

Author-accepted manuscript

Published citation

Z. Wu and L. Liu, "Collision-free Control Barrier Functions for General Ellipsoids via Separating Hyperplane," in *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Hangzhou, China: IEEE, Oct. 2025, pp. 19 637–19 644. DOI: [10.1109/IROS60139.2025.11247279](https://doi.org/10.1109/IROS60139.2025.11247279)

This is the author's accepted manuscript. Please cite the published IEEE article using the DOI above. The manuscript follows this notice page.

Collision-free Control Barrier Functions for General Ellipsoids via Separating Hyperplane

Zeming Wu, and Lu Liu, *Senior Member, IEEE*,

Abstract—This paper presents a novel collision avoidance method for general ellipsoids based on control barrier functions (CBFs) and separating hyperplanes. First, collision-free conditions for general ellipsoids are analytically derived using the concept of dual cones. These conditions are incorporated into the CBF framework by extending the system dynamics of controlled objects with separating hyperplanes, enabling efficient and reliable collision avoidance. The validity of the proposed collision-free CBFs is rigorously proven, ensuring their effectiveness in enforcing safety constraints. The proposed method requires only single-level optimization, significantly reducing computational time compared to state-of-the-art methods. Numerical simulations and real-world experiments demonstrate the effectiveness and practicality of the proposed algorithm.

I. INTRODUCTION

The rapid advancement of artificial intelligence has driven the deployment of autonomous systems into increasingly complex environments, such as self-driving vehicles in urban road networks [1–4] and robotic manipulators in cluttered production lines [5–8]. In these applications, the geometric shapes of controlled objects cannot be neglected, as simplified geometric models often yield overly conservative control policies that compromise task efficiency.

Various approaches have been developed to achieve geometric-aware collision avoidance, including trajectory optimization (TO)-based methods [9, 10], model predictive control (MPC)-based methods [11, 12], and CBF-based methods [6, 13, 14]. TO-based and MPC-based methods achieve geometric-aware collision avoidance by incorporating collision-free constraints into optimization problems. However, when precise geometric modeling is required, these collision-free constraints are generally non-convex, resulting in the need to solve non-convex optimization problems. Such problems are challenging to solve efficiently and reliably onboard, making these methods less suitable for safety-critical systems that demand real-time responsiveness. CBF-based methods, on the other hand, achieve geometric-aware collision avoidance by transforming the collision-free constraints into linear constraints with respect to the control inputs. These linear constraints can be seamlessly incorporated into a quadratic program (QP) that minimally modifies a nominal controller. Since QPs are convex problems that can

be solved efficiently and reliably, CBF-based methods have demonstrated their advantages in computational efficiency, particularly in the context of real-time safety-critical control.

To date, there are only a few CBF-based methods for geometric-aware collision avoidance. For instance, signed distance is utilized to design CBFs in [6] for collision avoidance between general primitives. However, the evaluation of signed distance involves non-differentiable algorithms like the Gilbert-Johnson-Keerthi algorithm [15], which makes the computation of the time derivatives of signed distance challenging. To circumvent this issue, the time derivatives of signed distance are approximated, resulting in a conservative controller. To eliminate the conservatism introduced by approximation, a duality-based CBF has been proposed in [13] for collision avoidance between polyhedra. Nonetheless, additional optimization problems and virtual states are required for the evaluation of the CBF, making the method less computationally efficient. To address the non-smooth nature of signed distance, growth distance [16] is utilized to design smooth CBFs in [14] for collision avoidance between convex primitives. The time derivative of growth distance is calculated by leveraging the Karush-Kuhn-Tucker (KKT) conditions. Despite these advancements, two challenges still hinder the application of CBF-based methods in real-world scenarios. Firstly, from a theoretical perspective, the validity of collision-free CBFs is questionable. For instance, the proposed CBF in [13] is not continuously differentiable, as shown in their simulation results, and the gradient of the CBF may vanish on the boundary of the safe set. These two features may affect the forward invariance properties of CBFs [17]. Secondly, from an implementation perspective, the evaluation of Euclidean distance, signed distance, and growth distance, along with their time derivatives, involves solving additional optimization problems. This results in a double-level optimization process, which reduces computational efficiency.

To address these challenges, this paper proposes a novel collision avoidance control method based on CBFs and the separating hyperplane theorem. The geometric shape of the controlled object is modeled as a general ellipsoid to balance theoretical rigor and computational efficiency. Since the geometry of most controlled objects, such as vehicles [4], manipulator components [14], and quadcopters [18], can be approximated by general ellipsoids, this modeling approach can meet the precision requirements in most scenarios. The main contributions of this paper are summarized as follows:

- 1) The collision-free conditions for two general ellipsoids are derived analytically based on the separating hyper-

This work was supported in part by the National Natural Science Foundation of China under Grants 62373314/62222318, and in part by the Research Grants Council of the Hong Kong Special Administrative Region of China under Projects CityU/11210424 and CityU/11207323.

The authors are with the Department of Mechanical Engineering, City University of Hong Kong, Hong Kong, SAR, China (e-mail: zemingwu5-c@my.cityu.edu.hk; lulu45@cityu.edu.hk).

plane theorem and the concept of the dual cone.

- 2) Leveraging these analytical collision-free conditions, novel collision-free CBFs are proposed, with their validity rigorously proven.
- 3) A collision avoidance control method is developed that avoids the need for solving additional optimization problems, offering computational efficiency compared to double-level optimization methods.
- 4) Simulations and experiments are conducted to demonstrate the effectiveness and practicality of the proposed collision avoidance control method.

II. PRELIMINARIES

A. Notation

Throughout this paper, $\mathbb{R}$ denotes the set of real numbers, $\mathbb{R}^n$ denotes the set of n -dimension column vector over $\mathbb{R}$, and $\mathbb{R}^{n \times m}$ denotes the set of m -by- n matrices over $\mathbb{R}$. Non-bold symbols are used for scalars $a \in \mathbb{R}$, bold lowercase symbols for vectors $\mathbf{a} \in \mathbb{R}^n$ and bold uppercase symbols for matrices $\mathbf{A} \in \mathbb{R}^{n \times m}$. Specially, $\mathbf{I}_d$ denotes the d -dimension identity matrix, and $SO(d)$ denotes the d -dimension special orthogonal group defined as $SO(d) = \{\mathbf{R} \in \mathbb{R}^{d \times d} \mid \mathbf{R}^T \mathbf{R} = \mathbf{I}_d, \det(\mathbf{R}) = 1\}$.

Given a proper cone $\mathcal{K} \subseteq \mathbb{R}^d$, the general inequality $\preceq_{\mathcal{K}}$ is defined as,

$$\mathbf{x} \preceq_{\mathcal{K}} \mathbf{y} \iff \mathbf{y} - \mathbf{x} \in \mathcal{K}.$$

Specially, $\preceq$ denotes the inequalities introduced by nonnegative orthant cone $\mathbb{R}_+^d$, and $\mathbf{x} \preceq \mathbf{y}$ means $\mathbf{x}$ is component-wise less than or equal to $\mathbf{y}$. Given a cone $\mathcal{K} \subset \mathbb{R}^d$, then the set

$$\mathcal{K}^* = \{\mathbf{y} \in \mathbb{R}^d \mid \mathbf{y}^T \mathbf{x} \geq 0, \forall \mathbf{x} \in \mathcal{K}\}, \quad (1)$$

is called the dual cone of $\mathcal{K}$.

A function $\alpha : \mathbb{R} \rightarrow \mathbb{R}$ is said to be an *extended class $\mathcal{K}_{\infty}$ function* if it is strictly increasing and $\alpha(0) = 0$.

