^- - \bar{x}_0) + \mathbf{L}Y] \tag{68}$$

On the other hand, under the policy Eq. (56), $(\hat{X} - \overline{X})$ is given as (see Eq. 40 of Ref. [18]; notation is adapted for consistency):

$$\hat{X} - \overline{X} = (I - \mathbf{B}\hat{\mathbf{K}})^{-1}[\mathbf{A}(\hat{x}_0^- - \bar{x}_0) + \mathbf{L}Y] \tag{69}$$

Eq. (67) implies that Eqs. (68) and (69) represent the same filtered state trajectory, completing the proof. $\qquad\square$

References

[1] Rayman, M. D., Fraschetti, T. C., Raymond, C. A., and Russell, C. T., "Coupling of System Resource Margins through the Use of Electric Propulsion: Implications in Preparing for the Dawn Mission to Ceres and Vesta," *Acta Astronautica*, Vol. 60, No. 10-11, 2007, pp. 930–938. https://doi.org/10.1016/j.actaastro.2006.11.012.

[2] Oh, D., Landau, D., Randolph, T., Timmerman, P., Chase, J., Sims, J., and Kowalkowski, T., "Analysis of System Margins on Deep Space Missions Using Solar Electric Propulsion," *44th AIAA/ASME/SAE/ASEE Joint Propulsion Conference and Exhibit*, American Institute of Aeronautics and Astronautics, Reston, Virigina, 2008, pp. 1–30. https://doi.org/10.2514/6.2008-5286.

[3] Zimmer, S., Ocampo, C., and Bishop, R., "Reducing Orbit Covariance for Continuous Thrust Spacecraft Transfers," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 46, No. 2, 2010, pp. 771–791. https://doi.org/10.1109/TAES.2010.5461656.

[4] Jenson, E. L., and Scheeres, D. J., "Multi-Objective Optimization of Covariance and Energy for Asteroid Transfers," *Journal of Guidance, Control, and Dynamics*, Vol. 44, No. 7, 2021, pp. 1253–1265. https://doi.org/10.2514/1.G005609.

[5] Oguri, K., and McMahon, J. W., "Stochastic Primer Vector for Robust Low-Thrust Trajectory Design Under Uncertainty," *Journal of Guidance, Control, and Dynamics*, Vol. 45, No. 1, 2022, pp. 84–102. https://doi.org/10.2514/1.G005970.

[6] Greco, C., Campagnola, S., and Vasile, M., "Robust Space Trajectory Design Using Belief Optimal Control," *Journal of Guidance, Control, and Dynamics*, 2022, pp. 1–18. https://doi.org/10.2514/1.G005704.

[7] Ridderhof, J., Pilipovsky, J., and Tsiotras, P., "Chance-Constrained Covariance Control for Low-Thrust Minimum-Fuel Trajectory Optimization," *AAS/AIAA Astrodynamics Specialist Conference*, AAS, South Lake Tahoe, CA (Virtual), 2020.

[8] Oguri, K., and Lantoine, G., "Stochastic Sequential Convex Programming for Robust Low-thrust Trajectory Design under Uncertainty," *AAS/AIAA Astrodynamics Specialist Conference*, Charlotte, NC, 2022.

[9] Benedikter, B., Zavoli, A., Wang, Z., Pizzurro, S., and Cavallini, E., "Convex Approach to Covariance Control with Application to Stochastic Low-Thrust Trajectory Optimization," *Journal of Guidance, Control, and Dynamics*, 2022, pp. 1–16. https://doi.org/10.2514/1.G006806.

[10] Varghese, J., Oguri, K., Wittick, P., and Doogan, T., "Nonlinear Programming Approach to Trajectory Optimization under Uncertainty: Direct Forward-Backward Shooting Formulation," *AAS/AIAA Space Flight Mechanics Meeting*, AAS, Kaua'i, HI, 2025.

[11] Kumagai, N., and Oguri, K., "Robust Cislunar Low-Thrust Trajectory Optimization under Uncertainties via Sequential Covariance Steering," *Journal of Guidance, Control, and Dynamics (under review)*, 2025.

[12] Ozaki, N., Campagnola, S., and Funase, R., "Tube Stochastic Optimal Control for Nonlinear Constrained Trajectory Optimization Problems," *Journal of Guidance, Control, and Dynamics*, Vol. 43, No. 4, 2020, pp. 645–655. https://doi.org/10.2514/1.G004363.

