Trajectory Planning for Safe Dual Control with Active Exploration

Kaleb Ben Naveed[1], Manveer Singh[1], Devansh R. Agrawal[1], and Dimitra Panagou[1,2]

***Abstract*— Planning safe trajectories under model uncertainty is a fundamental challenge. Robust planning ensures safety by considering worst-case realizations, yet ignores uncertainty reduction and leads to overly conservative behavior. Actively reducing uncertainty on-the-fly during a nominal mission defines the dual control problem. Most approaches address this by adding a weighted exploration term to the cost, tuned to trade off the nominal objective and uncertainty reduction, but without formal consideration of when exploration is beneficial. Moreover, safety is enforced in some methods but not in others. We study a budget-constrained dual control problem, where uncertainty is reduced subject to safety and a mission-level cost budget that limits the allowable degradation in task performance due to exploration. In this work, we propose `Dual-gatekeeper`, a framework that integrates robust planning with active exploration under formal guarantees of safety and budget feasibility. The key idea is that exploration is pursued only when it provides a verifiable improvement without compromising safety or violating the budget, enabling the system to balance immediate task performance with long-term uncertainty reduction in a principled manner. We provide two implementations of the framework based on different safety mechanisms and demonstrate its performance on quadrotor navigation and autonomous car racing case studies under parametric uncertainty.**

I. Introduction

Planning safe trajectories in the presence of model uncertainty is a fundamental challenge in control and robotics [1]–[4]. In many applications, the system must execute a nominal task such as reaching a goal or tracking a reference trajectory [5]–[8], while its dynamics depend on uncertain parameters (e.g., aerodynamic drag or tire–road friction). If these uncertainties are not properly accounted for during planning, the resulting trajectories may violate safety constraints.

Robust planning and control methods including tube-based MPC [2], [9], sampling-based approaches [3], [10], control barrier function (CBF) methods [11], [12], and contraction-based approaches [4], [13] address this issue by explicitly accounting for model uncertainty when enforcing safety constraints. However, these approaches actively do not consider how to reduce uncertainty in the unknown model parameters. As a result, the nominal task must be executed under conservative safety margins, which can degrade performance.

A complementary line of work studies the *dual control* problem [14], where the controller must simultaneously perform a task and reduce uncertainty in unknown but learnable model parameters. Existing approaches typically address this objective in one of two ways. Some explicitly encourage exploration by augmenting the control objective with information-seeking or uncertainty-reduction terms, thereby generating trajectories that are informative about unknown parameters [15]–[17]. Others rely on passive uncertainty reduction, where parameter estimates are updated only from data collected along nominal task execution, without deliberately selecting informative trajectories [16], [18], [19]. While these methods have shown that uncertainty can be reduced during control, most do not provide a principled mechanism to determine when exploration is actually worthwhile relative to nominal task progress, especially under constraints.

In this work, we study a *budget-constrained variant of the dual control problem*, in which the system must (i) satisfy state and input constraints under bounded model uncertainty, (ii) actively reduce parametric uncertainty through exploration, and (iii) ensure that the total mission cost remains within a prescribed budget, which represents the maximum allowable degradation in task performance a user is willing to tolerate for enhanced model information. This formulation captures practical scenarios in which exploration may improve long-term performance, but only if it can be certified to not compromise safety or incur excessive cost.

We propose a `Dual-gatekeeper` framework that integrates safety, active exploration, and budget feasibility at the architectural level. The proposed framework is inspired by the `gatekeeper` paradigm [9]. At each planning cycle, `Dual-gatekeeper` first computes a conservative, robust mission trajectory that guarantees safety while executing the task objective. In parallel, a set of informative candidate trajectories is generated to promote the identification of uncertain parameters. Each candidate is assigned a score based on its predicted uncertainty reduction and mission cost. A candidate is committed only if its execution is certified to remain safe and its predicted cost does not exceed the prescribed mission budget; otherwise, the system continues to execute the conservative mission trajectory. As a result, exploration is treated as a verifiable decision rather than an implicit optimization trade-off.

Contributions: In this paper, we

- introduce a framework for safe dual control that determines when executing an informative trajectory is beneficial relative to the nominal task, rather than incorporating information-seeking behavior as a weighted term in the objective. The framework ensures that informative trajectories are executed only when safety is preserved and the resulting deviation does not incur excessive cost;

*The authors would like to acknowledge the support of the National Science Foundation (NSF) under grant no. 2223845 and grant no. 1942907.

[1]Department of Robotics, University of Michigan, Ann Arbor, MI, 48109 USA. {kbnaveed@umich.edu}

[2]Department of Aerospace Engineering, University of Michigan, Ann Arbor, MI, 48109 USA.

- demonstrate the modularity of the framework by instantiating it with different safety mechanisms, including tube-based robust MPC [2] and the `gatekeeper` architecture [9], and evaluate it in two case studies: autonomous car racing and quadrotor navigation.

II. Related Work

This section reviews prior work on (i) robust safety under model uncertainty, and (ii) dual control and uncertainty reduction during task execution.

A. Robust Safety Under Model Uncertainty

Robust planning and control methods address safety under model uncertainty by constructing trajectories or feedback policies that guarantee constraint satisfaction despite uncertain dynamics, disturbances, or model mismatch. Tube-based model predictive control (MPC) methods enforce safety by tightening constraints and maintaining the closed-loop state within a robust tube around a nominal trajectory; examples include dynamic tube MPC and related extensions [2], [18], [20], [21]. Sampling-based approaches provide an alternative route to robustness by approximating chance or risk constraints using scenario-based formulations such as particle control or sample average approximation (SAA) [3], [10], [22]. Safety can also be enforced using control barrier function (CBF) methods, which impose barrier constraints to guarantee forward invariance of safe sets under uncertainty and disturbances [11], [12], [23], [24]. Complementary approaches based on contraction theory provide robustness certificates by establishing incremental stability through contraction metrics that certify the existence of safe feedback policies [4], [13]. While these approaches provide strong safety guarantees, they typically treat uncertainty as fixed and focus on ensuring safe task execution, rather than explicitly reasoning about how exploratory actions could reduce epistemic uncertainty to improve downstream performance.

B. Dual Control and Active Uncertainty Reduction

A complementary perspective is offered by the dual control problem, which explicitly balances exploitation (achieving the control objective) with exploration (actively reducing uncertainty) [14]. In many control problems, uncertainty can be divided into two components: *aleatoric uncertainty*, which reflects irreducible disturbances or noise, and *epistemic uncertainty*, which arises from unknown but learnable model parameters. Dual control primarily focuses on the latter by selecting control actions that both accomplish the task and generate informative data that improve the model. A wide range of approaches have been proposed along these lines [15]–[19], [25]–[37]. Some methods decouple learning and control by first reducing model uncertainty through a dedicated exploration phase and subsequently computing robust control policies based on the learned model [25]–[28]. Other approaches rely on *passive* uncertainty reduction, where model parameters are updated only from data naturally collected while executing the nominal control policy [6], [16], [18], [19], [38]. More recent work considers *active* exploration strategies that deliberately generate informative trajectories while simultaneously pursuing the mission objective [15], [17], [29], [30].

Another line of work considers hybrid reinforcement learning methods that balance exploration and exploitation through optimism and pessimism, using optimistic components to encourage broader exploration and pessimistic components to improve value estimation and stabilize learning. These methods incorporate the optimism–pessimism tradeoff directly within the actor-critic learning process through coupled actors, critics, or update rules [39]–[41].

However, in many of these approaches, the exploration–performance tradeoff is embedded directly within the optimization objective or the learning update. As a result, exploration is encouraged implicitly, either through weighted uncertainty-reduction terms or through coupled optimistic–pessimistic updates, rather than being treated as a separate decision. While this can promote exploratory behavior, it does not provide a principled mechanism for determining when exploration should be preferred if it conflicts with safety guarantees or mission-level cost constraints.

C. Positioning of This Work

This work connects robust safety methods with uncertainty-reducing decision making. Rather than introducing exploration through weighted objective terms, we consider an architecture in which informative trajectories are treated as candidate decisions that must satisfy safety and mission-level cost requirements before execution. The system therefore maintains a conservative mission trajectory that guarantees safe task completion while evaluating informative candidate trajectories that may reduce parametric uncertainty. The proposed `Dual-gatekeeper` framework commits to a candidate only if its safety can be certified and its predicted mission cost remains within the allowed budget; otherwise, execution continues along the conservative trajectory.

A recent work [42] studied this problem by generating informative candidate trajectories and comparing them against a conservative baseline trajectory to determine whether they can safely reduce parameter uncertainty without violating a mission-level cost budget. In that work, the framework was formulated using a finite-horizon backup trajectory, assumed to be available over the remaining mission horizon. This work extends that formulation as follows:

- We generalize the framework from a finite-horizon formulation, in which a safe backup trajectory must be known over the remaining mission horizon, to an infinite-horizon formulation that no longer requires such full-horizon backup information.
- The modularity of the framework is demonstrated through two safety instantiations. In the first, the framework is implemented using tube MPC, where safety is enforced by maintaining the closed-loop state within a robust tube around a nominal trajectory. In the second, the framework is implemented using a robust `gatekeeper` architecture resembling filtering-based

safety methods that modify candidate trajectories to ensure constraint satisfaction under uncertainty.
- Two case studies, namely quadrotor navigation and autonomous car racing, demonstrate these instantiations and illustrate how the same architecture can be integrated with different safety mechanisms while enabling uncertainty-reducing exploration.

III. Preliminaries & Problem Formulation

A. Notation

Let $\mathbb{R}$, $\mathbb{R}_{\geq 0}$, and $\mathbb{R}_{>0}$ denote the reals, non-negative reals, and positive reals, and $\mathbb{S}^n_{++}$ the symmetric positive definite matrices in $\mathbb{R}^{n\times n}$

B. System Model

Consider a class of nonlinear control-affine systems with parametric uncertainty and bounded additive disturbance:

$$\dot{x} = f_0(x) + F(x)\theta_f + (g_0(x) + G(x)\theta_g)u + w(t), \quad (1)$$

where $x \in \mathcal{X} \subset \mathbb{R}^n$ is the state, $u \in \mathcal{U} \subset \mathbb{R}^m$ is the control input, $\theta_f \in \Theta_f \subset \mathbb{R}^{p_f}$ is the unknown drift parameter vector contained in a known compact set, and $\theta_g = \begin{bmatrix}\theta_{g,1} & \cdots & \theta_{g,m}\end{bmatrix} \in \Theta_g \subset \mathbb{R}^{p_g\times m}$ is the unknown input parameter matrix contained in a known compact set, with $\theta_{g,j} \in \mathbb{R}^{p_g}$ denoting its j-th column. The uncertain parameters are collected into the vector as $\theta = \begin{bmatrix}\theta_f^\top & \theta_{g,1}^\top & \cdots & \theta_{g,m}^\top\end{bmatrix}^\top \in \Theta \subset \mathbb{R}^{p_f+mp_g}$. The functions $f_0 : \mathbb{R}^n \to \mathbb{R}^n$ and $g_0 : \mathbb{R}^n \to \mathbb{R}^{n\times m}$ are the known nominal drift and input maps, respectively. The functions $F : \mathbb{R}^n \to \mathbb{R}^{n\times p_f}$ and $G : \mathbb{R}^n \to \mathbb{R}^{n\times p_g}$ are the known drift and input regressors, respectively. Furthermore, we denote $f(x,\theta_f) = f_0(x) + F(x)\theta_f$ and $g(x,u,\theta_g) = (g_0(x) + G(x)\theta_g)u$.

Assumption 1. The additive disturbance $w : [t_0,\infty) \to \mathbb{R}^n$ is bounded $\sup_{t\geq t_0}\|w(t)\| = \overline{w}$.

Assumption 2. The full system state $x(t)\ \forall t \in [t_0,\infty)$ is assumed to be perfectly observed.

C. Linear in Parameter form & Parameter Identification

Under Assumption 1 and Assumption 2, the system dynamics (1) can be rearranged to obtain a regression model that is linear in the unknown parameter vector θ. Specifically, define the signal

$$z(t) := \dot{x}(t) - f_0(x(t)) - g_0(x(t))u(t). \quad (2)$$

Substituting (1) yields

$$z(t) = F(x(t))\theta_f + (G(x(t))\theta_g)u(t) + w(t). \quad (3)$$

Using the representation $\theta_g = \begin{bmatrix}\theta_{g,1} & \cdots & \theta_{g,m}\end{bmatrix}$, the input-dependent term can be written as

$$(G(x)\theta_g)u = \sum_{j=1}^{m} u_j G(x)\theta_{g,j}. \quad (4)$$

Define the regressor matrix

$$\Phi(x,u) := [F(x)\ \ u_1G(x)\ \ \cdots\ \ u_mG(x)] \in \mathbb{R}^{n\times(p_f+mp_g)}. \quad (5)$$

Then the dynamics admit the linear-in-parameters (LIP) representation

$$z(t) = \Phi(x(t),u(t))\theta + w(t). \quad (6)$$

1) Persistent Excitation: The LIP model (6) forms the basis for parameter identification. However, (6) depends on the state derivative $\dot{x}(t)$, which may not be directly measurable in practice. To remove this dependence, we integrate the dynamics over a finite time window as shown in [43]. Let $\Delta > 0$ denote the length of the integration window. Integrating (1) over the interval $[t-\Delta, t]$ gives

$$\begin{aligned} x(t) - x(t-\Delta) = &\int_{t-\Delta}^{t} \Big(f_0(x(s)) + F(x(s))\theta_f\Big)\, ds \\ &+ \int_{t-\Delta}^{t} (g_0(x(s)) + G(x(s))\theta_g)u(s)\, ds \\ &+ \int_{t-\Delta}^{t} w(s)\, ds. \end{aligned} \quad (7)$$

Using the definition of $\Phi(x,u)$, this can be rewritten as

$$\begin{aligned} x(t) - x(t-\Delta) = &\int_{t-\Delta}^{t} (f_0(x(s)) + g_0(x(s))u(s))\, ds \\ &+ \int_{t-\Delta}^{t} \Phi(x(s),u(s))\, ds\, \theta \\ &+ \int_{t-\Delta}^{t} w(s)\, ds. \end{aligned} \quad (8)$$

Define

$$\begin{aligned} Y(t) := &\ x(t) - x(t-\Delta) \\ &- \int_{t-\Delta}^{t} (f_0(x(s)) + g_0(x(s))u(s))\, ds, \end{aligned} \quad (9)$$

$$\mathcal{F}(t) := \int_{t-\Delta}^{t} \Phi(x(s),u(s))\, ds, \quad (10)$$

and

$$W_\Delta(t) := \int_{t-\Delta}^{t} w(s)\, ds. \quad (11)$$

Then the dynamics admit the derivative-free regression model

$$Y(t) = \mathcal{F}(t)\theta + W_\Delta(t). \quad (12)$$

The regression model (12) depends only on the measured state and input trajectories over the integration window and avoids direct use of $\dot{x}(t)$. To ensure identifiability, the regressor in (12) must be persistently exciting (PE) [43], [44], which is formally defined as follows.

Definition 1 (Persistent Excitation [43, *Def.* 4.3])**.** Consider the integral regression model in (12), where $\mathcal{F}(t)$ is the

regressor. The regressor is said to be persistently exciting if there exist constants $T > 0$ and $c > 0$ such that for all $t \geq t_0$,

$$\int_t^{t+T} \mathcal{F}(\tau)^\top \mathcal{F}(\tau)\, d\tau \geq cI. \tag{13}$$

2) Finite Excitation: The PE condition requires the trajectory to be sufficiently informative over every interval of length T. However, verifying PE online is difficult since it depends on the future trajectory and may require injecting probing signals into the control input. Instead, we store informative regression data collected along the trajectory. Let

$$\mathcal{H}_k = \{(Y_j, \mathcal{F}_j)\}_{j=1}^{M_k} \tag{14}$$

denote a history stack containing M_k previously stored regression tuples, where each tuple $(Y_j, \mathcal{F}_j)$ is generated from (12). The stored data satisfy

$$Y_j = \mathcal{F}_j \theta + W_{\Delta,j}, \qquad j = 1, \ldots, M_k. \tag{15}$$

Using these stored tuples, identifiability can be characterized through a finite excitation condition.

