Sensor-Based Distributionally Robust Control for Safe Robot Navigation in Dynamic Environments

Kehan Long   Yinzhuang Yi   Zhirui Dai   Sylvia Herbert   Jorge Cortés Nikolay Atanasov

Abstract

We introduce a novel method for mobile robot navigation in dynamic, unknown environments, leveraging onboard sensing and distributionally robust optimization to impose probabilistic safety constraints. Our method introduces a distributionally robust control barrier function (DR-CBF) that directly integrates noisy sensor measurements and state estimates to define safety constraints. This approach is applicable to a wide range of control-affine dynamics, generalizable to robots with complex geometries, and capable of operating at real-time control frequencies. Coupled with a control Lyapunov function (CLF) for path following, the proposed CLF-DR-CBF control synthesis method achieves safe, robust, and efficient navigation in challenging environments. We demonstrate the effectiveness and robustness of our approach for safe autonomous navigation under uncertainty and dynamic obstacles in simulations and real-world experiments with differential-drive robots.

Keywords

Control Barrier Function, Robot Safety, Distributionally Robust Optimization, Autonomous Vehicle Navigation

1 Introduction

Ensuring real-time, high-frequency robot control with safety guarantees in dynamic and unstructured environments is crucial for the effective deployment of autonomous mobile robots. Khatib (1986) introduced the seminal artificial potential fields approach to enable collision avoidance for real-time control of mobile robots. This concept has inspired a wealth of research into the joint consideration of path planning and control, including navigation functions (Rimon and Koditschek 1992), dynamic windows (Fox et al. 1997), and velocity vector fields (De Lima and Pereira 2013). However, many joint planning and control works typically rely on accurate map representations (Oleynikova et al. 2017; Herbert et al. 2017; Arslan and Koditschek 2019; Brito et al. 2019; Schaefer et al. 2021; Kondo et al. 2023; Liu et al. 2023a) updated from onboard sensing to facilitate safe autonomous navigation. In practice, this reliance creates computational bottlenecks, as high-level planning and map updates often occur at lower frequencies and may not adequately account for the uncertainties inherent in high-frequency control (Fox et al. 1997; Corke et al. 2000; Frazzoli et al. 2002; Huang and Grizzle 2023).

Certificate functions have been introduced as powerful tools to assert properties of dynamical systems, such as stability and safety. Among these, Lyapunov functions (Artstein 1983; Sontag 1989) guarantee asymptotic stability for dynamical systems, while barrier functions (Prajna and Jadbabaie 2004) certify forward invariance for desired safe sets. In recent years, control barrier functions (CBFs) have marked a significant advancement in encoding safety constraints for dynamical systems. In conjunction with control Lyapunov functions (CLFs), safe and stable controllers can be synthesized online for control-affine systems via quadratic programming (QP) (Ames et al. 2019). Due to their computational efficiency and formal guarantees, the CLF-CBF QP framework has become a mainstream approach for synthesizing safe and stable controls for various robot systems (Desai and Ghaffari 2022; Choi et al. 2023; Li et al. 2023; Liu et al. 2023b).

However, conventional CBF-based methods commonly assume exact knowledge of robot states, precise system dynamics, and accurate CBF representations. These assumptions rarely hold in practice due to compounded

Contextual Robotics Institute, University of California San Diego, La Jolla, CA 92093, USA.

Corresponding author:
Kehan Long, Contextual Robotics Institute, University of California San Diego, La Jolla, CA 92093, USA.
Email: k3long@ucsd.edu

uncertainties from sensor noise, model inaccuracies, and errors in state estimation. Such uncertainties can degrade theoretical safety guarantees and limit practical robustness. Recent approaches have focused on mitigating these issues by estimating CBFs directly from onboard measurements in unknown environments (Long et al. 2022; Xiao et al. 2022; Abdi et al. 2023; Hamdipoor et al. 2023; Abuaish et al. 2023). Nevertheless, these techniques rely heavily on accurate localization and simplified robot geometries, and typically involve computationally intensive environment reconstruction processes from sensory data.

Motivated by these critical limitations, we propose a novel formulation that leverages distributionally robust optimization (DRO) (Esfahani and Kuhn 2018) to explicitly and efficiently handle multiple realistic uncertainty sources in safe robot navigation. Our distributionally robust control barrier function (DR-CBF) approach directly integrates uncertain sensor measurements and state estimates into safety constraints without requiring precise environmental reconstruction or explicit uncertainty quantification. By using noisy CBF samples within the DRO framework, our formulation significantly enhances the robustness of safe robot navigation in complex, dynamic real-world scenarios. The main **contributions** of this paper are as follows.

- We develop a novel *distributionally robust control barrier function (DR-CBF)* formulation that provides probabilistic safety guarantees by explicitly handling uncertainties in sensor measurements and state estimation;
- Coupling the safety constraint with a control Lyapunov function (CLF), we introduce a CLF-DR-CBF quadratic program for synthesizing safe stabilizing controls for general nonlinear control-affine systems;
- Our approach leverages onboard sensor data and state estimates directly as noisy CBF samples, eliminating the need for precise environmental reconstruction and uncertainty quantification;
- We validate the safety, efficiency, and robustness of our approach in simulated and real experiments with autonomous differential-drive robots navigating in unknown and dynamic environments, illustrated in Fig. 1. An open-source implementation of our CLF-DR-CBF controller is available on our project page*.

2 Related Work

This section reviews related work on dynamic obstacle avoidance, distributionally robust optimization, and CLF-CBF techniques for safe stabilizing control.

Dynamic obstacle avoidance. Robot motion planning algorithms have a rich history, dating back to the

(a) Indoor environment   **(b)** Outdoor environment

Figure 1. ClearPath Jackal robot equipped with a LiDAR sensor navigating in unknown environments.

1950s with the introduction of the Dijkstra (Dijkstra 1959) and A^* (Hart et al. 1968) algorithms for search-based planning. Since then, a substantial amount of research has been dedicated to algorithms for motion planning and trajectory tracking control (Lozano-Perez 1983; Corke et al. 2000; Frazzoli et al. 2002; Huang and Grizzle 2023). A significant contribution was made by Khatib (1986), who introduced artificial potential fields to enable collision avoidance during not only the motion planning stage but also the real-time control of a mobile robot. The formulation was extended to a virtual force field (Borenstein et al. 1991), facilitating safe navigation in uncertain environments. Rimon and Koditschek (1992) developed navigation functions, a particular form of artificial potential functions, which simultaneously ensure collision avoidance and stabilization to a goal configuration. The dynamic window concept was introduced by Fox et al. (1997) to handle dynamic obstacles by proactively filtering out unsafe control actions. More recently, Bajcsy et al. (2019) leveraged Hamilton-Jacobi reachability for autonomous navigation, providing provable safety guarantees in unknown static environments while enabling real-time updates from sensor measurements and accommodating general nonlinear dynamics and planners. In the domain of safe motion planning for dynamic environments, various methods have been developed to handle uncertainties such as obstacle locations (Liu et al. 2023a), human behavior prediction (Schaefer et al. 2021), and localization errors (Summers 2018). Meanwhile, reinforcement learning techniques have advanced real-time autonomous navigation (Pfeiffer et al. 2018; Everett et al. 2021; Chen et al. 2022), often formulating the problem as a partially observable Markov decision process to enable model-free learning of control policies.

Many of these methods require knowledge or estimation of the environment geometry and dynamics, typically in the form of topological (Dudek et al. 1978) or metric (Chatila and Laumond 1985; Borenstein et al. 1991) map representations. The signed distance function (SDF) has emerged as a particularly valuable tool in this regard (Oleynikova et al. 2017; Han et al. 2019; Wu et al. 2023), offering distance and gradient information for safe control and navigation. However, constructing occupancy or SDF maps at control frequency is challenging in complex and dynamic environments.

Distributionally robust optimization. Distributionally robust optimization (DRO) considers parameter uncertainty in optimization problems, and is particularly effective when only a limited number of uncertainty samples are available. The method compensates for the potential discrepancy between empirical and true uncertainty distributions by utilizing uncertainty descriptors like moment ambiguity sets (Van Parys et al. 2016), Kullback–Leibler ambiguity sets (Jiang and Guan 2016), and Wasserstein ambiguity sets (Esfahani and Kuhn 2018; Xie 2021). The DRO framework has been increasingly utilized for its robust performance guarantees against distributional uncertainty in control (Boskos et al. 2024; Mestres et al. 2024a; Long et al. 2023a; Chriat and Sun 2024) and robotics (Ren and Majumdar 2022; Coulson et al. 2021; Ryu and Mehr 2024).

Lathrop et al. (2021) developed a Wasserstein safe variant of RRT, offering finite-sample probabilistic safety guarantees. A DRO approach was introduced by Ren and Majumdar (2022) to enhance policy robustness by iteratively training with adversarial environments generated through a learned generative model. Long et al. (2023b) proposed a DRO formulation for safe stabilizing control under model uncertainty, assuming that nominal safety and stability certificates are provided. Hakobyan and Yang (2022) introduced a distributionally robust risk map as a safety specification tool for mobile robots, reformulating optimal control problem over an infinite-dimensional probability distribution space into a tractable semidefinite program. Boskos et al. (2023) proposed a distributionally robust coverage control algorithm for a team of robots to optimally deploy in region with an unknown event probability density. A Wasserstein tube MPC is presented in Aolaritei et al. (2023) for stochastic systems, which utilizes Wasserstein ambiguity sets to construct tubes around nominal trajectories, enhancing robustness and efficiency with limited noise samples available. Compared to existing applications of DRO in robotics and control, our work explicitly incorporates multiple realistic sources of uncertainty, including noisy sensor measurements and state estimation errors, into a unified distributionally robust control barrier function (DR-CBF) formulation. Crucially, we ensure that the resulting safety constraints remain

convex, preserving computational tractability and enabling efficient real-time control synthesis.

Safe stabilizing control. Quadratic programming that integrates CLF and CBF constraints offers an efficient approach for synthesizing safe stabilizing control inputs for multi-robot navigation (Zhang et al. 2023), legged locomotion (Grandia et al. 2021), and humanoid operation (Khazoom et al. 2022). Despite the efficiency of the CLF-CBF QP in synthesizing safe stabilizing controls, it typically relies on perfect knowledge of system dynamics, state estimates, and barrier function constraints. However, in many robotics applications, various sources of uncertainty can significantly impact performance and reliability. Some recent studies have begun to explore this area, to address uncertainty in system dynamics (Dhiman et al. 2023; Emam et al. 2022), state estimates (Daş and Murray 2022; Wang and Xu 2023), and barrier function constraints (Long et al. 2021; Hamdipoor et al. 2023). These approaches typically utilize robust and probabilistic models to integrate uncertainty into the QP formulation, leading to convex reformulations that enhance robustness.

Particularly in safe robot navigation, a robot might need to estimate barrier functions based on its (noisy) observations. Long et al. (2021) introduced an incremental online learning approach for estimating the barrier function from LiDAR data and proposed a robust reformulation of the CLF-CBF QP by incorporating estimation errors. Majd et al. (2021) integrate time-based rapidly-exploring random trees (RRTs) with CBFs to enable safe navigation in dynamic environments densely populated by pedestrians. Dawson et al. (2022) developed an approach to learn observation-space CBFs utilizing distance data and proposed a two-mode hybrid controller to avoid deadlocks in navigation. A reactive planning algorithm was presented in Liu et al. (2023b) for the safe operation of a bipedal robot with multiple obstacles, utilizing a single differentiable CBF derived from LiDAR point clouds. Abdi et al. (2023) proposed a method for learning vision-based CBF from RGB-D images with pre-training, enabling safe navigation of autonomous vehicles in unseen environments. By effectively decomposing and predicting the spatial interactions of multiple obstacles, Yu et al. (2023) proposed compositional learning of sequential CBFs, enabling obstacle avoidance in dense dynamic environments. Keyumarsi et al. (2024) introduced an efficient Gaussian Process-based method for synthesizing CBFs from LiDAR data, showcasing its effectiveness in a turtlebot navigation task. Zhang et al. (2024) introduced an efficient LiDAR-based framework for goal-seeking and exploration of mobile robots in dynamic environments, utilizing minimum bounding ellipses to represent obstacles and Kalman filters to estimate their velocities.

Conformal Prediction. A related line of research to our work is the application of conformal prediction in robot navigation tasks. Conformal prediction (Shafer and Vovk 2008; Zhao et al. 2024) is a statistical tool for uncertainty quantification that provides valid prediction regions with a user-specified risk tolerance, making it particularly useful for ensuring safety in dynamic environments. Several recent works have explored the integration of conformal prediction into motion planning and control frameworks. Lindemann et al. (2023) used conformal prediction to obtain prediction regions for a model predictive controller. Yang et al. (2023) employed conformal prediction to quantify state estimation uncertainty and design a robust CBF controller based on the estimated uncertainty. An adaptive conformal prediction algorithm was developed by Dixit et al. (2023) to dynamically quantify prediction uncertainty and plan probabilistically safe paths around dynamic agents.

3 Preliminaries

This section introduces our notation and offers a brief review of CLF-CBF QP.

3.1 Notation

The sets of real, non-negative real, and natural numbers are denoted by $\mathbb{R}$, $\mathbb{R}_{\geq 0}$, and $\mathbb{N}$, respectively. For $N \in \mathbb{N}$, we write $[N] := \{1, 2, \ldots N\}$. We denote the distribution and expectation of a random variable Y by $\mathbb{P}$ and $\mathbb{E}_{\mathbb{P}}(Y)$, respectively. We use $\mathbf{0}$ and $\mathbf{1}$ to denote the vector with all entries equal to 0 and 1, respectively. For a scalar x, we define $(x)_+ := \max(x, 0)$. We denote by $\mathbf{I}_n \in \mathbb{R}^{n \times n}$ the identity matrix. For a scalar x and y, we use atan2(y, x) to denote the angle between the positive x-axis and the point (x, y) in radians. The interior and boundary of a set $\mathcal{C} \subset \mathbb{R}^n$ are denoted by Int$(\mathcal{C})$ and $\partial\mathcal{C}$. For a vector $\mathbf{x}$, the notation $|\mathbf{x}|$ represents its element-wise absolute value, while $\|\mathbf{x}\|_1$, $\|\mathbf{x}\|$, and $\|\mathbf{x}\|_\infty$ denote its L_1, L_2, and L_∞ norms, respectively. The gradient of a differentiable function $V : \mathbb{R}^n \to \mathbb{R}$ is denoted by ∇V, while its Lie derivative along a vector field $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^n$ is $\mathcal{L}_\mathbf{f} V = \nabla V^\top \mathbf{f}$. A continuous function $\alpha : [0, a) \to [0, \infty)$ is of class $\mathcal{K}$ if it is strictly increasing and $\alpha(0) = 0$. A continuous function $\alpha : \mathbb{R} \to \mathbb{R}$ is of extended class $\mathcal{K}_\infty$ if it is strictly increasing, $\alpha(0) = 0$, and $\lim_{r \to \infty} \alpha(r) = \infty$. The special orthogonal group of dimension p is denoted by SO(p), which is defined as the set of all $p \times p$ orthogonal matrices with determinant equal to 1: SO$(p) = \{\mathbf{R} \in \mathbb{R}^{p \times p} \mid \mathbf{R}^\top \mathbf{R} = \mathbf{I}_p, \det(\mathbf{R}) = 1\}$.

3.2 CLF-CBF Quadratic Program

Consider a non-linear control-affine system,

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{u} = [\mathbf{f}(\mathbf{x})\ \mathbf{g}(\mathbf{x})] \begin{bmatrix} 1 \\ \mathbf{u} \end{bmatrix} =: \mathbf{F}(\mathbf{x})\underline{\mathbf{u}}, \quad (1)$$

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

The notion of a control Lyapunov function (CLF) (Artstein 1983; Sontag 1989) plays a key role in certifying the stabilizability of control-affine systems.

Definition 3.1. A continuously differentiable function $V : \mathbb{R}^n \to \mathbb{R}$ is a *control Lyapunov function* (CLF) on $\mathcal{X}$ for system (1) if $V(\mathbf{x}) > 0$, $\forall \mathbf{x} \in \mathcal{X} \setminus \{\mathbf{0}\}$, $V(\mathbf{0}) = 0$, and

$$\inf_{\mathbf{u} \in \mathbb{R}^m} \text{CLC}(\mathbf{x}, \mathbf{u}) \leq 0, \quad \forall \mathbf{x} \in \mathcal{X}, \quad (2)$$

where $\text{CLC}(\mathbf{x}, \mathbf{u}) := \mathcal{L}_\mathbf{f} V(\mathbf{x}) + \mathcal{L}_\mathbf{g} V(\mathbf{x})\mathbf{u} + \alpha_V(V(\mathbf{x}))$ is the *control Lyapunov constraint* (CLC) defined for some class $\mathcal{K}$ function α_V.

To facilitate safe control synthesis, we consider a time-varying set $\mathcal{C}(t)$ defined as the zero superlevel set of a continuously differentiable function $h : \mathcal{X} \times \mathbb{R}_{\geq 0} \to \mathbb{R}$:

$$\mathcal{C}(t) := \{\mathbf{x} \in \mathcal{X} : h(\mathbf{x}, t) \geq 0\} \quad (3)$$

Safety of the system (1) can then be ensured by keeping the state $\mathbf{x}$ within the safe set $\mathcal{C}(t)$.

Definition 3.2. A continuously differentiable function $h : \mathbb{R}^n \times \mathbb{R}_{\geq 0} \to \mathbb{R}$ is a *time-varying control barrier function* (TV-CBF) on $\mathcal{X} \subseteq \mathbb{R}^n$ for (1) if there exists an extended class $\mathcal{K}_\infty$ function α_h with:

$$\sup_{\mathbf{u} \in \mathcal{U}} \text{CBC}(\mathbf{x}, \mathbf{u}, t) \geq 0, \quad \forall (\mathbf{x}, t) \in \mathcal{X} \times \mathbb{R}_{\geq 0}, \quad (4)$$

where the *control barrier constraint (CBC)* is:

$$\text{CBC}(\mathbf{x}, \mathbf{u}, t) := \dot{h}(\mathbf{x}, t) + \alpha_h(h(\mathbf{x}, t)) \quad (5)$$

$$= \mathcal{L}_\mathbf{f} h(\mathbf{x}, t) + \mathcal{L}_\mathbf{g} h(\mathbf{x}, t)\mathbf{u} + \frac{\partial h(\mathbf{x}, t)}{\partial t} + \alpha_h(h(\mathbf{x}, t)).$$

Definition 3.2 allows us to consider the set of control values $K_{\text{CBF}}(\mathbf{x}, t) := \{\mathbf{u} \in \mathbb{R}^m : \text{CBC}(\mathbf{x}, \mathbf{u}, t) \geq 0\}$ that render the set $\mathcal{C}(t)$ forward invariant.

