

Geometry-Aware Control Barrier Functions for Collision Avoidance via Bernstein Polynomial Approximations

Siwon Jo¹, Yanze Zhang², Yupeng Yang³, and Wenhao Luo²

Abstract—Safe navigation often relies on well-defined conditions based on the shape of robots and obstacles, and can be challenging when they have irregular geometries. While Control Barrier Functions (CBFs) offer an efficient mechanism to enforce safe set forward invariance, common shape surrogates (e.g., spheres or super-ellipsoids) either are overly conservative in unstructured scenes or require many local primitives, which inflates constraint counts and degrades real-time performance. In this paper, we introduce a novel geometry-aware Control Barrier Function (CBF) based on Bernstein–Polynomial Signed Distance Fields (BP-SDFs). It provides a unified way to represent the obstacles and robots, so as to represent the barrier function with a unified minimum distance. Benefiting from the differentiability of the Bernstein polynomials, one can easily enforce the control constraints in a closed loop. We validate the method’s efficiency and performance to guarantee safety in single-robot navigation and heterogeneous multi-robot collision avoidance via simulations under different environments.

I. INTRODUCTION

Safe navigation is both essential and inherently challenging in most robotics systems [1]–[4], especially when robots and obstacles exhibit irregular geometries. The specification of safe conditions depends on how shapes interact across poses rather than on point-to-point (e.g., center-to-center) distance [5], [6]. On the other hand, enforcing safe motions needs to consider the dynamics and control limits of the robots, as well as their impact on the evolving safe conditions. Thus, it is critical to characterize geometry-aware safety that reflects the true risk of contact and design controllers that can effectively enforce safety during operation.

Control Barrier Functions (CBFs) [7] have emerged as an efficient tool for certifying safety, which are typically implemented as safety filters that modify the nominal control inputs to ensure the robot is always within the safe set. However, traditional CBFs-related approaches use spheres [8] or super-ellipsoids [9] to define the safe state space by benefiting from their simplicity and differentiability, which would not work well for unstructured environments. In particular, in unstructured environments, existing methods either inflate the geometry of robots and obstacles [10] or approximate obstacle boundaries with sets of small spheres [11],

[12]. Nevertheless, the former introduces overly conservative behaviors, and the latter increases the number of constraints, which in turn impacts the efficiency of the solution.

In this context, the work in [13] builds non-conservative CBFs by computing differentiable signed distances (and penetration depth) between a polytopic robot and polytopic obstacles via convex programs in the Minkowski-difference space, which supports geometry-exact safety constraints within a real-time Control Lyapunov Function – Control Barrier Function (CLF–CBF) control framework. However, the safety is enforced pairwise per obstacle in static multi-obstacle scenarios, which may increase infeasibility under tight bounds and require significant computational overhead for complex environments.

Existing works have also explored learning unified CBFs for obstacles in real-time, but they often struggle with computational inefficiency, which limits their application in the real world. For example, work in [14] uses Gaussian Process (GP) to model the CBFs. However, maintaining and updating the GP posterior in the control loop scales cubically with the number of data points and carries sizable constants (e.g., kernel factorizations, hyperparameter tuning), which makes high-rate control difficult as data accumulate. Alternatively, [15] adopts an implicit neural representation of the CBF with backpropagated gradients but only considers single-robot cases, leaving inter-robot collision avoidance as an open challenge.

More recently, the authors in [16] used Bernstein polynomials to approximate signed distance fields (BP-SDF), thereby enabling the representation of safe space. While this advances geometric modeling, the method primarily focuses on static safety representation and does not explicitly show how the learned safe representation can be converted into *control constraints* with closed-loop safety guarantees. As a result, it is not directly applicable to controller synthesis in dynamic scenarios and multi-robot settings, where multiple moving robots may interact closely with one another. Building on this work, we propose to extend the BP-SDF to define novel CBFs that convert pose-dependent geometry into certifiable safety constraints directly governing collision-free motions for the robots in real time. Specifically, we represent both the robot and the workspace obstacles using a BP-SDF in their body frames and evaluate them in the world frame via rigid transformations. Then, the Karush–Kuhn–Tucker (KKT) conditions are analyzed to find the minimum distance that is used to define our geometry-aware CBFs. Benefiting from the differentiability of the Bernstein polynomials, we are able to explicitly define the admissible control space,

*This work was supported in part by the U.S. National Science Foundation under Grants 2530297 and 2528997.

¹Siwon Jo is with the GRASP Laboratory, University of Pennsylvania, Philadelphia, PA, USA. Email: jol16@seas.upenn.edu

²Yanze Zhang and Wenhao Luo are with the Department of Computer Science, University of Illinois Chicago, Chicago, IL, USA. Email:{yzhan361, wenhao}@uic.edu

³Yupeng Yang is with the Department of Computer Science, University of North Carolina at Charlotte, Charlotte, NC, USA. Email: yyang52@charlotte.edu

leading to collision-free movements of the robots with irregular shapes. This yields forward-invariant safety with minimal deviation from nominal commands and scales to dynamic, multi-robot scenes. Our **contributions** can be summarized as:

- We introduce a novel unified barrier function representation with a single SDF for multiple obstacles using Bernstein polynomials. This approach provides a common template for deriving geometry-aware control constraints for safe navigation in unstructured environments.
- We define a novel notion of *Geometry-aware Control Barrier Functions* built on Bernstein polynomial representations and explicitly derive control constraints from the proposed CBFs for arbitrary smooth convex SDF level sets, which yields closed-loop safety certificates.
- We include results from the experiments for multi-robot collision avoidance and multi-obstacle robot navigation that verify the effectiveness of the framework.

II. PRELIMINARIES

Considering a team of N robots moving in the d -dimensional space \mathbb{R}^d , the position of each robot is denoted as $\mathbf{x}_i \in \mathcal{X}_i \subset \mathbb{R}^d, \forall i \in \{1, \dots, N\}$, with dynamics governed by:

$$\dot{\mathbf{x}}_i = f_i(\mathbf{x}_i) + g_i(\mathbf{x}_i)\mathbf{u}_i \quad (1)$$

where $f_i : \mathbb{R}^d \rightarrow \mathbb{R}^d$ and $g_i : \mathbb{R}^d \rightarrow \mathbb{R}^{d \times q}$ are locally Lipschitz continuous and $\mathbf{u}_i \in \mathbb{R}^q$ is the control input for robot i .

A. Control Barrier Functions

In this section, we briefly review the concept of Control Barrier Functions (CBFs) [7]. Given the control-affine system in Eq. (1), a desired safe set for the state of robot i can be specified as the 0-superlevel set of a continuously differentiable function $h : \mathbb{R}^d \rightarrow \mathbb{R}$,

$$\mathcal{H}_i = \{\mathbf{x}_i \in \mathbb{R}^d : h(\mathbf{x}_i) \geq 0\}. \quad (2)$$

CBFs have been used to design control constraints that enforce the forward invariance of the safe set \mathcal{H}_i for robot i , i.e., if the system starts within \mathcal{H}_i , it remains there for all future time under satisfying controllers. The key result can be summarized as follows.

