Proceedings of the 2025 SICE Festival with Annual Conference
September 9-12, 2025, Chiang Mai, Thailand

Adaptive Cruise Control of Quadrotor UAVs Considering a Following Vehicle Based on Distributed Quadratic Programming

Ryota Aoyanagi[1†] and Tatsuya Ibuki[1]

[1]Department of Electronics and Bioinformatics, Meiji University, Kanagawa, Japan
(Tel: +81-44-934-7344; E-mail: ibuki@meiji.ac.jp)

Abstract: This paper proposes an adaptive cruise control (ACC) method for quadrotor UAVs that considers both the preceding and following quadrotors. The control method is an optimal control approach. It utilizes a quadratic programming (QP) with a collision avoidance constraint formulated using a parameter-adaptive control barrier function. By considering both the preceding and following quadrotors, cooperative distance control between quadrotors becomes possible, and the feasibility of solving the QP is expected to be improved. In addition, by solving the optimization problem locally in each quadrotor, distributed control is achieved. The present method is applied to ACC in linear quadrotor platooning, and its effectiveness is demonstrated through a simulation.

Keywords: Adaptive cruise control, Control barrier function, Quadrotor UAV, Quadratic programming

1. INTRODUCTION

Globally, more than one million people lose their lives in traffic accidents each year, making it a serious issue. Additionally, a growing shortage of taxi, bus, and truck drivers is also a concern. As a solution to these issues, research and development of autonomous driving and advanced driver-assistance systems (ADAS) are being actively pursued. One of the widely adopted ADAS technologies is adaptive cruise control (ACC). It is a system that maintains a safe distance from the preceding vehicle by controlling the speed. This capability not only improves driver comfort and safety but also alleviates traffic congestion and improves fuel efficiency. Currently, research on ACC primarily focuses on cars, but we anticipate that ACC for quadrotors will become commonplace in the future. Therefore, this study focuses on control methods for quadrotor ACC.

Control methods for ACC based on model predictive control (MPC) have been extensively studied [1, 2]. However, MPC is susceptible to model errors and disturbances, and its computational burden increases significantly with the number of constraints [3]. Therefore, a method utilizing control barrier functions (CBFs) [4] has been proposed to mitigate computational complexity. CBFs are used to transform system state constraints into constraints on the control input and are often applied to collision avoidance problems in multi-agent systems [5]. An ACC method has been proposed that combines CBFs with control Lyapunov functions (CLFs) [6] and solves a quadratic programming (QP) problem in real-time with these constraints [7]. The CBF in [7] is applied to constraints with relative degree 1 with respect to the system dynamics. The CBF for constraints with relative degree 2 has also been proposed [8]. Furthermore, an exponential CBF [9] and a high-order CBF (HOCBF) [10] have been proposed to handle safety constraints with arbitrarily high relative degrees. Additionally, an adaptive CBF has been proposed to guarantee system safety under parameter uncertainties [11]. However, these methods may result in infeasible QPs when control bounds are time-varying and the system dynamics are noisy. To address this issue, a parameter-adaptive CBF (PACBF) has been proposed [12]. The PACBF extends the HOCBF framework by incorporating penalty functions that dynamically adjust the barrier conditions, thereby improving feasibility under time-varying control bounds and uncertain dynamics.

In this paper, we propose a distributed control method that enables cooperative inter-quadrotor distance maintenance by considering both the preceding and following quadrotors through collision avoidance constraints based on the PACBF. Each quadrotor's controller solves a QP with these constraints. Furthermore, through simulations in the context of ACC applied to quadrotor swarms in linear formation flight, we demonstrate the benefits of distance control that considers both the preceding and following quadrotors. The reason for using the PACBF is to handle constraints with relative degree 2 with respect to the quadrotor dynamics and to prevent conflicts with input constraints, thereby improving the solvability of the QP. In this method, the positions, velocities, and accelerations of the preceding and following quadrotors are used. These values are assumed to be obtained solely from the sensors on each quadrotor, eliminating the need for communication between quadrotors and enabling a fully distributed control approach.

Our main contributions are summarized as follows: (i) we consider ACC not for commonly used car models, but for quadrotors, and propose a new safety criterion that includes a safety distance based on the relative velocity and a constraint on excessive tilting of the quadrotor; (ii) we present a new constraint that takes into account not only the preceding quadrotor but also the following one, unlike the traditional constraint that considers only the preceding one; and (iii) we demonstrate the effectiveness of the proposed method through numerical simulations, particularly the behavior resulting from considering the following quadrotor.

[†] Ryota Aoyanagi is the presenter of this paper.

2. PLOBLEM SETTINGS

In this section, we introduce the dynamics of quadrotors and set control objectives.

2.1. Quadrotor Dynamics and x-axis Position Control

In this paper, we consider the platooning of n quadrotors along the x-axis. Therefore, we set the positions in both the y-direction and z-direction to zero for simplicity, focusing solely on the motion along the x-axis. ACC in 3D space, including the y-axis and z-axis directions, is left for future work.

As derived from [13], the translational dynamics of quadrotor i, $i \in \{1, \dots, n\}$ along the x-axis can be described as

$$\ddot{x}_i = g_{\mathrm{grav}} \tan \theta_i, \tag{1}$$

where x_i is the position along the x-axis expressed in the world frame Σ_{w}, g_{grav} is gravity, and θ_i is the pitch angle around the y-axis, also expressed in Σ_{w} as shown in Fig. 1. In this figure, ω_{θ_i} denotes the angular velocity in the pitch direction and F is the thrust generated by the quadrotor, directed upward in the body-fixed frame Σ_{b}. The thrust F is used to compensate for gravity and to control the position along the z-axis. Differentiating both sides of Eq. (1) with respect to time yields:

$$\dddot{x}_i = \frac{g_{\mathrm{grav}}}{\cos^2 \theta_i} \omega_{\theta_i},$$

which includes the control input $\omega_{\theta_i} (= \dot{\theta}_i)$, the angular velocity. For simplicity, in this paper, we consider $\frac{g_{\mathrm{grav}}}{\cos^2 \theta_i} \omega_{\theta_i}$ as the control input and define it as u_i. That is, the control input of the quadrotor can be interpreted as the jerk in the x-axis direction. In summary, the dynamics of each quadrotor are given by the triple integrator:

$$\dddot{x}_i = u_i.$$

2.2. Control Objective

In this paper, the dynamics of quadrotor i are represented as follows:

$$\underbrace{\begin{bmatrix} \dot{x}_i(t) \\ \dot{v}_i(t) \\ \dot{a}_i(t) \end{bmatrix}}_{\dot{\boldsymbol{q}}_i(t)} = \underbrace{\begin{bmatrix} v_i(t) \\ a_i(t) \\ 0 \end{bmatrix}}_{f(\boldsymbol{q}_i(t))} + \underbrace{\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}}_{g(\boldsymbol{q}_i(t))} u_i(t). \tag{2}$$

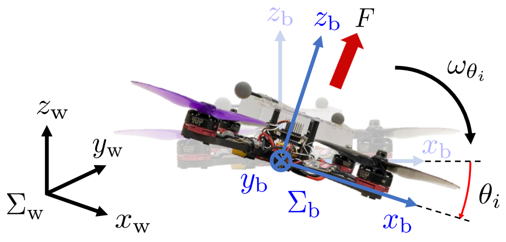


Fig. 1 Pitch angle and angular velocity of quadrotor.

