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

Robust Optimal Control for Autonomous Precision Landing via Set-based Dynamic Programming

Kamath, Abhinav; Vinod, Abraham P.; Elango, Purnanand; Di Cairano, Stefano; Weiss, Avishai

TR2026-015 January 14, 2026

Abstract

We present a real-time-capable set-based framework for closed-loop predictive control of autonomous systems using tools from computational geometry, dynamic programming, and convex optimization. The control architecture relies on the offline precomputation of the controllable tube, i.e, a time-indexed sequence of controllable sets, which are sets that contain all possible states that can reach a terminal set under state and control constraints. Sets are represented using constrained zonotopes (CZs), which are efficient encodings of convex polytopes that support fast set operations and enable tractable dynamic programming in high dimensions. Online, we obtain a globally optimal control profile via a forward rollout, i.e., by solving a series of one-step optimal control problems, each of which takes the current state to the next controllable set in the tube. Our key contributions are: (1) free-final-time optimality: we devise an optimal horizon computation algorithm to achieve global optimality, and (2) robustness: we handle stochastic uncertainty in both the state and control, with probabilistic guarantees, by constructing bounded disturbance sets. The optimal control approach we propose is exact (approximation-free) for optimal control problems with polytopic feasible sets, and conservative in the right direction for their robust variants. By means of an autonomous precision landing case study, we demonstrate globally optimal free-final-time guidance and robustness to navigation and actuation uncertainties.

AIAA SciTech Forum 2026

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

Robust Optimal Control for Autonomous Precision Landing via Set-based Dynamic Programming

Abhinav G. Kamath*,
University of Washington, Seattle, WA 98195, USA

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

We present a real-time-capable set-based framework for closed-loop predictive control of autonomous systems using tools from computational geometry, dynamic programming, and convex optimization. The control architecture relies on the offline precomputation of the controllable tube, i.e, a time-indexed sequence of controllable sets, which are sets that contain all possible states that can reach a terminal set under state and control constraints. Sets are represented using constrained zonotopes (CZs), which are efficient encodings of convex polytopes that support fast set operations and enable tractable dynamic programming in high dimensions. Online, we obtain a globally optimal control profile via a forward rollout, i.e., by solving a series of one-step optimal control problems, each of which takes the current state to the next controllable set in the tube. Our key contributions are: (1) free-final-time optimality: we devise an optimal horizon computation algorithm to achieve global optimality, and (2) robustness: we handle stochastic uncertainty in both the state and control, with probabilistic guarantees, by constructing bounded disturbance sets. The optimal control approach we propose is exact (approximation-free) for optimal control problems with polytopic feasible sets, and conservative in the right direction for their robust variants. By means of an autonomous precision landing case study, we demonstrate globally optimal free-final-time guidance and robustness to navigation and actuation uncertainties.

I. Introduction

Modern autonomous systems require advanced real-time-capable control methods. The control problem of autonomous precision landing is particularly challenging owing to payload capacity coming at a premium and estimation and control errors due to uncertainty in navigation and actuation [1, 2, 3, 4]. At the same time, the overarching goal of autonomous precision landing missions is to maximize the payload capacity by minimizing propellant consumption (optimal control), while remaining robust to navigation and actuation uncertainty (robust control). These competing requirements naturally lead to the need for a unified control architecture.

In this work, motivated by the autonomous precision landing problem, we develop a computationally tractable and real-time-capable approach to optimal and robust predictive control—for a class of systems—within a unified control architecture. Our approach is set-theoretic: we present general results for closed-loop optimal control and for control under modeled state and control uncertainty. We develop a computational framework for closed-loop control via dynamic programming over a controllable tube, i.e., a time-indexed sequence of controllable sets [5]. For computational tractability, we leverage constrained zonotopes, a representation of compact convex polytopes that admits efficient and scalable set operations [6].

Robust open-loop control with both state and control uncertainty can be prohibitively conservative, motivating the need for a closed-loop control framework [7]. Global optimality in a closed-loop setting is naturally

*Ph.D. Candidate, William E. Boeing Department of Aeronautics & Astronautics;
Research Intern (during the development of this work), Mitsubishi Electric Research Laboratories; agkamath@uw.edu
†Principal Research Scientist; abraham.p.vinod@ieee.org
‡Research Scientist; elango@merl.com
§Distinguished Research Scientist; dicairano@merl.com
¶Senior Principal Research Scientist; weiss@merl.com

characterized by dynamic programming [8]. Standard approaches to dynamic programming, however, require discretization of the state space and suffer from the curse of dimensionality, making them intractable for high-dimensional systems [9]. Recently, it was shown that for a class of nonlinear optimal control problems, dynamic programming can be made tractable by lossless convexification, a state-augmentation, and employing controllable tubes [5], which we adopt in this work. The tractability of dynamic programming, consequently, relies on the tractability of controllable tube generation, for which we use constrained zonotopes [6].

The main contributions of this paper are: (i) a general set-theoretic result for closed-loop free-final-time control, and (ii) robust control accounting for both control uncertainty and time-varying state uncertainty; all with computationally tractable implementations, as demonstrated by means of a comprehensive autonomous precision landing case study.

Optimal predictive control for constrained systems is typically achieved by either: (i) model predictive control (MPC) and its variants [10, 11, 12, 13], and (ii) dynamic programming (DP) and its variants [8, 14, 9, 15, 16]. MPC involves the generation of an open-loop control sequence over a time-horizon, only applying the first control input, and repeating the process at each step [10, 7, 17, 18, 13]. Despite providing a feedback action to close the loop, the nominal prediction across the time-horizon itself is feedback-agnostic, i.e., it is open-loop, and robust variants tend to be conservative in terms of handling disturbances [7]. DP, in contrast, yields globally optimal closed-loop policies, but standard techniques rely on state-space discretization and storage of value function evaluations for each state across the entire time-horizon, the number of which grows exponentially as a function of the state dimension, thus limiting its use to small-dimensional systems [9]. Owing to the closed-loop nature of DP, robust control is less conservative than open-loop approaches.

Set-based methods in control [19, 7] have recently emerged as a popular approach to deterministic optimal control [20, 21] and robust optimal control [22, 23, 24, 5]. Specifically, [20, 21, 22] propose open-loop approaches that involve the construction of a controllable set for the autonomous precision landing problem, and [25, 26, 27, 24, 28, 5] propose closed-loop approaches that involve the construction of a controllable *tube*, i.e., a sequence of controllable sets, applied to spacecraft rendezvous trajectory optimization [25, 27, 23, 28, 24]. Most set-theoretic approaches leverage polytopes, which are often limited to low dimensions due to computational challenges [19]. However, the introduction of *constrained zonotopes* (CZs) has enabled computationally tractable set operations in high dimensions [6], which are largely beneficial in the field of control [6, 23, 24, 5].

In the context of autonomous precision landing, while earlier work dealt with deterministic optimal control [29, 30], there has been recent work on stochastic guidance and feedback control for landing as well [31, 32, 33]. More recently, [21] proposed a deterministic open-loop control framework for lunar landing with divert capabilities, which was later extended to account for navigational uncertainty in [22]—these papers made use of a convex-optimization-based simplicial "facet-pushing" algorithm to construct the open-loop controllable set, first presented in [20]. The case study we present herein serves as a closed-loop extension to the work on autonomous precision landing with the proposed novel contributions: closed-loop free-final-time optimal control and closed-loop control with robustness to both actuation (control) uncertainties and time-varying navigation (state) uncertainties.

Organization of the paper: We present optimal control and set-based preliminaries in §II. We then describe the set-based control architecture for closed-loop free-final-time optimal control in §III and robust control in §IV. Finally, we demonstrate the proposed framework by means of an autonomous precision landing case study in §V.

II. Preliminaries

In this section, we present the preliminaries that will aid in the development of the proposed computational framework in the paper. In §II.A, we formulate the template deterministic optimal control problem and construct a corresponding discrete-time convex optimal control problem with a polytopic feasible set. In §II.B, we present a computationally tractable approach to generating a polytopic inner approximation of a second-order (quadratic) cone, intersected with a halfspace, to convert second-order cone constraint sets—which show up in several control problems, including autonomous precision landing—to polytopes. Then, in §II.C, we define constrained zonotopes and describe the associated set operations. The use of constrained zono-

topes enables the efficient generation of controllable tubes, which we describe in §II.D. Finally, we present the one-step optimal control problem—that is executed online as part of the forward rollout—in §II.E.

II.A. The Deterministic Optimal Control Problem

We first consider a class of free-final-time optimal control problems in continuous-time, in the Lagrange form, with a convex cost functional, a class of nonlinear dynamical systems, and convex state and control constraints, as shown in Problem P1.

Problem P1: Template Continuous-Time Optimal Control Problem (Nonconvex)

$$
\begin{aligned}
&\underset{t_\mathrm{f},\, u}{\text{minimize}} && \textit{Cost Functional}: && \int_0^{t_\mathrm{f}} \beta\, h(u(t))\, dt && && \text{(P1a)}\\
&\text{subject to} && \textit{Dynamics}: && \dot{x}(t) = A_\mathrm{c}\, x(t) + B_\mathrm{c}\, u(t) + G_\mathrm{c}\, h(u(t)) + d_\mathrm{c}, && \forall t \in [0, t_\mathrm{f}] && \text{(P1b)}\\
& && \textit{State Constraints:} && x(t) \in \mathcal{X}_\mathrm{cvx}, && \forall t \in [0, t_\mathrm{f}] && \text{(P1c)}\\
& && \textit{Control Constraints:} && u(t) \in \mathcal{U}_\mathrm{cvx}, && \forall t \in [0, t_\mathrm{f}] && \text{(P1d)}\\
& && \textit{Initial Condition:} && x(0) = x_\mathrm{i} && && \text{(P1e)}\\
& && \textit{Final Condition:} && x(t_\mathrm{f}) \in \mathcal{X}_{\mathrm{f}_\mathrm{cvx}} && && \text{(P1f)}
\end{aligned}
$$

In Problem P1, $t_\mathrm{f} \in \mathbb{R}_{++}$ is the final time (a free variable), $x(t) \in \mathbb{R}^{n_x}$ is the state, $u(t) \in \mathbb{R}^{n_u}$ is the control input, $\beta \in \mathbb{R}_{++}$ is a constant, and $h : \mathbb{R}^{n_u} \to \mathbb{R}$ is a convex function; $\mathcal{X}_\mathrm{cvx} \subset \mathbb{R}^{n_x}$ and $\mathcal{U}_\mathrm{cvx} \subset \mathbb{R}^{n_u}$ are the convex state and control constraint sets, respectively; $\mathcal{X}_{\mathrm{f}_\mathrm{cvx}}$ is the convex final state set; and $x_\mathrm{i} \in \mathbb{R}^{n_x}$ is the fixed initial condition. In the dynamics, $A_c \in \mathbb{R}^{n_x \times n_x}$ is the system matrix, $B_c \in \mathbb{R}^{n_x \times n_u}$ is the control input matrix, $G_c \in \mathbb{R}^{n_x}$ is a vector that governs the influence of the convex function, h, on the state evolution, and $d_c \in \mathbb{R}^{n_x}$ is the affine term. All the constraint sets considered in this paper are assumed to be compact. Note that Problem P1 is nonconvex due to the presence of convex functions, h, in the equality constraint given by Equation (P1b).

Problem P1 can be relaxed to Problem P2, which is convex. In Problem P2, $\sigma(t) \in \mathbb{R}$, which will be treated

Problem P2: Template Continuous-Time Optimal Control Problem (Convexified)

$$
\begin{aligned}
&\underset{t_\mathrm{f},\, u,\, \sigma}{\text{minimize}} && \textit{Cost Functional}: && J := \int_0^{t_\mathrm{f}} \beta\, \sigma(t)\, dt && && \text{(P2a)}\\
&\text{subject to} && \textit{Dynamics}: && \dot{x}(t) = A_\mathrm{c}\, x(t) + B_\mathrm{c}\, u(t) + G_\mathrm{c}\, \sigma(t) + d_\mathrm{c}, && \forall t \in [0, t_\mathrm{f}] && \text{(P2b)}\\
& && \textit{Convex Relaxation:} && h(u(t)) \le \sigma(t), && \forall t \in [0, t_\mathrm{f}] && \text{(P2c)}\\
& && \textit{State Constraints:} && x(t) \in \mathcal{X}_\mathrm{cvx}, && \forall t \in [0, t_\mathrm{f}] && \text{(P2d)}\\
& && \textit{Control Constraints:} && (u(t), \sigma(t)) \in \hat{\mathcal{U}}_\mathrm{cvx}, && \forall t \in [0, t_\mathrm{f}] && \text{(P2e)}\\
& && \textit{Initial Condition:} && x(0) = x_\mathrm{i} && && \text{(P2f)}\\
& && \textit{Final Condition:} && x(t_\mathrm{f}) \in \mathcal{X}_{\mathrm{f}_\mathrm{cvx}} && && \text{(P2g)}
\end{aligned}
$$

as an auxiliary control input, is the slack variable introduced to convexify the problem, and $\hat{\mathcal{U}}_\mathrm{cvx} \subset \mathbb{R}^{n_u+1}$ is the augmented control constraint set. Note that Problem P2 bears resemblance to the problem templates considered in the vast body of literature on lossless convexification in optimal control [34, 35, 5], which are shown to recover globally optimal solutions to the original nonconvex problems.

Cost-To-Go Augmentation Recently, in the discrete-time setting, a general version of Problem P1 was shown to be losslessly convexifiable under some conditions [5]. Specifically, [5] leverages dynamic programming for the synthesis of the online controller to achieve global optimality in a closed loop using the so-called *controllable tube* (see §II.D). A key insight in [5] is that the controllable tube must explicitly account for the

cost-to-go, which can be subsequently minimized in the forward rollout. Next, we discuss augmentation of the dynamical system with the cost-to-go dynamics.

From the cost functional in Equation (P2a), J, the cost-to-go at any $t \in [0, t_\mathrm{f}]$ is given by:

$$c(t) := \int_t^{t_\mathrm{f}} \beta\, \sigma(\tau)\, d\tau \tag{1}$$

Note that $J = c(0)$. Taking the time-derivative of the cost-to-go yield the auxiliary dynamical system:

$$\dot{c}(t) = \frac{d}{dt} \int_t^{t_\mathrm{f}} \beta\, \sigma(\tau)\, d\tau = \frac{d}{dt} \int_{t_\mathrm{f}}^{t} -\beta\, \sigma(\tau)\, d\tau = -\beta\, \sigma(t) \tag{2}$$

From Equation (1), the boundary conditions for this auxiliary dynamical system are given by:

$$c(0) = \int_0^{t_\mathrm{f}} \beta\, \sigma(t)\, dt = J = \text{free} \tag{3a}$$

$$c(t_\mathrm{f}) = 0 \tag{3b}$$

Augmenting the dynamical system with the *cost-to-go dynamics* (Equation (2)) and writing the cost functional in terms of the cost-to-go state yields Problem P3, which is equivalent to Problem P2.