Lemma 1. (Summarized from [7]) *Given a dynamical system affine in control and a desired set \mathcal{H}_i as the 0-superlevel set of a **continuously differentiable** function $h(\mathbf{x}_i) : \mathcal{X}_i \rightarrow \mathbb{R}$, the function h is called a control barrier function, if there exists an extended class- \mathcal{K} function¹ $\kappa(\cdot)$ such that $\sup_{\mathbf{u}_i \in \mathcal{U}_i} \{\dot{h}(\mathbf{x}_i, \mathbf{u}_i)\} \geq -\kappa(h(\mathbf{x}_i))$ for all $\mathbf{x}_i \in \mathcal{X}_i$. The admissible control space for any Lipschitz continuous controller $\mathbf{u}_i \in \mathcal{U}_i$ rendering \mathcal{H}_i forward invariant (i.e., keeping the system state \mathbf{x}_i in \mathcal{H}_i over time) thus becomes:*

$$\mathcal{B}(\mathbf{x}_i) = \{\mathbf{u}_i \in \mathcal{U}_i | L_f h(\mathbf{x}_i) + L_g h(\mathbf{x}_i)\mathbf{u}_i + \kappa(h(\mathbf{x}_i)) \geq 0\}$$

¹In the rest of this letter, we select the particular choice of $\kappa(h(\mathbf{x}_i)) = \gamma h(\mathbf{x}_i)$ with γ as a user-defined parameter [7].

where L_f and L_g denote the Lie derivatives along the vector fields f and g , respectively.

Existing work [8] leverages Lemma 1 to enforce pairwise distance constraints between robots or between robots and obstacles. However, directly encoding such constraints becomes cumbersome in environments with complex or irregular geometries. Note that when multiple CBF constraints are present, the safety guarantee relies on the composition of CBFs. Readers are referred to [9], [17] for detailed discussion. For systems with higher-order dynamics, the corresponding extension can be found in [18]. This motivates the use of a more general geometric representation of safe sets, which leads us to the notion of Signed Distance Fields (SDF) and Bernstein polynomials discussed next.

B. Signed Distance Fields Representation using Bernstein Polynomials

Existing work has adopted *Signed Distance Field* (SDF) [19] as a general representation of safe set of states. This provides an implicit and differentiable description of arbitrary geometries summarized below.

Definition 2. (Summarized from [19]) *Signed Distance Field* For a set $\Omega \subset \mathbb{R}^d$, the signed distance field is defined as

$$S_\Omega(\mathbf{p}) = \pm \inf_{\mathbf{p}' \in \partial\Omega} \|\mathbf{p} - \mathbf{p}'\|_2, \quad (3)$$

where \mathbf{p} is a point in the Euclidean space, with $S_\Omega(\mathbf{p}) < 0$ if $\mathbf{p} \in \Omega$, $S_\Omega(\mathbf{p}) = 0$ if $\mathbf{p} \in \partial\Omega$, and $S_\Omega(\mathbf{p}) > 0$ otherwise.

It is noted that directly storing or evaluating $S_\Omega(\mathbf{p})$ for arbitrary shapes can be computationally expensive, and in many cases the complex geometry of Ω makes an explicit analytic form intractable. To overcome this challenge, existing work has employed *Bernstein polynomial* (BP) approximations of SDFs [16], which yield a compact and smooth parameterization suitable for optimization and control.

Definition 3. (Summarized from [20]) *Bernstein Polynomial Basis* Given $\xi \in [0, 1]$, let $\phi(\xi) \in \mathbb{R}^Q$ be the vector of basis functions whose q -th element, for Bernstein polynomials of order Q (degree $Q-1$), is

$$\phi_q(\xi) = \binom{Q-1}{q} \xi^q (1-\xi)^{Q-1-q}, \quad q = 0, \dots, Q-1.$$

With this, for any point $\mathbf{p} \in \mathbb{R}^d$, the multivariate basis is constructed via the Kronecker product

$$\Psi(\mathbf{p}) = \bigotimes_{p=1}^d \phi(\xi_p) \in \mathbb{R}^{Q^d}, \quad (4)$$

where $\xi_p = \frac{[\mathbf{p}]_p - [\mathbf{p}]_p^{\min}}{[\mathbf{p}]_p^{\max} - [\mathbf{p}]_p^{\min}} \in [0, 1]$ and $p = 1, \dots, d$ are the dimension-wise normalized coordinates of \mathbf{p} (i.e., the normalized value of the point's p -th dimension), where $[\mathbf{p}]^{\min}$ and $[\mathbf{p}]^{\max}$ cover the maximum relative distance.

Thus, given Definition 3, the SDF is approximated as a weighted combination of the Q^d tensor-product Bernstein basis functions (order Q per dimension) [16]:

$$\hat{S}_\Omega(\mathbf{p}) = \langle \Psi(\mathbf{p}), \mathbf{w} \rangle = \Psi(\mathbf{p})^\top \mathbf{w}, \quad (5)$$

where $\hat{S}_\Omega(\mathbf{p}) \in \mathbb{R}$, $\Psi(\mathbf{p}) \in \mathbb{R}^{Q^d}$ is the multivariate Bernstein basis vector evaluated at the point \mathbf{p} , and $\mathbf{w} \in \mathbb{R}^{Q^d}$ are the coefficients. $\langle \cdot, \cdot \rangle$ is the operation of the inner product.

Following [16], the coefficients \mathbf{w} can be obtained via linear least-squares regression under the assumption that robot shapes are learned offline. Suppose we have K sampled points $\{\mathbf{p}^{(j)}, S_\Omega(\mathbf{p}^{(j)})\}_{j=1}^K$ with ground-truth signed distances $S_\Omega(\mathbf{p}^{(j)})$. By stacking the basis evaluations into a design matrix $\Psi \in \mathbb{R}^{Q^d \times K}$ and the ground-truth distances into $\mathbf{f} \in \mathbb{R}^K$, the linear regression problem is

$$\mathbf{w}^* = \arg \min_{\mathbf{w}} \|\Psi^\top \mathbf{w} - \mathbf{f}\|_2^2 \quad (6)$$

If Ψ has full row rank, the closed-form solution can be formally defined as:

$$\mathbf{w}^* = (\Psi \Psi^\top)^{-1} \Psi \mathbf{f} \quad (7)$$

With the closed-form coefficients \mathbf{w}^* , we can obtain the smooth SDF approximation $\hat{S}_\Omega(\mathbf{p}) = \Psi(\mathbf{p})^\top \mathbf{w}^*$ over the workspace. However, obtaining the solution via explicit inversion of a large matrix is unnecessary and often prohibitive, both computationally and in memory, and can be numerically unstable [21]. Readers are referred to [16] for more detailed discussions.

