Collision Cone Control Barrier Functions for Fixed Wing UAV

Ravi Agrawal*[1], Aryan Agarwal*[2], Manan Tayal[3], Shishir Kolathaya[3]

***Abstract*— Fixed-wing UAVs have transformed the transportation system with their high flight speed and long endurance, yet their safe operation in increasingly cluttered environments depends heavily on effective collision avoidance techniques. This paper presents a novel method for safely navigating an aircraft along a desired route while avoiding moving obstacles. We utilize a class of control barrier functions (CBFs) based on collision cones to ensure the relative velocity between the aircraft and the obstacle consistently avoids a cone of vectors that might lead to a collision. By demonstrating that the proposed constraint is a valid CBF for the aircraft, we can leverage its real-time implementation via Quadratic Programs (QPs), termed the CBF-QPs. Validation includes simulating control law along trajectories, showing effectiveness in both static and moving obstacle scenarios.**

I. Introduction

Fixed-wing UAVs are widely used in long-distance surveying, mapping, and cargo delivery. Ensuring the safe operation of such UAVs is crucial, as even minor collisions can lead to life threats and disasters. Various methods, such as artificial potential fields, reachability analysis, and nonlinear model predictive control (NMPC), are employed to tackle obstacle avoidance in UAVs.

Recently, the Control Barrier Functions (CBFs) [1] based approach has emerged as a promising strategy for ensuring the safe operation of autonomous systems. This model-based control design method offers a computationally efficient solution for handling complex situations while guaranteeing safety. CBFs can be formulated as a Quadratic Problem (QP) and solved online, making them suitable for real-time safety-critical applications. Unlike NMPC, which provides soft constraints on the system's trajectory, CBFs enforce hard safety constraints, offering superior safety guarantees.

For obstacle avoidance in dynamic environments, the collision cone method is widely used [2], [3], [4]. It defines a cone-shaped region between two objects to indicate potential collision areas, which can be avoided by preventing the relative velocity vector from falling within the cone. This approach is simple, efficient, and can be easily integrated into motion planning algorithms making it effective in dynamic environments.

Despite its effectiveness, the collision cone method has been mostly used in offline motion planning, with limited real-time implementations. By using CBF-QP formulations, we can create a new class of CBFs based on collision cones for real-time applications. This idea of **Collision Cone CBFs** was originally proposed in [5]–[7] for ground robots and, then extended to quadrotors [8], [9], we aim to extend this approach to ensure safety in the control the fixed-wing aircrafts which are underactuated and have non-holonomic constraints with higher degrees of freedom (DoF).

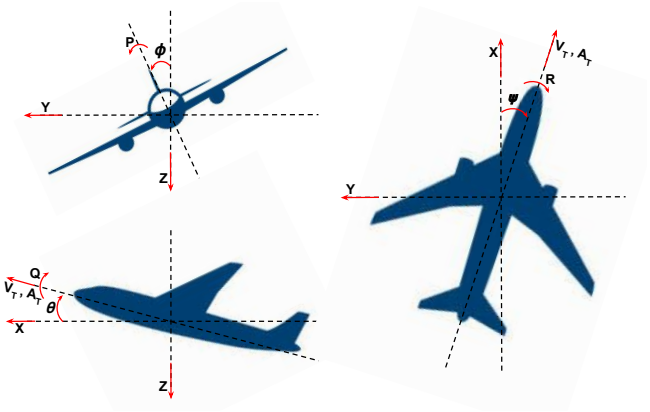


Fig. 1: Aircraft's linear and angular velocities defined in body-fixed frame. (X,Y,Z) represents earth-fixed frame.

The rest of this paper is organized as follows. Preliminaries explaining the fixed-wing UAV model, the concept of control barrier functions (CBFs), collision cone CBFs, and controller design are introduced in section II. The application of the above CBFs on the fixed-wing aircraft to avoid obstacles modelled as spheres is discussed in section III. The simulation setup and comparative results will be discussed in section IV. Finally, we present our concluding remarks in section V.