Problem P3: Template Continuous-Time Optimal Control Problem (Augmented)

minimize $t_\mathrm{f},\, u,\, \sigma,\, c(0)$	*Cost Functional*:	$c(0)$		(P3a)
subject to	*Dynamics*:	$\dot{x}(t) = A_\mathrm{c}\, x(t) + B_\mathrm{c}\, u(t) + G_\mathrm{c}\, \sigma(t) + d_\mathrm{c},$	$\forall t \in [0, t_\mathrm{f}]$	(P3b)
	Cost-to-go Dynamics:	$\dot{c}(t) = -\beta\, \sigma(t),$	$\forall t \in [0, t_\mathrm{f}]$	(P3c)
	State Constraints:	$(x(t), c(t)) \in \tilde{\mathcal{X}}_\mathrm{cvx},$	$\forall t \in [0, t_\mathrm{f}]$	(P3d)
	Control Constraints:	$(u(t), \sigma(t)) \in \tilde{\mathcal{U}}_\mathrm{cvx},$	$\forall t \in [0, t_\mathrm{f}]$	(P3e)
	Initial Condition:	$x(0) = x_\mathrm{i}$		(P3f)
	Final Condition:	$x(t_\mathrm{f}) \in \mathcal{X}_{\mathrm{f}_\mathrm{cvx}},\ c(t_\mathrm{f}) = 0$		(P3g)

In Problem P3, $\tilde{\mathcal{X}}_\mathrm{cvx} \subset \mathbb{R}^{n_x+1}$ is the augmented state constraint set and $\tilde{\mathcal{U}}_\mathrm{cvx} \subset \mathbb{R}^{n_u+1}$ is the augmented control constraint set, subsuming the constraint set defined by the convex relaxation in Equation (P2c).

We adopt a zero-order hold (ZOH), i.e., a piecewise-constant control parameterization, and time-discretize Problem P3. Assuming the true optimal control is a continuous signal, the control parameterization may lead to suboptimality, but since the dynamics themselves are exactly discretized, the continuous-time trajectory will exactly pass through the discrete temporal nodes. The state and control constraint sets are only considered at the nodes. Given (i) the convexity of the control constraint sets, and (ii) the ZOH control parameterization, the control constraints are guaranteed to be satisfied at and in between temporal nodes. For the state, however, these constraints are only imposed at (finitely many) temporal nodes and inter-sample constraint satisfaction is not guaranteed in general. The resulting discrete-time optimal control problem is shown in Problem P4.

Problem P4: Template Discrete-Time Optimal Control Problem (Conic)

minimize $N, c_1, u_1, \ldots, u_{N-1}, \sigma_1, \ldots, \sigma_{N-1}$	*Cost Function*:	c_1		(P4a)
subject to	*Dynamics*:	$(x_{k+1}, c_{k+1}) = A \cdot (x_k, c_k) + B \cdot (u_k, \sigma_k) + d,$	$k = 1, \ldots, N-1$	(P4b)
	State Constraints:	$(x_k, c_k) \in \tilde{\mathcal{X}}_\mathrm{cvx},$	$k = 1, \ldots, N$	(P4c)
	Control Constraints:	$(u_k, \sigma_k) \in \tilde{\mathcal{U}}_\mathrm{cvx},$	$k = 1, \ldots, N-1$	(P4d)
	Initial Condition:	$x_1 = x_\mathrm{i}$		(P4e)
	Final Condition:	$(x_N, c_N) \in \tilde{\mathcal{X}}_{\mathrm{f}_\mathrm{cvx}}$		(P4f)

In Problem P4, N is the number of temporal nodes (the horizon length), A and d are the discrete-time counterparts of $\text{blkdiag}\{A_\text{c}, 0\} \in \mathbb{R}^{(n_x+1)\times(n_x+1)}$ and $(d_\text{c}, 0) \in \mathbb{R}^{n_x+1}$, respectively, B is the discrete-time counterpart of $\begin{bmatrix} B_\text{c} & G_\text{c} \\ 0_{n_u} & -\beta \end{bmatrix} \in \mathbb{R}^{(n_x+1)\times(n_u+1)}$, and $\tilde{\mathcal{X}}_{\text{f}_\text{CVX}} := \mathcal{X}_f \times \{0\}$. We assume A is given by a matrix exponential and is hence always invertible. Problem P4 is conic since $\tilde{\mathcal{X}}_\text{CVX}$ and $\tilde{\mathcal{U}}_\text{CVX}$ can be arbitrary conic constraint sets.

The polytopic approximation of Problem P4 is given by Problem P5.

Problem P5: Template Discrete-Time Optimal Control Problem (Polytopic)

$$
\begin{aligned}
\underset{\substack{N,\, c_1,\\ u_1,\ldots,u_{N-1},\\ \sigma_1,\ldots,\sigma_{N-1}}}{\text{minimize}} \quad & \textit{Cost Function:} && c_1 && && \text{(P5a)}\\
\text{subject to} \quad & \textit{Dynamics:} && (x_{k+1}, c_{k+1}) = A\cdot(x_k, c_k) + B\cdot(u_k, \sigma_k) + d, && k = 1,\ldots,N-1 && \text{(P5b)}\\
& \textit{State Constraints:} && (x_k, c_k) \in \tilde{\mathcal{X}}, && k = 1,\ldots,N && \text{(P5c)}\\
& \textit{Control Constraints:} && (u_k, \sigma_k) \in \tilde{\mathcal{U}}, && k = 1,\ldots,N-1 && \text{(P5d)}\\
& \textit{Initial Condition:} && x_1 = x_\text{i} && && \text{(P5e)}\\
& \textit{Final Condition:} && (x_N, c_N) \in \tilde{\mathcal{X}}_\text{f} && && \text{(P5f)}
\end{aligned}
$$

In Problem P5, $\tilde{\mathcal{X}}$, $\tilde{\mathcal{U}}$, and $\tilde{\mathcal{X}}_\text{f}$ are the polytopic approximations of $\tilde{\mathcal{X}}_\text{CVX}$, $\tilde{\mathcal{U}}_\text{CVX}$, and $\tilde{\mathcal{X}}_{\text{f}_\text{CVX}}$, respectively. Note that there is no approximation involved if a constraint set is polytopic to begin with.

The optimal control approach we describe in this paper is *exact* for the template polytopic optimal control problem given by Problem P5. However, in the realm of optimal control, most problems are better represented by the template of Problem P1, which may involve second-order cone constraints that need to be approximated to fit the template of Problem P5, as shown. In §II.B, we provide a numerically tractable method to obtain "good" polytopic approximations to these convexified conic programs (Problem P4) for the special case wherein the conic constraint sets are quadratic cones, which are also known as standard/unit second-order cones (SOCs), ice-cream cones, or Lorentz cones [36].

II.B. Polytopic Inner Approximation of a Compact Quadratic Cone

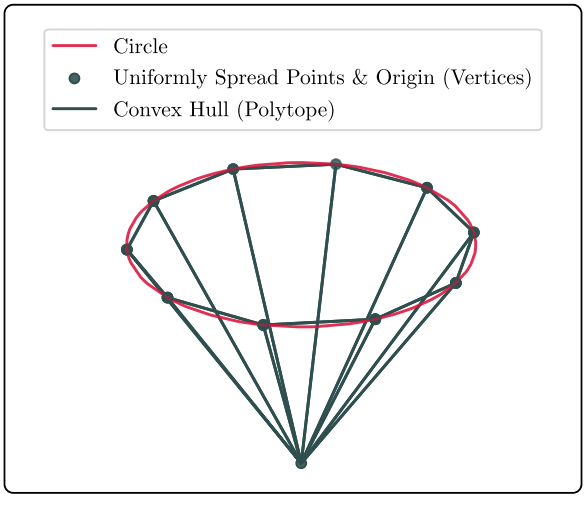

Figure 1: Polytopic inner approximation of a 3-dimensional compact quadratic cone.

For controllable tube generation as described in §II.D, the constraint sets are required to be polytopic, i.e., convex, closed, *and* bounded. There exist methods in the literature to obtain polyhedral *outer* approximations of an n-dimensional (unbounded, i.e., non-compact) quadratic cone [37, 38]. However, in order to

ensure that the generated controllable tube is conservative rather than infeasible, i.e., to guarantee feasibility of all points in the set, the polytopic constraint set must be an *inner* approximation of the original constraint set.

Motivated by these reasons, we propose a computationally tractable approach to obtaining a polytopic *inner* approximation of a closed and bounded, i.e., compact, n-dimensional quadratic cone, which is a special case of a quadratic cone, intersected with a halfspace.

We define a compact quadratic cone (CQC) as follows:

$$\mathbb{L}^n_{\text{CQC}} := \left\{(z,\, t) \;\middle|\; \|z\|_2 \leq t,\; 0 \leq t \leq t_{\max}\right\} \tag{4}$$

where $z \in \mathbb{R}^{n-1}$ and $t \in \mathbb{R}$.

To obtain a polytopic approximation of $\mathbb{L}^n_{\text{CQC}}$, we take a slice of $\mathbb{L}^n_{\text{CQC}}$ along $t = t_{\max}$:

$$\widetilde{\mathbb{L}}^n_{\text{CQC}} := \left\{(z,\, t) \;\middle|\; \|z\|_2 \leq t_{\max},\; t = t_{\max}\right\} \tag{5}$$

Here, $\mathcal{Z} := \left\{z \;\middle|\; \|z\|_2 \leq t_{\max}\right\}$ defines an $(n-1)$-dimensional hyperball.

Now, we proceed to uniformly spread points on the boundary of the $(n-1)$-dimensional hyperball by (approximately) solving a difference-of-convex (DC) optimization problem via the convex-concave procedure* [28, 26, 39]. The accuracy of the approximation can be improved by increasing the number of points. These $(n-1)$-dimensional points are then lifted to n dimensions with $t = t_{\max}$.

The convex hull of the union of these points and the origin (in n dimensions) is a polytope that is guaranteed to be an inner approximation of the original CQC. Figure 1 shows the polytopic approximation of a 3-dimensional CQC with 10 points spread on the boundary of the 2-dimensional hyperball, i.e., a circle.

II.C. Constrained Zonotopes

Polytopes are sets with a finite number of facets (flat sides). Convex polytopes that are closed and bounded, i.e., compact, can be represented either as the convex hull of a finite number of points/vertices (vertex representation: V-REP) or as the intersection of a finite number of halfspaces (halfspace representation: H-REP). In this work, we use the term *polytopes* to refer to compact convex polytopes.

Zonotopes are sets that are given by the Minkowski sum of line segments, called generators, with unit-box-constrained weights, known as latent variables, which are typically higher-dimensional than the set itself. *Constrained* zonotopes (CZs) are zonotopes with additional affine equality constraints on the weights (constrained generator representation: CG-REP)—they are a recently introduced class of sets that can be used to describe arbitrary polytopes, with distinct computational advantages over standard representations (V-REP, H-REP) [6]. In fact, a set is a constrained zonotope if and only if it is a polytope [6, Theorem 1]. While V-REP and H-REP polytopes suffer from the curse of dimensionality with exponential blow-up [40], CG-REP CZs do not (see Table 1). This key characteristic of CZs, in addition to the efficiency and accuracy of the corresponding set operations, has led to significant adoption in set-based methods, especially in the fields of estimation and control [6, 24].

Operation	V-REP Polytope	H-REP Polytope	CG-REP CZ
Intersection	×	✓	✓
Minkowski Sum	×	×	✓
Affine Map	✓	×†	✓

Table 1: Computational complexity for the required set operations in Algorithm 1: ✓ indicates that the operation has polynomial complexity and × indicates that it has exponential complexity (†exponential complexity in general, but polynomial complexity if the map is invertible).

*This is for $n \geq 3$; for $n \leq 2$, the uniform spreading problem can be solved in closed-form.

A CZ in CG-REP is defined by the following set [6, 41, 24, 28]:

$$\mathcal{Z} := \mathcal{Z}(G, c, A, b) = \{x \mid \exists\, \xi,\ x = G\,\xi + c,\ \|\xi\|_\infty \leq 1,\ A\,\xi = b\} \tag{6}$$

where $x \in \mathbb{R}^n$, $\xi \in \mathbb{R}^{n_g}$, $G \in \mathbb{R}^{n \times n_g}$, $c \in \mathbb{R}^n$, $A \in \mathbb{R}^{n_e \times n_g}$, and $b \in \mathbb{R}^{n_e}$; n is the dimension of the CZ, n_g is the number of generators (columns of G), and n_e is the number of equality constraints. Some useful set operations in terms of constrained zonotopes are presented in Table 2, where we use the following shorthand to denote a CG-REP CZ: $\mathcal{Z}_i := \{G_i, c_i, A_i, b_i\}$, where $i \in \mathbb{Z}_{++}$ is used to describe operations that involve more than one CZ. The subscript is dropped in the description of operations that only involve one CZ. A selector matrix is defined to be a matrix with one entry per row, which is unity.

Operation	**Method**	**Description**
Affine Map $R\,\mathcal{Z} + r$	$\{R\,G,\ R\,c + r,\ A,\ b\}$	$R\,\mathcal{Z} + r := \{R\,x + r \mid x \in \mathcal{Z}\}$, where $R \in \mathbb{R}^{n_r \times n}$ and $r \in \mathbb{R}^{n_r}$
Intersection $\mathcal{Z}_1 \cap \mathcal{Z}_2$	$\left\{\begin{bmatrix} G_1 & 0 \end{bmatrix},\ c_1,\ \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \\ G_1 & -G_2 \end{bmatrix},\ \begin{bmatrix} b_1 \\ b_2 \\ c_2 - c_1 \end{bmatrix}\right\}$	$\mathcal{Z}_1 \cap \mathcal{Z}_2 := \{x \in \mathcal{Z}_1 \mid x \in \mathcal{Z}_2\}$
Minkowski Sum $\mathcal{Z}_1 \oplus \mathcal{Z}_2$	$\left\{\begin{bmatrix} G_1 & G_2 \end{bmatrix},\ c_1 + c_2,\ \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix},\ \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}\right\}$	$\mathcal{Z}_1 \oplus \mathcal{Z}_2 := \{x_1 + x_2 \mid x_1 \in \mathcal{Z}_1,\ x_2 \in \mathcal{Z}_2\}$
Pontryagin Difference $\mathcal{Z}_1 \ominus \mathcal{Z}_2$	[24, Algorithm 3] (inner approximation)	$\mathcal{Z}_1 \ominus \mathcal{Z}_2 := \{x_1 \mid \forall x_2 \in \mathcal{Z}_2,\ x_1 + x_2 \in \mathcal{Z}_1\}$
Intersection with Affine Set $\mathcal{Z} \cap \{x \mid H\,x = h\}$	$\left\{G,\ c,\ \begin{bmatrix} A \\ H\,G \end{bmatrix},\ \begin{bmatrix} b \\ h - H\,c \end{bmatrix}\right\}$	$x \in \mathbb{R}^n$, $H \in \mathbb{R}^{n_h \times n}$, and $h \in \mathbb{R}^{n_h}$
Slice $\mathcal{Z} \cap \{x \mid E\,x = \bar{x}\}$	$\left\{G,\ c,\ \begin{bmatrix} A \\ E\,G \end{bmatrix},\ \begin{bmatrix} b \\ \bar{x} - E\,c \end{bmatrix}\right\}$	special case of intersection with affine set, where $\bar{x} \in \mathbb{R}^{n_{\bar{x}}}$, $n_{\bar{x}} \leq n$, and $E \in \mathbb{R}^{n_{\bar{x}} \times n}$ is a selector matrix
Projection $E\,\mathcal{Z}$	$\{E\,G,\ E\,c,\ A,\ b\}$	special case of affine map, where $E \in \mathbb{R}^{n_{\bar{x}} \times n}$ is a selector matrix, $n_{\bar{x}} \leq n$
Distance	$\underset{x \in \mathcal{Z}}{\text{minimize}}\ \|y - x\|_p$	$y \in \mathbb{R}^n$ is the point to be projected onto the set to compute the p-norm distance
Containment	$\underset{x \in \mathcal{Z}}{\text{minimize}}\ \|y - x\|_p$	$y \in \mathcal{Z}$ if the distance is zero; $y \notin \mathcal{Z}$ otherwise
Emptiness	$\underset{x \in \mathcal{Z}}{\text{minimize}}\ 0$	the set is empty if the problem is infeasible; the set is nonempty otherwise
Support	$\underset{x \in \mathcal{Z}}{\text{maximize}}\ \eta^\top x$	$\eta \in \mathbb{R}^n$ is the direction vector along which to evaluate the support function
Extreme Point	$\underset{x \in \mathcal{Z}}{\text{argmax}}\ \eta^\top x$	point on the boundary of the set along the direction vector, η

Table 2: Useful set operations in terms of constrained zonotopes, which can be found in [6, 41, 24, 28].