C. Problem Statement

In this paper, our objective is to develop a control framework that *leverages the Bernstein-polynomial approximation* \hat{S}_Ω of the SDF and *employs control barrier functions (CBFs)* to encode and enforce safety in realistic, complex environments. While the BP-SDF surrogate offers a compact and differentiable geometric representation, translating this approximation into formal safety guarantees remains non-trivial in cluttered scenes and in the presence of interacting agents. With that, given any control bound $\mathbf{u}_{\max} \in \mathbb{R}$ of each robot i , the problem becomes how to minimally modify any pre-defined nominal input \mathbf{u}^{nom} with BP-SDF-defined safe set, i.e., enforcing nonnegative distance between the robot $\hat{S}_\mathcal{R}(\cdot)$ and the non-robot $\hat{S}_\mathcal{O}(\cdot)$:

$$\mathbf{u}^* = \arg \min_{\mathbf{u}} \|\mathbf{u} - \mathbf{u}^{\text{nom}}\|^2 \quad (8)$$

$$\text{s.t. } \mathbf{x} \in \{\text{dist}(\hat{S}_\mathcal{O}(\cdot) = 0, \hat{S}_\mathcal{R}(\cdot) = 0) \geq 0\} \quad (9)$$

$$\|\mathbf{u}\| \leq \mathbf{u}_{\max}$$

Accordingly, in this paper, we focus on two questions:

- **Single-robot safe navigation.** Given a BP-SDF approximation of the workspace containing multiple irregular-shaped static obstacles, how can we transfer the state constraints in Eq. (9) to control constraints and design a control framework that can guarantee that the robot safely navigates in the workspace?

- **Multi-robot extension.** Given a team of N robots where each robot i maintains a robot-specific BP-SDF \hat{S}_Ω^i , how can this framework be extended to multiple robots so that collisions among moving agents are avoided?

III. METHOD

In this section, we build on the unified SDF representation in Eq. (5) to formulate novel control barrier functions that directly encode geometry-aware collision avoidance conditions and derive corresponding control constraints. We first present the single-robot SDF barrier certificates and then extend the approach to the multi-robot setting.

A. Unified Barrier Function with Bernstein polynomials

Let $\mathcal{O} = \bigcup_{j=1}^M o_j$ denote the union of all obstacles. Following the work [16], we use Bernstein polynomials to represent the Signed Distance Field (SDF) of the robot \mathbf{x}_i and obstacles $\mathbf{x}_o^{\text{obs}} \in \mathcal{O}$. As shown in Fig. 1(a) and leveraging Eq. (5), we represent the robot i by a SDF $\hat{S}_\mathcal{R}^i(\cdot)$ and the composite obstacle set by a single *unified* SDF $\hat{S}_\mathcal{O}(\cdot)$.

$$\hat{S}_\mathcal{R}^i(\mathbf{p}_i) = \langle \Psi(\mathbf{p}_i), \mathbf{w}_i \rangle, \quad \hat{S}_\mathcal{O}(\mathbf{p}_o^{\text{obs}}) = \langle \Psi(\mathbf{p}_o^{\text{obs}}), \mathbf{w}_o \rangle$$

where $\hat{S}_\mathcal{R}^i(\cdot)$ is in the robot's local frame and $\hat{S}_\mathcal{O}(\cdot)$ is in the unified obstacle frame. \mathbf{p}_i and $\mathbf{p}_o^{\text{obs}}$ represent the point \mathbf{p} in the robot frame and unified obstacle frame.

We define the function $\Phi(\mathbf{x}, \mathbf{p}_{\text{local}})$ to transform the point in the local frame $\mathbf{p}_{\text{local}}$ to the global frame \mathbf{p} . With this,

$$F_i(\mathbf{x}_i, \mathbf{p}) = \hat{S}_\mathcal{R}^i(\Phi_i^{-1}(\mathbf{x}_i, \mathbf{p})) \quad (10)$$

represents the distance between any point \mathbf{p} and the boundary of the robot i in the space.

Therefore, $F_i(\mathbf{x}_i, \mathbf{p}) = 0$ can be used to define the zero level set for the robot i given its SDF.

Inspired by [22], we first characterize the safe space using the minimum distance between the robot and the composite obstacle set. To this end, we formulate the following optimization problem to find the closest point \mathbf{p}^* on the obstacle boundary to the robot:

$$\mathbf{p}_{i,o}^* = \arg \min_{\mathbf{p}} F_i(\mathbf{x}_i, \mathbf{p}) \quad (11)$$

$$\text{s.t. } F_o(\mathbf{x}_o^{\text{obs}}, \mathbf{p}) = 0$$

As $F_i(\mathbf{x}_i, \mathbf{p})$ denotes the signed distance from \mathbf{p} to the boundary of robot i , minimizing it corresponds to finding the closest point to the robot. And the constraint $F_o(\mathbf{x}_o^{\text{obs}}, \mathbf{p}) = 0$ enforces that \mathbf{p} lies on the obstacle boundary. Therefore, as illustrated in Fig. 1, the solution $\mathbf{p}_{i,o}^*$ is exactly the obstacle boundary point with the minimum distance from the robot, which is then used to define the barrier function for collision avoidance. By solving the optimization problem in (11), one can obtain the optimal solution $\mathbf{p}_{i,o}^*$, which is the closest point on the obstacle boundary to robot i .

Hence, to formally characterize safety, we directly utilize the property of the BP-SDF, which returns the signed distance from the robot to any given point. By evaluating

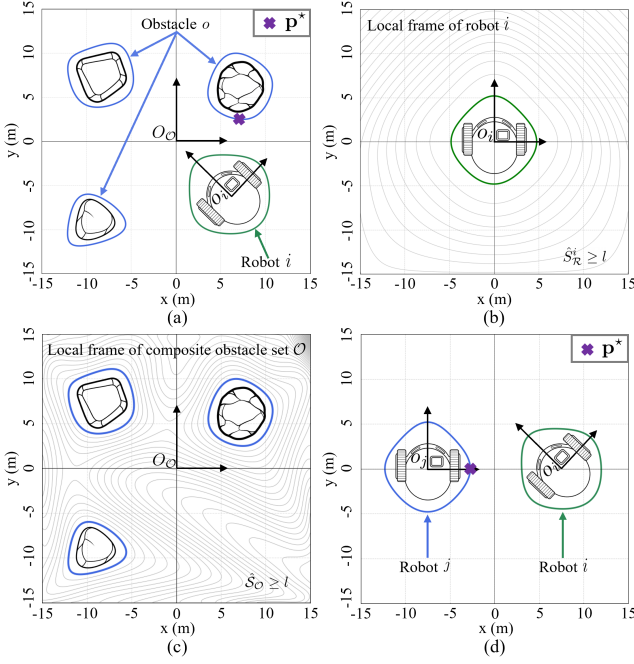


Fig. 1: Examples of the level sets for SDF approximation using Bernstein polynomials on robots and obstacles. (a) With a safety margin $l \in \mathbb{R}$, the green contour represents the l -level set of the approximated SDF of the robot i . The three blue contours represent the l -level set of the approximated SDF w.r.t. all three obstacles in the workspace using a single Bernstein polynomial representation. (b) The green contour represents the l -level set of the approximated SDF of the robot i in the robot frame O_i , satisfying $\hat{S}_{\mathcal{R}} \geq l$. (c) The blue contours represent the l -level set of the approximated SDF of the composite obstacle set \mathcal{O} in the frame $O_{\mathcal{O}}$, satisfying $\hat{S}_{\mathcal{O}} \geq l$. (d) The green and blue contours represent the l -level sets of the approximated SDF for robot i and j .