Definition 2 (Finite Excitation [43, *Def.* 4.5])**.** The history stack $\mathcal{H}_k$ is said to satisfy the finite excitation (FE) condition if there exists a constant $\lambda_{\mathrm{FE}} > 0$ such that

$$\lambda_{\min}\left(\sum_{j=1}^{M_k} \mathcal{F}_j^\top \mathcal{F}_j\right) \geq \lambda_{\mathrm{FE}}. \tag{16}$$

3) Set Membership Identification (SMID): Under Assumption 1, the regression relation (12) can be used to refine the parameter uncertainty set using set-membership identification (SMID) [45]. The key idea of SMID is to maintain a set of parameters that remain consistent with all observed regression data. Let Θ_k denote the feasible parameter set at time step k, defined as the set of all parameters consistent with the regression data collected up to time k. Let $\mathcal{H}_k = \{(Y_j, \mathcal{F}_j)\}_{j=1}^{M_k}$ denote the history stack introduced earlier, where each tuple satisfies

$$Y_j = \mathcal{F}_j \theta + W_{\Delta,j}, \qquad j = 1, \ldots, M_k. \tag{17}$$

Given bounded disturbance, the set of parameters consistent with the data can be characterized by linear inequality constraints. Let $\Theta_0 = \Theta$ denote the initial uncertainty set. At time step k, the feasible parameter set is updated for $\forall j \in \{1, \ldots, M_k\}$ as

$$\Theta_k = \{\theta \in \Theta_{k-1} \mid -\epsilon \mathbf{1} \leq Y_j - \mathcal{F}_j \theta \leq \epsilon \mathbf{1}\}, \tag{18}$$

where $\epsilon > 0$ accounts for bounded disturbance and numerical integration error. The feasible parameter set Θ_k can be interpreted as the intersection of halfspaces defined by the stored data. In practice, the bounds of Θ_k along each coordinate direction can be computed by solving linear programs. Let θ_i denote the i-th component of θ. The lower and upper bounds on θ_i are obtained as

$$\theta_i^{k,-} = \arg\min_\theta \ \theta_i, \quad \theta_i^{k,+} = \arg\max_\theta \ \theta_i, \tag{19}$$

subject to

$$-\epsilon \mathbf{1} \leq Y_j - \mathcal{F}_j \theta \leq \epsilon \mathbf{1}, \quad \theta \in \Theta_{k-1}. \tag{20}$$

The updated uncertainty set can then be written as the hyperrectangle

$$\Theta_k = [\theta_1^{k,-}, \theta_1^{k,+}] \times \cdots \times [\theta_p^{k,-}, \theta_p^{k,+}]. \tag{21}$$

The following result guarantees that the true parameter remains inside the identified set.

Lemma 1 ([45, Lemma 7.3])**.** *Suppose Assumption 1 holds and the parameter sets $\{\Theta_k\}$ are generated using the SMID update described above. Then the sets satisfy*

$$\Theta_k \subseteq \Theta_{k-1} \subseteq \Theta_0 \tag{22}$$

and the true parameter satisfies

$$\theta \in \Theta_k, \qquad \forall k \geq 0. \tag{23}$$

Lemma 1 shows that the SMID update generates a nested sequence of parameter sets that always contains the true parameter. As additional informative data are incorporated, the feasible parameter set shrinks monotonically. Consequently, the parametric uncertainty can be quantified by the *width* of the current feasible set Θ_k, defined next.

Definition 3 (Width of Parameter Set)**.** Let $\Theta_k \subset \mathbb{R}^p$ denote the feasible parameter set at time k, and let $\mathcal{D}$ be a finite set of unit directions. For any direction $d \in \mathcal{D}$, the width of Θ_k along d is defined as

$$w_d(\Theta_k) = \sup_{\theta \in \Theta_k} d^\top \theta - \inf_{\theta \in \Theta_k} d^\top \theta. \tag{24}$$

This quantity measures the extent of the feasible parameter set along direction d. In the scalar case ($p = 1$) with $d = 1$ and $\Theta_k = [\theta_{\min}, \theta_{\max}]$, the width reduces to

$$w_d(\Theta_k) = \theta_{\max} - \theta_{\min}. \tag{25}$$

D. Problem Statement

We consider a dual control setting in which a robot must complete a prescribed mission under bounded parametric uncertainty and additive disturbances. Safety is enforced through robust feedback policies that guarantee state and input constraint satisfaction for all admissible uncertainty realizations.

A key challenge is that the mission cost under robust safety requirements depends on the size of the uncertainty set: larger uncertainty leads to more conservative behavior and higher cost. Reducing uncertainty during execution, therefore, enables less conservative robust behavior.

In constrast to earlier work and formulations that continuously encourage exploration through weighted mission-exploration objectives (e.g., mission cost plus a weighted exploration term), uncertainty reduction in this work is pursued only when it remains feasible with respect to a prescribed exploration budget. Accordingly, the objective is to complete the mission safely under worst-case uncertainty while allowing uncertainty reduction during execution, subject to

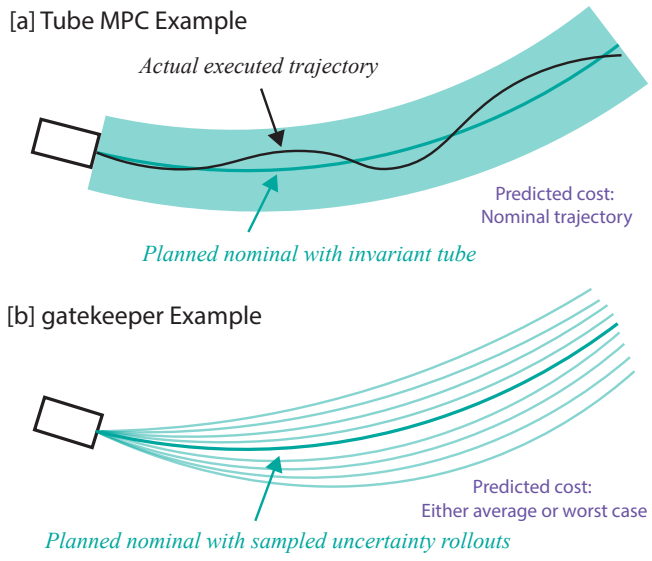

Fig. 1: Predicted Cost Example

safety constraints and an exploration budget that bounds the cumulative additional cost incurred by exploratory actions relative to the nominal mission behavior.

Now we formulate the problem mathematically. We first define a trajectory:

Definition 4 (Trajectory)**.** Let $\mathcal{T} = [t_0, t_f] \subset \mathbb{R}$. Let Π denote the set of admissible feedback policies $\pi : \mathbb{R} \times \mathcal{X} \to \mathcal{U}$. A trajectory induced by a policy $\pi \in \Pi$ is a pair $p = (p_x : \mathcal{T} \to \mathcal{X},\ p_u : \mathcal{T} \to \mathcal{U})$ such that

$$\dot{p}_x(t) = f(p_x(t), \hat{\theta}_f) + g(p_x(t), \pi(t, p_x(t)), \hat{\theta}_g), \quad \forall\, t \in \mathcal{T}. \tag{26}$$

Now consider the system (1) with parametric uncertainty set Θ and bounded disturbances. We define a robust feedback policy as follows.

Definition 5 (Robust feedback policy)**.** A feedback law $\pi^{\text{rob}} : \mathbb{R} \times \mathcal{X} \to \mathcal{U}$ is called a *robust feedback policy* if, when applied to the system (1), the resulting closed-loop trajectory $x(t)$ satisfies

$$\dot{x}(t) = f(x(t), \theta_f) + g(x(t), \theta_g)\, \pi^{\text{rob}}(t, x(t)) + w(t), \tag{27}$$

and for all admissible uncertainties and disturbances

$$x(t) \in \mathcal{S}, \quad \pi^{\text{rob}}(t, x(t)) \in \mathcal{U}, \qquad \forall t \geq t_0, \tag{28}$$

for every $\theta_f \in \Theta_f$, $\theta_g \in \Theta_g$, and $w(t) \in \mathcal{W}$.

Definition 5 provides an abstract characterization of a robust feedback policy that guarantees constraint satisfaction for all admissible uncertainties and disturbances. In practice, π_k^{rob} is implemented using specific robust control mechanisms, such as tube MPC [2], robust CBF-based safety filters [24], or robust version of `gatekeeper` [9]. Let $\pi_k^{\text{rob}} : \mathbb{R} \times \mathcal{X} \to \mathcal{U}$ denote the robust feedback policy available at planning time t_k. Given this policy, we denote by $p_k^{\text{rob}} = (p_{x,k}^{\text{rob}}, p_{u,k}^{\text{rob}})$ the corresponding robust trajectory induced by π_k^{rob} over the horizon $\mathcal{T}_k = [t_k, t_{k+1}]$, in the sense of Definition 4.

Definition 6 (Predicted Cost)**.** Let $\pi_k : \mathbb{R} \times \mathcal{X} \to \mathcal{U}$ denote an available feedback policy at planning time t_k, let $x_k \in \mathcal{X}$ denote the state at time t_k, and let $\mathcal{T}_k = [t_k, t_{k+1}]$ be a finite horizon of length $t_{k+1} - t_k > 0$. The predicted cost associated with π_k is the functional $\mathcal{J}^{k\to k+1} : \Pi \times \mathcal{X} \times 2^{\mathbb{R}} \to \mathbb{R}_{\geq 0}$

$$(\pi_k, x_k, \mathcal{T}_k) \mapsto \mathcal{J}^{k\to k+1}(\pi_k, x_k, \mathcal{T}_k), \tag{29}$$

which quantifies the *predicted* performance of executing the policy π_k from state x_k over the horizon $\mathcal{T}_k$.

Predicted Cost Examples: **(I)** In tube MPC [2], the robust policy π_k^{rob} is constructed from a finite-horizon nominal trajectory, an invariant tube, and an ancillary feedback controller that keeps the closed-loop state within the tube. The predicted cost $\mathcal{J}_{\text{rob}}^{k\to k+1}$ is evaluated along the nominal tube centerline over the horizon $T_k = [t_k, t_{k+1}]$. **(II)** In a robust `gatekeeper` framework [9], a candidate trajectory is first generated from a nominal planning policy and then subjected to a safety verification step under the uncertainty set Θ_k and disturbance set $\mathcal{W}$. If the candidate trajectory satisfies the safety constraints for all admissible uncertainties, it is *committed* and induces the robust policy π_k^{rob}. The predicted cost $\mathcal{J}_{\text{rob}}^{k\to k+1}$ is then estimated using N forward rollouts of the committed trajectory over the horizon $T_k = [t_k, t_{k+1}]$ under admissible uncertainty realizations, and taken as either the worst-case or expected mission cost across these N samples. Predicted cost of these methods is illustrated in Figure 1.

Problem 1. *Consider the system* (1) *with initial state* x_0 *and initial parametric uncertainty set* Θ_0*. Let* $t_0 < t_1 < \cdots < t_k < \cdots$ *denote replanning times, which are not necessarily uniformly spaced, and define* $\mathcal{T}_k = [t_k, t_{k+1}]$.

At each replanning time t_k*, a* robust reference policy $\pi_k^{\text{rob,ref}} : \mathcal{X} \to \mathcal{U}$ *is available, representing the default robust task-execution behavior under the current uncertainty set* Θ_k*. The goal is to select a sequence of robust policies* $\{\pi_k^{\text{rob,sol}}\}_{k=0}^{\infty}$ *so as to maximize cumulative uncertainty reduction over the replanning epochs, subject to safety constraints and a budget on the cumulative* exploration cost*, defined as the excess predicted cost incurred relative to the robust reference behavior:*

$$\max_{\{\pi_k^{\text{rob,sol}}\}_{k=0}^{\infty}} \quad \sum_{k=0}^{\infty} \Delta w_d(\Theta_k) \tag{30a}$$

$$\text{s.t.} \quad x(t) \in \mathcal{S}, \quad u(t) \in \mathcal{U}, \quad \forall t \geq t_0, \tag{30b}$$

$$\sum_{k=0}^{\infty} \Delta \mathcal{J}_{\text{exp}}^{k} \ \leq\ B_{\text{exp}}. \tag{30c}$$

where the uncertainty width reduction over the $k-$*th replanning interval is defined as*

$$\Delta w_d(\Theta_k) = w_d(\Theta_k) - w_d(\Theta_{k+1}), \tag{31}$$

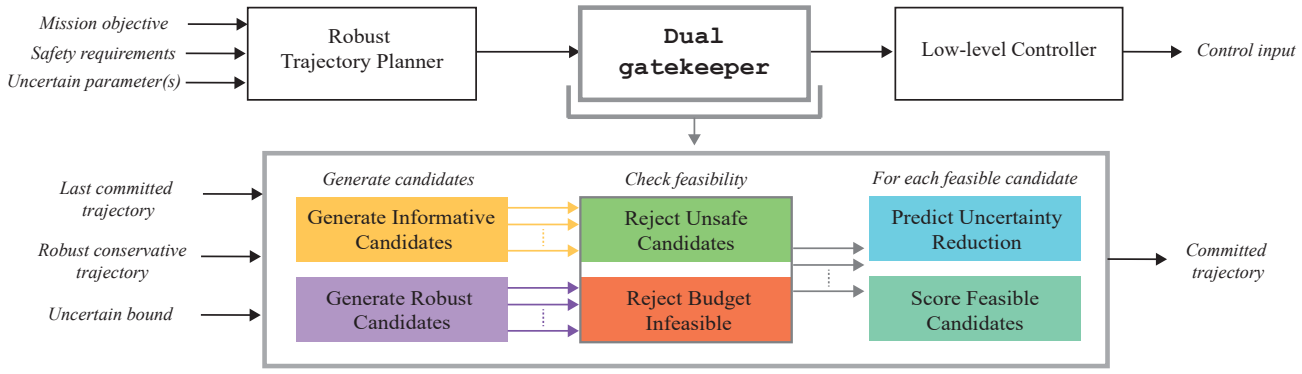

Fig. 2: The mission objective, safety requirements, and uncertainty bounds are used by a robust trajectory planner to compute a conservative backup trajectory. The dual gatekeeper generates informative and conservative candidates, rejects those that violate safety or budget constraints, predicts uncertainty reduction for the remaining candidates, and commits the highest-scoring feasible trajectory for execution by the low-level controller.

and the exploration cost incurred over $\mathcal{T}_k$ is defined as

$$\Delta\mathcal{J}^k_{\text{exp}} := \max\Big\{0,\ \mathcal{J}^{k\to k+1}_{\text{rob,sol}} - \mathcal{J}^{k\to k+1}_{\text{rob,ref}}\Big\}. \tag{32}$$

Remark 1. The exploration budget in (30c) is formulated in terms of the *predicted* exploration cost $\Delta\mathcal{J}^k_{\text{exp}}$, rather than the cost incurred during execution. Constraining the executed exploration cost directly is generally intractable, as it depends on the particular, unknown realizations of the parametric uncertainty and disturbances encountered during each replanning interval.

Instead, the predicted cost is evaluated prior to execution over the finite horizon $\mathcal{T}_k$ and may be defined as the worst-case cost over the admissible uncertainty and disturbance sets. Under this definition, bounding the cumulative predicted exploration cost guarantees that the executed exploration cost also remains within the prescribed budget B_{exp}, regardless of the realized uncertainty.

IV. Proposed Solution Framework

The overall framework is shown in Fig. 2 and illustrated through the pipeline example in Fig. 3. It is proposed as a solution to Problem 1. The objective is to actively reduce parametric uncertainty through exploration while guaranteeing safety and respecting a mission-level cost budget. We first formalize the general framework and then show how it can be instantiated using two different safety mechanisms: tube MPC (Section VI) and the robust gatekeeper architecture (Section VII).

A. Framework Overview

The framework operates in a receding-horizon manner with decisions made at discrete, not necessarily uniformly spaced, replanning times. At each replanning time t_k, a robust backup policy is first constructed to guarantee satisfaction of the state and input constraints under the current uncertainty set over the remaining mission horizon (Fig. 3[a]). This backup policy represents a conservative baseline behavior that optimizes the mission objective under worst-case uncertainty. Next, a collection of candidate policy segments is generated over finite horizons, including conservative candidates that preserve the backup behavior and informative candidates designed explicitly to promote uncertainty reduction (Fig. 3[b]). Depending on their construction, informative candidates may incur additional mission cost and may not satisfy safety constraints a priori.

Candidate policy segment pairs that cannot be certified as safe or that violate the mission-level budget are discarded (Fig. 3[c]). Among the remaining feasible candidates, the policy segment that achieves the largest predicted uncertainty reduction, discounted by horizon length, while remaining within the budget is selected and committed for execution (Fig. 3[d]). After execution, the uncertainty set is updated using newly collected data, and a new robust backup policy is constructed for the next replanning cycle (Fig. 3[e]).