II.D. Controllable Tube Generation (Offline)

We refer to the target final set as the *terminal set*. We refer to the set of all initial conditions from which a given final set is reachable, in the presence of state and control constraints, for a given trajectory time, as a *controllable set*. The union of all such controllable sets over time is referred to as the *controllable tube*.

We start with deterministic optimal control problems. For problems that fit the template of Problem P5, the exact controllable tube can be constructed by means of a simple backward set-recursion, given by Algorithm 1, which is executed offline. Algorithm 1 is computationally tractable even in high dimensions when the sets in question are represented as constrained zonotopes (CG-REP). This can be extended to handle problems

with uncertainties, which will be discussed in §IV. Note that the generated controllable (backward reachable) sets are guaranteed to be feasible with respect to Problem P4.

Here, N is the number of temporal nodes (the horizon length) and CS_k is the k^{th} controllable set, the union over $k = 1, \ldots, N$ of which is the controllable tube. If the cost-to-go is strictly monotonically decreasing (and since the cost-to-go is bounded by construction), this set recursion is guaranteed to terminate for a finite N, i.e., the set recursion will yield empty sets after N iterations, in which case the first nonempty controllable set, CS_1, corresponds to the maximum-feasible trajectory time. In practice, since N is not known a priori, this backward recursion is performed starting with index 1 until it terminates at an index, N, after which the indices are reversed.

II.E. One-Step Optimal Control (Online)

As shown in [5], closed-loop optimal control synthesis may be done via a *forward rollout*, i.e., by solving a sequence of one-step optimal control problems. The forward rollout involves minimization of the current cost-to-go, and one-step set-containment constraints: the next-step controllable set for the propagated state, and the current-step control constraint set for the control input. The optimal control problem—given by Problem P6—itself is a linear program (LP), and can be solved using efficient LP solvers [42, 43, 44, 45].

Problem P6: One-Step Optimal Control Problem (Polytopic)

$$
\begin{aligned}
&\underset{u_k,\,\sigma_k,\,c_k}{\text{minimize}} && \textit{Cost Function}: && c_k && \text{(P6a)}\\
&\text{subject to} && \textit{Dynamics}: && (x_{k+1}, c_{k+1}) = A \cdot (x_k, c_k) + B \cdot (u_k, \sigma_k) + d && \text{(P6b)}\\
&&& \textit{State Constraints (Controllable Set):} && (x_{k+1}, c_{k+1}) \in \text{CS}_{k+1} && \text{(P6c)}\\
&&& \textit{Control Constraints (Polytopic):} && (u_k, \sigma_k) \in \tilde{\mathcal{U}} && \text{(P6d)}
\end{aligned}
$$

Remark 1. *The control constraint set, $\tilde{\mathcal{U}}$, in the one-step optimal control problem, Problem P6, need not be the polytopic approximation. Instead, we can use the original conic control input set—since it is larger than the polytopic approximation by construction—and solve Problem P7. This has the two-fold benefit of: (i) rendering the convex relaxation tight with respect to the original conic constraint, and (ii) producing solutions with a lower cost. Consequently, instead of solving a sequence of 1-step linear programs, we would solve a sequence of 1-step conic programs, for which there exist efficient solvers [46, 47, 48, 49].*

Algorithm 1 Set Recursion for Controllable Tube Generation

Inputs: N, A, B, d, $\tilde{\mathcal{X}}$, $\tilde{\mathcal{X}}_{\text{f}}$, $\tilde{\mathcal{U}}$

1: $\text{CS}_N \leftarrow \tilde{\mathcal{X}}_{\text{f}}$
2: **for** $k = (N-1), (N-2), \ldots, 2, 1$ **do**
3: $\quad \text{CS}_k = \tilde{\mathcal{X}} \cap A^{-1}\left(\text{CS}_{k+1} \oplus \left(-B\tilde{\mathcal{U}} - d\right)\right)$ ▷ $\oplus$: Minkowski sum
4: **end for**

Return: $\text{CS}_{1,\ldots,N}$

Problem P7: One-Step Optimal Control Problem

$\underset{u_k, \sigma_k, c_k}{\text{minimize}}$	*Cost Function*:	c_k	(P7a)
subject to	*Dynamics*:	$(x_{k+1}, c_{k+1}) = A \cdot (x_k, c_k) + B \cdot (u_k, \sigma_k) + d$	(P7b)
	State Constraints (Controllable Set):	$(x_{k+1}, c_{k+1}) \in \text{CS}_{k+1}$	(P7c)
	Control Constraints (Conic):	$(u_k, \sigma_k) \in \tilde{\mathcal{U}}_{\text{CVX}}$	(P7d)

Remark 2. *The recursive feasibility of Problems P6 and P7 is guaranteed by Equations* (P6c) *and* (P7c)*, respectively.*

III. Free-Final-Time Optimal Control

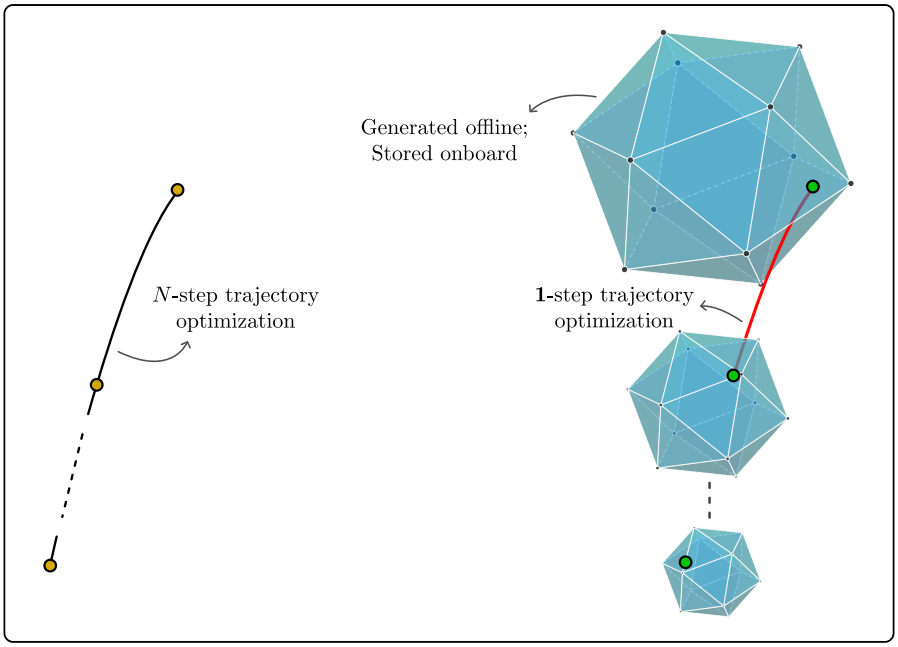

Figure 2: The open-loop approach (left) vs. the closed-loop approach using a controllable tube, i.e., a collection of controllable sets (right): the closed-loop approach (Algorithm 3) recovers a globally optimal solution to the open-loop problem (Problem P5).

In this section, we present the first main result, a computationally tractable set-based approach to free-final-time optimal control. We present the computation of the optimal horizon length in §III.A, and the corresponding free-final-time forward rollout algorithm in §III.B.

III.A. Optimal Horizon Computation

With the existing controllable tube architecture, we can compute the free-final-time cost-optimal closed-loop trajectory, for a given initial condition, directly. The proposed approach requires simple and cheap set operations and the solution to small-dimensional linear programs. A notable contribution here is that this method can apply to problems that fit the template of Problem P5 in general, as long as the backward set-recursion terminates for a finite N; i.e., we make no assumptions on the relationship between the cost metric and the trajectory time. However, additional information about this relationship, if known, can be leveraged to accelerate the search. For example, if the relationship is known to be unimodal [50, 51], a golden section search can be employed [52, 53].

During online execution, the initial condition of the system may be contained in several of the controllable

sets. In other words, there may exist feasible solutions to the terminal set starting from many different controllable sets—if the system can reach the terminal set in $M \leq N$ steps (with $M-1$ control actions) for example, it could potentially reach the terminal set in $M+1$ steps (with M control actions) and/or $M-1$ steps (with $M-2$ control actions), and so on, as well.

A natural approach then is to check for containment of the initial condition point in the controllable sets starting from the terminal set, going backward in time, and to then stop when the first containment is achieved. This would correspond to the minimum-time solution [19]. However, rather than choosing the minimum-time controllable set (or any other time-slice based on such heuristics), we can instead choose the time-slice that corresponds to the least possible cost-to-go, which is easy to evaluate in real-time.

Now, we present a general set-theoretic result for free-final-time optimal control that we then leverage within our computational closed-loop control framework.

Theorem 1. *Let $x = x_{\mathrm{i}} \in \mathbb{R}^{n_x}$ be the initial condition (current state). Further, let:*

$$\mathcal{K} := \{k \in \{1, \ldots, N\} \mid x_{\mathrm{i}} \in \{x \mid (x,\, c) \in \mathrm{CS}_k\}\} \tag{7}$$

$$\mathcal{C}_k(x_{\mathrm{i}}) := \{c \mid (x_{\mathrm{i}},\, c) \in \mathrm{CS}_k\}, \quad k \in \mathcal{K} \tag{8}$$

$$c_k := \min_{c \,\in\, \mathcal{C}_k(x_{\mathrm{i}})} \; c \tag{9}$$

where $\mathcal{K}$ is the set of indices of controllable sets CS_k, $k = 1, \ldots, N$, that contain x_{i}, and, for any $k \in \mathcal{K}$, $\mathcal{C}_k(x_{\mathrm{i}})$ is the slice of CS_k at x_{i} projected onto the cost-to-go coordinate, and c_k is the minimum of $\mathcal{C}_k(x_{\mathrm{i}})$. Assume $\mathcal{K}$ is nonempty and, for all $k \in \mathcal{K}$, CS_k is compact. Then the globally optimal horizon length is $N - k^\star$, where:

$$k^\star = \operatorname*{argmin}_{k \,\in\, \mathcal{K}} \; c_k \tag{10}$$

and the corresponding globally optimal cost is:

$$c^\star := c_{k^\star} \tag{11}$$

Proof. Consider any $k \in \mathcal{K}$. Let the optimal trajectory cost to go from x_{i} to $\mathcal{X}_{\mathrm{f}}$ in $N-k$ steps be J_k. By construction, $(x,\, c) \in \mathrm{CS}_k$ if and only if there exists a sequence of $N-k$ feasible one-step control actions from x to $\mathcal{X}_{\mathrm{f}}$, with cost less than or equal to c, i.e.:

$$J_k \leq c, \;\; \forall\, c \in \mathcal{C}_k(x_{\mathrm{i}}) \iff J_k \leq \min_{c \,\in\, \mathcal{C}_k(x_{\mathrm{i}})} \; c \iff J_k \leq c_k \tag{12}$$

Next, since the last dimension in CS_k is the cost-to-go state (by construction), we have that $(x_{\mathrm{i}},\, J_k) \in \mathrm{CS}_k$, which implies $J_k \in \mathcal{C}_k(x_{\mathrm{i}})$. Therefore, we have:

$$\min_{c \,\in\, \mathcal{C}_k(x_{\mathrm{i}})} \; c \leq J_k \iff c_k \leq J_k \tag{13}$$

From Equations (12) and (13), we have:

$$J_k = c_k \tag{14}$$

Taking the minimum of Equation (14) over $k \in \mathcal{K}$, we get:

$$\min_{k \,\in\, \mathcal{K}} \; J_k = \min_{k \,\in\, \mathcal{K}} \; c_k \tag{15}$$

which is attained at $k^\star = \operatorname{argmin}_{k \,\in\, \mathcal{K}} \, c_k$. Hence, the globally optimal horizon length is $N - k^\star$ and the corresponding globally optimal cost is $c^\star = c_{k^\star}$. □

Remark 3. *If CS_k, $k \in \mathcal{K}$, are compact convex sets, then each $\mathcal{C}_k(x_{\mathrm{i}})$ is a closed interval in $\mathbb{R}$, and $\min_{c \,\in\, \mathcal{C}_k(x_{\mathrm{i}})} \, c$ is its left endpoint.*

To compute the minimum cost-to-go, we first slice the $(n_x + 1)$-dimensional controllable tube along the n_x-dimensional initial condition point, project it onto the cost-to-go coordinate (to obtain an interval, i.e., a set in one dimension), and then compute the left extreme point of the remaining one-dimensional cost-to-go set (interval). This value corresponds to the minimum possible cost-to-go from that particular controllable set. Note that this set, although 1-dimensional, is a constrained zonotope, so computing the left extreme point requires solving a linear program.

We perform this computation for each of the time-slices that contain the initial condition, and pick the time-slice with the least cost-to-go among them. Then, the forward rollout algorithm, as described in §III.B, starting from this set, will produce the globally optimal trajectory.

Algorithm 2 Optimal Horizon Computation

Inputs: (1) Initial condition: x_i
(2) Controllable tube: $\mathrm{CS}_{1,\ldots,N}$

$\mathcal{K} \leftarrow$ `containment_check`$(x_\mathrm{i}, \mathrm{CS}_{1,\ldots,N})$ ▷ Equation (7)
for $k \in \mathcal{K}$ **do**
  $\mathcal{C}_k \leftarrow$ slice CS_k along x_i and project it onto the cost-to-go coordinate
  $c_k \leftarrow \min_{c \in \mathcal{C}_k} c$
end for
$k^\star \leftarrow \operatorname{argmin}_{k \in \mathcal{K}} c_k$

Return: $k^\star$ ▷ optimal start index

Algorithm 2 gives us the optimal starting index, $k^\star$, i.e., the controllable set index from which to initiate the sequence of $N - k^\star$ control actions, which is guaranteed to give us the globally optimal trajectory.

III.B. Forward Rollout

The *forward rollout* of one-step optimal control problems, from the current controllable set (with index k) to the next (with index $k + 1$), is described in Algorithm 3. Note that this is distinct from an *open-loop* approach, wherein the sequence of optimal control actions is determined in one shot—unlike that, the sequence of optimal control actions is determined one step at a time here, allowing for the incorporation of feedback at every step, making it a *closed-loop* approach; see Figure 2 for a depiction of the same.