Definition 3.3. Let t_0 be a fixed initial time. A time-varying set $\mathcal{C}(t)$ is said to be *forward invariant* under control law $\mathbf{u} : [t_0, \infty) \to \mathbb{R}^m$ if, for any initial state $\mathbf{x}_0 \in \mathcal{C}(t_0)$, there exists a unique maximal solution $\mathbf{x} : [t_0, t_1) \to \mathbb{R}^n$ to the system dynamics in (1) with $\mathbf{x}(t_0) = \mathbf{x}_0$, such that $\mathbf{x}(t) \in \mathcal{C}(t)$ for all $t \in [t_0, t_1)$.

Suppose we are given a baseline feedback controller $\mathbf{u} = \mathbf{k}(\mathbf{x})$ and we aim to ensure the safety and stability of the control-affine system (1). By observing that both the stability and safety constraints in (2), (4) are affine in the control input $\mathbf{u}$, a quadratic program (Ames et al. 2017) can

be formulated to synthesize a safe stabilizing controller:

$$(\mathbf{u}(\mathbf{x}, t), \delta) \in \underset{\mathbf{u} \in \mathbb{R}^m, \delta \in \mathbb{R}}{\arg \min} \|\mathbf{u} - \mathbf{k}(\mathbf{x})\|^2 + \lambda \delta^2, \tag{6}$$
$$\text{s.t. } \mathrm{CLC}(\mathbf{x}, \mathbf{u}) \leq \delta, \mathrm{CBC}(\mathbf{x}, \mathbf{u}, t) \geq 0,$$

where δ denotes a slack variable that relaxes the CLF constraint to ensure feasibility of the QP, controlled by the scaling factor $\lambda > 0$. As discussed in Ames et al. (2017); Mestres et al. (2023), if the CBF h has relative degree of 1 with respect to the system dynamics in (1), the controller in (6) is Lipschitz continuous in $\mathbf{x}$ and piecewise continuous in t. This guarantees unique solutions for the closed-loop system, ensuring the safe set $\mathcal{C}(t)$ remains forward invariant. However, a few practical limitations should be noted: (a) the use of the slack variable δ relaxes the strict satisfaction of the CLF stability constraint, creating a trade-off between safety and stability that depends on the design of the safe set; (b) in numerical implementations, the CBF-QP must be solved at a fixed rate, whereas the theoretical guarantees are valid for executions in continuous time; and (c) while the CBF-QP is generally feasible, it may become infeasible if input constraints limit the available control actions. Researchers have proposed methods to address these challenges, including employing Input-Constrained Control Barrier Functions (ICCBFs) (Agrawal and Panagou 2021), using neural network-based CBFs to account for input saturation (Liu et al. 2023c), and tuning of the class $\mathcal{K}_\infty$ functions (Parwana et al. 2022).

4 Problem Formulation

We consider a mobile robot that relies on noisy range measurements to traverse an unknown dynamic environment towards a desired goal. The robot's motion is governed by control-affine dynamics as in (1). Let $\phi : \mathcal{X} \to \mathbb{R}^p$ be the mapping that projects the robot's state $\mathbf{x}$ to its position vector $\phi(\mathbf{x}) \in \mathbb{R}^p$. We denote the robot orientation as $\mathbf{R}(\mathbf{x}) \in \mathrm{SO}(p)$.

Let $\mathcal{B}_0 \subset \mathbb{R}^p$ represent the robot's shape in its local coordinate frame, i.e., when the robot is at the origin with no rotation or velocity. To describe the shape of the robot in a general state $\mathbf{x}$, we introduce the following linear transformation:

$$\mathcal{B}(\mathbf{x}) = A(\mathbf{x})\mathcal{B}_0 + \mathbf{b}(\mathbf{x}), \tag{7}$$

where $A : \mathbb{R}^n \to \mathbb{R}^{p \times p}$ maps the robot's state $\mathbf{x}$ to an invertible transformation matrix (e.g., rotation), and $\mathbf{b} : \mathbb{R}^n \to \mathbb{R}^p$ defines the translation, both being continuously differentiable.

We use a *signed distance function* (SDF) to describe the robot'a shape. Let $\mathcal{S} \subset \mathbb{R}^p$ be a closed set. The signed distance function $d : \mathbb{R}^p \to \mathbb{R}$ computes the signed distance from a point $\mathbf{q} \in \mathbb{R}^p$ to the set boundary $\partial \mathcal{S}$:

$$d_\mathcal{S}(\mathbf{q}) = \begin{cases} -\min_{\mathbf{q}^* \in \partial \mathcal{S}} \|\mathbf{q} - \mathbf{q}^*\|, & \text{if } \mathbf{q} \in \mathcal{S}, \\ \min_{\mathbf{q}^* \in \partial \mathcal{S}} \|\mathbf{q} - \mathbf{q}^*\|, & \text{if } \mathbf{q} \notin \mathcal{S}. \end{cases} \tag{8}$$

This SDF provides a measure of how far a point $\mathbf{q}$ is from the set $\mathcal{S}$. The function takes negative values for points inside $\mathcal{S}$, positive values for points outside, and is zero on the boundary $\partial \mathcal{S}$.

When the set $\mathcal{S}$ represents the robot's body $\mathcal{B}(\mathbf{x})$, we call its signed distance function the *robot SDF*. To express the SDF of the robot's body in terms of its reference shape $\mathcal{B}_0$, we apply the linear transformation in (7):

$$d(\mathcal{B}(\mathbf{x}), \mathbf{q}) = d_{\mathcal{B}_0}\left(A(\mathbf{x})^{-1}(\mathbf{q} - \mathbf{b}(\mathbf{x}))\right), \tag{9}$$

where $d_{\mathcal{B}_0}(\cdot)$ is the SDF of the reference shape $\mathcal{B}_0$. Closed-form robot SDF expressions are available for simple geometries (e.g., circles, spheres) and can be approximated using feed-forward neural networks for more complex geometries (Park et al. 2019; Koptev et al. 2023).

The robot is equipped with a range sensor (e.g., LiDAR) mounted at a fixed position and orientation in the robot's body frame. The sensor emits multiple rays, each corresponding to a direction in the sensor's field of view, and generates distance measurements along these rays. The measurements are subject to additive noise, modeled as $\boldsymbol{\eta}(\phi(\mathbf{x}), \mathbf{R}(\mathbf{x})) = \bar{\boldsymbol{\eta}}(\phi(\mathbf{x}), \mathbf{R}(\mathbf{x})) + \mathbf{n}$, where $\bar{\boldsymbol{\eta}}(\phi(\mathbf{x}), \mathbf{R}(\mathbf{x}))$ represents the true distances, and $\mathbf{n}$ is the noise vector, assumed to be bounded and independent for each ray. The resulting noisy measurements are denoted as $\boldsymbol{\eta}(\phi(\mathbf{x}), \mathbf{R}(\mathbf{x})) = [\eta_1, \ldots, \eta_K]^\top \in [\eta_{\min}, \eta_{\max}]^K$, where K denotes the number of rays per sensor observation, and $\eta_{\min}, \eta_{\max}$ denote the sensor's minimum and maximum range, respectively.

The environment includes both static and dynamic obstacles. Let the obstacle set at the initial time $t = 0$ be represented as $\mathcal{O}_0 \subset \mathbb{R}^p$. At any time t, the obstacle space $\mathcal{O}(t) \subset \mathbb{R}^p$ is obtained by applying a smooth transformation to the initial obstacle set:

$$\mathcal{O}(t) = A_\mathcal{O}(t)\mathcal{O}_0 + \mathbf{b}_\mathcal{O}(t), \tag{10}$$

where $A_\mathcal{O} : \mathbb{R} \to \mathbb{R}^{p \times p}$ is a time-dependent transformation function mapping time t to an unknown transformation matrix, and $\mathbf{b}_\mathcal{O} : \mathbb{R} \to \mathbb{R}^p$ is a time-dependent translation function. To ensure smooth changes in the environment, we assume that both $A_\mathcal{O}$ and $\mathbf{b}_\mathcal{O}$ are continuously differentiable, preventing obstacles from experiencing sudden jumps, expansions, or contractions. Additionally, obstacles are assumed to have finite thickness, bounded velocities and accelerations, and smooth boundaries to ensure reliable detection.

At time t, the boundary of the obstacle space $\partial \mathcal{O}(t)$ is estimated from the noisy sensor measurements $\boldsymbol{\eta}$. The free
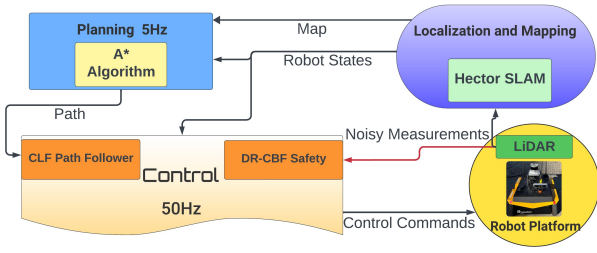
Figure 2. Overview of our approach for safe robot navigation in unknown dynamic environments. The system consists of three main components: (1) localization and mapping, (2) path planning, and (3) control. The contribution of this work lies in the control component, where a novel distributionally robust control barrier function is used to ensure safety in real-time, directly utilizing sensor data, and a control Lyapunov function is used to navigate cluttered and dynamic environments.

space, where the robot can operate safely, is the open set $\mathcal{F}(t) = \mathbb{R}^p \setminus \mathcal{O}(t)$.

Problem Consider a mobile robot with dynamics given by (1), equipped with a range sensor, operating in an unknown dynamic environment. Design a control policy that drives the robot safely and efficiently to a desired goal position $\mathbf{q}_G \in \mathbb{R}^p$. The policy must ensure that the robot's body remains within the free space, i.e., $\mathcal{B}(\mathbf{x}(t)) \subset \mathcal{F}(t)$ for all $t \geq 0$, while accounting for uncertainties in sensing and state estimation.

5 System Overview

This section describes the methods we use for localization and mapping, path planning, and control to enable safe robot navigation, Fig. 2 presents an overview of the robot autonomy components. Our contribution is the control method, which includes a distributionally robust time-varying control barrier function to guarantee safety in dynamic environments, presented in Sec. 6, and a control Lyapunov function for stable path-following control introduced in Sec. 7.

Localization and Mapping. The robot is equipped with a LiDAR scanner and uses the Hector SLAM algorithm (Kohlbrecher et al. 2011) to estimate its pose and build an occupancy map of the environment. We emphasize that the efficiency and accuracy of the constructed occupancy map may not be sufficient to ensure safe navigation in a dynamic environment. We use the map for high-level path planning and employ a low-level controller to guarantee safe tracking using CLF and CBF techniques.

Path Planning. We use the A^* planning algorithm (Hart et al. 1968) to generate a path $\gamma : [0, 1] \mapsto \mathbb{R}^p$ from the robot's current position $\phi(\mathbf{x}) \in \mathbb{R}^p$ to the goal $\mathbf{q}_G \in \mathbb{R}^p$. The path $\gamma(s)$ is parametrized by a scalar $s \in [0, 1]$ such

that $\gamma(0) = \phi(\mathbf{x})$ and $\gamma(1) = \mathbf{q}_G$. As the robot navigates through the environment, the map is continuously updated by the SLAM algorithm and the path is continuously replanned to adapt to changes in the map.

Control. Our control approach simultaneously guarantees the robot's tracking of the planned path γ and its safety from collisions in the dynamically changing environment. In Sec. 6, we develop a distributionally robust CBF that uses noisy distance measurements and estimated robot states to directly to guarantee safety with respect to dynamic obstacles. In Sec. 7, we develop a CLF to track the path γ with stability guarantees.

6 Distributionally Robust Safe Control

In this section, we present a distributionally robust control barrier function (DR-CBF) formulation that enables real-time safety guarantees in cluttered and dynamic environments by utilizing sensor data directly. This formulation is applicable to general control-affine systems (1), as introduced in Sec. 4.

We consider a CBF $h(\mathbf{x}, t)$ with super zero-level set $\mathcal{C}(t) = \{\mathbf{x} \in \mathcal{X} \mid h(\mathbf{x}, t) \geq 0\}$ that satisfies

$$\mathcal{C}(t) \subseteq \{\mathbf{x} \in \mathcal{X} \mid \mathcal{B}(\mathbf{x}(t)) \subseteq \mathcal{F}(t)\},$$

where $\mathcal{B}(\mathbf{x}(t))$ represents the robot's body at state $\mathbf{x}(t)$, and $\mathcal{F}(t)$ is the free space at time t. This establishes a connection between the CBF h and the environment geometry. To develop the DR-CBF formulation, we make the following assumption on the unknown CBF.

Assumption 6.1. *The CBF h has a uniform relative degree of 1 with respect to the system dynamics* (1)*, i.e., the time derivative of $h(\mathbf{x}, t)$ along* (1) *depends explicitly on the control input* **u**.

Under Assumption 6.1, we can write the control barrier constraint associated with $h(\mathbf{x}, t)$ as:

$$\text{CBC}(\mathbf{x}, \mathbf{u}, t) = \qquad\qquad\qquad (11)$$

$$[\nabla_{\mathbf{x}} h(\mathbf{x}, t)]^\top \mathbf{F}(\mathbf{x})\underline{\mathbf{u}} + \frac{\partial h(\mathbf{x}, t)}{\partial t} + \alpha_h(h(\mathbf{x}, t)) =$$

$$\underbrace{[\nabla_{\mathbf{x}} h(\mathbf{x}, t)^\top \mathbf{F}(\mathbf{x}), \; \alpha_h(h(\mathbf{x}, t)), \; \frac{\partial h(\mathbf{x}, t)}{\partial t}]}_{\boldsymbol{\xi}^\top(\mathbf{x}, t)} \begin{bmatrix} \mathbf{u} \\ 1 \\ 1 \end{bmatrix} \geq 0,$$

where we define an uncertainty vector $\boldsymbol{\xi}(\mathbf{x}, t) \in \mathbb{R}^{m+3}$, containing the elements of the time derivative of the barrier function, for each $(\mathbf{x}, t) \in \mathcal{X} \times \mathbb{R}$. Since the environment is unknown and both the sensor measurements and the robot state estimates are noisy, $\boldsymbol{\xi}$ cannot be determined exactly.

Our formulation addresses uncertainties arising from both the barrier function h and the state estimations $\mathbf{x}$. Unlike existing robust or probabilistic CBF approaches,

which often focus on uncertainties in system dynamics (Clark 2019; Long et al. 2022; Dhiman et al. 2023; Wang and Xu 2023; Breeden and Panagou 2023; Das and Burdick 2024), few works tackle the challenges posed by state estimation errors. This stems from the nonlinearity of the dynamics model $\mathbf{F}$ and the barrier function h, which makes it difficult to propagate state estimation errors in the control barrier constraint (CBC) in (11).

Our formulation addresses this challenge by leveraging the power of distributionally robust optimization. Instead of explicitly propagating state estimation errors through the system dynamics and barrier functions, we assume access to M state samples $\{\mathbf{x}_j\}_{j=1}^M$ from a state estimation algorithm. These samples can be obtained, for example, from the Gaussian distributions provided by a Kalman filter (Kalman 1960), particles generated by a particle filter (Djuric et al. 2003), or a graph-based localization algorithm (Grisetti et al. 2010). The state samples, combined with estimates of h (e.g., obtained directly from the distance measurements $\boldsymbol{\eta}$), are directly used to construct samples of the uncertainty vector $\{\boldsymbol{\xi}_i\}_{i=1}^N$, as detailed in Sec. 8.3. By managing uncertainties in $\boldsymbol{\xi}$ using distributionally robust optimization, we ensure the satisfaction of constraint (11) without requiring explicit error propagation.

Before introducing the control synthesis formulation, we review the preliminaries of chance constraints and distributionally robust optimization.

6.1 Chance Constraints and Distributionally Robust Optimization

Consider a random vector $\boldsymbol{\xi}$ with (unknown) distribution $\mathbb{P}^*$ supported on the set $\Xi \subseteq \mathbb{R}^k$. Let $G : \mathbb{R}^m \times \Xi \to \mathbb{R}$ define an inequality constraint $G(\mathbf{u}, \boldsymbol{\xi}) \leq 0$ (e.g., the CBC in (11)). Consider, then, the chance-constrained program,

$$\min_{\mathbf{u} \in \mathbb{R}^m} c(\mathbf{u}),$$
$$\text{s.t. } \mathbb{P}^*(G(\mathbf{u}, \boldsymbol{\xi}) \leq 0) \geq 1 - \epsilon, \tag{12}$$

where $c : \mathbb{R}^m \mapsto \mathbb{R}$ is a convex objective function (e.g., the objective function in (6)) and $\epsilon \in (0, 1)$ denotes a user-specified risk tolerance. Generally, the chance constraint in (12) leads to a non-convex feasible set. To address this, Nemirovski and Shapiro (2006) propose a convex conditional value-at-risk (CVaR) approximation of the original chance constraint.

Value-at-risk (VaR) at confidence level $1 - \epsilon$ for $\epsilon \in (0, 1)$ is defined as $\text{VaR}_{1-\epsilon}^{\mathbb{P}_q}(Q) := \inf_{s \in \mathbb{R}} \{s \mid \mathbb{P}_q(Q \leq s) \geq 1 - \epsilon\}$ for a random variable Q with distribution $\mathbb{P}_q$. As VaR does not provide information about the right tail of the distribution and leads to intractable optimization in general, one can employ CVaR instead, defined as $\text{CVaR}_{1-\epsilon}^{\mathbb{P}_q}(Q) = \mathbb{E}_{\mathbb{P}_q}[Q \mid Q \geq \text{VaR}_{1-\epsilon}^{\mathbb{P}_q}(Q)]$. The resulting

constraint

$$\text{CVaR}_{1-\epsilon}^{\mathbb{P}^*}(G(\mathbf{u}, \boldsymbol{\xi})) \leq 0 \tag{13}$$

creates a convex feasible set, which is a subset of the feasible set in the original chance-constrained problem (12). Additionally, CVaR can be written as the following convex program (Rockafellar and Uryasev 2000):

$$\text{CVaR}_{1-\epsilon}^{\mathbb{P}^*}(G(\mathbf{u}, \boldsymbol{\xi})) := \inf_{s \in \mathbb{R}} [\epsilon^{-1} \mathbb{E}_{\mathbb{P}^*}[(G(\mathbf{u}, \boldsymbol{\xi}) + s)_+] - s]. \tag{14}$$

The formulations in (12) and (13) require knowledge of $\mathbb{P}^*$ to be utilized. However, in many robotics applications, usually only samples of the uncertainty $\boldsymbol{\xi}$ are available (e.g., obtained from LiDAR distance measurements). This motivates us to consider distributionally robust formulations (Esfahani and Kuhn 2018; Xie 2021).