the robot's SDF at the closest obstacle-boundary point $\mathbf{p}_{i,o}^*$, we formally define the **Geometry-aware Control Barrier Function (CBF)** as:

$$h(\mathbf{x}_i, \mathbf{x}_o^{\text{obs}}) = F_i(\mathbf{x}_i, \mathbf{p}_{i,o}^*) \quad (12)$$

Therefore, $h(\mathbf{x}_i, \mathbf{x}_o^{\text{obs}}) \geq 0$ indicates that robot i remains away from the obstacle, while $h(\mathbf{x}_i, \mathbf{x}_o^{\text{obs}}) < 0$ indicates overlap between the robot and the obstacle. Accordingly, the safe set is defined as

$$\mathcal{H}_{i,o}^{\text{obs}} = \{(\mathbf{x}_i, \mathbf{x}_o^{\text{obs}}) \in \mathbb{R}^d \mid h(\mathbf{x}_i, \mathbf{x}_o^{\text{obs}}) \geq 0\}. \quad (13)$$

B. Single-robot Geometry-aware Control Barrier Certificates

By using the KKT condition to solve the problem in Eq. (11), we can obtain the optimal value of the barrier function h and the closest point to the robot, \mathbf{p}^* . We propose the following proposition to define the safe control space.

Proposition 4 (Geometry-aware Control Barrier Certificates). Consider robot i with control-affine dynamics (1) moving in a workspace with any-shaped static obstacles which are enclosed by a unified SDF $\hat{S}_{\mathcal{O}}$. The barrier function $h(\mathbf{x}_i, \mathbf{x}_o^{\text{obs}})$ and the corresponding obstacle boundary point are $\mathbf{p}_{i,o}^*$ in Eq. (11). Following Lemma. 1, for the robot

i and the environment with obstacles \mathcal{O} , the admissible control space for any Lipschitz continuous controller rendering the safe set in Eq. (13) forward invariant, thus becomes:

$$\mathcal{B}_{\mathbf{u}}^{\text{obs}} = \{ \mathbf{u}_i \in \mathcal{U}_i \mid a_i(\mathbf{x}_i, \mathbf{p}_{i,o}^*) + A_i(\mathbf{x}_i, \mathbf{p}_{i,o}^*) \mathbf{u}_i \geq -\gamma h(\mathbf{x}_i, \mathbf{x}_o^{\text{obs}}) \} \quad (14)$$

where $a_i(\mathbf{x}_i, \mathbf{p}_{i,o}^*) = \nabla_{\mathbf{x}_i} F_i(\mathbf{x}_i, \mathbf{p}_{i,o}^*)^\top f_i(\mathbf{x}_i)$ and $A_i(\mathbf{x}_i, \mathbf{p}_{i,o}^*) = \nabla_{\mathbf{x}_i} F_i(\mathbf{x}_i, \mathbf{p}_{i,o}^*)^\top g_i(\mathbf{x}_i)$.

Proof: We assume the BP-SDF level sets are convex such that the closest point $\mathbf{p}_{i,o}^*$ changes smoothly without discontinuous jumps. Additionally, $\|\nabla F_i\| \neq 0$ on the boundary of the BP-SDF learned offline.

With this, differentiating Eq. (12) with respect to time t using the chain rule yields:

$$\dot{h}(\mathbf{x}_i, \mathbf{x}_o^{\text{obs}}) = \nabla_{\mathbf{x}_i} F_i(\mathbf{x}_i, \mathbf{p}_{i,o}^*)^\top \dot{\mathbf{x}}_i + \nabla_{\mathbf{p}} F_i(\mathbf{x}_i, \mathbf{p}_{i,o}^*)^\top \dot{\mathbf{p}}_{i,o}^*$$

Based on the KKT conditions of the optimization problem in Eq. (11), the spatial gradient $\nabla_{\mathbf{p}} F_i(\mathbf{x}_i, \mathbf{p}^*)$ is parallel to the normal vector of the obstacle boundary at \mathbf{p}^* . Since $\dot{\mathbf{p}}^*$ is tangent to the obstacle boundary, their inner product is zero, i.e., $\nabla_{\mathbf{p}} F_i(\mathbf{x}_i, \mathbf{p}^*)^\top \dot{\mathbf{p}}^* = 0$.

Substituting this identity gives us the general form:

$$\dot{h}(\mathbf{x}_i, \mathbf{x}_o^{\text{obs}}) = \nabla_{\mathbf{x}_i} F_i(\mathbf{x}_i, \mathbf{p}_{i,o}^*)^\top \dot{\mathbf{x}}_i \quad (15)$$

Following Lemma 1, the safe control space can thus be given by Eq. (14). \square

This defines the admissible control space that ensures the robot to stay away from the obstacle boundary. Finally, one can define the safe controller \mathbf{u}_i^* for a robot navigating in a multi-obstacle scenario by minimally modifying the nominal control $\mathbf{u}_i^{\text{nom}}$ with Quadratic Programming (QP):

$$\mathbf{u}_i^* = \arg \min_{\mathbf{u}_i} \|\mathbf{u}_i - \mathbf{u}_i^{\text{nom}}\|_2^2 \quad (16)$$

$$\text{s.t. } a_i(\mathbf{x}_i, \mathbf{p}_{i,o}^*) + A_i(\mathbf{x}_i, \mathbf{p}_{i,o}^*) \mathbf{u}_i \geq -\gamma h(\mathbf{x}_i, \mathbf{x}_o^{\text{obs}}) \\ \|\mathbf{u}_i\| \leq \mathbf{u}_i^{\text{max}}$$

where $\mathbf{u}_i^{\text{max}}$ denotes the control bound for robot i .

C. Handling Non-Smoothness in Composite Obstacles

As defined in Section III-A, representing multiple obstacles as a single composite set \mathcal{O} provides a unified geometric constraint. However, in such a case, the closest point $\mathbf{p}_{i,o}^*$ on the obstacle boundary can discontinuously jump between different obstacles while the robot is moving. This can make the distance function non-smooth, which violates the continuous differentiability property of CBFs.

To deal with the composite set of obstacles scenario while maintaining smoothness, we adopt a Configuration Space (C-space) approach. By approximating the robot as a bounding circle of radius r [8], we inflate the ground-truth composite obstacles using the Minkowski sum to create the C-space obstacles, denoted as $\mathcal{O}_C = \mathcal{O} \oplus \mathcal{D}(r)$, where $\mathcal{D}(r)$ represents a disk of radius r . We then construct a unified BP-SDF for the C-space, denoted as $\hat{S}_{\mathcal{O}_C}(\cdot)$. However, if this causes overly conservative behaviors of the robot, we can define each obstacle as an individual safety constraint.

