Set Learning for Black-Box Systems via Dual Gaussian Process Control Barrier Functions

Milad Mesbahi, *M.S.E. Robotics,* Vineet Pasumarti, *M.S.E. Robotics*

***Abstract*—We propose a data–driven framework for learning safe operating regions of black–box robotic systems from interaction. The method maintains two Gaussian process models: one for dynamics residuals and one for a state–space barrier function, learned solely from observable quantities. These models induce a robust control barrier function constraint with GP uncertainty, yielding a convex quadratic program for real–time safety filtering and a safety–aware Bayesian optimizer for controller tuning. Under standard GP confidence assumptions, the framework is compatible with high–probability safety analyses developed for GP–based safe learning, but in this report we focus on an empirical study, demonstrating data–efficient expansion of the certified safe set in simulation while avoiding constraint violations.**

I. Introduction

Safe deployment of autonomous systems requires knowledge of both the dynamics and the states in which the system can operate without violating safety constraints. In practice, simulation- or physics-based models differ from reality due to unmodeled dynamics, parameter drift, and environmental variability, and the true safe operating region is rarely known apriori or is installation-specific (e.g., configuration-dependent limits for manipulators or changing flight envelopes for aerial robots) [1]–[3]. These uncertainties make *safe online learning* challenging: the robot must explore to identify both dynamics and safety margins, yet unsafe actions may cause irreversible damage, motivating data-driven methods that (i) treat dynamics and safety boundaries as unknown functions inferred from noisy observations, (ii) explicitly represent uncertainty in these functions, and (iii) use uncertainty-aware models to certify and expand a safe set during learning.

II. Problem Formulation

We consider a control-affine nonlinear system

$$\dot{\mathbf{x}} = f(\mathbf{x}) + G(\mathbf{x})\mathbf{u} \tag{1}$$

and assume access to a nominal model $f_0(\mathbf{x})$ and $G_0(\mathbf{x})$ obtained from simulation or first principles.

Assumption 1. *[4]–[6] The residual $g(\mathbf{x}) = f(\mathbf{x}) - f_0(\mathbf{x})$ lies in a reproducing kernel Hilbert space (RKHS) $\mathcal{H}_k$ with known kernel $k : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ and bounded norm $|g|_k \leq B$.*

Safety is encoded via a barrier function $h : \mathbb{R}^n \to \mathbb{R}$ defining the safe set: $\mathcal{S} = \{\mathbf{x} \in \mathbb{R}^n : h(\mathbf{x}) \geq 0\}$

Definition 1 (Control Barrier Function [7])**.** *A continuously differentiable function $h : \mathbb{R}^n \to \mathbb{R}$ is a* control barrier function *(CBF) for system* (1) *on the set $\mathcal{S} = \{\mathbf{x} : h(\mathbf{x}) \geq 0\}$ if there exists an extended class-$\mathcal{K}$ function $\alpha : \mathbb{R} \to \mathbb{R}$ such that for all $\mathbf{x} \in \mathcal{S}$,*

$$\sup_{\mathbf{u} \in \mathcal{U}} \left[\nabla h(\mathbf{x})^\top f(\mathbf{x}) + \nabla h(\mathbf{x})^\top G(\mathbf{x})\mathbf{u} + \alpha(h(\mathbf{x})) \right] \geq 0. \tag{2}$$

If h is a CBF and there exists a Lipschitz controller $\pi : \mathbb{R}^n \to \mathbb{R}^m$ satisfying (2), then $\mathcal{S}$ is forward invariant under the closed-loop dynamics [8].

Problem Statement. Since h^* is unknown, we cannot directly enforce (2). Instead, we must learn an estimate of h^* from data. Given a conservative initial safe set $\hat{\mathcal{S}}_0 \subseteq \mathcal{S}^*$, nominal models (f_0, h_0), and confidence level $\delta \in (0, 1)$, our goal is to design an online dual learning and control scheme that, with probability at least $1 - \delta$: (i) keeps the closed-loop state safe, $\mathbf{x}_t \in \mathcal{S}^*$ for all t, (ii) enlarges the certified safe set monotonically, $\hat{\mathcal{S}}_t \subseteq \hat{\mathcal{S}}_{t+1}$, and (iii) encourages $\hat{\mathcal{S}}_t$ to approach $\mathcal{S}^*$ as $t \to \infty$, while allowing performance-oriented control in the interior of $\hat{\mathcal{S}}_t$ [5], [6], [9].