Assuming finitely many samples $\{\boldsymbol{\xi}_i\}_{i \in [N]}$ from the true distribution of $\mathbb{P}^*$ are available, we first describe a way of constructing an ambiguity set of distributions that agree with the empirical distribution. Let $\mathcal{P}_p(\Xi) \subseteq \mathcal{P}(\Xi)$ be the set of Borel probability measures with finite p-th moment with $p \geq 1$. The p-Wasserstein distance between two probability measures μ, ν in $\mathcal{P}_p(\Xi)$ is defined as:

$$W_p(\mu, \nu) := \left(\inf_{\beta \in \mathbb{Q}(\mu, \nu)} \left[\int_{\Xi \times \Xi} \eta(\boldsymbol{\xi}, \boldsymbol{\xi}')^p \mathrm{d}\beta(\boldsymbol{\xi}, \boldsymbol{\xi}') \right] \right)^{\frac{1}{p}}, \tag{15}$$

where $\mathbb{Q}(\mu, \nu)$ denotes the collection of all measures on $\Xi \times \Xi$ with marginals μ and ν on the first and second factors, and η denotes the metric in the space Ξ. Throughout the paper, we take $\eta(\boldsymbol{\xi}, \boldsymbol{\xi}') = \|\boldsymbol{\xi} - \boldsymbol{\xi}'\|_1$ and consider the ambiguity set corresponding to the 1-Wasserstein distance. We denote by $\mathbb{P}_N := \frac{1}{N} \sum_{i=1}^N \delta_{\boldsymbol{\xi}_i}$ the discrete empirical distribution of the available samples $\{\boldsymbol{\xi}_i\}_{i \in [N]}$, and define an ambiguity set, $\mathcal{M}_N^r := \{\mu \in \mathcal{P}_p(\Xi) \mid W_p(\mu, \mathbb{P}_N) \leq r\}$, as a ball of distributions with radius r centered at $\mathbb{P}_N$.

Remark 6.2. (Choice of Wasserstein ball radius): There is a connection between the sample size N and the Wasserstein radius r for constructing the ambiguity set $\mathcal{M}_N^r$. A distribution $\mathbb{P}$ is light-tailed if there exists an exponent ρ such that $A := \mathbb{E}_{\mathbb{P}}[\exp\|\boldsymbol{\xi}\|^\rho] = \int_{\Xi} \exp\|\boldsymbol{\xi}\|^\rho \mathbb{P}(d\boldsymbol{\xi}) < \infty$. If the true distribution $\mathbb{P}^*$ is light-tailed, the choice of $r = r_N(\bar{\epsilon})$ given in Esfahani and Kuhn (2018, Theorem 3.5) is

$$r_N(\bar{\epsilon}) = \begin{cases} \left(\frac{\log(c_1 \bar{\epsilon}^{-1})}{c_2 N} \right)^{\frac{1}{\max\{k, 2\}}} & \text{if } N \geq \frac{\log(c_1 \bar{\epsilon}^{-1})}{c_2}, \\ \left(\frac{\log(c_1 \bar{\epsilon}^{-1})}{c_2 N} \right)^{\frac{1}{\rho}} & \text{else,} \end{cases} \tag{16}$$

where c_1, c_2 are positive constants that depend on ρ, A and k, ensures that the ambiguity ball $\mathcal{M}_N^{r_N(\bar{\epsilon})}$ contains $\mathbb{P}^*$ with probability at least $1 - \bar{\epsilon}$. ●

Fig. 3 provides an illustration of the Wasserstein ambiguity set and its relation to the samples, the empirical distribution, and the true distribution.
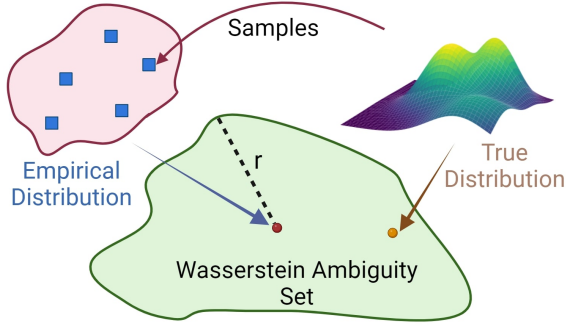
Figure 3. Wasserstein ambiguity set illustration. The figure shows the relationship between the samples, empirical distribution, true distribution, and the Wasserstein ambiguity set. The blue squares represent the available samples from the true distribution (yellow dot), which form the empirical distribution (red dot). The Wasserstein ambiguity set (green region) is constructed as a ball of distributions centered at the empirical distribution, with a radius r that depends on the sample size and the desired confidence level. The ambiguity set aims to contain the true distribution with high probability.

6.2 Distributionally Robust Safety Constraint

Consistently with our exposition of DRO in the previous section, we make the following assumption.

Assumption 6.3. *At each* $(\mathbf{x}, t) \in \mathcal{X} \times \mathbb{R}$, *N samples of the vector* $\boldsymbol{\xi}$ *in* (11) *can be obtained, denoted by* $\{\boldsymbol{\xi}_i\}_{i \in [N]}$.

The samples $\{\boldsymbol{\xi}_i\}_{i \in [N]}$ can be obtained using sensor measurements and state estimation (we discuss this in detail in Sec. 8.3). In many robotic systems, sensing and state estimation may operate at lower frequencies than the control loop. For example, LiDAR measurements or SLAM-based localization may provide updates at about 10 Hz, while the control loop may require computations at 50 Hz. Our DR-CBF formulation addresses this challenge by incorporating samples of $\boldsymbol{\xi}$ derived from potentially delayed or uncertain sensor data and state estimations. By accounting for both the asynchrony and uncertainty inherent in sensing and estimation, our approach ensures probabilistic safe performance under realistic conditions.

Inspired by the CLF-CBF QP formulation in (6), we consider the following distributionally robust formulation to ensure safety with high probability:

$$(\mathbf{u}(\mathbf{x}, t), \delta) = \underset{\mathbf{u} \in \mathbb{R}^m, \delta \in \mathbb{R}}{\arg\min} \|\mathbf{u} - \mathbf{k}(\mathbf{x})\|^2 + \lambda \delta^2,$$

$$\text{s.t.} \quad \mathrm{CLC}(\mathbf{x}, \mathbf{u}) \le \delta, \tag{17a}$$

$$\inf_{\mathbb{P} \in \mathcal{M}_N^r} \mathbb{P}(\mathrm{CBC}(\mathbf{x}, \mathbf{u}, \boldsymbol{\xi})) \ge 0) \ge 1 - \epsilon, \tag{17b}$$

where $\mathcal{M}_N^r$ denotes the ambiguity set with radius r around the empirical distribution $\mathbb{P}_N$. The explicit time dependency of $\mathbf{u}$ on t stems from the random vector $\boldsymbol{\xi}(\mathbf{x}, t)$ in the CBF constraint. The formulation in (17) addresses the inherent

uncertainty in the safety constraint without assuming a specific probabilistic model for $\boldsymbol{\xi}$. The Wasserstein radius r defines the acceptable deviation of the true distribution of $\boldsymbol{\xi}$ from the empirical distribution $\mathbb{P}_N$.

If a controller $\mathbf{u}^*(\mathbf{x}, t)$ satisfies (17b), the following result ensures that the closed-loop system satisfies a chance constraint under the true distribution.

Lemma 6.4. **(Chance-constraint satisfaction under the true distribution):** *Assume the distribution* $\mathbb{P}^*$ *of* $\boldsymbol{\xi}$ *is light-tailed and the Wasserstein radius* $r_N(\bar{\epsilon})$ *is set according to* (16). *If the controller* $\mathbf{u}^*(\mathbf{x}, t)$ *satisfies* (17b) *with* $r = r_N(\bar{\epsilon})$, *then*

$$\mathbb{P}^*(\mathrm{CBC}(\mathbf{x}, \mathbf{u}^*(\mathbf{x}, t), \boldsymbol{\xi})) \ge 0) \ge (1 - \epsilon)(1 - \bar{\epsilon}). \tag{18}$$

Proof. Consider the events $A := \{\mathbb{P}^* \in \mathcal{M}_N^{r_N(\bar{\epsilon})}\}$ and $B := \{\mathrm{CBC}(\mathbf{x}, \mathbf{u}^*(\mathbf{x}, t), \boldsymbol{\xi})) \ge 0\}$. From Esfahani and Kuhn (2018, Theorem 3.4), we have $\mathbb{P}^*(A) \ge 1 - \bar{\epsilon}$. From (17b), we have that

$$\inf_{\mathbb{P} \in \mathcal{M}_N^{r_N(\bar{\epsilon})}} \mathbb{P}(B) \ge 1 - \epsilon. \tag{19}$$

Now, consider the probability of the event B under the true distribution $\mathbb{P}^*$:

$$\mathbb{P}^*(B) \ge \mathbb{P}^*(B \cap A) = \mathbb{P}^*(B|A)\mathbb{P}^*(A) \tag{20}$$

$$\ge \left(\inf_{\mathbb{P} \in \mathcal{M}_N^{r_N(\bar{\epsilon})}} \mathbb{P}(B) \right) \mathbb{P}^*(A) \ge (1 - \epsilon)(1 - \bar{\epsilon}). \quad \square$$

Our previous work (Long et al. 2024) presents a similar result but for a CLF in the context of stabilization. According to Lemma 6.4, the safety of the closed-loop system is guaranteed with high probability. However, the optimization problem in (17) is intractable (Hota et al. 2019; Esfahani and Kuhn 2018) due to the infimum over the Wasserstein ambiguity set. In Sec. 6.3, we discuss our approach to identify tractable reformulations of (17) and facilitate online safe control synthesis.

6.3 Tractable Convex Reformulation

Next, we demonstrate how the samples $\{\boldsymbol{\xi}_i\}_{i \in [N]}$ from Assumption 6.3 can be used to obtain a tractable reformulation of (17).

Proposition 6.5. **(Distributionally robust safe control synthesis):** *Given samples* $\{\boldsymbol{\xi}_i\}_{i \in [N]}$ *of* $\boldsymbol{\xi}$ *as in* (11), *if* $(\mathbf{u}^*, \delta^*, s^*, \{\beta_i^*\}_{i \in [N]})$ *is a solution to the quadratic program:*

$$\min_{\mathbf{u} \in \mathbb{R}^m, \delta \in \mathbb{R}, s \in \mathbb{R}, \beta_i \in \mathbb{R}} \|\mathbf{u} - \mathbf{k}(\mathbf{x})\|^2 + \lambda \delta^2, \tag{21}$$

$$\text{s.t.} \quad \mathrm{CLC}(\mathbf{x}, \mathbf{u}) \le \delta,$$

$$r\|\underline{\mathbf{u}}\|_\infty \le s\epsilon - \frac{1}{N}\sum_{i=1}^N \beta_i,$$

$$\beta_i \ge s - [\underline{\mathbf{u}} \;\; 1 \;\; 1]^\top \boldsymbol{\xi}_i, \;\; \beta_i \ge 0, \;\; \forall i \in [N],$$

then $(\mathbf{u}^*, \delta^*)$ *is also a solution to the distributionally robust chance-constrained program in* (17).

Proof. The safety constraint (17b) is equivalent to $\sup_{\mathbb{P}\in\mathcal{M}_N^r}\mathbb{P}(-\mathrm{CBC}(\mathbf{x},\mathbf{u},\boldsymbol{\xi}) \ge 0) \le \epsilon$. Using the CVaR approximation of the chance constraint (13), we obtain a convex conservative approximation of (17b):

$$\sup_{\mathbb{P}\in\mathcal{M}_N^r} \mathrm{CVaR}_{1-\epsilon}^{\mathbb{P}}(-\mathrm{CBC}(\mathbf{x},\mathbf{u},\boldsymbol{\xi})) \le 0. \quad (22)$$

From (14), this is equivalent to

$$\sup_{\mathbb{P}\in\mathcal{M}_N^r} \inf_{s\in\mathbb{R}} \left[\frac{1}{\epsilon}\mathbb{E}_{\mathbb{P}}[(-\mathrm{CBC}(\mathbf{x},\mathbf{u},\boldsymbol{\xi})+s)_+] - s\right] \le 0. \quad (23)$$

Based on Hota et al. (2019, Lemma V.8) and Esfahani and Kuhn (2018, Theorem 6.3), with the 1-Wasserstein distance, the following inequality is a sufficient condition for (23) to hold:

$$rL(\mathbf{u}) + \inf_{s\in\mathbb{R}}\left[\frac{1}{N}\sum_{i=1}^N(-\mathrm{CBC}(\mathbf{x},\mathbf{u},\boldsymbol{\xi}_i)+s)_+ - s\epsilon\right] \le 0 \quad (24)$$

where $L(\mathbf{u})$ is the Lipschitz constant of $-\mathrm{CBC}(\mathbf{x},\mathbf{u},\boldsymbol{\xi})$ in $\boldsymbol{\xi}$. Now, from (11), we have $\mathrm{CBC}(\mathbf{x},\mathbf{u},\boldsymbol{\xi}) = [\underline{\mathbf{u}} \;\; 1 \;\; 1]^\top \boldsymbol{\xi}$. Therefore, we can define the convex function $L : \mathbb{R}^m \mapsto \mathbb{R}_{>0}$ by

$$L(\mathbf{u}) = \|[\underline{\mathbf{u}} \;\; 1 \;\; 1]^\top\|_\infty = \max\{1, \|\underline{\mathbf{u}}\|_\infty\} = \|\underline{\mathbf{u}}\|_\infty \quad (25)$$

The function $\boldsymbol{\xi} \mapsto -\mathrm{CBC}(\mathbf{x},\mathbf{u},\boldsymbol{\xi})$ is Lipschitz in $\boldsymbol{\xi}$ with constant $L(\mathbf{u})$. This is because the Lipschitz constant of a differentiable affine function equals the dual norm of its gradient, and the dual norm of the L_1 norm is the L_∞ norm. Thus, the following is a conservative approximation of (17),

$$\min_{\mathbf{u}\in\mathbb{R}^m, \delta\in\mathbb{R}} \|\mathbf{u} - \mathbf{k}(\mathbf{x})\|^2 + \lambda\delta^2, \quad (26)$$

s.t. $\mathrm{CLC}(\mathbf{x},\mathbf{u}) \le \delta$,

$$rL(\mathbf{u}) + \inf_{s\in\mathbb{R}}\left[\frac{1}{N}\sum_{i=1}^N(-\mathrm{CBC}(\mathbf{x},\mathbf{u},\boldsymbol{\xi}_i)+s)_+ - s\epsilon\right] \le 0.$$

Lastly, as shown in Long et al. (2023b, Proposition IV.1), the bi-level optimization in (26) can be rewritten as (21). $\square$

Formally, the safe set $\mathcal{C}(t)$ is defined by a single CBF $h(\mathbf{x},t)$, but its value, gradient, and time derivative (defining $\boldsymbol{\xi}(\mathbf{x},t)$) are not directly known in practice. By leveraging observations of $\boldsymbol{\xi}$ derived from sensor measurements and

state estimates, and applying Proposition 6.5, we are able to synthesize safe controllers with distributionally robust guarantees. Due to the convex reformulation of (17b), our approach inherently introduces some conservatism. However, in practice, this conservatism can be effectively managed by tuning the Wasserstein radius r and the safety probability ϵ. These parameters provide flexibility to adapt the formulation based on specific application requirements, sensor capabilities, and state estimation errors.

The Lipschitz continuity and regularity of distributionally robust controllers are characterized in Mestres et al. (2024a). Together with Lemma 6.4, this enables safe robot control with point-wise probabilistic guarantees in unknown dynamic environments.

Remark 6.6. (Uncertainty in System Dynamics): We have assumed that there is no uncertainty in the system dynamics $\mathbf{F}$ to simplify the presentation in Sec. 6.3. However, our approach can be extended if this is not the case as long as samples of $\mathbf{F}(\mathbf{x})$ are available. These samples can be combined with samples of robot state and $h(\mathbf{x},t)$ to construct the uncertainty vector $\{\boldsymbol{\xi}_i\}_{i=1}^N$ and ensure the validity of Proposition 6.5. $\bullet$

7 Control Lyapunov Function Based Path Following

In this section, we introduce a control strategy that accurately tracks the planned path γ. This control strategy will serve as the basis for specifying the control Lyapunov constraint (CLC) in our distributionally robust safe control synthesis in (21).

We begin by stating the following assumptions:

Assumption 7.1. *The state space* $\mathcal{X} \subset \mathbb{R}^n$ *is compact.*

Assumption 7.2. *Given a desired reference point* $\mathbf{q} \in \mathbb{R}^p$, *let* $\mathcal{E}(\mathbf{q}) = \{\mathbf{x} \in \mathcal{X} : \phi(\mathbf{x}) = \mathbf{q}\}$. *For each* $\mathbf{q}$, *assume that there exists a continuously differentiable function* $V : \mathcal{X} \times \mathbb{R}^p \to \mathbb{R}_{\ge 0}$ *with the following properties.*

1. *The function* $V(\mathbf{x})$ *is positive definite with respect to the error* $\phi(\mathbf{x}) - \mathbf{q}$, *i.e.,* $V(\mathbf{x}) > 0$ *for all* $\mathbf{x} \notin \mathcal{E}(\mathbf{q})$ *and* $V(\mathbf{x}) = 0$ *if* $\mathbf{x} \in \mathcal{E}(\mathbf{q})$.
2. *There exists a continuous control law* $\hat{\mathbf{u}}(\mathbf{x}, \mathbf{q})$ *such that the time derivative of* V *along the trajectories of the system* (1) *satisfies* $\dot{V}(\mathbf{x}) < 0$ *for all* $\mathbf{x} \notin \mathcal{E}(\mathbf{q})$.
3. *For all* $\mathbf{x} \in \mathcal{E}(\mathbf{q})$, *we have* $\mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\hat{\mathbf{u}}(\mathbf{x}, \mathbf{q}) = \mathbf{0}$.

A specific construction of such a function is provided in Sec. 8.1.