[13] Yuan, H., Li, D., He, G., and Wang, J., "Uncertainty-Resilient Constrained Rendezvous Trajectory Optimization via Stochastic Feedback Control and Unscented Transformation," *Acta Astronautica*, Vol. 219, 2024, pp. 264–277. https://doi.org/10.1016/j.actaastro.2024.03.017.

[14] Kumagai, N., and Oguri, K., "Sequential Chance-Constrained Covariance Steering for Robust Cislunar Trajectory Design under Uncertainties," *AAS/AIAA Astrodynamics Specialist Conference*, Broomfield, CO, 2024.

[15] Okamoto, K., Goldshtein, M., and Tsiotras, P., "Optimal Covariance Control for Stochastic Systems Under Chance Constraints," *IEEE Control Systems Letters*, Vol. 2, No. 2, 2018, pp. 266–271. https://doi.org/10.1109/LCSYS.2018.2826038.

[16] Chen, Y., Georgiou, T. T., and Pavon, M., "Optimal Steering of a Linear Stochastic System to a Final Probability Distribution - Part I," *IEEE Transactions on Automatic Control*, Vol. 63, No. 9, 2018, pp. 3112–3118. https://doi.org/10.1109/TAC.2018.2791362.

[17] 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. https://doi.org/10.1109/LRA.2019.2901546.

[18] Ridderhof, J., Okamoto, K., and Tsiotras, P., "Chance Constrained Covariance Control for Linear Stochastic Systems With Output Feedback," *IEEE Conference on Decision and Control (CDC)*, IEEE, Nice, France, 2020, pp. 1758–1763. https://doi.org/10.1109/CDC42340.2020.9303731.

[19] Aleti, D., Oguri, K., and Kumagai, N., "Chance-Constrained Output-Feedback Control without History Feedback: Application to NRHO Stationkeeping," *AAS/AIAA Astrodynamics Specialist Conference*, Big Sky, MT, 2023.

[20] Liu, F., Rapakoulias, G., and Tsiotras, P., "Optimal Covariance Steering for Discrete-Time Linear Stochastic Systems," *IEEE Transactions on Automatic Control*, 2024, pp. 1–16. https://doi.org/10.1109/TAC.2024.3472788.

[21] Pilipovsky, J., and Tsiotras, P., "Computationally Efficient Chance Constrained Covariance Control with Output Feedback," , Mar. 2024. https://doi.org/10.48550/arXiv.2310.02485.

[22] Açıkmeşe, 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.

[23] Açıkmeşe, B., and Blackmore, L., "Lossless Convexification of a Class of Optimal Control Problems with Non-Convex Constraints," *Automatica*, Vol. 47, No. 2, 2011, pp. 341–347. https://doi.org/10.1016/j.automatica.2010.10.037.

[24] Mao, Y., Szmuk, M., and Acikmese, B., "Successive Convexification of Non-Convex Optimal Control Problems and Its Convergence Properties," *IEEE Conference on Decision and Control (CDC)*, IEEE, 2016, pp. 3636–3641. https://doi.org/10.1109/CDC.2016.7798816.

[25] Bonalli, R., Lew, T., and Pavone, M., "Analysis of Theoretical and Numerical Properties of Sequential Convex Programming for Continuous-Time Optimal Control," 2020, pp. 1–16.

[26] Oguri, K., "Successive Convexification with Feasibility Guarantee via Augmented Lagrangian for Non-Convex Optimal Control Problems," *IEEE Conference on Decision and Control (CDC)*, IEEE, Singapore, Singapore, 2023, pp. 3296–3302. https://doi.org/10.1109/CDC49753.2023.10383462.

[27] Lu, P., and Liu, X., "Autonomous Trajectory Planning for Rendezvous and Proximity Operations by Conic Optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 36, No. 2, 2013, pp. 375–389. https://doi.org/10.2514/1.58436.

[28] Szmuk, M., Reynolds, T. P., and Açıkmeşe, B., "Successive Convexification for Real-Time Six-Degree-of-Freedom Powered Descent Guidance with State-Triggered Constraints," *Journal of Guidance, Control, and Dynamics*, Vol. 43, No. 8, 2020, pp. 1399–1413. https://doi.org/10.2514/1.G004549.

[29] Oguri, K., and McMahon, J. W., "Robust Spacecraft Guidance Around Small Bodies Under Uncertainty: Stochastic Optimal Control Approach," *Journal of Guidance, Control, and Dynamics*, Vol. 44, No. 7, 2021, pp. 1295–1313. https://doi.org/10.2514/1.G005426.