III. Methods

We maintain two Gaussian process (GP) models: one for the dynamics residual g and one for the barrier function h^*. We first establish GP preliminaries, then present each component.

A. Gaussian Process Preliminaries

A GP defines a distribution over functions $\phi : \mathbb{R}^d \to \mathbb{R}$ specified by a mean function $m : \mathbb{R}^d \to \mathbb{R}$ and kernel $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$, written $\phi \sim \mathcal{GP}(m, k)$ [10]. Given observations $\{(\mathbf{z}_i, y_i)\}_{i=1}^N$ with $y_i = \phi(\mathbf{z}_i) + \omega_i$ and $\omega_i \sim \mathcal{N}(0, \sigma^2)$, the posterior at $\mathbf{z} \in \mathbb{R}^d$ is Gaussian with

$$\mu(\mathbf{z}) = \mathbf{k}(\mathbf{z})^\top (\mathbf{K} + \sigma^2 \mathbf{I})^{-1} \mathbf{y}, \tag{3}$$

$$\sigma^2(\mathbf{z}) = k(\mathbf{z}, \mathbf{z}) - \mathbf{k}(\mathbf{z})^\top (\mathbf{K} + \sigma^2 \mathbf{I})^{-1} \mathbf{k}(\mathbf{z}), \tag{4}$$

where $[\mathbf{k}(\mathbf{z})]_i = k(\mathbf{z}, \mathbf{z}_i)$, $[\mathbf{K}]_{ij} = k(\mathbf{z}_i, \mathbf{z}_j)$, and $\mathbf{y} = [y_1, \ldots, y_N]^\top$.

Lemma 1 (Confidence Bound [11])**.** *Let $\phi \in \mathcal{H}_k$ with $\|\phi\|_{\mathcal{H}_k} \leq B$. Define $\beta_t = 2B + 300\gamma_t \log^3(t/\delta)$, where γ_t is the maximum information gain for t samples under k. Then with probability at least $1 - \delta$, for all $t \geq 1$ and $\mathbf{z} \in \mathbb{R}^d$:*

$$|\phi(\mathbf{z}) - \mu_{t-1}(\mathbf{z})| \leq \beta_t^{1/2} \sigma_{t-1}(\mathbf{z}). \tag{5}$$

B. Dynamics Residual Learning

We place a GP prior on the residual: $g \sim \mathcal{GP}(0, k_g)$. At each timestep, we observe

$$\mathbf{y}_t = \dot{\mathbf{x}}_t - f_0(\mathbf{x}_t) - G(\mathbf{x}_t)\mathbf{u}_t + \boldsymbol{\omega}_t, \tag{6}$$

a noisy measurement of $g(\mathbf{x}_t)$, where $\boldsymbol{\omega}_t \sim \mathcal{N}(\mathbf{0}, \sigma_g^2\mathbf{I})$. The dataset $\mathcal{D}_g^t = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^t$ yields posterior mean μ_g^t and covariance Σ_g^t via (3)–(4). The learned dynamics is

$$\hat{f}_t(\mathbf{x}) = f_0(\mathbf{x}) + \mu_g^t(\mathbf{x}). \tag{7}$$

By Lemma 1 and Assumption 1, the true drift satisfies $f(\mathbf{x}) \in \mathcal{E}_t(\mathbf{x})$ with probability at least $1-\delta$, where the *confidence set* is

$$\mathcal{E}_t(\mathbf{x}) = \left\{ \hat{f}_t(\mathbf{x}) + \boldsymbol{\Delta} : \|(\Sigma_g^t(\mathbf{x}))^{-1/2}\boldsymbol{\Delta}\|_2 \le \beta_t^{1/2} \right\}. \tag{8}$$