II. Preliminaries

In this section, first, we will describe the kinematics of fixed-wing UAV. Next, we will formally introduce Control Barrier Functions (CBFs) and their importance in providing formal safety guarantees in safety-critical applications. Finally, we will introduce Collision Cone Control Barrier Function (C3BF) approach.

A. Fixed Wing UAV model

In this work, we adapt the *3D Dubins kinematic model* [10], that describes the motion of fixed-wing aircraft. Kinematic models are more generalised and are invariant with changes in inertia or vechile dimensions, thus readily used for navigation and path planning tasks. The state of an aircraft in Dubins model is described by $x = [x_p, y_p, z_p, \phi, \theta, \psi, V_T]$.

This research is supported by PMRF.

[1]VNIT, Nagpur.
[2]Indian Institute of Technology Kharagpur.
[3]Cyber-Physical Systems, Indian Institute of Science (IISc), Bengaluru.
{manantayal, shishirk}@iisc.ac.in .
* denotes equal first author contribution.

$$\underbrace{\begin{bmatrix}\dot{x}_p\\ \dot{y}_p\\ \dot{z}_p\\ \dot{\phi}\\ \dot{\theta}\\ \dot{\psi}\\ \dot{V}_{\mathrm{T}}\end{bmatrix}}_{\dot{x}} = \underbrace{\begin{bmatrix} V_{\mathrm{T}}c\theta c\psi \\ V_{\mathrm{T}}c\theta s\psi \\ -V_{\mathrm{T}}s\theta \\ \frac{g}{V_{\mathrm{T}}}\begin{bmatrix} s\phi c\phi s\theta \\ -(s\phi)^2 c\theta \\ s\phi c\phi \end{bmatrix} \\ 0 \end{bmatrix}}_{f(x)} + \underbrace{\begin{bmatrix} \begin{bmatrix}0&0&0\\0&0&0\\0&0&0\end{bmatrix} \\ \begin{bmatrix}0&1&s\phi t\theta\\0&0&c\phi\\0&0&\frac{s\phi}{c\theta}\end{bmatrix} \\ \begin{bmatrix}1&0&0\end{bmatrix} \end{bmatrix}}_{g(x)} \underbrace{\begin{bmatrix}A_T\\P\\Q\end{bmatrix}}_{u} \tag{1}$$

x_p, y_p, and z_p denote the coordinates of the aircraft's center of gravity in an earth-fixed frame. ϕ, θ and ψ represents the (roll, pitch & yaw) orientation of the aircraft. (see Fig. 1). P, Q, R denote the respective angular velocities of the aircraft in the body frame, V_T is the total longitudinal speed of the aircraft and A_T is the longitudinal acceleration of the aircraft. g is the gravitational acceleration constant.

B. Control barrier functions (CBFs)

Here, we formally introduce Control Barrier Functions (CBFs) and their applications in the context of safety. Given the aircraft model (1), we have the nonlinear model in the control affine form:

$$\dot{x} = f(x) + g(x)u \tag{2}$$

where $x \in \mathcal{D} \subseteq \mathbb{R}^n$ is the state of system, and $u \in \mathbb{U} \subseteq \mathbb{R}^m$ the input for the system. Assume that the functions $f : \mathbb{R}^n \to \mathbb{R}^n$ and $g : \mathbb{R}^n \to \mathbb{R}^{n\times m}$ are continuously differentiable. Specific formulation of f, g for the aircraft were described in (1). Given a Lipschitz continuous control law $u = k(x)$, the resulting closed loop system $\dot{x} = f_{cl}(x) = f(x)+g(x)k(x)$ yields a solution $x(t)$, with initial condition $x(0) = x_0$. Consider a set $\mathcal{C}$ defined as the *super-level set* of a continuously differentiable function $h : \mathcal{D} \subseteq \mathbb{R}^n \to \mathbb{R}$ yielding,

$$\mathcal{C} = \{x \in \mathcal{D} \subset \mathbb{R}^n : h(x) \geq 0\} \tag{3}$$
$$\partial\mathcal{C} = \{x \in \mathcal{D} \subset \mathbb{R}^n : h(x) = 0\} \tag{4}$$
$$\text{Int}\,(\mathcal{C}) = \{x \in \mathcal{D} \subset \mathbb{R}^n : h(x) > 0\} \tag{5}$$

