Confidence-Adaptive Lipschitz-Managed Control Barrier Functions for Risk-Aware Control

by

Pieter-Jan van Dolderen

MSc Thesis to be defended on

04/12/2025

Supervisor & Chair committee:	C. Pek
Examination committee:	L. Ferranti
Faculty:	Mechanical Engineering
Department:	Cognitive Robotics
Programme:	MSc. Robotics
Project Duration:	December 2024 - December 2025

TU Delft

Confidence-Adaptive Lipschitz-Managed Control Barrier Functions for Risk-Aware Control

Pieter-Jan van Dolderen

***Abstract*—Ensuring safety for dynamic systems in the presence of state estimation errors is challenging because safety guarantees must hold for the unknown true state while only an uncertain estimate is available. Control Barrier Function (CBF) based safety filters can become overly conservative under state estimation uncertainty, leading to deadlocks in narrow passages even when a safe path exists. We present Confidence-Adaptive Lipschitz-Managed Control Barrier Functions (CALM–CBFs) to reduce this conservatism, guaranteeing safety while improving the robot's overall performance. CALM–CBF adapts the safety margin to both local estimation confidence and how sensitive the safety constraint is in the current part of the state space, rather than using a single worst-case margin everywhere. A risk-aware supervisor then trades off allowed speed against conservatism, relaxing the margin when progress stalls near tight gaps and restoring stricter behavior when clearance improves. In simulation, our method reduces the conservatism up to 68.5% compared to a measurement-robust CBF baseline, while still avoiding collisions and resolving deadlocks in narrow passages.**

I. Introduction

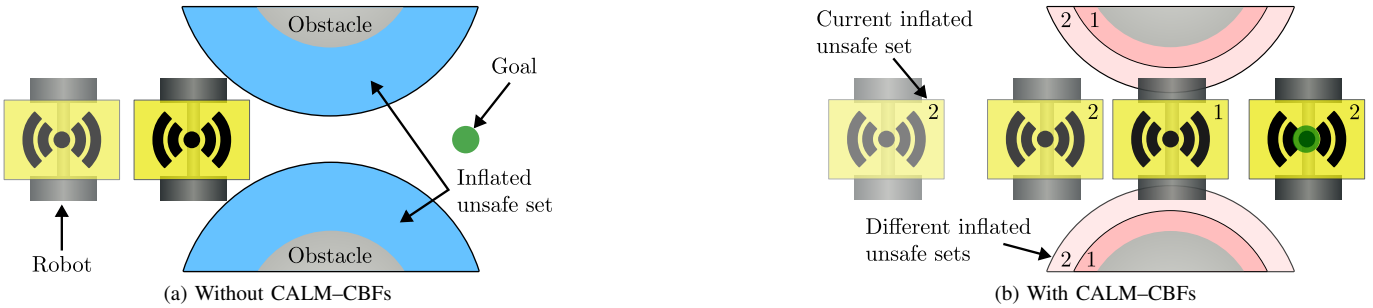
As autonomous systems move from structured labs to crowded, dynamic environments, controllers must guarantee that system trajectories remain within defined safe sets while still achieving their task objectives. A promising architectural pattern is the safety filter: a lightweight, real-time module that minimally adjusts a nominal controller's proposed input only when a safety constraint is at risk of violation, thereby preserving performance whenever possible [1]. Within this paradigm, Control Barrier Functions (CBFs) stand out because they provide a mathematical, optimization-friendly route to guarantee forward invariance: the filter enforces an inequality (often via a quadratic program) that keeps the state inside a prescribed safe set while deviating as little as needed from the nominal action [2]. These safety guarantees and real-time implementability make CBFs a promising mechanism for robot safety.

In theory, CBF-based safety guarantees are derived under the ideal assumption that the controller has access to the true state $\mathbf{x}$. On real robotic systems, however, we only have noisy, delayed, and partial measurements, from which an estimator reconstructs a state estimate $\widehat{\mathbf{x}}$. This means that, in deployment, the safety constraints are evaluated on $\widehat{\mathbf{x}}$ instead of $\mathbf{x}$. If we replace $\mathbf{x}$ by $\widehat{\mathbf{x}}$ in the CBF condition, the original invariance proof no longer holds, because the estimation error $\epsilon = \mathbf{x} - \widehat{\mathbf{x}}$ is ignored. A common way to recover guarantees is to assume that this error is bounded by $\epsilon_{\max}$, i.e., $\|\mathbf{x} - \widehat{\mathbf{x}}\| \leq \epsilon_{\max}$, and to inflate the obstacles or shrink the safe set by an uncertainty margin that covers the maximum error. We call this the robust or worst-case approach. In CBF implementations, this appears as an additional safety term in the constraint, chosen such that if the inequality holds for $\widehat{\mathbf{x}}$ with this margin, then the true state remains safe. The downside is visible in Figure 1a: when the assumed error bounds are large or conservative, the inflated obstacles block narrow passages, so the safety filter intervenes too aggressively and the robot gets stuck even though a collision-free path exists.

Instead of inflating all obstacles by the same worst-case, i.e., maximum, amount (Figure 1a), our method lets the inflation depend on where the robot is and how confident the estimator is in that region. As a result, the inflated unsafe set can shrink, opening the narrow passage in Figure 1b while still keeping the true state safe.

Concretely, this leads to the following contributions:

1) *Confidence-Adaptive Lipschitz-Managed CBFs (CALM–CBFs).* We extend Measurement-Robust CBFs [3] with two ingredients that make the margin adaptive: (i) a



(a) Without CALM–CBFs (b) With CALM–CBFs

Fig. 1: (a) A fixed maximum inflation overestimates risk, blocks the passage, and the robot cannot complete the task. (b) The robot can change the inflation to safely pass through the gap and reach the goal.

state-dependent error bound provided by a perception module, so that well-observed areas are inflated less than poorly observed ones, and (ii) precomputed, local sensitivity bounds (Lipschitz envelopes) on a grid, which avoid using a single pessimistic constant everywhere. On top of this, a lightweight risk-aware supervisor adjusts a tunable risk level together with a speed limit: higher risk levels correspond to a conservative, worst-case margin and higher speeds in open space, while lower risk levels reject extreme outliers, reduce the margin in tight gaps, and simultaneously slow the robot down. Together, these components produce locally adaptive safety margins with controlled risk, as illustrated in Figure 1b.

2) *Experimental validation.* We evaluate CALM–CBFs in simulation on two environments (a narrow Gap and a Cluttered world) with identical dynamics, planner, and nominal controller. The results show that our method consistently decreases the conservatism compared to the Measurement-Robust CBF baseline.

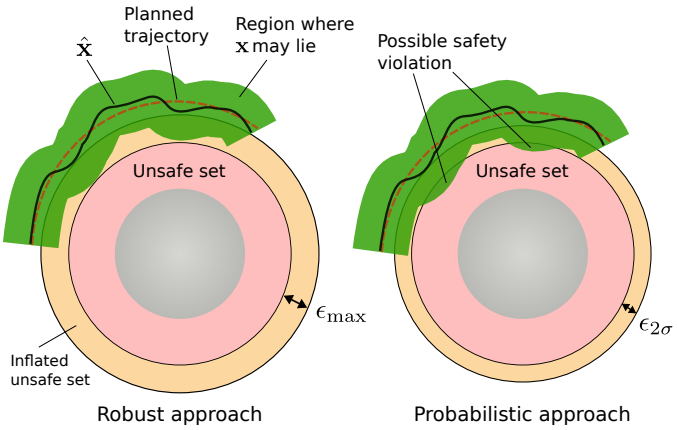


Fig. 2: Left: a robust CBF inflates the unsafe set by a worst-case margin $\epsilon = \epsilon_{\max}$. Right: a probabilistic CBF uses a probabilistic margin, e.g., $\epsilon = 2\sigma$. The region labeled safe is larger than in the robust case, at the cost of a small probability of violation where the confidence tube intersects the unsafe set.

II. Related Work

There are multiple approaches for CBFs with state–estimation uncertainty. One common line of work is the worst–case or robust approach [3–15], which augments the CBF condition with an explicit uncertainty margin based on the maximum estimation error $\epsilon_{\max}$. Intuitively, the safety is achieved by inflating the unsafe set with a worst-case margin, see Figure 2 (left). Measurement-robust CBFs (MR–CBFs) make this concrete by adding extra terms to the CBF inequality that scale with Lipschitz bounds on the CBF and the size of the error set [3, 5]. In the QP, this appears as an offset that tightens the constraint, or equivalently, as inflating the obstacles. The same idea is extended to higher-relative-degree systems via High-Order Measurement-Robust CBFs [12]. The strength of worst-case approaches is their strict safety guarantee: as long as the estimation error stays within the assumed bound, the true state will remain safe. However, they typically rely on global worst-case error bounds, so all regions of the state space are treated with the same conservatism, which can drastically shrink the usable safe set and induce conservative behavior.

In contrast to the worst-case approach, probabilistic approaches handle state estimation uncertainty by requiring safety with high probability, rather than absolute certainty [16–20]. These methods assume a statistical model of the estimation error and then impose CBF conditions that must hold except for a small risk δ of violation. Intuitively, this possible violation makes it possible for the CBF to relax the inflated unsafe region, see Figure 2 (right). Probabilistic approaches are less conservative, but the drawback is that they do not have the strict safety guarantees, and they rely on accurate uncertainty models or assumed error distributions.

Existing methods tend to choose between strict robustness with uniform conservatism or reduced conservatism at the price of probabilistic safety and stronger modeling assumptions. In this work, we aim to combine desirable aspects of both. We retain the measurement-robust CBF structure and its worst-case guarantee as a reference point, but make the uncertainty margin state-dependent and introduce a tunable risk level that can locally relax the margin (and simultaneously slow the robot down) when progress is blocked, while recovering conservative behavior when required.

III. Preliminaries

We first review safety-critical control via Control Barrier Functions (CBFs) following [2]. Then, we extend CBFs with state estimation uncertainty via the concept of MR–CBFs [3].

A. Control Barrier Functions

Consider a continuous-time, control-affine nonlinear system of the form:

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

where $\mathbf{x} \in \mathbb{R}^n$ is the state, $\mathbf{u} \in \mathbb{R}^m$ is the control input, and the vector fields $f : \mathbb{R}^n \to \mathbb{R}^n$ and $g : \mathbb{R}^n \to \mathbb{R}^{n\times m}$ are locally Lipschitz continuous.

To formalize safety, we define a set $\mathcal{C}$ (the safe set) as the superlevel set of a continuously differentiable function $h : \mathbb{R}^n \to \mathbb{R}$:

$$\mathcal{C} \triangleq \{\mathbf{x} \in \mathbb{R}^n \mid h(\mathbf{x}) \geq 0\}. \tag{2}$$

Intuitively, $h(\mathbf{x}) \geq 0$ defines the safe region and $h(\mathbf{x}) < 0$ the unsafe region. A set $\mathcal{C}$ is forward invariant for system (1) if, whenever $\mathbf{x}(0) \in \mathcal{C}$, all subsequent states satisfy $\mathbf{x}(t) \in \mathcal{C}$ for all $t \geq 0$. In other words, if the system starts safely, it remains safe for all time. The system is safe with respect to $\mathcal{C}$ if $\mathcal{C}$ is forward invariant. Ensuring forward invariance of $\mathcal{C}$ is the central goal of safety-critical control.