C. Control Barrier Learning

We model the unknown safe set via a latent control barrier function

$$\mathcal{S}^* = \{x \in \mathcal{X} : h^*(x) \ge 0\}, \tag{9}$$

which is not available to the controller. Instead, we learn a function $h : \mathcal{X} \to \mathbb{R}$ from data and use its sign to define a data-driven safe-set estimate $\hat{\mathcal{S}} = \{x : h(x) \ge 0\}$, in the spirit of non-parametric Gaussian CBFs and supervised CBF–learning approaches that parametrize safety certificates directly from data [12]–[15]. We place a Gaussian process prior $h(\cdot) \sim \mathcal{GP}(0, k_h)$ and collect training data from closed-loop rollouts. During each rollout we observe state sequences $\{x_k\}_{k=0}^T$ together with a binary outcome indicating whether the trajectory remained constraint-satisfying or experienced a violation. From these observables we construct scalar labels

$$y_k = \ell\big(x_k,\ \text{outcome},\ \tau\big),$$

where τ denotes trajectory-level statistics and $\ell : \mathcal{X} \times \{0,1\} \times \mathcal{T} \to \mathbb{R}$ is a task-dependent scoring function. By design, ℓ assigns $y_k > 0$ to states on trajectories with ample safety margin, $y_k < 0$ to states that precede observed violations, and $y_k \approx 0$ to near-miss states with small minimum margin. The map ℓ is constructed solely from measurable features (e.g., distances, detector outputs) and does not require access to h^* or any latent parameters. Some application-specific feature engineering and data selection are therefore unavoidable, but all subsequent safety reasoning is carried out by the learned GP barrier rather than a hand-designed analytic CBF [12].

Conditioning the GP on the dataset $\mathcal{D}_h^t = \{(x_i, y_i)\}_{i=1}^N$ yields posterior mean $\mu_h^t(x)$ and variance $\sigma_h^t(x)$. Our dataset $\mathcal{D}_h^t = \{(x_i, y_i)\}_{i=1}^N$, $\mathbf{K}_h \in \mathbb{R}^{N\times N}$ has entries $[\mathbf{K}_h]_{ij} = k_h(x_i, x_j)$, and $\mathbf{y}_h = [y_1, \dots, y_N]^\top$.

For control, we use a conservative lower confidence bound

$$h_{\text{lcb}}^t(\mathbf{x}) = \mu_h^t(\mathbf{x}) - \beta_h^{1/2}\sigma_h^t(\mathbf{x}), \tag{10}$$

where β_h is chosen per Lemma 1. The certified estimated safe set is

$$\hat{\mathcal{S}}_t = \{\mathbf{x} \in \mathbb{R}^n : h_{\text{lcb}}^t(\mathbf{x}) \ge 0\}. \tag{11}$$

For the squared exponential kernel $k_h(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp(-\|\mathbf{x} - \mathbf{x}'\|_{\mathbf{L}^{-2}}^2/2)$ with $\mathbf{L} = \operatorname{diag}(\ell_1, \dots, \ell_n)$, the gradient of the posterior mean is

$$\nabla\mu_h^t(\mathbf{x}) = \sum_{i=1}^{|\mathcal{D}_h^t|} \alpha_i\, k_h(\mathbf{x}_i, \mathbf{x})\, \mathbf{L}^{-2}(\mathbf{x}_i - \mathbf{x}), \tag{12}$$

where $\boldsymbol{\alpha} = (\mathbf{K}_h + \sigma^2\mathbf{I})^{-1}\mathbf{y}_h$.

D. Safety Filter

Given a nominal control $\mathbf{u}_{\text{nom}}$ (from another policy from RL, LQR, etc.), we compute a safe control by solving a quadratic program that enforces (2) for all dynamics in the confidence set (8).