B. Control Barrier Function

Consider the following control affine system,

$$\dot{\mathbf{s}} = \mathbf{f}(\mathbf{s}) + \mathbf{g}(\mathbf{s})\mathbf{u}, \quad (2)$$

where $\mathbf{s} \in \mathbb{R}^n$ and $\mathbf{u} \in \mathbb{R}^m$ are the state and control, respectively. The functions $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^n$ and $\mathbf{g} : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times m}$ are locally Lipschitz. Let $h : \mathbb{R}^n \rightarrow \mathbb{R}$ be a continuously differentiable function, and define the *safe set* $\mathcal{S}$ as the zero super-level set of h , that is,

$$\mathcal{S} = \{\mathbf{s} \mid h(\mathbf{s}) \geq 0\}. \quad (3)$$

Then, a control barrier function can be defined as follows.

Definition 1 (Control Barrier Functions [19]). Let $\mathcal{S} \subset \bar{\mathcal{S}} \subset \mathbb{R}^n$ be the zero super-level set of a continuously differentiable function $h : \bar{\mathcal{S}} \rightarrow \mathbb{R}$. Then, h is said to be a *control barrier function* on $\bar{\mathcal{S}}$ if there exists an extended class $\mathcal{K}_{\infty}$ function α such that the time derivative of h along the trajectory of system (2) satisfies

$$\sup_{\mathbf{u} \in \mathbb{R}^m} \underbrace{L_{\mathbf{f}}h(\mathbf{s}) + L_{\mathbf{g}}h(\mathbf{s})\mathbf{u}}_{\dot{h}(\mathbf{s}, \mathbf{u})} \geq -\alpha(h(\mathbf{s})), \quad \forall \mathbf{s} \in \bar{\mathcal{S}}, \quad (4)$$

where $L_{\mathbf{f}}h(\cdot) = \frac{\partial h}{\partial \mathbf{s}}(\cdot)^T \mathbf{f}(\cdot)$ and $L_{\mathbf{g}}h(\cdot) = \frac{\partial h}{\partial \mathbf{s}}(\cdot)^T \mathbf{g}(\cdot)$ denote the Lie derivatives of h with respect to $\mathbf{f}$ and $\mathbf{g}$, respectively.

The existence of CBFs ensures the existence of controllers that guarantee the forward invariance of the safe set $\mathcal{S}$. This property is formally stated in the following lemma.

Lemma 1 ([19]). Let $\mathcal{S} \subset \bar{\mathcal{S}} \subset \mathbb{R}^n$ be a set defined as the zero super-level set of a function h such that:

- 1) $h : \bar{\mathcal{S}} \subset \mathbb{R}^n \rightarrow \mathbb{R}$ is continuously differentiable on $\bar{\mathcal{S}}$,
- 2) h is a control barrier function on $\bar{\mathcal{S}}$,
- 3) $\frac{\partial h}{\partial \mathbf{s}}(\mathbf{s}) \neq \mathbf{0}$ for all $\mathbf{s} \in \partial \mathcal{S}$.

Define the following set induced by $h(\mathbf{s})$:

$$\mathcal{U}_{cbf}(\mathbf{s}) = \left\{ \mathbf{u} \mid L_{\mathbf{f}}h(\mathbf{s}) + L_{\mathbf{g}}h(\mathbf{s})\mathbf{u} \geq -\alpha(h(\mathbf{s})) \right\}. \quad (5)$$

Then, any Lipschitz continuous controller $\mathbf{u}(\mathbf{s}) \in \mathcal{U}_{cbf}(\mathbf{s})$ for system (2) renders the set $\mathcal{S}$ forward invariant. Additionally, the set $\mathcal{S}$ is asymptotically stable in $\bar{\mathcal{S}}$.

Given a desired controller $\mathbf{u}^d(\mathbf{s})$, the forward invariance of the safe set $\mathcal{S}$ can be achieved by minimally modifying $\mathbf{u}^d(\mathbf{s})$ such that the condition (5) is satisfied. Noting that the condition (5) is a linear constraint on $\mathbf{u}$, the safe controller $\mathbf{u}^*$ can be computed via the following QP:

$$\mathbf{u}^* = \arg \min_{\mathbf{u}} \quad \frac{1}{2} \|\mathbf{u} - \mathbf{u}^d(\mathbf{s})\|_2^2 \quad (6a)$$

$$\text{s.t.} \quad L_{\mathbf{f}}h(\mathbf{s}) + L_{\mathbf{g}}h(\mathbf{s})\mathbf{u} \geq -\alpha(h(\mathbf{s})). \quad (6b)$$

III. COLLISION-FREE CONDITIONS FOR GENERAL ELLIPSOIDS

In this section, novel collision-free conditions between two general ellipsoids are proposed based on separating hyperplanes. The geometry of the i -th controlled object in the inertial frame is denoted as $\mathcal{G}_i$, which is defined as

$$\mathcal{G}_i = \{\mathbf{y} \mid \mathbf{y} = \mathbf{R}_i \mathbf{x} + \boldsymbol{\rho}_i, \mathbf{x} \in \mathcal{B}_i\}, \quad (7)$$

where $\mathcal{B}_i \subset \mathbb{R}^d$ is a compact convex set representing the geometry of the i -th controlled object in its body frame. Moreover, $\mathbf{R}_i \in SO(d)$ and $\boldsymbol{\rho}_i \in \mathbb{R}^d$ are the rotation matrix and translation vector, respectively, that transform the body frame of object i to the inertial frame. Note that $\mathbf{R}_i$ and $\boldsymbol{\rho}_i$ should be regarded as the state of object i . In this paper, $\mathcal{B}_i$ is assumed to be a general ellipsoid, which is defined as

$$\mathcal{B}_i = \{\mathbf{x} \in \mathbb{R}^d \mid \|\mathbf{Q}_i^{-1} \mathbf{x}\|_{p_i} \leq 1\}, \quad (8)$$

where $p_i > 1$ is the order of ellipsoid, and $\mathbf{Q}_i \in \mathbb{R}^{d \times d}$ is an invertible matrix.

To ensure the safety of objects i and j , their geometries must not overlap, i.e., $\mathcal{G}_i \cap \mathcal{G}_j = \emptyset$. This geometric constraint can be expressed mathematically using the following theorem:

Theorem 1 (Separating Hyperplane Theorem [20]). Let $\mathcal{G}_i$ and $\mathcal{G}_j$ be two nonempty disjoint convex sets in $\mathbb{R}^d$, i.e.,

$\mathcal{G}_i \cap \mathcal{G}_j = \emptyset$. Then, there exist a normal vector $\mathbf{n}_{ij} \in \mathbb{R}^d$, $\mathbf{n}_{ij} \neq \mathbf{0}$, and an offset $\gamma_{ij} \in \mathbb{R}$ such that

$$\mathbf{n}_{ij}^T \mathbf{y} \geq \gamma_{ij}, \quad \forall \mathbf{y} \in \mathcal{G}_i, \quad (9a)$$

$$\mathbf{n}_{ij}^T \mathbf{y} \leq \gamma_{ij}, \quad \forall \mathbf{y} \in \mathcal{G}_j. \quad (9b)$$

The hyperplane $\{\mathbf{y} \mid \mathbf{n}_{ij}^T \mathbf{y} = \gamma_{ij}\}$ is called a separating hyperplane for the sets $\mathcal{G}_i$ and $\mathcal{G}_j$.