Algorithm 3 Forward Rollout

Inputs: (1) Initial condition: x_i
(2) Controllable tube: $\mathrm{CS}_{1,\ldots,N}$

$k^\star \leftarrow$ `optimal_horizon`$(x_\mathrm{i}, \mathrm{CS}_{1,\ldots,N})$ ▷ Algorithm 2
$x_{k^\star} \leftarrow x_\mathrm{i}$
for $k = k^\star, \ldots, N-1$ **do**
  $x_{k+1}, u_k \leftarrow$ `one_step_optimal_control`(x_k, CS_{k+1}) ▷ solve Problem P7
end for

Return: $u_{k^\star,\ldots,N-1}$ ▷ globally optimal control sequence

IV. Robust Control

We describe the robust optimal control problem and distinguish between robust open-loop optimal control and robust closed-loop optimal control in §IV.A. We assume that the state measurements and the control input are subject to stochastic uncertainty, and further, that the uncertainties in the state are time-varying;

we describe the modeling of these uncertainties in §IV.B. The robust control approach we propose, however, requires the characterization of bounded disturbance sets, such that the closed-loop system is robust to any state and control disturbances that lie within these sets.

Rather than arbitrarily choosing bounded sets for this purpose, we are interested in the systematic construction of bounded sets that allow us to make the claim that the trajectory, even in the presence of random disturbance vectors sampled from known (possibly unbounded) probability distributions, is guaranteed to stay within the controllable tube with a user-specified probability. We describe this procedure in generality in §IV.C. In the case that the uncertainty is bounded to begin with, the corresponding bounded disturbance sets can be directly used within our framework.

In the presence of disturbances that are coupled in time, the construction of the bounded disturbance sets requires additional considerations. Once the disturbance sets are constructed, the state and control constraint sets, and in turn, the controllable tube, are robustified, i.e., tightened, such that they are robust to the worst-case disturbances from the bounded sets; we describe the construction of the bounded disturbance sets and the robustification of the control and state constraints in §IV.D. Note that these steps are performed offline. Within the confines of this shrunken feasible space, the controller synthesized online is cost-optimal, making this approach one of *robust optimal control*, in that we prioritize robustness to disturbances first, and then optimize performance among the admissible controls.

IV.A. The Robust Optimal Control Problem

In the deterministic case, the open-loop and closed-loop optimal control problems are equivalent [9]. In the robust case, however, there are differences—specifically, as shown in §IV.A.3, the robust open-loop optimal control problem described in §IV.A.1 is more conservative than the robust closed-loop optimal control problem described in §IV.A.2.

IV.A.1. Robust Open-Loop Optimal Control

Consider the optimal control problem given by Problem P5, but with the inclusion of bounded additive disturbances, $w_k \in \mathcal{W}_k \subset \mathbb{R}^{n_x+1}$, where $\mathcal{W}_k$, $k = 1, \dots, N$, are compact; see Problem P8. This problem represents the robust *open-loop* optimal control problem, since any control input along the future horizon only depends on the state at the beginning of the horizon.

Problem P8: Robust Open-Loop Optimal Control Problem (Polytopic)

$$\min_{\substack{N,\, c_1 \\ u_1,\dots,u_{N-1},\\ \sigma_1,\dots,\sigma_{N-1}}} \ \max_{w_1,\dots,w_N} \quad c_1 \tag{P8a}$$

$$\text{s.t.} \quad (x_{k+1}, c_{k+1}) = A \cdot (x_k, c_k) + B \cdot (u_k, \sigma_k) + d + w_k, \quad k = 1, \dots, N-1 \tag{P8b}$$

$$(x_k, c_k) \in \tilde{\mathcal{X}}, \quad k = 1, \dots, N \tag{P8c}$$

$$(u_k, \sigma_k) \in \tilde{\mathcal{U}}, \quad k = 1, \dots, N-1 \tag{P8d}$$

$$w_k \in \mathcal{W}_k, \quad k = 1, \dots, N \tag{P8e}$$

$$x_1 = x_{\mathrm{i}} \tag{P8f}$$

$$(x_N, c_N) + w_N \in \tilde{\mathcal{X}}_{\mathrm{f}} \tag{P8g}$$

In practice, the optimal control problem is first robustified (i.e., the constraints are tightened) with respect to the worst-case disturbances from the bounded sets (owing to the maximization over them), and then the entire sequence of controls is solved for at once.

IV.A.2. Robust Closed-Loop Optimal Control

Consider Problem P5 again, with the same bounded additive disturbances, $w_k \in \mathcal{W}_k \subset \mathbb{R}^{n_x+1}$, $k = 1, \dots, N$. Now, instead of sequential maximization over disturbance and minimization over controls, we leverage the additional information provided by the value of the current state [9], and formulate an interleaved minimization and maximization problem; see Problem P9. This problem represents the robust *closed-loop* optimal

control problem, since the control input at any given point in time explicitly depends on the state at the same time.

Problem P9: Robust Closed-Loop Optimal Control Problem (Polytopic)

$$\min_{N,\,c_1}\ \min_{u_1}\max_{w_1}\ \ldots\ \min_{u_{N-1}}\max_{w_{N-1}}\max_{w_N}\ c_1 \tag{P9a}$$

s.t.

$$(x_{k+1}, c_{k+1}) = A \cdot (x_k, c_k) + B \cdot (u_k, \sigma_k) + d + w_k, \quad k = 1, \ldots, N-1 \tag{P9b}$$

$$(x_k, c_k) \in \tilde{\mathcal{X}}, \quad k = 1, \ldots, N \tag{P9c}$$

$$(u_k, \sigma_k) \in \tilde{\mathcal{U}}, \quad k = 1, \ldots, N-1 \tag{P9d}$$

$$w_k \in \mathcal{W}_k, \quad k = 1, \ldots, N \tag{P9e}$$

$$x_1 = x_\mathrm{i} \tag{P9f}$$

$$(x_N, c_N) + w_N \in \tilde{\mathcal{X}}_\mathrm{f} \tag{P9g}$$

In practice, Problem P9 is solved by means of dynamic programming—first, a backward recursion, which accounts for robustification, as shown in Algorithm 4, and then a forward rollout, i.e., a sequence of one-step optimal control problems, akin to Problem P7, but with the robustified constraint sets instead.

IV.A.3. Robust Closed-Loop Optimal Control vs. Robust Open-Loop Optimal Control

Consider the *max-min inequality* [54, Equation (5.46)]:

$$\sup_{z\in Z}\ \inf_{y\in Y}\ f(y,z) \leq \inf_{y\in Y}\ \sup_{z\in Z}\ f(y,z) \tag{16}$$

for any $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$, any $Y \subseteq \mathbb{R}^n$, and any $Z \subseteq \mathbb{R}^m$. When the maximum and the minimum exist, Equation (16) can be written as follows:

$$\max_{z\in Z}\ \min_{y\in Y}\ f(y,z) \leq \min_{y\in Y}\ \max_{z\in Z}\ f(y,z) \tag{17}$$

Repeated application of the max-min inequality in Equation (17) to Problem P9 (closed-loop) results in the minimization and maximization operations grouped together, as in Problem P8 (open-loop). Thus, we conclude that (i) the robust closed-loop optimal control problem has a lower cost than that of its open-loop counterpart, and (ii) the open-loop approach is *more conservative* than the closed-loop approach.

IV.B. State and Control Uncertainty Modeling

Let $y := (x, c) \in \mathbb{R}^{n_x+1}$ be the augmented state and $s := (u, \sigma) \in \mathbb{R}^{n_u+1}$ be the augmented controls.

Consider the following deterministic discrete-time dynamics (Equation (P5b)) in terms of y and s:

$$y_{k+1} = A\, y_k + B\, s_k + d, \quad k = 1, \ldots, N-1 \tag{18}$$

We first consider the case where some (or all) of the control inputs are subject to an additive disturbance that is Gaussian and independent and identically distributed (i.i.d.):

$$\tilde{s}_k := s_k + E_w^u\, w_k^u, \quad w_k^u \sim \mathcal{N}(0, \Sigma^u), \quad k = 1, \ldots, N-1 \tag{19}$$

where $\Sigma^u \in \mathbb{S}^{\tilde{n}_u}$ is the control disturbance covariance and $\tilde{n}_u \leq n_u + 1$ is the dimension of the uncertain controls. We define an *embedding matrix* to be a matrix that maps/embeds the vector it left-multiplies to/in a (potentially) higher dimension—it only has one entry per column, and that entry is unity (and thus an embedding matrix is full column rank); an embedding matrix is akin to the transpose of a selector matrix. Here, $E_w^u \in \mathbb{R}^{(n_u+1)\times\tilde{n}_u}$ is the uncertain-control embedding matrix, i.e., it lifts w_k^u from $\tilde{n}_u$ dimensions to $n_u + 1$ dimensions. For $k = 1, \ldots, N-1$, we have:

$$y_{k+1} = A\, y_k + B\, \tilde{s}_k + d \tag{20a}$$

$$= A\, y_k + B\, (s_k + E_w^u\, w_k^u) + d \tag{20b}$$

$$= A\,y_k + B\,s_k + B\,E_w^u\,w_k^u + d \tag{20c}$$

Now, additionally, consider the case where some (or all) of the state measurements, $\tilde{y}_k, k = 1, \ldots, N-1$, are uncertain with independent additive Gaussian disturbances with time-varying covariances:

$$\tilde{y}_k := y_k + E_w^x\,w_k^x, \quad w_k^x \sim \mathcal{N}(0, \Sigma_k^x), \quad k = 1, \ldots, N \tag{21}$$

where $\Sigma_k^x \in \mathbb{S}^{\tilde{n}_x}$, $k = 1, \ldots, N$, are the (time-varying) state covariances, where $\tilde{n}_x \leq n_x + 1$ is the dimension of the uncertain component of the state and $E_w^x \in \mathbb{R}^{(n_x+1)\times\tilde{n}_x}$ is the uncertain-state embedding matrix. Therefore, for $k = 1, \ldots, N-1$, we have:

$$\tilde{y}_{k+1} = y_{k+1} + E_w^x\,w_{k+1}^x \tag{22a}$$
$$= A\,y_k + B\,s_k + B\,E_w^u\,w_k^u + d + E_w^x\,w_{k+1}^x \tag{22b}$$
$$= A\,(\tilde{y}_k - E_w^x\,w_k^x) + B\,s_k + B\,E_w^u\,w_k^u + d + E_w^x\,w_{k+1}^x \tag{22c}$$
$$= A\,\tilde{y}_k + B\,s_k + d + (B\,E_w^u\,w_k^u + E_w^x\,w_{k+1}^x - A\,E_w^x\,w_k^x) \tag{22d}$$

Finally, we have:

$$\tilde{y}_{k+1} = A\,\tilde{y}_k + B\,s_k + d + w_k \tag{23}$$

where, for $k = 1, \ldots, N-1$, we have:

$$w_k \;= B\,E_w^u\,w_k^u + E_w^x\,w_{k+1}^x - A\,E_w^x\,w_k^x \;= \begin{bmatrix} B\,E_w^u & E_w^x & -A\,E_w^x \end{bmatrix} \begin{bmatrix} w_k^u \\ w_{k+1}^x \\ w_k^x \end{bmatrix} := M\,\hat{w}_k \tag{24}$$

where $\hat{w}_k := (w_k^u, w_{k+1}^x, w_k^x) \in \mathbb{R}^{\tilde{n}_u + 2\,\tilde{n}_x}$ is the concatenation of the disturbance vectors influencing the dynamics, and $M := \begin{bmatrix} B\,E_w^u & E_w^x & -A\,E_w^x \end{bmatrix} \in \mathbb{R}^{(n_x+1)\times(\tilde{n}_u + 2\,\tilde{n}_x)}$ is a linear map. Further, note that $w_N = E_w^x\,w_N^x$. Equation (23) represents the uncertain dynamical system.

IV.C. Representing Stochastic Uncertainty as a Bounded Disturbance Set

For Gaussian disturbances, which have unbounded distributions, ellipsoids are the best option to represent bounded disturbance sets, since they are the natural sets of confidence regions due to the shape of the Gaussian kernel. Each bounded set is the p-level set of a Gaussian distribution, which, in turn, guarantees that the true state at a given step, in the presence of a disturbance sampled from this Gaussian distribution, will lie within the corresponding robust controllable set with probability at least p [26].

The logical interpretation of this is as follows:

(i) The probability that a random disturbance vector sampled from the Gaussian distribution lies within the corresponding bounded disturbance set is p (by construction of the bounded disturbance set).

(ii) The true state at a given step is guaranteed to lie within the corresponding controllable set for any disturbance from within the bounded disturbance set (by construction/robustification of the controllable tube).

(iii) From (i) and (ii), the probability that the true state at a given step will lie inside the corresponding controllable set for a random disturbance sampled from the Gaussian distribution is at least p.

We assume the instantaneous disturbance, $w \in \mathbb{R}^n$, is Gaussian, with mean $\mu \in \mathbb{R}^n$ and covariance $\Sigma \in \mathbb{S}^n$, i.e., $w \sim \mathcal{N}(\mu, \Sigma)$.

Consider the ellipsoid with shape matrix Σ, centered around $\mu \in \mathbb{R}^n$ and parameterized by size parameter $R^2 \in [0, \infty)$ [26]:

$$\mathcal{E}(R^2; \mu, \Sigma) := \{q \in \mathbb{R}^n \mid (q-\mu)^\top \Sigma^{-1} (q-\mu) \leq R^2\} \tag{25}$$

Now, let $\eta \in \mathbb{R}^n$ be a standard normal random vector, i.e., $\eta \sim \mathcal{N}(0, I_n)$. Then, w can be written in terms of η as follows [55]:

$$w = \Sigma^{\frac{1}{2}}\,\eta + \mu \tag{26}$$

Then,

$$w \in \mathcal{E}(R^2; \mu, \Sigma) \implies (w-\mu)^\top \Sigma^{-1}(w-\mu) \leq R^2 \tag{27a}$$

$$\implies \left(\Sigma^{\frac{1}{2}}\,\eta + \mu - \mu\right)^\top \Sigma^{-1} \left(\Sigma^{\frac{1}{2}}\,\eta + \mu - \mu\right) \leq R^2 \tag{27b}$$

$$\implies \left(\eta^\top \Sigma^{\frac{1}{2}}\right) \Sigma^{-\frac{1}{2}}\,\Sigma^{-\frac{1}{2}} \left(\Sigma^{\frac{1}{2}}\,\eta\right) \leq R^2 \tag{27c}$$

$$\implies \eta^\top \eta \leq R^2 \tag{27d}$$

$$\implies \eta \in \mathcal{E}(R^2; 0, I_n) \tag{27e}$$

Consequently, we have [26, §3.3.2]:

$$\mathbb{P}_w\big(w \in \mathcal{E}(R^2; \mu, \Sigma)\big) = \mathbb{P}_\eta\big(\eta \in \mathcal{E}(R^2; 0, I_n)\big) \tag{28a}$$

$$= \mathbb{P}_\eta\big(\chi^2(n) \leq R^2\big) \tag{28b}$$

$$= F_{\chi^2(n)}(R^2) \tag{28c}$$

where, since η is a normal random variable, $\eta^\top \eta = \|\eta\|_2^2$ can be replaced with $\chi^2(n)$, which is a chi-squared random variable with n degrees of freedom, $F_{\chi^2(n)}(R^2)$ being its cumulative distribution function.