By Lemma 1, the true drift satisfies $f(\mathbf{x}) = \hat{f}_t(\mathbf{x}) + \boldsymbol{\Delta}$ for some $\boldsymbol{\Delta}$ with $\|(\Sigma_g^t(\mathbf{x}))^{-1/2}\boldsymbol{\Delta}\|_2 \le \beta_t^{1/2}$, with probability at least $1-\delta$. Safety requires (2) to hold for all such $\boldsymbol{\Delta}$:

$$\min_{\substack{\boldsymbol{\Delta}\in\mathbb{R}^n \\ \|(\Sigma_g^t(\mathbf{x}))^{-1/2}\boldsymbol{\Delta}\|_2 \le \beta_t^{1/2}}} \nabla h(\mathbf{x})^\top\big(\hat{f}_t(\mathbf{x}) + \boldsymbol{\Delta} + G(\mathbf{x})\mathbf{u}\big) + \alpha\big(h(\mathbf{x})\big) \ \ge\ 0. \tag{13}$$

Proposition 1. *The robust constraint* (13) *is equivalent to*

$$\nabla h(\mathbf{x})^\top\big(\hat{f}_t(\mathbf{x}) + G(\mathbf{x})\mathbf{u}\big) - \beta_t^{1/2}\big\|(\Sigma_g^t(\mathbf{x}))^{1/2}\nabla h(\mathbf{x})\big\|_2 + \alpha\big(h(\mathbf{x})\big) \ \ge\ 0. \tag{14}$$

This follows from the support function of an ellipsoidal uncertainty set; see, e.g., [6], [16]. In practice, we enforce (14) using the conservative barrier estimate h_{lcb}^t in place of h, and obtain the applied control input u_t by solving a convex quadratic program

$$\begin{aligned} u_t^\star \ &=\ \arg\min_{u\in\mathcal{U}}\ \|u - u_{\text{nom}}\|_2^2 \qquad (15) \\ &\text{s.t. } \nabla h_{\text{lcb}}^t(\mathbf{x}_t)^\top\big(\hat{f}_t(\mathbf{x}_t) + G(\mathbf{x}_t)u\big) - \beta_t^{1/2}\big\|(\Sigma_g^t(\mathbf{x}_t))^{1/2}\nabla h_{\text{lcb}}^t(\mathbf{x}_t)\big\|_2 \\ &\qquad\qquad + \alpha\big(h_{\text{lcb}}^t(\mathbf{x}_t)\big)\ \ge 0, \end{aligned}$$

which enforces the GP–robust CBF constraint as a hard inequality [12]. The controller and CBF-QP treat $\{x : h_{\text{lcb}}^t(x) \ge 0\}$ as the certified safe set, so that data collected online both refines the geometry of the learned barrier and shrinks its epistemic uncertainty; over time, this yields progressively less conservative yet probabilistically safe behavior without requiring a hand-designed analytic CBF for the task, which can be quite difficult in practice [17].

E. Safe Bayesian Optimization

The framework extends to optimizing controller parameters $\boldsymbol{\theta} \in \Theta \subset \mathbb{R}^p$ while maintaining safety, as proposed in [18]. Let $s : \Theta \to \mathbb{R}$ denote safety (minimum h_{lcb}^t over a trajectory) and $r : \Theta \to \mathbb{R}$ denote reward. The goal is

$$\max_{\boldsymbol{\theta}\in\Theta}\ r(\boldsymbol{\theta}) \quad \text{subject to} \quad s(\boldsymbol{\theta}) \ge 0. \tag{16}$$

We place independent GP priors $s \sim \mathcal{GP}(0, k_s)$ and $r \sim \mathcal{GP}(0, k_r)$. After n evaluations, the posteriors yield μ_s^n, σ_s^n and μ_r^n, σ_r^n. Define the safe parameter set

$$\Theta_n^{\text{safe}} = \big\{\boldsymbol{\theta} \in \Theta : \mu_s^n(\boldsymbol{\theta}) - \beta_s^{1/2}\sigma_s^n(\boldsymbol{\theta}) \ge 0\big\}. \tag{17}$$

At each iteration, we select parameters via one of two strategies:

Expansion: Maximize safety uncertainty near the boundary of Θ_n^{safe}:

$$\boldsymbol{\theta}_{\text{exp}} = \arg\max_{\boldsymbol{\theta}\in\Theta}\ \sigma_s^n(\boldsymbol{\theta}) \cdot \mathbf{1}\big[\mu_s^n(\boldsymbol{\theta}) - \beta_s^{1/2}\sigma_s^n(\boldsymbol{\theta}) \ge -\epsilon\big]. \tag{18}$$