It is assumed that $\text{Int}\,(\mathcal{C})$ is non-empty and $\mathcal{C}$ has no isolated points, i.e. $\text{Int}\,(\mathcal{C}) \neq \phi$ and $\overline{\text{Int}\,(\mathcal{C})} = \mathcal{C}$. The system is safe w.r.t. the control law $u = k(x)$ if $\forall\, x(0) \in \mathcal{C} \implies x(t) \in \mathcal{C}$ $\forall t \geq 0$. We can mathematically verify if the controller $k(x)$ is safeguarding or not by using Control Barrier Functions (CBFs), which is defined next.

Definition 1 (Control barrier function (CBF)): Given the set $\mathcal{C}$ defined by (3)-(5)*, with $\frac{\partial h}{\partial x}(x) \neq 0$ $\forall x \in \partial\mathcal{C}$, the function h is called the control barrier function (CBF) defined on the set $\mathcal{D}$, if there exists an extended class $\mathcal{K}$ function κ such that for all $x \in \mathcal{D}$:*

$$\underbrace{sup}_{u\in\mathbb{U}} \left[\underbrace{\mathcal{L}_f h(x) + \mathcal{L}_g h(x) u}_{\dot{h}(x,u)} + \kappa\left(h(x)\right) \right] \geq 0 \tag{6}$$

where $\mathcal{L}_f h(x) = \frac{\partial h}{\partial x} f(x)$ and $\mathcal{L}_g h(x) = \frac{\partial h}{\partial x} g(x)$ are the Lie derivatives.

Given this definition of a CBF, we know from [1] and [11] that any Lipschitz continuous control law $k(x)$ satisfying the inequality: $\dot{h} + \kappa(h) \geq 0$ ensures safety of $\mathcal{C}$ if $x(0) \in \mathcal{C}$, and asymptotic convergence to $\mathcal{C}$ if $x(0)$ is outside of $\mathcal{C}$.

C. Safety Filter Design

Having described the CBF, we can now describe the Quadratic Programming (QP) formulation of CBFs. CBFs act as *safety filters* which take the desired input $u_{des}(x,t)$ and modify this input in a minimal way:

$$\begin{aligned} u^*(x,t) = \underset{u\in\mathbb{U}\subseteq\mathbb{R}^m}{\arg\min} \|u - u_{des}(x,t)\|^2 \\ \text{s.t. } \mathcal{L}_f h(x) + \mathcal{L}_g h(x) u + \kappa\left(h(x)\right) \geq 0 \end{aligned} \tag{7}$$

This is called the Control Barrier Function based Quadratic Program (CBF-QP). The explicit form of the CBF-QP control u^* can be obtained by solving the above optimization problem using KKT conditions:

$$u^*(x,t) = u_{des}(x,t) + u_{safe}(x,t) \tag{8}$$

where $u_{safe}(x,t)$ is given by

$$u_{safe}(x,t) \;=\; \begin{cases} 0 & \text{for } \psi(x,t) \geq 0 \\ -\frac{\mathcal{L}_g h(x)^T \psi(x,t)}{\mathcal{L}_g h(x) \mathcal{L}_g h(x)^T} & \text{for } \psi(x,t) < 0 \end{cases} \tag{9}$$

where $\psi(x,t) := \dot{h}\left(x, u_{ref}(x,t)\right) + \kappa\left(h(x)\right)$. The sign change of ψ yields a switching type of a control law.