[30] Oguri, K., "Chance-Constrained Control for Safe Spacecraft Autonomy: Convex Programming Approach," *2024 American Control Conference (ACC)*, 2024, pp. 2318–2324. https://doi.org/10.23919/ACC60939.2024.10645008.

[31] Ra, M. A. P., and Oguri, K., "Chance-Constrained Sensing-Optimal Path Planning for Safe Angles-only Autonomous Navigation," *AAS/AIAA Astrodynamics Specialist Conference*, Broomfield, CO, 2024.

[32] Sagliano, M., Seelbinder, D., Theil, S., and Lu, P., "Six-Degree-of-Freedom Rocket Landing Optimization via Augmented Convex–Concave Decomposition," *Journal of Guidance, Control, and Dynamics*, Vol. 47, No. 1, 2024, pp. 20–35. https://doi.org/10.2514/1.G007570.

[33] Hotz, A., and Skelton, R. E., "Covariance Control Theory," *International Journal of Control*, Vol. 46, No. 1, 1987, pp. 13–32. https://doi.org/10.1080/00207178708933880.

[34] Bakolas, E., "Finite-Horizon Covariance Control for Discrete-Time Stochastic Linear Systems Subject to Input Constraints," *Automatica*, Vol. 91, 2018, pp. 61–68. https://doi.org/10.1016/j.automatica.2018.01.029.

[35] Geeraert, J. L., and McMahon, J. W., "Square-Root Unscented Schmidt–Kalman Filter," *Journal of Guidance, Control, and Dynamics*, Vol. 5090, 2017, pp. 1–8. https://doi.org/10.2514/1.G002921.

[36] Tapley, B. D., Schutz, B. E., and Born, G. H., *Statistical Orbit Determination*, Elsevier, Amsterdam, Netherlands, 2004. https://doi.org/10.1016/B978-0-12-683630-1.X5019-X.

[37] Rapakoulias, G., and Tsiotras, P., "Comment on "Convex Approach to Covariance Control with Application to Stochastic Low-Thrust Trajectory Optimization"," *Journal of Guidance, Control, and Dynamics*, 2023, pp. 1–2. https://doi.org/10.2514/1.G007420.

[38] Kumagai, N., and Oguri, K., "Robust Cislunar Low-Thrust Trajectory Optimization under Uncertainties via Sequential Covariance Steering," , Feb. 2025. https://doi.org/10.48550/arXiv.2502.01907.

[39] Zhang, Y., Cheng, M., Nan, B., and Li, S., "Stochastic Trajectory Optimization for 6-DOF Spacecraft Autonomous Rendezvous and Docking with Nonlinear Chance Constraints," *Acta Astronautica*, Vol. 208, 2023, pp. 62–73. https://doi.org/10.1016/j.actaastro.2023.04.004.

[40] Benedikter, B., Zavoli, A., Wang, Z., Pizzurro, S., and Cavallini, E., "Convex Approach to Covariance Control for Low-Thrust Trajectory Optimization with Mass Uncertainty," *AIAA SCITECH 2023 Forum*, American Institute of Aeronautics and Astronautics, National Harbor, MD & Online, 2023. https://doi.org/10.2514/6.2023-2321.

[41] Gates, C. R., "A Simplified Model of Midcourse Maneuver Execution Errors," Tech. rep., Jet Propulsion Laboratory, California Institute of Technology (Report No. 32-504), Pasadena, CA , Oct. 1963.

[42] Greco, C., Campagnola, S., and Vasile, M., "Robust Space Trajectory Design Using Belief Optimal Control," *Journal of Guidance, Control, and Dynamics*, Vol. 45, No. 6, 2022, pp. 1060–1077. https://doi.org/10.2514/1.G005704.

[43] Benedikter, B., Zavoli, A., Wang, Z., Pizzurro, S., and Cavallini, E., "Covariance Control for Stochastic Low-Thrust Trajectory Optimization," *AIAA SCITECH 2022 Forum*, American Institute of Aeronautics and Astronautics, Reston, Virginia, 2022, pp. 1–19. https://doi.org/10.2514/6.2022-2474.

[44] Kumagai, N., and Oguri, K., "Adaptive-Mesh Sequential Convex Programming for Space Trajectory Optimization," *Journal of Guidance, Control, and Dynamics*, 2024. https://doi.org/10.2514/1.G008107.