Exploitation: Maximize reward upper confidence bound within Θ_n^{safe}:

$$\boldsymbol{\theta}_{\text{opt}} = \arg\max_{\boldsymbol{\theta}\in\Theta_n^{\text{safe}}}\ \mu_r^n(\boldsymbol{\theta}) + \beta_r^{1/2}\sigma_r^n(\boldsymbol{\theta}). \tag{19}$$

Algorithm 1 Dual GP Safe Set Learning

```
Require: Nominal model $(f_0, G)$, base controller $\pi_\theta$, GPs for residual $g$ and barrier $h$, initial safe set $\hat{\mathcal{S}}_0$
for episode $k = 1, 2, \dots$ do
    Sample $x_0 \in \hat{\mathcal{S}}_{k-1}$, set trajectory $\tau \leftarrow \{x_0\}$
    for $t = 0, \dots, T-1$ do
        Observe $x_t$ and compute nominal control $u_{\text{nom}} \leftarrow \pi_\theta(x_t)$
        Query dynamics GP at $x_t$ to obtain $\hat{f}_t(x_t)$ and $\Sigma_g^t(x_t)$
        Query barrier GP at $x_t$ to obtain $\mu_h^t(x_t)$, $\sigma_h^t(x_t)$, and $\nabla h_t$
        Compute $h_{\text{lcb}}^t(x_t)$ and robust CBF constraint using (12), (16)
        Solve CBF-QP to obtain safe control $u_t$
        Apply $u_t$, observe $x_{t+1}$, append to $\tau$
        Form residual measurement from (8) and update dynamics GP
    end for
    if no constraint violation observed in $\tau$ then
        For a subsample of $\tau$, construct labels $y_h(x)$ using (25)
        Update barrier GP with $(x, y_h(x))$
    end if
    Update certified safe set $\hat{\mathcal{S}}_k$ from the LCB safe set (13)
end for
```

The algorithm alternates with probability ρ_{exp} for expansion, ensuring monotonic growth of Θ_n^{safe}.

IV. Simulation Study and Results

A. Discrete Gridworld

We first validate our approach on a discrete gridworld environment that captures the essential structure of the safe set learning problem. The gridworld is composed of 15 by 15 cells with two types of hazards: circular obstacles that cause immediate failure upon contact, and slip zones where unknown probabilistic dynamics affects the agent's cell transitions. The agent observes only its current cell and local surrounding within a 1 cell radius. The locations and extents of the hazards are unknown apriori.

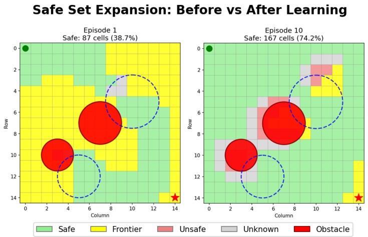

Fig. 2: Safe-set expansion across 10 episodes in the discrete gridworld.

1) Discrete CBF Formulation: In the discrete gridworld setting, the CBF constraint simplifies to action enumeration. For each candidate action $a \in \{\text{UP}, \text{DOWN}, \text{LEFT}, \text{RIGHT}, \text{STAY}\}$, we compute the expected next-state safety value accounting for slip uncertainty:

$$\bar{h}(s') = \mu_h(s') - \beta^{1/2}\sigma_h(s') - \gamma \cdot \hat{p}_{\text{slip}}(s) \cdot \Delta_{\text{slip}},$$