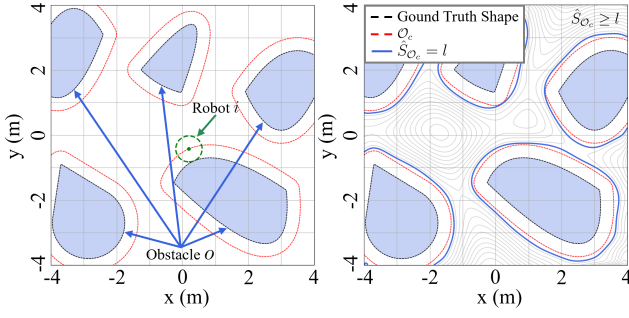


Fig. 2: C-space (red) of composite obstacle set \mathcal{O}_C inflated by the radius r of the robot i (green). The right figure shows the SDF $\hat{S}_{\mathcal{O}_C}$ of the C-space (blue contours) with a user-defined safety margin level set l .

In this C-space formulation, the robot is treated as a point \mathbf{x}_i . The barrier function is defined by the BP-SDF of the C-space at the robot's position:

$$h_i^{obs}(\mathbf{x}_i) = \hat{S}_{\mathcal{O}_C}(\mathbf{x}_i) = \Psi(\mathbf{x}_i)^\top \mathbf{w}_o \quad (17)$$

Because the Bernstein polynomial approximation is inherently smooth and continuously differentiable, evaluating $\hat{S}_{\mathcal{O}_C}(\mathbf{x}_i)$ guarantees a smooth CBF everywhere in the workspace.

D. Multi-robot Geometry-aware Control Barrier Certificates

We continue to extend our Geometry-aware Control Barrier Function in (12) for the multi-robot collision avoidance case. Consider a team of N robots, similarly to (11), for any pair of robots i and j , one can calculate the closest point $\mathbf{p}_{i,j}^*$ on the boundary j to robot i . Then for any pairwise collision avoidance in the robot team, the safety of $\mathbf{x} = \{\mathbf{x}_1, \dots, \mathbf{x}_i, \dots, \mathbf{x}_N\}$ is defined as:

$$h(\mathbf{x}_i, \mathbf{x}_j) = F_i(\mathbf{x}_i, \mathbf{p}_{i,j}^*), \forall i \neq j \quad (18)$$

With this, $h(\mathbf{x}_i, \mathbf{x}_j) \geq 0$ indicates that, from robot i 's view, the robot j is sufficiently away from robot i . Accordingly, the safe set is defined as

$$\mathcal{H}_{i,j} = \{\mathbf{x} \in \mathbb{R}^{dN} \mid h(\mathbf{x}_i, \mathbf{x}_j) \geq 0\}, \forall i \neq j. \quad (19)$$

Proposition 5 (Multi-robot Geometry-aware Control Barrier Certificates). *Considering a team of N robots moving in the same workspace, each robot has its own robot-specific SDF approximated by Bernstein polynomials. $\mathbf{u} = \{\mathbf{u}_1, \dots, \mathbf{u}_N\}$ is the joint control of the N robots. As shown in Fig. 1(d), for any pair of robots i and j , the optimal point $\mathbf{p}_{i,j}^*$ moves along the surface of robot j . By taking the time derivative of the boundary constraint $F_j(\mathbf{x}_j, \mathbf{p}_{i,j}^*) = 0$ and applying the KKT conditions, the inter-robot collision avoidance can be enforced using the following **Multi-robot Geometry-aware Control Barrier Certificates**:*

$$\begin{aligned} \mathcal{B}_{\mathbf{u}}^{\text{robots}} = \{ & \mathbf{u} \in \mathcal{U} \mid a_i(\mathbf{x}_i, \mathbf{p}_{i,j}^*) + a_j(\mathbf{x}_j, \mathbf{p}_{i,j}^*) \\ & + A_i(\mathbf{x}_i, \mathbf{p}_{i,j}^*)\mathbf{u}_i + A_j(\mathbf{x}_j, \mathbf{p}_{i,j}^*)\mathbf{u}_j \\ & \geq -\gamma h(\mathbf{x}_i, \mathbf{x}_j)\} \end{aligned} \quad (20)$$

where $a_j(\mathbf{x}_j, \mathbf{p}_{i,j}^*) = \lambda \nabla_{\mathbf{x}_j} F_j(\mathbf{x}_j, \mathbf{p}_{i,j}^*)^\top f_j(\mathbf{x}_j)$, $A_j(\mathbf{x}_j, \mathbf{p}_{i,j}^*) = \lambda \nabla_{\mathbf{x}_j} F_j(\mathbf{x}_j, \mathbf{p}_{i,j}^*)^\top g_j(\mathbf{x}_j)$, and the scalar multiplier $\lambda = \frac{\|\nabla_{\mathbf{p}} F_i(\mathbf{x}_i, \mathbf{p}_{i,j}^*)\|}{\|\nabla_{\mathbf{p}} F_j(\mathbf{x}_j, \mathbf{p}_{i,j}^*)\|}$.

Proof: Taking the time derivative of $h(\mathbf{x}_i, \mathbf{x}_j) = F_i(\mathbf{x}_i, \mathbf{p}_{i,j}^*)$ yields $\dot{h}(\mathbf{x}_i, \mathbf{x}_j) = \nabla_{\mathbf{x}_i} F_i^\top \dot{\mathbf{x}}_i + \nabla_{\mathbf{p}} F_i^\top \dot{\mathbf{p}}_{i,j}^*$. From the KKT stationarity conditions, we have $\nabla_{\mathbf{p}} F_i = -\lambda \nabla_{\mathbf{p}} F_j$. Substituting this gives $\dot{h}(\mathbf{x}_i, \mathbf{x}_j) = \nabla_{\mathbf{x}_i} F_i^\top \dot{\mathbf{x}}_i - \lambda \nabla_{\mathbf{p}} F_j^\top \dot{\mathbf{p}}_{i,j}^*$.

Because $\mathbf{p}_{i,j}^*$ must remain on the boundary of robot j , the constraint $F_j(\mathbf{x}_j, \mathbf{p}_{i,j}^*) = 0$ is satisfied at all times. Differentiating this gives $\nabla_{\mathbf{x}_j} F_j^\top \dot{\mathbf{x}}_j + \nabla_{\mathbf{p}} F_j^\top \dot{\mathbf{p}}_{i,j}^* = 0$, which allows us to substitute out the velocity $\dot{\mathbf{p}}_{i,j}^*$, yielding:

$$\dot{h}(\mathbf{x}_i, \mathbf{x}_j) = \nabla_{\mathbf{x}_i} F_i(\mathbf{x}_i, \mathbf{p}_{i,j}^*)^\top \dot{\mathbf{x}}_i + \lambda \nabla_{\mathbf{x}_j} F_j(\mathbf{x}_j, \mathbf{p}_{i,j}^*)^\top \dot{\mathbf{x}}_j \quad (21)$$

Following Lemma 1, one can obtain the safe control space for the multi-robot system in (20). \square