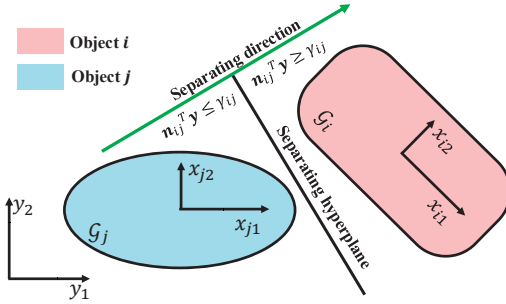


Fig. 1: An illustration of the separating hyperplane theorem.

The geometric illustration of Theorem 1 can be found in Fig. 1. Theorem 1 provides a key insight for collision avoidance between two general ellipsoids, namely, ensuring the existence of separating hyperplanes at all times. However, due to the universal quantifier in constraint (9), expressing the existence conditions of separating hyperplane $(\mathbf{n}_{ij}, \gamma_{ij})$ as the zero super-level sets of continuously differentiable functions remains an open problem. To address this gap, a new form of constraint (9) that without the universal quantifier is derived analytically using the dual cone of p -norm cone.

Lemma 2 ([20]). Given an order $p > 1$, the p -norm cone is defined as $\mathcal{K}_p = \{(\beta, \mathbf{z}) \mid \|\mathbf{z}\|_p \leq \beta\}$. Then its dual cone is $\mathcal{K}_p^* = \{(\lambda, \boldsymbol{\mu}) \mid \|\boldsymbol{\mu}\|_q \leq \lambda\}$ with $\frac{1}{q} + \frac{1}{p} = 1$.

Rather than directly deriving the existence conditions of separating hyperplanes for two disjoint general ellipsoids, we first characterize the conditions for a hyperplane that ensures an entire ellipsoid on one side of it, as stated in the following lemma.

Lemma 3. Given a general ellipsoid $\mathcal{G}_i$ described by (7) and (8), then the hyperplane plane

$$\mathcal{H} = \{\mathbf{y} \mid \mathbf{n}_{ij}^T \mathbf{y} = \gamma_{ij}\}, \quad (10a)$$

ensures entire $\mathcal{G}_i$ on one side of it in the following sense,

$$\mathbf{n}_{ij}^T \mathbf{y}_i \geq \gamma_{ij}, \quad \forall \mathbf{y}_i \in \mathcal{G}_i, \quad (10b)$$

if and only if $\mathbf{n}_{ij} \neq \mathbf{0}$ and

$$\|(\mathbf{R}_i \mathbf{Q}_i)^T \mathbf{n}_{ij}\|_{q_i} \leq \rho_i^T \mathbf{n}_{ij} - \gamma_{ij}, \quad (10c)$$

with $\frac{1}{q_i} + \frac{1}{p_i} = 1$.

Proof. Sufficiency ($\Rightarrow$): Since $\mathcal{G}_i$ is a general ellipsoid and thus bounded, there always exists a hyperplane $\mathcal{H} = \{\mathbf{y} \mid \mathbf{n}_{ij}^T \mathbf{y} = \gamma_{ij}\}$ that satisfies condition (10a). The assumption that $\mathcal{H}$ is a hyperplane implies $\mathbf{n}_{ij} \neq \mathbf{0}$. Using the

definition of the p -norm cone in Lemma 2, the ellipsoid $\mathcal{G}_i$ can be reformulated as

$$\mathcal{G}_i = \{\mathbf{y} \mid \mathbf{y} = \mathbf{R}_i \mathbf{x} + \boldsymbol{\rho}_i, \|\mathbf{Q}_i^{-1} \mathbf{x}\|_{p_i} \leq 1\} \quad (11a)$$

$$= \{\mathbf{y} \mid \mathbf{y} = \mathbf{R}_i \mathbf{Q}_i \mathbf{z} + \boldsymbol{\rho}_i, \|\mathbf{z}\|_{p_i} \leq 1\} \quad (11b)$$

$$= \{\mathbf{y} \mid \mathbf{y} = \mathbf{R}_i \mathbf{Q}_i \mathbf{z} + \boldsymbol{\rho}_i, \forall (1, \mathbf{z}) \in \mathcal{K}_{p_i}\}. \quad (11c)$$

The condition (10b) can then be reformulated as

$$\mathbf{n}_{ij}^T (\mathbf{R}_i \mathbf{Q}_i \mathbf{z}) + (\rho_i^T \mathbf{n}_{ij} - \gamma_{ij}) \cdot 1 \geq 0, \quad \forall (1, \mathbf{z}) \in \mathcal{K}_{p_i}. \quad (12a)$$

Since $\mathcal{K}_{p_i}$ is a cone, $(1, \mathbf{z}) \in \mathcal{K}_{p_i}$ and $\beta \geq 0$ imply $(\beta, \beta \mathbf{z}) \in \mathcal{K}_{p_i}$. Consequently, multiplying both sides of (12a) by β , the following inequality is obtained

$$\mathbf{n}_{ij}^T (\mathbf{R}_i \mathbf{Q}_i \tilde{\mathbf{z}}) + (\rho_i^T \mathbf{n}_{ij} - \gamma_{ij}) \cdot \beta \geq 0, \quad \forall (\beta, \tilde{\mathbf{z}}) \in \mathcal{K}_{p_i}, \quad (12b)$$

where $\tilde{\mathbf{z}} = \beta \mathbf{z}$. By the definition of the dual cone (1),

$$(\rho_i^T \mathbf{n}_{ij} - \gamma_{ij}, (\mathbf{R}_i \mathbf{Q}_i)^T \mathbf{n}_{ij}) \in \mathcal{K}_{p_i}^*. \quad (12c)$$

Finally, applying Lemma 2, the condition (10c) is obtained.

Necessity ($\Leftarrow$): If there exist a nonzero normal vector $\mathbf{n}_{ij}$ and an offset γ_{ij} such that condition (10c) is satisfied, then Lemma 2 implies

$$(\rho_i^T \mathbf{n}_{ij} - \gamma_{ij}, (\mathbf{R}_i \mathbf{Q}_i)^T \mathbf{n}_{ij}) \in \mathcal{K}_{p_i}^*. \quad (13a)$$

By the definition of dual cone, the following inequality holds:

$$\mathbf{n}_{ij}^T (\mathbf{R}_i \mathbf{Q}_i \mathbf{z}) + (\rho_i^T \mathbf{n}_{ij} - \gamma_{ij}) \cdot 1 \geq 0, \quad \forall (1, \mathbf{z}) \in \mathcal{K}_{p_i}. \quad (13b)$$

Rearranging the above inequality, we have

$$\mathbf{n}_{ij}^T (\mathbf{R}_i \mathbf{Q}_i \mathbf{z} + \boldsymbol{\rho}_i) \geq \gamma_{ij}, \quad \forall (1, \mathbf{z}) \in \mathcal{K}_{p_i}. \quad (13c)$$

Noting that $(1, \mathbf{z}) \in \mathcal{K}_{p_i} \implies (\mathbf{R}_i \mathbf{Q}_i \mathbf{z} + \boldsymbol{\rho}_i) \in \mathcal{G}_i$ according to (11c), we obtain the final inequality:

$$\mathbf{n}_{ij}^T \mathbf{y} \geq \gamma_{ij}, \quad \forall \mathbf{y} \in \mathcal{G}_i. \quad (13d)$$

The proof is complete. $\blacksquare$

Based on Lemma 3, the existence conditions of separating hyperplanes for two disjoint general ellipsoids is characterized by the following theorem.