Control Barrier Functions (CBFs) are a way to guarantee forward invariance of $\mathcal{C}$ for the closed-loop system. We first introduce an extended class-$\mathcal{K}$ function α which is a continuous function $\alpha : \mathbb{R} \to \mathbb{R}$ that is strictly increasing

and satisfies $\alpha(0) = 0$. Typically, α is chosen as a linear function $\alpha(s) = \gamma s$ where $\gamma > 0$. $\mathcal{C}$ is forward invariant if there exists an extended class-$\mathcal{K}$ function α such that for all $\mathbf{x}$ in the domain $\mathcal{C}$,

$$\sup_{\mathbf{u}\in\mathbb{R}^m} [L_f h(\mathbf{x}) + L_g h(\mathbf{x})u] \geq -\alpha(h(\mathbf{x})), \tag{3}$$

where $L_f h$ and $L_g h$ denote the Lie derivatives of h along f and g, respectively.

Given a nominal controller $\mathbf{u}_{\text{nom}} = k(\mathbf{x})$, the following Quadratic Program (QP) finds the minimum perturbation on $\mathbf{u}_{\text{nom}}$ to guarantee safety:

$$\mathbf{u}_{\text{CBF}} = \arg\min_{\mathbf{u}\in\mathbb{R}^m} \frac{1}{2}\|\mathbf{u} - \mathbf{u}_{\text{nom}}\|^2 \qquad \text{(CBF-QP)}$$
$$\text{s.t.} \quad L_f h(\mathbf{x}) + L_g h(\mathbf{x})\mathbf{u} \geq -\alpha(h(\mathbf{x})).$$

B. Measurement-Robust Control Barrier Functions

The CBF formulation in III-A assumes perfect knowledge of the state, which is unrealistic in real-world applications. The MR–CBF formulation is a general way to let the CBF deal with state estimation uncertainty. It does not assume a particular observer structure; instead, it assumes that the measurement error $\epsilon(\mathbf{x})$ comes from a known measurement-dependent, compact pointwise set $\mathcal{E}(\mathbf{y})$, where $\mathbf{y} = p(\mathbf{x})$, $p(\mathbf{x}) : \mathbb{R}^n \to \mathbb{R}^k$, is the state-dependent sensor measurement. The error set can be conservatively characterized via the function ϵ: $\mathbb{R}^k \to \mathbb{R}_{\geq 0}$ defined as:

$$\epsilon(\mathbf{y}) \triangleq \max_{\mathbf{e}\in\mathcal{E}(\mathbf{y})} \|\mathbf{e}\|_2. \tag{4}$$

Based on the measurements, an estimate of the state $\widehat{\mathbf{x}} \triangleq \widehat{\mathbf{q}}(\mathbf{y}) = \mathbf{x} + \mathbf{e}(\mathbf{x})$ can be constructed. This results in the following closed-loop system:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{k}(\mathbf{y}, \widehat{\mathbf{x}}). \tag{5}$$

This altogether leads to the main theorem of [3]:

Theorem 1: Given a safe set $\mathcal{C} \subset \mathbb{R}^n$, assume that $L_{\mathbf{f}}h$, $L_{\mathbf{g}}h$, and $\alpha \circ h$ are Lipschitz continuous on $\mathcal{C}$ with Lipschitz constants $\mathfrak{L}_{L_{\mathbf{f}}h}, \mathfrak{L}_{L_{\mathbf{g}}h}$, and $\mathfrak{L}_{\alpha\circ h} \in \mathbb{R}_{\geq 0}$, respectively. Define the function $\epsilon : \mathbb{R}^k \to \mathbb{R}_{\geq 0}$ as in (4), and define the functions $a, b : \mathbb{R}^k \to \mathbb{R}_{\geq 0}$ as $a(\mathbf{y}) = (\mathfrak{L}_{L_{\mathbf{f}}h} + \mathfrak{L}_{\alpha\circ h})\epsilon(\mathbf{y})$ and $b(\mathbf{y}) = \mathfrak{L}_{L_{\mathbf{g}}h}\epsilon(\mathbf{y})$. If $\mathbf{k} : \mathbb{R}^k \times \mathbb{R}^n \to \mathbb{R}^m$ is a Lipschitz continuous controller satisfying:

$$L_{\mathbf{f}}h(\widehat{\mathbf{x}}) + L_{\mathbf{g}}h(\widehat{\mathbf{x}})\mathbf{k}(\mathbf{y}, \widehat{\mathbf{x}}) - (a(\mathbf{y}) + b(\mathbf{y})\|\mathbf{k}(\mathbf{y}, \widehat{\mathbf{x}})\|_2) \geq -\alpha(h(\widehat{\mathbf{x}})) \tag{6}$$

for all $\mathbf{x} \in \mathcal{C}$, with $\mathbf{y} = \mathbf{p}(\mathbf{x})$ and $\widehat{\mathbf{x}} = \widehat{\mathbf{q}}(\mathbf{y})$, then the system (5) is safe with respect to $\mathcal{C}$.

Intuitively, this theorem says the following. Because we only know $\widehat{\mathbf{x}}$, the true state $\mathbf{x}$ may lie anywhere inside an error ball around $\widehat{\mathbf{x}}$, whose radius is given by $\epsilon(\mathbf{y})$. The terms $a(\mathbf{y})$ and $b(\mathbf{y})$ are Lipschitz-based buffers: they upper-bound how much the barrier function h and its derivatives can change when $\mathbf{x}$ moves within that error ball. By subtracting $a(\mathbf{y}) + b(\mathbf{y})\|k(\mathbf{y}, \hat{\mathbf{x}})\|_2$ in (6), we tighten the CBF constraint on the estimated state so that, for every possible true state in the error ball, the original CBF inequality still holds. Geometrically, this is equivalent to inflating the unsafe set by a measurement-dependent margin; any trajectory that satisfies the tightened inequality at $\widehat{\mathbf{x}}$ is then guaranteed to keep the true state $\mathbf{x}$ inside the safe set $\mathcal{C}$. As long as we can bound the worst-case estimation error for each measurement, the MR–CBF construction yields a safety guarantee that is independent of how the estimate $\widehat{\mathbf{x}}$ is obtained. In the remainder of the paper, we will refer to the quantity

$$M(\mathbf{y}, \widehat{\mathbf{x}}) \triangleq a(\mathbf{y}) + b(\mathbf{y}) \left\|\mathbf{k}(\mathbf{y}, \widehat{\mathbf{x}})\right\|_2, \tag{7}$$

as the (MR–CBF) safety margin, and focus on how to construct this margin less conservatively. When unambiguous, we write $M(\widehat{\mathbf{x}})$, understanding that a and b are evaluated at the associated estimated state $\widehat{\mathbf{x}}$.

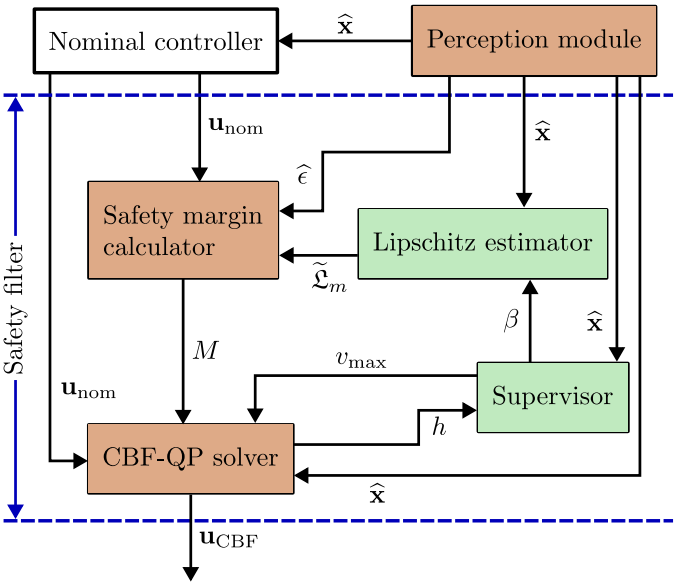
IV. Problem Statement

We study a mobile robot that must reach a goal position in an environment with static obstacles and state–estimation uncertainty. Our primary safety objective is to ensure that, under the MR–CBF assumptions on the estimation error, the true state remains inside a prescribed safe set $\mathcal{C}$ for all time (forward invariance). In addition, we care about task performance: the robot should still make progress toward its goal and avoid artificial deadlocks that arise purely from conservative safety margins, for example, in narrow openings that are in principle traversable. Our goal is therefore to design a safety filter that preserves the MR–CBF forward–invariance guarantee while constructing the uncertainty margin less conservatively.

Running example: We focus on a 2D holonomic (omni-wheel) mobile robot that can command velocities directly in the x and y directions (relative degree 1). The robot transports packages through a factory hall. In this running example, we zoom in on the robot passing a narrow passage, see Figure 1.

V. Confidence-Adaptive Lipschitz-Managed Control Barrier Functions

This section develops our CALM–CBF safety filter. Before going into the details, we first give a high-level overview of the architecture in Fig. 3. Starting from a nominal controller and a perception module that provides an estimated state $\widehat{\mathbf{x}}$ and a measure of estimation error $\widehat{\epsilon}$, we construct a safety margin M that adapts to where the robot is and how well it is observed. The perception module supplies a state-dependent error bound $\widehat{\epsilon}$, which replaces the single global error bound $\epsilon_{\max}$ used in classical MR–CBFs. A grid-based Lipschitz estimator precomputes the sensitivity of the barrier function across different parts of the state space. On top of this, a simple risk-aware supervisor switches between a small number of risk levels: conservative, higher levels use larger margins and allow higher speeds in open space, while permissive, lower levels reduce the margin and simultaneously cap the speed in tight areas. All of these components feed into the same MR–CBF equation (6).



$\widehat{\mathbf{x}}$	estimated state
$\mathbf{u}_{\text{nom}}$	nominal control input
$\mathbf{u}_{\text{CBF}}$	safety-filtered control input
$\widetilde{\mathfrak{L}}_m$	Lipschitz constants for $m \in \{L_f h,\ L_g h,\ \alpha \circ h\}$
M	safety margin accounting for state-estimation error
h	safety function value
$v_{\max}$	maximum allowed velocity
$\widehat{\epsilon}$	perception-driven error bound

Fig. 3: High-level overview of the CALM–CBF pipeline. Orange blocks indicate components that are modified with respect to a standard MR–CBF pipeline; green blocks are newly introduced in CALM–CBF.

We assume that the perception module is provided by the user. A perception module provides an estimated state $\widehat{\mathbf{x}}$ and the estimated error of the estimated state $\widehat{\epsilon}(\widehat{\mathbf{x}})$.