7.1 Stabilization to a Goal Position

We now establish the asymptotic stability of the goal set $\mathcal{E}(\mathbf{q})$ for the closed-loop system dynamics (1) with control law $\hat{\mathbf{u}}(\mathbf{x}, \mathbf{q})$.

Lemma 7.3. (Asymptotic stability of the goal set): *Under Assumptions 7.1 and 7.2, the goal set $\mathcal{E}(\mathbf{q}) = \{\mathbf{x} \in \mathcal{X} : \phi(\mathbf{x}) = \mathbf{q}\}$ is asymptotically stable for the closed-loop dynamics (1) under the control law $\hat{\mathbf{u}}(\mathbf{x}, \mathbf{q})$.*

Proof. By Assumption 7.1, $\mathcal{X}$ is compact. Since $\mathcal{E}$ is a closed subset of $\mathcal{X}$, it follows that $\mathcal{E}(\mathbf{q})$ is also compact. In addition, by LaSalle's Invariance Principle (Khalil 2002), the conditions in Assumption 7.2 imply that the closed-loop system trajectory converges to the largest invariant set contained in $\{\mathbf{x} \in \mathcal{X} : \dot{V}(\mathbf{x}) = 0\} \subset \mathcal{E}(\mathbf{q})$. This implies that $\mathcal{E}$ is asymptotically attractive too. Therefore, $\mathcal{E}(\mathbf{q})$ is asymptotically stable for the closed-loop system dynamics under the control law $\hat{\mathbf{u}}(\mathbf{x}, \mathbf{q})$. $\qquad\square$

Lemma 7.3 establishes that, under the control law $\hat{\mathbf{u}}(\mathbf{x}, \mathbf{q})$ in Assumption 7.2, the position of the system in (1) satisfies $\phi(\mathbf{x}(t)) \to \mathbf{q}$ as $t \to \infty$.

7.2 CLF-Based Path Following

Building upon the stability result in Lemma 7.3, we now extend the position convergence to path following. Our goal is to achieve smooth navigation by dynamically adjusting a moving goal point along the planned path γ. Inspired by reference governor control techniques (Garone and Nicotra 2015; Li et al. 2020), we consider a scalar $g(t) \in [0, 1]$ with dynamics:

$$\dot{g} = \frac{k}{1 + \|\phi(\mathbf{x}) - \gamma(g)\|}(1 - g^\zeta), \qquad (27)$$

where $k \in \mathbb{R}_{>0}$ is a scaling factor, and $\zeta \in \mathbb{N}$ ensures that g asymptotically approaches but never exceeds 1. The dynamics in (27) are designed such that the reference point $\gamma(g)$ moves along the path γ at a speed inversely proportional to the distance between the current robot position $\phi(\mathbf{x})$ and $\gamma(g)$, facilitating a responsive path following behavior. The initial condition for g is set to $g(0) = 0$, corresponding to the starting point of the path $\gamma(0)$.

Lemma 7.4. (Asymptotic stability of the governor dynamics): *The equilibrium point $g^* = 1$ is asymptotically stable for the governor dynamics in (27).*

Proof. Note that $[0, 1]$ is forward invariant under (27). Consider the candidate Lyapunov function:

$$V_g(g) = \frac{1}{2}(1 - g)^2. \qquad (28)$$

Note that V_g is positive definite with respect to $g = 1$. Its time derivative along the trajectories of (27) is

$$\dot{V}_g(g) = -(1 - g)\dot{g} = -\frac{k(1 - g)}{1 + \|\phi(\mathbf{x}) - \gamma(g)\|}(1 - g^\zeta)$$

Since $\dot{V}_g(g) \leq 0$ for all $g \in [0, \infty)$ and $\dot{V}_g(g) = 0$ if and only if $g = 1$, we conclude that $g = 1$ is globally asymptotically stable (over $[0, \infty)$) for the governor dynamics (27). $\qquad\square$

From Lemma 7.3, for a goal position $\mathbf{q}$, the control-affine system under the control law $\hat{\mathbf{u}}(\mathbf{x}, \mathbf{q})$ converges to the set of equilibrium points $\mathcal{E}(\mathbf{q}) = \{\mathbf{x} \in \mathcal{X} : \phi(\mathbf{x}) = \mathbf{q}\}$. In the path-following context, we make $\mathbf{q}$ move along the path γ, resulting in the control law $\hat{\mathbf{u}}(\mathbf{x}, \gamma(g))$ and the equilibrium set $\mathcal{E}(\gamma(g))$. The following result formalizes the asymptotic convergence of the interconnected system.

Theorem 7.5. (Asymptotic stability of the interconnected system): *Consider the interconnected system consisting of the governor dynamics (27) and the closed-loop dynamics (1) with the control law $\hat{\mathbf{u}}(\mathbf{x}, \gamma(g))$. Under Assumptions 7.1 and 7.2, there exists a sufficiently small $k^* > 0$ such that, for all $k \in (0, k^*]$, the equilibrium set $\mathcal{E}(\gamma(1)) \times \{1\}$ is asymptotically stable for the interconnected system.*

Proof. We prove the result using singular perturbation theory (Khalil 2002). We view the control-affine dynamics with state $\mathbf{x} \in \mathcal{X}$ as the fast subsystem and the governor dynamics with state $g \in [0, 1]$ as the slow subsystem. First, we analyze the reduced-order model, obtained by setting $\dot{g} = 0$ in (27):

$$0 = \frac{k}{1 + \|\phi(\mathbf{x}) - \gamma(g)\|}(1 - g^\zeta). \qquad (29)$$

The solution to (29) is $g = 1$, which corresponds to the endpoint of the path $\gamma(1)$. By Lemma 7.3, when $g = 1$, the equilibrium set $\mathcal{E}(\gamma(1))$ for the constant reference point $\gamma(1)$ is asymptotically stable for the control-affine dynamics with the control law $\hat{\mathbf{u}}(\mathbf{x}, \gamma(1))$.

Next, we consider the boundary-layer system, obtained by introducing a fast time scale $\tau = t/\nu$ and taking the limit $\nu \to 0$:

$$\frac{d\mathbf{x}}{d\tau} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\hat{\mathbf{u}}(\mathbf{x}, \gamma(\bar{g})), \qquad (30)$$

where $\bar{g} \in [0, 1]$ is treated as a fixed parameter. By Lemma 7.3, for each fixed $\bar{g}$, the equilibrium set $\mathcal{E}(\gamma(\bar{g}))$ is asymptotically stable for the boundary-layer system (30).

Next, we analyze the reduced slow system, obtained by substituting the quasi-steady-state solution $\mathbf{x}(g) \in \mathcal{E}(\gamma(1))$ into the slow subsystem:

$$\dot{g} = \frac{k}{1 + \|\gamma(g) - \gamma(g)\|}(1 - g^\zeta) = k(1 - g^\zeta). \qquad (31)$$

This system has a unique equilibrium point $g^* = 1$, which is globally asymptotically stable on $[0, \infty)$.

Consequently, as discussed in Khalil (2002, Appendix C.3), there exists a sufficiently small $k^* > 0$ such that,

for all $k \in (0, k^*]$, the equilibrium set $\mathcal{E}(\gamma(1)) \times \{1\}$ is asymptotically stable for the interconnected system. $\square$

In practical implementations, the control input is applied at a finite sampling rate. To preserve stability guarantees, we use a sampling frequency of 50 Hz in the simulations and experiments, ensuring that the discretized behavior closely approximates the continuous-time result.

Remark 7.6. (Practical considerations for control bounds): The original CLF-CBF QP formulation in (6) assumes no control bounds, a condition often not met in real-world robot applications due to physical limitations, such as maximum speed and acceleration. To ensure the applicability of our approach within these practical constraints, one can tune the parameters k in the governor dynamics (27) to establish a smooth path-following behavior that respects the control bounds. Specifically, increasing k slows down the progression of the reference goal $\gamma(g)$, reducing the control effort required. ●

Remark 7.7. (Practical considerations for convergence): Theorem 7.5 establishes the asymptotic stability of the equilibrium set $\bar{\mathcal{E}}(\gamma(1)) \times 1$ for the interconnected system, which implies that the robot's position $\phi(\mathbf{x}(t))$ converges to the endpoint $\gamma(1)$ of the path as $t \to \infty$. However, in practical applications, it is important to consider the finite-time convergence of the robot to its destination. To address this, we introduce a threshold $\mu^* > 0$ and consider the robot to have effectively reached its destination when $|\phi(\mathbf{x}(t)) - \gamma(1)| \le \mu^*$. By setting an appropriate value for μ^*, we can guarantee that the robot completes its navigation task within a finite time, while still ensuring that it reaches a sufficiently close vicinity of the goal $\gamma(1)$. ●

8 Application to Differential-Drive Robot

While the theory of our control method is applicable to general control-affine systems (1), our experiments focus on a wheeled differential-drive robot.

We consider a robot with state $\mathbf{x} := [x, y, \theta]^\top \in \mathcal{X} \subseteq \mathbb{R}^2 \times [-\pi, \pi)$, input $\mathbf{u} := [v, \omega]^\top \in \mathcal{U} \subset \mathbb{R}^2$, and dynamics:

$$\dot{\mathbf{x}} = \begin{bmatrix} \cos(\theta) & 0 \\ \sin(\theta) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}. \quad (32)$$

The function $\phi : \mathcal{X} \to \mathbb{R}^2$ projecting the robot state $\mathbf{x}$ to its position is $\phi(\mathbf{x}) = \phi([x, y, \theta]^\top) = [x, y]^\top$.

We define the state space of the robot as $\mathcal{X} = \mathcal{D} \times [-\pi, \pi)$, where $\mathcal{D} \subset \mathbb{R}^2$ is a sufficiently large compact set containing the environment of interest, including the goal point $\mathbf{q}_G$ and the planned path γ.

In this section, we instantiate the control strategy developed in the previous sections to the case of differential-drive dynamics in (32). We first design a

(a) Path tracking **(b)** Safe navigation
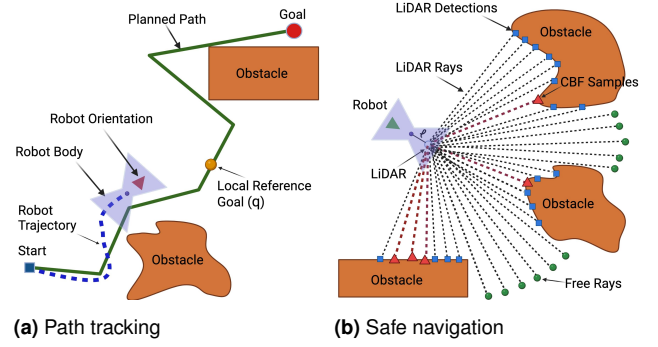
Figure 4. (a) A robot is depicted following a path generated by a motion planning algorithm, and a dynamic local reference goal is highlighted in yellow. (b) The robot senses the environment with a 360-degree LiDAR sensor mounted at $\tilde{\mathbf{x}}$. The CBF samples $\{h_i(\mathbf{x})\}_{i=1}^N$ are the rays with boundary points highlighted as red triangles, and are selected based on the distance from the LiDAR detections to the robot body.

CLF that enables stabilization to a desired goal point, as requested by Assumption 7.2. Then, we discuss the validity of using the SDF as a CBF candidate for ensuring safety. Next, we provide a discussion on how to select CBF samples based on range sensor measurements, which is crucial for the practical implementation of the proposed DR-CBF formulation. To address uncertainties in both localization and sensor data, we also introduce a unified error model that captures how localization errors propagate to LiDAR measurements, impacting the robot's perception of obstacles. By combining the unicycle-specific CLF and the data-driven CBF, we obtain a CLF-DR-CBF QP formulation tailored to the unicycle dynamics, enabling safe and efficient navigation in unknown dynamic environments.

8.1 Unicycle Stabilization to a Goal Position

We design a CLF that enables stabilization of the unicycle dynamics (32) to a desired goal point $\mathbf{q}_G \in \mathbb{R}^2$.

Inspired by İşleyen et al. (2023), for a local reference point $\mathbf{q} \in \mathbb{R}^2$, we define V as follows:

$$V(\mathbf{x}) = \begin{cases} \frac{1}{2}(k_v \|\mathbf{q} - \phi(\mathbf{x})\|^2 + k_\omega \mathrm{atan2}(e_v^\perp, e_v)^2), & \phi(\mathbf{x}) \ne \mathbf{q}, \\ 0, & \phi(\mathbf{x}) = \mathbf{q}, \end{cases} \quad (33)$$

where $\phi(\mathbf{x})$ denotes the current robot position, and $k_v, k_\omega > 0$ are user-specified control gains for linear and angular errors. The error terms e_v and $e_v^\perp$ are defined as:

$$e_v = \begin{bmatrix} \cos \theta \\ \sin \theta \end{bmatrix}^\top (\mathbf{q} - \phi(\mathbf{x})), \; e_v^\perp = \begin{bmatrix} -\sin \theta \\ \cos \theta \end{bmatrix}^\top (\mathbf{q} - \phi(\mathbf{x})). \quad (34)$$

When $\phi(\mathbf{x}) \ne \mathbf{q}$, the first part of $V(\mathbf{x})$ represents the squared Euclidean distance between $\phi(\mathbf{x})$ and $\mathbf{q}$, while the second part quantifies the squared angular alignment

error. Fig. 4a illustrates the unicycle robot tracking a local reference point $\mathbf{q}$ on the planned path.

The time derivative of $V(\mathbf{x})$ is:

$$\dot{V}(\mathbf{x}) = \mathcal{L}_{\mathbf{g}}V(\mathbf{x})\mathbf{u} = \begin{bmatrix} -k_v e_v + k_\omega \dfrac{\text{atan2}(e_v^\perp, e_v)}{\|\mathbf{q} - \phi(\mathbf{x})\|^2} e_v^\perp \\ -k_\omega \text{atan2}(e_v^\perp, e_v) \end{bmatrix}^\top \mathbf{u}. \tag{35}$$

The following result provides a control law that ensures the satisfaction of the CLC in (2).

Lemma 8.1. (Control Lyapunov constraint satisfaction): *Let α_V be a class $\mathcal{K}$ function satisfying $\lim_{r \to 0^+} \frac{\alpha_V(r)}{r} = 0$. For any state $\mathbf{x} \in \mathcal{X}$, the following control law*

$$\mathbf{u}(\mathbf{x}) = \begin{cases} -\alpha_V(V(\mathbf{x})) \dfrac{\mathcal{L}_{\mathbf{g}}V(\mathbf{x})^\top}{\|\mathcal{L}_{\mathbf{g}}V(\mathbf{x})^\top\|^2}, & \phi(\mathbf{x}) \neq \mathbf{q}, \\ \mathbf{0}, & \phi(\mathbf{x}) = \mathbf{q}, \end{cases} \tag{36}$$

ensures that $\mathcal{L}_{\mathbf{f}}V(\mathbf{x}) + \mathcal{L}_{\mathbf{g}}V(\mathbf{x})\mathbf{u}(\mathbf{x}) + \alpha_V(V(\mathbf{x})) \leq 0$ is satisfied. Furthermore, the control law is Lipschitz continuous for $\phi(\mathbf{x}) \neq \mathbf{q}$ and continuous at $\mathbf{q}$.

Proof. First, note that for any $\mathbf{x} \in \mathcal{X}$ such that $\phi(\mathbf{x}) = \mathbf{q}$, we have $V(\mathbf{x}) = 0$ and $\mathcal{L}_{\mathbf{g}}V(\mathbf{x}) = \mathbf{0}$, regardless of the orientation θ. In this case, the control law $\mathbf{u}(\mathbf{x})$ given by (36) is simply $\mathbf{0}$, which trivially satisfies the CLC: $\mathcal{L}_{\mathbf{f}}V(\mathbf{x}) + \mathcal{L}_{\mathbf{g}}V(\mathbf{x})\mathbf{u}(\mathbf{x}) + \alpha_V(V(\mathbf{x})) \leq 0$.

Next, we show that $\mathcal{L}_{\mathbf{g}}V(\mathbf{x}) \neq \mathbf{0}$ for all $\mathbf{x} \in \mathcal{X}$ such that $\phi(\mathbf{x}) \neq \mathbf{q}$. Suppose, by contradiction, that $\mathcal{L}_{\mathbf{g}}V(\mathbf{x}) = \mathbf{0}$. This implies $k_\omega \text{atan2}(e_v^\perp, e_v) = 0$, which means $e_v^\perp = 0$. From (34), we have $e_v^2 + e_v^{\perp 2} = (\mathbf{q} - \phi(\mathbf{x}))^2$. Since $e_v^\perp = 0$ and $\phi(\mathbf{x}) \neq \mathbf{q}$, we obtain $e_v = \pm|\mathbf{q} - \phi(\mathbf{x})| \neq 0$, contradicting the assumption that $\mathcal{L}_{\mathbf{g}}V(\mathbf{x}) = \mathbf{0}$. Hence, $\mathcal{L}_{\mathbf{g}}V(\mathbf{x}) \neq \mathbf{0}$ for all $\mathbf{x} \in \mathcal{X}$ such that $\phi(\mathbf{x}) \neq \mathbf{q}$.

Now, for all $\mathbf{x} \in \mathcal{X}$ such that $\phi(\mathbf{x}) \neq \mathbf{q}$, the control law $\mathbf{u}(\mathbf{x})$ given by (36) is the closed-form solution to the optimization problem:

$$\min_{\mathbf{u} \in \mathbb{R}^m} \|\mathbf{u}\|^2, \tag{37}$$
$$\text{s.t.} \quad \mathcal{L}_{\mathbf{g}}V(\mathbf{x})\mathbf{u} + \alpha_V(V(\mathbf{x})) \leq 0.$$

Since $V(\mathbf{x})$ is smooth with bounded derivatives for $\phi(\mathbf{x}) \neq \mathbf{q}$, it follows that $\mathbf{u}(\mathbf{x})$ is Lipschitz continuous on the set $\{\mathbf{x} \in \mathcal{X} : \phi(\mathbf{x}) \neq \mathbf{q}\}$.