Theorem 2. Let $\mathcal{G}_i$ and $\mathcal{G}_j$ be two disjoint general ellipsoids described by (7) and (8). Then, $\mathcal{H} = \{\mathbf{y} \mid \mathbf{n}_{ij}^T \mathbf{y} = \gamma_{ij}\}$ is a separating hyperplane for $\mathcal{G}_i$ and $\mathcal{G}_j$ in the following sense:

$$\mathbf{n}_{ij}^T \mathbf{y}_i \geq \gamma_{ij} \geq \mathbf{n}_{ij}^T \mathbf{y}_j, \quad \forall \mathbf{y}_i \in \mathcal{G}_i, \forall \mathbf{y}_j \in \mathcal{G}_j, \quad (14)$$

if and only if there exists a nonzero normal vector $\mathbf{n}_{ij} \neq \mathbf{0}$ and an offset γ_{ij} such that

$$\|(\mathbf{R}_i \mathbf{Q}_i)^T \mathbf{n}_{ij}\|_{q_i} \leq \rho_i^T \mathbf{n}_{ij} - \gamma_{ij}, \quad (15a)$$

$$\|-(\mathbf{R}_j \mathbf{Q}_j)^T \mathbf{n}_{ij}\|_{q_j} \leq -\rho_j^T \mathbf{n}_{ij} + \gamma_{ij}. \quad (15b)$$

Proof. According to Lemma 3, the first inequality in (14) holds if and only if condition (15a) is satisfied. It remains to prove the second inequality in (14), which is equivalent to the following inequality:

$$-\mathbf{n}_{ij}^T \mathbf{y}_j \geq -\gamma_{ij}, \quad \forall \mathbf{y}_j \in \mathcal{G}_j. \quad (16a)$$

Applying Lemma 3, the inequality (16a) holds if and only if

$$\| -(\mathbf{R}_j \mathbf{Q}_j)^T \mathbf{n}_{ij} \|_{q_i} \leq -\rho_i^T \mathbf{n}_{ij} + \gamma_{ij}, \quad (16b)$$

which is equivalent to (15b). The proof is complete. $\blacksquare$

Theorem 2 provides the analytical form of the feasible set of separating hyperplanes for two general ellipsoids. By incorporating the constraints (15) into the CBFs framework, collision avoidance between two general ellipsoids can be achieved. However, while the constraints (15) can be incorporated into the CBFs framework, ensuring $\mathbf{n}_{ij} \neq \mathbf{0}$ remains a challenge. This challenge will be addressed in the next section.

IV. COLLISION AVOIDANCE CONTROL VIA CBFs WITH SEPARATING HYPERPLANES

In this section, a novel collision avoidance control method is proposed by incorporating the collision-free conditions (15) into the CBF framework.

The object i is assumed to be fully controlled, more precisely, the dynamics of rotation matrix $\mathbf{R}_i$ and translation vector ρ_i is described by,

$$\dot{\mathbf{R}}_i = \mathbf{R}_i \hat{\boldsymbol{\omega}}_i, \quad \dot{\rho}_i = \mathbf{R}_i \mathbf{v}_i, \quad (17)$$

where $\boldsymbol{\omega}_i \in \mathbb{R}^{\frac{d(d-1)}{2}}$ and $\mathbf{v}_i \in \mathbb{R}^d$ are the angular velocity and translational velocity of object i in its body frame. And $\wedge$ is the operator that maps a vector to a screw symmetric matrix, for examples, given $\omega \in \mathbb{R}$ and $\boldsymbol{\omega} = (\omega_1, \omega_2, \omega_3) \in \mathbb{R}^3$, the corresponding operator $\wedge$ is defined as,

$$\hat{\omega} = \begin{bmatrix} 0 & -\omega \\ \omega & 0 \end{bmatrix}, \quad \hat{\boldsymbol{\omega}} = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}. \quad (18)$$

To evaluate the collision-free conditions (15) between two convex primitives, a separating hyperplane is required. A natural approach is to extend the state of the system with a nonzero normal vector $\mathbf{n}_{ij} \in \mathbb{R}^d$ and an offset $\gamma_{ij} \in \mathbb{R}$. However, fulfilling the requirement $\mathbf{n}_{ij} \neq \mathbf{0}$ without introducing conservatism remains a challenge. In this paper, this challenge is addressed by enforcing the normal vector to be a unit vector, i.e., $\mathbf{n}_{ij}^T \mathbf{n}_{ij} = 1$. This approach does not introduce any conservatism because the hyperplane $\{\mathbf{y} \mid \mathbf{n}_{ij}^T \mathbf{y} = \gamma_{ij}\}$ is equivalent to $\{\mathbf{y} \mid \hat{\mathbf{n}}_{ij}^T \mathbf{y} = \hat{\gamma}_{ij}\}$, where $\hat{\mathbf{n}}_{ij} = \mathbf{n}_{ij} / \|\mathbf{n}_{ij}\|_2$ and $\hat{\gamma}_{ij} = \gamma_{ij} / \|\mathbf{n}_{ij}\|_2$.

To preserve the norm of $\mathbf{n}_{ij}$, the dynamics of the normal vector $\mathbf{n}_{ij}$ and the offset γ_{ij} are designed as

$$\dot{\mathbf{n}}_{ij} = (\mathbf{I}_d - \mathbf{n}_{ij} \mathbf{n}_{ij}^T) \boldsymbol{\eta}_{ij}, \quad \dot{\gamma}_{ij} = \delta_{ij}, \quad (19)$$

where $\boldsymbol{\eta}_{ij} \in \mathbb{R}^d$ and $\delta_{ij} \in \mathbb{R}$ are the control inputs for $\mathbf{n}_{ij}$ and γ_{ij} , respectively. The above dynamics ensure that the norm of $\mathbf{n}_{ij}$ remains unchanged, as the time derivative of $\mathbf{n}_{ij}^T \mathbf{n}_{ij}$ satisfies

$$\begin{aligned} 2\mathbf{n}_{ij}^T \dot{\mathbf{n}}_{ij} &= 2\mathbf{n}_{ij}^T (\mathbf{I}_d - \mathbf{n}_{ij} \mathbf{n}_{ij}^T) \boldsymbol{\eta}_{ij} \\ &= 2(\mathbf{n}_{ij}^T - \mathbf{n}_{ij}^T \mathbf{n}_{ij}) \boldsymbol{\eta}_{ij} = 0. \end{aligned} \quad (20)$$

For notational simplicity and clarity, intermediate variables $\lambda_i, \lambda_j \in \mathbb{R}$ and $\boldsymbol{\mu}_i, \boldsymbol{\mu}_j \in \mathbb{R}^d$ are defined as follows:

$$\lambda_i = \rho_i^T \mathbf{n}_{ij} - \gamma_{ij}, \quad \boldsymbol{\mu}_i = (\mathbf{R}_i \mathbf{Q}_i)^T \mathbf{n}_{ij}, \quad (21a)$$

$$\lambda_j = -\rho_j^T \mathbf{n}_{ij} + \gamma_{ij}, \quad \boldsymbol{\mu}_j = -(\mathbf{R}_j \mathbf{Q}_j)^T \mathbf{n}_{ij}. \quad (21b)$$

Following the discussion in Section III, the collision-free conditions (15) can be expressed as

$$h_i(\lambda_i, \boldsymbol{\mu}_i) = \lambda_i - \|\boldsymbol{\mu}_i\|_{q_i} \geq 0, \quad (22a)$$

$$h_j(\lambda_j, \boldsymbol{\mu}_j) = \lambda_j - \|\boldsymbol{\mu}_j\|_{q_j} \geq 0. \quad (22b)$$

To incorporate the above constraints into the CBFs framework, the time derivatives of functions h_i and h_j are required, which are given by the following lemma.