A. Redefining the Error Function

Classical MR–CBF formulations use a global error bound $\epsilon_{\max}$, which is simple but conservative. In our setting, a perception module provides a state estimate and a state–dependent error bound. We keep the construction of this bound as an implementation detail and only impose abstract properties that are sufficient to retain the MR–CBF safety guarantee. For simplicity, we define $\widehat{\epsilon}(\widehat{\mathbf{x}})$ instead of $\widehat{\epsilon}(\mathbf{y})$, this is not a problem because $\widehat{\mathbf{x}} = \widehat{\mathbf{q}}(\mathbf{y})$. We first need to define the admissible perception module:

Definition 1 (Admissible Perception Module): A perception/estimation module is *admissible* if it produces $(\widehat{\mathbf{x}}, \widehat{\epsilon}(\widehat{\mathbf{x}}))$ satisfying:

P1. Over-approximation: $\|\mathbf{x} - \widehat{\mathbf{x}}\| \le \widehat{\epsilon}(\widehat{\mathbf{x}})$.
P2. Boundedness: $0 \le \widehat{\epsilon}(\widehat{\mathbf{x}}) \le \epsilon_{\max}$.
P3. Regularity: $\widehat{\epsilon}$ is locally Lipschitz.

The next proposition shows that these conditions are sufficient to retain the MR–CBF safety guarantee.

Proposition 1 (Safety under admissible perception): Suppose the perception module is admissible in the sense of Definition 1. Then the MR–CBF inequality (6), with $a(\cdot)$ and $b(\cdot)$ constructed from $\widehat{\epsilon}(\widehat{\mathbf{x}})$, guarantees safety with respect to $\mathcal{C}$ for the closed-loop system (5).

Proof sketch. Let $\mathcal{B}(\widehat{\mathbf{x}}) = \{\mathbf{x} : \|\mathbf{x} - \widehat{\mathbf{x}}\| \le \widehat{\epsilon}(\widehat{\mathbf{x}})\}$. By property P1, the true state lies in $\mathcal{B}(\widehat{\mathbf{x}})$. For any $\mathbf{x} \in \mathcal{B}(\widehat{\mathbf{x}})$, local Lipschitz bounds give

$$\begin{aligned} L_{\mathbf{f}} h(\mathbf{x}) &\ge L_{\mathbf{f}} h(\widehat{\mathbf{x}}) - \mathfrak{L}_{L_{\mathbf{f}} h}\, \widehat{\epsilon}(\widehat{\mathbf{x}}), \\ L_{\mathbf{g}} h(\mathbf{x})\mathbf{u} &\ge L_{\mathbf{g}} h(\widehat{\mathbf{x}})\mathbf{u} - \mathfrak{L}_{L_{\mathbf{g}} h}\widehat{\epsilon}(\widehat{\mathbf{x}})\|\mathbf{u}\|_2, \\ \alpha(h(\mathbf{x})) &\le \alpha(h(\widehat{\mathbf{x}})) + \mathfrak{L}_{\alpha\circ h}\widehat{\epsilon}(\widehat{\mathbf{x}}). \end{aligned} \tag{8}$$

Substituting these into the nominal CBF inequality yields exactly the MR–CBF constraint with parameters

$$\begin{aligned} a(\widehat{\mathbf{x}}) &= (\mathfrak{L}_{L_{\mathbf{f}} h} + \mathfrak{L}_{\alpha\circ h})\widehat{\epsilon}(\widehat{\mathbf{x}}), \\ b(\widehat{\mathbf{x}}) &= \mathfrak{L}_{L_{\mathbf{g}} h}\widehat{\epsilon}(\widehat{\mathbf{x}}). \end{aligned} \tag{9}$$

Thus, if the filter enforces the robust inequality at $(\mathbf{y}, \widehat{\mathbf{x}})$, then $L_{\mathbf{f}} h(\mathbf{x}) + L_{\mathbf{g}} h(\mathbf{x})\,\mathbf{u} \ge -\alpha(h(\mathbf{x}))$ for all $\mathbf{x} \in \mathcal{B}(\widehat{\mathbf{x}})$, and forward invariance of $\mathcal{C}$ follows from Theorem 1. □

Intuitively, an admissible perception module allows the safety margin to follow the estimator's reported error instead of a fixed worst-case error. In regions where the robot is well observed, $\widehat{\epsilon}(\widehat{\mathbf{x}})$ is smaller than $\epsilon_{\max}$, so the error ball and thus the inflated unsafe set are smaller; in poorly observed regions, $\widehat{\epsilon}(\widehat{\mathbf{x}})$ grows again and the inflated obstacles become bigger. Figure 4 illustrates this in the running example: using a global bound $\epsilon_{\max}$ (blue) inflates the two circular obstacles so much that the passage between them almost disappears, whereas using the state-dependent radius $\widehat{\epsilon}(\widehat{\mathbf{x}})$ (pink) yields a smaller inflated layer that still covers all states consistent with the estimated error.

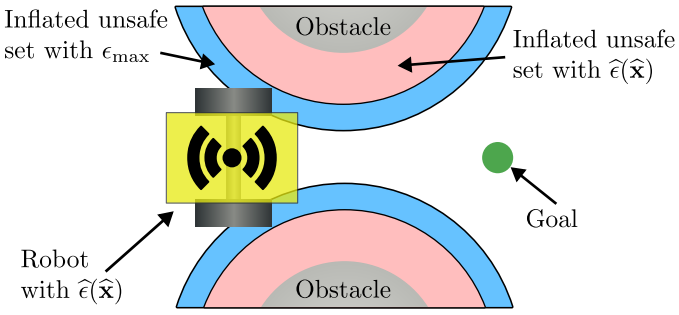


Fig. 4: Effect of replacing a global error bound $\epsilon_{\max}$ (blue) by the perception-driven radius $\widehat{\epsilon}(\widehat{\mathbf{x}})$ (pink) on the running example.

Because Definition 1 only requires over-approximation, boundedness, and regularity, we do not commit to a specific formula for $\widehat{\epsilon}(\cdot)$; any estimator that satisfies these properties can be plugged into (9), allowing the same MR–CBF structure to be used with different perception modules.

B. Grid-based Percentile Lipschitz Estimator

The robot in Figure 4 is still unable to traverse the narrow passage, even with the perception-driven error radius. The

safety margin remains conservative because it is based on global Lipschitz constants that reflect worst-case behavior over the entire workspace. To reduce this conservatism, we introduce a grid-based Lipschitz estimator that makes these sensitivity bounds local and context-aware.

The key idea is that, at any given time, the robot can only move to a limited neighborhood of states in the next few control steps—its *reachable set* [21]. We therefore partition the state space into grid cells and, offline, estimate how fast the CBF terms ($L_f h$, $L_g h$, and $\alpha \circ h$) can change inside each cell by sampling pairs of points and computing finite-difference slopes. Online, we then look only at the cells that intersect the current reachable set and take a percentile of the sampled slopes in those cells. This yields *local percentile Lipschitz bounds* that ignore far-away extremes and filter outliers, producing smaller, situation-dependent margin terms while preserving the MR–CBF structure. We now formalize this construction.

Definition 2 (Grid-based Percentile Lipschitz Estimator): Let $\mathcal{X} \subset \mathbb{R}^n$ be a compact region of interest (i.e., the working domain of the robot) and let $\{G_{ij}\}_{(i,j)\in\mathcal{I}}$ be a partition of $\mathcal{X}$ into axis-aligned cells. For each cell G_{ij} and each quantity $m \in \{L_f h,\ L_g h,\ \alpha \circ h\}$, define the *local slope set* by sampling pairs $(z_p, z_q) \in G_{ij} \times G_{ij}$:

$$\mathcal{S}_m(G_{ij}) \triangleq \left\{ \frac{\|m(z_p) - m(z_q)\|}{\|z_p - z_q\|} : z_p \neq z_q,\ z_p, z_q \in G_{ij} \right\}. \tag{10}$$

Each element of $\mathcal{S}_m(G_{ij})$ is the finite-difference slope of m between two points in the same cell; see Fig. 5.

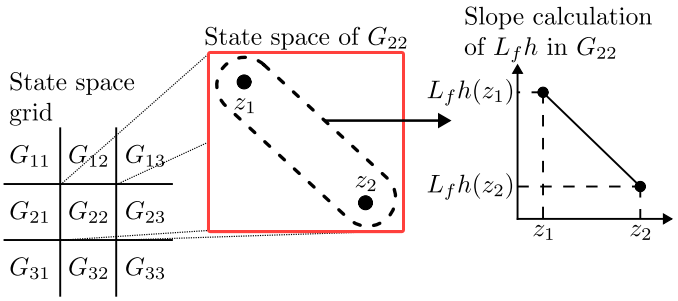


Fig. 5: Illustration of a local slope for $L_f h$ in the grid. Two sample states z_1 and z_2 inside one cell G_{22} are used to calculate the finite-difference slope.

Given a percentile level $\beta \in (0, 100]$, the *cellwise percentile Lipschitz bound* is

$$\widehat{\mathfrak{L}}_m^{(ij)}(\beta) \triangleq \mathrm{Percentile}_\beta\big(\mathcal{S}_m(G_{ij})\big). \tag{11}$$

At time t, let $\mathcal{R}^{(2)}(t)$ denote the two-step reachable set under admissible inputs. Let the index set of *active cells* be

$$\mathcal{I}(t) \triangleq \big\{(i,j) \in \mathcal{I} : G_{ij} \cap \mathcal{R}^{(2)}(t) \neq \emptyset\big\}. \tag{12}$$

The *runtime local Lipschitz envelope* used in the safety margin is then

$$\widetilde{\mathfrak{L}}_m(\beta, t) \triangleq \max_{(i,j)\in\mathcal{I}(t)} \widehat{\mathfrak{L}}_m^{(ij)}(\beta) \tag{13}$$

with $m \in \{L_f h,\ L_g h,\ \alpha \circ h\}$.

After the definition, it is helpful to visualize how these ingredients are combined online. Starting from the current state and control limits, we compute the two-step reachable set $\mathcal{R}^{(2)}(t)$. Then we mark all cells that intersect this set, as the active cells $\mathcal{I}(t)$. Finally, we take the maximum cellwise percentile Lipschitz bound over the active cells $\mathcal{I}(t)$, yielding the runtime envelope $\widetilde{\mathfrak{L}}_m(\beta, t)$ in (13), which is used in the safety margin. This pipeline is summarized in Fig. 6.

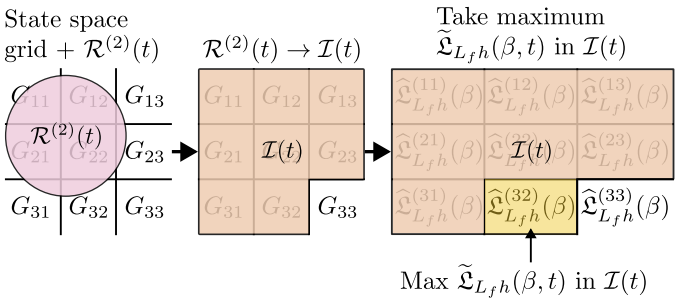


Fig. 6: Overview of the grid-based Lipschitz estimator for $\widetilde{\mathfrak{L}}_{L_f h}$ at runtime. Since, $\widehat{\mathfrak{L}}_{L_f h}^{(32)}(\beta)$ is the maximum Lipschitz constant over the current active cells, $\widetilde{\mathfrak{L}}_{L_f h}(\beta, t) = \widehat{\mathfrak{L}}_{L_f h}^{(32)}(\beta)$.