Next, to show continuity on $\mathcal{X}$, we prove that $\lim_{\mathbf{x} \to \mathbf{x}_G} \mathbf{u}(\mathbf{x}) = \mathbf{0}$ for any $\mathbf{x}_G \in \mathcal{X}$ such that $\phi(\mathbf{x}_G) = \mathbf{q}$. Based on (35), we can write

$$\lim_{\mathbf{x} \to \mathbf{x}_G} |\mathbf{u}(\mathbf{x})| = \lim_{\mathbf{x} \to \mathbf{x}_G} \left| \alpha_V(V(\mathbf{x})) \frac{\mathcal{L}_{\mathbf{g}}V(\mathbf{x})^\top}{\|\mathcal{L}_{\mathbf{g}}V(\mathbf{x})^\top\|^2} \right|$$
$$\leq \lim_{\mathbf{x} \to \mathbf{x}_G} \frac{\alpha_V(V(\mathbf{x}))}{V(\mathbf{x})} \cdot \lim_{\mathbf{x} \to \mathbf{x}_G} \left| \frac{V(\mathbf{x})}{\|\mathcal{L}_{\mathbf{g}}V(\mathbf{x})^\top\|} \right| = 0,$$

where the last equality holds because the first limit is zero by the assumption on α_V, and the second limit is bounded since $\frac{V(\mathbf{x})}{|\mathcal{L}_{\mathbf{g}}V(\mathbf{x})^\top|}$ remains bounded as $\mathbf{x} \to \mathbf{x}_G$. Therefore, the control law $\mathbf{u}(\mathbf{x})$ is continuous on $\mathcal{X}$. $\qquad\square$

The control law minimizes the norm of $\mathbf{u}$ while ensuring $\dot{V}(\mathbf{x}) < 0$. Therefore, the function V in (33) together with the control law $\mathbf{u}(\mathbf{x})$ in (36) satisfy Assumption 7.2. Therefore, Lemma 7.3 ensures that the system is guided towards the goal point $\mathbf{q}$. Since the control law $\mathbf{u}(\mathbf{x}) = \mathbf{0}$ whenever $\phi(\mathbf{x}) = \mathbf{q}$, the orientation $\theta(t)$ may converge to any value, depending on the initial conditions. Finally, by Theorem 7.5, the closed-loop dynamics interconnected with the governor dynamics has the unicycle robot track the planned path γ in $\mathbb{R}^2$.

8.2 Signed Distance Function as Control Barrier Function

In unknown and dynamic environments, the precise computation of the barrier function $h(\mathbf{x}, t)$ or the construction of a probabilistic model is challenging. This section focuses on showing that the robot SDF (9) is a valid CBF for (32) under appropriate assumptions.

We make the following assumptions:

Assumption 8.2. (Regularity of Robot Shape): *The robot shape $\mathcal{B}_0$ has a smooth boundary and is compact.*

Based on the robot SDF in (9), we define the candidate CBF as:

$$h(\mathbf{x}, t) = d(\mathcal{B}(\mathbf{x}), \mathcal{O}(t)) := \inf_{\mathbf{q} \in \mathcal{O}(t)} d_{\mathcal{B}_0}\left(\mathbf{R}(\mathbf{x})^\top(\mathbf{q} - \phi(\mathbf{x}))\right), \tag{38}$$

where $\mathcal{O}(t) \subset \mathbb{R}^2$ is the obstacle set at time t, and $\mathbf{R}(\mathbf{x}) \in SO(2)$ is the rotation matrix that describes the unicycle's orientation. By Assumption 8.2, we know that $d_{\mathcal{B}_0}(\cdot)$ is 1-Lipschitz continuous (Fitzpatrick 1980). The presence of the infimum operator in (38) ensures that the candidate CBF $h(\mathbf{x}, t)$ is Lipschitz continuous. However, it is important to note that $h(\mathbf{x}, t)$ is, in general, not smooth due to both the properties of $d_{\mathcal{B}_0}(\cdot)$ and the infimum operation.

Therefore, we review some concepts from nonsmooth analysis to handle the potential non-differentiability of h.

Definition 8.3. (Generalized Gradient): Let $h : \mathcal{X} \times \mathbb{R}_{\geq 0} \to \mathbb{R}$ be Lipschitz continuous near $\mathbf{z} := (\mathbf{x}, t) \in \mathbb{R}^4$, and suppose $\mathcal{Z}$ is any set of Lebesgue measure zero in $\mathbb{R}^4$. The *Clarke generalized gradient* (Clarke 1981) of h at $\mathbf{z}$ is defined as

$$\partial h(\mathbf{z}) = \text{co} \left\{ \lim_{i \to \infty} \nabla h(\mathbf{z}_i) \,\middle|\, \mathbf{z}_i \to \mathbf{z}, \mathbf{z}_i \notin \Omega_h \cup \mathcal{Z} \right\}, \tag{39}$$

where co denotes the convex hull and Ω_h is the set of points where h fails to be differentiable.

Using this definition, we extend the notion of time-varying control barrier functions to functions that are not necessarily differentiable everywhere.

Definition 8.4. (Nonsmooth Time-Varying Control Barrier Function): A Lipschitz continuous and regular function $h : \mathcal{X} \times \mathbb{R}_{\geq 0} \to \mathbb{R}$ is called a *nonsmooth time-varying control barrier function* on a set $\mathcal{H} \subset \mathbb{R}^4$ if there exists an extended class-$\mathcal{K}$ function α such that for all $\mathbf{x} \in \mathcal{X}$ and all $t \geq 0$,

$$\sup_{\mathbf{u} \in \mathcal{U}} \inf_{(\mathbf{y},z) \in \partial h(\mathbf{x},t)} \left\{ \mathbf{y}^\top \left(f(\mathbf{x}) + g(\mathbf{x})\mathbf{u} \right) + z \right\} \geq -\alpha(h(\mathbf{x},t)),$$
(40)

where $\partial h(\mathbf{x}, t)$ is the generalized gradient of h.

Next, we show that $h(\mathbf{x}, t)$ in (38) is a valid time-varying CBF candidate for the dynamics in (32) under the following assumptions.

Assumption 8.5. (Boundedness and Regularity of Obstacles): *We assume that:*

- *The initial obstacle set $\mathcal{O}_0$ is bounded, and the transformations $A_{\mathcal{O}}(t)$ and $\mathbf{b}_{\mathcal{O}}(t)$ in (10) are uniformly bounded over t.*
- *There exists a constant $B > 0$ such that for all $(\mathbf{x}, t)$ and all $(\mathbf{y}, z) \in \partial h(\mathbf{x}, t)$, it holds that $|z| \leq B$.*

The following result establishes the boundedness of the obstacle set $\mathcal{O}(t)$. Its proof is straightforward and therefore omitted.

Lemma 8.6. (Obstacle Set Boundedness): *Under Assumption 8.5, the obstacle set $\mathcal{O}(t)$ is bounded for all $t \geq 0$.*

Next, we show that $h(\mathbf{x}, t)$ is Lipschitz.

Lemma 8.7. (Lipschitzness of Non-smooth CBF): *Under Assumptions 8.2 and 8.5, the function $h(\mathbf{x}, t)$ is Lipschitz in $(\mathbf{x}, t)$.*

Proof. For each fixed $\mathbf{q} \in \mathcal{O}(t)$, we write

$$h_{\mathbf{q}}(\mathbf{x}) = d_{\mathcal{B}_0} \left(\mathbf{R}(\mathbf{x})^\top \left(\mathbf{q} - \phi(\mathbf{x}) \right) \right).$$

Since $\mathbf{R}(\mathbf{x})$ and $\phi(\mathbf{x})$ are continuously differentiable, they are also Lipschitz continuous on the compact set $\mathcal{X}$. Let $L_{\mathbf{R}}$ and L_ϕ be their Lipschitz constants, respectively, so that $\|\mathbf{R}(\mathbf{x}_1) - \mathbf{R}(\mathbf{x}_2)\| \leq L_{\mathbf{R}} \|\mathbf{x}_1 - \mathbf{x}_2\|$ and $\|\phi(\mathbf{x}_1) - \phi(\mathbf{x}_2)\| \leq L_\phi \|\mathbf{x}_1 - \mathbf{x}_2\|$. Now, for any $\mathbf{x}_1, \mathbf{x}_2 \in \mathcal{X}$ and fixed $\mathbf{q} \in \mathcal{O}(t)$, one has

$$|h_{\mathbf{q}}(\mathbf{x}_1) - h_{\mathbf{q}}(\mathbf{x}_2)|$$
$$= \left| d_{\mathcal{B}_0} \left(\mathbf{R}(\mathbf{x}_1)^\top (\mathbf{q} - \phi(\mathbf{x}_1)) \right) - d_{\mathcal{B}_0} \left(\mathbf{R}(\mathbf{x}_2)^\top (\mathbf{q} - \phi(\mathbf{x}_2)) \right) \right|$$
$$\leq \left\| \mathbf{R}(\mathbf{x}_1)^\top (\mathbf{q} - \phi(\mathbf{x}_1)) - \mathbf{R}(\mathbf{x}_2)^\top (\mathbf{q} - \phi(\mathbf{x}_2)) \right\|$$
$$\leq \left\| \left[\mathbf{R}(\mathbf{x}_1)^\top - \mathbf{R}(\mathbf{x}_2)^\top \right] (\mathbf{q} - \phi(\mathbf{x}_1)) \right\| +$$

$$\left\| \mathbf{R}(\mathbf{x}_2)^\top \left(\phi(\mathbf{x}_2) - \phi(\mathbf{x}_1) \right) \right\|.$$
(41)

where the first equality comes from the fact that $d_{\mathcal{B}_0}$ is 1-Lipschitz (Fitzpatrick 1980).

For the first term in (41), we have

$$\left\| \left[\mathbf{R}(\mathbf{x}_1)^\top - \mathbf{R}(\mathbf{x}_2)^\top \right] (\mathbf{q} - \phi(\mathbf{x}_1)) \right\|$$
$$\leq L_{\mathbf{R}} \|\mathbf{x}_1 - \mathbf{x}_2\| \cdot (\|\mathbf{q}\| + \|\phi(\mathbf{x}_1)\|).$$

Since $\mathcal{X}$ is bounded by assumption and $\mathcal{O}(t)$ is also bounded, cf. Lemma 8.6, there exist $M_{\mathbf{q}}, M_\phi > 0$ such that $\|\mathbf{q}\| \leq M_{\mathbf{q}}$ and $\|\phi(\mathbf{x}_1)\| \leq M_\phi$. Therefore:

$$\left\| \left[\mathbf{R}(\mathbf{x}_1)^\top - \mathbf{R}(\mathbf{x}_2)^\top \right] (\mathbf{q} - \phi(\mathbf{x}_1)) \right\|$$
$$\leq L_{\mathbf{R}}(M_{\mathbf{q}} + M_\phi) \|\mathbf{x}_1 - \mathbf{x}_2\|.$$

For the second term in (41), since $\|\mathbf{R}(\mathbf{x})\| = 1$, we have

$$\left\| \mathbf{R}(\mathbf{x}_2)^\top (\phi(\mathbf{x}_2) - \phi(\mathbf{x}_1)) \right\| \leq L_\phi \|\mathbf{x}_1 - \mathbf{x}_2\|.$$

Combining the bounds from the two terms, we conclude

$$|h_{\mathbf{q}}(\mathbf{x}_1) - h_{\mathbf{q}}(\mathbf{x}_2)| \leq L_h \|\mathbf{x}_1 - \mathbf{x}_2\|,$$
(42)

with $L_h = (L_{\mathbf{R}}(M_{\mathbf{q}} + M_\phi) + L_\phi)$. As established in Rockafellar and Wets (2009, Proposition 9.10), the pointwise infimum of a collection of Lipschitz functions is Lipschitz with a constant bounded by the maximum of the individual Lipschitz constants. Therefore, since $h_{\mathbf{q}}(\mathbf{x})$ is uniformly Lipschitz with constant L_h and $\mathcal{O}(t)$ is bounded, the infimum $h(\mathbf{x}, t) = \inf_{\mathbf{q} \in \mathcal{O}(t)} h_{\mathbf{q}}(\mathbf{x})$ is Lipschitz with the same constant L_h.

Furthermore, by Assumption 8.5, $h(\mathbf{x}, t)$ is Lipschitz in t with constant B. $\qquad\square$

Next, our goal is to show that for all $(\mathbf{x}, t) \in \mathcal{X} \times \mathbb{R}_{\geq 0}$, there exists a control input $\mathbf{u} \in \mathcal{U}$ such that the CBF condition (40) holds.

Proposition 8.8. (Sufficient Conditions for Signed Distance Function as a Nonsmooth Time-Varying Control Barrier Function): *Under Assumptions 8.2 and 8.5, the function $h(\mathbf{x}, t)$ defined in (38) satisfies the nonsmooth CBF condition (40) for the unicycle dynamics (32), provided that $\inf_{(\mathbf{0},z) \in \partial h(\mathbf{x},t)} z \geq -\alpha_h(h(\mathbf{x},t))$.*

Proof. Using the unicycle dynamics (32), for any $(\mathbf{y}, z) \in \partial h(\mathbf{x}, t)$, the expression inside the infimum in (40) becomes

$$\mathbf{y}^\top \left(f(\mathbf{x}) + g(\mathbf{x})\mathbf{u} \right) + z = \qquad (43)$$
$$\mathbf{y}^\top \left(\begin{bmatrix} \cos\theta \\ \sin\theta \\ 0 \end{bmatrix} v + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \omega \right) + z = a_v v + a_\omega \omega + z,$$

where

$$a_v = \mathbf{y}^\top \begin{bmatrix} \cos\theta \\ \sin\theta \\ 0 \end{bmatrix}, \quad a_\omega = \mathbf{y}^\top \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}.$$

From Lemma 8.7, $\|y\| \leq L_h$. It follows that $|a_v|$, $|a_\omega| \leq L_h$. We now consider two cases:

Case 1: $(\mathbf{0}, z) \notin \partial h(\mathbf{x}, t)$. In this case, $\mathbf{y} \neq \mathbf{0}$, so at least one of a_v or a_ω is non-zero. Define $\varepsilon = a_v^2 + a_\omega^2 > 0$ and choose

$$v = \frac{a_v}{\varepsilon}\beta, \quad \omega = \frac{a_\omega}{\varepsilon}\beta, \tag{44}$$

where $\beta > 0$ is a scaling factor. Then, since $z \in [-B, B]$ (cf. Assumption 8.5), we get

$$a_v v + a_\omega \omega + z = \left(\frac{a_v^2 + a_\omega^2}{\varepsilon} \right)\beta + z \geq \beta - B. \tag{45}$$

The scaling factor β can be chosen sufficiently large to ensure that the nonsmooth CBF condition (40) is satisfied.

Case 2: $(\mathbf{0}, z) \in \partial h(\mathbf{x}, t)$. In this case, the spatial gradient of h vanishes, and the expression in (43) simplifies to:

$$\mathbf{y}^\top (f(\mathbf{x}) + g(\mathbf{x})\mathbf{u}) + z = z.$$

The nonsmooth CBF condition (40) holds in this case because of the hypotheses of the statement. □

Regarding the requirement of Proposition 8.8, intuitively, when $\mathbf{y} = \mathbf{0}$, the spatial gradient does not provide directional guidance for the robot, and the safety condition depends solely on the temporal evolution of h, i.e., the behavior of the obstacles. The requirement $\inf_{(\mathbf{0},z)\in\partial h(\mathbf{x},t)} z \geq -\alpha_h(h(\mathbf{x},t))$ ensures that the nonsmooth CBF condition is satisfied in this case.

In practice, we need to ensure that the control inputs v and ω satisfy the control constraints $|v| \leq v_{\max}$ and $|\omega| \leq \omega_{\max}$. Based on (44) and (45), this means that the bound B on the partial time derivative of the distance function must be compatible with the control bounds to ensure the inequality (40) holds. In other words, to guarantee safety, the obstacles must move at speeds compatible with the motion capabilities of the robot.

8.3 Sensor-Based CBF Sample Selection

As the robot navigates through unknown and dynamic environments relying on noisy LiDAR measurements, precisely determining the obstacle set $\mathcal{O}(t)$ and thus computing $h(\mathbf{x}, t)$ is not feasible. Therefore, we leverage the distributionally robust CBF formulation from Sec. 6 and obtain samples $\{\boldsymbol{\xi}_i\}_{i=1}^N$, where recall

$$\boldsymbol{\xi} = [\mathbf{F}^\top(\mathbf{x})\nabla_\mathbf{x} h(\mathbf{x}, t), \ \alpha_h(h(\mathbf{x}, t)), \ \frac{\partial h(\mathbf{x}, t)}{\partial t}],$$

directly from the distance measurements $\boldsymbol{\eta}(\phi(\mathbf{x}), \mathbf{R}(\mathbf{x})) = [\eta_1(\phi(\mathbf{x}), \mathbf{R}(\mathbf{x})), \ldots, \eta_K(\phi(\mathbf{x}), \mathbf{R}(\mathbf{x}))]^\top$. Note that the system dynamics $\mathbf{F}(\mathbf{x})$ and $\mathcal{K}_\infty$ function α_h are assumed to be known and deterministic.

In practical robotic systems, sensor measurements, such as LiDAR readings, are collected at discrete intervals determined by the sensor's frequency. These measurements must be processed, along with state estimation and control synthesis, within the time constraints of the control loop to ensure real-time applicability.

To account for state estimation uncertainty, we consider M samples of the robot's estimated pose, denoted as $\{\mathbf{x}^{(j)}\}_{j=1}^M$, where each sample represents a possible true pose of the differential-drive robot given its localization error distribution. For each sample $\mathbf{x}^{(j)}$, we have K corresponding LiDAR measurements in the robot's local frame, $\{\boldsymbol{\eta}_i^{(j)}\}_{i=1}^K$. These measurements are transformed into the global frame using the estimated pose, yielding LiDAR hit $\mathbf{q}_i^{(j)}(t) \in \mathbb{R}^2$ with i-th LiDAR measurement of pose $\mathbf{x}^j$ at time t. By aggregating these transformed measurements, we construct a comprehensive set of LiDAR points in the global frame:

$$\mathcal{P}(t) = \left\{ \mathbf{q}_i^{(j)}(t) \mid i = 1, \ldots, K; \ j = 1, \ldots, N \right\} \subset \mathbb{R}^2, \tag{46}$$

resulting in a total of $N \times K$ points. This set effectively captures the combined uncertainties in both state estimation (localization) and sensor measurements. To account for dynamic obstacles, we may obtain estimates of the time derivatives $\{ \frac{\partial \mathbf{q}_i^j}{\partial t} \}$ using a radar sensor, Doppler LiDAR, or a LiDAR velocity estimation algorithm (Yang et al. 2022).