Here, $x_i(t)$ is the position, $v_i(t)$ is the velocity, and $a_i(t)$ is the acceleration of quadrotor i, and the control input $u_i(t)$ is the jerk, all expressed in the world frame.

The goal of this work is to propose a distributed distance control method that considers the following quadrotor as well as the preceding one. In this paper, we consider distance maintenance and platooning behavior of quadrotors flying along a straight line. To achieve this, a safe distance $d > 0$ from the preceding and following quadrotors must be maintained, and the tilt of the quadrotor must be limited to avoid instability due to excessive tilting. Under these conditions, we aim to fly as close as possible to the desired speed $v_{\mathrm{d},i} > 0$ and to minimize the jerk during flight. We first define "dangerous/safe inter-quadrotor distance".

Definition 1: Quadrotors $i-1$ and i for $i \in \{2, \dots, n\}$ are said to be at a dangerous inter-quadrotor distance if

$$x_{i-1} - x_i < T(v_i - v_{i-1}) + d,$$

with $T > 0$. Then, the quadrotors are said to maintain a safe inter-quadrotor distance (see Fig. 2) if

$$x_{i-1}(t) - x_i(t) \geq T(v_i(t) - v_{i-1}(t)) + d \quad \forall t \geq 0. \tag{3}$$

In conventional methods based on automobile models, the safe inter-quadrotor distance is typically either constant or proportional to the speed of the own vehicle. However, the safe distance actually depends on the relative velocity. For example, the required safe distance differs significantly when the preceding vehicle is traveling much faster than the own vehicle compared to when it is traveling much slower. Therefore, we express the safe inter-quadrotor distance using the relative velocity, as shown in Eq. (3). This is based on the concept of time-to-collision [14], where the T represents the remaining time until a collision occurs, assuming that the relative velocity between the two vehicles remains constant. A disadvantage of using the relative velocity is that if the preceding vehicle frequently changes speed, the own vehicle will also repeatedly accelerate and decelerate in response. In the case of automobiles, this leads to discomfort for the passengers. But when quadrotors do not carry passengers, this does not become an issue.

Moreover, we define "unstable/stable attitudes of quadrotors".

Definition 2: Quadrotor i is said to be in an unstable attitude if

$$\theta_{\mathrm{c}} < |\theta_i| \, .$$

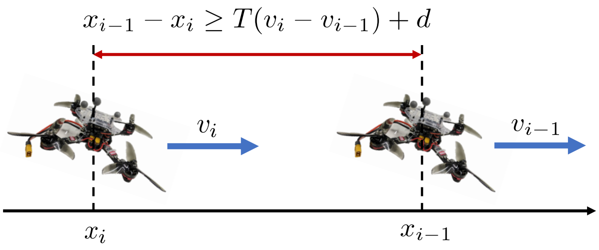


Fig. 2 Safe inter-quadrotor distance.

Here, $\theta_c > 0$ is the maximum allowable pitch angle. On the contrary, the quadrotors are said to maintain a stable pitch angle if

$$|\theta_i(t)| \leq \theta_{\mathrm{c}} \quad \forall t \geq 0. \tag{4}$$

In summary, the objectives and constraints of each quadrotor are as follows:

Objective 1: To maintain the desired speed $v_{\mathrm{d},i}$;
Objective 2: To minimize the jerk u_i;
Constraint 1: To maintain the safe distance (3);
Constraint 2: To maintain the safe pitch angle (4).

3. CONTROL METHOD

We propose a control method based on the HOCBF, PACBF, and CLF, whose details are presented in the appendix. Definitions 3–5 and Facts 1 and 2, which are used in this section, are also provided in the appendix.

3.1. Design of Collision Avoidance Constraints

In this section, we formulate collision avoidance constraints considering both the preceding and following quadrotors using the PACBF, while taking into account the quadrotor dynamics (2).

Based on Eq. (3), $h_i(\boldsymbol{q}_{i-1}(t), \boldsymbol{q}_i(t))$, $i \in \{2, \ldots, n\}$ is defined as follows:

$$\begin{aligned} h_i(\boldsymbol{q}_{i-1}(t), \boldsymbol{q}_i(t)) := x_{i-1}(t) - x_i(t) \\ - T(v_i(t) - v_{i-1}(t)) - d. \end{aligned} \tag{5}$$

We also define the safe set

$$\begin{aligned} C_i := \left\{ \{\boldsymbol{q}_{i-1}, \boldsymbol{q}_i\} \in \mathbb{R}^3 \times \mathbb{R}^3 : h_i(\boldsymbol{q}_{i-1}, \boldsymbol{q}_i) \geq 0 \right\}, \\ i \in \{2, \ldots, n\}. \end{aligned} \tag{6}$$

Then, we define time-varying penalty functions $p_{\mathrm{p},i}(t) \geq 0$, $i \in \{1, \ldots, n-1\}$ and $p_{\mathrm{f},i}(t) \geq 0$, $i \in \{2, \ldots, n\}$, and let $\boldsymbol{p}_i(t) = [p_{\mathrm{p},i-1}(t) \ p_{\mathrm{f},i}(t)]^{\mathrm{T}}$, $i \in \{2, \ldots, n\}$. Then we use a PACBF with $m = 2$, the relative degree of Eq. (5). We define $\psi_i(\cdot)$, $i \in \{2, \ldots, n\}$ as

$$\begin{aligned} \psi_i(\boldsymbol{q}_{i-1}(t), \boldsymbol{q}_i(t), \boldsymbol{p}_i(t)) := \dot{h}_i(\boldsymbol{q}_{i-1}(t), \boldsymbol{q}_i(t)) \\ + (p_{\mathrm{p},i-1}(t) + p_{\mathrm{f},i}(t)) h_i(\boldsymbol{q}_{i-1}(t), \boldsymbol{q}_i(t)). \end{aligned}$$

We also define the auxiliary dynamics for $p_{\mathrm{p},i}(t)$, $p_{\mathrm{f},i}(t)$ according to Eq. (A.7), by adopting the simple form as follows:

$$\dot{p}_{\mathrm{p},i}(t) = \nu_{\mathrm{p},i}(t), \quad \dot{p}_{\mathrm{f},i}(t) = \nu_{\mathrm{f},i}(t). \tag{7}$$

Based on Definition 4 and Fact 2, we obtain a condition that must be satisfied to ensure collision avoidance, as follows:

$$\begin{aligned} \dot{\psi}_i(\boldsymbol{q}_{i-1}(t), \boldsymbol{q}_i(t), \boldsymbol{p}_i(t)) \\ + \alpha \psi_i(\boldsymbol{q}_{i-1}(t), \boldsymbol{q}_i(t), \boldsymbol{p}_i(t)) \geq 0, \end{aligned} \tag{8}$$

where $\alpha > 0$ is constant.