Proposition 2 (Worst-case safety guarantee for $\beta = 100$): Let $\widehat{\mathfrak{L}}_m^{(ij)}(\beta)$ and $\widetilde{\mathfrak{L}}_m(\beta, t)$ be defined as in (11) and (13). Set $\beta = 100$ and let $\mathcal{U}(t) \triangleq \bigcup_{(i,j)\in\mathcal{I}(t)} G_{ij}$ be the union state spaces of the active cells. If the coverage condition $\mathcal{B}(\widehat{\mathbf{x}}) \cup \mathcal{R}^{(2)}(t) \subseteq \mathcal{U}(t)$ holds, then for every $m \in \{L_f h, L_g h, \alpha \circ h\}$ the envelope $\widetilde{\mathfrak{L}}_m(\beta, t)$ is an upper bound on the true local Lipschitz constant of m over $\mathcal{B}(\widehat{\mathbf{x}}) \cup \mathcal{R}^{(2)}(t)$. Consequently, the MR–CBF parameters

$$\begin{aligned} a(\mathbf{y}) &= (\widetilde{\mathfrak{L}}_{L_f h}(100, t) + \widetilde{\mathfrak{L}}_{\alpha\circ h}(100, t))\widehat{\epsilon}(\widehat{\mathbf{x}}) \\ b(\mathbf{y}) &= \widetilde{\mathfrak{L}}_{L_g h}(100, t)\widehat{\epsilon}(\widehat{\mathbf{x}}), \end{aligned} \tag{14}$$

are robust for all $\mathbf{x} \in \mathcal{B}(\widehat{\mathbf{x}})$, and substituting them into the MR–CBF constraint of (6) recovers the worst-case safety guarantee of Theorem 1.

Proof sketch. Setting $\beta = 100$ in (11) makes each cellwise bound $\widehat{\mathfrak{L}}_m^{(ij)}(100)$ the worst-case finite-difference slope of m on G_{ij}. The runtime envelope (13) then ensures that, for every point z in the union $\mathcal{U}(t) = \bigcup_{(i,j)\in\mathcal{I}(t)} G_{ij}$, the true local Lipschitz constant of m is upper-bounded by $\widetilde{\mathfrak{L}}_m(100, t)$.

Under the coverage condition $\mathcal{B}(\widehat{\mathbf{x}}) \cup \mathcal{R}^{(2)}(t) \subseteq \mathcal{U}(t)$, the MR–CBF parameters constructed from these envelopes are therefore valid Lipschitz-based buffers for all $\mathbf{x} \in \mathcal{B}(\widehat{\mathbf{x}})$. Substituting them into the robust barrier inequality yields exactly the MR–CBF condition (6), so forward invariance of $\mathcal{C}$ follows from Theorem 1. □

The construction satisfies

$$\beta_1 < \beta_2 \;\Rightarrow\; \widetilde{\mathfrak{L}}_m(\beta_1, t) \le \widetilde{\mathfrak{L}}_m(\beta_2, t), \tag{15}$$

i.e., increasing the percentile level (15) cannot reduce the selected bound. Hence, the mechanism does not produce optimistic underestimates.

Intuitively, $\beta = 100$ recovers the worst-case intra-cell Lipschitz constants, and the active-cell max in (13) ensures no "far-field" extreme influences the margin unless it is reachable. Decreasing β allows the robot to manage its risk behavior.

C. Risk-aware Percentile Logic

Building on the grid-based percentile Lipschitz estimator, we now introduce a supervisory policy that adapts the percentile level β and an associated velocity cap $v_{\max}$ online. The goal is to trade conservatism for progress in a controlled way: when progress stalls, the supervisor relaxes the envelope (smaller β) while simultaneously reducing control authority (smaller $v_{\max}$) to keep risk bounded; when clearance and progress are ample, it tightens the envelope (larger β) and increases $v_{\max}$ to recover efficiency and safety.

Figure 7 illustrates this on the running example. Near the bottleneck, the supervisor lowers the level β: more outliers in the Lipschitz estimates are ignored, and the inflated unsafe sets become smaller so that a safe corridor opens. Once the robot exits the narrow region and progress improves with more clearance, the supervisor climbs back to higher levels, restoring a larger margin and higher velocity. We formalize this switching logic as the *Risk-aware Percentile Supervisor*.

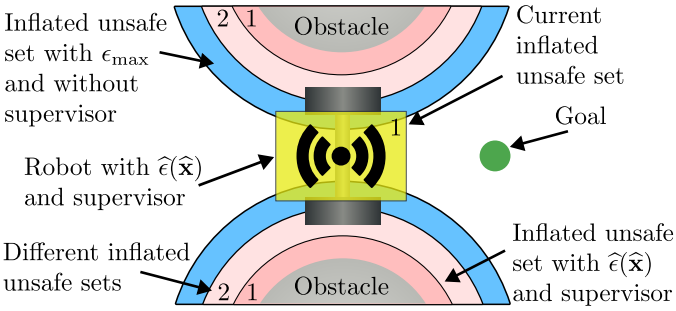


Fig. 7: In the narrow passage, the supervisor drops to a lower level (smaller β_i, smaller $v_{\max,i}$), producing a smaller inflated unsafe set that still covers the estimator's confidence ball but leaves a traversable corridor.

Definition 3 (Risk-aware Percentile Supervisor): Let $L \in \mathbb{N}$ be the number of risk levels. Choose sequences of percentile levels $(\beta_i)_{i=1}^{L}$ and maximum velocities $(v_{\max,i})_{i=1}^{L}$ such that $0 < \beta_1 \le \beta_2 \le \cdots \le \beta_L \le 100$, and $0 < v_{\max,1} \le v_{\max,2} \le \cdots \le v_{\max,L}$.

Let $\widehat{h} \in \mathbb{R}^{N_{\text{obstacles}}}$ be the vector of estimated barrier values and define $\widehat{h}_{\min} \triangleq \min_\ell \widehat{h}_\ell$. Let Π_T be a scalar progress metric computed over a sliding time window of length $T > 0$.

Choose safety and progress thresholds $h_{\text{rel}} < h_{\text{up}}$ and $\eta_{\text{rel}} < \eta_{\text{up}}$ (hysteresis). Introduce a hold time $T_{\text{hold}} > 0$, the minimum time the supervisor must remain at a level before another switch is allowed. Let $\tau(t) \ge 0$ denote the elapsed time since the last level change.

Define the logical conditions

$$\text{relax} \triangleq (\Pi_T < \eta_{\text{rel}}) \;\wedge\; (\widehat{h}_{\min} < h_{\text{rel}}) \;\wedge\; (\tau(t) \ge T_{\text{hold}}),$$
$$\text{up} \triangleq (\Pi_T > \eta_{\text{up}}) \;\wedge\; (\widehat{h}_{\min} > h_{\text{up}}) \;\wedge\; (\tau(t) \ge T_{\text{hold}}).$$

Given the current level $i \in \{1, \dots, L\}$, the supervisor updates the level according to

$$i^+(t) = \begin{cases} \max\{1,\, i-1\}, & \text{if relax,} \\ \min\{L,\, i+1\}, & \text{if up,} \\ i, & \text{otherwise.} \end{cases}$$

At level i, the safety filter uses the percentile envelope $\widetilde{L}_m(\beta_i, t)$ in the MR–CBF margin and enforces the velocity cap $v_{\max,i}$ as an additional constraint in the CBF–QP.

Each level induces a different safety margin in the MR-CBF constraint, and hence a different set of states for which the QP can still find an admissible input. To reason about what happens when the supervisor switches between levels, it is useful to understand how these induced safe sets are related. The next proposition shows that, under monotonicity assumptions on the envelopes, the safe sets are nested: moving to a higher level can only make the set of safe states smaller.

Proposition 3 (Measured safe set nesting): For each supervisor level $i \in \{1, \dots, L\}$, let $\mathcal{C}_i$ denote the safe set induced at that level, i.e., the MR–CBF constraint (6) evaluated with the runtime envelopes $\widetilde{\mathfrak{L}}_m(\beta_i, t)$. Assume the envelope is monotone in the percentile as defined by (15). Then the level-wise safe sets satisfy

$$\mathcal{C}_1 \supseteq \mathcal{C}_2 \supseteq \cdots \supseteq \mathcal{C}_L. \tag{16}$$

Proof sketch. Moving from level i to $i+1$ increases β and increases $v_{\max}$; by (15), the selected envelope for the MR–CBF margin cannot decrease. Hence, the tightened inequality at level $i+1$ implies a subset of the feasible $(\mathbf{y}, \mathbf{u})$ pairs relative to level i, yielding $\mathcal{C}_{i+1} \subseteq \mathcal{C}_i$. □

VI. Experimental Results

We evaluate CALM-CBFs in simulation across two canonical environments—a narrow Gap and a Cluttered environment—using three CBF variants (R0–R2) under identical dynamics, planner, and nominal controller. The goals are to (E1) quantify conservatism and intervention statistics, (E2) study the sensitivity of the risk-aware supervisor, and (E3) investigate the effect of estimator noise on the method.

A. Perception Module

To focus on the effect of the safety margin rather than on a specific estimator design, we use a simple, geometric sensor model that still captures the key feature we care about: spatially varying estimation quality. Concretely, we assume fixed beacons placed in the workspace. Near multiple beacons, the position estimate is accurate (low noise), whereas in poorly covered regions, it is noisy. This yields a tunable, interpretable

noise field that can be visualized and reasoned about (see Fig. 8), while remaining rich enough to test CALM–CBF in both well- and poorly-observed areas.

We model each sensor's spatial influence as a clipped, distance-based magnitude. For a sensor with center $\mathbf{s}_{\text{center}}$ and maximum range $s_{\max}$, its magnitude at position $\mathbf{x} \in \mathbb{R}^2$ is

$$s(\mathbf{x}) \;=\; \text{clip}\left(1 - \frac{\|\mathbf{x} - \mathbf{s}_{\text{center}}\|_2}{s_{\max}},\, 0,\, 1\right),$$

and the aggregate magnitude is the sum over all sensors,

$$S(\mathbf{x}) \;=\; \sum_{r=1}^{N_s} s_r(\mathbf{x}).$$

We map S to a scalar noise level $\sigma : \mathbb{R}^2 \to \mathbb{R}_{\geq 0}$ using two plateaus $\sigma_{\min} \leq \sigma_{\max}$ and a threshold S_{thres} (set to 2 in our experiments):

$$\sigma(\mathbf{x}) = \begin{cases} \sigma_{\max} + \dfrac{1}{S_{\text{thres}}}\big(\sigma_{\min} - \sigma_{\max}\big) S(\mathbf{x}), & S(\mathbf{x}) \leq S_{\text{thres}}, \\ \sigma_{\min}, & S(\mathbf{x}) > S_{\text{thres}}. \end{cases}$$

Because the true state is unknown, we compute a *worst-case* noise over the estimator's confidence ball

$$\mathbb{B}(\widehat{\mathbf{x}}, \epsilon_{\max}) \;\triangleq\; \{\mathbf{z} \in \mathbb{R}^2 :\; \|\mathbf{z} - \widehat{\mathbf{x}}\|_2 \leq \epsilon_{\max}\},$$

via

$$\sigma^\star(\widehat{\mathbf{x}}, \epsilon_{\max}) \;\triangleq\; \max_{\mathbf{z} \in \mathbb{B}(\widehat{\mathbf{x}}, \epsilon_{\max})} \sigma(\mathbf{z}). \tag{17}$$

For efficient evaluation, we use an interior and boundary ring discretization with radii $\{r_j\}_{j=1}^{J} \subset (0, 1]$ and angles $\theta_i = \frac{2\pi i}{N}$:

$$\widetilde{\sigma}^\star(\widehat{\mathbf{x}}, \epsilon_{\max}) \;\triangleq\; \max_{\substack{j=1,\ldots,J \\ i=0,\ldots,N-1}} \sigma\big(\widehat{\mathbf{x}} + r_j\, \epsilon_{\max}\, [\cos\theta_i,\; \sin\theta_i]^\top\big). \tag{18}$$

An example calculation for $\widetilde{\sigma}^\star(\widehat{\mathbf{x}}, \epsilon_{\max})$ is shown in Figure 8. The confidence ball $\mathbb{B}(\widehat{\mathbf{x}}, \epsilon_{\max})$ around the estimated state $\widehat{\mathbf{x}}$ is discretized by interior and boundary rings. The sensor model evaluates $\sigma(\cdot)$ on these samples and returns the maximum value $\widetilde{\sigma}^\star(\widehat{\mathbf{x}}, \epsilon_{\max})$. In this case, $\widetilde{\sigma}^\star(\widehat{\mathbf{x}}, \epsilon_{\max}) = \sigma_{\max}$ since $\sigma_{\max}$ is part of $\mathbb{B}(\widehat{\mathbf{x}}, \epsilon_{\max})$.