Lemma 4. *The time derivatives of functions $h_i(\lambda_i, \boldsymbol{\mu}_i)$ and $h_j(\lambda_j, \boldsymbol{\mu}_j)$ along the system dynamics (17), (19) and (21) are derived as,*

$$\dot{h}_i = \mathbf{a}_i^T \boldsymbol{\omega}_i + \mathbf{b}_i^T \mathbf{v}_i + \mathbf{c}_i^T \boldsymbol{\eta}_{ij} + \mathbf{d}_i^T \delta_{ij}, \quad (23a)$$

$$\dot{h}_j = \mathbf{a}_j^T \boldsymbol{\omega}_j + \mathbf{b}_j^T \mathbf{v}_j + \mathbf{c}_j^T \boldsymbol{\eta}_{ij} + \mathbf{d}_j^T \delta_{ij}, \quad (23b)$$

where the coefficient vectors are detailed as

$$\mathbf{b}_i = \mathbf{R}_i^T \mathbf{n}_{ij} \frac{\partial h_i}{\partial \lambda_i}, \quad \mathbf{d}_i = -\frac{\partial h_i}{\partial \lambda_i},$$

$$\mathbf{c}_i = (\mathbf{I}_d - \mathbf{n}_{ij} \mathbf{n}_{ij}^T) \left(\rho_i \frac{\partial h_i}{\partial \lambda_i} + \mathbf{R}_i \mathbf{Q}_i \frac{\partial h_i}{\partial \boldsymbol{\mu}_i} \right),$$

$$\mathbf{b}_j = -\mathbf{R}_j^T \mathbf{n}_{ij} \frac{\partial h_j}{\partial \lambda_j}, \quad \mathbf{d}_j = \frac{\partial h_j}{\partial \lambda_j},$$

$$\mathbf{c}_j = -(\mathbf{I}_d - \mathbf{n}_{ij} \mathbf{n}_{ij}^T) \left(\rho_j \frac{\partial h_j}{\partial \lambda_j} + \mathbf{R}_j \mathbf{Q}_j \frac{\partial h_j}{\partial \boldsymbol{\mu}_j} \right).$$

For $d=2$, the coefficient vectors $\mathbf{a}_i$ and $\mathbf{a}_j$ are given as

$$\mathbf{a}_i = \mathbf{n}_{ij}^T \mathbf{R}_i \hat{\mathbf{Q}}_i \frac{\partial h_i}{\partial \boldsymbol{\mu}_i}, \quad (25a)$$

$$\mathbf{a}_j = -\mathbf{n}_{ij}^T \mathbf{R}_j \hat{\mathbf{Q}}_j \frac{\partial h_j}{\partial \boldsymbol{\mu}_j}. \quad (25b)$$

For $d=3$, the coefficient vectors $\mathbf{a}_i$ and $\mathbf{a}_j$ are given as

$$\mathbf{a}_i = (\widehat{\mathbf{R}_i^T \mathbf{n}_{ij}})^T \mathbf{Q}_i \frac{\partial h_i}{\partial \boldsymbol{\mu}_i}, \quad (26a)$$

$$\mathbf{a}_j = -(\widehat{\mathbf{R}_j^T \mathbf{n}_{ij}})^T \mathbf{Q}_j \frac{\partial h_j}{\partial \boldsymbol{\mu}_j}. \quad (26b)$$

Proof. The proof is given in Appendix A. $\blacksquare$

Note that, the differences between (25a) and (26a) are caused by the slightly differences of the $\wedge$ operator between $d=2$ and $d=3$ as shown in (18).

To establish that the collision-free CBFs (22) constitute valid control barrier functions, we must verify that their coefficient vectors are non-zero and continuously differentiable with respect to the system state. These properties are critical to ensure the CBFs are well-defined and can reliably enforce safety constraints within the control framework. A detailed discussion of their necessity is provided in [17, Remark 5]. The validity of the proposed CBFs is formally established by the following theorem.

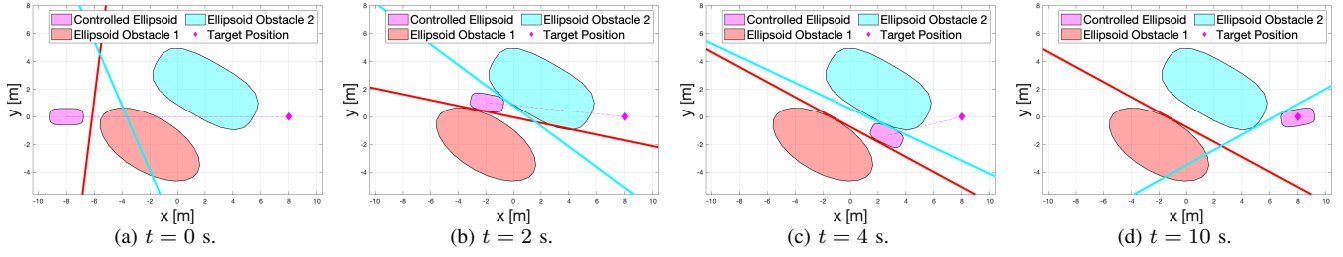


Fig. 2: The snapshots of the simulation with the proposed collision avoidance method. The controlled ellipsoid navigates to its target position while avoiding collisions with the two ellipsoid obstacles.

Theorem 3. Define the safe set $\mathcal{S}_{ij}$ as

$$\mathcal{S}_{ij} = \left\{ (\mathbf{R}_i, \boldsymbol{\rho}_i, \mathbf{R}_j, \boldsymbol{\rho}_j, \mathbf{n}_{ij}, \gamma_{ij}) \mid h_i \geq 0, h_j \geq 0 \right\}.$$

Then, h_i and h_j are valid CBFs for the dynamics (17) and (19) on $\mathcal{S}_{ij}$, as formalized below:

- 1) The coefficient vectors $(\mathbf{a}_i, \mathbf{b}_i, \mathbf{c}_i, \mathbf{d}_i)$ and $(\mathbf{a}_j, \mathbf{b}_j, \mathbf{c}_j, \mathbf{d}_j)$ given by Lemma 4 are continuous.
- 2) The coefficient vectors $(\mathbf{a}_i, \mathbf{b}_i, \mathbf{c}_i, \mathbf{d}_i)$ and $(\mathbf{a}_j, \mathbf{b}_j, \mathbf{c}_j, \mathbf{d}_j)$ will not vanish, i.e., become zero vectors.
- 3) There exists at least one control input that simultaneously satisfies $\dot{h}_i \geq -\alpha(h_i)$ and $\dot{h}_j \geq -\alpha(h_j)$.

Proof. The proof is provided in Appendix B. ■

The above theorem also ensures that, for any states inside the safe set $\mathcal{S}_{ij}$, at least one control input can be found such that renders the set $\mathcal{S}_{ij}$ forward invariant.

Given the nominal controller $\mathbf{u}_{ij}^d = (\boldsymbol{\omega}_i^d, \mathbf{v}_i^d, \boldsymbol{\omega}_j^d, \mathbf{v}_j^d)$ for two controlled objects, the collision-free control input $\mathbf{u}_{ij}^*$ for the controlled objects and the control input $(\boldsymbol{\eta}_{ij}^*, \delta_{ij}^*)$ for the separating hyperplane can be obtained by solving the following QP:

$$\min_{\boldsymbol{\eta}_{ij}, \delta_{ij}} \frac{1}{2} \|\mathbf{u}_{ij} - \mathbf{u}_{ij}^d\|_2^2 \quad (27a)$$

$$\text{s.t.} \quad \dot{h}_i \geq -\alpha(h_i), \quad (27b)$$

$$\dot{h}_j \geq -\alpha(h_j), \quad (27c)$$

where $\dot{h}_i$ and $\dot{h}_j$ are given by (23).