Since we cannot compute $h(\mathbf{x}, t)$ exactly, we use the available LiDAR points set $\mathcal{P}(t)$ to approximate the barrier function and its gradients. From this aggregated data, we select N samples that minimize the following criterion:

$$\frac{\partial}{\partial t} d_{\mathcal{B}_0} \left(\mathbf{R}(\mathbf{x})^\top (\mathbf{q}_i(t) - \phi(\mathbf{x})) \right) +$$
$$\alpha_h \left(d_{\mathcal{B}_0} \left(\mathbf{R}(\mathbf{x})^\top (\mathbf{q}_i(t) - \phi(\mathbf{x})) \right) \right), \tag{47}$$

for $i = 1, \ldots, K$. Note that the generalized gradient $\partial h_i(\mathbf{x}, t)$ can be computed correspondingly by utilizing the robot SDF $d_{\mathcal{B}_0}$. This criterion effectively identifies obstacle points where the combined effect of the barrier function's rate of change and its current value is most critical, highlighting the samples where the safety constraint, as defined in (5), is closest to being violated.

With this procedure, given each $\mathbf{x} \in \mathcal{X}$, we have available samples $\{\boldsymbol{\xi}_i\}_{i \in [N]}$ for the vector $\boldsymbol{\xi}$.

Remark 8.9. (Sample selection for static environments): If $\frac{\partial h}{\partial t} = 0$, the selection of the N samples reduces to finding the minimum N values of $h_i(\mathbf{x})$, representing the distance of N closest detected obstacle points to the robot. This is illustrated in Fig. 4b. ●

With the sample collection strategy for $\{\boldsymbol{\xi}_i\}_{i \in [N]}$ and the CLF design in (33), we formulate the CLF-DR-CBF optimization problem in (21) to synthesize control inputs for the unicycle robot.

The differential-drive specific CLF is combined with the DR-CBF constraint, which is based on the CBF samples obtained from the range sensor measurements and estimated robot poses. This approach enables safe robot navigation in unknown dynamic environments, while directly utilizing the sensor measurements to ensure safety.

8.4 Application to Other Robotic Systems

Our CLF-DR-CBF formulation can be applied to other control-affine robotic systems described by (1). To apply this formulation to a different system, two key components are required: a CLF $V(\mathbf{x})$ and a CBF $h(\mathbf{x}, t)$:

- **CLF:** $V(\mathbf{x})$ should encode the stability with respect to a desired equilibrium. In many cases, it is more practical to design a CLF for a subset of the state space, such as position, rather than the full state, particularly for underactuated systems like unicycles;
- **CBF:** $h(\mathbf{x}, t)$ defines the safety specifications based on the robot's geometry and its interaction with the environment. The CBF samples can be obtained, for example, from onboard sensory data.

Note that systematic approaches exist for constructing CLFs, including sum-of-squares programming (Jarvis-Wloszek et al. 2003; Dai et al. 2024) and neural network-based approaches (Chang et al. 2019; Boffi et al. 2021). Additionally, specific classes of systems, such as linear systems and differentially flat systems, are more tractable in this context, as discussed in (Mestres et al. 2024b). Once these components are defined, the CLF-DR-CBF formulation can be directly applied to synthesize a distributionally robust safe and stable controller.

For instance, a table-top manipulator evolves in a high-dimensional configuration space, where safety constraints must prevent both obstacle collisions and self-collisions. A CLF can be designed to stabilize the manipulator's end-effector toward a target pose in task space. This typically involves defining a Lyapunov function over task-space error (e.g., pose error in SE(3)) and mapping its gradient back to joint space via the manipulator's Jacobian. Alternatively, inverse kinematics can be used to convert the task-space target into a reference joint configuration, around which a CLF is constructed. The CBF, on the other hand, can be constructed using signed distance functions (Koptev et al. 2023; Li et al. 2024), which encode collision avoidance constraints. CBF samples can similarly be obtained from onboard sensory data (e.g., images) and robot state estimations.
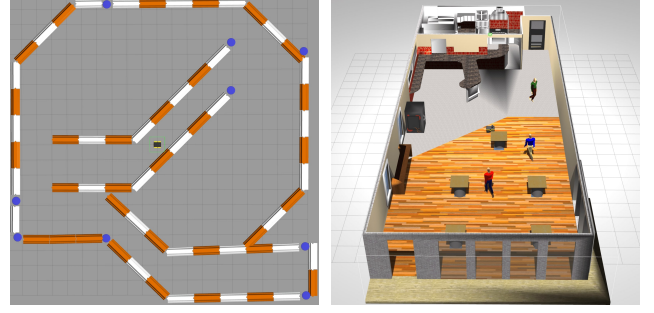
(a) Static environment   **(b)** Dynamic environment

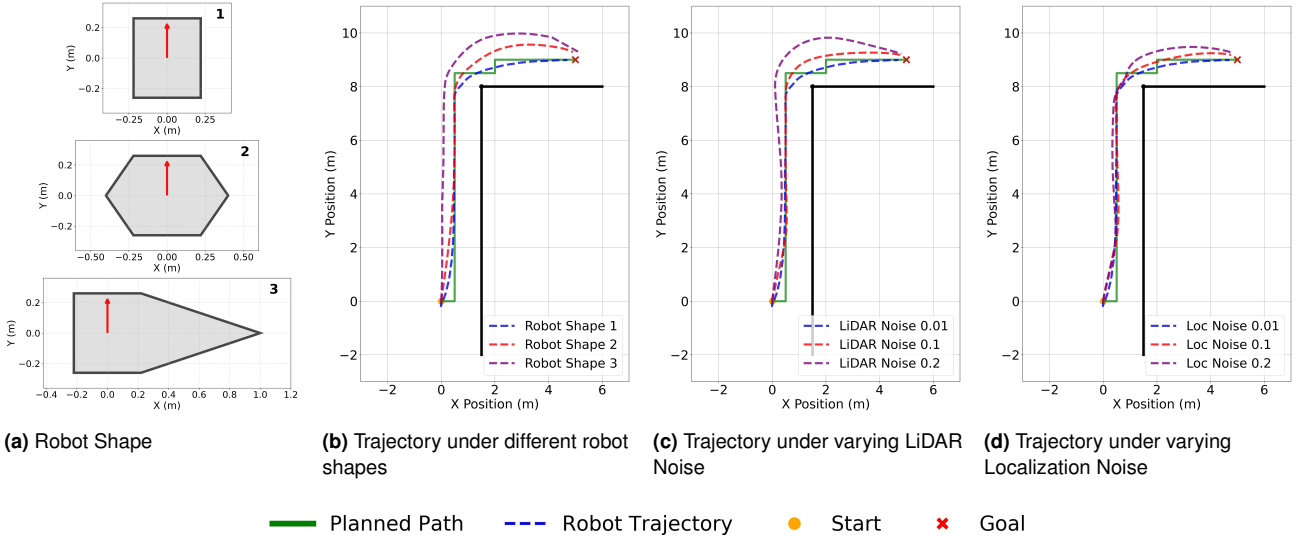**Figure 5.** Simulated environments in Gazebo.

9 Evaluation

In this section, we evaluate our CLF-DR-CBF QP formulation through several simulation and real-world experiments.

We compare our approach with two other safe control strategies, the nominal CLF-CBF QP in (6) and a CLF-Gaussian Process (GP)-CBF second-order cone program (SOCP) (Long et al. 2022). The nominal CLF-CBF QP approach utilizes the closest LiDAR point to define a single CBF $h(\mathbf{x}, t)$ and its gradients at each time step. In the CLF-GP-CBF SOCP method, a real-time GP-SDF model (Wu et al. 2021) of the unknown environment is constructed using LiDAR data, from which the CBF, its gradient, and uncertainty information are determined. While the GP-SDF mapping process contributes to safety by continuously updating the environment representation, it also incurs computational overhead due to the real-time update of the GP-SDF model. For a fair evaluation, we solve each of the optimization programs to generate control signals using the Interior Point Optimizer through the CasADi framework (Andersson et al. 2019).

In the following simulations and experiments, a consistent set of parameters is utilized to ensure reproducibility of the results. The linear velocity is constrained in $[-1.2, 1.2]$ m/s and the angular velocity is limited within $[-1, 1]$ rad/s. The nominal control input $\mathbf{k}(\mathbf{x})$ is set to $[1.2, 0]^{\top}$, directing the robot to move forward at 1.2 m/s. While we use a constant nominal controller, note that our formulation supports more complex, state-dependent nominal controllers. The scaling factor is $\lambda = 50$. Table 1 summarizes other parameter values.

The layout of this section is as follows. Sec. 9.1 presents simulation results and compares with the two baseline approaches in static environments (e.g., Fig. 5a). In Sec. 9.2, we evaluate our approach in dynamic Gazebo environments (Koenig and Howard 2004) with pedestrians, shown in Fig. 5b. Finally, in Sect. 9.3, we test our CLF-DR-CBF QP formulation in dynamic cluttered real-world

(a) Robot Shape

(b) Trajectory under different robot shapes

(c) Trajectory under varying LiDAR Noise

(d) Trajectory under varying Localization Noise

Figure 6. Evaluation of robot navigation performance under varying conditions. The simulations demonstrate the system's behavior across different robot shapes and noise conditions, with (a) showing the tested robot shapes, (b) illustrating the impact of shape variation on navigation, and (c-d) analyzing the effects of sensor and localization uncertainties on trajectory execution. For (c) and (d), Shape 1 from (a) is used as the test case.

Table 1. Simulation and experiment parameters. The class $\mathcal{K}$ function α_V for CLF and the class $\mathcal{K}_\infty$ for CBF are assumed to be linear. The parameters k_v and k_ω are control gains for linear and angular velocities, respectively, ϵ the risk tolerance of the CLF-DR-CBF QP formulation, and N the DR-CBF sample size.

Parameters	α_V	α_h	k_v	k_ω	ϵ	N
Value	1.0	1.5	0.05	0.4	0.1	5

environments. In all results, the A^* algorithm is employed for path planning, operating at a frequency of 5 Hz. Our CLF-DR-CBF QP formulation is used for real-time safe navigation, running at 50 Hz.

9.1 Simulated Static Environments

The first set of simulations aims to validate the robustness and adaptability of the proposed CLF-DR-CBF QP formulation in static environments.

Hypothesis: Our CLF-DR-CBF QP method can ensure safe navigation by dynamically adjusting to varying robot shapes and compensating for sensor and localization noise.

Setup: The robot is tasked to follow the planned path while avoiding obstacles. Gaussian noise with varying standard deviations (σ) is added to the LiDAR measurements, while the localization error is modeled as a Gaussian random vector with varying standard deviation levels as well. Diverse robot shape geometries (Figure 6a) are considered to evaluate the method's adaptability. Unless otherwise noted, the default robot shape is shape 1, representing the original Jackal robot. The trajectories

are analyzed under different conditions to assess safety, adaptability, and smooth navigation.

Results and discussion: Figure 6b demonstrates the trajectories for each robot shape. The results show that our approach dynamically adjusts the robot's path based on its geometry, ensuring safety by maintaining adequate clearance from obstacles while achieving smooth navigation. For symmetric shapes like robot shape 1 (original Jackal robot), the executed path is closer to the obstacle boundary. In contrast, for highly asymmetric shapes like robot shape 3, the executed path deviates more significantly to account for the robot's geometry to ensure safety. Unless otherwise noted, the simulations presented in the following are performed using robot shape 1.

Figures 6c and 6d explore the effects of sensor and localization noise on navigation performance. In Figure 6c, our CLF-DR-CBF QP method successfully compensates for sensor inaccuracies, maintaining safety even at higher noise levels ($\sigma = 0.2$). Similarly, Figure 6d evaluates the impact of localization noise on trajectory execution, where the robot effectively adapts to inaccuracies in state estimation, showcasing the robustness of our distributionally robust formulation.

We next evaluate the performance of our CLF-DR-CBF QP formulation in static environments simulated in Gazebo (Fig. 5a), comparing it with two baseline approaches: Nominal CLF-CBF QP and CLF-GP-CBF SOCP (Long et al. 2022).

Hypothesis: Our CLF-DR-CBF QP formulation ensures robust and safe navigation under varying levels of sensor

Table 2. Performance metrics for static environments under varying LiDAR and localization noise. The metrics include stuck rate and collision rate out of 1000 trials and tracking error (m) (mean $\pm$ std). The stuck rate reflects cases of infeasibility or the robot being trapped in local optima. The robot starts from the origin, and goals are placed at least 10 meters away in Figure 5a.

LiDAR Noise	Localization Noise	Method	Stuck Rate	Collision Rate	Tracking Error (m)
$\sigma = 0.001$	$\sigma = 0.01$	CLF-DR-CBF QP	0.0	0.0	1.33 ± 0.19
		CLF-GP-CBF SOCP	0.2	0.0	1.38 ± 0.31
		Nominal CLF-CBF QP	0.0	0.2	1.26 ± 0.20
	$\sigma = 0.05$	CLF-DR-CBF QP	0.1	0.0	1.88 ± 0.35
		CLF-GP-CBF SOCP	0.3	15.8	2.16 ± 0.47
		Nominal CLF-CBF QP	0.1	15.5	1.92 ± 0.34
$\sigma = 0.05$	$\sigma = 0.01$	CLF-DR-CBF QP	0.0	0.0	1.77 ± 0.38
		CLF-GP-CBF SOCP	25.1	0.2	1.97 ± 0.52
		Nominal CLF-CBF QP	1.7	4.4	1.82 ± 0.41
	$\sigma = 0.05$	CLF-DR-CBF QP	1.8	0.7	2.22 ± 0.57
		CLF-GP-CBF SOCP	31.3	13.9	2.58 ± 0.78
		Nominal CLF-CBF QP	3.3	25.8	2.23 ± 0.59
$\sigma = 0.1$	$\sigma = 0.01$	CLF-DR-CBF QP	1.2	0.0	2.31 ± 0.66
		CLF-GP-CBF SOCP	60.7	0.0	2.55 ± 0.79
		Nominal CLF-CBF QP	4.7	7.7	2.52 ± 0.72
	$\sigma = 0.05$	CLF-DR-CBF QP	6.6	1.8	2.61 ± 0.75
		CLF-GP-CBF SOCP	65.4	22.0	2.78 ± 0.91
		Nominal CLF-CBF QP	8.9	58.3	2.55 ± 0.69

and localization noise. Compared to baseline methods, our approach should achieve lower failure rates (stuck and collision) and demonstrate superior adaptability to noise conditions without compromising computational efficiency.

Setup: The simulations are conducted in a static Gazebo environment where the robot is tasked to achieve a predefined goal while avoiding obstacles. Gaussian noise with standard deviation σ (ranging from 0.001 to 0.1) is added to the LiDAR measurements, and localization noise with σ values up to 0.05 is introduced. For each noise level, 1000 trials are conducted with randomly placed goal locations at least 10 meters away from the robot's starting position. The evaluation metrics include stuck rate, collision rate, and average tracking error (mean $\pm$ std), as summarized in Table 2.

The stuck rate and collision rate together determine the success rate, representing the percentage of trials where the robot successfully reaches the goal without safety violations. The stuck rate captures two failure modes. First, the optimization program may become infeasible due to large uncertainties, particularly in the GP-CBF method, where the GP-SDF map's high variance at large LiDAR noise levels often renders the SOCP problem infeasible. Second, the robot may become trapped in a corner (local minimum), unable to make further progress toward the goal.

Results and discussion: The results in Table 2 highlight the robustness of the CLF-DR-CBF QP method, which consistently achieves low stuck and collision rates, even under high noise levels. In contrast, the GP-CBF

and Nominal CLF-CBF QP methods exhibit significant performance degradation in challenging noise conditions.

As localization noise increases, the two baseline methods are more prone to collisions. This is because their formulations do not explicitly account for uncertainties in the robot state estimation. Our CLF-DR-CBF QP method remains robust by explicitly addressing localization uncertainties in its formulation. Similarly, at higher LiDAR noise levels, the GP-CBF method struggles due to significant variance in the GP-SDF estimation, often rendering its optimization program infeasible. The CLF-DR-CBF QP method, by directly using LiDAR measurements and incorporating distributionally robust constraints, avoids reliance on explicit map construction, exhibiting a higher probability of reliable performance at higher noise levels. Table 3 highlights the computational efficiency of the proposed method. The CLF-DR-CBF QP formulation achieves computation times comparable to the Nominal CLF-CBF QP method while significantly outperforming the GP-CBF method. This advantage stems from the CLF-DR-CBF QP method's direct use of LiDAR measurements without requiring computationally expensive GP map construction.

Overall, these results demonstrate the robustness and efficiency of the proposed CLF-DR-CBF QP formulation. By directly handling noisy sensor data and avoiding reliance on explicit map reconstruction, the method effectively balances computational efficiency and robust safety, making it suitable for real-world applications

Table 3. Computation time comparison between different control approaches (in seconds). The values represent the mean ± standard deviation of the computation time along the robot trajectory. The total computation time for each method is the sum of the GP map training time (if applicable), inference time, and control synthesis solver time. The CLF-DR-CBF QP and Nominal CLF-CBF QP methods have similar total computation time, as they do not require map updates. For these two methods, the inference time refers to processing the LiDAR data as CBF samples and corresponding gradients. The CLF-GP-CBF SOCP method has the highest total computation time due to the additional overhead of GP map training.

Method	Map Training	Inference	Controller Solver	Total Computation Time
CLF-DR-CBF QP	0	0.0002	0.0071 ± 0.0022	0.0073 ± 0.0022
CLF-GP-CBF SOCP	0.0086 ± 0.0031	0.0003	0.0096 ± 0.0028	0.0185 ± 0.0059
Nominal CLF-CBF QP	0	0.0002	0.0064 ± 0.0023	0.0066 ± 0.0023

Table 4. Performance metrics for dynamic environments over 1000 trials. The metrics include success rate, stuck rate, collision rate, and task completion time (mean ± std). The CLF-DR-CBF QP outperforms both baselines in terms of success rate and collision avoidance, demonstrating its robustness in dynamic settings.