Assuming zero-mean Gaussian measurement noise, we adopt a 99.7% confidence bound ($\pm 3\sigma$) and define the perception-driven error radius fed to the MR–CBF margin as

$$\widehat{\epsilon}(\widehat{\mathbf{x}}) \;=\; 3\,\widetilde{\sigma}^\star(\widehat{\mathbf{x}}, \epsilon_{\max}). \tag{19}$$

Each sensor has a finite range $s_{\max}$ and a piecewise-linear magnitude $s_r(\cdot)$; the induced noise map $\sigma(\cdot)$ is locally Lipschitz and bounded in the workspace. The ring approximation $\widetilde{\sigma}^\star(\widehat{\mathbf{x}}, \epsilon_{\max})$ is chosen dense enough so that $\widetilde{\sigma}^\star \geq \sigma(\mathbf{x})$ on the confidence ball. Thus $\widehat{\epsilon}(\widehat{\mathbf{x}}) = 3\,\widetilde{\sigma}^\star(\widehat{\mathbf{x}}, \epsilon_{\max})$ satisfies (P1)–(P3) in Definition 1, making it an admissible perception module. If the 3σ interpretation is adopted, the resulting guarantee is high-probability (99.7%).

B. Experimental Setup

We evaluate three CBF variants in two canonical 2D worlds designed to isolate (i) conservatism in narrow passages and (ii) behavior in clutter.

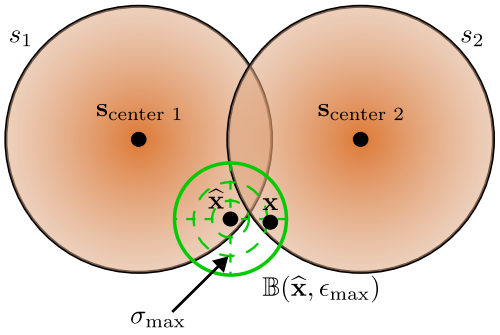


Fig. 8: Illustration of the perception-driven noise bound at one pose. The confidence ball $\mathbb{B}(\widehat{\mathbf{x}}, \epsilon_{\max})$ around the estimated state $\widehat{\mathbf{x}}$ is discretized by interior and boundary rings. The sensor model evaluates $\sigma(\cdot)$ on these samples and returns the maximum value $\widetilde{\sigma}^\star(\widehat{\mathbf{x}}, \epsilon_{\max})$, which serves as a worst-case noise level for all states consistent with the estimate.

a) Environments: Each environment has its own start/goal poses shown in Figure 9.

Gap environment. A 1-D bottleneck with six openings of widths $w \in \{1.75, 1.90, 2.05, 2.20, 2.35, 2.50\}$ m, see Fig. 9a. This yields evenly spaced maximum admissible inflation $M_{\max} = \{0.168, 0.243, 0.318, 0.393, 0.468, 0.543\}$ m. This environment makes the safety margin $M(\widehat{\mathbf{x}})$ the dominant limiting factor and allows comparisons across robots.

Cluttered environment. A dense field of static obstacles is used to demonstrate behavior in complex scenes (see Fig. 9b). Planning difficulty is greater, but the aim is qualitative validation that the method scales beyond the bottleneck case.

b) Robots (CBF strategies): All robots are square with $w_{\text{robot}} = h_{\text{robot}} = 1.0\,\text{m}$, leading to an effective diagonal clearance $d_{\text{robot}} = 1.414\,\text{m}$ in the CBF model. Each robot has the same dynamics, estimator, controller, and sample time (50 Hz); only the construction of the CBF safety margin differs. This choice isolates the effect of the different margin designs: R0 acts as a pure MR–CBF baseline, R1 shows what can be gained by redefining only the error function, and R2 adds the full CALM–CBF machinery on top of the same underlying system. The estimated state is calculated as

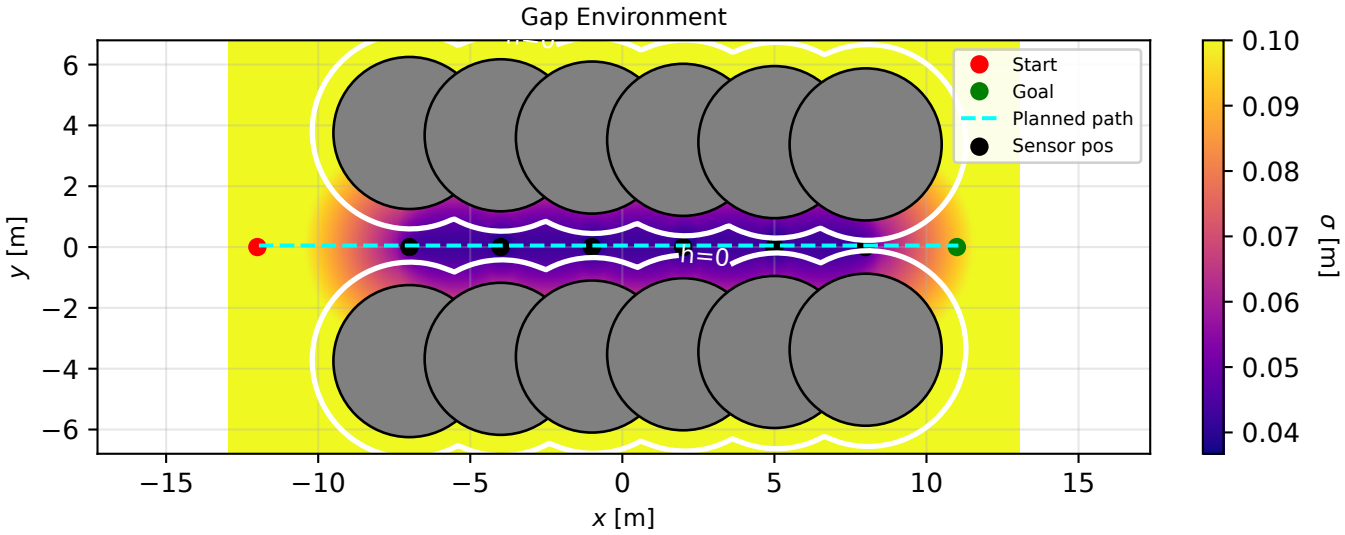
$$\widehat{\mathbf{x}} = \mathbf{x} + \mathcal{N}(0, \sigma(\mathbf{x})). \tag{20}$$

The following CBF strategies are examined during the experiments:

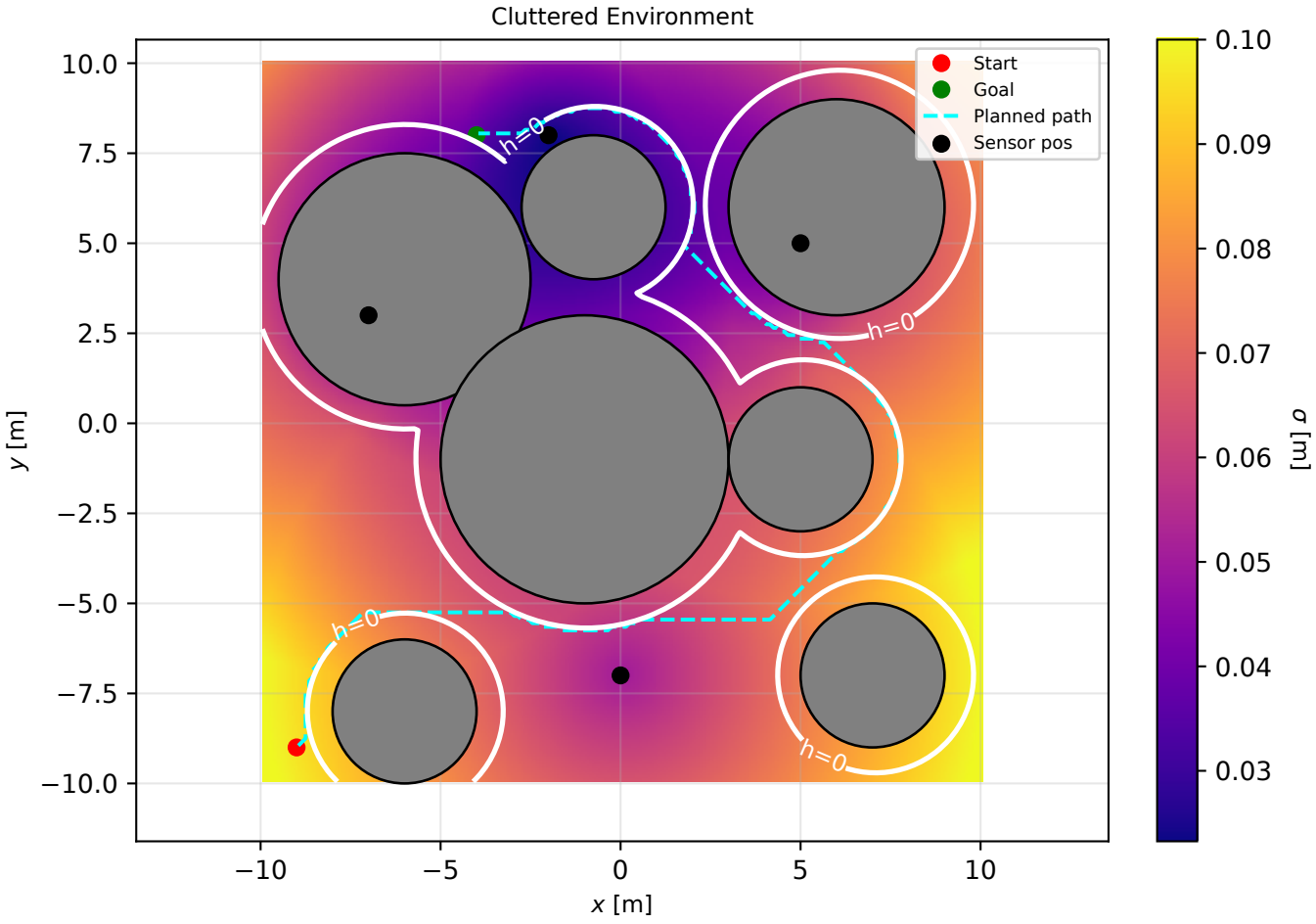
R0. Baseline MR–CBF: uses a global error bound $\epsilon(\widehat{\mathbf{x}}) = \epsilon_{\max}$ and global Lipschitz constants: $\mathfrak{L}_{L_f h}, \mathfrak{L}_{L_g h}$, and $\mathfrak{L}_{\alpha \circ h}$.