By iteratively refining the uncertainty set and recomputing the robust backup policy, the framework enables progressively less conservative behavior while maintaining safety and budget feasibility throughout the mission.

B. Robust Backup Policy Construction

At each replanning time t_k, the framework begins by constructing a robust backup policy that guarantees safe mission execution under the current uncertainty set.

Definition 7 (Robust backup policy)**.** At replanning time t_k, a *robust backup policy* is a feedback law $\pi^{\text{rob},B}_k : \mathbb{R}\times\mathcal{X} \to \mathcal{U}$ together with a fixed backup horizon $T_B > 0$ such that, when applied to the system (1), the resulting closed-loop trajectory satisfies $x(t) \in \mathcal{S}(t),\ \forall w(t) \in \mathcal{W},\ \forall t \in [t_k,\, t_k + T_B]$, and for all admissible uncertainty realizations $\theta \in \Theta_k$.

The robust backup policy $\pi^{\text{rob},B}_k$ defines a conservative baseline behavior that guarantees constraint satisfaction under worst-case uncertainty over the fixed horizon $[t_k,\, t_k + T_B]$. The framework then proceeds to generate candidate policy segments that may temporarily deviate from the backup behavior in order to reduce parametric uncertainty.

Remark 2. The backup horizon T_B may be selected either as a fixed constant throughout the mission or adapted at each

replanning time t_k. Both choices are compatible with the proposed framework. Concrete examples of fixed and adaptive selections of T_B are provided in the two instantiations presented in Sections VI and VII.

C. Candidate Policy Segment Generation

At each replanning time t_k, candidate policies are generated in *pairs* over a common finite horizon: a conservative policy segment that preserves the robust backup behavior and an informative policy segment that may deviate in order to reduce uncertainty in the unknown parameter θ.

Candidate horizons are generated by uniformly increasing the horizon length in increments of T_c, with the horizon length capped by the backup horizon T_B. Accordingly, the set of candidate horizon lengths is defined as

$$\mathcal{T}_k^c = \{\, T_i^{c,k} \;:\; T_i^{c,k} = \min\{iT_c,\ T_B\}\,\}_{i=1}^{N_k}, \tag{33}$$

where $N_k := \lceil T_B/T_c \rceil$. Each candidate horizon length $T_i^{c,k}$ induces the time interval $[t_k,\ t_k + T_i^{c,k}]$ over which both the conservative and informative policy segments are generated and evaluated.

Definition 8 (Conservative candidate policy segment)**.** The $i-$th *conservative candidate policy segment* generated at time t_k over the horizon $T_i^{c,k}$ is defined as the restriction of the robust backup policy $\pi_{k,i}^{\mathrm{rob},B}$ to the interval $[\,t_k,\ t_k + iT_c\,]$, and therefore preserves the backup behavior over this horizon.

Definition 9 (Informative candidate policy segment)**.** The $i-$th *informative candidate policy segment* generated at time t_k over the horizon $T_i^{c,k}$ is a feedback policy $\pi_{k,i}^{I}$ defined on $[\,t_k,\ t_k + iT_c\,]$ and designed to reduce uncertainty in the unknown parameter θ. The informative policy segment is required to satisfy a terminal recoverability condition: the closed-loop state at time t_k+iT_c must lie in a set from which the robust backup policy $\pi_k^{\mathrm{rob},B}$ can be safely applied for the remainder of the mission.

Definition 10 (Candidate policy segment pair)**.** The $i-$th *candidate policy segment pair* generated at time t_k is the pair

$$(\pi_{k,i}^{I},\ \pi_{k,i}^{\mathrm{rob},B})$$

defined over the common horizon $T_i^{c,k} = [\,t_k,\ t_k + iT_c\,]$, where $\pi_{k,i}^{I}$ is an informative policy segment and $\pi_{k,i}^{\mathrm{rob},B}$ denotes the restriction of the robust backup policy $\pi_k^{\mathrm{rob},B}$ to the same horizon.

Depending on how the informative policy segment is constructed, safety may or may not be ensured over the candidate horizon. We therefore define a notion of *validity* to identify those candidate policy segment pairs for which safety can be certified.

Definition 11 (Valid candidate policy segment pair)**.** A candidate policy segment pair $(\pi_{k,i}^{I},\ \pi_{k,i}^{\mathrm{rob},B})$ associated with horizon $T_i^{c,k}$ is said to be *valid* if there exists a corresponding *robustified informative policy segment* $\pi_{k,i}^{\mathrm{rob},I}$, defined over the same horizon $T_i^{c,k}$, such that when applied to the system (1), the resulting closed-loop behavior satisfies the state and input constraints for all admissible uncertainty realizations and disturbances over $T_i^{c,k}$.

The set of all valid candidate policy segment pairs generated at time t_k is defined as

$$\mathcal{V}_k^c = \Big\{(\pi_{k,i}^{\mathrm{rob},I},\ \pi_{k,i}^{\mathrm{rob},B}) \mid \text{pair valid by } \textit{Def. 11}\Big\}. \tag{34}$$

We also define the set of all conservative candidates as

$$\mathcal{V}_k^{\mathrm{cons}} = \{\, \pi_{k,i}^{\mathrm{rob},B}\}_{i=1}^{N_k}. \tag{35}$$

D. Committing a Candidate Policy Segment

Having defined the set of valid candidate policy segment pairs $\mathcal{V}_k^c$ and the set of conservative policy segments $\mathcal{V}_k^{\mathrm{cons}}$, the final step at replanning time t_k is to decide which policy segment to commit for execution. The guiding principle is to drive uncertainty reduction as rapidly as possible while never compromising safety or violating the mission-level budget.

Each valid candidate policy segment pair $(\pi_{k,i}^{\mathrm{rob},I},\ \pi_{k,i}^{\mathrm{rob},B}) \in \mathcal{V}_k^c$ is assigned a score that reflects the predicted uncertainty reduction achieved by its informative policy segment over the common horizon. To prioritize early information gain, this score is discounted by the horizon length:

$$s_i^{c,k} \;=\; \exp\big(-\lambda T_i^{c,k}\big)\,\Delta\xi_i, \qquad \lambda > 0, \tag{36}$$

where $\Delta\xi_i$ denotes the predicted reduction in the *average* directional width of the uncertainty set.

Specifically, if Θ^k denotes the uncertainty set at time t_k and $\Theta^{k+1,i}$ denotes the predicted uncertainty set after executing candidate i, then

$$\Delta\xi_i \;=\; \frac{1}{|\mathcal{D}|}\sum_{d\in\mathcal{D}}\Big(w_d(\Theta^k) - w_d(\Theta^{k+1,i})\Big), \tag{37}$$

where $w_d(\Theta)$ is defined in *Def.* 3. The prediction of $\Theta^{k+1,i}$ and the computation of $w_d(\Theta)$ are detailed in Section V.

To enforce budget feasibility, let $\mathcal{J}_{\mathrm{exec}}^k$ denote the accumulated predicted exploration cost incurred up to replanning time t_k, defined as

$$\mathcal{J}_{\mathrm{exec}}^k \;:=\; \sum_{j=0}^{k-1}\Delta\mathcal{J}_{\mathrm{exp}}^j. \tag{38}$$

For a candidate policy segment indexed by i, define the predicted exploration cost incurred over its candidate horizon $T_i^{c,k} = [t_k,\, t_k + T_i^{c,k}]$ as the excess predicted cost relative to the conservative segment:

$$\Delta\mathcal{J}_{\mathrm{exp}}^k(i) \;:=\; \max\Big\{0,\ \mathcal{J}_{\mathrm{rob},I}^{k\to k+i} - \mathcal{J}_{\mathrm{rob},B}^{k\to k+i}\Big\}. \tag{39}$$

A candidate pair is budget-feasible if committing its informative policy segment does not cause the cumulative exploration cost to exceed the budget B_{exp}. Accordingly, we

define the feasible index set

$$\mathcal{F}_k^c = \Big\{ i \ : \ (\pi_{k,i}^{\mathrm{rob},I}, \pi_{k,i}^{\mathrm{rob},B}) \in \mathcal{V}_k^c, \tag{40}$$

$$\mathcal{J}_{\mathrm{exec}}^k + \Delta\mathcal{J}_{\mathrm{exp}}^k(i) \le B_{\mathrm{exp}} \Big\}. \tag{41}$$

The index of the committed candidate is defined as

$$i^\star = \begin{cases} \arg\max\limits_{i \in \mathcal{F}_k^c} s_i^{c,k}, & \text{if } \mathcal{F}_k^c \neq \emptyset, \\ 1, & \text{if } \mathcal{F}_k^c = \emptyset. \end{cases} \tag{42}$$

If $\mathcal{F}_k^c \neq \emptyset$, the robust informative policy segment $\pi_{k,i^\star}^{\mathrm{rob},I}$ is committed over the horizon $T_{i^\star}^{c,k}$. If $\mathcal{F}_k^c = \emptyset$, no informative candidate is both valid and budget-feasible, and the framework commits the conservative policy segment with the shortest horizon, $\pi_{k,1}^{\mathrm{rob},B}$.

Definition 12 (Committed policy segment)**.** At replanning time t_k, the committed policy segment π_k^{com} is defined as

$$\pi_k^{\mathrm{com}} \ = \ \begin{cases} \pi_{k,i^\star}^{\mathrm{rob},I}, & \mathcal{F}_k^c \neq \emptyset, \\ \pi_{k,1}^{\mathrm{rob},B}, & \mathcal{F}_k^c = \emptyset, \end{cases} \tag{43}$$

where $\pi_{k,1}^{\mathrm{rob},B}$ denotes the restriction of the robust backup policy to the shortest conservative horizon.

After committing the selected policy segment, the system executes it and the next replanning time is set to

$$t_{k+1} = t_k + T_{i^\star}^{c,k}. \tag{44}$$

At time t_{k+1}, the uncertainty set is updated using set membership identification (SMID) [38], [46], the robust backup policy is recomputed, and the procedure repeats.

E. Safety and Budget Guarantees

This subsection establishes that the proposed framework guarantees *(i)* safety with respect to state and input constraints and *(ii)* feasibility with respect to the prescribed exploration budget. The guarantees are stated for the closed-loop execution induced by the sequence of committed policy segments generated by the framework.

Theorem 1. *Let*

$$\pi^{\mathrm{sol}} = \{\pi_0^{\mathrm{com}}, \ \pi_1^{\mathrm{com}}, \ \ldots\} \tag{45}$$

denote the sequence of policy segments committed by the framework, where each π_k^{com} is executed over the interval $[t_k, t_{k+1}]$ with $t_{k+1} = t_k + T_{i^\star}^{c,k}$.

If at each replanning time t_k the committed policy segment is selected according to Def. 12, then the resulting closed-loop trajectory satisfies

$$x(t) \in \mathcal{S}(t), \quad u(t) \in \mathcal{U}, \qquad \forall t \ge t_0, \tag{46}$$

and the cumulative exploration cost satisfies

$$\sum_{k=0}^{\infty} \Delta\mathcal{J}_{\mathrm{exp}}^k \ \le \ B_{\mathrm{exp}}. \tag{47}$$

Proof. We prove safety and budget feasibility by induction over the replanning times. At t_0, safety holds by construction of the initial robust backup policy. The accumulated exploration cost is zero, hence budget feasibility holds. Assume that at replanning time t_k the state x_k is admissible and the accumulated exploration cost satisfies $\mathcal{J}_{\mathrm{exec}}^k \le B_{\mathrm{exp}}$. By *Def.* 12, the committed policy segment π_k^{com} is either: *(i)* a conservative backup segment, or *(ii)* a robustified informative segment from a budget-feasible candidate pair.

In case (i), safety follows directly from the definition of the robust backup policy, and no exploration cost is incurred.

In case (ii), validity of the candidate pair guarantees that $\pi_{k,i^\star}^{\mathrm{rob},I}$ satisfies all state and input constraints over $[t_k, t_{k+1}]$. Budget feasibility follows from the feasibility check at t_k, which enforces

$$\mathcal{J}_{\mathrm{exec}}^k + \Delta\mathcal{J}_{\mathrm{exp}}^k(i^\star) \ \le \ B_{\mathrm{exp}}. \tag{48}$$

After executing the committed policy segment, the uncertainty set is updated and the accumulated exploration cost is incremented accordingly. Thus, $\mathcal{J}_{\mathrm{exec}}^{k+1} \le B_{\mathrm{exp}}$, and the induction hypothesis holds at t_{k+1}. By induction, safety and budget feasibility are preserved at every replanning time. □

Algorithm 1: Uncertainty Shrinkage Prediction via Parallel Rollouts

Input: Candidate policy segment $\pi_{k,i}$ over horizon $T_i^{c,k} = [t_k, t_k + iT_c]$; state x_k; uncertainty set Θ^k; disturbance bound $\mathcal{W}$; SMID update rule; direction set $\mathcal{D}$; number of rollouts N.

Output: Predicted uncertainty reduction $\Delta\xi_i$

Initialize rollout buffer $\{\Delta\xi_i^{(\ell)}\}_{\ell=1}^N \leftarrow 0$;
for $\ell = 1, \ldots, N$ ***in parallel*** **do**
  Sample $\theta^{(\ell)} \sim \mathcal{U}(\Theta^k)$;
  Sample disturbances $w_j^{(\ell)} \sim \mathcal{U}(\mathcal{W})$ for $j = 1, \ldots, N_i$;
  Forward simulate (1) from x_k under $\pi_{k,i}$ using $(\theta^{(\ell)}, \{w_j^{(\ell)}\})$;
  Form SMID data $\{(\Phi_j^{(\ell)}, z_j^{(\ell)})\}_{j=1}^{N_i}$;
  $\Theta^{k+1,i,(\ell)} \leftarrow \mathrm{SMID}\Big(\Theta^k, \{(\Phi_j^{(\ell)}, z_j^{(\ell)})\}_{j=1}^{N_i}\Big)$;
  $\Delta\xi_i^{(\ell)} \leftarrow \frac{1}{|\mathcal{D}|} \sum_{d \in \mathcal{D}} \left(w_d(\Theta^k) - w_d(\Theta^{k+1,i,(\ell)})\right)$;
end
$\Delta\xi_i \leftarrow \frac{1}{N} \sum_{\ell=1}^N \Delta\xi_i^{(\ell)}$;

V. Prediction of Uncertainty Shrinkage

To evaluate informative candidate policy segments, the framework must predict the reduction in parametric uncertainty induced by executing a candidate trajectory. We consider two approaches for this prediction. The first approach is a simulation-based method that estimates uncertainty shrinkage through forward rollouts under sampled parameter and disturbance realizations. The second approach predicts uncertainty shrinkage directly from the planned trajectory by analyzing the set of parameters that remain consistent with the resulting regression measurements under bounded noise. We describe these two approaches below.

A. Simulation-Based Uncertainty Shrinkage Prediction

To evaluate informative candidate policy segments, the framework predicts the expected reduction in parametric uncertainty induced by executing a candidate policy segment

$\pi^{\mathrm{rob},I}_{k,i}$ over the horizon $T^{c,k}_i = [t_k,\ t_k + iT_c]$. This prediction is obtained via forward simulation of the closed-loop system for a finite number of admissible parameter and disturbance realizations.

Given the current state x_k and uncertainty set Θ^k, N independent rollouts are generated by sampling $\theta^{(\ell)} \overset{\text{i.i.d.}}{\sim} \mathcal{U}(\Theta^k)$ and a sequence of additive disturbances $\{w^{(\ell)}_j\}_{j=1}^{N_i} \overset{\text{i.i.d.}}{\sim} \mathcal{U}(\mathcal{W})$, where N_i denotes the number of discrete simulation steps over the horizon $T^{c,k}_i$. For each $\ell \in \{1,\dots,N\}$, the system dynamics (1) are forward simulated from x_k under the informative policy segment $\pi^I_{k,i}$ (Algorithm 1, Lines 2–5).

From each rollout, a sequence of regressor–measurement pairs $\{(\Phi^{(\ell)}_j, z^{(\ell)}_j)\}_{j=1}^{N_i}$ is constructed and used to compute a predicted post-execution uncertainty set $\Theta^{k+1,i,(\ell)}$ via set-membership identification (Algorithm 1, Lines 6–7).