D. Collision Cone CBF (C3BF) candidate for fixed wing UAVs

We now formally introduce the proposed CBF candidate for fixed wing UAVs. Let us assume that the obstacle is centered at $(x_o(t), y_o(t), z_o(t))$ having maximum dimension of r_{obs}. We assume that $x_o(t), y_o(t), z_o(t)$ are differentiable and their derivatives are piece-wise constants. r_{uav} represents the maximum radius of the sphere that circumscribes the UAV, d_s is the minimum distance to be maintained between the UAV and the obstacle. The proposed approach combines the idea of potential unsafe directions given by collision cone (Fig. 2) as an unsafe set to formulate a CBF as in [5]. Consider the following CBF candidate:

$$h(x,t) = < p_{\text{rel}}, v_{\text{rel}} > + \|p_{\text{rel}}\| \|v_{\text{rel}}\| \cos\alpha, \tag{10}$$

where p_{rel} is the relative position vector between the body center of the aircraft and the center of the obstacle, v_{rel} is the relative velocity, $< \cdot, \cdot >$ is the dot product of 2 vectors and α is the half angle of the cone, the expression of $\cos\alpha$ is given by $\frac{\sqrt{\|p_{\text{rel}}\|^2 - r^2}}{\|p_{\text{rel}}\|}$ (see Fig. 2). Precise mathematical definitions for $p_{\text{rel}}, v_{\text{rel}}$ will be given in the next section. The

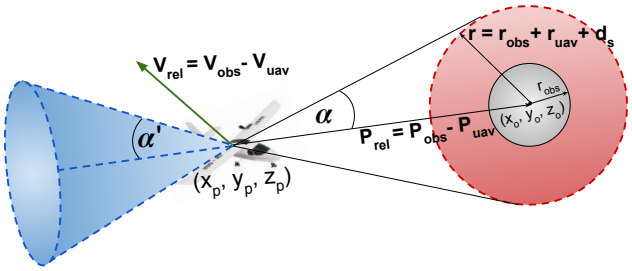


Fig. 2: **3D CBF** candidate: The dimensions of the obstacle are comparable to each other, it can be assumed as a sphere

proposed constraint simply ensures that the angle between $p_{\text{rel}}, v_{\text{rel}}$ is less than $180° - \alpha$.

In [5], it was shown that the proposed candidate (10) is valid CBF for wheeled mobile robots, i.e., the unicycle and bicycle, and for quadrotors. With this result, CBF-QPs were constructed that yielded collision-avoiding behaviors in these models. We aim to extend this to the class of fixed wing UAVs.

III. Collision Cone CBFs on Fixed Wing UAV

Having described the Collision Cone CBF candidate, we will see their application on fixed wing UAVs in this section. We consider our CBF candidate in its naive form and one extended with a backstepping-based approach.

A. Naive C3BF candidate

We first obtain the relative position and velocity vectors between the body centre of the fixed wing aircraft and the obstacle:

$$p_{\text{rel}} := \begin{bmatrix} x_o \\ y_o \\ z_o \end{bmatrix} - \begin{bmatrix} x_p \\ y_p \\ z_p \end{bmatrix} \tag{11}$$

x_o, y_o, z_o represents the obstacle location as a function of time. Also, since the obstacles are of constant velocity, we have $\ddot{x}_o = \ddot{y}_o = \ddot{z}_o = 0$. We obtain its relative velocity as

$$v_{\text{rel}} := \dot{p}_{rel} \tag{12}$$

Now, we calculate the $\dot{v}_{\text{rel}}$ term which contains our inputs, i.e, (A_{T}, P, Q), as follows:

$$\dot{v}_{\text{rel}} := -\begin{bmatrix} c\psi c\theta & 0 & -V_{\text{T}}(s\phi s\psi + s\theta c\phi c\psi) \\ s\psi c\theta & 0 & V_{\text{T}}(s\phi c\psi - s\psi s\theta c\phi) \\ -s\theta & 0 & -V_{\text{T}} c\phi c\theta \end{bmatrix} \begin{bmatrix} A_{\text{T}} \\ P \\ Q \end{bmatrix} + \text{additional terms.} \tag{13}$$

Please note that the additional terms in the above equation, refer to those terms that do not contain the input terms (A_{T}, P, Q) and thus do not contribute to the calculation of $\mathcal{L}_g h$.