Algorithm 2 Safe Bayesian Optimization of Controller Parameters

```
Require: Parameter domain $\Theta$, expansion probability $\rho_{\text{exp}}$, tolerance $\epsilon > 0$
Initialize safety GP for $s(\theta)$ and reward GP for $r(\theta)$
Choose an initial safe $\theta_0$, evaluate with Algorithm 1, obtain $(s_0, r_0)$, update GPs
for iteration $n = 1, 2, \dots$ do
    Form safe set $\Theta_{n-1}^{\text{safe}}$ using the safety LCB (17)
    if rand() $< \rho_{\text{exp}}$ then                          ▷ Expansion step
        Select $\theta_n$ by maximizing safety uncertainty near the boundary using (18)
    else                                                          ▷ Exploitation step
        Select $\theta_n$ by maximizing reward UCB over $\Theta_{n-1}^{\text{safe}}$ using (19)
    end if
    Run Algorithm 1 with controller parameterized by $\theta_n$
    Compute safety summary $s_n$ (e.g., $\min_t h_{\text{lcb}}^t(x_t)$) and task reward $r_n$
    Update safety and reward GPs with $(\theta_n, s_n)$ and $(\theta_n, r_n)$
end for
return $\theta^\star \leftarrow \arg\max_{\theta \in \Theta_n^{\text{safe}}} \mu_r^n(\theta)$
```

2) Gridworld Results: We run 10 learning episodes on a grid with two circular obstacles and two slip zones. The agent starts from a bootstrapped safe region of radius 3 cells around the origin. Figure 2 shows the monotonic expansion of the certified safe set. Episode 1 coverage of 38.7% (87 cells) grows to 74.2% (167 cells) by episode 10, with the expansion rate decreasing as the frontier approaches true obstacle boundaries. We record 5 total collisions over 10 episodes, with 4 occurring in episodes 1-5 and only 1 in episodes 6-10. After episode 6, the learned safety GP provided sufficient coverage to prevent further violations.

B. Continuous

We extend our framework on a 1D position continuous control task intentional model mismatch in both dynamics and safety. A point mass moves within bounds $|p| \leq p_{\max}$ under wind disturbance, with state $\mathbf{x} = [p, v]^\top$ and discrete-time dynamics

$$p_{t+1} = p_t + \Delta t\, v_t, \quad v_{t+1} = v_t + \Delta t\big(u_t + d(p_t, v_t, t)\big), \tag{21}$$

where $d(\cdot)$ is an unknown disturbance. We introduce two sources of mismatch that mimic common modeling errors. First, a *dynamics mismatch*: the true disturbance combines position-dependent wind, velocity drag, time-varying gusts, and directional bias

$$d(p, v, t) = \underbrace{c_w \operatorname{sgn}(p) \frac{p^2}{1 + p^2}}_{\text{position push}} - \underbrace{c_d\, v|v|}_{\text{drag}} + \underbrace{A_g \sin(\omega_g t)}_{\text{gust}} + d_b, \tag{22}$$

while the analytic baseline assumes only a bounded disturbance $|d| \leq \bar{d}$, underestimating the true magnitude near the boundary. This wind model captures how constraints can

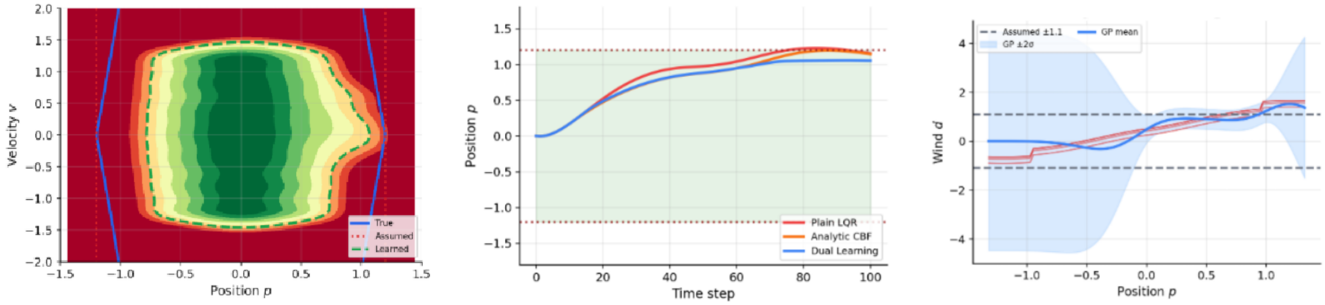

Fig. 1: Safe-set and dynamics learning under model mismatch. *Left:* true (blue), assumed analytic (red), and learned GP (green) safe-set boundaries in (p, v). *Middle:* position trajectories for plain LQR, analytic CBF, and dual learning under aggressive tracking. *Right:* GP model of the wind disturbance, showing mean and confidence bands versus position.