The effect of candidate i on the uncertainty set is quantified by the reduction in the *average directional width*. For each rollout ℓ, the reduction is computed as

$$\Delta\xi^{(\ell)}_i = \frac{1}{|\mathcal{D}|}\sum_{d\in\mathcal{D}}\Big(w_d(\Theta^k) - w_d(\Theta^{k+1,i,(\ell)})\Big), \tag{49}$$

where $w_d(\cdot)$ denotes the directional width defined in *Def.* 3; see Algorithm 1, Line 8.

Finally, the predicted uncertainty reduction associated with candidate i is defined as the empirical average over the N independent rollouts,

$$\Delta\xi_i = \frac{1}{N}\sum_{\ell=1}^{N}\Delta\xi^{(\ell)}_i, \tag{50}$$

which is used directly in the candidate scoring rule (36).

B. Data-Consistency–Based Uncertainty Shrinkage Prediction

In this section, we quantify the pre-execution impact of a planned trajectory on parameter uncertainty via the *width* in *Def.* 3. We present the method for a general trajectory $p = (p_x, p_u)$ on $[t_i, t_f]$. Let t_j, $j = 1,\dots,N_j$ denote the sampling times, and define $\Phi_j = \Phi(p_x(t_j), p_u(t_j)) \in \mathbb{R}^{c\times p}$, $z_j = z(t_j) \in \mathbb{R}^c$, $w_j = w(t_j) \in \mathbb{R}^c$, $\|w_j\|_\infty \le \overline{w}$. Stacking the regressors gives

$$A = \begin{bmatrix}\Phi_1 & \Phi_2 & \cdots & \Phi_{N_j}\end{bmatrix}^\top \in \mathbb{R}^{M\times p}, \quad M = N_j c. \tag{51}$$

Now let $\theta^\star$ denote the true (unknown) parameter. Two parameters θ and $\theta^\star$ can produce the same stacked data under bounded noises w_1, w_2 if and only if

$$A\theta + w_1 = A\theta^\star + w_2, \quad \|w_1\|_\infty, \|w_2\|_\infty \le \overline{w}. \tag{52}$$

The key question is: *under what conditions can two distinct parameters θ and $\theta^\star$ produce the same measurements within the noise bound?* If many such parameters remain feasible, the uncertainty set stays large; if few remain, it shrinks. Hence, the fewer alternative parameters that fit the data within the noise bound, the greater the uncertainty reduction achieved by the planned trajectory.

Remark 3. The uncertainty shrinkage prediction in this section characterizes uncertainty reduction along a planned trajectory under bounded additive disturbances and bounded parametric uncertainty, but does not account for deviation between the planned trajectory and the executed closed-loop state-input trajectory, which is left for future work.

Lemma 2. *Let $\theta^\star \in \Theta$ be the (unknown) true parameter, and define $e_\theta = \theta - \theta^\star \in \mathbb{R}^p$. There exists some w_j with $\|w_j\|_\infty \le \overline{w}$ such that*

$$\|\Phi_j(\theta^\star - \theta) + w_j\|_\infty \le \overline{w} \iff \|\Phi_j e_\theta\|_\infty \le 2\overline{w}, \tag{53}$$

and combining across all j yields $\|Ae_\theta\|_\infty \le 2\overline{w}$.

Proof. We first show that (53) holds. ($\Rightarrow$) Since $e_\theta = \theta - \theta^\star$, we have $\Phi_j(\theta^\star - \theta) = -\Phi_j e_\theta$. Thus

$$\begin{aligned}\|\Phi_j(\theta^\star - \theta) + w_j\|_\infty &= \|-\Phi_j e_\theta + w_j\|_\infty &\text{(54a)}\\ &= \|\Phi_j e_\theta - w_j\|_\infty \le \overline{w} &\text{(54b)}\end{aligned}$$

Now, by the triangle inequality:

$$\begin{aligned}\|\Phi_j e_\theta\|_\infty &= \|(\Phi_j e_\theta - w_j) + w_j\|_\infty &\text{(55a)}\\ &\le \|\Phi_j e_\theta - w_j\|_\infty + \|w_j\|_\infty &\text{(55b)}\\ &\le \overline{w} + \overline{w} = 2\overline{w}. &\text{(55c)}\end{aligned}$$

($\Leftarrow$) Let $\|\Phi_j e_\theta\|_\infty \le 2\overline{w}$, choose $w_j = \mathrm{clip}(\Phi_j e_\theta, \overline{w})$, where $\mathrm{clip}(\Phi_j e_\theta, \overline{w}) = \mathrm{sign}(\Phi_j e_\theta) \odot \min\{|\Phi_j e_\theta|, \overline{w}\}$. Thus,

$$\begin{aligned}\|w_j\|_\infty &\le \overline{w} &\text{(56a)}\\ \|\Phi_j e_\theta - w_j\|_\infty &= \max\{\|\Phi_j e_\theta\|_\infty - \overline{w}, 0\} \le \overline{w} &\text{(56b)}\end{aligned}$$

Since $\Phi_j(\theta^\star - \theta) = -\Phi_j e_\theta$ and using (54), we have

$$\|\Phi_j(\theta^\star - \theta) + w_j\|_\infty = \|\Phi_j e_\theta - w_j\|_\infty \le \overline{w} \tag{57}$$

Thus, combining (53) for all j yields $\|Ae_\theta\|_\infty \le 2\overline{w}$. □

Lemma 2 states that two parameters are *indistinguishable* when their offset e_θ satisfies $\|Ae_\theta\|_\infty \le 2\overline{w}$, meaning that the observed data could equally well be explained by either parameter, given bounded noise. Offsets exceeding $2\overline{w}$ are ruled out, motivating the definition of the feasible error set.

Definition 13 (Error set)**.** Given the planned trajectory (through A), $\overline{w}$, and $e_\theta = \theta - \theta^\star$, define

$$\mathcal{E}_\theta \ := \ \{\, e_\theta \in \mathbb{R}^p : \ \|Ae_\theta\|_\infty \le 2\overline{w} \,\}. \tag{58}$$

The error set $\mathcal{E}_\theta := \{e_\theta \in \mathbb{R}^p : \|Ae_\theta\|_\infty \le 2\overline{w}\}$ collects all feasible offsets $e_\theta = \theta - \theta^\star$ under the trajectory and noise bound. Equivalently, θ is feasible iff $\theta - \theta^\star \in \mathcal{E}_\theta$, i.e., $\theta \in \theta^\star + \mathcal{E}_\theta$. Since $\theta \in \Theta$, the feasible parameter set is

$$\Theta_{N_j}(\theta^\star) \ = \ \Theta \cap (\theta^\star + \mathcal{E}_\theta). \tag{59}$$

where $\Theta_{N_j}(\theta^\star)$ is the predicted parameter set after N_j planned samples. We measure the width of the predicted set $\Theta_{N_j}(\theta^\star)$ along a direction $d \in \mathbb{R}^p$. Choosing $d = e_{\theta,i}$ yields the width of the i-th parameter. By definition,

$$w_d(\Theta_{N_j}(\theta^\star)) = w_d(\Theta \cap (\theta^\star + \mathcal{E}_\theta)). \tag{60}$$

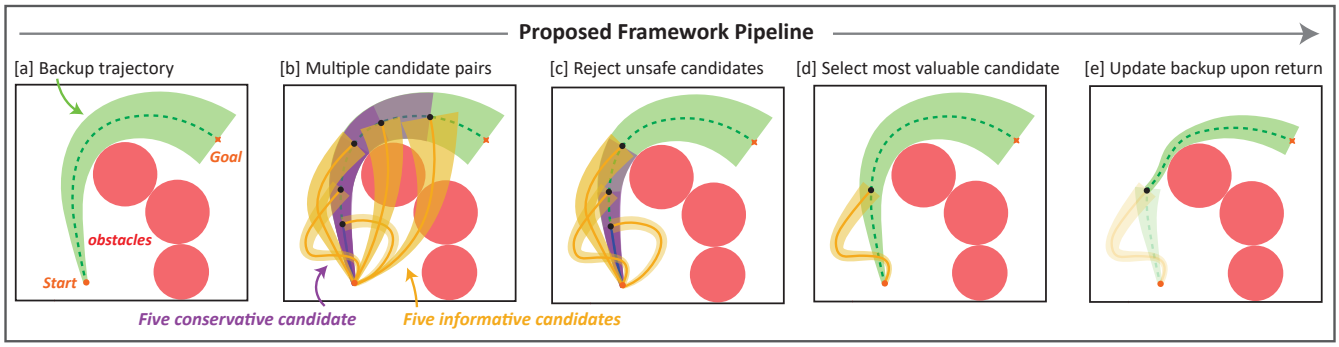

Fig. 3: The proposed framework at a glance. Starting from a conservative backup trajectory, multiple candidate trajectories are generated, including both conservative and informative options. Unsafe candidates are rejected, and the most valuable safe candidate is selected for execution. Upon returning to the backup trajectory, the backup plan is updated. The candidate trajectories represent variations in the trajectory state space; their depiction in the physical environment is purely illustrative.

Because the width of an intersection cannot exceed that of either set, and width is translation-invariant ($w_d(\theta^\star + \mathcal{E}_\theta) = w_d(\mathcal{E}_\theta)$), we obtain

$$w_d(\Theta_{N_j}(\theta^\star)) \;\le\; \min(w_d(\Theta),\, w_d(\mathcal{E}_\theta)). \tag{61}$$

We next show how $w_d(\mathcal{E}_\theta)$ can be computed from the planned trajectory.

Lemma 3. *For any $d \in \mathbb{R}^p \setminus \{0\}$,*

$$w_d(\mathcal{E}_\theta) = 2\, h_{\mathcal{E}_\theta}(d), \qquad h_{\mathcal{E}_\theta}(d) = \sup_{e_\theta \in \mathcal{E}_\theta} d^\top e_\theta \tag{62}$$

Proof. For any $e_\theta \in \mathcal{E}_\theta$,

$$\|Ae_\theta\|_\infty = \|-Ae_\theta\|_\infty = \|A(-e_\theta)\|_\infty \le 2\overline{w}. \tag{63}$$

Since $-e_\theta \in \mathcal{E}_\theta$ and $0 \in \mathcal{E}_\theta$, this implies $w_d(\mathcal{E}_\theta) = \sup_{e_\theta \in \mathcal{E}_\Theta} d^\top e_\theta - \inf_{e_\theta \in \mathcal{E}_\Theta} d^\top e_\theta = 2 \sup_{e_\theta \in \mathcal{E}_\Theta} d^\top e_\theta$. □

Since $\|Ae_\theta\|_\infty \le 2\overline{w} \iff -2\overline{w}\mathbf{1}_M \le Ae_\theta \le 2\overline{w}\mathbf{1}_M$, by (62) it suffices to compute

$$\begin{aligned} &\max_{e_\theta \in \mathbb{R}^p} d^\top e_\theta \\ &\text{s.t. } -2\overline{w}\mathbf{1}_M \le Ae_\theta \le 2\overline{w}\mathbf{1}_M. \end{aligned} \tag{64}$$

Now, we introduce multipliers $\lambda_1, \lambda_2 \in \mathbb{R}^M_{\ge 0}$ and write the Lagrangian of the (64) as $\mathcal{L}(e_\theta, \lambda_1, \lambda_2) = d^\top e_\theta + \lambda_1^\top(2\overline{w}\mathbf{1}_M - Ae_\theta) + \lambda_2^\top(2\overline{w}\mathbf{1}_M + Ae_\theta)$. The dual is finite iff $A^\top(\lambda_2 - \lambda_1) = d$, giving

$$\begin{aligned} h_{\mathcal{E}_\theta}(d) = \min_{\lambda_1, \lambda_2 \ge 0}\; & 2\overline{w}(\mathbf{1}_M^\top \lambda_1 + \mathbf{1}_M^\top \lambda_2) \\ \text{s.t.}\quad & A^\top(\lambda_2 - \lambda_1) = d. \end{aligned} \tag{65}$$

Let $\lambda_d = \lambda_2 - \lambda_1$. Choosing $\lambda_2 = (\lambda_d)_+$ and $\lambda_1 = (\lambda_d)_-$ yields

$$h_{\mathcal{E}_\theta}(d) = 2\overline{w} \min_{\lambda_d:\, A^\top \lambda_d = d} \|\lambda_d\|_1. \tag{66}$$

Therefore,

$$w_d(\Theta_{N_j}(\theta^\star)) \le \min\left(w_d(\Theta),\; 2h_{\mathcal{E}_\theta}(d)\right). \tag{67}$$

If the executed closed-loop state-input trajectory coincides with the planned trajectory, then the predicted width upper bounds the width computed from the realized measurements, since the prediction is computed under worst-case bounded additive disturbances.

C. Choice of Prediction Approach

Both approaches can be used to predict uncertainty shrinkage induced by a candidate policy segment. Approach 1 relies on closed-loop rollouts and therefore captures the effects of tracking error and disturbances on the resulting regression data. The rollouts are independent and can be parallelized across samples. Approach 2 predicts uncertainty shrinkage directly from the planned regression data and avoids forward simulation, making it computationally cheaper. However, this prediction assumes that the executed trajectory matches the planned trajectory and therefore does not account for tracking error. In practice, Approach 1 is preferred when the effect of execution errors on uncertainty reduction is important, while Approach 2 provides a faster approximation when computational resources are limited.

VI. Tube MPC Instantiation of the Proposed Framework

We now describe a tube MPC instantiation of the proposed framework for finite-horizon, goal-directed navigation. In this instantiation, the backup horizon T_B is chosen adaptively and defined at each replanning time t_k as $T_B := t_f - t_k$, corresponding to the remaining mission duration to reach the terminal goal set $\mathcal{G}$. While the framework is formulated in terms of feedback policies, for this instantiation we reason directly in terms of trajectories to simplify exposition. The resulting construction induce feedback policies and is fully consistent with Section IV.

A. Robust Backup Policy via Tube MPC

At planning time t_k, the robust backup policy $\pi_k^{\mathrm{rob},B}$ is generated using tube MPC. In this instantiation, tube MPC is used to compute a nominal state-input trajectory $(p_{k,x}^{\mathrm{rob}},\, p_{k,u}^{\mathrm{rob}})$ together with a tube cross-section $\mathcal{E}_k(t)$, defined as

$$\mathcal{E}_k(t) = \mathcal{E}(t; \Theta_k,\, \overline{w},\, p_{k,x}^{\mathrm{rob}}(t),\, p_{k,u}^{\mathrm{rob}}(t)), \tag{68}$$

which bounds the deviation between the true state and the nominal trajectory for all $\theta \in \Theta_k$ and all admissible disturbances. The resulting robust tube is

$$\Omega_k^{\mathrm{rob}}(t) = p_{k,x}^{\mathrm{rob}}(t) \oplus \mathcal{E}_k(t) = \{\, x \in \mathcal{X} : x - p_{k,x}^{\mathrm{rob}}(t) \in \mathcal{E}_k(t) \,\}. \tag{69}$$

To enforce robust constraint satisfaction, the nominal trajectory is required to satisfy tightened state, input, and terminal constraints defined as

$$\overline{\mathcal{S}}_k(t) = \mathcal{S} \ominus \mathcal{E}_k(t), \quad \overline{\mathcal{U}}_k(t) = \mathcal{U} \ominus \Delta\mathcal{U}_k(t), \tag{70a}$$

$$\overline{\mathcal{G}}_k = \mathcal{G} \ominus \mathcal{E}_k(t_f), \tag{70b}$$

where $\Delta\mathcal{U}_k(t)$ bounds the input deviation induced by the ancillary controller. Satisfaction of the tightened constraints by the nominal trajectory implies satisfaction of the original constraints by the closed-loop system.

The nominal trajectory is obtained by solving a finite-horizon optimal control problem over the backup horizon $T_B = t_f - t_k$,

$$\min_{p_x(\cdot),\, p_u(\cdot)} \quad \int_{t_k}^{t_k+T_B} \ell(p_x(t), p_u(t))\, dt + \ell_T(p_x(t_k + T_B)) \tag{71a}$$

$$\text{s.t.} \quad \dot{p}_x(t) = f\left(p_x(t), \hat{\theta}_f\right) + g\left(p_x(t), \hat{\theta}_g\right) p_u(t), \tag{71b}$$

$$p_x(t) \in \overline{\mathcal{S}}_k(t), \quad \forall t \in [t_k, t_k + T_B] \tag{71c}$$

$$p_u(t) \in \overline{\mathcal{U}}_k(t), \quad \forall t \in [t_k, t_k + T_B], \tag{71d}$$

$$p_x(t_k + T_B) \in \overline{\mathcal{G}}_k. \tag{71e}$$

Given the nominal solution, an ancillary feedback law $\pi_k^{\mathrm{arc}} : \mathbb{R} \times \mathcal{X} \to \mathcal{U}$ is designed such that the control input

$$u(t) = p_{k,u}^{\mathrm{rob}}(t) + \pi_k^{\mathrm{arc}}(t, x(t)) \tag{72}$$

renders $\Omega_k^{\mathrm{rob}}(t)$ forward invariant. The resulting feedback law defines the robust backup policy

$$\pi_k^{\mathrm{rob},B}(t, x) = p_{k,u}^{\mathrm{rob}}(t) + \pi_k^{\mathrm{arc}}(t, x(t)), \tag{73}$$

which guarantees $x(t) \in \mathcal{S}$, $u(t) \in \mathcal{U}$, and $x(t_f) \in \mathcal{G}$ for all admissible uncertainties when applied over $[t_k, t_k + T_B]$.