Given a user-specified probability, p, we set:

$$R^2 = F^{-1}_{\chi^2(n)}(p) \tag{29}$$

which gives us:

$$F_{\chi^2(n)}(R^2) = p \tag{30}$$

and hence, gives us the following guarantee:

$$\mathbb{P}_w\big(w \in \mathcal{E}(R^2; \mu, \Sigma)\big) = p \tag{31}$$

IV.D. Robust Controllable Tube Generation

Robust controllable tube generation involves the contruction of bounded disturbance sets, which we present in §IV.D.1, and the tightening, i.e., *robustification*, of the control and state constraint sets, which we present in §IV.D.2 and §IV.D.3, respectively. The set recursion algorithm for robust controllable tube generation is provided in §IV.D.3.

IV.D.1. Bounded Disturbance Set Construction

While w_k^x and w_k^u in Equations (21) and (19), respectively, are independent across time, $\hat{w}_k$ is not, due to the coupling of w_k^x between time-steps. For the purpose of bounded disturbance set construction, and, consequently, robust controllable tube generation, however, we make the assumption that $\hat{w}_k$ are independent, i.e., we assume that $\hat{w}_k \sim \mathcal{N}(0, \Sigma_k^{\hat{w}})$, where $\Sigma_k^{\hat{w}} := \text{blkdiag}\{\Sigma^u, \Sigma_{k+1}^x, \Sigma_k^x\} \in \mathbb{S}^{\tilde{n}_u + 2\,\tilde{n}_x}$, $k = 1, \ldots, N-1$. Note that this assumption is conservative, since it ignores the temporal correlation in $\hat{w}_k$ (the temporal correlation manifests as additional affine constraints on w_k, which are ignored, leading to robustification with respect to a larger disturbance set). Further, since a Gaussian distribution is closed under linear transformations of random variables [56, Theorem 3.3.3], w_k is also Gaussian.

The corresponding bounded disturbance sets are constructed as follows. First, we consider the following ellipsoids (see Equation (25)) for $k = 1, \ldots, N-1$:

$$\mathcal{E}_k(R_k^2; 0, \Sigma_k^{\hat{w}}) = \{q \in \mathbb{R}^{\tilde{n}_u + 2\,\tilde{n}_x} \mid q^\top (\Sigma_k^{\hat{w}})^{-1} q \leq R_k^2\} \tag{32}$$

and for $k = N$:

$$\mathcal{E}_N(R_N^2; 0, \Sigma_N^x) = \{q \in \mathbb{R}^{\tilde{n}_x} \mid q^\top (\Sigma_N^x)^{-1} q \leq R_N^2\} \tag{33}$$

Since the robust controllable tube generated using these bounded disturbance sets will be tightened with respect to the worst-case disturbances from these sets, the tube will be guaranteed to be robust with respect to the original disturbance vectors, $\hat{w}_k := (w_k^u, w_{k+1}^x, w_k^x) \in \mathbb{R}^{\tilde{n}_u + 2\,\tilde{n}_x}$, $k = 1, \ldots, N-1$, as well.

Next, to get the effective disturbance sets for $k = 1, \ldots, N-1$, each in $\mathbb{R}^{n_x+1}$, we take linear transformations of $\mathcal{E}_k(R_k^2; 0, \Sigma_k^{\hat{w}})$ with respect to M, which are ellipsoids themselves [57], i.e.,

$$\mathcal{W}_k := \{Mq \mid M \in \mathbb{R}^{(n_x+1)\times(\tilde{n}_u+2\,\tilde{n}_x)},\ q \in \mathcal{E}_k(R_k^2; 0, \Sigma_k^{\hat{w}})\} \tag{34a}$$

$$= \{l \in \mathbb{R}^{n_x+1} \mid l \in \mathcal{E}_k(R_k^2; 0, M\Sigma_k^{\hat{w}} M^\top)\} \tag{34b}$$

For $k = N$, we have:

$$\mathcal{W}_N := \{E_w^x\, q \mid E_w^x \in \mathbb{R}^{(n_x+1)\times\tilde{n}_x},\ q \in \mathcal{E}_N(R_N^2; 0, \Sigma_N^x)\} \tag{35a}$$

$$= \{l \in \mathbb{R}^{n_x+1} \mid l \in \mathcal{E}_N(R_N^2; 0, E_w^x\, \Sigma_N^x (E_w^x)^\top)\} \tag{35b}$$

Further, in accordance with Equation (29), we set $R_k^2 = F^{-1}_{\chi^2(\tilde{n}_u+2\,\tilde{n}_x)}(p)$, $k = 1, \ldots, N-1$, and $R_N^2 = F^{-1}_{\chi^2(\tilde{n}_x)}(p)$, wherein, in order to guarantee that the probability of the trajectory at the final-step N lying inside the terminal set CS_N is λ, we choose p as follows [26]:

$$p = \lambda^{\frac{1}{N}} \tag{36}$$

where $\lambda \in (0, 1)$ and N is the horizon length. That is to say that with this choice of p, the probability that the trajectory at the terminal time-step, N, will lie inside the terminal set, is at least $p^N = \lambda$.

IV.D.2. Robustification of Control Constraints

For the closed-loop system to be robust to control disturbances, we need to tighten the control constraint set such that the commanded control lying in the tightened constraint set implies that the true uncertain control lies in the original control constraint set, for all possible control disturbances within the bounded control disturbance set. Again, the tightened constraints need to be polytopic for robust controllable tube construction, but can be conic for forward rollout. Note that the robustified control constraint set needs to be considered in the forward rollout in the robust case.

Tightening of convex constraints in general is problem-dependent, since it depends on the specific structure of the constraint set under consideration; for example, a robustified halfspace constraint can be modeled as an SOC constraint [54, §4.4.2]. In §V.D, we provide one such way to achieve robustification of the convex control constraints with respect to the worst-case control disturbance for the autonomous precision landing problem.

IV.D.3. Robustification of State Constraints and Robust Set Recursion

While we explicitly robustify the control constraint set, as shown in §IV.D.2, the state constraint sets are implictly robustified via the backward set-recursion for robust controllable tube generation, given by Algorithm 4. Note that Algorithm 4 is similar to Algorithm 1, with the key differences being (i) the use of the robustified control constraint set, $\tilde{\mathcal{U}}_{\text{robust}}$, and (ii) the Pontryagin difference step, which implicitly robustifies the state constraints with respect to the effective disturbance sets, $\mathcal{W}_k$, $k = 1, \ldots, N$. Note that $\mathcal{W}_k$ (at each time-step) accounts for the influence of both the state and control disturbances on the evolution of the state.

Algorithm 4 Set Recursion for Robust Controllable Tube Generation

Inputs: N, A, B, d, $\tilde{\mathcal{X}}$, $\tilde{\mathcal{X}}_\mathrm{f}$, $\tilde{\mathcal{U}}_\mathrm{robust}$, $\mathcal{W}_{[1:N]}$

1: $\mathrm{CS}_N \leftarrow \tilde{\mathcal{X}}_\mathrm{f} \ominus \mathcal{W}_N$ ▷ $\ominus$: Pontryagin difference

2: **for** $k = (N-1), (N-2), \ldots, 2, 1$ **do**

3: $\quad \mathrm{CS}_k = \tilde{\mathcal{X}} \cap A^{-1}\left((\mathrm{CS}_{k+1} \ominus \mathcal{W}_k) \oplus \left(-B\tilde{\mathcal{U}}_\mathrm{robust} - d\right)\right)$ ▷ $\oplus$: Minkowski sum

4: **end for**

Return: $\mathrm{CS}_{1,\ldots,N}$

V. Autonomous Precision Landing

To demonstrate the controllable tube-based optimal and robust control framework that we propose in the previous sections, we consider an autonomous precision landing case study. We formulate the optimal control problem in §V.A, describe the closed-loop simulation setup in §V.B, and demonstrate free-final-time optimal control in §V.C and robust control in §V.D. We use `pycvxset` [28] for all the set-based computations.

V.A. Formulation

The original nonconvex continuous-time 3-DoF minimum-fuel precision landing guidance optimal control problem [29] is given by Problem P10. In Problem P10, t_f is the final time, $r(t) \in \mathbb{R}^3$ is the position,

Problem P10: Continuous-Time Precision Landing Problem (Nonconvex)

$$
\begin{aligned}
\underset{t_\mathrm{f},\, T}{\text{minimize}} \quad & -m(t_\mathrm{f}) && \text{(P10a)}\\
\text{subject to} \quad & \forall\, t \in [0, t_\mathrm{f}] && \\
& \dot{r}(t) = v(t) && \text{(P10b)}\\
& \dot{v}(t) = \frac{1}{m(t)}\, T(t) - g\, e_z && \text{(P10c)}\\
& \dot{m}(t) = -\alpha \left\| T(t) \right\|_2 && \text{(P10d)}\\
& T_\mathrm{min} \leq \left\| T(t) \right\|_2 \leq T_\mathrm{max} && \text{(P10e)}\\
& e_z^\top\, T(t) \geq \left\| T(t) \right\|_2 \cos\theta_\mathrm{max} && \text{(P10f)}\\
& H_\mathrm{GS}\, r(t) \leq h_\mathrm{GS} && \text{(P10g)}\\
& \left\| r(t) \right\|_\infty \leq r_\mathrm{max} && \text{(P10h)}\\
& \left\| v(t) \right\|_\infty \leq v_\mathrm{max} && \text{(P10i)}\\
& m_\mathrm{dry} \leq m(t) \leq m_\mathrm{wet} && \text{(P10j)}\\
& r(0) = r_\mathrm{i},\ \ v(0) = v_\mathrm{i},\ \ m(0) = m_\mathrm{wet} && \text{(P10k)}\\
& r(t_\mathrm{f}) = r_\mathrm{f},\ v(t_\mathrm{f}) = v_\mathrm{f},\ m(t_\mathrm{f}) \geq m_\mathrm{dry} && \text{(P10l)}
\end{aligned}
$$

$v(t) \in \mathbb{R}^3$ is the velocity, $m(t) \in \mathbb{R}$ is the vehicle mass, $T(t) \in \mathbb{R}^3$ is the thrust vector in Cartesian coordinates, $g \in \mathbb{R}_{++}$ is the acceleration due to gravity, α is the thrust-specific fuel consumption, T_min, $T_\mathrm{max} \in \mathbb{R}_{++}$ are the minimum and maximum thrust magnitude bounds, respectively, $e_z^\top := (0, 0, 1)$, θ_max is the maximum tilt angle from the vertical, $H_\mathrm{GS} \in \mathbb{R}^{n_h \times 3}$ and $h_\mathrm{GS} \in \mathbb{R}^{n_h}$ are the parameters defining the glideslope constraint (as the intersection of n_h halfspaces; see [35, Equations S10 and S11]), r_max, $v_\mathrm{max} \in \mathbb{R}_{++}$ are the component-wise bounds on position and velocity, respectively, m_wet, $m_\mathrm{dry} \in \mathbb{R}_{++}$ are the wet-mass and dry-mass of the vehicle, respectively, r_i, $r_\mathrm{f} \in \mathbb{R}^3$ are the initial and final conditions for the position, respectively, and v_i, $v_\mathrm{f} \in \mathbb{R}^3$ are the initial and final conditions for the velocity, respectively. Since this is a three-degree-of-freedom formulation, all quantities are defined relative to a reference frame fixed to the celestial body under consideration.

The autonomous precision landing problem has been extensively studied, and it has been shown that the nonconvex thrust lower bound constraint in Equation (P10e) can be losslessly convexified by means of a convex relaxation, such that the solution to the relaxed problem is a globally optimal solution to the original problem [29, 30, 35]. The thrust pointing constraint given by Equation (P10f) would be nonconvex for $\theta_{\max} > 90°$, but we only require $\theta_{\max} \leq 90°$. Note that the state constraints in Equations (P10h) and (P10i) are only included to ensure that the state is bounded (as required for controllable tube generation). Typically, $r_{\max}$ and $v_{\max}$ are set to large values, however, and are not expected to be active.

Problem P10 can be equivalently written in terms of the mass-normalized thrust by means of a log-mass transformation [29], with $u(t) := \frac{1}{m(t)} T(t)$ and $z(t) := \ln m(t)$. Consequently, the control magnitude bounds become:

$$T_{\min} e^{-z(t)} \leq \|u(t)\|_2 \leq T_{\max} e^{-z(t)} \tag{37}$$

and the log-mass dynamics are:

$$\dot{z}(t) = -\alpha \|u(t)\|_2 \tag{38}$$

both of which are nonconvex. First, we replace the time-varying bounds in Equation (37) with the following time-invariant bounds, such that the resulting constraint is a conservative approximation of the original constraint:

$$\frac{T_{\min}}{m_{\text{dry}}} \leq \|u(t)\|_2 \leq \frac{T_{\max}}{m_{\text{wet}}} \tag{39}$$

Note that the lower bound constraint in Equation (39) is still nonconvex. For this, we can employ one of the following two remedies:

(i) The standard convex relaxation to convexify the nonconvex lower bound [29, 30, 35], i.e.,

$$\sigma(t) \geq \frac{T_{\min}}{m_{\text{dry}}} \tag{40}$$

Note that this does not exactly fit the template of Problem P1, since this serves as a convex relaxation to the original nonconvex constraint. However, when implemented, we observe that the relaxation is tight in practice, i.e., lossless convexification holds [29, 30, 35]. Although we consider a polytopic *inner* approximation to the conic frustum defined by the control magnitude bounds, satisfaction of the polytopic constraint does not necessarily guarantee satisfaction of the original constraint—the lower bound constraint can be violated, since the polytopic approximation we consider is of a convex relaxation to a nonconvex constraint set. This has the adverse consequence of expanding the feasible control space—beyond that of the original control constraint set—in some regions. That said, the degree to which this constraint is violated can be arbitrarily controlled by choosing the number of points to be spread on the unit sphere. Further, this issue can be eliminated entirely by ensuring that the projection of the polytopic approximation onto the non-lifted space is an inner approximation to the original nonconvex set.

(ii) Replacing the lower bound constraint with the following conservative convex constraint:

$$e_z^\top u(t) \geq \frac{T_{\min}}{m_{\text{dry}}} \tag{41}$$

With the conservative constraint given by Equation (41), the control magnitude, and in turn, the slack variable, are implicitly lower-bounded by $\frac{T_{\min}}{m_{\text{dry}}}$. With Equation (41), the problem fits the template of Problem P1, in which case the only nonconvexity stems from the log-mass dynamics (Equation (38)). For the discretized version of this template, there exists a blanket lossless convexification guarantee [5]. We consider the conservative convex constraint given by Equation (41) in the remainder of this paper.

Further, we replace $\|u\|_2$ by σ in the corresponding pointing constraint, $e_z^\top u(t) \geq \|u(t)\|_2 \cos\theta_{\max}$ (even though it is convex), hence converting the second-order cone into an equivalent halfspace, which mitigates the need for an additional polytopic approximation.