Method	Success Rate (%)	Stuck Rate (%)	Collision Rate (%)	Time (s)
CLF-DR-CBF QP	**93.2**	**5.1**	**1.7**	10.7 ± 2.2
CLF-GP-CBF SOCP	60.5	36.3	3.2	13.6 ± 2.9
Nominal CLF-CBF QP	61.7	8.5	29.8	10.1 ± 2.1
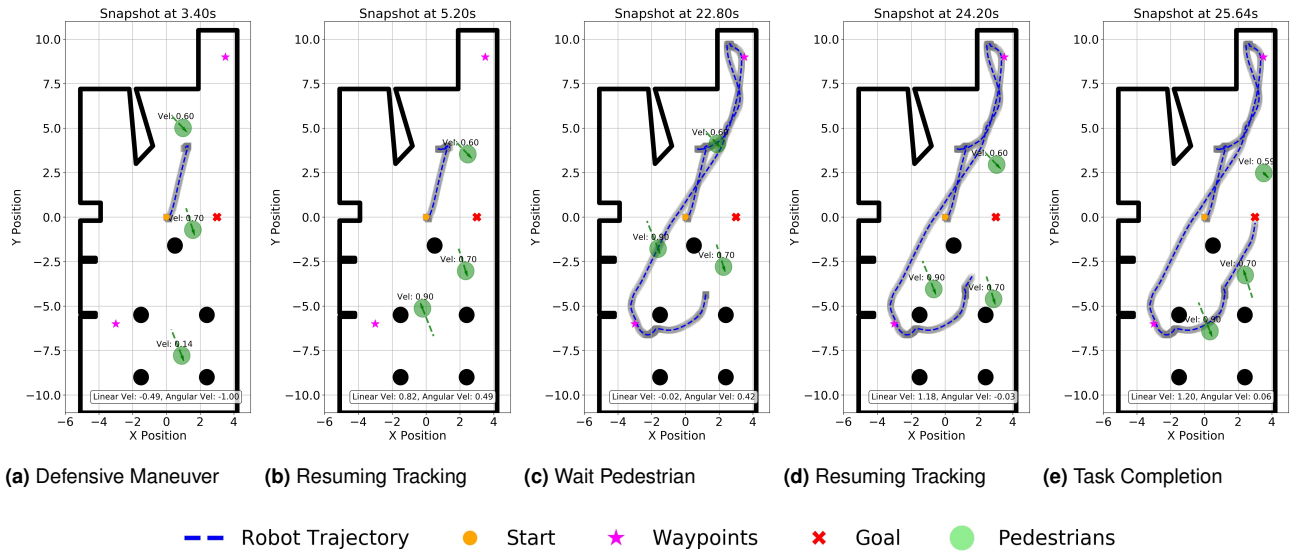
Figure 7. Snapshots showing safe robot navigation in a simulated dynamic environment with three pedestrians, as depicted in Fig. 5b. The ground-truth static environment (e.g., walls, table base) is plotted in black. Each pedestrian is represented by a light green circle, with trajectory over the past second and current velocity also displayed. (a) At $t = 3.4$s, the robot adjusts its trajectory due to an approaching pedestrian, adopting a defensive maneuver by rotating left (-1 rad/s) and moving backwards (-0.49m/s). (b) By $t = 5.2$s, as the pedestrian clears, the robot accelerates forward (0.89m/s) to track its planned path towards the first waypoint. (c) At $t = 22.8$s, facing another pedestrian crossing its planned path, the robot stops (-0.02m/s) to allow the pedestrian to pass. (d) At $t = 24.2$s, the pedestrian has moved away, enabling the robot to resume its course towards the goal. (e) The complete trajectory at $t = 25.6$s shows the robot successfully navigated to two waypoints and the final goal, ensuring safety in a dynamically changing environment.

where sensing noise, localization noise, and computational constraints are significant challenges.

9.2 Simulated Dynamic Environments

The dynamic environment simulations are conducted in Gazebo (cf. Fig. 5b), designed to mimic real-world scenarios with pedestrians modeled using the social force model (Helbing and Molnar 1995; Moussaïd et al. 2010).

Hypothesis: Our CLF-DR-CBF QP approach will outperform baseline methods in handling time-varying constraints under noisy conditions. Specifically, we expect our method to achieve higher success rates, lower collision rates, and efficient task completion times, due to its ability

to incorporate sensor noise and localization uncertainty directly into the control formulation.

Setup: The robot starts at $(0,0)$ with an initial orientation of $0°$, and the goal locations are randomly placed at least 6 meters away. Pedestrians are also randomly positioned in the environment, with velocities bounded by $B = 1$ m/s. Both the static and dynamic elements of the environment are unknown, and the robot relies on noisy LiDAR measurements (Gaussian noise with $\sigma = 0.05$) for collision avoidance. In all simulations below, the A^* planning algorithm operates independently of the pedestrian motion, and the real-time pedestrian avoidance relies on our CLF-DR-CBF QP formulation (or the two baseline approaches). We conducted 1000 trials for each method, measuring metrics such as success rate, stuck rate, collision rate, and average task completion time. Success is defined as reaching the goal while maintaining safety, while a trial is considered stuck if the robot fails to reach the goal within 20 seconds.

Results and discussion: Table 4 summarizes the quantitative results. The proposed CLF-DR-CBF QP approach achieves the highest success rate (93.2%) and the lowest collision rate (1.7%) among the three methods. By directly handling sensor noise through its distributionally robust formulation, the CLF-DR-CBF QP method ensures safety while maintaining efficient task completion times. The GP-CBF method exhibits a lower success rate and higher stuck rate due to its reliance on the GP-SDF map, which becomes computationally expensive and less reliable in dynamic environments. The Nominal CLF-CBF QP approach suffers from the highest collision rate (29.8%), highlighting its limitations in handling dynamic obstacles with sensor noise.

We next present some qualitative results in Fig. 7 in the same dynamic environment. The robot is tasked to sequentially visit two waypoints before reaching a designated goal at $(3,0)$. In Fig. 7a, at $t = 3.4$s, the robot encounters a pedestrian on a collision course with its planned path to the first waypoint at the top right. With our CLF-DR-CBF QP controller, the robot employs a defensive maneuver. This adjustment shows the methodology's capability to anticipate potential hazards and react accordingly.

As the pedestrian clears the immediate area, the robot resumes its path tracking towards the first waypoint by 5.2s (Fig. 7b). This behavior highlights the efficiency of our approach in balancing mission objectives with the need for safety.

The challenge intensifies at $t = 22.8$s when another pedestrian intersects the robot's planned route (Fig. 7c). In response, the robot stops to allow the pedestrian to pass safely. Once the pedestrian has passed, the robot continues
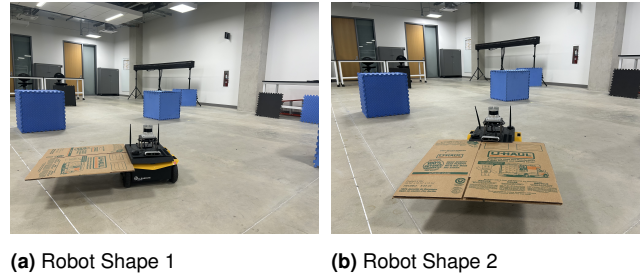

(a) Robot Shape 1   **(b)** Robot Shape 2

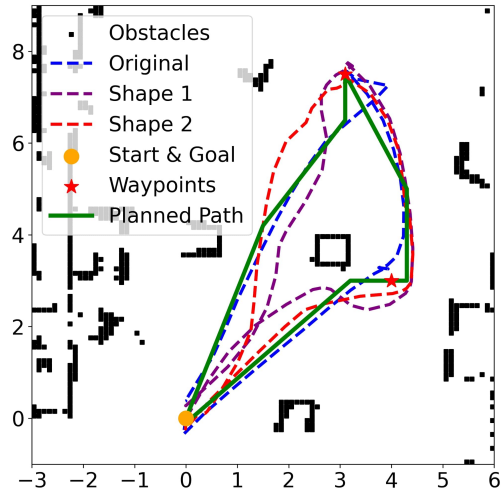**Figure 8.** Robot shapes used in real-world experiments.

**Figure 9.** Comparison of trajectories for the three robot shapes.

its journey towards the goal, as observed at $t = 24.2$s (Fig. 7d).

The successful completion of the task is shown in Fig. 7e, where the robot reaches its final goal after safely navigating past all dynamic obstacles at $t = 25.64$s. This simulation shows the CLF-DR-CBF QP controller's ability for robust path tracking and obstacle avoidance in a dynamic environment.

9.3 Real-World Experiments

We carried out real-world experiments using a differential-drive ClearPath Jackal robot (Fig. 1). The robot was equipped with an Intel i7-9700TE CPU with 32GB RAM, an Ouster OS1-32 LiDAR, and a UM7 9-axis IMU, and a velocity controller accepting linear and angular velocity.

Setup: The experiments took place in a lab environment, designed to test various challenging scenarios. The robot relied solely on noisy LiDAR measurements for navigation. We first tested our approach in an area of the lab with static obstacles, using three different robot shapes: the original shape (Fig. 1a), Shape 1 (Fig. 8a), and Shape 2 (Fig. 8b). For quantitative evaluation, we ran 50 trials per shape in environments populated with randomly placed obstacles

(a) Thin chair legs   **(b)** Narrow passage with pedestrians   **(c)** Pedestrian approaching
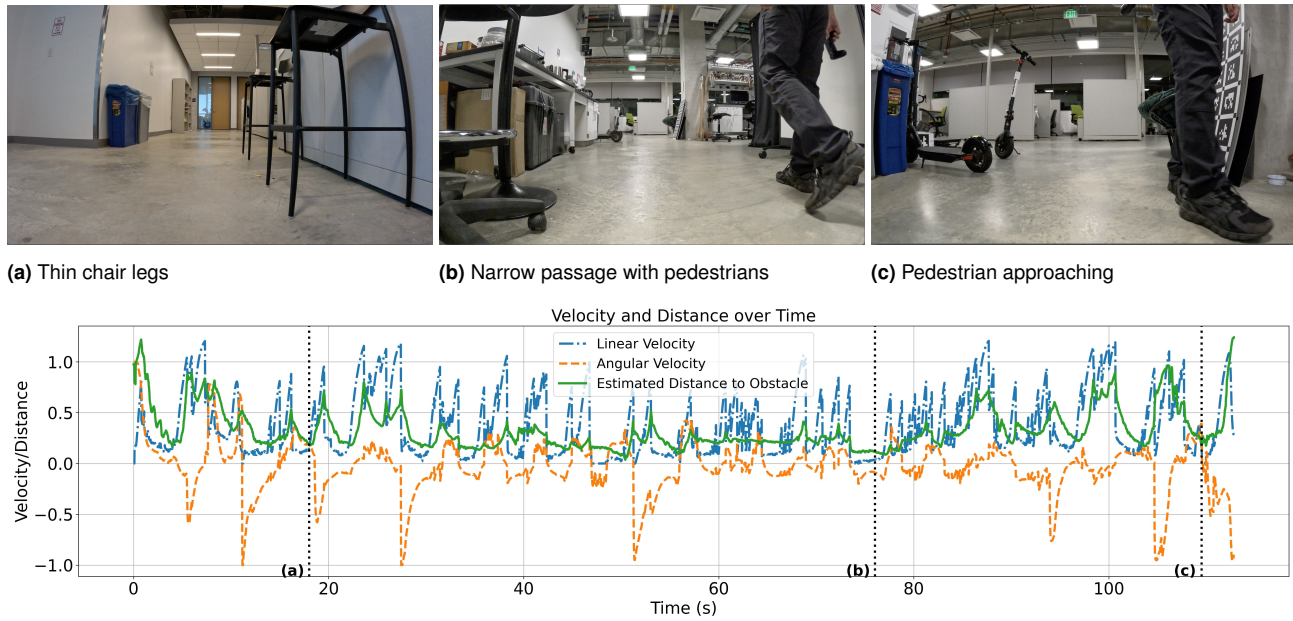
Figure 10. Evaluation of our CLF-DR-CBF QP approach in a real lab environment. The top row illustrates challenging scenarios, including (a) navigating around thin chair legs, (b) passing through a narrow passage with pedestrians, and (c) handling an approaching pedestrian. The bottom plot shows the robot's velocity profile and distance to obstacles over time, with 3 vertical dotted lines marking the specific time instances corresponding to the challenging scenarios in the top row.
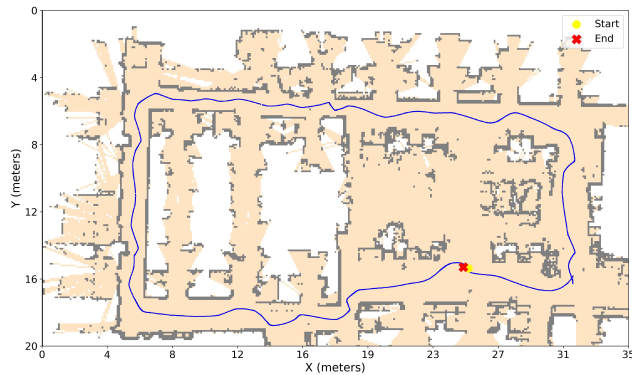
Figure 11. Robot trajectory (blue) and estimated occupancy map (yellow and gray) of the lab environment.

Table 5. Performance metrics for real-world experiments with different robot shapes. Metrics include success rate, stuck rate, and collision rate over 50 trials per shape.

Shape	Success(%)	Stuck (%)	Collision(%)
Original	100	0	0
Shape 1	84	16	0
Shape 2	90	10	0

Table 5 summarizes the results, showing the success, stuck, and collision rates for the three robot shapes under the proposed CLF-DR-CBF QP formulation. Our approach achieved high success rates across all robot shapes, demonstrating its robustness to variations in geometry and dynamic obstacles in the environment. In our evaluation, a robot was considered "stuck" if it did not reach its goal within 60 seconds.

When the robot shape becomes larger (Shapes 1 and 2), it is more likely to get stuck due to limited maneuverability in tight spaces. This is particularly evident for Shape 1, where the asymmetry makes the robot significantly wider on one side, further complicating its ability to bypass obstacles. Additionally, the larger size of Shapes 1 and 2 reduces their ability to navigate around obstacles within the required time, leading to reduced success rates. Despite these challenges, all three shapes achieved a 0% collision rate, demonstrating the effectiveness of our CLF-DR-CBF QP formulation in maintaining safe navigation.

(e.g., cubes and pyramids) and three randomly walking pedestrians.

Results and discussion: Figure 9 shows the trajectories of the robot with the three shapes navigating the same environment by following a pre-planned path. Notably, the planned path was generated assuming the nominal robot shape and did not account for the differences introduced by Shape 1 and Shape 2. Consequently, the original shape demonstrates a minimal deviation from the planned path, whereas Shape 1 exhibits the largest deviations due to its wider, asymmetrical design.

Next, we demonstrated the performance of our CLF-DR-CBF QP formulation using the original robot shape in a full-lab navigation task. The robot successfully handled various real-world challenges, such as thin chair legs, narrow passages with pedestrians, and approaching pedestrians (Fig. 10). In contrast to the CLF-GP-CBF SOCP formulation, which relies on GP regression to construct CBFs and cannot handle dynamic environments effectively, our CLF-DR-CBF QP formulation maintains safety while solely depending on noisy LiDAR measurements. The bottom plot in Fig. 10 presents the distance to the obstacles and the robot's velocity profile over time, highlighting the robot's ability to maintain a safe distance while efficiently navigating towards its goal. For more details, please refer to the accompanying videos on the project webpage†.

Fig. 11 depicts the estimated occupancy map and the executed trajectory using our CLF-DR-CBF QP controller. The robot successfully navigates through the cluttered environment, avoiding both static and dynamic obstacles, and reaches its desired goal position.

10 Conclusion and Future Work

We introduced a novel strategy for ensuring safety of mobile robots navigating autonomously in unknown dynamically changing environments. Our distributionally robust control barrier function formulation leverages sensor measurements and state estimates directly, eliminating the need for precise knowledge of CBFs, which may be slow and inaccurate to obtain in dynamic environments. By combining the DR-CBF with a control Lyapunov function for path tracking, we developed a CLF-DR-CBF quadratic program that enables safe autonomous navigation for robots with control-affine dynamics and arbitrary shape. Our approach underscores the efficiency and effectiveness of employing sensor-based DR-CBF constraints to handle uncertainty in measurements and state estimates. The simulation and experiment results suggest that further exploration into integrating sensor measurements and perception estimates directly into the robot planning and control methods may lead to significant progress in deploying reliable autonomous robot systems in the real world.

While our methodology demonstrates robustness and effectiveness, it has certain limitations. Sections 6 present a general framework for control-affine systems, relying on the assumption that valid CBFs with relative degree 1 exist for these systems. However, identifying such functions can be nontrivial, particularly for complex or high-dimensional systems. To provide a practical demonstration, we employed a signed distance function as a CBF for a differential-drive robot in Section 8 and conducted our evaluations on this specific model.

Future work will aim to extend this methodology to more complex robot systems, such as mobile manipulators and humanoids. Additionally, we plan to explore high-order control barrier functions to address challenges associated with designing valid CBFs, further bridging the gap between theoretical safety assurances and practical deployments in real-world scenarios.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Supplemental material

Supplemental material for this article is available online

References

Abdi H, Raja G and Ghabcheloo R (2023) Safe control using vision-based control barrier function (V-CBF). In: *IEEE International Conference on Robotics and Automation (ICRA)*. pp. 782–788.

Abuaish A, Srinivasan M and Vela PA (2023) Geometry of radial basis neural networks for safety biased approximation of unsafe regions. In: *2023 American Control Conference (ACC)*. pp. 1459–1466.

Agrawal DR and Panagou D (2021) Safe control synthesis via input constrained control barrier functions. In: *2021 60th IEEE Conference on Decision and Control (CDC)*. pp. 6113–6118.

Ames A, Coogan S, Egerstedt M, Notomista G, Sreenath K and Tabuada P (2019) Control barrier functions: Theory and applications. In: *European Control Conference (ECC)*. pp. 3420–3431.

Ames AD, Xu X, Grizzle JW and Tabuada P (2017) Control barrier function based quadratic programs for safety critical systems. *IEEE Transactions on Automatic Control* 62(8): 3861–3876.

Andersson JAE, Gillis J, Horn G, Rawlings JB and Diehl M (2019) CasADi – A software framework for nonlinear optimization and optimal control. *Mathematical Programming Computation* 11(1): 1–36. DOI:10.1007/s12532-018-0139-4.

Aolaritei L, Fochesato M, Lygeros J and Dörfler F (2023) Wasserstein tube MPC with exact uncertainty propagation. In:

---

† https://existentialrobotics.org/DRO_Safe_Navigation/

IEEE Conference on Decision and Control (CDC). pp. 2036–2041.

Arslan O and Koditschek DE (2019) Sensor-based reactive navigation in unknown convex sphere worlds. *The International Journal of Robotics Research* 38(2-3): 196–223.

Artstein Z (1983) Stabilization with relaxed controls. *Nonlinear Analysis: Theory, Methods & Applications* 7: 1163–1173.

Bajcsy A, Bansal S, Bronstein E, Tolani V and Tomlin CJ (2019) An efficient reachability-based framework for provably safe autonomous navigation in unknown environments. In: *2019 IEEE 58th Conference on Decision and Control (CDC)*. pp. 1758–1765.

Boffi N, Tu S, Matni N, Slotine JJ and Sindhwani V (2021) Learning stability certificates from data. In: *Conference on Robot Learning*. PMLR, pp. 1341–1350.