B. Predicted Mission Cost under Tube MPC

Recall from Definition 6 that the predicted mission cost of executing a policy π_k from state x_k over the horizon $\mathcal{T}_k = [t_k, t_f]$ is denoted by $\mathcal{J}^{k\to f}(\pi_k, x_k, \mathcal{T}_k)$.

In the tube MPC instantiation, the predicted mission cost associated with the robust backup policy $\pi_k^{\mathrm{rob},B}$ is defined as

$$\mathcal{J}_{\mathrm{rob}}^{k\to f} \;=\; \mathcal{J}^{k\to f}\left(\pi_k^{\mathrm{rob},B}, x_k, \mathcal{T}_k\right), \tag{74}$$

and is evaluated using the nominal centerline trajectory returned by the tube MPC optimization,

$$\mathcal{J}_{\mathrm{rob}}^{k\to f} = \int_{t_k}^{t_f} \ell(p_{k,x}^{\mathrm{rob}}(t), p_{k,u}^{\mathrm{rob}}(t))\, dt + \ell_T(p_{k,x}^{\mathrm{rob}}(t_f)).$$

The mission cost is evaluated using the nominal centerline trajectory rather than the executed closed-loop trajectory. While the ancillary feedback controller may introduce tracking deviations to ensure tube invariance, these deviations do not affect the feasibility or safety guarantees and are not explicitly accounted for in the mission-level planning objective. The predicted mission cost therefore serves as a consistent planning-time metric for enforcing the mission-level budget constraint.

Algorithm 2: Instantiation I of Proposed Framework

Input: Current state x_k, uncertainty set Θ^k, exploration budget B_{exp}
Output: Committed policy segment π_k^{com}

Compute robust backup trajectory $(p_{k,x}^{\mathrm{rob}}, p_{k,u}^{\mathrm{rob}})$ via tube MPC;
Construct robust tube $\Omega_k^{\mathrm{rob}}(t)$ and backup policy $\pi_k^{\mathrm{rob},B}$;
Generate candidate horizons $\mathcal{T}_i^{c,k} = [t_k,\, t_k + iT_c]$;
for $i = 1, \ldots, N_k$ **do**
  Define conservative candidate $\pi_{k,i}^{\mathrm{rob},B}$ as the restriction of $\pi_k^{\mathrm{rob},B}$;
  Solve informative trajectory optimization to obtain $p_k^{\mathrm{info},i}$;
  Attempt to construct tube $\Omega_k^{\mathrm{info},i}$ around $p_k^{\mathrm{info},i}$;
  if *a robust tube* $\Omega_k^{\mathrm{info},i}$ *exists* **then**
    mark candidate pair as valid;
  else
    reject candidate;
  end
end
Predict uncertainty reduction $\Delta\xi_i$ using Algorithm 1;
Evaluate exploration cost $\Delta J_{\mathrm{exp}}^k(i)$;
Form the feasible candidate set satisfying the exploration budget constraint;
if *feasible candidates exist* **then**
  select $i^\star = \arg\max s_i^{c,k}$;
  commit informative candidate $\pi_{k,i^\star}^{\mathrm{rob},I}$;
else
  commit conservative segment $\pi_{k,1}^{\mathrm{rob},B}$;
end
return π_k^{com};

C. Conservative Candidate Policy Segment

For each candidate horizon

$$\mathcal{T}_i^{c,k} = [t_k,\, t_k + iT_c], \tag{75}$$

the conservative candidate policy segment is defined exactly as in Definition 8 as the restriction of the robust backup policy:

$$\pi_{k,i}^{\mathrm{rob},B} \;=\; \pi_k^{\mathrm{rob},B}|_{\mathcal{T}_i^{c,k}}. \tag{76}$$

Equivalently, the associated conservative candidate trajectory $p_k^{\mathrm{rob},i}$ is obtained by restricting the robust tube trajectory to the same horizon. No additional optimization is performed at this step.

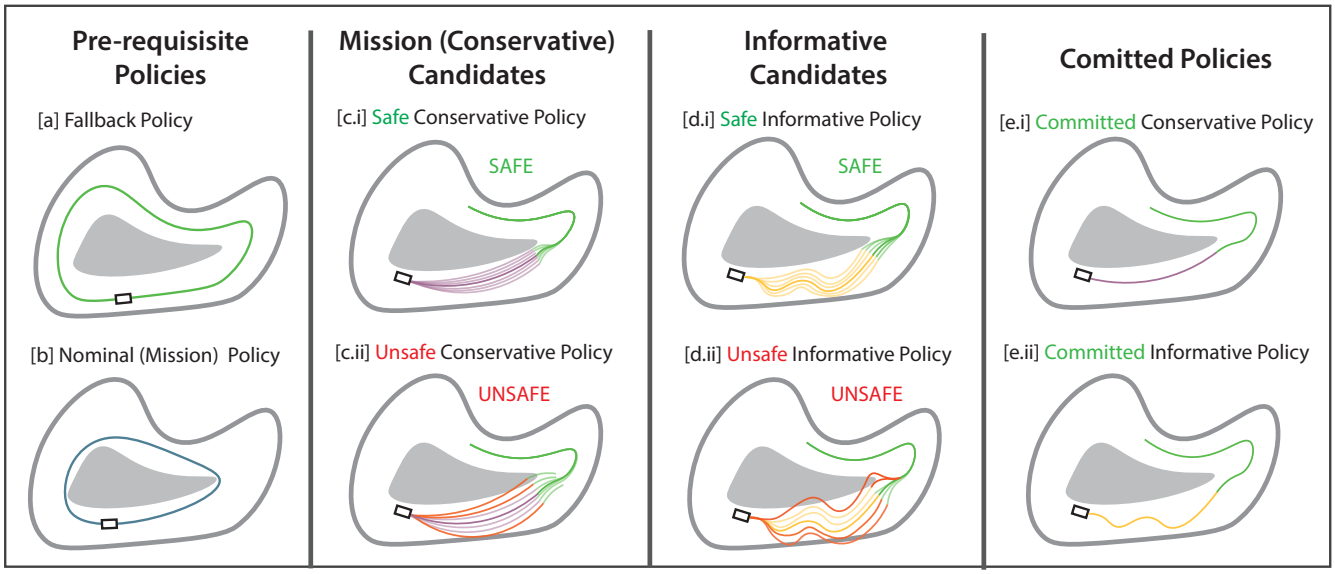

Fig. 4: Illustration of the gatekeeper instantiation. Starting from prerequisite policies—a backup policy and a nominal mission policy—the framework generates conservative and informative candidate policies. Candidates that cannot be certified as safe are rejected. Among the remaining candidates, a safe conservative or safe informative policy may be committed for execution based on the score.

D. Informative Candidate Trajectory Generation

For each horizon $\mathcal{T}_i^{c,k}$, we compute an informative candidate trajectory $p_k^{\text{info},i} = (p_{k,x}^{\text{info},i}, p_{k,u}^{\text{info},i})$ by solving

$$\min_{x(\cdot),u(\cdot)} \quad \int_{t_k}^{t_k+iT_c} \ell(x(\tau),u(\tau))\,d\tau \;-\; \gamma \log\det(\mathcal{I}_{k,i}+\epsilon I) \tag{77a}$$

$$\text{s.t.} \quad \dot{x}(t) = f(x(t),\hat{\theta}_f) + g(x(t),\hat{\theta}_g)u(t), \tag{77b}$$

$$x(t_k) = x_k, \tag{77c}$$

$$x(t_k+iT_c) = p_{k,x}^{\text{rob},i}(t_k+iT_c). \tag{77d}$$

The information matrix associated with candidate i is defined as

$$\mathcal{I}_{k,i} := \int_{t_k}^{t_k+iT_c} \Phi(x(\tau),u(\tau))^\top W_\theta \Phi(x(\tau),u(\tau))\,d\tau, \tag{78}$$

where $W_\theta \in \mathbb{S}_{++}$ is a user-selected weighting matrix, $\gamma \in \mathbb{R}_{>0}$ is the weight on the information reward, and $\epsilon \in \mathbb{R}_{>0}$ is a small regularization constant. The terminal constraint enforces the terminal recoverability condition of Definition 9 by requiring that the informative trajectory rejoins the conservative candidate induced by the robust backup policy.

Remark 4. Robustness to bounded uncertainties and satisfaction of safety constraints are not explicitly enforced during informative candidate generation. After computing $p_k^{\text{info},i}$, safety is assessed by attempting to construct a tube $\Omega_k^{\text{info},i}$ around the informative trajectory. If such a tube exists, the informative candidate admits a robust realization and is referred to as a *safe informative candidate*; otherwise, it is rejected.

E. Candidate Policy Segment Pair and Validity

The informative policy segment $\pi_{k,i}^I$, and the conservative candidate policy segment $\pi_{k,i}^{\text{rob},B}$ together form the candidate policy segment pair $(\pi_{k,i}^I,\ \pi_{k,i}^{\text{rob},B})$, as defined in Definition 10. A candidate pair is said to be *valid* if there exists a corresponding robust informative policy segment $\pi_{k,i}^{\text{rob},I}$ (Definition 11) whose closed-loop execution satisfies state and input constraints for all admissible uncertainties over $\mathcal{T}_i^{c,k}$. Among all valid candidate pairs, the framework selects a candidate that satisfies the budget constraint based on the predicted mission cost. If no informative candidate is valid and budget-feasible, the system commits to the robust backup policy.

VII. `gatekeeper` (Safety Filter) Instantiation of the Proposed Framework

We now present an alternative instantiation of the proposed framework based on the `gatekeeper` architecture. In contrast to the tube MPC instantiation in Section VI, which enforces safety through robust trajectory planning and tube invariance, this instantiation separates nominal trajectory generation from safety certification. Safety is enforced by a `gatekeeper` module that evaluates candidate policies through forward simulation under admissible uncertainty realizations and executes a policy only if it satisfies the required safety conditions with high confidence. This instantiation is particularly suitable for systems where high-performance planners are available but do not explicitly account for model uncertainty.

Fig. 4 illustrates the policies considered in the `gatekeeper` instantiation for the autonomous car racing example. The process begins with a fallback policy that provides a conservative safe behavior robust to the initial uncertainty set. Since the uncertainty set is refined online

through set-membership updates and only shrinks over time, this fallback policy remains safe throughout execution. In the racing scenario, it may correspond to a slow trajectory that follows the track centerline with large safety margins. A nominal mission trajectory is then generated by a high-performance planner, and an informative trajectory is generated to reduce parametric uncertainty. Unlike in the tube-based instantiation, these trajectories are not robust by construction. Because they are computed using the current parameter estimate, they may violate safety constraints when the true parameter differs from the estimate. Their safety must therefore be verified before execution using `gatekeeper`.

For each candidate horizon, the framework constructs two candidate policies by restricting the nominal mission and informative trajectories to that horizon. Each policy applies its corresponding candidate segment over the candidate horizon and then switches to the fallback policy thereafter. The `gatekeeper` evaluates each policy under the current uncertainty set to determine whether it can be safely executed. Among the candidate policies that satisfy the safety and budget constraints, the framework selects the one with the highest score for execution. Thus, safety is enforced at the policy-verification stage rather than during nominal trajectory generation.

Algorithm 3: Instantiation II of Proposed Framework

Input: Current state x_k, uncertainty set Θ_k, disturbance set $\mathcal{W}$, exploration budget B_{exp}
Output: Committed policy π_k^{com}

Compute nominal MPC trajectory $(p_{k,x}^{\text{nom}}, p_{k,u}^{\text{nom}})$;
Generate informative trajectory $(p_{k,x}^{\text{info}}, p_{k,u}^{\text{info}})$;
Generate candidate horizons $\mathcal{T}_i^{c,k} = [t_k,\, t_k + iT_c]$;
for $i = 1, \ldots, N_k$ **do**
  Define nominal candidate segment $p_k^{\text{nom},i}$ as the restriction of the nominal trajectory to $\mathcal{T}_i^{c,k}$;
  Define informative candidate segment $p_k^{\text{info},i}$ as the restriction of the informative trajectory to $\mathcal{T}_i^{c,k}$;
  Construct candidate policies $\pi_{k,i}^{\text{nom}}$ and $\pi_{k,i}^{\text{info}}$ by following the corresponding candidate segment and then the fallback policy;
  Evaluate safety of $\pi_{k,i}^{\text{nom}}$ and $\pi_{k,i}^{\text{info}}$ using N rollouts under sampled (θ, w);
  Compute empirical safety probabilities $P_{\text{safe}}(\pi_{k,i}^{\text{nom}})$ and $P_{\text{safe}}(\pi_{k,i}^{\text{info}})$;
  Estimate predicted mission cost from the rollout simulations;
  Predict uncertainty reduction $\Delta\xi_i$ using Algorithm 1;
  Compute the score of each safe candidate satisfying the exploration budget constraint;
end
Form the feasible candidate set satisfying the exploration budget and safety constraint;
if *feasible candidates exist* **then**
  select the candidate with the highest score and commit it as π_k^{com};
else
  set $\pi_k^{\text{com}} = \pi_{k-1}^{\text{com}}$;
end
return π_k^{com};

A. Nominal Mission Policy via MPC

At planning time t_k, a nominal trajectory is generated using a standard model predictive controller that does not explicitly account for parametric uncertainty. In particular, the MPC planner computes a nominal state–input trajectory

$$(p_{k,x}^{\text{nom}},\, p_{k,u}^{\text{nom}})$$

over the horizon T_B by solving the finite-horizon optimal control problem

$$\min_{p_x(\cdot),\, p_u(\cdot)} \quad \int_{t_k}^{t_k+T_B} \ell(p_x(t), p_u(t))\, dt + \ell_T(p_x(t_k+T_B)) \tag{79a}$$

$$\text{s.t.} \quad \dot{p}_x(t) = f\Big(p_x(t), \hat{\theta}_f\Big) + g\Big(p_x(t), \hat{\theta}_g\Big)\, p_u(t), \tag{79b}$$

$$p_x(t_k) = x_k, \tag{79c}$$

$$p_x(t) \in \mathcal{S}, \quad \forall t \in [t_k, t_k + T_B], \tag{79d}$$

$$p_u(t) \in \mathcal{U}, \quad \forall t \in [t_k, t_k + T_B]. \tag{79e}$$

Because the nominal trajectory is computed using the parameter estimate $\hat{\theta}$, it may violate safety constraints when the true parameter θ differs from the estimate. Safety is therefore enforced by the `gatekeeper` verification mechanism described next.

B. Informative Trajectory Generation

At planning time t_k, the informative trajectory is generated by solving an information-seeking control problem using the current parameter estimate $\hat{\theta}$. Specifically, the informative trajectory

$$(p_{k,x}^{\text{info}},\, p_{k,u}^{\text{info}})$$

is obtained by solving

$$\min_{x(\cdot),u(\cdot)} \quad \int_{t_k}^{t_k+T_B} \ell(x(\tau), u(\tau))\, d\tau \;-\; \gamma \log\det(\mathcal{I}_k + \epsilon I) \tag{80a}$$

$$\text{s.t.} \quad \dot{x}(t) = f(x(t), \hat{\theta}_f) + g(x(t), \hat{\theta}_g) u(t), \tag{80b}$$

$$x(t_k) = x_k. \tag{80c}$$

The associated information matrix is defined as

$$\mathcal{I}_k := \int_{t_k}^{t_k+T_B} \Phi(x(\tau), u(\tau))^\top W_\theta \Phi(x(\tau), u(\tau))\, d\tau, \tag{81}$$

where $\Phi(x, u)$ is the regressor defined in (5), $W_\theta \in \mathbb{S}_{++}$ is a user-selected weighting matrix, $\gamma \in \mathbb{R}_{>0}$ weights the information-seeking objective, and $\epsilon \in \mathbb{R}_{>0}$ is a small regularization constant.