It is worth noting that the control input $(\boldsymbol{\eta}_{ij}, \delta_{ij})$ for the separating hyperplane is not incorporated into the objective function of the optimization problem (27). This is because the normal vector $\mathbf{n}_{ij}$ and offset γ_{ij} are virtual states that should adapt passively according to collision avoidance demands. Since the objective function in the optimization problem (27) is positive semi-definite, there may be infinitely many optimal solutions for $(\boldsymbol{\eta}_{ij}, \delta_{ij})$. This is not problematic, as all such control inputs ensure collision-free behavior. Moreover, the Lipschitz continuity of the control input can be guaranteed by selecting the optimal solution with the minimum norm.

V. VALIDATION AND COMPARISONS

In this section, numerical simulations and real-world experiments are conducted to demonstrate the effectiveness of the proposed algorithm. The proposed algorithm is also compared with the following state-of-the-art collision avoidance algorithms based on CBFs.

A. Simulation

Numerical simulations are conducted to verify the efficacy of the proposed collision avoidance control method. The simulation scenario involves one fully controlled general ellipsoid $\mathcal{G}_0$, whose dynamics follow (17). The initial position of $\mathcal{G}_0$ is $[-8; 0]$, and its initial rotation matrix is $[1, 0; 0, 1]$. The geometric parameters of $\mathcal{G}_0$ are set as $\mathbf{Q}_0 = [1.2, 0; 0, 0.6]$ and $p_0 = 4$. Additionally, two ellipsoid obstacles, $\mathcal{G}_1$ and $\mathcal{G}_2$, are centered at $\boldsymbol{\rho}_1 = [-2; -2]$ and $\boldsymbol{\rho}_2 = [2; 2]$, respectively. Their rotation matrices are $\mathbf{R}_1 = \mathbf{R}_2 = [\cos(\pi/6), \sin(\pi/6); -\sin(\pi/6), \cos(\pi/6)]$, with parameter matrices $\mathbf{Q}_1 = \mathbf{Q}_2 = [4, 0; 0, 2]$ and orders $p_1 = 2$ and $p_2 = 3$. The control objective is to drive $\mathcal{G}_0$ to the target position $\boldsymbol{\rho}_0^d = [8; 0]$ while avoiding collisions with the obstacles. To achieve this, the nominal controllers for $\mathcal{G}_0$ are designed as $\boldsymbol{\omega}_0^d = \mathbf{0}$ and $\mathbf{v}_0^d = -k_\rho \mathbf{R}_0^T (\boldsymbol{\rho}_0 - \boldsymbol{\rho}_0^d)$, where $k_\rho = 0.3$. The class $\mathcal{K}_\infty$ function is designed as $\alpha(h) = 20h$.

Snapshots of the simulation are shown in Fig. 2. The two lines represent the separating hyperplanes between $\mathcal{G}_0$ and the two static ellipsoids, with each hyperplane colored to match its corresponding obstacle. The hyperplanes adapt passively to the motion of $\mathcal{G}_0$. As demonstrated in the snapshots, the hyperplanes consistently separate $\mathcal{G}_0$ from the obstacles, ensuring collision-free navigation.

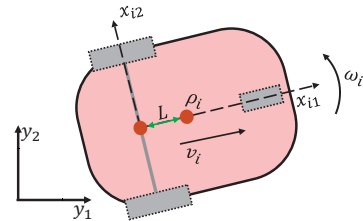


Fig. 3: The nonholonomic vehicle model. The control inputs are the angular velocity ω_i and the forward translational velocity of the rear axle v_i in the body frame of the vehicle.

B. Experiment

Experiments based on the Robotarium platform [21] are conducted to demonstrate that our control method can be

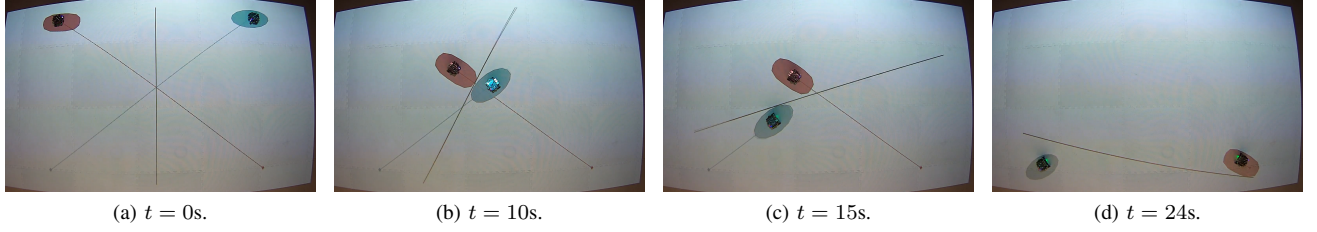


Fig. 4: Snapshots of the experiment with two nonholonomic vehicles. During the navigation, a separating hyperplane, colored in black, always separates the two vehicles. The vehicles navigate to their target positions without collision.

applied to vehicles with nonholonomic dynamics. For the case $d = 2$, the nonholonomic dynamics are described by:

$$\dot{R}_i = R_i \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \omega_i, \quad (28a)$$

$$\dot{\rho}_i = R_i \cdot \begin{bmatrix} 1 & 0 \\ 0 & L \end{bmatrix} \cdot \begin{bmatrix} v_i \\ \omega_i \end{bmatrix}, \quad (28b)$$

where $\omega_i \in \mathbb{R}$ is the angular velocity in the body frame of vehicle i , $v_i \in \mathbb{R}$ is the forward translational velocity of the rear axle in its body frame, and $L \in \mathbb{R}$ is the distance from the center of the rear axle to the geometric center ρ_i , as illustrated in Fig. 3. The time derivatives of the CBFs (22) under the dynamics (28) can be derived using the procedures outlined in Appendix A.

The control objective of the experiment is to drive the geometric center ρ_i of vehicle i to its target position ρ_i^d . This is achieved using the following nominal controller:

$$\begin{bmatrix} v_i^d \\ \omega_i^d \end{bmatrix} = -k_p \cdot \begin{bmatrix} 1 & 0 \\ 0 & 1/L \end{bmatrix} \cdot R_i^T \cdot (\rho_i - \rho_i^d), \quad k_p > 0. \quad (29)$$

Fig. 4a shows the initial configurations of the two differential-driven vehicles, each stabilized at its target position, marked with the corresponding color. Fig. 5 depicts the evolution of the CBFs during the experiment. Since the values of the proposed CBFs remain greater than 0 throughout the experiment, no collisions occur between the two vehicles.

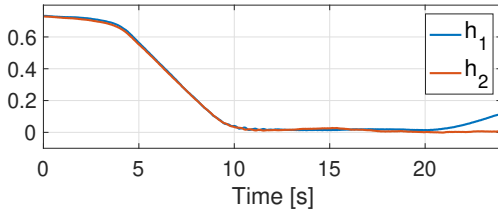


Fig. 5: The evolution of CBFs in the experiment.

C. Benchmark Comparisons

To demonstrate the advantages in computational efficiency, the proposed collision avoidance control method is compared with the following state-of-the-art CBF-based collision avoidance methods:

- 1) The duality-based safety-critical control (DB-CBF) [13] approximates the time derivative of CBFs (the Euclidean distance between two objects) using the Lagrange dual problem. This method requires solving a primal problem to obtain the optimal dual variables, in addition to solving the CBF-QP problem, resulting in a double-level optimization process. It is worth noting that the original method is designed for polyhedra and has been customized for general ellipsoids according to the paradigm in [11].
- 2) The differentiable optimization-based CBFs (DO-CBF) [14] derive the time derivative of CBFs (the growth distance [16] between two objects) from the KKT conditions. Since the KKT conditions require solving a primal problem first, this method also involves a double-level optimization process. Unlike our method and the method in [13], this approach does not introduce additional virtual states into the CBF-QP, resulting in a lower-dimensional QP to solve.