The (Mayer) cost function, after the log-mass transformation, can be given by:

$$J := -z(t_\mathrm{f}) \tag{42}$$

which can be equivalently expressed as a running cost:

$$J = -\int_0^{t_\mathrm{f}} \dot{z}(\tau)\, d\tau = -\int_0^{t_\mathrm{f}} -\alpha\,\sigma(\tau)\, d\tau = \int_0^{t_\mathrm{f}} \alpha\,\sigma(\tau)\, d\tau \tag{43}$$

Now, the cost-to-go can be defined as follows:

$$c(t) := \int_t^{t_\mathrm{f}} \alpha\,\sigma(\tau)\, d\tau \tag{44}$$

Taking the time-derivative of Equation (44), we get:

$$\dot{c}(t) = \frac{d}{dt}\int_t^{t_\mathrm{f}} \alpha\,\sigma(\tau)\, d\tau = \frac{d}{dt}\int_{t_\mathrm{f}}^{t} -\alpha\,\sigma(\tau)\, d\tau = -\alpha\,\sigma(t) \tag{45}$$

From Equation (44), we can derive the boundary conditions for this auxiliary dynamical system:

$$c(0) = \int_0^{t_\mathrm{f}} \alpha\,\sigma(\tau)\, d\tau \tag{46a}$$

$$c(t_\mathrm{f}) = 0 \tag{46b}$$

Now, consider the following:

$$z(t_\mathrm{f}) = z(0) - \int_0^{t_\mathrm{f}} \alpha\,\sigma(\tau)\, d\tau \tag{47a}$$

$$\implies \int_0^{t} \alpha\,\sigma(\tau)\, d\tau = z(0) - z(t_\mathrm{f}) \tag{47b}$$

Substituting Equation (47b) in Equation (3a), we get:

$$c(0) = z(0) - z(t_\mathrm{f}) = \ln m_\mathrm{wet} - z(t_\mathrm{f}) \tag{48}$$

Collecting Equations (45), (46b), and (48), we have the following auxiliary dynamical system:

$$\dot{c}(t) = -\alpha\,\sigma(t) \tag{49a}$$

$$c(0) = \ln m_\mathrm{wet} - z(t_\mathrm{f}) \tag{49b}$$

$$c(t_\mathrm{f}) = 0 \tag{49c}$$

with the bounds on the cost-to-go state being:

$$0 \leq c(t) \leq \ln m_\mathrm{wet} - \ln m_\mathrm{dry} = \ln \frac{m_\mathrm{wet}}{m_\mathrm{dry}} \tag{50}$$

Although the right-hand-side (RHS) of Equation (49) is identical to the RHS of the log-mass dynamics given by Equation (38), Equations (49) represent a different system due to the different boundary conditions. In this work, we employ the standard procedure of augmenting the physical system with the auxiliary cost-to-go system, making the resulting system 8-dimensional, which, in turn, leads to 8-dimensional polytopes (CZs) in the controllable tube.

Remark 4. *The similarity between the log-mass dynamics (Equation (38)) and the cost-to-go dynamics (Equation (49)) can be exploited to eliminate the log-mass variable and dynamics entirely, and to equivalently rewrite the problem in terms of the cost-to-go variable and dynamics, in which case the polytopes in the controllable tube would be 7-dimensional, instead of 8-dimensional. Particularly, we observe (and can leverage the fact) that the log-mass and the cost-to-go are related by:*

$$c(t) = z(t) - z(t_\mathrm{f}) \tag{51}$$

with the bounds on the cost-to-go state given by Equation (50). Using this formulation would require additional considerations to account for the initial condition of the log-mass at every step; particularly, this would require either (i) taking the intersection of the controllable set with the halfspaces defined by the log-mass bounds (prior to solving the one-step optimal control problem), in which case potential infeasibility can be detected prior to solving the one-step OCP, or (ii) imposing the halfspaces directly in the one-step OCP, in which case potential infeasibility would be detected via solving the one-step OCP itself.

Given how well CZs scale with problem dimension, we do not observe a marked difference in performance with the formulation resulting from Remark 4, and hence retain the 8-dimensional formulation for simplicity. Note that the cost-to-go state, $c(t)$, serves as a proxy for the (instantaneous) "required fuel" to reach the target.

Finally, we discretize the log-mass-transformed and cost-to-go-augmented problem using a ZOH control parameterization, as described in §II.A—the discretized conic optimal control problem is given by Problem P11. The state constraints in Problem P11 are polytopic to begin with. The control constraint set, however, is conic, due to Equation (P11c)—the control input slack constraint that arises as a result of lossless convexification is a 4-dimensional quadratic cone (note that the following applies in both continuous time and in discrete time):

Problem P11: Discrete-Time Precision Landing Problem (Conic)

$$\begin{aligned}
\underset{\substack{N,\, c_1,\\ u_1,\ldots,u_{N-1},\\ \sigma_1,\ldots,\sigma_{N-1}}}{\text{minimize}} \quad & c_1 && && \text{(P11a)}\\
\text{subject to} \quad & (r_{k+1}, v_{k+1}, z_{k+1}, c_{k+1}) = A\cdot(r_k, v_k, z_k, c_k) + B\cdot(u_k, \sigma_k) + d, && k = 1,\ldots,N-1 && \text{(P11b)}\\
& \|u_k\|_2 \le \sigma_k && k = 1,\ldots,N-1 && \text{(P11c)}\\
& \sigma_k \le \frac{T_\mathrm{max}}{m_\mathrm{wet}} && k = 1,\ldots,N-1 && \text{(P11d)}\\
& e_z^\top u_k \ge \frac{T_\mathrm{min}}{m_\mathrm{dry}} && k = 1,\ldots,N-1 && \text{(P11e)}\\
& e_z^\top u_k \ge \sigma_k\,\cos\theta_\mathrm{max} && k = 1,\ldots,N-1 && \text{(P11f)}\\
& H_\mathrm{GS}\, r_k \le h_\mathrm{GS} && k = 2,\ldots,N-1 && \text{(P11g)}\\
& \|r_k\|_\infty \le r_\mathrm{max} && k = 2,\ldots,N-1 && \text{(P11h)}\\
& \|v_k\|_\infty \le v_\mathrm{max} && k = 2,\ldots,N-1 && \text{(P11i)}\\
& \ln m_\mathrm{dry} \le z_k \le \ln m_\mathrm{wet} && k = 2,\ldots,N-1 && \text{(P11j)}\\
& 0 \le c_k \le \ln\frac{m_\mathrm{wet}}{m_\mathrm{dry}} && k = 2,\ldots,N-1 && \text{(P11k)}\\
& r_1 = r_\mathrm{i},\ \ v_1 = v_\mathrm{i},\ \ z_1 = \ln m_\mathrm{wet},\ \ c_1 \le \ln\frac{m_\mathrm{wet}}{m_\mathrm{dry}} && && \text{(P11l)}\\
& r_N = r_\mathrm{f},\ v_N = v_\mathrm{f},\ z_N \ge \ln m_\mathrm{dry},\ c_N = 0 && && \text{(P11m)}
\end{aligned}$$

$$\mathcal{U}_\mathrm{slack} := \left\{(u,\,\sigma) \in \mathbb{R}^4 \;\middle|\; \|u\|_2 \le \sigma,\ \sigma \ge 0\right\} \tag{52}$$

where $u \in \mathbb{R}^3$ and $\sigma \in \mathbb{R}$. The set is closed and convex, but not compact, and a linear approximation of this set would yield an unbounded set. However, since the controllable tube generation method using constrained zonotopes requires the control input constraint set to be a bounded polytope, we also consider the upper

bound constraint on the control magnitude, which is a halfspace:

$$\mathcal{U}_{\text{UB}} := \left\{ (u,\, \sigma) \in \mathbb{R}^4 \;\middle|\; \sigma \leq \frac{T_{\max}}{m_{\text{wet}}} \right\} \tag{53}$$

Now, consider the intersection of the sets given by Equations (52) and (53), $\mathcal{U}_{\text{slack}}$ and $\mathcal{U}_{\text{UB}}$, respectively:

$$\mathcal{U}_{\text{CQC}} := \mathcal{U}_{\text{slack}} \cap \mathcal{U}_{\text{UB}} = \left\{ (u,\, \sigma) \in \mathbb{R}^4 \;\middle|\; \|u\|_2 \leq \sigma,\; 0 \leq \sigma \leq \frac{T_{\max}}{m_{\text{wet}}} \right\} \tag{54}$$

The set given by Equation (54), $\mathcal{U}_{\text{CQC}}$, is a compact quadratic cone. We compute a polytopic inner approximation of $\mathcal{U}_{\text{CQC}}$ by following the procedure described in §II.B.

With this approximation applied to the discretized precision landing guidance problem, we get the discrete-time polytopic precision landing guidance problem that fits the template of Problem P5, which is shown in Problem P12. Note that Equations (P11c) and (P11d) in Problem P11 are replaced by Equation (P12c) in Problem P12.

Problem P12: Discrete-Time Precision Landing Problem (Polytopic)

$$\begin{aligned}
\underset{\substack{N,\, c_1,\\ u_1,\ldots,u_{N-1},\\ \sigma_1,\ldots,\sigma_{N-1}}}{\text{minimize}} \quad & c_1 && && \text{(P12a)}\\
\text{subject to} \quad & (r_{k+1}, v_{k+1}, z_{k+1}, c_{k+1}) = A \cdot (r_k, v_k, z_k, c_k) + B \cdot (u_k, \sigma_k) + d, && k = 1, \ldots, N-1 && \text{(P12b)}\\
& (u_k, \sigma_k) \in \mathcal{U}_{\text{CQC}} && k = 1, \ldots, N-1 && \text{(P12c)}\\
& e_z^\top u_k \geq \frac{T_{\min}}{m_{\text{dry}}} && k = 1, \ldots, N-1 && \text{(P12d)}\\
& e_z^\top u_k \geq \sigma_k \cos\theta_{\max} && k = 1, \ldots, N-1 && \text{(P12e)}\\
& H_{\text{GS}}\, r_k \leq h_{\text{GS}} && k = 2, \ldots, N-1 && \text{(P12f)}\\
& \|r_k\|_\infty \leq r_{\max} && k = 2, \ldots, N-1 && \text{(P12g)}\\
& \|v_k\|_\infty \leq v_{\max} && k = 2, \ldots, N-1 && \text{(P12h)}\\
& \ln m_{\text{dry}} \leq z_k \leq \ln m_{\text{wet}} && k = 2, \ldots, N-1 && \text{(P12i)}\\
& 0 \leq c_k \leq \ln \frac{m_{\text{wet}}}{m_{\text{dry}}} && k = 2, \ldots, N-1 && \text{(P12j)}\\
& r_1 = r_{\text{i}},\;\; v_1 = v_{\text{i}},\;\; z_1 = \ln m_{\text{wet}},\;\; c_1 \leq \ln \frac{m_{\text{wet}}}{m_{\text{dry}}} && && \text{(P12k)}\\
& r_N = r_{\text{f}},\; v_N = v_{\text{f}},\; z_N \geq \ln m_{\text{dry}},\; c_N = 0 && && \text{(P12l)}
\end{aligned}$$

V.B. Closed-Loop Simulation Setup

In this subsection, we describe the problem and closed-loop simulation parameters in §V.B.1, and discuss the sources of approximation and conservatism in §V.B.2.

V.B.1. Problem Parameters

The problem parameter values chosen for the optimal and robust control simulations are listed in Table 3. In the table, Δt refers to the discretization sampling time, and n_{points} refers to the number of points to be spread on the 3D unit sphere for the polytopic approximation of the conic control constraint set.

The closed-loop robust control (Monte Carlo) simulation is run for a fixed final time, i.e., a fixed horizon length, N, in order to stay consistent with the adopted probabilistic framework. The 3-sigma control uncertainty is denoted by $\delta^u_{3\sigma}$—note that $R_u = \frac{\delta^u_{3\sigma}}{3}$ and $\Sigma^u = R_u^2\, I_3 \in \mathbb{R}^3$. The time-varying 3-sigma position and velocity uncertainties are denoted by $\delta^r_{3\sigma}(k)$ and $\delta^v_{3\sigma}(k)$, respectively, where $k = 1, \ldots, N$. Similarly, note that $\Sigma^x_k = \text{blkdiag}\left\{ \left(\frac{\delta^r_{3\sigma}(k)}{3}\right)^2 I_3,\, \left(\frac{\delta^v_{3\sigma}(k)}{3}\right)^2 I_3 \right\} \in \mathbb{R}^6$. Finally, we note that $\delta^r_{3\sigma}(1)$ corresponds to a 30

m initial 3-sigma navigation uncertainty in position, and $\delta^v_{3\sigma}(1)$ corresponds to a 0.6 $\mathrm{m\,s^{-1}}$ initial 3-sigma navigation uncertainty in velocity; the value for $\delta^u_{3\sigma}$ corresponds to roughly 0.5% the maximum control magnitude.

Parameter	Value
g	$1.625\ \mathrm{m\,s^{-2}}$
T_{full}	$10500\ \mathrm{kg\,m\,s^{-2}}$
$T_{\max}$	$0.8\,T_{\mathrm{full}}$
$T_{\min}$	$0.2\,T_{\mathrm{full}}$
m_{wet}	1905 kg
m_{dry}	1505 kg
$\theta_{\max}$	50°
$r_{\max}$	4000 m
$v_{\max}$	$100\ \mathrm{m\,s^{-1}}$
r_{f}	$(0,0,0)$ m
v_{f}	$(0,0,0)\ \mathrm{m\,s^{-1}}$
$\gamma_{\max}$	80°
H_{GS}	$\begin{bmatrix} \cos\gamma_{\max} & 0 & -\sin\gamma_{\max} \\ 0 & \cos\gamma_{\max} & -\sin\gamma_{\max} \\ -\cos\gamma_{\max} & 0 & -\sin\gamma_{\max} \\ 0 & -\cos\gamma_{\max} & -\sin\gamma_{\max} \end{bmatrix}$
h_{GS}	$(0,0,0,0)$

Parameter	Value
N	free
Δt	3 s
α	$0.00115\ \mathrm{s\,m^{-1}}$
r_{i}	$(875,0,635)$ m
v_{i}	$(40,0,-30)\ \mathrm{m\,s^{-1}}$
n_{points}	302
N	20
Δt	15 s
α	$0.0002875\ \mathrm{s\,m^{-1}}$
r_{i}	$(4000,4000,4000)$ m
v_{i}	$(-10,-10,-10)\ \mathrm{m\,s^{-1}}$
n_{points}	14
λ	0.95
$\delta^u_{3\sigma}$	$0.023\ \mathrm{m\,s^{-2}}$
$\delta^r_{3\sigma}(k)$	$1.5\,(N-k+1)$ m
$\delta^v_{3\sigma}(k)$	$0.03\,(N-k+1)\ \mathrm{m\,s^{-1}}$

Table 3: Problem parameters. Left: common; right top: closed-loop optimal control simulation; right bottom: closed-loop robust control (Monte Carlo) simulations, where $k = 1, \ldots, N$.

V.B.2. Sources of Approximation and Conservatism

Here, we catalog the sources of approximation and conservatism in the proposed closed-loop control framework. As shown in Equation (37), the original bounds on the control magnitude are (time-varying) nonlinear functions of the log-mass; instead, we only consider conservatively chosen time-invariant bounds, as shown in Equation (39). Further, we introduce conservatism in the control magnitude lower bound, as shown in Equation (41). Next, we consider a polytopic inner approximation to the control constraint set for controllable tube generation via set recursion, which effectively shrinks the controllable space—this approximation can be made arbitrarily accurate by choosing more points to spread on the unit sphere (see §II.B).