C. Candidate Policy Segments

For each candidate horizon

$$\mathcal{T}_i^{c,k} = [t_k,\, t_k + iT_c], \tag{82}$$

the framework constructs two candidate policies by applying a candidate segment over $\mathcal{T}_i^{c,k}$ and then following the fallback policy for all future time.

The first candidate segment is obtained by restricting the

nominal mission trajectory to the horizon $\mathcal{T}_i^{c,k}$:

$$p_k^{\mathrm{nom},i} = p_k^{\mathrm{nom}}|_{\mathcal{T}_i^{c,k}}. \tag{83}$$

The second candidate segment is obtained by restricting the informative trajectory to the same horizon:

$$p_k^{\mathrm{info},i} = p_k^{\mathrm{info}}|_{\mathcal{T}_i^{c,k}}. \tag{84}$$

The corresponding candidate policies are then defined as follows: $\pi_{k,i}^{\mathrm{nom}}$ applies the nominal candidate segment $p_k^{\mathrm{nom},i}$ over $\mathcal{T}_i^{c,k}$ and then follows the fallback policy for all future time, while $\pi_{k,i}^{\mathrm{info}}$ applies the informative candidate segment $p_k^{\mathrm{info},i}$ over $\mathcal{T}_i^{c,k}$ and then follows the same fallback policy for all future time.

The overall construction is illustrated in Figure 4. In particular, panels [c.i] and [c.ii] show safe and unsafe conservative (nominal) candidate policies, respectively, while panels [d.i] and [d.ii] show safe and unsafe informative candidate policies, respectively.

D. Robust `gatekeeper` Safety Verification

Given a candidate policy

$$\pi_{k,i} \in \{\pi_{k,i}^{\mathrm{nom}}, \pi_{k,i}^{\mathrm{info}}\},$$

the `gatekeeper` determines whether it can be safely executed under the current uncertainty set Θ_k. Although each candidate policy is defined over an infinite horizon, its safety is assessed over a finite verification horizon.

For a candidate horizon

$$\mathcal{T}_i^{c,k} = [t_k,\, t_k + iT_c], \tag{85}$$

we define a fallback horizon of duration $T_{\mathrm{fb}} > 0$ and evaluate the resulting trajectory over

$$\mathcal{T}_i^{v,k} = [t_k,\, t_k + iT_c + T_{\mathrm{fb}}]. \tag{86}$$

Over this interval, the system first follows the candidate segment induced by $\pi_{k,i}$ over $\mathcal{T}_i^{c,k}$ and then follows the fallback policy over the remaining interval

$$[t_k + iT_c,\, t_k + iT_c + T_{\mathrm{fb}}]. \tag{87}$$

To assess safety, the `gatekeeper` performs N forward simulations of the closed-loop system under sampled realizations of the uncertain parameters and disturbances:

$$\dot{x}(t) = f(x(t), \theta_f) + g(x(t), \theta_g)u(t) + w(t), \tag{88}$$

where $\theta \sim \mathcal{U}(\Theta_k)$ and $w(t) \sim \mathcal{U}(\mathcal{W})$. Each rollout is initialized at the current state x_k and applies the control induced by the candidate policy over the verification horizon $\mathcal{T}_i^{v,k}$.

Let $\mathcal{X}_{\mathrm{fb}} \subseteq \mathcal{S}$ denote the fallback set. A rollout is declared safe if all state and input constraints are satisfied over $\mathcal{T}_i^{v,k}$ and the terminal state at the end of the fallback horizon satisfies

$$x(t_k + iT_c + T_{\mathrm{fb}}) \in \mathcal{X}_{\mathrm{fb}}. \tag{89}$$

This terminal fallback-set $\mathcal{X}_{\mathrm{fb}}$ plays the role of the backup-set condition in the standard `gatekeeper` formulation [9].

Let $\mathcal{S}^{(\ell)}(\pi_{k,i}) \in \{0,1\}$ denote the safety indicator for rollout ℓ, where $\mathcal{S}^{(\ell)}(\pi_{k,i}) = 1$ if the above conditions are satisfied, and $\mathcal{S}^{(\ell)}(\pi_{k,i}) = 0$ otherwise. The empirical safety probability of the candidate policy is then defined as

$$P_{\mathrm{safe}}(\pi_{k,i}) = \frac{1}{N}\sum_{\ell=1}^{N} \mathcal{S}^{(\ell)}(\pi_{k,i}). \tag{90}$$

The candidate policy is declared safe if

$$P_{\mathrm{safe}}(\pi_{k,i}) \geq 1 - \delta, \tag{91}$$

where $\delta \in (0,1)$ is a user-specified risk tolerance. If this condition is satisfied, the candidate may be committed for execution; otherwise, it is rejected by the `gatekeeper`. In the racing instantiation, the fallback set is chosen as a neighborhood of the track centerline. Here, we use empirical safety probability as the acceptance criterion, but other risk measures [47], such as conditional value-at-risk (CVaR), could also be used to quantify rollout risk.

E. Valid Candidate Policy Segment Pairs

Having defined the `gatekeeper` safety verification procedure, we now specify validity in this instantiation. For each candidate horizon $\mathcal{T}_i^{c,k}$, the framework constructs a nominal candidate policy $\pi_{k,i}^{\mathrm{nom}}$ and an informative candidate policy $\pi_{k,i}^{\mathrm{info}}$. The pair is declared valid if both policies are certified safe by `gatekeeper` under the current uncertainty set Θ_k.

In the notation of *Def.* 11, $\pi_{k,i}^{\mathrm{nom}}$ corresponds to the conservative policy segment $\pi_{k,i}^{\mathrm{rob},B}$, while a certified-safe informative policy $\pi_{k,i}^{\mathrm{info}}$ corresponds to the robustified informative policy segment $\pi_{k,i}^{\mathrm{rob},I}$. Accordingly, a valid candidate policy segment pair in this instantiation is written as

$$(\pi_{k,i}^{\mathrm{rob},I},\, \pi_{k,i}^{\mathrm{rob},B}) = (\pi_{k,i}^{\mathrm{info}},\, \pi_{k,i}^{\mathrm{nom}}),$$

only when both $\pi_{k,i}^{\mathrm{info}}$ and $\pi_{k,i}^{\mathrm{nom}}$ are certified safe by `gatekeeper`. Once validity is established, budget feasibility is then evaluated.

VIII. Results & Discussion

We validate the proposed framework through two case studies that illustrate two different safety instantiations: (a) quadrotor navigation and (b) autonomous car racing. The quadrotor navigation example demonstrates the tube MPC instantiation described in Section VI, while the autonomous car racing example illustrates the gatekeeper-based safety filtering instantiation. In both cases, the framework evaluates conservative and informative candidate trajectories and commits a trajectory only when it satisfies the safety and budget constraints while providing the highest predicted uncertainty reduction.

A. Safe Quadrotor Navigation

We validate the framework on a quadrotor navigation task. Backup trajectories are generated with tube MPC [2] and tracked using a sliding mode controller. The cost functional,

$$J(p_x, p_u) = \int_{t_0}^{t_f} (\alpha\|u(t)\|^2 + \beta\|p_x(t) - r_{\mathrm{goal}}\|^2)\, dt, \tag{92}$$

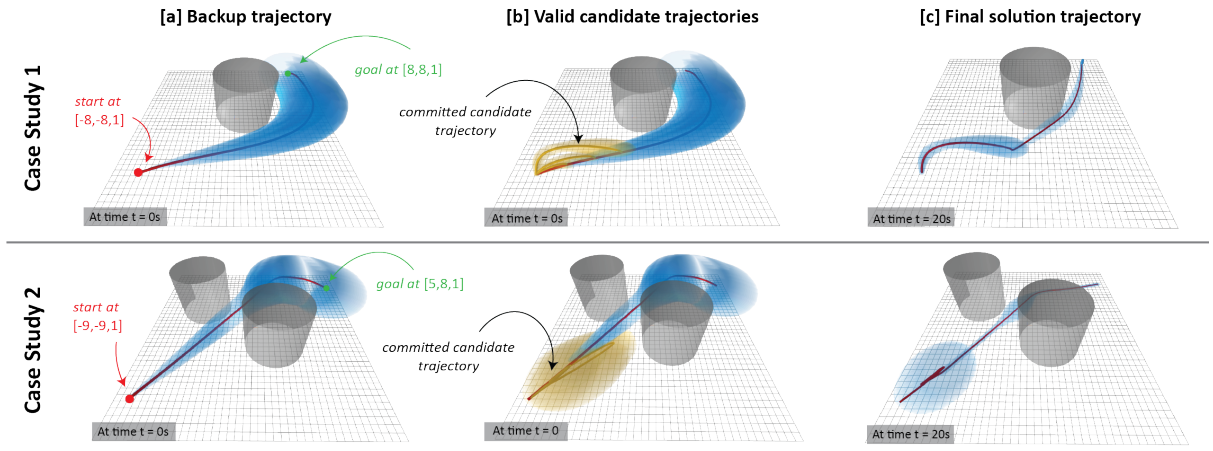

Fig. 5: Backup, candidate, and final solution trajectories for Case Study 1 (top) and Case Study 2 (bottom).

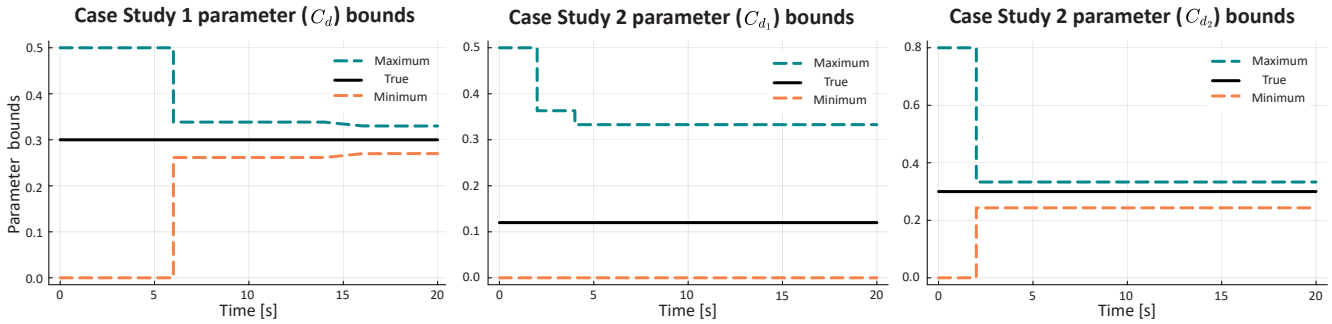

Fig. 6: Parameter bound evolution are shown for two studies: Case Study 1 (C_d, left) and Case Study 2 (C_{d_1}, middle; C_{d_2}, right).

System	Method	Budget (%)	Total Cost (%)	Uncertainty Reduction (%)	
				Param 1	Param 2
1	Baseline [2]	–	100.0	0	–
	Proposed	110.0	82.5	88.0	–
2	Baseline [2]	–	100.0	0	0
	Proposed	110.0	81.3	34.0	88.8

TABLE I: Budget and cost are shown as percentages relative to the baseline solution (100%). For System 2, reductions are reported per parameter dimension.

penalizes control effort and deviation from the goal, where $\alpha, \beta > 0$ are weights, $p_x(t)$ the nominal state, $u(t)$ the input, and r_{goal} the goal. In both cases we set $T_C = 2.0$s.

*1) **Case Study 1: Quadrotor with Drag**:* The first case study, illustrated in Fig. 5, considers a quadrotor modeled as a double integrator with nonlinear aerodynamic drag,

$$\ddot{r} = -C_d \|\dot{r}\| \dot{r} + g + u + d, \tag{93}$$

where $r \in \mathbb{R}^3$ is the inertial position, $g \in \mathbb{R}^3$ the gravitational acceleration, $u \in \mathbb{R}^3$ the control input, $d \in \mathbb{R}^3$ an additive disturbance, and $C_d \in \mathbb{R}$ the unknown drag coefficient. The measurement $y \in \mathbb{R}^3$ corresponds to r.

As shown in Fig. 6, committing a 6-second informative trajectory reduced the admissible interval for C_d, tightening the bounds around the true parameter and demonstrating the framework's ability to shrink parametric uncertainty online.

Table I reports the corresponding mission cost. Relative to the conservative baseline (set to 100%), the proposed method achieved only 82.5% of the cost, while remaining within the budget of 110%. Thus, the approach not only reduced parameter uncertainty but also improved overall efficiency compared to the baseline backup solution.

*2) **Case Study 2: Quadrotor with Vector Drag**:* We extend the setup of Case Study 1 by considering a quadrotor with vector drag dynamics:

$$\ddot{r} = -C_{d_1} \dot{r} - C_{d_2} \|\dot{r}\| \dot{r} + g + u + d, \tag{94}$$

where $C_{d1}, C_{d2} \in \mathbb{R}$ are the unknown drag coefficients.

In this case, the robot executed informative trajectories that reduced the parameter set in both directions, tightening the bounds from $[0.0, 0.50] \times [0.0, 0.80]$ to $[0.0, 0.33] \times [0.25, 0.34]$. The asymmetric shrinkage reflects the relative excitation of the regressors: more data were collected along C_{d2}, yielding a stronger contraction of its bounds. Since candidate generation did not enforce safety, and the robot operated in a narrow corridor, many informative candidates were invalidated. Consequently, fewer safe informative trajectories were committed, producing only modest reduction in C_{d1} compared to C_{d2}. The total cost was 81.3%, well below the 110% budget, normalized to the 100% baseline.

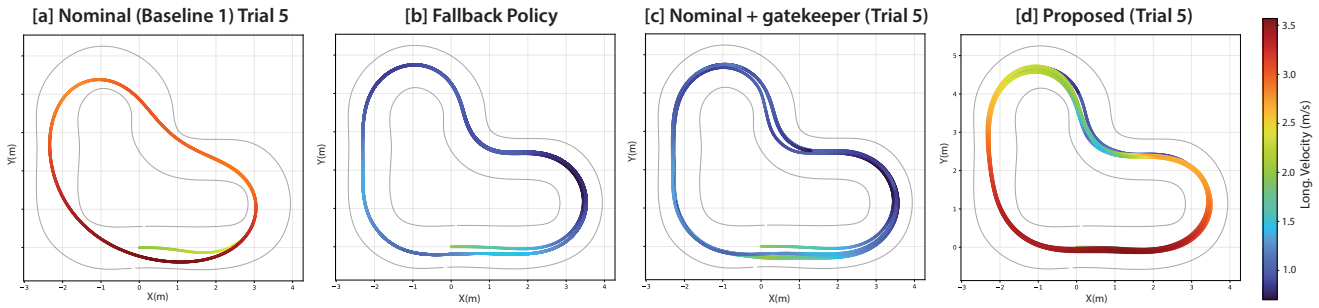

Fig. 7: Trajectories over the last 5 laps for each method. For (a), the successful Trial 5 run is shown. Additional details on the trials can be found in Table III.

Method	Safe Runs (%)	Average Lap Time (s)	Best Final Lap Time (s)	μ Uncertainty Reduction (%)	Budget Consumed (%)
[1] Nominal [6] (No Safety)	30	5.01	4.92	–	–
[2] Weighted Nominal–Informative (No Safety)	20	7.87	7.52	–	–
[3] Fallback Only	100	17.0	16.8	–	–
[4] Nominal + `gatekeeper`	100	15.7	12.5	–	–
[5] [2] + `gatekeeper`	80	21.6	17.5	96.2	–
Proposed: `Dual-gatekeeper`	100	8.07	6.48	95.5	23.8

TABLE II: Performance comparison for the autonomous racing task over 10 independent trials per method. Each trial is executed for at most 10 laps. A trial is counted as a Safe Run only if all 10 laps are completed without any safety violation. Safe Runs reports the percentage of such trials. Average Lap Time is computed over all completed laps, and Final Lap Time denotes the lap time achieved at the end of each trial.

3) Implementation details: In practice, generating a robust tube trajectory based on the method described in [2] requires solving a nonlinear optimization problem and can take anywhere from 100 milliseconds to one second, depending on the problem size and solver warm-start conditions. The generation of an informative candidate trajectory is typically faster, on the order of a few hundred milliseconds, though this is also highly dependent on the system dynamics and horizon length. Both the backup and candidate trajectory optimization problems are implemented using InfiniteOpt.jl, which provides a flexible framework for modeling dynamic optimization problems in Julia.