All methods are evaluated on a MacBook Pro laptop with an M4 Pro chip and 18 GB of RAM. The optimization problems, including the CBF-QP (27) and the primal problems of DB-CBF and DO-CBF, are solved using the MOSEK software (version 10.2) with MATLAB interfaces [22].

In each simulation, the scenario is designed as the random position stabilization of 10 ellipsoids under the dynamics (17) with $d = 2$. To achieve collision avoidance between each pair of agents, the DB-CBF method must solve 45 primal problems, which are conic optimization problems with a dimension of 4. Similarly, the DO-CBF method must solve 45 conic optimization problems with a dimension of 3. All these primal problems are solved sequentially.

In addition to the primal problem, all these methods must solve a CBF-QP to obtain safe control inputs. The DB-CBF method introduces 45×4 virtual control inputs in addition to 30 physical control inputs, resulting in a CBF-QP with a dimension of 210. Moreover, 45 extra constraints are introduced into the CBF-QP in addition to the 45 CBF constraints. The DO-CBF method does not introduce any additional virtual control inputs, and only 45 CBF constraints are needed. The proposed method introduces 3 virtual control inputs for each pair of agents and requires 2 CBF constraints for each pair, resulting in a CBF-QP with a dimension of 165 and 90 constraints.

TABLE I: Benchmark Comparison

Method	Primal Problem		CBF-QP			Total Time
	Dim.	Avg. Time	Dim.	Constr.	Avg. Time	
DB-CBF [13]	45 × 4	13.8 ms	210	90	14.7 ms	32.2 ms
DO-CBF [14]	45 × 3	9.2 ms	30	45	3.3 ms	14.1 ms
Proposed	—	—	165	90	10.9 ms	11.8 ms

The bold entities present the best performance results for each column.

Table I reports the dimensions of the optimization problems, the number of constraints, and the average computational times for the different methods. As expected, the total computational time of the DB-CBF method is longer than that of the DO-CBF method and our proposed method, as it involves two high-dimensional optimization problems. Although the proposed method has a CBF-QP with a higher dimension, it still outperforms the DO-CBF method in this scenario due to its single-level optimization feature.

VI. CONCLUSIONS AND FUTURE WORKS

This paper has proposed a collision avoidance control method for general ellipsoids based on the separating hyperplane theorem. Collision-free CBFs are analytically constructed using collision-free conditions derived from the dual cone. Thanks to the analytical form of the CBFs, the proposed collision avoidance method does not require solving additional optimization problems beyond the CBF-QP, which expedites the process for achieving safe control. Simulations and experiments have been conducted with various system dynamics to verify the effectiveness and extendability of the proposed method.

Future work includes extending the method from general ellipsoids to general convex primitives, implementing the proposed control method in a distributed manner, and actively driving the hyperplane to avoid potential deadlocks between objects.

APPENDIX

A. Proof of Lemma 4

The time derivatives of functions h_i and h_j are given by

$$\begin{aligned}\dot{h}_i &= \frac{\partial h_i}{\partial \lambda_i} \dot{\lambda}_i + \frac{\partial h_i}{\partial \mu_i} \dot{\mu}_i, \\ \dot{h}_j &= \frac{\partial h_j}{\partial \lambda_j} \dot{\lambda}_j + \frac{\partial h_j}{\partial \mu_j} \dot{\mu}_j.\end{aligned}\quad (30)$$

Taking the time derivative on both sides of (21), the time derivatives of the intermediate variable λ_i are obtained as

$$\begin{aligned}\dot{\lambda}_i &= \mathbf{n}_{ij}^T \dot{\mathbf{p}}_i + \rho_i^T \dot{\mathbf{n}}_{ij} + \dot{\gamma}_{ij} \\ &= \mathbf{n}_{ij}^T \mathbf{R}_i \mathbf{v}_i + \rho_i^T (\mathbf{I}_d - \mathbf{n}_{ij} \mathbf{n}_{ij}^T) \eta_{ij} - \delta_{ij}.\end{aligned}$$

Similarly, the time derivative of intermediate variable λ_j is

$$\dot{\lambda}_j = -\mathbf{n}_{ij}^T \mathbf{R}_i \mathbf{v}_j - \rho_j^T (\mathbf{I}_d - \mathbf{n}_{ij} \mathbf{n}_{ij}^T) \eta_{ij} + \delta_{ij}.$$

Regarding the time derivative of intermediate variable μ_i ,

$$\begin{aligned}\dot{\mu}_i &= \mathbf{Q}_i^T \dot{\mathbf{R}}_i^T \mathbf{n}_{ij} + \mathbf{Q}_i^T \mathbf{R}_i^T \dot{\mathbf{n}}_{ij} \\ &= \mathbf{Q}_i^T \dot{\omega}_i^T \mathbf{R}_i^T \mathbf{n}_{ij} + \mathbf{Q}_i^T \mathbf{R}_i^T (\mathbf{I}_d - \mathbf{n}_{ij} \mathbf{n}_{ij}^T) \eta_{ij}.\end{aligned}$$

For the case where $d = 2$, according to the definition of the operation $\wedge$, we have $\widehat{\omega}_i^T \mathbf{R}_i^T \mathbf{n}_{ij} = \widehat{\mathbf{1}}^T \mathbf{R}_i^T \mathbf{n}_{ij} \omega_i$, that is,

$$\dot{\mu}_i = (\mathbf{R}_i \widehat{\mathbf{1}} \mathbf{Q}_i)^T \mathbf{n}_{ij} \omega_i + (\mathbf{R}_i \mathbf{Q}_i)^T (\mathbf{I}_d - \mathbf{n}_{ij} \mathbf{n}_{ij}^T) \eta_{ij}.$$

For the case where $d = 3$, according to the definition of the operation $\wedge$, we have $\widehat{\omega}_i^T \mathbf{R}_i^T \mathbf{n}_{ij} = (\mathbf{R}_i^T \widehat{\mathbf{n}}_{ij}) \omega_i$, that is,

$$\dot{\mu}_i = \mathbf{Q}_i^T (\widehat{\mathbf{R}_i^T \mathbf{n}_{ij}}) \omega_i + (\mathbf{R}_i \mathbf{Q}_i)^T (\mathbf{I}_d - \mathbf{n}_{ij} \mathbf{n}_{ij}^T) \eta_{ij}.$$

Similar results for the intermediate variable μ_j can be obtained. For the case where $d = 2$,

$$\dot{\mu}_j = -(\mathbf{R}_j \widehat{\mathbf{1}} \mathbf{Q}_j)^T \mathbf{n}_{ij} \omega_j - (\mathbf{R}_j \mathbf{Q}_j)^T (\mathbf{I}_d - \mathbf{n}_{ij} \mathbf{n}_{ij}^T) \eta_{ij}.$$

For the case where $d = 3$,

$$\dot{\mu}_j = -\mathbf{Q}_j^T (\widehat{\mathbf{R}_j^T \mathbf{n}_{ij}}) \omega_j - (\mathbf{R}_j \mathbf{Q}_j)^T (\mathbf{I}_d - \mathbf{n}_{ij} \mathbf{n}_{ij}^T) \eta_{ij}.$$

Substituting the time derivatives of the intermediate variables into (30), the coefficient matrices can be obtained. The proof is complete.