Having introduced the Collision Cone CBF candidate in II-D, we state the following result about its validity:

Theorem 1: Given the fixed wing UAV model (1)*, the proposed CBF candidate* (10) *with* $p_{\text{rel}}, v_{\text{rel}}$ *defined by* (11)*,* (12) *is a valid CBF defined for the set* $\mathcal{C}$*.*

The way in which the controlled system works is shown in Fig. 3 and described in detail in (IV). The main shortcoming of this CBF is that it is valid only for the set $\mathcal{C}$. Intuitively, this is due to the fact that u_{safe} can only alter A_{T} and Q, but not P. This can be verified by seeing how $\mathcal{L}_g h$ is constructed using $\dot{v}_{\text{rel}}$ in (13):

$$\mathcal{L}_g h = \begin{bmatrix} < \xi(x,t), \begin{bmatrix} c\psi c\theta \\ s\psi c\theta \\ -s\theta \end{bmatrix} > \\ < \xi(x,t), \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} > \\ < \xi(x,t), \begin{bmatrix} -V_{\text{T}}(s\phi s\psi + s\theta c\phi c\psi) \\ V_{\text{T}}(s\phi c\psi - s\psi s\theta c\phi) \\ -V_{\text{T}} c\phi c\theta \end{bmatrix} > \end{bmatrix}^T, \tag{14}$$

where $\xi(x,t) = p_{\text{rel}} + v_{\text{rel}} \sqrt{\|p_{\text{rel}}\|^2 - r^2} / \|v_{\text{rel}}\|$

This shows that the coefficient of P in $\mathcal{L}_g h(x) u$ is 0, rendering input P uncontrollable by the safety filter.

B. C3BF Candidate with Backstepping

The safety set can be extended from $\mathcal{C}$ to $\mathcal{D}$ by considering a version of (10) extended by backstepping. Using the defined kinematics (1), we can determine the desired roll rate R_{des} from the desired acceleration a_{des} based on the given trajectory. Since R is a determined quantity, we *backstep* from R to find P, based on the method described in [12]. The new CBF candidate is as follows:

$$h(x,t) = < p_{\text{rel}}, v_{\text{rel}} > + \|p_{\text{rel}}\| \|v_{\text{rel}}\| \cos\alpha - \frac{1}{2\lambda}(R_{des} - R)^2, \tag{15}$$

where $\lambda > 0$ is a scaling constant.

We have the following result for the above CBF candidate extended through backstepping:

Theorem 2: Given the fixed wing UAV model (1)*, the proposed CBF candidate* (15) *with* $p_{\text{rel}}, v_{\text{rel}}$ *defined by* (11)*,* (12) *is a valid CBF defined for the set* $\mathcal{D}$*.*

The proof of the above result is beyond the scope of this paper and will be taken up in a future work. A descriptive comparison of (10) and (15) is given in the following section.

IV. Results and Discussions

In this section we describe the setup and parameters used for the Python simulation. Next, we do a comparative analysis of naive and backstepping based C3BFs, understanding how they affect the controlled system. Finally, we compare teh results from both our proposed C3BF candidates to an existing CBF

A. Simulation Setup

We have validated the C3BF-QP based controller on fixed wing UAVs for Naive and Backstepped CBF cases. We have used an exponentially stable velocity tracking controller (Appendix B [10]). Note that the choice of reference controller does not affect the validity of the CBFs proposed above, any valid velocity tracking controller can be used.

For the class $\mathcal{K}$ function in the CBF inequality, we chose $\kappa(h) = \gamma h$, where $\gamma = 1$. The scaling parameter in (15) was chosen to be $\lambda = 10^{-4}$. The gravitational constant used in (1) is $g = 9.81m/s^2$. The collision radius was chosen to be $r = r_{obs} + r_{uav} + d_s = 100m$, where r_{obs} is the collision radius of the obstacle, r_{uav} is the collision radius of the controlled aircraft and d_s is a safety measure.

B. C3BF - Naive vs Backstepped

The way in which the controlled system works is layered with a base trajectory tracking controller with a C3BF-QP safety layer for collision avoidance. The base controller u_{des} is designed to make the aircraft follow a given trajectory. When an obstacle, static or having constant velocity, intersects the aircraft's trajectory with impending collision, the C3BF-QP safety filter starts to alter the u_{des} in order to obtain a net control policy which ensures that the aircraft remains 'safe', i.e. avoids the collision.