With this, one can define the following safe controller \mathbf{u}^* for the team of robots by minimally modifying their nominal control \mathbf{u}^{nom} with the following QP:

$$\begin{aligned} \mathbf{u}^* = \arg \min_{\mathbf{u}} \quad & \|\mathbf{u} - \mathbf{u}^{\text{nom}}\|^2 \quad (22) \\ \text{s.t.} \quad & a_i(\mathbf{x}_i, \mathbf{p}_{i,j}^*) + a_j(\mathbf{x}_j, \mathbf{p}_{i,j}^*) + A_i(\mathbf{x}_i, \mathbf{p}_{i,j}^*)\mathbf{u}_i \\ & + A_j(\mathbf{x}_j, \mathbf{p}_{i,j}^*)\mathbf{u}_j \geq -\gamma h(\mathbf{x}_i, \mathbf{x}_j), \forall i \neq j \\ & \|\mathbf{u}_i\| \leq \mathbf{u}_i^{\text{max}}, \quad \forall i = 1, \dots, N \end{aligned}$$

Remark 1. *While it has been proven that a minimum distance function between two convex sets is a smooth function [23], the learned SDFs are approximations, and some SDF level sets can be non-convex even if the ground truth shape is convex. To handle the non-smoothness, one can formulate a constrained Quadratic Program (QP) for offline learning and strictly enforce convexity on all level sets by constraining second-order difference matrix of the Bernstein weights w to be positive semi-definite.*

Remark 2. *It is possible that the learned SDFs with Bernstein polynomials and the method in [16] has a small approximation error. A user-defined safety margin level set l can be used to guarantee the enclosure of the ground truth shape in the learned shape by changing the constraints in Eq. (11) as $F_o(\mathbf{x}_o^{\text{obs}}, \mathbf{p}) - l = 0$ ($l \geq 0 \in \mathbb{R}$).*

Remark 3. *It is possible that our method would get into a deadlock. Readers can refer to [24]–[26] for single-robot and multi-robot deadlock resolution.*

IV. RESULTS

In this section, we conduct simulations to demonstrate the robustness and versatility of the proposed framework. By evaluating its performance across diverse environments, robot dynamics, and homogeneous as well as heterogeneous teams, we establish its broad applicability in ensuring safe and reliable multi-robot coordination. Note that the proposed algorithm is evaluated under *different* system dynamics, including single-integrator and unicycle dynamics.

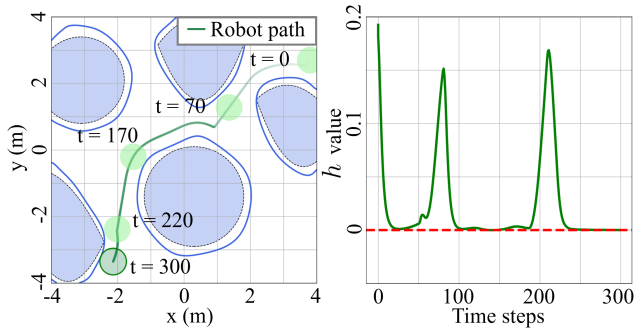


Fig. 3: **Left:** A single BP-SDF ($Q = 23$) is used to represent five distinct static obstacles, and its 0.27-level set is used to enclose the ground-truth shapes of the static obstacles. The green line shows the robot trajectory starting from $(3.8, 2.7)$ and visiting the designated goal position $(-2.2, -3.5)$. **Right:** The green line represents the value of the barrier function over time steps. The barrier function is obtained from a single CBF constructed using a BP-SDF, which compactly encodes all obstacle-related safety constraints into one unified condition.

In the first experiment, we consider a single robot navigating through an environment populated with multiple obstacles of varying shapes. The robot dynamics are modeled as a single-integrator system, such that $\dot{\mathbf{x}}_i = \mathbf{u}_i$, where $\mathbf{x}_i \in \mathbb{R}^2$ denotes the robot position and $\mathbf{u}_i \in \mathbb{R}^2$ is the control input. The navigation objective is to reach the designated time-invariant goal position while avoiding collisions with obstacles. The nominal task controller is $\mathbf{u}^{\text{nom}} = -k(\mathbf{x}_i - \mathbf{x}_i^{\text{goal}})$, where k is the control gain. As illustrated in Figure 3, the robot starts at $\mathbf{x} = [3.8, 2.7]^\top$ and visits the goal $\mathbf{x}^{\text{goal}} = [-2.2, -3.5]^\top$.

To enforce safety, we construct the control barrier function (CBF) as in Eq. (12). \mathbf{p}^* lies on the 0.27-level set (setting $l = 0.27$ as stated in Remark. 2) of the signed distance field approximated by a Bernstein polynomial with order $Q = 23$. The use of the 0.27-level set ensures that all five obstacles are fully enclosed within a *single* unified BP-SDF. In Figure 3, the dashed contours indicate the ground-truth obstacle boundaries, while the fading green curve in the left subfigure illustrates the robot’s trajectory. The right subfigure further depicts the temporal evolution of the barrier function, confirming that the safety constraint is satisfied throughout the navigation process.

A. Safe Navigation of Two Robots under Unicycle Dynamics

In the second set of experiments, we consider robots governed by unicycle dynamics:

$$\dot{\mathbf{x}}_i = \begin{bmatrix} \dot{x}_i \\ \dot{y}_i \\ \dot{\theta}_i \end{bmatrix} = \begin{bmatrix} \cos(\theta_i) & 0 \\ \sin(\theta_i) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_i \\ \omega_i \end{bmatrix}, \quad (23)$$

where $[x_i, y_i, \theta_i]^\top$ denotes the state of robot i in the global frame, v_i is the linear velocity, and ω_i is the angular velocity. The control limits are $|v_i| \leq 1$ m/s and $|\omega_i| \leq \pi/2$ rad/s.

Given the robot’s current state (x_i, y_i, θ_i) and its corresponding time-invariant goal position (x_i^g, y_i^g) , the nominal task controller is designed as $\hat{v}_i = k_v \rho_i$, $\hat{\omega}_i = k_\omega \alpha_i$, where $k_v, k_\omega > 0$ are proportional gains. Specifically, ρ_i denotes

the Euclidean distance to the goal and α_i the heading error, given by $\rho_i = \sqrt{(x_i^g - x_i)^2 + (y_i^g - y_i)^2}$, $\theta_i^g = \text{atan2}(y_i^g - y_i, x_i^g - x_i)$, $\alpha_i = \theta_i^g - \theta_i$. Alternatively, one may employ well-developed unicycle controllers, e.g., [27], to construct the nominal controller \mathbf{u}^{nom} and drive the robot toward the desired goal. Note that a detailed discussion of such nominal controller design is outside the scope of this paper.

Fig. 4 illustrates the two-robot position-switching task: the figures show the robot motions with approximated shapes obtained from BP-SDFs. Fig. 5 shows the linear and angular velocity inputs and their respective limits, together with the evolution of barrier functions. The results confirm that the robots complete the task without collision under different dynamics, while satisfying both the safety constraint and control bounds.

B. Safe Navigation of Four Heterogeneous Robots

The robot dynamics and the design of the task-related controller in this experiment follow the same formulation as in Section. IV-A.

In Fig. 6, the heterogeneous example involves four robots with distinct geometries approximated by its own SDF. Accordingly, four separate control barrier functions (h_1, h_2, h_3, h_4) are required. The results in the Fig. 4 and Fig. 6 demonstrate that the proposed framework extends naturally from homogeneous to heterogeneous multi-robot teams while ensuring safety.