Furthermore, the condition is distributed to the adjacent quadrotors $i-1$ and i, to obtain an individual condition for each quadrotor. Based on the dynamics (2) and (7), Eq. (8) is expressed as follows:

$$\begin{aligned} & Tu_{i-1}(t) + \nu_{\mathrm{p},i-1}(t) h_i(\boldsymbol{q}_{i-1}(t), \boldsymbol{q}_i(t)) + \alpha \dot{x}_{i-1}(t) \\ & - Tu_i(t) + \nu_{\mathrm{f},i}(t) h_i(\boldsymbol{q}_{i-1}(t), \boldsymbol{q}_i(t)) - \alpha \dot{x}_i(t) \\ & + p_{\mathrm{p},i-1}(t)(\alpha h_i(\boldsymbol{q}_{i-1}(t), \boldsymbol{q}_i(t)) + \dot{h}_i(\boldsymbol{q}_{i-1}(t), \boldsymbol{q}_i(t))) \\ & + p_{\mathrm{f},i}(t)(\alpha h_i(\boldsymbol{q}_{i-1}(t), \boldsymbol{q}_i(t)) + \dot{h}_i(\boldsymbol{q}_{i-1}(t), \boldsymbol{q}_i(t))) \\ & + (1 + \alpha T)\ddot{x}_{i-1}(t) - (1 + \alpha T)\ddot{x}_i(t) \geq 0. \end{aligned} \tag{9}$$

According to the approach in [5], Eq. (9) is distributed to quadrotors $i-1$ and i as follows:

$$\begin{aligned} & Tu_{i-1}(t) + \nu_{\mathrm{p},i-1}(t) h_i(\boldsymbol{q}_{i-1}(t), \boldsymbol{q}_i(t)) + \alpha \dot{x}_{i-1}(t) \\ & + (1 + \alpha T)\ddot{x}_{i-1}(t) + p_{\mathrm{p},i-1}(t)(\alpha h_i(\boldsymbol{q}_{i-1}(t), \boldsymbol{q}_i(t)) \\ & + \dot{h}_i(\boldsymbol{q}_{i-1}(t), \boldsymbol{q}_i(t))) \geq 0, \end{aligned} \tag{10}$$

$$\begin{aligned} & - Tu_i(t) + \nu_{\mathrm{f},i}(t) h_i(\boldsymbol{q}_{i-1}(t), \boldsymbol{q}_i(t)) - \alpha \dot{x}_i(t) \\ & - (1 + \alpha T)\ddot{x}_i(t) + p_{\mathrm{f},i}(t)(\alpha h_i(\boldsymbol{q}_{i-1}(t), \boldsymbol{q}_i(t)) \\ & + \dot{h}_i(\boldsymbol{q}_{i-1}(t), \boldsymbol{q}_i(t))) \geq 0. \end{aligned} \tag{11}$$

In this way, the condition (8) can be distributed to each quadrotor, enabling decentralized inter-quadrotor distance control. If the relative position, velocity, and acceleration can be measured by the quadrotor itself, then inter-quadrotor communication is not required.

In summary, the conditions that quadrotor i must satisfy to avoid collisions with the preceding and following quadrotors are given by

$$\begin{aligned} & - Tu_i(t) + \nu_{\mathrm{f},i}(t) h_i(\boldsymbol{q}_{i-1}(t), \boldsymbol{q}_i(t)) - (1 + \alpha T)\ddot{x}_i(t) \\ & - \alpha \dot{x}_i(t) + p_{\mathrm{f},i}(t)(\alpha h_i(\boldsymbol{q}_{i-1}(t), \boldsymbol{q}_i(t)) \\ & + \dot{h}_i(\boldsymbol{q}_{i-1}(t), \boldsymbol{q}_i(t))) \geq 0, \end{aligned} \tag{12}$$

$$\begin{aligned} & Tu_i(t) + \nu_{\mathrm{p},i}(t) h_{i+1}(\boldsymbol{q}_i(t), \boldsymbol{q}_{i+1}(t)) + (1 + \alpha T)\ddot{x}_i(t) \\ & + \alpha \dot{x}_i(t) + p_{\mathrm{p},i}(t)(\alpha h_{i+1}(\boldsymbol{q}_i(t), \boldsymbol{q}_{i+1}(t)) \\ & + \dot{h}_{i+1}(\boldsymbol{q}_i(t)), \boldsymbol{q}_{i+1}(t)) \geq 0. \end{aligned} \tag{13}$$

Eq. (12) serves as the constraint for collision avoidance with quadrotor $i-1$, while Eq. (13) serves as the constraint for collision avoidance with quadrotor $i+1$. Note that the first quadrotor ($i = 1$) and the last quadrotor ($i = n$) are subject to only one of these constraints, as they have no preceding or following quadrotor, respectively.

Then, we have the following result:

Theorem 1: Suppose the initial states satisfy $\{\boldsymbol{q}_{i-1}(0), \boldsymbol{q}_i(0)\} \in C_i$, $i \in \{2, \ldots, n\}$. Then, any Lipschitz continuous controllers $u_{i-1}(t)$ and $u_i(t)$ that satisfy Eqs. (10) and (11), respectively, will render the set C_i forward invariant for $i \in \{2, \ldots, n\}$.

Proof: Based on Fact 2, if the condition (8) is satisfied, $\psi_i(\boldsymbol{q}_{i-1}(t), \boldsymbol{q}_i(t), \boldsymbol{p}_i(t)) \geq 0 \ \forall t > 0$ is guaranteed. Since the relative degree of $h_i(\boldsymbol{q}_{i-1}(t), \boldsymbol{q}_i(t))$ with respect to the dynamics (2) is 2, $\dot{h}_i(\boldsymbol{q}_{i-1}(t), \boldsymbol{q}_i(t))$ does not include $u_{i-1}(t)$ nor $u_i(t)$. Therefore, as shown in Eq. (9), the condition for collision avoidance can be represented as a sum of terms involving the system inputs $u_{i-1}(t)$, $\nu_{\mathrm{p},i-1}(t)$, $u_i(t)$, and $\nu_{\mathrm{f},i}(t)$, without any product terms between the inputs. This structure allows the condition (9) to be divided between quadrotors $i-1$ and i as Eqs. (10) and (11). Moreover, the satisfaction of Eq. (8) for all $t \geq 0$ guarantees $\psi_i(\boldsymbol{q}_{i-1}(t), \boldsymbol{q}_i(t), \boldsymbol{p}_i(t)) \geq 0$, i.e. $\dot{h}_i(\boldsymbol{q}_{i-1}(t), \boldsymbol{q}_i(t)) + (p_{\mathrm{p},i-1}(t) + p_{\mathrm{f},i}(t))h_i(\boldsymbol{q}_{i-1}(t), \boldsymbol{q}_i(t)) \geq 0 \; \forall t \geq 0$. This means $h_i(\boldsymbol{q}_{i-1}(t), \boldsymbol{q}_i(t)) \geq 0 \;\; \forall t \geq 0$. The same can be applied to $h_{i+1}(\boldsymbol{q}_i(t), \boldsymbol{q}_{i+1}(t))$. Thus the safe inter-quadrotor distances between the preceding and following quadrotors are maintained. ■

3.2. Design of Other Constraints

Along with *Constraint 1*, we formulate the constraints to achieve *Objective 1*, *Constraint 2*, and those related to the penalty function required for the use of the PACBF.