In the case of the naive C3BF (10), u_{safe} derived from (7) alters the *desired* A_{T} and Q, provided by u_{des} such that the new control policy $u_{des} + u_{safe}$ avoids the obstacle on the path. The aircraft only pitches up or down in order to avoid the obstacle while slowing down or speeding up accordingly.

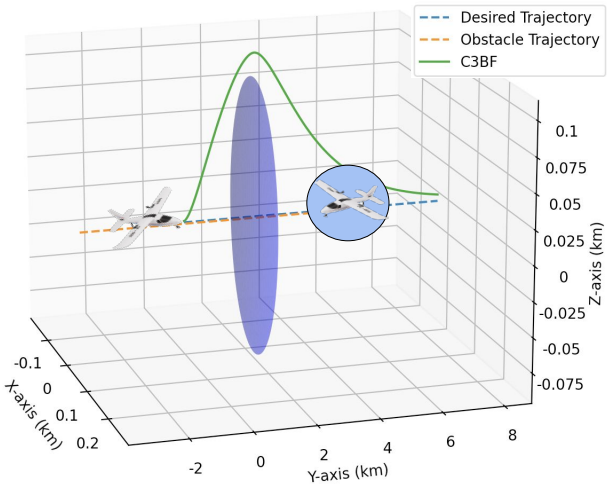
In case of obstacles which cannot be flown over, e.g. pillars, geofences, etc., u_{safe} of (10) is unable to maneuver the aircraft sideways, and can only make the aircraft stop in order to ensure safety. Hence the C3BF described by (15) can generate u_{safe} from (9) which brings change in all of A_{T}, P and Q, making the new CBF valid for the entire set $\mathcal{D}$. This also ensures stability of set $\mathcal{C}$ with respect to set $\mathcal{D}$. However, for the case of an obstacle with a finite 3D collision radius, following a constant velocity trajectory or static in the path of the aircraft, (10) and (15) produce u_{safe} which behave identically, since the obstacle can always be flown over.

C. C3BF vs existing CBF

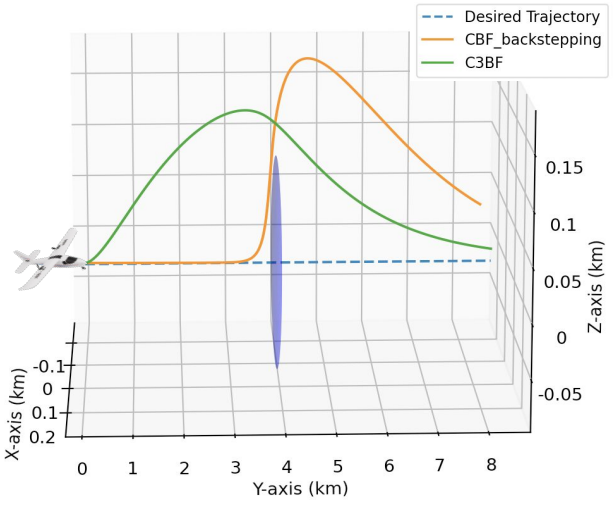
Comparing the action of the CBF proposed in [10] to the C3BFs proposed in (10) and (15), (Fig. 3 b, c) we see that the C3BFs safety filter start acting on the system much earlier allowing for a smoother path. Also the action of the CBF-QP in [10] is more conservative, taking a path with a large deviation from the trajectory, whereas the action of C3BF-QPs from (10) and (15) chooses a path with minimal deviation while ensuring safety. This also shows that C3BFs create control policies which require a lower effort.