In the robust case, specifically, there are three additional sources of conservatism: (i) the independence assumption on the effective disturbance for robustification, in which case the controllable tube is robust to the worst-case disturbance at each time-step without accounting for coupling of the state disturbances in time, (ii) invocation of the triangle and reverse triangle inequalities to tighten the control constraints in §V.D.3, and (iii) the Pontryagin difference algorithm [24, Algorithm 3] in the set recursion for robust controllable tube generation, which returns an inner approximation of the true Pontryagin difference that worsens as the latent dimension of the constrained zonotope(s) increases, i.e., as the number of columns, n_g, of G in Equation (6) increases. Choosing more points on the unit sphere to obtain the polytopic approximation of the conic control set, for example, will lead to constrained zonotopes that are larger in latent dimension, thus making the Pontryagin difference more conservative. Note, however, that the conservatism introduced by (i), (ii), and (iii) is *in the right direction*, i.e., they lead to the robust controllable tube being *more* robust than necessary, thus preserving all the claimed probability guarantees.

V.C. Optimal Control: The Optimal Guidance Algorithm

We solve the deterministic, free-final-time (minimum-fuel) autonomous precision landing guidance problem using Algorithm 1 for controllable tube generation and Algorithm 3 for the forward rollout. Within the forward rollout, we solve Problem P6 as is and do not invoke Remark 1 here to demonstrate global optimality. For comparison, we also solve Problem P12 directly in an open-loop setting, and show the control magnitude profiles in Figure 3.

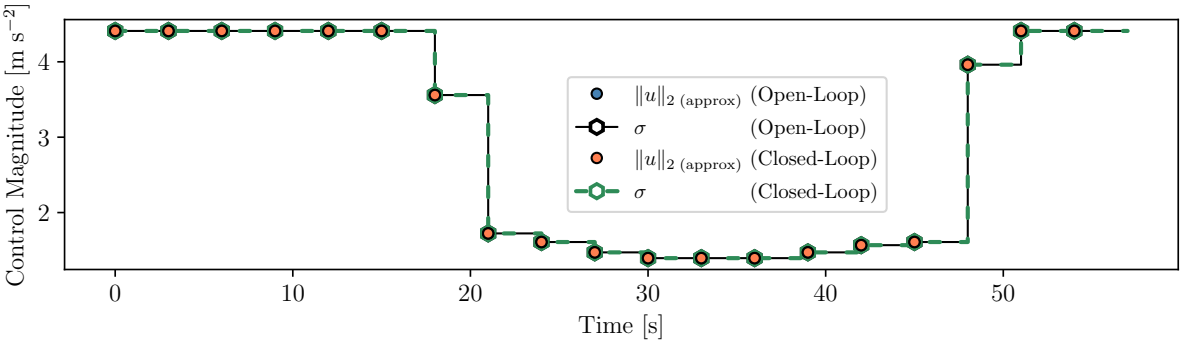

Figure 3: Algorithm 3—with the one-step optimal control problem in accordance with Problem P6—recovers a globally optimal solution to the polytopic open-loop *free-final-time* (minimum-fuel) problem (Problem P12), which is solved in concert with a golden section search.

V.D. Robust Control: Robustness to Navigation and Actuation Uncertainties

In this subsection, we describe robust control in the context of the autonomous precision landing problem. Specifically, we model navigation and actuation uncertainty in §V.D.1, describe practical aspects of robust controllable tube generation in §V.D.2, and describe robustification of the control and state constraints in §V.D.3 and §V.D.4, respectively. Finally, we present Monte Carlo simulation results in §V.D.5.

V.D.1. Navigation and Actuation Uncertainty Models

We assume that the estimated position and velocity are subject to additive navigation (state) uncertainty and that the control input is subject to additive actuation (control) uncertainty, both of which are modeled as Gaussian disturbances, in accordance with §IV.B. Note that we treat the augmented control slack variable, σ, such that it is not directly perturbed; instead, we robustify the original control constraints first and then convexify them (i.e., introduce σ). We also assume that the log-mass state and the cost-to-go state can be estimated perfectly (since they are functions of the *applied* control input and not the commanded control input).

The bounded disturbance sets for the uncertainties, as required by Algorithm 4, are constructed in accordance with §IV.C.

V.D.2. Robust Controllable Tube Generation

We schedule the state covariances such that the corresponding disturbance ellipses decrease in size over time (the standard deviations decrease linearly)—this accounts for the improvement in the accuracy of the navigation estimates as the system approaches the landing site (as more information is acquired by the sensors). In the robust controllable tube set recursion algorithm (Algorithm 4), the terminal set constrained zonotope needs to be full-dimensional and have a MinRow representation for the Pontryagin difference step to be computationally tractable [24, Definition 3 and Proposition 1]. To generate this full-dimensional terminal set, we first perform a deterministic set recursion (Algorithm 1) for one Δt-second time interval, and then proceed with robust set recursion. This has the added benefit of guaranteeing the existence of a solution to the original terminal set for any true state lying in the thus-generated full-dimensional terminal set, and can thus account for reserve fuel, for example, in an exact manner, without requiring conservative estimates. In practice, we observe that the conditions required for Pontryagin differencing to be tractable [24, Definition 3 and Proposition 1] do not hold when the set is generated via a one-step recursion. To mitigate

this, we perform a two-step recursion, but with a sampling time of $\frac{\Delta t}{2}$ instead (such that the effective interval is still Δt).

V.D.3. Robustification of Control Constraints

Here, we present one approach to robustifying the control constraint set with respect to the worst-case control disturbance.

Control Magnitude Bounds To robustify the control magnitude bound constraint, we first consider the original constraint given by Equation (39), and tighten the bounds with respect to the worst-case disturbance from the control disturbance set. Once the tightened bounds are obtained, the lower bound constraint is replaced with the conservative convex counterpart of Equation (41).

The control magnitude constraint, Equation (39), with actuation uncertainty, is given by:

$$\frac{T_{\min}}{m_{\text{dry}}} \leq \|u_k + w_k^u\|_2 \leq \frac{T_{\max}}{m_{\text{wet}}}, \quad k = 1, \ldots, N-1 \tag{55}$$

To be able to account for the slack variable that will be introduced for lossless convexification, we seek a representation of Equation (55) that is solely in terms of $\|u_k\|_2$. As such, we approximate the thrust magnitude constraint given by Equation (55) as follows.

Invoking the triangle and reverse triangle inequalities, we get:

$$|\|u_k\|_2 - \|w_k^u\|_2| \leq \|u_k + w_k^u\|_2 \leq \|u_k\|_2 + \|w_k^u\|_2 \tag{56}$$

We make the assumption that $\|w_k^u\|_2 \leq \|u_k\|_2$, i.e., the magnitude of the bounded disturbance will be less than or equal to the control input magnitude, to ensure that any constraint tightening we perform will not make the control magnitude constraint set infeasible. With this assumption, Equation (56) can be written as:

$$\|u_k\|_2 - \|w_k^u\|_2 \leq \|u_k + w_k^u\|_2 \leq \|u_k\|_2 + \|w_k^u\|_2 \tag{57}$$

Now, our goal is to represent Equation (57) such that it is solely in terms of u_k. For this, we consider the Equation (57) for all possible disturbances in the disturbance set:

$$\|u_k\|_2 - \|w_k^u\|_2 \leq \|u_k + w_k^u\|_2 \leq \|u_k\|_2 + \|w_k^u\|_2 \quad \forall\, w_k^u \in \mathcal{W}_u \tag{58a}$$

$$\iff \inf_{w_k^u \in \mathcal{W}_u} (\|u_k\|_2 - \|w_k^u\|_2) \leq \|u_k + w_k^u\|_2 \leq \sup_{w_k^u \in \mathcal{W}_u} (\|u_k\|_2 + \|w_k^u\|_2) \tag{58b}$$

$$\iff \|u_k\|_2 - \sup_{w_k^u \in \mathcal{W}_u} \|w_k^u\|_2 \leq \|u_k + w_k^u\|_2 \leq \|u_k\|_2 + \sup_{w_k^u \in \mathcal{W}_u} \|w_k^u\|_2 \tag{58c}$$

Therefore,

$$\|u_k\|_2 - \psi_u \leq \|u_k + w_k^u\|_2 \leq \|u_k\|_2 + \psi_u \tag{59}$$

where,

$$\psi_u := \sup_{w_k^u \in \mathcal{W}_u} \|w_k^u\|_2 \tag{60}$$

Now, the control bound constraint can be rewritten (approximated) in terms of u_k as follows:

$$\frac{T_{\min}}{m_{\text{dry}}} \leq \|u_k\|_2 - \psi_u \leq \|u_k + w_k^u\|_2 \leq \|u_k\|_2 + \psi_u \leq \frac{T_{\max}}{m_{\text{wet}}} \tag{61a}$$

$$\implies \frac{T_{\min}}{m_{\text{dry}}} + \psi_u \leq \|u_k\|_2 \leq \frac{T_{\max}}{m_{\text{wet}}} - \psi_u \tag{61b}$$

Equation (61b) represents a conservative tightening of the thrust magnitude bound constraint set, in that it is guaranteed to be smaller than or equal to the tightened constraint set that would have resulted had the approximation not been introduced.

Further, we derive a closed-form expression for ψ_u by making the assumption that the control disturbance is isotropic, in that the projection of the effective disturbance ellipsoid (given by Equation (34b)) onto the control dimensions is a 3-dimensional ball, which in turn implies that Σ_u in Equation (19) is of the form $R_u^2\, I_3$, where $R_u > 0$ is the radius of the 3D ball. Therefore:

$$\psi_u = R_u \tag{62}$$

$$\implies \frac{T_{\min}}{m_{\text{dry}}} + R_u \leq \|u_k\|_2 \leq \frac{T_{\max}}{m_{\text{wet}}} - R_u \tag{63}$$

Note that the control disturbance can be ellipsoidal as well if desired, but we make the isotropic assumption for simplicity. Finally, we (i) retain the upper bound constraint in Equation (63), but replace the 2-norm term with the slack variable, and (ii) replace the lower bound constraint with the conservative convex constraint (see Equation (41)), to yield the following tightened constraints:

$$\sigma_k \leq \frac{T_{\max}}{m_{\text{wet}}} - R_u \tag{64a}$$

$$e_z^\top u_k \geq \frac{T_{\min}}{m_{\text{dry}}} + R_u \tag{64b}$$

Thrust Pointing As with the control magnitude bound constraints, the thrust pointing constraint with actuation uncertainty, for $k = 1, \ldots, N-1$, is robustified and reformulated as follows:

$$e_z^\top (u_k + w_k^u) \geq \|u_k + w_k^u\|_2 \cos\theta_{\max}, \quad \forall\, w_k^u \in \mathcal{W}_u \tag{65a}$$

$$\iff \inf_{w_k^u \in \mathcal{W}_u} e_z^\top (u_k + w_k^u) \geq \sup_{w_k^u \in \mathcal{W}_u} \|u_k + w_k^u\|_2 \cos\theta_{\max} \tag{65b}$$

$$\impliedby \inf_{w_k^u \in \mathcal{W}_u} (e_z^\top u_k + e_z^\top w_k^u) \geq \sup_{w_k^u \in \mathcal{W}_u} (\|u_k\|_2 + \|w_k^u\|_2) \cos\theta_{\max} \tag{65c}$$

$$\iff e_z^\top u_k + \inf_{w_k^u \in \mathcal{W}_u} e_z^\top w_k^u \geq (\|u_k\|_2 + \sup_{w_k^u \in \mathcal{W}_u} \|w_k^u\|_2) \cos\theta_{\max} \tag{65d}$$

$$\iff e_z^\top u_k - \psi_u \geq (\|u_k\|_2 + \psi_u) \cos\theta_{\max} \tag{65e}$$

$$\iff e_z^\top u_k - R_u \geq (\|u_k\|_2 + R_u) \cos\theta_{\max} \tag{65f}$$

$$\iff e_z^\top u_k - R_u \geq (\sigma_k + R_u) \cos\theta_{\max} \tag{65g}$$

V.D.4. Robustification of State Constraints

The position and velocity states are assumed to be subject to time-varying uncertainty. While the log-mass and cost-to-go states are assumed to be estimated perfectly, and while we do not explicitly subject the augmented control slack variable, σ, to uncertainty, the influence of actuation uncertainty on them calls for additional consideration.

Specifically, since there are no explicit sources of disturbance affecting the log-mass and cost-to-go states, the Pontryagin difference step in the robust set recursion algorithm (Algorithm 4) is agnostic to the influence of control disturbances on them. As such, we need to "manually" ensure that the bounds on these states are appropriately tightened.

The log-mass dynamics with actuation uncertainty are given as follows, for $k = 1, \ldots, N-1$:

$$z_{k+1} = z_k - \alpha \|u_k + w_k^u\|_2 \tag{66}$$

We consider the worst-case mass depletion and for this, we consider the most adversarial disturbance. From the RHS of Equation (59), since $\alpha > 0$, we get:

$$z_{k+1} = z_k - \alpha (\|u_k\|_2 + \psi_u) = z_k - \alpha (\|u_k\|_2 + R_u) \tag{67}$$

Similarly, the worst-case cost-to-go "depletion", for $k = 1, \ldots, N-1$, would be:

$$c_{k+1} = c_k - \alpha (\|u_k\|_2 + R_u) \tag{68}$$

Rather than explicitly tightening the bounds for these states—which would result in time-varying state constraint sets—we directly consider the worst-case depletion dynamics given by Equations (67) and (68) in the robust set recursion (with the original bounds)—this is an equivalent approach that implicitly achieves the effect of tightening the bounds, since both z and c are strictly monotonically decreasing quantities.

The Pontryagin difference step in Algorithm 4 accounts for robustification of the constraints on the remaining states that are influenced by control and/or state disturbances.

V.D.5. Monte Carlo Simulation

To demonstrate the closed-loop robust control framework and highlight the probabilistic guarantees, we perform a Monte Carlo simulation with the problem parameters given in Table 3. For the same problem parameters (and boundary conditions), we run 100 cases with different random disturbance sequences sampled from Gaussian distributions, in accordance with §IV. We set the probability that the terminal state estimate, $\tilde{y}_N$, lies in the robustified terminal set, $\tilde{\mathcal{X}}_\mathrm{f} \ominus \mathcal{W}_N$—which corresponds to the probability that the true terminal state, y_N, lies in the true terminal set, $\tilde{\mathcal{X}}_\mathrm{f}$—to be $\lambda = 95\%$. In other words, the true terminal state is guaranteed to lie in the true terminal set with an 95% probability. We plot the terminal values of the true position in Figure 4 and the true velocity in Figure 5; all 100 trials are successful, in that all the true terminal state values lie inside the true terminal set in all 100 trials. Note that the success-rate is 100% despite the requested probability only being 95%, owing to the conservatism in the robustification (in the right direction).