Velocity Stabilization: To achieve *Objective 1*, the CLF is used to stabilize the velocity. We define a CLF $V(\boldsymbol{q}_i(t)) := (1/2)(a_i(t) - a_{\mathrm{d},i}(t))^2$, where $a_{\mathrm{d},i}(t) = k_{\mathrm{p}}(v_{\mathrm{d},i} - v_i(t)) - k_{\mathrm{d}} a_i(t)$, with $k_{\mathrm{p}} > 0$ and $k_{\mathrm{d}} > 0$. From Definition 5, the velocity will tend to be stabilized at the desired value $v_{\mathrm{d},i}$ if any control input $u_i(t)$ satisfies

$$\mathcal{L}_f V(\boldsymbol{q}_i(t)) + \mathcal{L}_g V(\boldsymbol{q}_i(t)) u_i(t) + \epsilon V(\boldsymbol{q}_i(t)) \leq \delta_{\mathrm{acc},i}(t) \quad \forall t \geq 0. \tag{14}$$

Here, $\delta_{\mathrm{acc},i}(t) \geq 0$ is a relaxation factor to ensure consistency with other constraints and $\epsilon > 0$.

Pitch Angle Limitation: To satisfy *Constraint 2*, an HOCBF of relative degree 2 is used. From Eq. (1), to satisfy Eq. (4), the acceleration $a_i(t)$ must satisfy

$$-g \tan\theta_{\mathrm{c}} \leq a_i(t) \leq g \tan\theta_{\mathrm{c}} \;\; \forall t \geq 0.$$

Let $h_{\theta 1}(\boldsymbol{q}_i(t)) := g\tan\theta_{\mathrm{c}} - a_i(t)$, $h_{\theta 2}(\boldsymbol{q}_i(t)) := a_i(t) + g\tan\theta_{\mathrm{c}}$, and choose $\alpha_1(h_{\theta 1}(\boldsymbol{q}_i(t))) = \alpha_1 h_{\theta 1}(\boldsymbol{q}_i(t))$, $\alpha_1(h_{\theta 2}(\boldsymbol{q}_i(t))) = \alpha_1 h_{\theta 2}(\boldsymbol{q}_i(t))$ in Definition 3, where $\alpha_1 > 0$. Then, the pitch angle is kept within the restricted range if any control input $u_i(t)$ satisfies

$$\mathcal{L}_f h_{\theta 1}(\boldsymbol{q}_i(t)) + \mathcal{L}_g h_{\theta 1}(\boldsymbol{q}_i(t)) u_i(t) + \alpha_1 h_{\theta 1}(\boldsymbol{q}_i(t)) \geq 0, \tag{15}$$

$$\mathcal{L}_f h_{\theta 2}(\boldsymbol{q}_i(t)) + \mathcal{L}_g h_{\theta 2}(\boldsymbol{q}_i(t)) u_i(t) + \alpha_1 h_{\theta 2}(\boldsymbol{q}_i(t)) \geq 0. \tag{16}$$

Penalty Function Limitation: To guarantee $p_{\mathrm{f},i}(t) \geq 0$ and $p_{\mathrm{p},i}(t) \geq 0$ in Eqs. (12) and (13), we regard $p_{\mathrm{f},i}(t)$ and $p_{\mathrm{p},i}(t)$ as CBFs. We choose class $\mathcal{K}$ functions as $\alpha_{\mathrm{p}}(p_{\mathrm{f},i}(t)) = \alpha_{\mathrm{p}} p_{\mathrm{f},i}(t)$ and $\alpha_{\mathrm{p}}(p_{\mathrm{p},i}(t)) = \alpha_{\mathrm{p}} p_{\mathrm{p},i}(t)$, where $\alpha_{\mathrm{p}} > 0$. Given the dynamics of $p_{\mathrm{f},i}(t)$ and $p_{\mathrm{p},i}(t)$ in Eq. (7), the control inputs $\nu_{\mathrm{f},i}(t)$ and $\nu_{\mathrm{p},i}(t)$ must satisfy

$$\nu_{\mathrm{f},i}(t) + \alpha_{\mathrm{p}} p_{\mathrm{f},i}(t) \geq 0, \tag{17}$$

$$\nu_{\mathrm{p},i}(t) + \alpha_{\mathrm{p}} p_{\mathrm{p},i}(t) \geq 0, \tag{18}$$

with $p_{\mathrm{f},i}(0) > 0$ and $p_{\mathrm{p},i}(0) > 0$.

Penalty Function Stabilization: To stabilize $p_{\mathrm{f},i}(t)$ and $p_{\mathrm{p},i}(t)$ to their desired values p_{fd} and p_{pd}, CLFs are used. We define $V_{\mathrm{f}}(p_{\mathrm{f},i}(t)) := (p_{\mathrm{f},i}(t) - p_{\mathrm{fd}})^2$ and $V_{\mathrm{p}}(p_{\mathrm{p},i}(t)) := (p_{\mathrm{p},i}(t) - p_{\mathrm{pd}})^2$. $p_{\mathrm{f},i}(t)$ and $p_{\mathrm{p},i}(t)$ will tend to be stabilized at the desired values if any control inputs $\nu_{\mathrm{f},i}(t)$ and $\nu_{\mathrm{p},i}(t)$ satisfy

$$2(p_{\mathrm{f},i}(t) - p_{\mathrm{fd}})\nu_{\mathrm{f},i}(t) + \epsilon V_{\mathrm{f}}(p_{\mathrm{f},i}(t)) \leq \delta_{\mathrm{f},i}(t), \tag{19}$$

$$2(p_{\mathrm{p},i}(t) - p_{\mathrm{pd}})\nu_{\mathrm{p},i}(t) + \epsilon V_{\mathrm{p}}(p_{\mathrm{p},i}(t)) \leq \delta_{\mathrm{p},i}(t). \tag{20}$$

Here, $\delta_{\mathrm{f},i}(t) \geq 0$ and $\delta_{\mathrm{p},i}(t) \geq 0$ are relaxation factors to ensure consistency with other constraints.