Borenstein J, Koren Y et al. (1991) The vector field histogram-fast obstacle avoidance for mobile robots. *IEEE Transactions on Robotics and Automation* 7(3): 278–288.

Boskos D, Cortés J and Martínez S (2023) Data-driven distributionally robust coverage control by mobile robots. In: *IEEE Conference on Decision and Control (CDC)*. pp. 2030–2035.

Boskos D, Cortés J and Martínez S (2024) High-confidence data-driven ambiguity sets for time-varying linear systems. *IEEE Transactions on Automatic Control* 69(2): 797–812.

Breeden J and Panagou D (2023) Robust control barrier functions under high relative degree and input constraints for satellite trajectories. *Automatica* 155: 111109.

Brito B, Floor B, Ferranti L and Alonso-Mora J (2019) Model predictive contouring control for collision avoidance in unstructured dynamic environments. *IEEE Robotics and Automation Letters* 4(4): 4459–4466.

Chang YC, Roohi N and Gao S (2019) Neural Lyapunov control. In: *Advances in Neural Information Processing Systems*, volume 32.

Chatila R and Laumond J (1985) Position referencing and consistent world modeling for mobile robots. In: *IEEE International Conference on Robotics and Automation (ICRA)*, volume 2. pp. 138–145.

Chen P, Pei J, Lu W and Li M (2022) A deep reinforcement learning based method for real-time path planning and dynamic obstacle avoidance. *Neurocomputing* 497: 64–75.

Choi JJ, Castañeda F, Jung W, Zhang B, Tomlin CJ and Sreenath K (2023) Constraint-guided online data selection for scalable data-driven safety filters in uncertain robotic systems. *arXiv preprint arXiv:2311.13824* .

Chriat A and Sun C (2024) Wasserstein distributionally robust control barrier function using conditional value-at-risk with differentiable convex programming. In: *AIAA SCITECH 2024 Forum*. p. 0725.

Clark A (2019) Control barrier functions for complete and incomplete information stochastic systems. In: *2019 American Control Conference (ACC)*. pp. 2928–2935. DOI: 10.23919/ACC.2019.8814901.

Clarke FH (1981) Generalized gradients of lipschitz functionals. *Advances in Mathematics* 40(1): 52–67.

Corke P, Trevelyan J, Brock O and Khatib O (2000) Elastic strips: A framework for integrated planning and execution. In: *Experimental Robotics VI*. Springer, pp. 329–338.

Coulson J, Lygeros J and Dörfler F (2021) Distributionally robust chance constrained data-enabled predictive control. *IEEE Transactions on Automatic Control* 67(7): 3289–3304.

Dai H, Jiang C, Zhang H and Clark A (2024) Verification and synthesis of compatible control lyapunov and control barrier functions. In: *2024 IEEE 63rd Conference on Decision and Control (CDC)*. pp. 8178–8185. DOI:10.1109/CDC56724. 2024.10885943.

Das E and Burdick JW (2024) Robust control barrier functions using uncertainty estimation with application to mobile robots. *arXiv preprint arXiv:2401.01881* .

Dawson C, Lowenkamp B, Goff D and Fan C (2022) Learning safe, generalizable perception-based hybrid control with certificates. *IEEE Robotics and Automation Letters* 7(2): 1904–1911.

Daş E and Murray RM (2022) Robust safe control synthesis with disturbance observer-based control barrier functions. In: *IEEE Conference on Decision and Control (CDC)*. pp. 5566–5573.

De Lima DA and Pereira GAS (2013) Navigation of an autonomous car using vector fields and the dynamic window approach. *Journal of Control, Automation and Electrical Systems* 24: 106–116.

Desai M and Ghaffari A (2022) CLF-CBF based quadratic programs for safe motion control of nonholonomic mobile robots in presence of moving obstacles. In: *IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*. pp. 16–21.

Dhiman V, Khojasteh MJ, Franceschetti M and Atanasov N (2023) Control barriers in bayesian learning of system dynamics. *IEEE Transactions on Automatic Control* 68(1): 214–229.

Dijkstra EW (1959) A note on two problems in connexion with graphs. *Numerische Mathematik* 1: 269–271.

Dixit A, Lindemann L, Wei SX, Cleaveland M, Pappas GJ and Burdick JW (2023) Adaptive conformal prediction for motion planning among dynamic agents. In: *Learning for Dynamics and Control Conference*. PMLR, pp. 300–314.

Djuric PM, Kotecha JH, Zhang J, Huang Y, Ghirmai T, Bugallo MF and Miguez J (2003) Particle filtering. *IEEE signal processing magazine* 20(5): 19–38.

Dudek G, Jenkin M, Milios E and Wilkes D (1978) Robotic exploration as graph construction. *J. Comput., vol* 7(3).

Emam Y, Glotfelter P, Wilson S, Notomista G and Egerstedt M (2022) Data-driven robust barrier functions for safe, long-term operation. *IEEE Transactions on Robotics* 38(3): 1671–1685.

Esfahani PM and Kuhn D (2018) Data-driven distributionally robust optimization using the Wasserstein metric: performance guarantees and tractable reformulations. *Mathematical Programming* 171: 115–166.

Everett M, Chen YF and How JP (2021) Collision avoidance in pedestrian-rich environments with deep reinforcement learning. *IEEE Access* 9: 10357–10377.

Fitzpatrick S (1980) Metric projections and the differentiability of distance functions. *Bulletin of the Australian Mathematical Society* 22(2): 291–312.

Fox D, Burgard W and Thrun S (1997) The dynamic window approach to collision avoidance. *IEEE Robotics Autom. Mag.* 4: 23–33.

Frazzoli E, Dahleh MA and Feron E (2002) Real-time motion planning for agile autonomous vehicles. *Journal of guidance, control, and dynamics* 25(1): 116–129.

Garone E and Nicotra MM (2015) Explicit reference governor for constrained nonlinear systems. *IEEE Transactions on Automatic Control* 61(5): 1379–1384.

Grandia R, Taylor AJ, Ames AD and Hutter M (2021) Multi-layered safety for legged robots via control barrier functions and model predictive control. In: *IEEE International Conference on Robotics and Automation (ICRA)*. pp. 8352–8358.

Grisetti G, Kümmerle R, Stachniss C and Burgard W (2010) A tutorial on graph-based slam. *IEEE Intelligent Transportation Systems Magazine* 2(4): 31–43.

Hakobyan A and Yang I (2022) Distributionally robust risk map for learning-based motion planning and control: A semidefinite programming approach. *IEEE Transactions on Robotics* 39(1): 718–737.

Hamdipoor V, Meskin N and Cassandras CG (2023) Safe control synthesis using environmentally robust control barrier functions. *European Journal of Control* 74: 100840. 2023 European Control Conference Special Issue.

Han L, Gao F, Zhou B and Shen S (2019) Fiesta: Fast incremental euclidean distance fields for online motion planning of aerial robots. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. pp. 4423–4430.

Hart PE, Nilsson NJ and Raphael B (1968) A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics* 4(2): 100–107.

Helbing D and Molnar P (1995) Social force model for pedestrian dynamics. *Physical Review E* 51(5): 4282.

Herbert SL, Chen M, Han S, Bansal S, Fisac JF and Tomlin CJ (2017) FaSTrack: A modular framework for fast and guaranteed safe motion planning. In: *IEEE Conference on Decision and Control (CDC)*. pp. 1517–1522.

Hota AR, Cherukuri AK and Lygeros J (2019) Data-driven chance constrained optimization under Wasserstein ambiguity sets. In: *American Control Conference (ACC)*. pp. 1501–1506.

Huang JK and Grizzle JW (2023) Efficient anytime clf reactive planning system for a bipedal robot on undulating terrain. *IEEE Transactions on Robotics* 39(3): 2093–2110.

İşleyen A, van de Wouw N and Arslan Ö (2023) Feedback motion prediction for safe unicycle robot navigation. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. pp. 10511–10518.

Jarvis-Wloszek Z, Feeley R, Tan W, Sun K and Packard A (2003) Some controls applications of sum of squares programming. In: *42nd IEEE International Conference on Decision and Control*, volume 5. pp. 4676–4681. DOI:10.1109/CDC.2003.1272309.

Jiang R and Guan Y (2016) Data-driven chance constrained stochastic program. *Mathematical Programming* 158: 291–327.

Kalman RE (1960) A new approach to linear filtering and prediction problems. *ASME Journal of Basic Engineering* 82: 35–45.

Keyumarsi S, Atman MWS and Gusrialdi A (2024) Lidar-based online control barrier function synthesis for safe navigation in unknown environments. *IEEE Robotics and Automation Letters* 9(2): 1043–1050.

Khalil HK (2002) *Control of nonlinear systems*. Prentice Hall, New York, NY.

Khatib O (1986) Real-time obstacle avoidance for manipulators and mobile robots. *The International Journal of Robotics Research* 5(1): 90–98.

Khazoom C, Gonzalez-Diaz D, Ding Y and Kim S (2022) Humanoid self-collision avoidance using whole-body control with control barrier functions. In: *IEEE International Conference on Humanoid Robots (Humanoids)*. pp. 558–565.

Koenig N and Howard A (2004) Design and use paradigms for gazebo, an open-source multi-robot simulator. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, volume 3. pp. 2149–2154.

Kohlbrecher S, von Stryk O, Meyer J and Klingauf U (2011) A flexible and scalable slam system with full 3d motion estimation. In: *IEEE International Symposium on Safety, Security, and Rescue Robotics*. pp. 155–160.

Kondo K, Figueroa R, Rached J, Tordesillas J, Lusk PC and How JP (2023) Robust mader: Decentralized multiagent trajectory planner robust to communication delay in dynamic environments. *IEEE Robotics and Automation Letters* .

Koptev M, Figueroa N and Billard A (2023) Neural joint space implicit signed distance functions for reactive robot manipulator control. *IEEE Robotics and Automation Letters* 8(2): 480–487. DOI:10.1109/LRA.2022.3227860.

Lathrop P, Boardman B and Martínez S (2021) Distributionally safe path planning: Wasserstein safe RRT. *IEEE Robotics and Automation Letters* 7(1): 430–437.

Li J, Liu Q, Jin W, Qin J and Hirche S (2023) Robust safe learning and control in an unknown environment: An uncertainty-separated control barrier function approach. *IEEE Robotics and Automation Letters* 8(10): 6539–6546.

Li Y, Zhang Y, Razmjoo A and Calinon S (2024) Representing robot geometry as distance fields: Applications to whole-body manipulation. In: *Proc. IEEE Intl Conf. on Robotics and Automation (ICRA)*. pp. 15351–15357.

Li Z, Arslan Ö and Atanasov N (2020) Fast and safe path-following control using a state-dependent directional metric. In: *IEEE International Conference on Robotics and Automation (ICRA)*. pp. 6176–6182.

Lindemann L, Cleaveland M, Shim G and Pappas GJ (2023) Safe planning in dynamic environments using conformal prediction. *IEEE Robotics and Automation Letters* 8(8): 5116–5123.

Liu J, Adu CE, Lymburner L, Kaushik V, Trang L and Vasudevan R (2023a) Radius: Risk-aware, real-time, reachability-based motion planning. In: *Robotics: Science and Systems (RSS)*.

Liu J, Li M, Huang JK and Grizzle JW (2023b) Realtime safety control for bipedal robots to avoid multiple obstacles via CLF-CBF constraints. *arXiv preprint arXiv:2301.01906* .

Liu S, Liu C and Dolan J (2023c) Safe control under input limits with neural control barrier functions. In: *Conference on Robot Learning*. PMLR, pp. 1970–1980.

Long K, Cortés J and Atanasov N (2024) Distributionally robust policy and lyapunov-certificate learning. *IEEE Open Journal of Control Systems* 3: 375–388.

Long K, Dhiman V, Leok M, Cortés J and Atanasov N (2022) Safe control synthesis with uncertain dynamics and constraints. *IEEE Robotics and Automation Letters* 7(3): 7295–7302.

Long K, Qian C, Cortés J and Atanasov N (2021) Learning barrier functions with memory for robust safe navigation. *IEEE Robotics and Automation Letters* 6(3): 4931–4938.

Long K, Yi Y, Cortes J and Atanasov N (2023a) Distributionally robust Lyapunov function search under uncertainty. In: *Learning for Dynamics and Control Conference*. PMLR, pp. 864–877.

Long K, Yi Y, Cortés J and Atanasov N (2023b) Safe and stable control synthesis for uncertain system models via distributionally robust optimization. In: *American Control Conference (ACC)*. pp. 4651–4658.

Lozano-Perez (1983) Spatial planning: A configuration space approach. *IEEE Transactions on Computers* C-32(2): 108–120.

Majd K, Yaghoubi S, Yamaguchi T, Hoxha B, Prokhorov D and Fainekos G (2021) Safe navigation in human occupied environments using sampling and control barrier functions. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. pp. 5794–5800.

Mestres P, Allibhoy A and Cortés J (2023) Robinson's counterexample and regularity properties of optimization-based controllers. *arXiv preprint arXiv:2311.13167* .

Mestres P, Long K, Atanasov N and Cortés J (2024a) Feasibility analysis and regularity characterization of distributionally robust safe stabilizing controllers. *IEEE Control Systems Letters* 8: 91–96.

Mestres P, Nieto-Granda C and Cortés J (2024b) Safe and dynamically-feasible motion planning using control lyapunov and barrier functions. *arXiv preprint arXiv:2410.08364* .

Moussaïd M, Perozo N, Garnier S, Helbing D and Theraulaz G (2010) The walking behaviour of pedestrian social groups and its impact on crowd dynamics. *PloS one* 5(4): e10047.

Nemirovski A and Shapiro A (2006) Convex approximations of chance constrained programs. *SIAM J. Optim.* 17: 969–996.

Oleynikova H, Taylor Z, Fehr M, Siegwart R and Nieto J (2017) Voxblox: Incremental 3d Euclidean signed distance fields for on-board MAV planning. In: *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. pp. 1366–1373.

Park JJ, Florence P, Straub J, Newcombe R and Lovegrove S (2019) DeepSDF: Learning Continuous Signed Distance Functions for Shape Representation. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. pp. 165–174.

Parwana H, Mustafa A and Panagou D (2022) Trust-based rate-tunable control barrier functions for non-cooperative multi-agent systems. In: *2022 IEEE 61st Conference on Decision and Control (CDC)*. pp. 2222–2229.

Pfeiffer M, Shukla S, Turchetta M, Cadena C, Krause A, Siegwart R and Nieto J (2018) Reinforced imitation: Sample efficient deep reinforcement learning for mapless navigation by leveraging prior demonstrations. *IEEE Robotics and Automation Letters* 3(4): 4423–4430.

Prajna S and Jadbabaie A (2004) Safety verification of hybrid systems using barrier certificates. In: *International Workshop on Hybrid Systems: Computation and Control*. Springer, pp. 477–492.

Ren AZ and Majumdar A (2022) Distributionally robust policy learning via adversarial environment generation. *IEEE Robotics and Automation Letters* 7(2): 1379–1386.

Rimon E and Koditschek DE (1992) Exact robot navigation using artificial potential functions. *IEEE Transactions on Robotics and Automation* 8(5): 501–518.

Rockafellar RT and Uryasev S (2000) Optimization of conditional value-at-risk. *Journal of Risk* 2: 21–41.

Rockafellar RT and Wets RJB (2009) *Variational analysis*, volume 317. Springer Science & Business Media.

Ryu K and Mehr N (2024) Integrating predictive motion uncertainties with distributionally robust risk-aware control for safe robot navigation in crowds. In: *2024 IEEE*

International Conference on Robotics and Automation (ICRA). pp. 2410–2417.

Schaefer S, Leung K, Ivanovic B and Pavone M (2021) Leveraging neural network gradients within trajectory optimization for proactive human-robot interactions. In: *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, pp. 9673–9679.

Shafer G and Vovk V (2008) A tutorial on conformal prediction. *Journal of Machine Learning Research* 9(3).

Sontag ED (1989) A 'universal' construction of Artstein's theorem on nonlinear stabilization. *Systems & Control Letters* 13(2): 117–123.

Summers T (2018) Distributionally robust sampling-based motion planning under uncertainty. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. pp. 6518–6523.

Van Parys BPG, Kuhn D, Goulart PJ and Morari M (2016) Distributionally robust control of constrained stochastic systems. *IEEE Transactions on Automatic Control* 61(2): 430–442.

Wang Y and Xu X (2023) Disturbance observer-based robust control barrier functions. In: *American Control Conference (ACC)*. pp. 3681–3687.

Wu L, Lee KMB, Le Gentil C and Vidal-Calleja T (2023) Log-gpis-mop: A unified representation for mapping, odometry, and planning. *IEEE Transactions on Robotics* 39(5): 4078–4094.

Wu L, Lee KMB, Liu L and Vidal-Calleja T (2021) Faithful Euclidean distance field from log-gaussian process implicit surfaces. *IEEE Robotics and Automation Letters* 6(2): 2461–2468.

Xiao W, Wang TH, Chahine M, Amini A, Hasani R and Rus D (2022) Differentiable control barrier functions for vision-based end-to-end autonomous driving. *arXiv preprint arXiv:2203.02401* .

Xie W (2021) On distributionally robust chance constrained programs with Wasserstein distance. *Math. Program.* 186: 115–155.

Yang S, Pappas GJ, Mangharam R and Lindemann L (2023) Safe perception-based control under stochastic sensor uncertainty using conformal prediction. In: *IEEE Conference on Decision and Control (CDC)*. pp. 6072–6078.

Yang W, Gong Z, Huang B and Hong X (2022) Lidar with velocity: Correcting moving objects point cloud distortion from oscillating scanning lidars by fusion with camera. *IEEE Robotics and Automation Letters* 7(3): 8241–8248.

Yu H, Hirayama C, Yu C, Herbert S and Gao S (2023) Sequential neural barriers for scalable dynamic obstacle avoidance. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. pp. 11241–11248.

Zhang S, Garg K and Fan C (2023) Neural graph control barrier functions guided distributed collision-avoidance multi-agent control. In: *Conference on Robot Learning (CoRL)*. PMLR, pp. 2373–2392.

Zhang Y, Tian G, Wen L, Yao X, Zhang L, Bing Z, He W and Knoll A (2024) Online efficient safety-critical control for mobile robots in unknown dynamic multi-obstacle environments. *arXiv preprint arXiv:2402.16449* .

Zhao Y, Yu X, Deshmukh JV and Lindemann L (2024) Conformal predictive programming for chance constrained optimization. *arXiv preprint arXiv:2402.07407* .