[45] Oguri, K., Lantoine, G., and Sweetser, T. H., "Robust Solar Sail Trajectory Design under Uncertainty with Application to NEA Scout Mission," *AIAA SCITECH Forum*, American Institute of Aeronautics and Astronautics, Reston, Virginia, 2022. https://doi.org/10.2514/6.2022-1627.

[46] Oguri, K., "Chance-Constrained Control for Safe Spacecraft Autonomy: Convex Programming Approach," *arXiv preprint*, 2024.

[47] Sims, J. A., "Delta-V Gravity-Assist Trajectory Design: Theory and Practice," Ph.D. thesis , 1996.

[48] Ellison, D. H., and Englander, J. A., "High-Fidelity Multiple-Flyby Trajectory Optimization Using Multiple Shooting," *AAS/AIAA Astrodynamics Specialist Conference*, Portland, ME, 2019.

[49] Oguri, K., and Lantoine, G., "Lossless Control-Convex Formulation for Solar-Sail Trajectory Optimization via Sequential Convex Programming," *Journal of Guidance, Control, and Dynamics*, 2024. https://doi.org/10.2514/1.G008361.

[50] Mao, Y., Szmuk, M., Xu, X., and Acikmese, B., "Successive Convexification: A Superlinearly Convergent Algorithm for Non-convex Optimal Control Problems," *arXiv preprint*, 2018, pp. 1–35.

[51] Bertsekas, D. P., *Constrained Optimization and Lagrange Multiplier Methods*, Elsevier, 1982. https://doi.org/10.1016/C2013-0-10366-2.

[52] Lantoine, G., and Russell, R. P., "A Hybrid Differential Dynamic Programming Algorithm for Constrained Optimal Control Problems. Part 1: Theory," *Journal of Optimization Theory and Applications*, Vol. 154, No. 2, 2012, pp. 382–417. https://doi.org/10.1007/s10957-012-0039-0.

[53] Han, S. P., and Mangasarian, O. L., "Exact Penalty Functions in Nonlinear Programming," *Mathematical Programming*, Vol. 17, No. 1, 1979, pp. 251–269. https://doi.org/10.1007/BF01588250.

[54] Mayne, D., and Polak, E., "An Exact Penalty Function Algorithm for Control Problems with State and Control Constraints," *IEEE Transactions on Automatic Control*, Vol. 32, No. 5, 1987, pp. 380–387. https://doi.org/10.1109/TAC.1987.1104615.

[55] Boyd, S., and Vandenberghe, L., *Convex Optimization*, Cambridge University Press, Cambridge, England, 2004. https://doi.org/10.1017/CBO9780511804441.

[56] Oguri, K., "Successive Convexification with Feasibility Guarantee via Augmented Lagrangian for Non-Convex Optimal Control Problems," *arXiv preprint*, 2023.

[57] Kumagai, N., and Oguri, K., "Chance-Constrained Gaussian Mixture Steering to a Terminal Gaussian Distribution," , Mar. 2024.

[58] Boone, S., and McMahon, J., "Spacecraft Maneuver Design with Non-Gaussian Chance Constraints Using Gaussian Mixtures," *AAS/AIAA Astrodynamics Specialist Conference*, 2022.

[59] Nakka, Y. K., and Chung, S.-J., "Trajectory Optimization for Chance-Constrained Nonlinear Stochastic Systems," *IEEE Conference on Decision and Control (CDC)*, IEEE, Nice, France, 2019, pp. 3811–3818. https://doi.org/10.1109/CDC40024.2019.9028893.

[60] Grant, M., and Boyd, S., "CVX: Matlab Software for Disciplined Convex Programming, Version 2.1," , Mar. 2014.

[61] Mosek ApS, "The MOSEK Optimization Toolbox for Matlab Manual, Version 8.1." http://docs.mosek.com/9.0/toolbox/index.html , 2017.

[62] Oguri, K., Lantoine, G., Hart, W., and McMahon, J., "Science Orbit Design with a Quasi-Frozen Beta Angle: Effects of Body Obliquity on J2-perturbed Dynamics," *Celestial Mechanics and Dynamical Astronomy*, Vol. 132, No. 10, 2020. https://doi.org/10.1007/s10569-020-09987-z.

[63] Dachwald, B., Macdonald, M., McInnes, C. R., Mengali, G., and Quarta, A. A., "Impact of Optical Degradation on Solar Sail Mission Performance," *Journal of Spacecraft and Rockets*, Vol. 44, No. 4, 2007, pp. 740–749. https://doi.org/10.2514/1.21432.