V. Conclusions

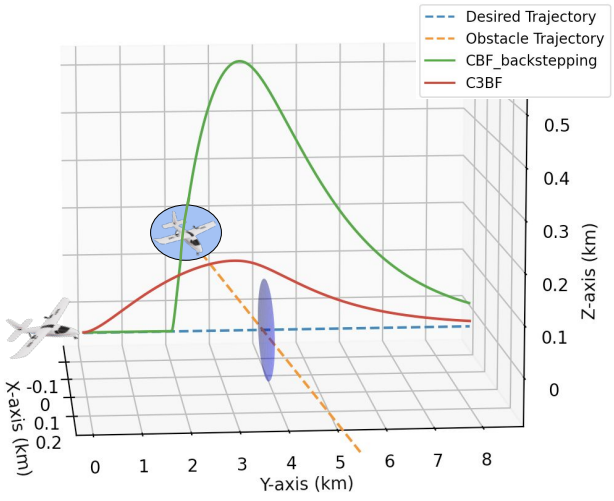
We have shown that Collision Cone CBF works for fixed wing UAVs. However, the point to be noted is that this work is under progress and we are yet to compare the results of Naive C3BF with C3BF with Backstepping for cases which set them apart in simulations. Moreover, we also plan to extend this analysis to collision avoidance with long obstacles like trees and towers, while providing theoretical guarantees.



(a)



(b)



(c)

Fig. 3: Aircraft's behaviour following C3BF: Obstacle incoming head-on to the aircraft (a), Comparision of aircraft's behaviour following C3BF and CBF in [10] for a static obstacle (b) and for constant velocity obstacle (c). In all three cases, the collision radius $r = 100m$.

REFERENCES

[1] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs for safety critical systems," *IEEE Transactions on Automatic Control*, vol. 62, no. 8, pp. 3861–3876, aug 2017. [Online]. Available: https://doi.org/10.1109%2Ftac.2016.2638961

[2] P. Fiorini and Z. Shiller, "Motion planning in dynamic environments using the relative velocity paradigm," in *IEEE International Conference on Robotics and Automation*, 1993, pp. 560–565 vol.1.

[3] ——, "Motion planning in dynamic environments using velocity obstacles," *The International Journal of Robotics Research*, vol. 17, no. 7, pp. 760–772, 1998. [Online]. Available: https://doi.org/10.1177/027836499801700706

[4] A. Chakravarthy and D. Ghose, "Obstacle avoidance in a dynamic environment: a collision cone approach," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 28, no. 5, pp. 562–574, 1998.

[5] P. Thontepu, B. G. Goswami, M. Tayal, N. Singh, S. S. P I, S. S. M G, S. Sundaram, V. Katewa, and S. Kolathaya, "Collision cone control barrier functions for kinematic obstacle avoidance in ugvs," in *2023 Ninth Indian Control Conference (ICC)*, 2023, pp. 293–298.

[6] B. G. Goswami, M. Tayal, K. Rajgopal, P. Jagtap, and S. Kolathaya, "Collision cone control barrier functions: Experimental validation on ugvs for kinematic obstacle avoidance," *arXiv preprint arXiv:2310.10839*, 2023.

[7] M. Tayal and S. Kolathaya, "Safe legged locomotion using collision cone control barrier functions (c3bfs)," *arXiv preprint arXiv:2309.01898*, 2023.

[8] M. Tayal, R. Singh, J. Keshavan, and S. Kolathaya, "Control barrier functions in dynamic uavs for kinematic obstacle avoidance: a collision cone approach," *arXiv preprint arXiv:2303.15871*, 2023.

[9] M. Tayal, B. Giri Goswami, K. Rajgopal, R. Singh, T. Rao, J. Keshavan, P. Jagtap, and S. Kolathaya, "A Collision Cone Approach for Control Barrier Functions," *arXiv e-prints*, p. arXiv:2403.07043, 2024.

[10] T. G. Molnar, S. K. Kannan, J. Cunningham, K. Dunlap, K. L. Hobbs, and A. D. Ames, "Collision avoidance and geofencing for fixed-wing aircraft with control barrier functions," 2024.

[11] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, "Control barrier functions: Theory and applications," in *2019 18th European Control Conference (ECC)*, 2019.

[12] A. J. Taylor, P. Ong, T. G. Molnar, and A. D. Ames, "Safe backstepping with control barrier functions," 2022.