3.3. Controller

To achieve *Objective 2*, the controller of each quadrotor should minimize its control input. Additionally, according to the approach in [12], the control inputs $\nu_{\mathrm{f},i}(t)$ and $\nu_{\mathrm{p},i}(t)$ in Eq. (7) also need to be minimized. Furthermore, $\delta_{\mathrm{acc},i}(t)$ in Eq. (14), along with $\delta_{\mathrm{f},i}(t)$ in Eq. (19) and $\delta_{\mathrm{p},i}(t)$ in Eq. (20) are subject to minimization as well. Therefore, the cost function to be minimized is

$$J_i(t) = \int_0^{T_{\mathrm{e}}} \big(|u_i(t)| + w\nu_{\mathrm{f},i}(t) + w\nu_{\mathrm{p},i}(t) + r\delta_{\mathrm{f},i}^2(t) + r\delta_{\mathrm{p},i}^2(t) + p_{\mathrm{acc}}\delta_{\mathrm{acc},i}^2(t) \big)\, \mathrm{d}t.$$

Here, $w > 0$, $r > 0$, and $p_{\mathrm{acc}} > 0$. We note that $\nu_{\mathrm{f},i}(t)$ and $\delta_{\mathrm{f},i}(t)$ are excluded in the case of the first quadrotor ($i = 1$), while $\nu_{\mathrm{p},i}(t)$ and $\delta_{\mathrm{p},i}(t)$ are excluded in the case of the last quadrotor ($i = n$).

Finally, we summarize the previous discussion and formulate the QP. To achieve ACC, each quadrotor must satisfy multiple constraints as described in Sections 3.1 and 3.2, including *Constraints 1* and *2*, while pursuing the objectives of maintaining the desired speed (*Objective 1*) and minimizing the jerk input $u_i(t)$ (*Objective 2*).

Thus the QP to be solved is

$$\boldsymbol{u}_i^*(t) = \underset{\boldsymbol{u}_i(t)}{\operatorname{argmin}} \;\; \frac{1}{2}\boldsymbol{u}_i^{\mathrm{T}}(t) H_i \boldsymbol{u}_i(t) + F_i^{\mathrm{T}} \boldsymbol{u}_i(t)$$

$$\text{s.t.} \;\; \begin{cases} (12) – (20), \; i \in \{2, \ldots, n-1\} \\ (13) – (16), (18), (20), \;\; i = 1 \\ (12), (14) – (17), (19), \;\; i = n \end{cases}$$

$$H_i = \begin{bmatrix} 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2p_{\mathrm{acc}} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2r & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2r \end{bmatrix},$$

$$F_i = \begin{bmatrix} 0 \\ 0 \\ w \\ 0 \\ w \\ 0 \end{bmatrix}, \; \boldsymbol{u}_i(t) = \begin{bmatrix} u_i(t) \\ \delta_{\mathrm{acc},i}(t) \\ \nu_{\mathrm{p},i}(t) \\ \delta_{\mathrm{p},i}(t) \\ \nu_{\mathrm{f},i}(t) \\ \delta_{\mathrm{f},i}(t) \end{bmatrix}, \; i \in \{2, \ldots, n-1\}$$

$$H_i = \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 2p_{\mathrm{acc}} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2r \end{bmatrix}, \; F_i = \begin{bmatrix} 0 \\ 0 \\ w \\ 0 \end{bmatrix}, \; i = 1, \, n$$

$$\boldsymbol{u}_1(t) = \begin{bmatrix} u_1(t) \\ \delta_{\mathrm{acc},1}(t) \\ \nu_{\mathrm{p},1}(t) \\ \delta_{\mathrm{p},1}(t) \end{bmatrix}, \boldsymbol{u}_n(t) = \begin{bmatrix} u_n(t) \\ \delta_{\mathrm{acc},n}(t) \\ \nu_{\mathrm{f},n}(t) \\ \delta_{\mathrm{f},n}(t) \end{bmatrix}.$$

Since each quadrotor solves its own optimization problem based solely on its own state variables, as well as the relative distance, velocity, and acceleration with respect to its neighboring quadrotors, it can independently determine the control input. This approach eliminates the need for communication between quadrotors, enabling a fully distributed control strategy. Furthermore, since the PACBF is used for collision avoidance constraints, conflicts with input constraints can be prevented, which is expected to improve the feasibility of the QP.

4. SIMUMLATION RESULTS

Through the simulation of a five-quadrotors platoon, we confirm that the inter-quadrotor distance control which considers both the preceding and following quadrotors is achieved. The QPs are solved using MATLAB's `quadprog` command, and the dynamics are integrated using `ode45`. The initial positions of the quadrotors are $x_1(0) = 170\,[\mathrm{m}]$, $x_2(0) = 160\,[\mathrm{m}]$, $x_3(0) = 130\,[\mathrm{m}]$, $x_4(0) = 80\,[\mathrm{m}]$, and $x_5(0) = 0\,[\mathrm{m}]$. Additionally, the initial velocities are $v_1(0) = 5\,[\mathrm{m/s}]$, $v_2(0) = 6\,[\mathrm{m/s}]$, $v_3(0) = 8\,[\mathrm{m/s}]$, $v_4(0) = 10\,[\mathrm{m/s}]$, and $v_5(0) = 12\,[\mathrm{m/s}]$, the desired velocities are $v_{\mathrm{d},1} = 5\,[\mathrm{m/s}]$, $v_{\mathrm{d},2} = 8\,[\mathrm{m/s}]$, $v_{\mathrm{d},3} = 10\,[\mathrm{m/s}]$, $v_{\mathrm{d},4} = 12\,[\mathrm{m/s}]$, and $v_{\mathrm{d},5} = 14\,[\mathrm{m/s}]$, and the initial accelerations are $a_{1\sim5}(0) = 0\,[\mathrm{m/s^2}]$. Therefore, the initial pitch angles are $\theta_{1\sim5}(0) = 0\,[\mathrm{rad}]$.

Furthermore, the initial values of the penalty functions are $p_{\mathrm{p},i}(0) = p_{\mathrm{f},i}(0) = 0.5\ \forall i \in \{1, \ldots, 5\}$, and their desired values are $p_{\mathrm{pd}} = p_{\mathrm{fd}} = 0.7$. The other simulation parameters are set as follows: the maximum allowable pitch angle $\theta_{\mathrm{c}} = 0.2\,[\mathrm{rad}]$, the safe distance $d = 2.0\,[\mathrm{m}]$, $T = 2.0$, $\alpha = 0.01$, $\alpha_1 = 1$, $\alpha_{\mathrm{p}} = 1$, $\epsilon = 10$, $p_{\mathrm{acc}} = e^{-12}$, $r = 0.5$, $w = 2e^{-12}$, $k_{\mathrm{p}} = 30$, and $k_{\mathrm{d}} = 70$.