exhibit complex nonlinear behavior that is difficult to encode analytically without online adaptation. Second, a *barrier mismatch*: the true safe set has velocity–position coupling

$$h^*(\mathbf{x}) = p_{\max}^2 - p^2 - \lambda_h |v||p|, \tag{23}$$

whereas the analytic baseline uses $h_0(\mathbf{x}) = p_{\max}^2 - p^2$, naively ignoring velocity effects.

Task. An LQR controller tracks a reference p_{ref} near the boundary with weights $Q = \text{diag}(q_p, q_v)$. Larger q_p produces more aggressive tracking but increases the risk of constraint violation.

Observable Labels. A key challenge is learning the barrier without oracle access to $h^*(\mathbf{x})$. Our framework focuses on learning from *safe trajectories*: the conservative CBF–QP produces rollouts that, in our simulations, remain within $\mathcal{S}^*$, so true violations are rare and most data come from near–safe behavior. We therefore construct labels from observables along surviving trajectories,

$$y_h(\mathbf{x}) = \frac{p_{\max} - |p|}{p_{\max}} - \lambda_y \frac{|v|}{v_{\max}} \frac{|p|}{p_{\max}}, \tag{24}$$

which heuristically encode that safety margin decreases with (i) proximity to the wall and (ii) high velocity near the boundary. These labels depend only on measured states (p, v) and do not require access to h^* or latent parameters. After each episode, we update the barrier GP with subsampled trajectory labels, progressively refining $\hat{\mathcal{S}}_t$ while preserving its conservative, inner-approximation character.

Methods. We compare: ***Plain LQR***: no safety filter (performance upper bound); ***Analytic CBF***: mismatched barrier h_0 and assumed bound $|d| \leq \bar{d}$; ***Dual Learning***: learns dynamics and barrier via Algorithm 1.

Training uses two phases: conservative exploration (low q_p, high CBF gain α) followed by moderate exploration, accumulating approximately 400 barrier samples and 2500 dynamics samples over 25 episodes.

Results. Figure 1 (left) compares the true, assumed, and learned safe sets. The position-only analytic barrier substantially overestimates safety at high speeds, failing to contract near the wall. In contrast, the observation-based barrier GP learns the velocity-dependent tightening and yields a conservative inner approximation of $\mathcal{S}^*$. Figure 1 (middle) shows closed-loop trajectories under aggressive LQR weights $(q_p, q_v) = (5.0, 1.0)$: plain LQR drives the state close to or beyond the boundary, the analytic CBF occasionally prevents violations but admits near-misses, while the dual learning controller maintains a larger safety margin. Figure 1 (right) illustrates dynamics learning: with relatively few samples, the GP mean tracks the true wind field and the uncertainty inflates in poorly explored regions, inducing more conservative CBF constraints there. Table I summarizes safety over 8 independent episodes. An episode is counted as safe if $\min_t h^*(x_t) \geq 0$.

The analytic CBF underestimates disturbance near boundaries where wind bias compounds, and permits high-velocity approaches that violate the true barrier (23). Our method learns both effects from data.

TABLE I: Safety comparison

Method	Safe rate
Plain LQR	0/8 (0%)
Analytic CBF	5/8 (62%)
Dual Learning	**8/8 (100%)**

V. Discussion and Conclusion

By jointly modeling dynamics residuals and a latent control barrier function, we successfully constructed a GP–robust CBF constraint that can be enforced via a convex quadratic program, and coupled it with a SafeOpt-style Bayesian optimizer for controller tuning.

Our experiments exposed several practical challenges. First, the barrier labels are hand-designed heuristics built from observable quantities. Second, the GP machinery introduces computational overhead that grows with the number of data points. Third, we did not develop a safety guarantee for the coupled dynamics–barrier learning loop.