R1. MR–CBF + Local error: replaces the global bound by the perception-driven $\epsilon(\widehat{\mathbf{x}}) = \widehat{\epsilon}(\widehat{\mathbf{x}}) \leq \epsilon_{\max}$ but still uses global Lipschitz constants ($\mathfrak{L}_{L_f h}, \mathfrak{L}_{L_g h}$, and $\mathfrak{L}_{\alpha \circ h}$).

R2. CALM–CBF: uses $\epsilon(\widehat{\mathbf{x}}) = \widehat{\epsilon}(\widehat{\mathbf{x}})$ and the grid-local, percentile Lipschitz envelopes with the risk-aware switching supervisor: $\widetilde{\mathfrak{L}}_{L_f h}(\beta, t), \widetilde{\mathfrak{L}}_{L_g h}(\beta, t)$ and $\widetilde{\mathfrak{L}}_{\alpha \circ h}(\beta, t)$. The default parameters for the supervisor are $T = 0.1\,\text{s}$, $T_{\text{hold}} = 0.1\,\text{s}$, $\eta_{\text{rel}} = 0.02\,\text{m}$, $\eta_{\text{up}} = 0.05\,\text{m}$, $h_{\text{rel}} =$



(a) Gap environment



(b) Cluttered environment

Fig. 9: Overview of the two simulated environments used for evaluation. Obstacles are shown in gray; the start and goal are marked by green and red dots. The cyan line indicates the planned trajectory. White contour lines denote the CBF boundary $h(\mathbf{x}) = 0$, separating safe and unsafe regions. The black dots mark sensor locations; their spatial coverage induces the noise field $\sigma(\mathbf{x})$, visualized as a color map.

$0.02\,\mathrm{m}$, and $h_{\mathrm{up}} = 0.06\,\mathrm{m}$. The supervisor hyperparameters were selected after preliminary sweeps over reasonable ranges to obtain trajectories with clear but bounded switching behavior. This makes the sensitivity of CALM–CBF to each parameter visible in the hyperparameter study.

c) Effective clearance c_{eff} as a conservatism metric: To quantify how conservative a CBF strategy is in a narrow passage, we define the *effective clearance*

$$c_{\mathrm{eff}} \triangleq w_{\min} - d_{\mathrm{robot}}, \tag{21}$$

where d_{robot} is the robot's effective diagonal size and $w_{\min}$ is the minimal geometric opening between the two closest obstacles at the stuck pose. Thus c_{eff} measures how much extra free space the safety filter requires beyond the minimum physical clearance needed for the robot to pass. In the experiments, we use the average c_{eff} over runs as our main measure of conservatism for each CBF strategy.

d) Computation of the minimal opening $w_{\min}$: To report the local opening at the stuck pose, we measure the gap along the vector between the two circles that passes through the robot. The obstacles have center $\mathbf{c}_i$ and radius r_i, $\forall i \in \{1, 2\}$. Let the robot position be $\mathbf{p} = (x, y)^\top$ and define the nonzero direction between the circles as $\mathbf{v} = \mathbf{c_2} - \mathbf{c_1}$. Parameterize the line as $\ell(t) = \mathbf{p} + t\,\widehat{\mathbf{v}}$, with $\widehat{\mathbf{v}} = \mathbf{v}/\|\mathbf{v}\|$. The intersection of ℓ with the circles is an interval in t:

$$\begin{aligned} I_i &= [\,t_i - s_i,\ t_i + s_i\,], \\ t_i &= \widehat{\mathbf{v}}^\top(\mathbf{c}_i - \mathbf{p}), \\ s_i &= \sqrt{\max\{\,r_i^2 - \|(\mathbf{I} - \widehat{\mathbf{v}}\widehat{\mathbf{v}}^\top)(\mathbf{c}_i - \mathbf{p})\|_2^2,\ 0\}}. \end{aligned}$$

The possible openings are the distance between the two intersection intervals $w_1 = (t_1 - s_1) - (t_2 + s_2)$ and $w_2 = (t_2 - s_2) - (t_1 + s_1)$. The minimal gap is the nonnegative opening:

$$w_{\min} = \max\Big\{0,\ \max\{w_1,\ w_2\}\Big\}, \tag{22}$$

which is 0 when the intervals overlap or touch.

C. Experiments

Each paragraph starts with a short overview of the experiment, followed by the results of the specific experiment.

***E1** – Overall performance*

Short overview:

- Goal: quantify conservatism of the different robots in the gap world and demonstrate behavior in clutter.
- Design: Each robot runs three seeds per environment (3 seeds$\times$3 robots$\times$2 envs $= 18$ runs).
- Metrics: c_{eff}, safety margin $M(\widehat{\mathbf{x}})$, CBF activation and average norm ($\|u_{\mathrm{CBF}} - u_{\mathrm{nom}}\|_2 / N_{interventions}$).
- Expectation: $c_{\mathrm{eff}}^{\mathrm{R2}} < c_{\mathrm{eff}}^{\mathrm{R1}} < c_{\mathrm{eff}}^{\mathrm{R0}}$; R2 reduces conservatism by tightening margins locally and adapting percentiles, R1 improves over R0 by using local error only.

The average c_{eff} over three seeds is smallest for R2 and largest for R0 in both environments: in the Gap world $\{0.97,\ 0.45,\ \mathbf{0.34}\}\,\mathrm{m}$ for {R0, R1, R2}, and in Cluttered $\{1.08,\ 0.41,\ \mathbf{0.34}\}\,\mathrm{m}$ (Table I). This confirms that the redefined error function and the Lipschitz envelopes effectively reduce conservatism by 65.2% and 68.5% for the gap and cluttered environment, respectively. Moreover, R0 and R1 are not able to reach the goal locations in both environments. The safety–margin traces support this interpretation: R2 tightens where clearance and progress allow but relaxes in tight passages, whereas R0 maintains a uniformly inflated margin; R1 sits in between, see Figure 10. Intervention statistics further show that R0 intervenes most often and most strongly in the Gap world, while R1 and R2 reduce the average intervention magnitude to 0.47; in the Cluttered world, R2 attains both a lower intervention rate (77.4% vs. $83.3\%/83.1\%$ for R0/R1) and the smallest mean intervention (0.30), indicating that percentile Lipschitzing not only opens narrower gaps but also yields milder corrections. Overall, these results validate CALM–CBFs: by holding planning and dynamics fixed and varying only margin construction, we observe consistent, setup-aligned improvements from R0→R1→R2 in both c_{eff} and intervention characteristics.

TABLE I: E1 – Overall performance across the Gap and Cluttered environments. Reported are the effective clearance c_{eff} (m), the fraction of time the CBF is active (% CBF), and the mean intervention magnitude $|\Delta\mathbf{u}|$ per intervention. Values are averaged over three seeds for each robot variant R0–R2.

	Gap environment			Cluttered environment		
Robot	R0	R1	R2	R0	R1	R2
c_{eff} [m]	0.97	0.45	0.34	1.08	0.41	0.34
% CBF	55.7	30.6	36.5	83.3	83.1	77.4
$\lvert\Delta\mathbf{u}\rvert/N_{\mathrm{interventions}}$	0.56	0.47	0.47	0.46	0.34	0.30

***E2** – Sensitivity of the supervisor to hyperparameters*

Short overview:

- Goal: sensitivity of the risk-aware supervisor.
- Design: R2 runs in the gap environment. One hyperparameter is changed at a time, while keeping the other hyperparameters constant with the default parameters. The used hyperparameters are:

$$\begin{aligned} T &\in \{1, 5, 25, 50, 75, 100\}, \\ T_{\mathrm{hold}} &\in \{1, 5, 25, 50, 75, 100\}, \\ h_{\mathrm{rel}} &\in \{0.01, 0.02, 0.03, 0.04, 0.05\}, \\ h_{\mathrm{up}} &\in \{0.04, 0.06, 0.08, 0.10, 0.12\}, \\ \eta_{\mathrm{rel}} &\in \{0.01, 0.02, 0.03, 0.04, 0.05\}, \\ \eta_{\mathrm{up}} &\in \{0.03, 0.05, 0.07, 0.09, 0.11\}. \end{aligned}$$

  Each run is done once with the same seed. This leads to a total of 32 runs.
- Metrics: average percentile, switch count, and time-to-goal.
- Expectation: larger T and T_{hold} reduce jitter but slow reactions (longer time-to-goal); lower h_{rel}, h_{up}, η_{rel} or η_{up} bias toward tighter (more conservative) operation.

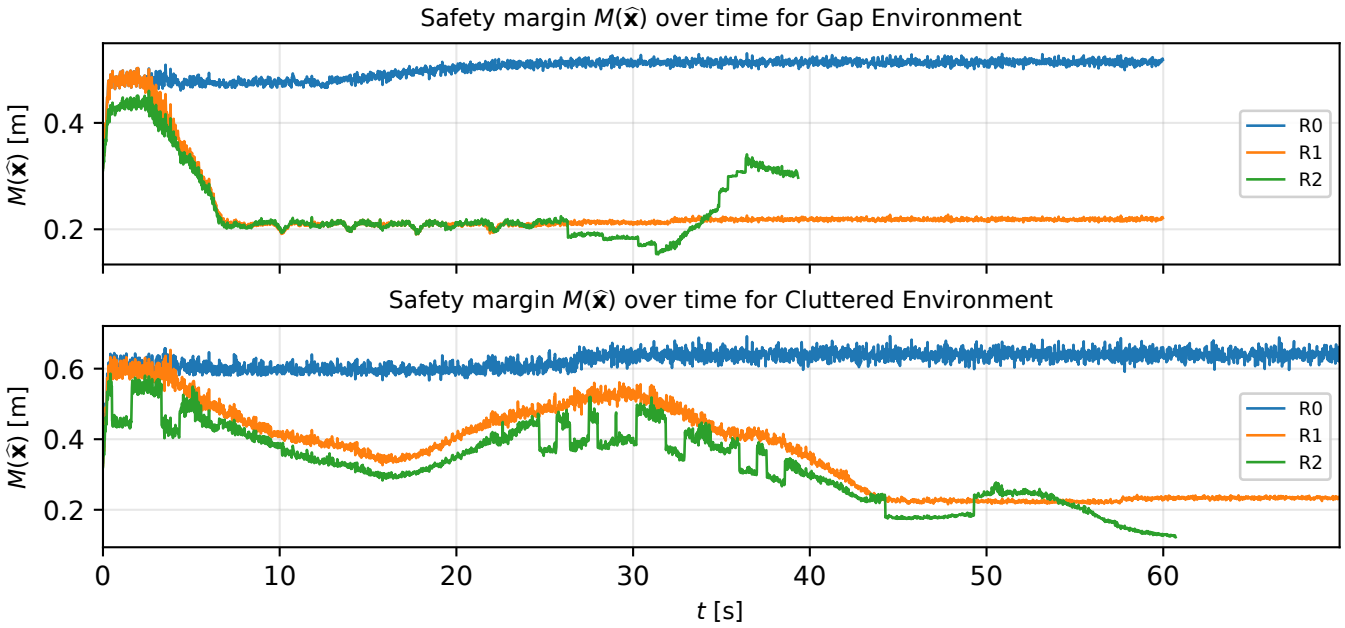


Fig. 10: E1 - Safety margin $M(\hat{\mathbf{x}})$ over time in the Gap (top) and Cluttered (bottom) environments for robots R0–R2 for 1 seed. R0 and R1 do not reach the goal location in both environments.