It is worth noting that, as illustrated in Fig. 6, heterogeneous robot shapes are difficult to approximate using Minkowski-sum-based approaches, which typically rely on conservative simplifications such as bounding circles or convex polygons [13]. In contrast, our framework introduces a new CBF formulation that ensures safety under BP-SDF-based modeling of complex and realistic robot geometries.

C. Quantitative Results

We evaluate our framework on a desktop with 64GB of RAM and an AMD Ryzen 9 3900X processor. For each obstacle count $m \in \{1, 2, 4, 6, \dots, 24\}$, we conducted five independent randomized trials, each with a *distinct* obstacle configuration and *different* start and goal position pair, and report two summary metrics. (i) The average per-timestep computation time $\bar{\tau}$ (s/step) and (ii) the ground-truth minimum robot-obstacle distance d^* (true set-to-set Euclidean distance). Fig. 7 shows mean curves with shaded area: in the **top** panel, the band denotes ± 1 standard deviation and $\bar{\tau}$ shows slow growth as m increases, indicating the efficiency of our proposed method; in the **bottom** panel, the band also denotes ± 1 standard deviation and $d^* > 0$ for all m , demonstrating collision-free operation and robustness in complex environments.

V. CONCLUSION

In this paper, we proposed a novel method to define Control Barrier Functions using Bernstein-Polynomial Signed Distance Field (BP-SDF). This method utilizes Bernstein

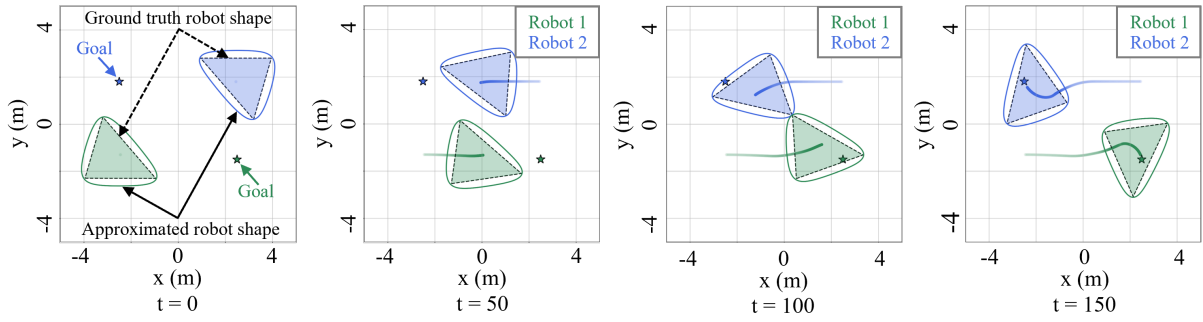


Fig. 4: Two-robot position-switching under unicycle dynamics. The solid lines and dashed lines represent the approximated shapes by BP-SDFs and ground truth shapes of the robots, respectively. Snapshots at $t = 0, 50, 100,$ and 150 are shown as representative time instances. The visualizations indicate that the robots maintain collision-free execution throughout the entire position-switching process.

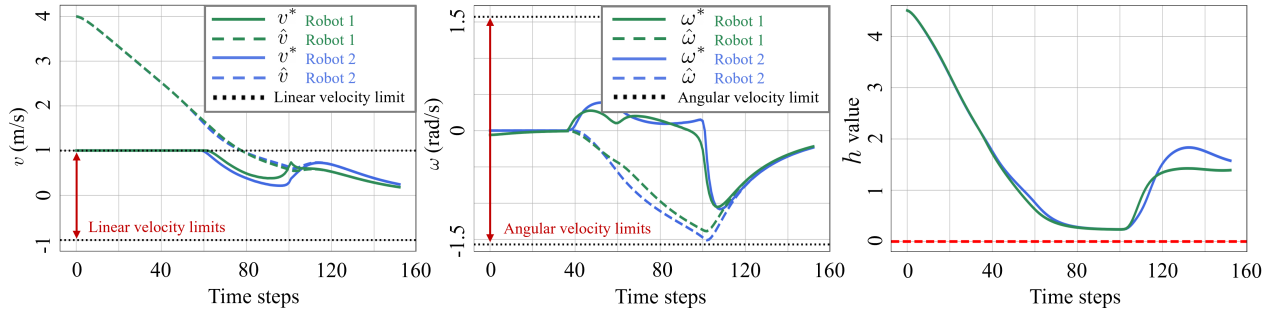


Fig. 5: Control input and safety analysis for the two-robot position-switching task. **Left:** linear velocity inputs v^* (solid) compared with nominal task inputs \hat{v} (dashed) for both robots, together with the linear velocity limits. **Middle:** angular velocity inputs ω^* (solid) compared with nominal task inputs $\hat{\omega}$ (dashed), with the corresponding angular velocity limits. **Right:** evolution of the barrier function over time, which remains nonnegative, verifying that the safety constraint is satisfied throughout the entire execution.

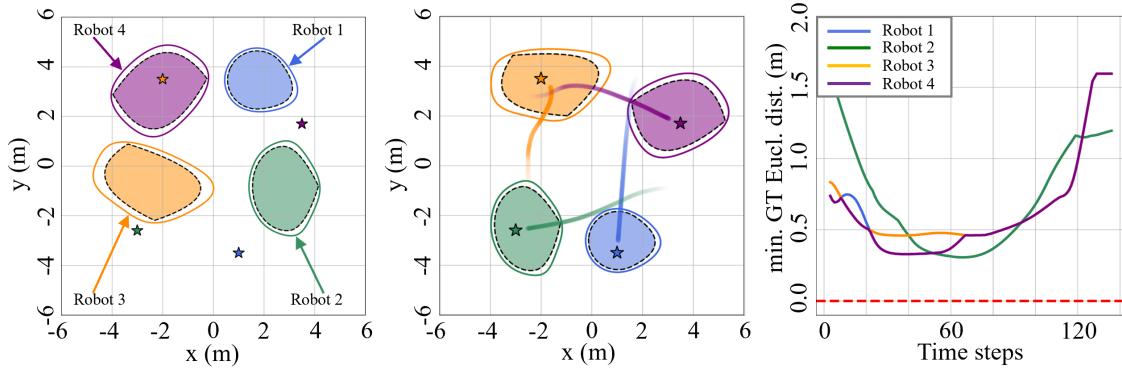


Fig. 6: Collision avoidance among four mobile robots with different smooth convex shapes represented by the respective SDF level sets. The right figure shows the minimum Euclidean distance between the ground truth shapes remaining positive over time, which validates the safety guarantees as the robots move to the designated goal positions (marked by stars with the same colors as the robots).

polynomials (BPs) to unify the representation of obstacles and robots. The numerical results across diverse environments, robot dynamics, and homogeneous as well as heterogeneous teams verified the effectiveness of the proposed method. For future work, we will investigate updating the BP-SDF coefficients online using real-time observations, so that the geometry model can be refined after the initial offline learning stage when the shapes of the obstacles change over time.