The simulation results are shown in Fig. 3. Figs. 3(a)–(d) illustrate the time responses of the velocities, accelerations, pitch angles, and jerk inputs, respectively. By observing these figures along with Fig. 3(e), which shows the CBF $h_i(\boldsymbol{q}_{i-1}, \boldsymbol{q}_i(t))$, it can be seen that as the inter-quadrotor distance decreases, the following quadrotor decelerates while the preceding quadrotor accelerates. The ability to accelerate for collision avoidance is due to the CBF being allowed to converge with a certain margin ($h_i(\boldsymbol{q}_{i-1}, \boldsymbol{q}_i) \neq 0$). This ensures that even with acceleration, the condition $h_i(\boldsymbol{q}_{i-1}, \boldsymbol{q}_i) \geq 0$ is maintained. This approach helps maintain the inter-quadrotor distance above a safe threshold. Therefore, it can be concluded that the quadrotors cooperatively regulate their spacing. Furthermore, Fig. 3(f) presents the time response of the penalty function. From this figure and Fig. 3(b), it is shown that by adjusting the penalty function and dynamically adapting the barrier condition when the acceleration changes, the occurrence of infeasibility in the QP problem is prevented, thereby enabling inter-quadrotor distance control.

5. CONCLUSION

This paper presented a distributed ACC method for quadrotors that considers both the preceding and following vehicles. It demonstrates that by solving an optimization problem with collision avoidance constraints that take both the preceding and following vehicles into account, each quadrotor can appropriately switch between acceleration and deceleration depending on the situation, thus achieving cooperative cruise control. Unlike traditional methods that consider only the preceding vehicle, the proposed method allows acceleration for collision avoidance—a feature not previously possible.

The future directions of this work are the following: (i) investigating the potential for oscillatory behavior, such as repeated acceleration and deceleration, resulting from the increased degree of freedom in acceleration, and exploring possible solutions to mitigate such effects; (ii) demonstrating the effectiveness of the method in real-world scenarios through experiments using quadrotors; and (iii) extending the quadrotor ACC to 3D space, including the y-axis and z-axis directions.

REFERENCES

[1] D. Corona, M. Lazar, B. Schutter, and M. Heemels, "A hybrid MPC approach to the design of a smart adaptive cruise controller", in *Proceedings of the 2006 IEEE International Conference on Contol Applications*, pp. 231–235, 2006.

[2] V. Bageshwar, W. Garrard, and R. Rajamani, "Model predictive control of transitional maneuvers for adaptive cruise control vehicles", *IEEE Transactions on Vehicular Technology*, Vol. 53, No. 5, pp. 365–374, 2004.

[3] W. Wang, W. Xiao, A. Gonzalez-Garcia, J. Swevers, C. Ratti, and D. Rus, "Robust model predictive control with control barrier functions for autonomous surface vessels", in *Proceedings of the 2024 IEEE International Conference on Robotics and Automation*, pp. 6089–6095, 2024.

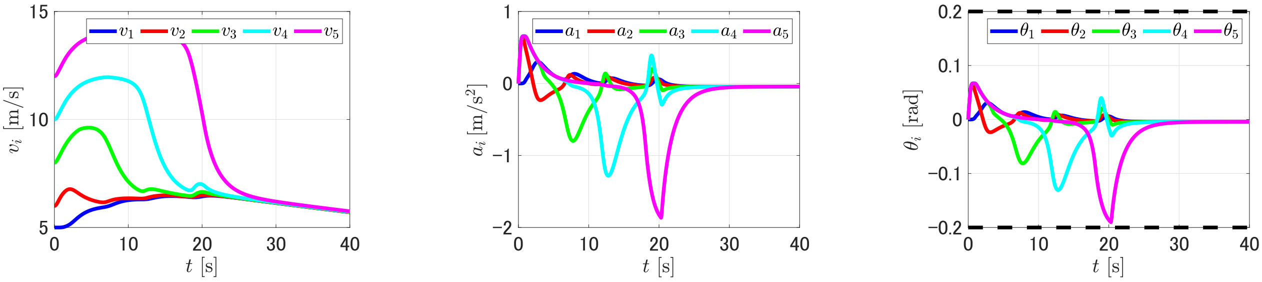
[4] P. Wieland and F. Allgöwer, "Construvtive safety using control barrier functions", in *Proceedings of the 7th IFAC Symposium on Nonlinear Control System*, pp. 462–467, 2007.

[5] T. Ibuki, T. Hirano, R. Funada, and M. Sampei, "Optimization-based distributed safety control with applications to collision avoidance for mobile robotic networks", *Advanced Robotics*, Vol. 37, No. 1–2, pp. 87–98, 2023 .

[6] A. D. Ames, K. Galloway, and J.W. Grizzle, "Control lyapunov functions and hybrid zero dynamics", in *Proceedings of the 51st IEEE Conference on Decision and Control*, pp. 6837–6842, 2012.

[7] A. D. Ames, J.W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs with applications to adaptive cruise control", in *Proceedings of the 53rd IEEE Conference on Decision and Control*, pp. 6271–6278, 2014.

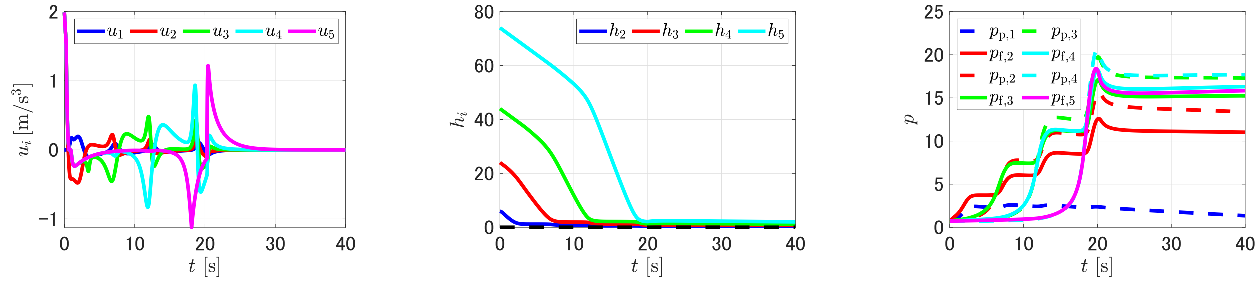
[8] S. C. Hsu, X. Xu, and A. D. Ames, "Control barrier



(a) Time responses of velocities.

(b) Time responses of accelerations.

(c) Time responses of pitch angles.



(d) Time responses of jerk inputs.

(e) Time responses of h_i.

(f) Time responses of penalty functions.

Fig. 3 Simumlation results.

function based quadratic programs with application to bipedal robotic walking”, in *Proceedings of the American Control Conference 2015*, pp. 4542–4548, 2015.

[9] Q. Nguyen and K. Sreenath, “Exponential control barrier functions for enforcing high relative-degree safety-critical constraints”, in *Proceedings of the American Control Conference 2016*, pp. 322–328, 2016.