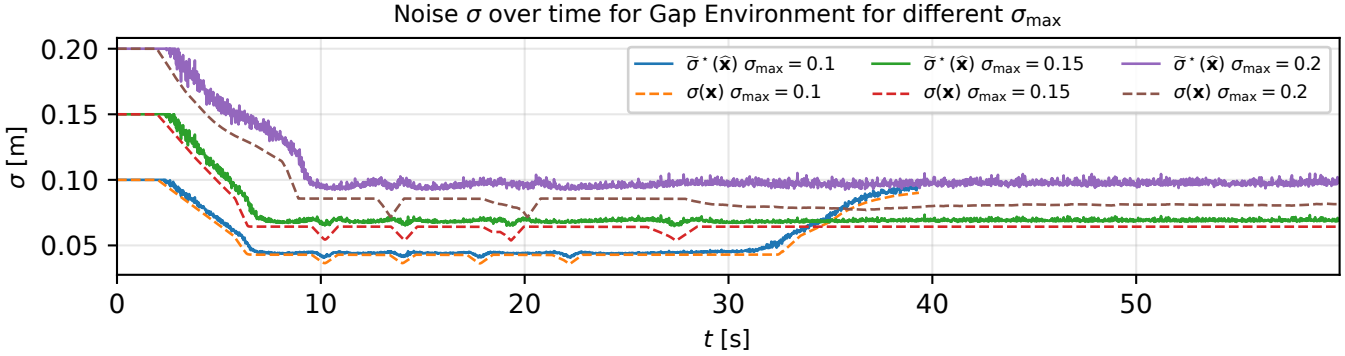


Fig. 11: E3 - Estimated noise $\tilde{\sigma}^\star(\hat{\mathbf{x}})$ versus true noise $\sigma(\mathbf{x})$ over time in the Gap environment for $\sigma_{\max} \in \{0.1,\, 0.15,\, 0.2\}$. In these runs, the robot reaches the goal only for $\sigma_{\max} = 0.1$.

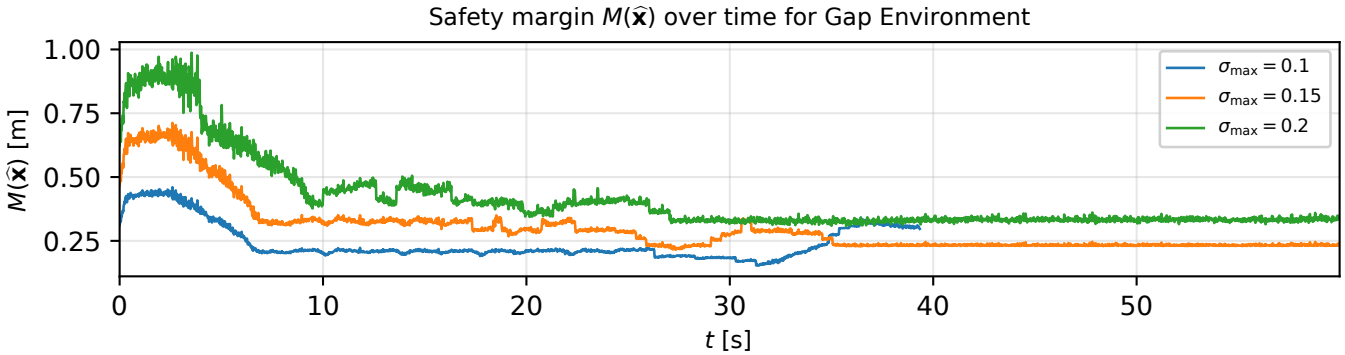


Fig. 12: E3 - Safety margin $M(\hat{\mathbf{x}})$ over time for different $\sigma_{\max}$. Increasing $\sigma_{\max}$ inflates the perception-driven error and thus the margin. In these runs, the robot reaches the goal only for $\sigma_{\max} = 0.1$.

This sensitivity study varies one switching parameter at a time in the Gap world to probe how the supervisor's design (Sec. V-C) trades responsiveness, conservatism, and speed. Longer progress windows T smooth the metric Π_T and reduce chattering (from 23 to 8 switches), but they also delay level changes and lengthen time-to-goal ($35.0 \to 40.5$ s); interestingly, the mean percentile $\bar{\beta}$ increases slightly (toward tighter envelopes) because the smoothed progress more easily satisfies the "up" condition, although too slowly to offset the latency cost (Table II). A larger hold time T_{hold} likewise suppresses switching ($56 \to 5$) and increases t_{goal} ($35.0 \to 43.7$ s), but here $\bar{\beta}$ decreases markedly ($93.6 \to 85.5$): with smoothness, the policy often remains at lower levels (smaller β and smaller $v_{\max}$), which explains the slowdown despite less conservative margins (Table III). The safety hysteresis thresholds mainly tune how often relax/up is allowed. Progress thresholds η_{rel} and η_{up} show small, monotone effects: higher η_{up} reduces switches and nudges $\bar{\beta}$ downward (harder to move up), while η_{rel} barely affects timing in this setup (Tables IV-V). Raising h_{rel} makes relaxation easier, increasing switch count (to 31) and yielding modestly shorter t_{goal} in our runs (down to 34.4 s) with little effect on $\bar{\beta}$ (Table VI). Making h_{up} bigger lowers both switches and $\bar{\beta}$ (it takes more time for the system to switch to a higher level) with negligible impact on t_{goal} (Table VII). Overall, the results align with the supervisor design: (i) temporal smoothing (large T, T_{hold}) decrease switching but risks slower completion; (ii) level thresholds ($h_{\text{rel}}, h_{\text{up}}, \eta_{\text{rel}}, \eta_{\text{up}}$) primarily bias where the system spends time in the level ladder, shifting the balance between envelope tightness (β) and velocity cap $v_{\max}$, with measurable but secondary impact on t_{goal} in this environment.

TABLE II: E2 – Sensitivity of the risk-aware supervisor to the progress-window length T.

T	1	5	25	50	75	100
t_{goal}	35.02	35.02	36.72	39.36	39.40	40.48
N_{switches}	23	23	10	8	8	8
$\bar{\beta}$	91.29	92.88	94.79	94.82	94.69	93.90

TABLE III: E2 – Sensitivity of the risk-aware supervisor to the hold time T_{hold}.

T_{hold}	1	5	25	50	75	100
t_{goal}	35.02	35.02	38.52	42.42	42.90	43.68
N_{switches}	56	23	13	8	5	5
$\bar{\beta}$	93.63	92.88	89.38	88.86	86.78	85.50

TABLE IV: E2 – Sensitivity of the risk-aware supervisor to the relax progress threshold η_{rel}.

η_{rel}	0.01	0.02	0.03	0.04	0.05
t_{goal}	35.74	35.02	35.02	35.02	35.02
N_{switches}	21	23	23	25	24
$\bar{\beta}$	92.88	92.88	92.71	92.63	92.54

TABLE V: E2 – Sensitivity of the risk-aware supervisor to the up progress threshold η_{up}.

η_{up}	0.03	0.05	0.07	0.09	0.11
t_{goal}	35.74	35.02	35.74	35.74	35.74
N_{switches}	23	23	19	18	18
$\bar{\beta}$	93.06	92.88	92.67	91.99	91.72

TABLE VI: E2 – Sensitivity of the risk-aware supervisor to the relax safety threshold h_{rel}.

h_{rel}	0.01	0.02	0.03	0.04	0.05
t_{goal}	35.02	35.02	35.74	34.40	34.40
N_{switches}	21	23	21	31	31
$\bar{\beta}$	92.94	92.88	92.79	92.90	92.69

TABLE VII: E2 – Sensitivity of the risk-aware supervisor to the up safety threshold h_{up}.

h_{up}	0.04	0.06	0.08	0.10	0.12
t_{goal}	35.74	35.02	35.74	35.02	35.02
N_{switches}	23	23	13	11	11
$\bar{\beta}$	93.01	92.88	92.69	92.47	92.32

***E3** – Effect of estimator noise*

Short overview:

- Goal: effect of estimator noise magnitude on CALM-CBFs.
- Design: R2 runs in the gap environment. The noise is varied as $\sigma_{\max} \in \{0.1, 0.15, 0.2\}$ (i.e. $\epsilon_{\max} \in \{0.3, 0.45, 0.6\}$). Each run is done for the same seed, leading to 3 runs.
- Metrics: c_{eff}, $M(\widehat{\mathbf{x}})$, true vs. estimated noise.
- Expectation: higher $\sigma_{\max}$ increases $\widehat{\epsilon}$, hence M, thus increasing c_{eff} (more conservative).

This experiment isolates how the estimator's noise ceiling $\sigma_{\max}$ propagates into the safety margin and influences the required effective clearance c_{eff}. The results match the design: larger $\sigma_{\max}$ inflates $M(\widehat{\mathbf{x}})$ over the entire trajectory (Fig. 12), which increases c_{eff} from 0.34 to 0.51 to 0.65 m (Table VIII). The noise traces (Fig. 11) show that the estimated noise $\tilde{\sigma}^{\star}(\hat{\mathbf{x}})$ is bigger than $\sigma(\mathbf{x})$ and shifts upward with $\sigma_{\max}$. Importantly, only for $\sigma_{\max} = 0.1$ the robot reaches the goal; at 0.15 and 0.2, the obstacles are inflated too much, leading to a deadlock even though the nominal path remains collision-free. Overall, this experiment confirms that conservatism rises predictably with estimator noise, but high $\sigma_{\max}$ can eliminate traversability in narrow gaps.

TABLE VIII: E3 – Effect of maximum noise $\sigma_{\max}$ on required effective clearance c_{eff}. In these runs, the robot only reaches the goal for $\sigma_{\max} = 0.1$.

$\sigma_{\max}$	0.10	0.15	0.20
c_{eff} [m]	0.34	0.51	0.65

VII. Discussion

Our original goal was to retain the measurement-robust CBF (MR–CBF) safety guarantee while reducing the conservatism that can cause deadlocks in traversable narrow passages. The experimental results support this goal. In E1, localizing both the error bound and the Lipschitz envelopes enables CALM–CBF (R2) to pass significantly tighter gaps than the MR–CBF baseline (R0), reducing the required effective clearance c_{eff} by 65.2% in the Gap environment and 68.5% in the Cluttered environment, while remaining collision-free in all runs. At the same time, intervention statistics show that R2 not only opens narrower gaps but also yields milder and less frequent corrections than R0.