B. Safe Autonomous Car Racing

We next evaluate the `gatekeeper`-based instantiation of the framework on an autonomous car racing task. In this case study, the objective is to complete a lap safely while actively reducing parametric uncertainty in the tire–road interaction model. The racing dynamics are given in Appendix C.1, with the tire friction coefficient μ modeled as an unknown but bounded parameter. A linear-in-parameter representation of the dynamics is provided in Appendix C.2. The nominal mission trajectory is computed using a linearized MPC solved via sequential quadratic programming (SQP), with an average solve time of approximately 10ms. Our implementation builds on the open-source racing MPC framework of [6]. Informative trajectories are generated using model predictive path integral control (MPPI) implemented in JAX, with an average solve time of approximately 15ms.

The uncertainty in μ directly affects the vehicle's lateral force generation and therefore has a significant impact on both safety and performance during aggressive racing maneuvers. Reducing this uncertainty enables less conservative behavior and improved lap performance. Additional details on the racing model, controller design, and simulation setup are deferred to Appendix C.

The gatekeeper instantiation uses three policy components: a fallback policy, a nominal mission policy, and an informative policy. The fallback policy provides conservative safe behavior under uncertainty, the nominal policy generates high-performance racing behavior aimed at minimizing lap time, and the informative policy excites the dynamics to reduce uncertainty in the tire friction coefficient μ. We test the following baselines in this paper.

- **Baseline 1:** Executes only the nominal controller.
- **Baseline 2:** Optimizes a weighted objective of lap time and uncertainty reduction without safety guarantees.
- **Baseline 3:** Executes only the fallback policy.
- **Baseline 4:** `gatekeeper` evaluating only nominal segments with fallback; no informative candidates.
- **Baseline 5:** `gatekeeper` evaluating candidates from a weighted objective; no pure nominal candidates.
- **Proposed framework.** Generates both nominal and informative candidates and commits the best feasible one.

The results are summarized in Table II, with detailed trial-wise outcomes provided in Table III.

First, Baselines 1 and 2 exhibit poor safety performance,

Trial	μ_{planned}	μ_{true}	Success (First, Last Lap Time over 10 Laps)				
			Baseline 1	Baseline 2	Baseline 4	Baseline 5	Proposed
1	0.28	0.90	×	×	✓ (12.7, 12.9)	✓ (20.6, 18.6)	✓ (17.4, 7.60)
2	0.47	0.90	×	×	✓ (13.9, 13.4)	×	✓ (17.2, 7.32)
3	0.64	0.90	×	×	✓ (13.4, 15.4)	✓ (21.6, 19.1)	✓ (17.7, 6.89)
4	0.81	0.90	✓ (5.58, 4.92)	×	✓ (15.7, 16.0)	✓ (21.6, 18.3)	✓ (16.9, 7.20)
5	0.90	0.90	✓ (5.54, 5.01)	✓ (8.44, 7.52)	✓ (15.4, 14.7)	✓ (23.8, 17.7)	✓ (17.1, 6.52)
6	1.12	0.90	✓ (5.58, 5.10)	✓ (8.06, 7.90)	✓ (17.6, 15.8)	✓ (20.6, 18.6)	✓ (17.3, 6.57)
7	1.36	0.90	×	×	✓ (16.9, 17.1)	✓ (21.4, 18.4)	✓ (17.7, 7.22)
8	1.58	0.90	×	×	✓ (17.2, 17.3)	×	✓ (17.5, 7.21)
9	1.73	0.90	×	×	✓ (17.0, 17.1)	✓ (21.2, 18.4)	✓ (18.2, 7.72)
10	1.95	0.90	×	×	✓ (17.2, 17.0)	✓ (23.7, 18.9)	✓ (17.9, 7.32)

TABLE III: Trial-wise outcomes with $\mu_{\text{planned}} \sim \mathcal{U}(0.2, 2.0)$ and fixed true friction $\mu_{\text{true}} = 0.90$. A checkmark denotes completion of 10 laps without collision (with first and last lap times in seconds), while × denotes failure.

completing only 30% and 20% of trials successfully, respectively. As shown in Table III, failures occur across a wide range of μ_{planned}, indicating that both purely nominal and weighted exploration strategies are unable to maintain safety under model mismatch.

In contrast, the `gatekeeper`-based methods substantially improve safety, but the rollout-based `gatekeeper` test remains probabilistic and therefore can still admit trajectories that violate constraints under realizations not captured during verification. This is evident in the results: Baseline 5 achieves only 80% safe runs despite using the gatekeeper. Baseline 3 (fallback only) is highly conservative, resulting in significantly larger lap times. While Baseline 4 permits nominal candidate trajectories that pass the `gatekeeper` test under the current uncertainty set, it does not reduce uncertainty and therefore remains limited by conservativeness.

Baseline 5 reduces uncertainty (96.2%), but its reliance on a weighted objective leads to overly aggressive informative behavior. In particular, a high weight on the information term causes the trajectory to deviate significantly from the racing line, after which the system repeatedly returns to the fallback policy. This results in degraded performance despite successful uncertainty reduction.

The proposed `Dual-gatekeeper` achieves both safety and strong performance, reducing the average lap time to $8.07\,\text{s}$ and achieving 95.5% uncertainty reduction (Table II). From Table III, we observe that after initial laps with higher times, the final lap times consistently decrease to approximately 6.5–$7.7\,\text{s}$, indicating that uncertainty reduction enables progressively less conservative behavior.

Notably, only 23.8% of the exploration budget is utilized, suggesting that a small number of informative trajectories are sufficient to significantly reduce uncertainty. Once the uncertainty is reduced, the framework predominantly commits high-performance trajectories, leading to improved lap times.

Finally, performance remains influenced by the fallback policy, which constrains transitions between trajectories. In particular, even when nominal trajectories are safe, transitioning to the fallback policy may require conservative behavior. Improving the fallback design could further enhance performance.

IX. Limitations and Future Work

First, the framework assumes that the unknown parameters are time-invariant. The corresponding feasible parameter set is updated over time via SMID and shrinks as informative data are incorporated. While standard in set-membership identification, this assumption is restrictive in scenarios with time-varying or drifting parameters (e.g., changing friction or wind). In such cases, the feasible set may not contract monotonically and previously collected data may become inconsistent. Future work will focus on extending the framework to handle time-varying parameters, for example through adaptive or forgetting-based set updates.

Second, the current framework relies on a prescribed exploration budget to determine whether informative trajectories can be executed. In the limiting case where the budget is zero, only the conservative robust policy is executed. However, this limitation arises from the absence of a principled way to quantify the impact of parametric uncertainty on the future mission cost. While reducing uncertainty intuitively leads to less conservative behavior and improved performance, the current framework does not provide a provable relationship between the size of the uncertainty set and the resulting robust cost. Future work will focus on developing methods that explicitly characterize how uncertainty affects future mission cost, enabling certificates that guarantee when uncertainty reduction leads to a net performance improvement, even under zero exploration budget.

Third, budget feasibility is enforced using predicted cost, which preserves the validity of the framework. The prediction of uncertainty reduction is based on planned trajectories and does not account for discrepancies between planned and executed trajectories due to tracking error, disturbances, or model mismatch. As a result, the realized uncertainty reduction may differ from the predicted value used in candidate selection. Future work will focus on execution-aware uncertainty prediction and improved alignment between predicted and realized outcomes.

X. Conclusion

We presented a dual control framework that integrates safety, budget feasibility, and active uncertainty reduction within a unified decision-making architecture. By treating informative trajectories as certifiable decisions and committing only those that satisfy safety and cost constraints, the framework enables reliable execution while systematically reducing parametric uncertainty. Case studies demonstrate improved performance over conservative baselines, achieving lower mission cost and tighter uncertainty bounds. Future work will focus on incorporating safety directly into candidate generation and developing principled methods to quantify the impact of uncertainty reduction on future mission cost.

References

[1] K. B. Naveed, D. R. Agrawal, D. M. Cherenson, H. Lee, A. Gilbert, H. Parwana, V. S. Chipade, W. Bentz, and D. Panagou, "Enabling safety for aerial robots: Planning and control architectures," *arXiv preprint arXiv:2504.08601*, 2025.

[2] B. T. Lopez, J.-J. E. Slotine, and J. P. How, "Dynamic tube mpc for nonlinear systems," in *2019 American Control Conference (ACC)*. IEEE, 2019, pp. 1655–1662.

[3] T. Lew, R. Bonalli, and M. Pavone, "Risk-averse trajectory optimization via sample average approximation," *IEEE Robotics and Automation Letters*, vol. 9, no. 2, pp. 1500–1507, 2023.

[4] A. Sasfi, M. N. Zeilinger, and J. Köhler, "Robust adaptive mpc using control contraction metrics," *Automatica*, vol. 155, p. 111169, 2023.

[5] D. Hanover, A. Loquercio, L. Bauersfeld, A. Romero, R. Penicka, Y. Song, G. Cioffi, E. Kaufmann, and D. Scaramuzza, "Autonomous drone racing: A survey," *IEEE Transactions on Robotics*, vol. 40, pp. 3044–3067, 2024.

[6] H. Xue, E. L. Zhu, J. M. Dolan, and F. Borrelli, "Learning model predictive control with error dynamics regression for autonomous racing," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 13 250–13 256.

[7] M. F. AL-Sunni, H. Almubarak, K. Horng, and J. M. Dolan, "Lla-mpc: Fast adaptive control for autonomous racing," in *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2025, pp. 1969–1976.

[8] M. Krinner, A. Romero, L. Bauersfeld, M. Zeilinger, A. Carron, and D. Scaramuzza, "MPCC++: Model Predictive Contouring Control for Time-Optimal Flight with Safety Constraints," in *Proceedings of Robotics: Science and Systems*, Delft, Netherlands, July 2024.

[9] D. R. Agrawal, R. Chen, and D. Panagou, "gatekeeper: Online safety verification and control for nonlinear systems in dynamic environments," *IEEE Transactions on Robotics*, 2024.

[10] L. Blackmore, M. Ono, A. Bektassov, and B. C. Williams, "A probabilistic particle-control approximation of chance-constrained stochastic predictive control," *IEEE transactions on Robotics*, vol. 26, no. 3, pp. 502–517, 2010.

[11] K. Garg and D. Panagou, "Robust control barrier and control lyapunov functions with fixed-time convergence guarantees," in *2021 American Control Conference (ACC)*, 2021, pp. 2292–2297.

[12] J. J. Choi, D. Lee, K. Sreenath, C. J. Tomlin, and S. L. Herbert, "Robust control barrier–value functions for safety-critical control," in *2021 60th IEEE Conference on Decision and Control (CDC)*, 2021, pp. 6814–6821.

[13] S. Singh, A. Majumdar, J.-J. Slotine, and M. Pavone, "Robust online motion planning via contraction theory and convex optimization," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 5883–5890.

[14] A. Feldbaum, "Theory of dual control," *Autom. Remote Control*, vol. 22, no. 1, pp. 3–19, 1961.

[15] A. Parsi, D. Liu, A. Iannelli, and R. S. Smith, "Dual adaptive mpc using an exact set-membership reformulation," *IFAC-PapersOnLine*, vol. 56, no. 2, pp. 8457–8463, 2023.

[16] L. Barcelos, A. Lambert, R. Oliveira, P. Borges, B. Boots, and F. Ramos, "Dual Online Stein Variational Inference for Control and Dynamics," in *Proceedings of Robotics: Science and Systems*, 2021.

[17] B. Luo, Y. Zhang, A. Dubey, and A. Mukhopadhyay, "Act as you learn: Adaptive decision-making in non-stationary markov decision processes," in *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems*, 2024, pp. 1301–1309.

[18] A. Sasfi, M. N. Zeilinger, and J. Köhler, "Robust adaptive mpc using control contraction metrics," *Automatica*, vol. 155, p. 111169, 2023.

[19] H. Hu, D. Isele, S. Bae, and J. F. Fisac, "Active uncertainty reduction for safe and efficient interaction planning: A shielding-aware dual control approach," *The International Journal of Robotics Research*, vol. 43, no. 9, pp. 1382–1408, 2024.

[20] J. Sieber, A. Didier, and M. N. Zeilinger, "Computationally efficient system level tube-mpc for uncertain systems," *Automatica*, vol. 180, p. 112466, 2025.

[21] W. D. Compton, N. Csomay-Shanklin, C. Johnson, and A. D. Ames, "Dynamic tube mpc: Learning tube dynamics with massively parallel simulation for robust safety in practice," in *2025 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2025, pp. 2613–2619.

[22] T. Lew, R. Bonalli, and M. Pavone, "Sample average approximation for stochastic programming with equality constraints," *SIAM Journal on Optimization*, vol. 34, no. 4, pp. 3506–3533, 2024.

[23] J. J. Choi, D. Lee, K. Sreenath, C. J. Tomlin, and S. L. Herbert, "Robust control barrier–value functions for safety-critical control," in *2021 60th IEEE Conference on Decision and Control (CDC)*. IEEE, 2021, pp. 6814–6821.

[24] L. Knoedler, O. So, J. Yin, M. Black, Z. Serlin, P. Tsiotras, J. Alonso-Mora, and C. Fan, "Safety on the fly: Constructing robust safety filters via policy control barrier functions at runtime," *IEEE Robotics and Automation Letters*, 2025.

[25] T. Lew, A. Sharma, J. Harrison, A. Bylard, and M. Pavone, "Safe active dynamics learning and control: A sequential exploration–exploitation framework," *IEEE Transactions on Robotics*, vol. 38, no. 5, pp. 2888–2907, 2022.

[26] A. Parsi, A. Iannelli, and R. S. Smith, "Active exploration in adaptive model predictive control," in *2020 59th IEEE Conference on Decision and Control (CDC)*. IEEE, 2020, pp. 6186–6191.

[27] T. Kim, J. Mun, J. Seo, B. Kim, and S. Hong, "Bridging Active Exploration and Uncertainty-Aware Deployment Using Probabilistic Ensemble Neural Network Dynamics," in *Proceedings of Robotics: Science and Systems*, Daegu, Republic of Korea, July 2023.

[28] A. Davydov, F. Djeumou, M. Greiff, M. Suminaka, M. Thompson, J. Subosits, and T. Lew, "First, learn what you don't know: Active information gathering for driving at the limits of handling," *IEEE Robotics and Automation Letters*, 2025.

[29] M. Hibbard, A. P. Vinod, J. Quattrociocchi, and U. Topcu, "Safely: safe stochastic motion planning under constrained sensing via duality," *IEEE Transactions on Robotics*, vol. 39, no. 5, pp. 3464–3478, 2023.

[30] M. Prajapat, J. Köhler, M. Turchetta, A. Krause, and M. N. Zeilinger, "Safe guaranteed exploration for non-linear systems," *IEEE Transactions on Automatic Control*, 2025.

[31] A. Mesbah, "Stochastic model predictive control with active uncertainty learning: A survey on dual control," *Annual Reviews in Control*, vol. 45, pp. 107–117, 2018.

[32] Z. Li, W.-H. Chen, and J. Yang, "A dual control perspective for exploration and exploitation in autonomous search," in *2022 European Control Conference (ECC)*. IEEE, 2022, pp. 1876–1881.

[33] E. Arcari, L. Hewing, M. Schlichting, and M. Zeilinger, "Dual stochastic mpc for systems with parametric and structural uncertainty," in *Learning for Dynamics and Control*. PMLR, 2020, pp. 894–903.

[34] J. W. Knaup and P. Tsiotras, "Adaptive dual covariance steering with active parameter estimation," in *2024 IEEE 63rd Conference on Decision and Control (CDC)*. IEEE, 2024, pp. 659–664.

[35] B. Johnson, Q. Zhu, R. Prucka, M. Barron, M. Figueroa-Santos, and M. Castanier, "Implicit dual-control for visibility-aware navigation in unstructured environments," *arXiv preprint arXiv:2507.04371*, 2025.

[36] R. Soloperto, J. Köhler, and F. Allgöwer, "Augmenting mpc schemes with active learning: Intuitive tuning and guaranteed performance," *IEEE Control Systems Letters*, vol. 4, no. 3, pp. 713–718, 2020.

[37] B. Zhang, Z. Zhou, and R. Vasudevan, "Provably-Safe, Online System Identification ," in *Proceedings of Robotics: Science and Systems*, Los Angeles, CA, USA, July 2025.