[10] W. Xiao and C. Belta, “Control barrier function for systems with high relative degree”, in *Proceedings of the 58th IEEE Conference on Decision and Control*, pp. 474–479, 2019.

[11] A. J. Taylor and A. D. Ames, “Adaptive safety with control barrier function”, in *Proceedings of the American Control Conference 2020*, pp. 1399–1405, 2020.

[12] W. Xiao, C. Belata, and C. G. Cassandras, “Adaptive control barrier functions”, *IEEE Transactions on Automatic Control*, Vol. 67, No. 5, pp. 2267–2281, 2022.

[13] L. R. G. Carrillo, A. E. D. López, R. Lozano, and C. Pégard, *Quad Rotorcraft Control: Vision-Based Hovering and Navigation*, Springer, 2013.

[14] J. C. Hayward, “Near miss determination through use of a scale of danger”, in *Proceedings of the 51st Annual Meeting of the Highway Research Board* , Vol. 384, pp. 24–34, 1972.

APPENDIX

We explain the HOCBF [10] and the PACBF [12] necessary for deriving collision avoidance constraints. In this section, we consider the following affine system:

$$\dot{\boldsymbol{q}} = f_o(\boldsymbol{q}) + g_o(\boldsymbol{q})\boldsymbol{u}. \tag{A.1}$$

In this formulation, $\boldsymbol{q} \in \mathbb{R}^{n_q}$, $f_o : \mathbb{R}^{n_q} \to \mathbb{R}^{n_q}$, and $g_o : \mathbb{R}^{n_q} \to \mathbb{R}^{n_q \times n_u}$ are locally Lipschitz, and $\boldsymbol{u} \in U \subset \mathbb{R}^{n_u}$. U denotes the control constraint set defined as

$$U := \{\boldsymbol{u} \in \mathbb{R}^{n_u} : \boldsymbol{u}_{\min} \leq \boldsymbol{u} \leq \boldsymbol{u}_{\max}\},$$

with $\boldsymbol{u}_{\min}, \boldsymbol{u}_{\max} \in \mathbb{R}^{n_u}$ are the lower and upper bounds of the control input, respectively.

To begin with, we introduce the HOCBF to handle constraints with a relative degree greater than one. In this paper, since function h is used to define a constraint $h(\boldsymbol{q}) \geq 0$, we will refer to the relative degree of h as the relative degree of the constraint.

For a constraint $h(\boldsymbol{q}) \geq 0$ with relative degree m, $h : \mathbb{R}^{n_q} \to \mathbb{R}$, and $\psi_0(\boldsymbol{q}) := h(\boldsymbol{q})$, we define a sequence of functions $\psi_j : \mathbb{R}^{n_q} \to \mathbb{R}$, $j \in \{1, \ldots, m\}$ as

$$\psi_j(\boldsymbol{q}) := \dot{\psi}_{j-1}(\boldsymbol{q}) + \alpha_j(\psi_{j-1}(\boldsymbol{q})), \tag{A.2}$$

where $\alpha_j(\cdot)$ denotes a $(m-j)$th-order differentiable class $\mathcal{K}$ function. Furthermore, we define a sequence of sets C_j, $j \in \{1, \ldots, m\}$ associated with Eq. (A.2) as

$$C_j := \{\boldsymbol{q} \in \mathbb{R}^{n_q} : \psi_{j-1}(\boldsymbol{q}) \geq 0\}. \tag{A.3}$$

The HOCBF is defined as follows:

Definition 3 (HOCBF [10]): A function $h : \mathbb{R}^{n_q} \to \mathbb{R}$ is said to be an HOCBF of relative degree m for system (A.1) if there exist $(m-j)$th-order differentiable class $\mathcal{K}$ functions α_j, $j \in \{1, \ldots, m-1\}$ and class $\mathcal{K}$ function

α_m such that

$$\sup_{\boldsymbol{u}\in U}\big(\mathcal{L}_{f_o}^{m}h(\boldsymbol{q}) + (\mathcal{L}_{g_o}\mathcal{L}_{f_o}^{m-1}h(\boldsymbol{q}))\boldsymbol{u} + S(h(\boldsymbol{q})) + \alpha_m(\psi_{m-1}(\boldsymbol{q}))\big) \geq 0, \quad \text{(A.4)}$$

for all $\boldsymbol{q} \in C_1\cap,\dots,\cap C_m$. Here, $\mathcal{L}_{f_o}^{m}$ and $\mathcal{L}_{g_o}$ are the Lie derivatives along f_o (g_o) m (one) times, and

$$S(h(\boldsymbol{q})) = \sum_{j=1}^{m-1}\mathcal{L}_{f_o}^{j}(\alpha_{m-j}\circ\psi_{m-j-1})(\boldsymbol{q}),$$

where '$\circ$' denotes the composition of functions. Moreover, $h(\boldsymbol{q})$ is such that $\mathcal{L}_{g_o}\mathcal{L}_{f_o}^{m-1}h(\boldsymbol{q}) \neq 0$ on the boundary of the set $C_1\cap,\dots,\cap C_m$.

Then, the following fact holds:

Fact 1 ([10]): Given an HOCBF $h(\boldsymbol{q})$ from Definition 3 with the associated sets $C_1,\dots,C_m$ defined by Eq. (A.3), if $\boldsymbol{q}(0) \in C_1\cap,\dots,\cap C_m$, then any Lipschitz continuous controller $\boldsymbol{u}(t)$ that satisfies Eq. (A.4) for all $t \geq 0$ renders $C_1\cap,\dots,\cap C_m$ forward invariant for system (A.1).

Subsequently, we introduce the PACBF. For a constraint $h(\boldsymbol{q}) \geq 0$ with relative degree m, $h: \mathbb{R}^{n_q} \to \mathbb{R}$, and $\psi_0(\boldsymbol{q}) := h(\boldsymbol{q})$, we define a time-varying penalty function $p_j(t) \geq 0$, $j \in \{1,\dots,m\}$ and let $\boldsymbol{p}(t) := [p_1(t)\ \dots\ p_m(t)]^{\mathrm{T}}$. We also define the sequence of functions, in the same way as Eq. (A.2), as follows:

$$\begin{aligned}
&\psi_1(\boldsymbol{q},\boldsymbol{p}(t)) := \dot{\psi}_0(\boldsymbol{q}) + p_1(t)\alpha_1(\psi_0(\boldsymbol{q})),\\
&\psi_j(\boldsymbol{q},\boldsymbol{p}(t)) := \dot{\psi}_{j-1}(\boldsymbol{q},\boldsymbol{p}(t)) + p_j(t)\alpha_j(\psi_{j-1}(\boldsymbol{q},\boldsymbol{p}(t))),\\
&\qquad\qquad j \in \{2,\dots,m\}. \quad \text{(A.5)}
\end{aligned}$$

Here, $\alpha_j(\cdot)$, $j \in \{1,\dots,m-1\}$ is a $(m-j)$th-order differentiable class $\mathcal{K}$ function, and $\alpha_j(\cdot)$ is a class $\mathcal{K}$ function.