Finally, we made progress toward a vision-inertial (VIO) quadrotor model, though we could not fully address the additional complications within the project timeframe. The quadrotor introduces higher-dimensional state and control spaces, attitude-position coupling, and perception-driven estimates with their own failure modes. These factors complicate barrier label design, GP state representation, and CBF-QP feasibility under aggressive maneuvers. Future work includes (i) completing VIO quadrotor integration in simulation to learn safe flight-envelopes; (ii) developing GP approximations and kernels tailored to underactuated flight; and (iii) closing the loop between safe BO and GP-CBF filtering for end-to-end tuning.

References

[1] L. Brunke, M. Greeff, A. W. Hall, Z. Yuan, S. Zhou, J. Panerati, and A. P. Schoellig, "Safe learning in robotics: From learning-based control to safe reinforcement learning," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 5, no. 1, pp. 411–444, 2022.

[2] P. Liu, K. Zhang, D. Tateo, S. Jauhri, Z. Hu, J. Peters, and G. Chalvatzaki, "Safe reinforcement learning of dynamic high-dimensional robotic tasks: navigation, manipulation, interaction," *arXiv preprint arXiv:2209.13308*, 2022.

[3] S. Sun and C. C. de Visser, "Quadrotor safe flight envelope prediction in the high-speed regime: A monte-carlo approach," in *AIAA Scitech 2019 Forum*, 2019, p. 0948.

[4] P. Jagtap, G. J. Pappas, and M. Zamani, "Control barrier functions for unknown nonlinear systems using gaussian processes," in *2020 59th IEEE Conference on Decision and Control (CDC)*. IEEE, 2020, pp. 3699–3704.

[5] F. Berkenkamp, M. Turchetta, A. Schoellig, and A. Krause, "Safe model-based reinforcement learning with stability guarantees," *Advances in neural information processing systems*, vol. 30, 2017.

[6] F. Castaneda, J. J. Choi, W. Jung, B. Zhang, C. J. Tomlin, and K. Sreenath, "Recursively feasible probabilistic safe online learning with control barrier functions," *arXiv preprint arXiv:2208.10733*, 2022.

[7] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, "Control barrier functions: Theory and applications," 2019. [Online]. Available: https://arxiv.org/abs/1903.11199

[8] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs for safety critical systems," *IEEE Transactions on Automatic Control*, vol. 62, no. 8, pp. 3861–3876, 2016.

[9] A. Taylor, A. Singletary, Y. Yue, and A. Ames, "Learning for safety-critical control with control barrier functions," in *Learning for dynamics and control*. PMLR, 2020, pp. 708–717.

[10] C. K. Williams and C. E. Rasmussen, *Gaussian processes for machine learning*. MIT press Cambridge, MA, 2006, vol. 2, no. 3.

[11] N. Srinivas, A. Krause, S. M. Kakade, and M. Seeger, "Gaussian process optimization in the bandit setting: No regret and experimental design," *arXiv preprint arXiv:0912.3995*, 2009.

[12] M. A. Khan, T. Ibuki, and A. Chatterjee, "Gaussian control barrier functions: Non-parametric paradigm to safety," *IEEE Access*, vol. 10, pp. 99 823–99 836, 2022.

[13] M. Aali and J. Liu, "Learning high-order control barrier functions for safety-critical control with gaussian processes," in *2024 American Control Conference (ACC)*. IEEE, 2024, pp. 1–6.

[14] S. Liu, A. Kumar, J. Fisac, R. Adams, and P. Ramadge, "Probf: Learning probabilistic safety certificates with barrier functions," *arXiv preprint arXiv:2112.12210*, 2021.

[15] F. Zhu, T. Pati, and S. Z. Yong, "Learning safe data-driven control barrier functions for unknown continuous systems," *IEEE Control Systems Letters*, 2025.

[16] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[17] L. Lindemann, H. Hu, A. Robey, H. Zhang, D. Dimarogonas, S. Tu, and N. Matni, "Learning hybrid control barrier functions from data," in *Conference on robot learning*. PMLR, 2021, pp. 1351–1370.

[18] F. Berkenkamp, A. P. Schoellig, and A. Krause, "Safe controller optimization for quadrotors with gaussian processes," in *2016 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2016, pp. 491–496.