[38] B. T. Lopez, "Adaptive robust model predictive control for nonlinear systems," Ph.D. dissertation, MIT, 2019.

[39] F. Li, M. Fu, W. Chen, F. Zhang, H. Zhang, H. Qu, and Z. Yi, "Improving exploration in actor–critic with weakly pessimistic value

estimation and optimistic policy optimization," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 35, no. 7, pp. 8783–8796, 2024.
[40] B. Tasdighi, N. Werge, Y.-S. Wu, and M. Kandemir, "Exploring pessimism and optimism dynamics in deep reinforcement learning," in *Seventeenth European Workshop on Reinforcement Learning*, 2024.
[41] T. Moskovitz, J. Parker-Holder, A. Pacchiano, M. Arbel, and M. Jordan, "Tactical optimism and pessimism for deep reinforcement learning," *Advances in Neural Information Processing Systems*, vol. 34, pp. 12 849–12 863, 2021.
[42] K. B. Naveed, D. R. Agrawal, and D. Panagou, "A formal gatekeeper framework for safe dual control with active exploration," *arXiv preprint arXiv:2510.06351*, 2025.
[43] M. Cohen and C. Belta, *Adaptive Control Lyapunov Functions*. Cham: Springer International Publishing, 2023, pp. 57–76. [Online]. Available: https://doi.org/10.1007/978-3-031-29310-8_4
[44] K. S. Narendra and A. M. Annaswamy, "Persistent excitation in adaptive systems," *International Journal of Control*, vol. 45, no. 1, pp. 127–160, 1987.
[45] M. Cohen and C. Belta, "Robust safety-critical control for systems with actuation uncertainty," in *Adaptive and Learning-Based Control of Safety-Critical Systems*. Springer, 2023, pp. 117–131.
[46] M. Milanese and C. Novara, "Set membership identification of nonlinear systems," *Automatica*, vol. 40, no. 6, pp. 957–975, 2004.
[47] A. Majumdar and M. Pavone, "How should a robot assess risk? towards an axiomatic theory of risk in robotics," in *Robotics Research: The 18th International Symposium ISRR*. Springer, 2019, pp. 75–84.

APPENDIX

A. Interpretation of the Predicted Width

This subsection interprets the predicted uncertainty width by relating it to the width obtained from realized measurements under idealized execution (no tracking error), and is not used by the algorithm or the main theoretical guarantees.

Assumption 3. On $[t_i, t_f]$, the executed (closed-loop) state–input trajectory coincides with the planned candidate $p = (p_x, p_u)$. The stacked regressor induced by the executed samples is $A_{\mathrm{act}} = [\,\Phi(x(t_j), u(t_j))\,]_{j=1}^{N_j} \in \mathbb{R}^{M\times p}$, with $M = N_j c$. Because execution matches the plan, $A_{\mathrm{act}} = A$ by (51). Measurements satisfy $z_j = \Phi_j\theta^\star + w_j$ with $\|w_j\|_\infty \le \overline{w}$ for all $j = 1, \dots, N_j$.

Given Assumption 3, consider the parameter set consistent with the realized measurements,

$$\Theta_{\mathrm{act}} \;:=\; \Big\{\theta \in \Theta :\; \|z - A_{\mathrm{act}}\theta\|_\infty \le \overline{w}\Big\}, \quad z = A_{\mathrm{act}}\theta^\star + w, \tag{95}$$

where A_{act} stacks the regressors induced by the executed samples.

Proposition 1. Under Assumption 3, for all $d \in \mathcal{D}$,

$$w_d(\Theta_{\mathrm{act}}) \;\le\; w_d(\Theta_{N_j}(\theta^\star)) \;\le\; \min\Big(w_d(\Theta),\, 2\,h_{\mathcal{E}_\theta}(d)\Big), \tag{96}$$

where $h_{\mathcal{E}_\theta}(d)$ admits the exact dual form (66).

Proof. Let $e_\theta = \theta - \theta^\star$. From the construction of Θ_{act} and $z = A_{\mathrm{act}}\theta^\star + w$, we have

$$\theta \in \Theta_{\mathrm{act}} \iff \|z - A_{\mathrm{act}}\theta\|_\infty \le \overline{w}, \tag{97a}$$
$$\iff \|A_{\mathrm{act}}(\theta^\star - \theta) + w\|_\infty \le \overline{w}. \tag{97b}$$

Under Assumption 3, $A_{\mathrm{act}} = A$. Hence, for each row $a_j^\top$ of A,

$$|a_j^\top e_\theta - w_j| \le \overline{w} \;\Rightarrow\; |a_j^\top e_\theta| \le \overline{w} + |w_j| \le 2\overline{w}, \quad \forall j.$$

Therefore $\|Ae_\theta\|_\infty \le 2\overline{w}$ (Lemma 2), which implies $e_\theta \in \mathcal{E}_\theta$ and

$$\Theta_{\mathrm{act}} \;\subseteq\; \Theta \cap (\theta^\star + \mathcal{E}_\theta) \;=\; \Theta_{N_j}(\theta^\star). \tag{98}$$

Since the width is monotone under set inclusion,

$$w_d(\Theta_{\mathrm{act}}) \;\le\; w_d(\Theta_{N_j}(\theta^\star)).$$

Finally, translation invariance of the width and Lemma 3 yield

$$w_d(\Theta_{N_j}(\theta^\star)) \;\le\; \min\left(w_d(\Theta),\, w_d(\mathcal{E}_\theta)\right),$$

with $w_d(\mathcal{E}_\theta) = 2\,h_{\mathcal{E}_\theta}(d)$. The dual representation (66) follows from strong LP duality. □

B. Useful Properties

Corollary 1. *This directional width is translation invariant:* $w_d(\theta_0 + \mathcal{C}) = w_d(\mathcal{C})$.

$$\begin{aligned} w_d(\mathcal{C} + a) &= \sup_{x\in\mathcal{C}} d^\top(x + a) \;-\; \inf_{x\in\mathcal{C}} d^\top(x + a) \\ &= (\sup_{x\in\mathcal{C}} d^\top x + d^\top a) \;-\; (\inf_{x\in\mathcal{C}} d^\top x + d^\top a) \\ &= \sup_{x\in\mathcal{C}} d^\top x \;-\; \inf_{x\in\mathcal{C}} d^\top x \\ &= w_d(\mathcal{C}). \end{aligned}$$

C. Autonomous Car Racing Dynamics & LIP

1) Dynamic Bicycle Model: The vehicle is modeled using a planar dynamic bicycle model with uncertain tire-road friction. The state is

$$x = \begin{bmatrix} p_x & p_y & \psi & v_x & v_y & \omega & \delta \end{bmatrix}^\top \in \mathbb{R}^7, \tag{99}$$

where (p_x, p_y) is the global position, ψ the yaw angle, v_x and v_y the body-frame longitudinal and lateral velocities, ω the yaw rate, and δ the steering angle. The control input is

$$u = \begin{bmatrix} F_d & F_b & \dot{\delta}_{\mathrm{cmd}} \end{bmatrix}^\top \in \mathbb{R}^3, \tag{100}$$

where F_d and F_b denote drive and braking forces and $\dot{\delta}_{\mathrm{cmd}}$ is the commanded steering rate.

The global kinematics are

$$\dot{p}_x = v_x\cos\psi - v_y\sin\psi, \tag{101a}$$
$$\dot{p}_y = v_x\sin\psi + v_y\cos\psi, \tag{101b}$$
$$\dot{\psi} = \omega. \tag{101c}$$

The tire slip angles are

$$\alpha_f = \delta - \tan^{-1}\left(\frac{l_f\omega + v_y}{v_x + \epsilon}\right), \quad \alpha_r = \tan^{-1}\left(\frac{l_r\omega - v_y}{v_x + \epsilon}\right), \tag{102}$$

where l_f and l_r denote the distances from the center of mass

to the front and rear axles. The normal loads are

$$F_{z,f} = \frac{mgl_r}{2l}, \qquad F_{z,r} = \frac{mgl_f}{2l}, \tag{103}$$

where $l = l_f + l_r$. Lateral tire forces are modeled using a simplified Pacejka formulation

$$F_{y,f} = \mu F_{z,f} \sin\Big(C_f \tan^{-1}(B_f \alpha_f)\Big), \tag{104a}$$

$$F_{y,r} = \mu F_{z,r} \sin\Big(C_r \tan^{-1}(B_r \alpha_r)\Big), \tag{104b}$$

where μ is the tire friction coefficient. In this study, μ is treated as an unknown but bounded parameter to be learned online.

The longitudinal tire forces are

$$F_{x,f} = \frac{1}{2} k_d F_d + \frac{1}{2} k_b F_b - \frac{1}{2} f_r mg \frac{l_r}{l}, \tag{105a}$$

$$F_{x,r} = \frac{1}{2}(1-k_d) F_d + \frac{1}{2}(1-k_b) F_b - \frac{1}{2} f_r mg \frac{l_f}{l}, \tag{105b}$$

where k_d and k_b denote the front-axle drive and braking distributions, respectively, and f_r is the rolling resistance coefficient. The body-frame dynamics are

$$\dot{v}_x = \frac{1}{m}\Big(2F_{x,r} + 2F_{x,f}\cos\delta - 2F_{y,f}\sin\delta\Big) - \frac{1}{2}\rho A C_d v_x^2 + \omega v_y, \tag{106a}$$

$$\dot{v}_y = \frac{1}{m}\Big(2F_{y,r} + 2F_{y,f}\cos\delta + 2F_{x,f}\sin\delta\Big) - \omega v_x, \tag{106b}$$

$$\dot{\omega} = \frac{1}{J_z}\Big(- 2F_{y,r} l_r + (2F_{y,f}\cos\delta + 2F_{x,f}\sin\delta) l_f\Big), \tag{106c}$$

$$\dot{\delta} = \dot{\delta}_{\mathrm{cmd}}. \tag{106d}$$

2) Linear in Parameter Form: For the uncertainty reduction module, the model is written in linear-in-parameter form as

$$\dot{x} = f_0(x) + g_0(x)u + \Phi(x)\mu + w(t), \tag{107}$$

where $\Phi(x)$ is the regressor associated with the friction parameter and $w(t)$ is a bounded disturbance. Defining the lateral tire forces without the friction coefficient as

$$\bar{F}_{y,f}(x) = F_{z,f} \sin\Big(C_f \tan^{-1}(B_f \alpha_f(x))\Big), \tag{108a}$$

$$\bar{F}_{y,r}(x) = F_{z,r} \sin\Big(C_r \tan^{-1}(B_r \alpha_r(x))\Big), \tag{108b}$$

the regressor takes the form

$$\Phi(x) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -\frac{2}{m}\sin\delta\, \bar{F}_{y,f}(x) \\ \frac{2}{m}(\bar{F}_{y,r}(x) + \cos\delta\, \bar{F}_{y,f}(x)) \\ \frac{1}{J_z}(-2l_r\, \bar{F}_{y,r}(x) + 2l_f \cos\delta\, \bar{F}_{y,f}(x)) \\ 0 \end{bmatrix}. \tag{109}$$

where μ is the tire friction coefficient. In this study μ is treated as an unknown but bounded parameter to be learned online. The resulting dynamics can be written in linear-in-parameter form

$$\dot{x} = f_0(x) + g_0(x)u + \Phi(x)\mu + w(t), \tag{110}$$

where $\Phi(x)$ is the regressor associated with the friction parameter and $w(t)$ is a bounded disturbance.

3) Nominal MPC Planner: The fallback policy corresponds to a conservative controller that follows the track centerline using a pure pursuit strategy with reduced speed, providing a safe fallback behavior that maintains large safety margins with respect to track boundaries.

The nominal MPC solves

$$\min_{x_{0:N}, u_{0:N-1}} \quad \sum_{k=0}^{N-1} \ell(x_k, u_k, \Delta u_k) + \ell_N(x_N), \tag{111}$$

with stage cost

$$\begin{aligned} \ell(x_k, u_k, \Delta u_k) = \; & q_t t_k^2 + q_{e_\psi} e_{\psi,k}^2 \\ & + q_v (v_{x,k} - v_{\mathrm{ref},k})^2 \\ & + q_{v_y} v_{y,k}^2 + q_\omega \omega_k^2 \\ & + u_k^\top R u_k + \Delta u_k^\top R_\Delta \Delta u_k, \end{aligned} \tag{112}$$

where $\ell_N(x_N)$ is the terminal cost.

Informative candidate trajectories are generated using a nonlinear MPPI planner. The MPPI objective augments the racing cost with an information-seeking term (81). The resulting candidate trajectories are passed to the gatekeeper safety verification module, which evaluates safety through forward simulation under sampled uncertainty realizations and commits the trajectory that satisfies safety and budget constraints while achieving the highest predicted uncertainty reduction.

*4) **Fallback Policy**:* A conservative safety policy designed offline. It is implemented as a low-speed pure pursuit controller tracking the centerline with large safety margins across the admissible range of μ. Candidate trajectories are formed by concatenating a policy segment with the fallback policy. Conservative candidates use nominal segments, while informative candidates use informative segments (Figure 4).

*5) **Nominal Mission Policy**:* A performance-oriented controller that tracks a racing line to minimize lap time. It is generated using a linearized MPC formulation (Appendix C.3) and does not account for uncertainty.

*6) **Informative Policy**:* A controller that excites the dynamics to reduce uncertainty in μ. It may deviate from nominal behavior to improve parameter estimation. Trajectories are obtained from an excitation-driven optimization problem (Appendix C).

To build intuition for the error set $\mathbb{E}_\theta$ and the intersection $\Theta \cap (\theta^* + \mathbb{E}_\theta)$, we consider a simple one-dimensional parameter estimation problem.

D. Setup

Suppose a robot is trying to learn an unknown scalar parameter $\theta \in \mathbb{R}$, representing, for example, an aerodynamic drag coefficient.

At time k, the robot maintains a feasible parameter set

$$\Theta_k \subset \mathbb{R}.$$

Assume its current knowledge is

$$\Theta_k = [0.0,\, 0.5].$$

The true (unknown) parameter is

$$\theta^* = 0.3.$$

Measurements are corrupted by bounded noise. Let the noise bound be

$$\bar{w} = 0.05,$$

which implies a worst-case masking level of

$$2\bar{w} = 0.10.$$

E. Informative Excitation

The robot executes an informative trajectory that induces a regressor $\Phi = 2.0$ at each measurement. Suppose it collects 5 measurements. The resulting stacked regressor matrix is

$$A = \begin{bmatrix} 2.0 \\ 2.0 \\ 2.0 \\ 2.0 \\ 2.0 \end{bmatrix}.$$

F. Error Set Construction

We define the error $e_\theta := \theta - \theta^*$. The indistinguishable error set $\mathbb{E}_\theta$ contains all parameter offsets that cannot be ruled out due to noise.

From the measurement model, indistinguishability implies

$$\|Ae_\theta\|_\infty \leq 2\bar{w}.$$

Substituting the values in this example,

$$|2.0 \cdot e_\theta| \leq 0.10.$$

Solving for e_θ gives

$$|e_\theta| \leq 0.05.$$

Thus, the error set is

$$\mathbb{E}_\theta = [-0.05,\, 0.05].$$

G. Shifted Indistinguishable Set

To recover the set of parameters that are indistinguishable from the truth, we shift the error set by the true parameter:

$$\theta^* + \mathbb{E}_\theta = 0.3 + [-0.05,\, 0.05].$$

This yields the indistinguishable parameter region

$$\theta^* + \mathbb{E}_\theta = [0.25,\, 0.35].$$

H. Final Set Update (Intersection)

The robot updates its knowledge by intersecting its prior feasible set with the indistinguishable region:

$$\Theta_{k+1} = \Theta_k \cap (\theta^* + \mathbb{E}_\theta).$$

Substituting the sets,

$$\Theta_{k+1} = [0.0,\, 0.5] \cap [0.25,\, 0.35].$$

Therefore,

$$\Theta_{k+1} = [0.25,\, 0.35].$$

I. Interpretation

This example illustrates the geometry of set-membership identification:

The error set $\mathbb{E}_\theta$ captures all parameter offsets that cannot be distinguished from noise given the excitation level. The regressor magnitude directly determines how tightly this set contracts.

The shifted set $\theta^* + \mathbb{E}_\theta$ represents all parameters that would produce measurements indistinguishable from the true system.

Finally, intersecting with the prior feasible set shrinks the uncertainty. In this example, the uncertainty reduces from a width of 0.5 to 0.1, demonstrating how informative excitation improves parameter certainty.