From the perspective of the broader field, CALM–CBF can be seen as a step toward combining the advantages of robust and probabilistic CBF approaches. Classical robust methods inflate the unsafe set with a global worst-case margin, ensuring strict safety but often at the cost of conservatism. Probabilistic methods relax this by allowing a small violation probability, but rely on accurate distributional models and abandon worst-case guarantees. Our framework keeps the robust MR–CBF semantics as a reference point and introduces spatially adaptive margins and a tunable risk level on top. The admissible perception interface decouples the safety filter from a specific estimator implementation, and the grid-based envelopes can be precomputed as costmaps, making the method compatible with existing planning and control modules.

At the same time, several limitations remain. First, all experiments are conducted on 2D, holonomic, relative-degree-one dynamics with static obstacles. This choice isolates the effect of the margin mechanism, but many robotic systems are higher-order or nonholonomic. Extending the grid-local/percentile idea to high-relative-degree CBFs and dynamic environments will require bounding additional derivative terms and accounting for admissible velocity and acceleration sets. Second, the perception model is synthetic and Gaussian, with an engineered noise field; in practice, real sensors may exhibit non-Gaussian errors, biases, or unmodeled correlations. E3 highlights that large noise ceilings ($\sigma_{\max}$) can remove traversability in narrow gaps, so the benefits of CALM–CBF are constrained by estimator quality.

VIII. Conclusion

The main takeaway of this work is that MR–CBFs do not have to rely on a single global worst-case margin: by combining perception-driven error bounds, grid-local Lipschitz envelopes, and a simple risk-aware supervisor, the MR–CBF safety margin can be made spatially adaptive and tunable while keeping the original CBF–QP structure. In our narrow-gap and cluttered environments, this design allows CALM–CBF to resolve deadlocks and maintain collision-free behavior. Conceptually, this turns the safety margin from a fixed conservative constant into a design knob that trades off caution against speed in a controlled way.

A first direction of future work is to extend CALM–CBFs to higher-order ($r \geq 2$). For higher order systems, the envelopes must bound terms such as $L_f h$, $L_f^2 h$, and $L_g L_f h$. Their Lipschitz constants (e.g., $\mathfrak{L}_{L_f h}$, $\mathfrak{L}_{L_f^2 h}$, $\mathfrak{L}_{L_g L_f h}$) depend on admissible velocity/acceleration sets, not only position. A natural extension is to co-design the supervisor so that, in addition to $(\beta, v_{\max})$, it also selects dynamics-aware limits (e.g., $a_{\max}$) and scales the corresponding constants consistently, preserving feasibility without over-inflating margins. A second direction could be to transfer CALM–CBFs to real-world systems.

Acknowledgment

I want to thank Chris Pek for his time, guidance, insights, and support throughout this project. I also want to thank Moses Ebere for his constructive feedback and clear thinking during the sessions. I am thankful to my wife, Julia, for her continuous encouragement and unwavering love during this journey. Above all, I thank God for giving me the strength, wisdom, and perseverance to complete this work.

References

[1] Kai-Chieh Hsu, Haimin Hu, and Jaime Fernández Fisac. *The Safety Filter: A Unified View of Safety-Critical Control in Autonomous Systems*. 2023. arXiv: 2309.05837 [eess.SY]. URL: https://arxiv.org/abs/2309.05837.

[2] Aaron D. Ames et al. "Control Barrier Functions: Theory and Applications". In: *2019 18th European Control Conference (ECC)*. 2019, pp. 3420–3431. DOI: 10.23919/ECC.2019.8796030.

[3] Ryan K. Cosner et al. *Measurement-Robust Control Barrier Functions: Certainty in Safety with Uncertainty in State*. arXiv:2104.14030 [eess]. Apr. 2021. DOI: 10.48550/arXiv.2104.14030. URL: http://arxiv.org/abs/2104.14030.

[4] Rin Takano and Masaki Yamakita. "Robust Constrained Stabilization Control Using Control Lyapunov and Control Barrier Function in the Presence of Measurement Noises". In: *2018 IEEE Conference on Control Technology and Applications (CCTA)*. Aug. 2018, pp. 300–305. DOI: 10.1109/CCTA.2018.8511637. URL: https://ieeexplore.ieee.org/document/8511637/?arnumber=8511637.

[5] Sarah Dean et al. *Guaranteeing Safety of Learned Perception Modules via Measurement-Robust Control Barrier Functions*. 2020. arXiv: 2010.16001 [eess.SY]. URL: https://arxiv.org/abs/2010.16001.

[6] Kunal Garg and Dimitra Panagou. "Robust Control Barrier and Control Lyapunov Functions with Fixed-Time Convergence Guarantees". In: *2021 American Control Conference (ACC)*. ISSN: 2378-5861. May 2021, pp. 2292–2297. DOI: 10.23919/ACC50511.2021.9482751. URL: https://ieeexplore.ieee.org/document/9482751/?arnumber=9482751.

[7] Yujie Wang and Xiangru Xu. "Observer-based Control Barrier Functions for Safety Critical Systems". In: *2022 American Control Conference (ACC)*. arXiv:2110.00923 [math]. June 2022, pp. 709–714. DOI: 10.23919/ACC53348.2022.9867262. URL: http://arxiv.org/abs/2110.00923.

[8] Yuhao Zhang, Sequoyah Walters, and Xiangru Xu. "Control Barrier Function Meets Interval Analysis: Safety-Critical Control with Measurement and Actuation Uncertainties". In: *2022 American Control Conference (ACC)*. ISSN: 2378-5861. June 2022, pp. 3814–3819. DOI: 10.23919/ACC53348.2022.9867681. URL: https://ieeexplore.ieee.org/document/9867681/?arnumber=9867681&tag=1.

[9] Devansh R. Agrawal and Dimitra Panagou. "Safe and Robust Observer-Controller Synthesis using Control Barrier Functions". In: *IEEE Control Systems Letters* 7 (2023). arXiv:2211.14364 [cs], pp. 127–132. ISSN: 2475-1456. DOI: 10.1109/LCSYS.2022.3185142. URL: http://arxiv.org/abs/2211.14364.

[10] Jinfeng Chen, Zhiqiang Gao, and Qin Lin. *Robust Control Barrier Functions for Safe Control Under Uncertainty Using Extended State Observer and Output Measurement*. arXiv:2308.13943 [eess]. Aug. 2023. DOI: 10.48550/arXiv.2308.13943. URL: http://arxiv.org/abs/2308.13943.

[11] Binghan He and Takashi Tanaka. "Barrier Pairs for Safety Control of Uncertain Output Feedback Systems". In: *2023 American Control Conference (ACC)*. ISSN: 2378-5861. May 2023, pp. 3669–3674. DOI: 10.23919/ACC55779.2023.10156174. URL: https://ieeexplore.ieee.org/document/10156174/?arnumber=10156174.

[12] Pradeep Sharma Oruganti, Parinaz Naghizadeh, and Qadeer Ahmed. "Safe Control Using High-Order Measurement Robust Control Barrier Functions". In: *2023 American Control Conference (ACC)*. ISSN: 2378-5861. May 2023, pp. 4148–4154. DOI: 10.23919/ACC55779.2023.10155975. URL: https://ieeexplore.ieee.org/document/10155975/?arnumber=10155975.

[13] Binghan He and Takashi Tanaka. "Safety Control of Uncertain MIMO Systems Using Dynamic Output Feedback Barrier Pairs". In: *IEEE Transactions on Automatic Control* (2024). Conference Name: IEEE Transactions on Automatic Control, pp. 1–14. ISSN: 1558-2523. DOI: 10.1109/TAC.2024.3462288. URL: https://ieeexplore.ieee.org/document/10681624/?arnumber=10681624.

[14] Pradeep Sharma Oruganti, Parinaz Naghizadeh, and Qadeer Ahmed. "Robust Control Barrier Functions for Sampled-Data Systems". In: *IEEE Control Systems Letters* 8 (2024). Conference Name: IEEE Control Systems Letters, pp. 103–108. ISSN: 2475-1456. DOI: 10.1109/LCSYS.2023.3346311. URL: https://ieeexplore.ieee.org/document/10371360/?arnumber=10371360.

[15] Yu Zhang et al. "Robust Dual-Filter Safety Control for Mobile Robots in Dynamic Multiobstacle Environments". In: *IEEE/ASME Transactions on Mechatronics* (2025). Conference Name: IEEE/ASME Transactions on Mechatronics, pp. 1–12. ISSN: 1941-014X. DOI: 10.1109/TMECH.2024.3521038. URL: https://ieeexplore.ieee.org/document/10838340/?arnumber=10838340.

[16] Andrew Clark. "Control Barrier Functions for Complete and Incomplete Information Stochastic Systems". en. In: *2019 American Control Conference (ACC)*. Philadelphia, PA, USA: IEEE, July 2019, pp. 2928–2935. ISBN: 978-1-5386-7926-5. DOI: 10.23919/ACC.2019.8814901. URL: https://ieeexplore.ieee.org/document/8814901/.

[17] Andrew Clark, Zhouchi Li, and Hongchao Zhang. "Control Barrier Functions for Safe CPS Under Sensor Faults and Attacks". In: *2020 59th IEEE Conference on Decision and Control (CDC)*. ISSN: 2576-2370. Dec. 2020, pp. 796–803. DOI: 10.1109/CDC42340.2020.9303766. URL: https://ieeexplore.ieee.org/document/9303766/?arnumber=9303766.

[18] Niloofar Jahanshahi, Pushpak Jagtap, and Majid Zamani. "Synthesis of Partially Observed Jump-Diffusion Systems via Control Barrier Functions". In: *IEEE Control Systems Letters* 5.1 (Jan. 2021). Conference Name: IEEE Control Systems Letters, pp. 253–258. ISSN: 2475-1456. DOI: 10.1109/LCSYS.2020.3001562. URL: https://ieeexplore.ieee.org/document/9115020/?arnumber=9115020.

[19] Mouhyemen Khan, Tatsuya Ibuki, and Abhijit Chatterjee. "Safety Uncertainty in Control Barrier Functions using Gaussian Processes". In: *2021 IEEE International Conference on Robotics and Automation (ICRA)*. ISSN: 2577-087X. May 2021, pp. 6003–6009. DOI: 10.1109/ICRA48506.2021.9561504. URL: https://ieeexplore.ieee.org/document/9561504/?arnumber=9561504.

[20] Matti Vahs, Christian Pek, and Jana Tumova. *Belief Control Barrier Functions for Risk-aware Control*. arXiv:2309.06499 [cs]. Sept. 2023. DOI: 10.48550/arXiv.2309.06499. URL: http://arxiv.org/abs/2309.06499.

[21] I.M. Mitchell, A.M. Bayen, and C.J. Tomlin. "A time-dependent Hamilton-Jacobi formulation of reachable sets for continuous dynamic games". In: *IEEE Transactions on Automatic Control* 50.7 (2005), pp. 947–957. DOI: 10.1109/TAC.2005.851439.