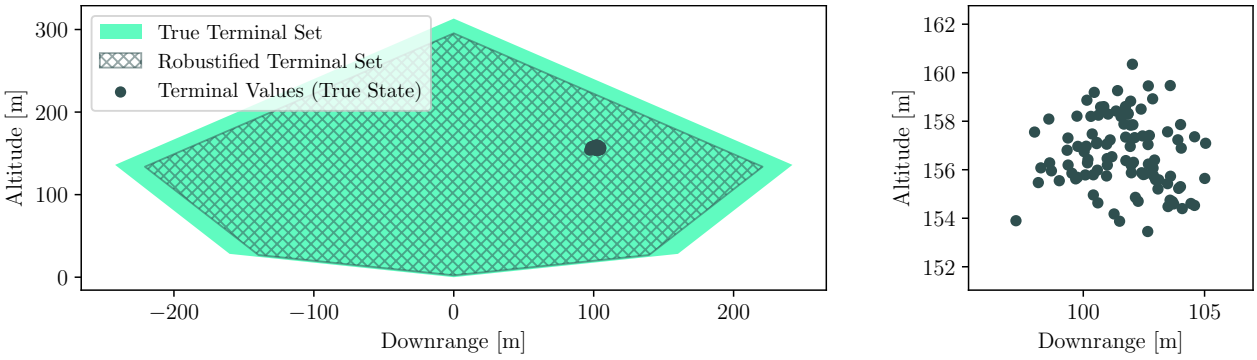

Figure 4: Terminal position values from the Monte Carlo simulation, projected onto the x-z plane.

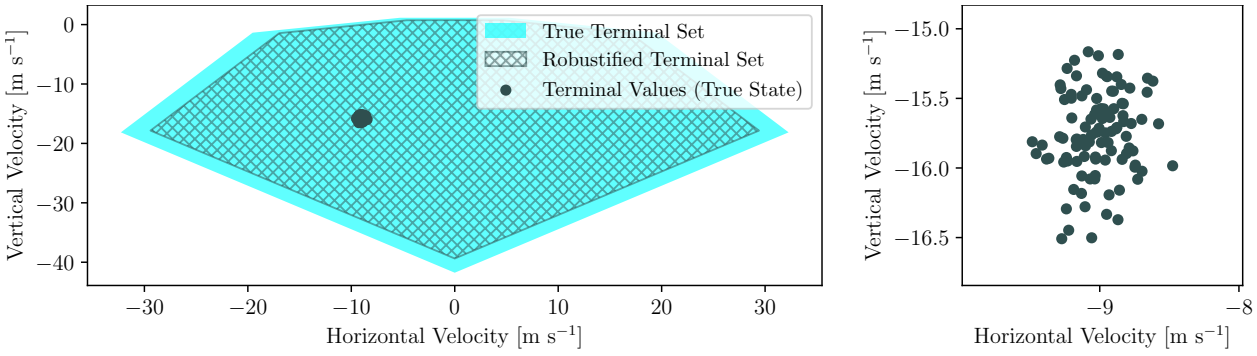

Figure 5: Terminal velocity values from the Monte Carlo simulation, projected onto the x-z plane.

Conclusions

We have developed a unified computational framework for closed-loop optimal and robust predictive control for autonomous systems. Motivated by the autonomous precision landing problem, we have proposed a set-based framework for free-final-time optimal control and robust control with respect to both state and

control uncertainty. The framework is based on dynamic programming over controllable tubes, achieving computational tractability using constrained zonotopes, a parameterization of compact convex polytopes that enables efficient high-dimensional set operations. The proposed control architecture has been validated through a comprehensive autonomous precision landing case study.

This work aims at bridging the gap between traditional open-loop guidance approaches and closed-loop control methods for applications such as precision landing.

Avenues for future research include achieving *resilient control*—i.e., ensuring safety of the system in the face of unmodeled uncertainties such as disruptions and faults [58, 59, 60]—using the developed set-based control architecture, generalizing the class of problems considered, reducing the number of sources of approximation, and implementing the computational framework via high-performance code to assess onboard viability for autonomous systems in terms of memory requirements and solve-times.

Acknowledgment

The authors gratefully acknowledge Kento Tomita for his feedback on an initial draft and Behçet Açıkmeşe for his generous guidance, insightful comments, and valuable discussions throughout the development of this work.

References

[1] Starek, J. A., Açıkmeşe, B., Nesnas, I. A., and Pavone, M., "Spacecraft autonomy challenges for next-generation space missions," *Advances in control system technology for aerospace applications*, Springer, 2015, pp. 1–48.

[2] Blackmore, L., "Autonomous precision landing of space rockets," *Frontiers of Engineering: Reports on Leading-Edge Engineering from the 2016 Symposium*, Vol. 46, The Bridge Washington, DC, USA, 2016, pp. 15–20.

[3] Carson, J. M., Munk, M. M., Sostaric, R. R., Estes, J. N., Amzajerdian, F., Blair, J. B., Rutishauser, D. K., Restrepo, C. I., Dwyer-Cianciolo, A. M., Chen, G., et al., "The SPLICE project: Continuing NASA development of GN&C technologies for safe and precise landing," *AIAA SciTech 2019 forum*, 2019, p. 0660.

[4] Malyuta, D., Yu, Y., Elango, P., and Açıkmeşe, B., "Advances in trajectory optimization for space vehicle control," *Annual Reviews in Control*, Vol. 52, 2021, pp. 282–315.

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

[6] Scott, J. K., Raimondo, D. M., Marseglia, G. R., and Braatz, R. D., "Constrained zonotopes: A new tool for set-based estimation and fault detection," *Automatica*, Vol. 69, 2016, pp. 126–136.

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

[8] Bellman, R., "The theory of dynamic programming," *Bulletin of the American Mathematical Society*, Vol. 60, No. 6, 1954, pp. 503–515.

[9] Bertsekas, D., *Dynamic programming and optimal control: Volume I*, Vol. 4, Athena scientific, 2012.

[10] Garcia, C. E., Prett, D. M., and Morari, M., "Model predictive control: Theory and practice—A survey," *Automatica*, Vol. 25, No. 3, 1989, pp. 335–348.

[11] Raimondo, D. M., Limon, D., Lazar, M., Magni, L., and ndez Camacho, E. F., "Min-max model predictive control of nonlinear systems: A unifying overview on stability," *European Journal of Control*, Vol. 15, No. 1, 2009, pp. 5–21.

[12] Mayne, D. Q., Kerrigan, E. C., Van Wyk, E., and Falugi, P., "Tube-based robust nonlinear model predictive control," *International journal of robust and nonlinear control*, Vol. 21, No. 11, 2011, pp. 1341–1353.

[13] Malyuta, D., Açıkmeşe, B., and Cacan, M., "Robust model predictive control for linear systems with state and input dependent uncertainties," *2019 American Control Conference (ACC)*, IEEE, 2019, pp. 1145–1151.

[14] Bellman, R., "Dynamic programming," *Science*, Vol. 153, No. 3731, 1966, pp. 34–37.

[15] Powell, W. B., *Approximate Dynamic Programming: Solving the curses of dimensionality*, Vol. 703, John Wiley & Sons, 2007.

[16] Powell, W. B., "What you should know about approximate dynamic programming," *Naval Research Logistics (NRL)*, Vol. 56, No. 3, 2009, pp. 239–249.

[17] Weiss, A., Baldwin, M., Erwin, R. S., and Kolmanovsky, I., "Model predictive control for spacecraft rendezvous and docking: Strategies for handling constraints and case studies," *IEEE Transactions on Control Systems Technology*, Vol. 23, No. 4, 2015, pp. 1638–1647.

[18] Açıkmeşe, B., Carson III, J. M., and Bayard, D. S., "A robust model predictive control algorithm for incrementally conic uncertain/nonlinear systems," *International Journal of Robust and Nonlinear Control*, Vol. 21, No. 5, 2011, pp. 563–590.

[19] Blanchini, F. and Miani, S., *Set-theoretic methods in control*, Springer, 2015.

[20] Eren, U., Dueri, D., and Açıkmeşe, B., "Constrained reachability and controllability sets for planetary precision landing via convex optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 38, No. 11, 2015, pp. 2067–2083.

[21] Srinivas, N., Vinod, A. P., Di Cairano, S., and Weiss, A., "Lunar Landing with Feasible Divert using Controllable Sets," *AIAA SCITECH 2024 Forum*, 2024, p. 0324.

[22] Lishkova, Y., Vinod, A., Di Cairano, S., and Weiss, A., "Divert-feasible lunar landing under navigational uncertainty," *2024 IEEE Conference on Decision and Control (CDC)*, IEEE, 2024, pp. 1–8.

[23] Vinod, A. P., Weiss, A., and Di Cairano, S., "Inscribing and separating an ellipsoid and a constrained zonotope: Applications in stochastic control and centering," *Proc. Conf. Dec. & Ctrl*, 2024, pp. 1–8.

[24] Vinod, A. P., Weiss, A., and Di Cairano, S., "Projection-free computation of robust controllable sets with constrained zonotopes," *Automatica*, Vol. 175, 2025, pp. 112211.

[25] Vinod, A. P. and Oishi, M. M., "Stochastic reachability of a target tube: Theory and computation," *Automatica*, Vol. 125, 2021, pp. 109458.

[26] Gleason, J. D., Vinod, A. P., and Oishi, M. M., "Lagrangian approximations for stochastic reachability of a target tube," *Automatica*, Vol. 128, 2021, pp. 109546.

[27] Vinod, A. P., Weiss, A., and Di Cairano, S., "Abort-safe spacecraft rendezvous under stochastic actuation and navigation uncertainty," *2021 60th IEEE Conference on Decision and Control (CDC)*, IEEE, 2021, pp. 6620–6625.

[28] Vinod, A. P., "pycvxset: A Python package for convex set manipulation," *Proceedings of the American Control Conference (ACC), Denver, USA*, 2025.

[29] 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.

[30] Açıkmeşe, B., Carson, J. M., and Blackmore, L., "Lossless convexification of nonconvex control bound and pointing constraints of the soft landing optimal control problem," *IEEE transactions on control systems technology*, Vol. 21, No. 6, 2013, pp. 2104–2113.

[31] Ridderhof, J. and Tsiotras, P., "Uncertainty quantication and control during mars powered descent and landing using covariance steering," *2018 AIAA Guidance, Navigation, and Control Conference*, 2018, p. 0611.

[32] Ridderhof, J. and Tsiotras, P., "Minimum-fuel powered descent in the presence of random disturbances," *AIAA scitech 2019 forum*, 2019, p. 0646.

[33] Ridderhof, J. and Tsiotras, P., "Minimum-fuel closed-loop powered descent guidance with stochastically derived throttle margins," *Journal of Guidance, Control, and Dynamics*, Vol. 44, No. 3, 2021, pp. 537–547.

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

[35] Malyuta, D., Reynolds, T. P., Szmuk, M., Lew, T., Bonalli, R., Pavone, M., and Açıkmeşe, B., "Convex optimization for trajectory generation: A tutorial on generating dynamically feasible trajectories reliably and efficiently," *IEEE Control Systems Magazine*, Vol. 42, No. 5, 2022, pp. 40–113.

[36] Lobo, M. S., Vandenberghe, L., Boyd, S., and Lebret, H., "Applications of second-order cone programming," *Linear algebra and its applications*, Vol. 284, No. 1-3, 1998, pp. 193–228.

[37] Ben-Tal, A. and Nemirovski, A., "On polyhedral approximations of the second-order cone," *Mathematics of Operations Research*, Vol. 26, No. 2, 2001, pp. 193–205.

[38] Vinel, A. and Krokhmal, P. A., "Polyhedral approximations in p-order cone programming," *Optimization Methods and Software*, Vol. 29, No. 6, 2014, pp. 1210–1237.

[39] Lipp, T. and Boyd, S., "Variations and extension of the convex–concave procedure," *Optimization and Engineering*, Vol. 17, 2016, pp. 263–287.

[40] Tiwary, H. R., "On the hardness of computing intersection, union and Minkowski sum of polytopes," *Discrete & Computational Geometry*, Vol. 40, 2008, pp. 469–479.

[41] Raghuraman, V. and Koeln, J. P., "Set operations and order reductions for constrained zonotopes," *Automatica*, Vol. 139, 2022, pp. 110204.

[42] Meindl, B. and Templ, M., "Analysis of commercial and free and open source solvers for linear optimization problems," *Eurostat and Statistics Netherlands within the project ESSnet on common tools and harmonised methodology for SDC in the ESS*, Vol. 20, 2012, pp. 64.

[43] Gearhart, J. L., Adair, K. L., Durfee, J. D., Jones, K. A., Martin, N., and Detry, R. J., "Comparison of open-source linear programming solvers," 2013.

[44] Huangfu, Q. and Hall, J. J., "Parallelizing the dual revised simplex method," *Mathematical Programming Computation*, Vol. 10, No. 1, 2018, pp. 119–142.

[45] Applegate, D., Díaz, M., Hinder, O., Lu, H., Lubin, M., O'Donoghue, B., and Schudy, W., "PDLP: A Practical First-Order Method for Large-Scale Linear Programming," *arXiv preprint arXiv:2501.07018*, 2025.

[46] Domahidi, A., Chu, E., and Boyd, S., "ECOS: An SOCP solver for embedded systems," *2013 European control conference (ECC)*, IEEE, 2013, pp. 3071–3076.

[47] Yu, Y., Elango, P., Açıkmeşe, B., and Topcu, U., "Extrapolated proportional-integral projected gradient method for conic optimization," *IEEE Control Systems Letters*, Vol. 7, 2022, pp. 73–78.

[48] Goulart, P. J. and Chen, Y., "Clarabel: An interior-point solver for conic programs with quadratic objectives," *arXiv preprint arXiv:2405.12762*, 2024.

[49] Chari, G. M. and Açikmeşe, B., "QOCO: A quadratic objective conic optimizer with custom solver generation," *arXiv preprint arXiv:2503.12658*, 2025.

[50] Açıkmeşe, B. and Ploen, S., "A powered descent guidance algorithm for Mars pinpoint landing," *AIAA guidance, navigation, and control conference and exhibit*, 2005, p. 6288.

[51] Açıkmeşe, B., Scharf, D., Blackmore, L., and Wolf, A., "Enhancements on the convex programming based powered descent guidance algorithm for Mars landing," *AIAA/AAS astrodynamics specialist conference and exhibit*, 2008, p. 6426.

[52] Bertsekas, D. P., "Nonlinear programming," *Journal of the Operational Research Society*, Vol. 48, No. 3, 1997, pp. 334–334.

[53] Kochenderfer, M. J. and Wheeler, T. A., *Algorithms for optimization*, Mit Press, 2019.

[54] Boyd, S. and Vandenberghe, L., *Convex Optimization*, Cambridge University Press, Cambridge, England, March 2004.

[55] Billingsley, P., *Probability and Measure*, Wiley Series in Probability & Mathematical Statistics: Probability & Mathematical Statistics, John Wiley & Sons, Nashville, TN, May 1995.

[56] Tong, Y. L., *The multivariate normal distribution*, Springer Science & Business Media, 2012.

[57] Kurzhanskiy, A. A. and Varaiya, P., "Ellipsoidal toolbox (ET)," *Proceedings of the 45th IEEE Conference on Decision and Control*, IEEE, 2006, pp. 1498–1503.

[58] Rieger, C. G., Gertman, D. I., and McQueen, M. A., "Resilient control systems: Next generation design research," *2009 2nd Conference on Human System Interactions*, IEEE, 2009, pp. 632–636.

[59] Chamon, L. F., Amice, A., Paternain, S., and Ribeiro, A., "Resilient control: Compromising to adapt," *2020 59th IEEE Conference on Decision and Control (CDC)*, IEEE, 2020, pp. 5703–5710.

[60] Elango, P., Sarsilmaz, S. B., and Açıkmeşe, B., "Deferred-Decision Trajectory Optimization," *arXiv preprint arXiv:2502.06623*, 2025.