B. Proof of Theorem 3

Due to the similarity in proving the conclusions for h_i and h_j , we only prove the results for h_i . The same results can be obtained straightforwardly for h_j .

According to Lemma 4, the coefficient vector $(\mathbf{a}_i, \mathbf{b}_i, \mathbf{c}_i, \mathbf{d}_i)$ is a continuous function with respect to the state of the system if and only if $\partial h_i / \partial \lambda_i$ and $\partial h_i / \partial \mu_i$ are continuous functions with respect to (λ_i, μ_i) . It can be verified that $\partial h_i / \partial \lambda_i = 1$ is always continuous, while $\partial h_i / \partial \mu_i$ is continuous unless $\mu_i = \mathbf{0}$. By the definition of μ_i , the following equality holds:

$$\mu_i^T \mu_i = \mathbf{n}_{ij}^T (\mathbf{R}_i \mathbf{Q}_i) (\mathbf{R}_i \mathbf{Q}_i)^T \mathbf{n}_{ij}. \quad (31)$$

The matrix $(\mathbf{R}_i \mathbf{Q}_i) (\mathbf{R}_i \mathbf{Q}_i)^T$ is positive definite since $\mathbf{R}_i$ and $\mathbf{Q}_i$ are both invertible matrices. Furthermore, the vector $\mathbf{n}_{ij} \neq \mathbf{0}$ because the dynamics (19) preserve its norm. Consequently, $\mu_i^T \mu_i > 0$, and thus $\mu_i \neq \mathbf{0}$. This proves that $\partial h_i / \partial \mu_i$ is always continuous under the system dynamics (19), which establishes the first point.

By Lemma 4, we have $\mathbf{d}_i = (\partial h_i / \partial \lambda_i)^T = 1$. Consequently, the coefficient vectors $(\mathbf{a}_i, \mathbf{b}_i, \mathbf{c}_i, \mathbf{d}_i)$ cannot become zero, which proves the second point.

Note that $0 \geq -\alpha(h_i)$ and $0 \geq -\alpha(h_j)$ on $\mathcal{S}_{ij}$. As a result, the control input $(\omega_i, \mathbf{v}_i, \omega_j, \mathbf{v}_j, \eta_{ij}, \delta_{ij}) = \mathbf{0}$ is always feasible according to (23). The proof is complete.

REFERENCES

- [1] L. Ferranti, L. Lyons, R. R. Negenborn, T. Keviczky, and J. Alonso-Mora, "Distributed nonlinear trajectory optimization for multi-robot motion planning," *IEEE Trans. Control Syst. Technol.*, vol. 31, no. 2, pp. 809–824, 2022.
- [2] Z. Huang, S. Shen, and J. Ma, "Decentralized iLQR for cooperative trajectory planning of connected autonomous vehicles via dual consensus admm," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 11, pp. 12 754–12 766, 2023.
- [3] R. Firoozi, L. Ferranti, X. Zhang, S. Nejadnik, and F. Borrelli, "A distributed multi-vehicle coordination algorithm for navigation in tight environments," *IEEE Trans. Veh. Technol.*, vol. 73, no. 10, pp. 14 499–14 509, 2024.
- [4] B. Dai, R. Khorrambakht, P. Krishnamurthy, and F. Khorrami, "Sailing through point clouds: Safe navigation using point cloud based control

- barrier functions,” *IEEE Robot. Autom. Lett.*, vol. 9, no. 9, pp. 7731–7738, 2024.
- [5] Y. Wen and P. Pagilla, “Path-constrained and collision-free optimal trajectory planning for robot manipulators,” *IEEE Trans. Autom. Sci. Eng.*, vol. 20, no. 2, pp. 763–774, 2022.
 - [6] A. Singletary, W. Guffey, T. G. Molnar, R. Sinnet, and A. D. Ames, “Safety-critical manipulation for collision-free food preparation,” *IEEE Robot. Autom. Lett.*, vol. 7, no. 4, pp. 10954–10961, 2022.
 - [7] A. U. Raghunathan, D. K. Jha, and D. Romeres, “Pyrobocop: Python-based robotic control and optimization package for manipulation and collision avoidance,” *IEEE Trans. Autom. Sci. Eng.*, vol. 22, pp. 1435–1450, 2024.
 - [8] S. Wei, B. Dai, R. Khorrambakht, P. Krishnamurthy, and F. Khorrami, “Diffocclusion: Differentiable optimization based control barrier functions for occlusion-free visual servoing,” *IEEE Robot. Autom. Lett.*, vol. 9, no. 4, pp. 3235–3242, 2024.
 - [9] Q. Wang, Z. Wang, L. Pei, C. Xu, and F. Gao, “A linear and exact algorithm for whole-body collision evaluation via scale optimization,” in *Proc. IEEE Int. Conf. Robot. Autom.* IEEE, 2023, pp. 3621–3627.
 - [10] K. Tracy, T. A. Howell, and Z. Manchester, “Differentiable collision detection for a set of convex primitives,” in *Proc. IEEE Int. Conf. Robot. Autom.* IEEE, 2023, pp. 3663–3670.
 - [11] X. Zhang, A. Liniger, and F. Borrelli, “Optimization-based collision avoidance,” *IEEE Trans. Control Syst. Technol.*, vol. 29, no. 3, pp. 972–983, 2020.
 - [12] Z. Wu, Z. Wang, and H. Zhang, “GPU-accelerated optimization-based collision avoidance,” in *Proc. IEEE Int. Conf. Robot. Autom.* IEEE, 2024, pp. 7561–7567.
 - [13] A. Thirugnanam, J. Zeng, and K. Sreenath, “Duality-based convex optimization for real-time obstacle avoidance between polytopes with control barrier functions,” in *Proc. Am. Control Conf.* IEEE, 2022, pp. 2239–2246.
 - [14] B. Dai, R. Khorrambakht, P. Krishnamurthy, V. Gonçalves, A. Tzes, and F. Khorrami, “Safe navigation and obstacle avoidance using differentiable optimization based control barrier functions,” *IEEE Robot. Autom. Lett.*, vol. 8, no. 9, pp. 5376–5383, 2023.
 - [15] E. G. Gilbert, D. W. Johnson, and S. S. Keerthi, “A fast procedure for computing the distance between complex objects in three-dimensional space,” *IEEE J. Robot. Autom.*, vol. 4, no. 2, pp. 193–203, 2002.
 - [16] C. J. Ong and E. G. Gilbert, “Growth distances: New measures for object separation and penetration,” *IEEE Trans. Robot. Autom.*, vol. 12, no. 6, pp. 888–903, 1996.
 - [17] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, “Control barrier functions: Theory and applications,” in *Proc. IEEE Eur. Control Conf.* IEEE, 2019, pp. 3420–3431.
 - [18] Z. Han, Z. Wang, N. Pan, Y. Lin, C. Xu, and F. Gao, “Fast-racing: An open-source strong baseline for SE(3) planning in autonomous drone racing,” *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 8631–8638, 2021.
 - [19] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, “Control barrier function based quadratic programs for safety critical systems,” *IEEE Trans. Autom. Control*, vol. 62, no. 8, pp. 3861–3876, 2016.
 - [20] S. P. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.
 - [21] D. Pickem, P. Glotfelter, L. Wang, M. Mote, A. Ames, E. Feron, and M. Egerstedt, “The robotarium: A remotely accessible swarm robotics research testbed,” in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 1699–1706.
 - [22] M. ApS, *The MOSEK optimization toolbox for MATLAB manual. Version 9.0.*, 2019. [Online]. Available: <http://docs.mosek.com/9.0/toolbox/index.html>