REFERENCES

- [1] Y. Ren, F. Zhu, G. Lu, Y. Cai, L. Yin, F. Kong, J. Lin, N. Chen, and F. Zhang, "Safety-assured high-speed navigation for mavs," *Science Robotics*, vol. 10, no. 98, p. ead06187, 2025.
- [2] Y. Zhang, Y. Lyu, S. E. Demir, X. Zhou, Y. Yang, J. Wang, and W. Luo, "Courteous mpc for autonomous driving with cbf-inspired risk assessment," in *2024 IEEE 27th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2024, pp. 2317–2323.
- [3] Y. Lyu, W. Luo, and J. M. Dolan, "Risk-aware safe control for decentralized multi-agent systems via dynamic responsibility allocation," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023, pp. 9737–9744.

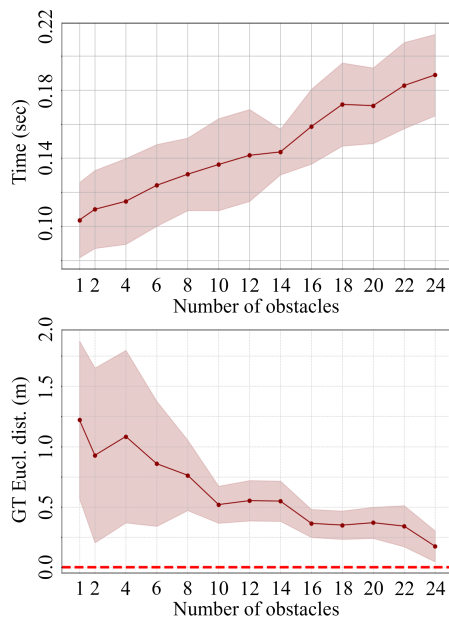


Fig. 7: **Quantitative results.** Curves show means over five trials. **Top:** per-time-step computation time; shaded area denotes ± 1 standard deviation. **Bottom:** minimum ground-truth robot–obstacle distance; shaded area denotes ± 1 standard deviation.

- [4] T. Kim and D. Panagou, “Visibility-aware rrt* for safety-critical navigation of perception-limited robots in unknown environments,” *IEEE Robotics and Automation Letters*, 2025.
- [5] M. W. Otte, S. G. Richardson, J. Mulligan, and G. Grudic, “Path planning in image space for autonomous robot navigation in unstructured environments,” *Journal of Field Robotics*, vol. 26, no. 2, pp. 212–240, 2009.
- [6] S. Wu, Y. Fang, N. Sun, B. Lu, X. Liang, and Y. Zhao, “Optimization-free smooth control barrier function for polygonal collision avoidance,” *IEEE Transactions on Cybernetics*, 2025.
- [7] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, “Control barrier functions: Theory and applications,” in *18th European control conference (ECC)*. Ieee, 2019, pp. 3420–3431.
- [8] L. Wang, A. D. Ames, and M. Egerstedt, “Safety barrier certificates for collisions-free multirobot systems,” *IEEE Transactions on Robotics*, vol. 33, no. 3, pp. 661–674, 2017.
- [9] —, “Safe certificate-based maneuvers for teams of quadrotors using differential flatness,” in *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 3293–3298.
- [10] W. Xiao, N. Mehdipour, A. Collin, A. Y. Bin-Nun, E. Frazzoli, R. D. Tebbens, and C. Belta, “Rule-based optimal control for autonomous driving,” in *Proceedings of the ACM/IEEE 12th International Conference on Cyber-Physical Systems*, 2021, pp. 143–154.
- [11] Y. Yang, Y. Lyu, and W. Luo, “Minimally constrained multi-robot coordination with line-of-sight connectivity maintenance,” in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 7684–7690.
- [12] Y. Yang, Y. Lyu, Y. Zhang, S. Yi, and W. Luo, “Decentralized Multi-Robot Line-of-Sight Connectivity Maintenance under Uncertainty,” in *Proceedings of Robotics: Science and Systems (RSS)*, Delft, Netherlands, July 2024.
- [13] Y.-H. Chen, S. Liu, W. Xiao, C. Belta, and M. Otte, “Control barrier functions via minkowski operations for safe navigation among polytopic sets,” in *2025 IEEE 64th Conference on Decision and Control (CDC)*, 2025, pp. 4481–4488.
- [14] M. A. Khan, T. Ibuki, and A. Chatterjee, “Gaussian control barrier functions: Non-parametric paradigm to safety,” *IEEE Access*, vol. 10, pp. 99 823–99 836, 2022.
- [15] K. Long, C. Qian, J. Cortés, and N. Atanasov, “Learning barrier functions with memory for robust safe navigation,” *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 4931–4938, 2021.
- [16] Y. Li, Y. Zhang, A. Razmjoo, and S. Calinon, “Representing robot geometry as distance fields: Applications to whole-body manipulation,” in *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 15 351–15 357.
- [17] P. Ong, B. Capelli, L. Sabattini, and J. Cortés, “Nonsmooth control barrier function design of continuous constraints for network connectivity maintenance,” *Automatica*, vol. 156, p. 111209, 2023.
- [18] W. Xiao and C. Belta, “Control barrier functions for systems with high relative degree,” in *2019 IEEE 58th conference on decision and control (CDC)*. IEEE, 2019, pp. 474–479.
- [19] M. Mukadam, J. Dong, X. Yan, F. Dellaert, and B. Boots, “Continuous-time gaussian process motion planning via probabilistic inference,” *The International Journal of Robotics Research*, vol. 37, no. 11, pp. 1319–1340, 2018.
- [20] S. Calinon, “Mixture models for the analysis, edition, and synthesis of continuous time series,” in *Mixture Models and Applications*. Springer, 2019, pp. 39–57.
- [21] W. W. Hager, “Updating the inverse of a matrix,” *SIAM review*, vol. 31, no. 2, pp. 221–239, 1989.
- [22] B. Dai, R. Khorrambakhht, P. Krishnamurthy, and F. Khorrami, “Sailing through point clouds: Safe navigation using point cloud based control barrier functions,” *IEEE Robotics and Automation Letters*, 2024.
- [23] A. Thirugnanam, J. Zeng, and K. Sreenath, “Control barrier functions for collision avoidance between strongly convex regions,” *arXiv preprint arXiv:2306.13259*, 2025.
- [24] M. F. Reis, A. P. Aguiar, and P. Tabuada, “Control barrier function-based quadratic programs introduce undesirable asymptotically stable equilibria,” *IEEE Control Systems Letters*, vol. 5, no. 2, pp. 731–736, 2020.
- [25] Y. Zhang, Y. Lyu, S. Jo, Y. Yang, and W. Luo, “Adaptive deadlock avoidance for decentralized multi-agent systems via cbf-inspired risk measurement,” in *2025 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2025, pp. 14 917–14 923.
- [26] W. Lee, J. Sim, J. Kim, S. Jo, W. Luo, and C. Nam, “Merry-go-round: Safe control of decentralized multi-robot systems with deadlock prevention,” in *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2025, pp. 4589–4595.
- [27] C. Samson, “Control of chained systems application to path following and time-varying point-stabilization of mobile robots,” *IEEE transactions on Automatic Control*, vol. 40, no. 1, pp. 64–77, 2002.