Furthermore, we define a sequence of sets C_j, $j \in \{1,\dots,m\}$ associated with Eq. (A.5) as follows:

$$\begin{aligned}
&C_1 := \{\boldsymbol{q} \in \mathbb{R}^{n_q} : \psi_0(\boldsymbol{q}) \geq 0\},\\
&C_j := \{(\boldsymbol{q},\boldsymbol{p}(t)) \in \mathbb{R}^{n_q}\times\mathbb{R}^m : \psi_{j-1}(\boldsymbol{q},\boldsymbol{p}(t)) \geq 0\},\\
&\qquad\qquad j \in \{2,\dots,m\}. \quad \text{(A.6)}
\end{aligned}$$

Based on Definition 3 and Fact 1, it is necessary that $p_j(t) \geq 0\ \ \forall j \in \{1,\dots,m-1\}$. Therefore, we define each $p_j(t)$ to be an HOCBF and introduce auxiliary dynamics (for details, refer to [11]). As state variables for the auxiliary dynamics, define $\boldsymbol{\pi}_j(t) := [\pi_{j,1}(t)\ \pi_{j,2}(t)\ \dots\ \pi_{j,m-j}(t)]^{\mathrm{T}} \in \mathbb{R}^{m-j}$, $j \in \{1,\dots,m-2\}$, where $\pi_{j,k}(t) \in \mathbb{R}$, $k \in \{1,\dots,m-j\}$. In addition, we define $\boldsymbol{\pi}_{m-1}(t) = p_{m-1}(t) \in \mathbb{R}$, which is differentiable of the first order. Finaly, we define input-output linearizable auxiliary dynamics for each p_j through the auxiliary state $\boldsymbol{\pi}_j$ as follows:

$$\begin{aligned}
&\dot{\boldsymbol{\pi}}_j = F_j(\boldsymbol{\pi}_j) + G_j(\boldsymbol{\pi}_j)\nu_j,\ j \in \{1,\dots,m-1\}, \quad \text{(A.7)}\\
&y_j = p_j.
\end{aligned}$$

Here, y_j denotes the output, $F_j : \mathbb{R}^{m-j} \to \mathbb{R}^{m-j}$, $G_i : \mathbb{R}^{m-j} \to \mathbb{R}^{m-j}$, and $\nu_j \in \mathbb{R}$ denotes the control input for the auxiliary dynamics (A.7). The exact forms of F_j and G_j are used to guarantee the nonnegativity of p_j, so they do not have a significant impact on the system performance. For simplicity, linear forms are typically adopted [11].

We define $\boldsymbol{\nu} := [\nu_1\ \dots\ \nu_{m-1}]^{\mathrm{T}}$, where ν_j, $j \in \{1,\dots,m-1\}$ are the control inputs of the auxiliary dynamics (A.7). Since p_j is an HOCBF with relative degree $m-j$ for dynamics (A.7), based on Eq. (A.4), we define a constraint set U_{cbf} for $\boldsymbol{\nu}$ $\forall j \in \{1,\dots,m-1\}$ as

$$\begin{aligned}
U_{\mathrm{cbf}} = \{\boldsymbol{\nu} \in \mathbb{R}^{m-1} : \mathcal{L}_{F_j}^{m-j}p_j + (\mathcal{L}_{G_j}\mathcal{L}_{F_j}^{m-j-1}p_j)\nu_j\\
+ S(p_j) + \alpha_{m-j}(\psi_{j,m-j-1}(p_j)) \geq 0\},
\end{aligned}$$

where $\psi_{j,m-j-1}(\cdot)$ is defined similarly to Eq. (A.2)

The PACBF is defined as follows:

Definition 4 (PACBF [12]): Let C_j, $j \in \{1,\dots,m\}$ be defined by Eq. (A.6), $\psi_j(\boldsymbol{q},\boldsymbol{p})$, $j \in \{1,\dots,m\}$ be defined by Eq. (A.5), and the auxiliary dynamics be defined by Eq. (A.7). A function $h : \mathbb{R}^{n_q} \to \mathbb{R}$ is said to be a PACBF of relative degree m for system (A.1) if each p_j, $j \in \{1,\dots,m-1\}$ is an HOCBF with relative degree $m-j$ for the auxiliary dynamics (A.7), and there exist $(m-j)$th-order differentiable class $\mathcal{K}$ functions α_j, $j \in \{1,\dots,m-1\}$, and a class $\mathcal{K}$ function α_m such that

$$\begin{aligned}
\sup_{\boldsymbol{u}\in U,\boldsymbol{\nu}\in U_{\mathrm{cbf}}}\Big(&\mathcal{L}_{f_o}^{m}h(\boldsymbol{q}) + (\mathcal{L}_{g_o}\mathcal{L}_{f_o}^{m-1}h(\boldsymbol{q}))\boldsymbol{u} + R(h(\boldsymbol{q}),\boldsymbol{p})\\
&+ \sum_{j=1}^{m-1}(\mathcal{L}_{F_j}^{m-j}p_j)\alpha_j(\psi_{j-1}) + p_m\alpha_m(\psi_{m-1})\\
&+ \sum_{j=1}^{m-1}\alpha_j(\psi_{j-1})(\mathcal{L}_{G_j}\mathcal{L}_{F_j}^{m-j-1}p_j)\nu_j\Big) \geq 0, \quad \text{(A.8)}
\end{aligned}$$

for all $\boldsymbol{q} \in C_1$, $(\boldsymbol{q},\boldsymbol{p}) \in C_2\cap,\dots,\cap C_m$, and all $p_m \geq 0$. In Eq. (A.8), $R(h(\boldsymbol{q}),\boldsymbol{p})$ denotes the remaining Lie derivative terms of $h(\boldsymbol{q})$ (or $\boldsymbol{p}$) along f_o (or F_j) with degree less than m (or $m-j$), similarly to Eq. (A.4).

Then, the following fact holds:

Fact 2 ([12]): Given a PACBF $h(\boldsymbol{q})$ from Definition 4 with the associated sets $C_1, C_2, \dots, C_m$ defined by Eq. (A.6), if $\boldsymbol{q}(0) \in C_1$ and $(\boldsymbol{q}(0),\boldsymbol{p}(0)) \in C_2\cap,\dots,\cap C_m$, then any Lipschitz continuous controllers $\boldsymbol{u}(t)$ and $\boldsymbol{\nu}(t)$ that satisfy Eq. (A.8) for all $t \geq 0$ renders the set C_1 forward invariant for system (A.1) and $C_2\cap\dots\cap C_m$ forward invariant for systems (A.1) and (A.7), respectively.

Finally, we introduce the CLF for stabilizing the velocity.

Definition 5 (CLF [6]): A continuously differentiable function $V : \mathbb{R}^{n_q} \to \mathbb{R}$ is an exponentially stabilizing CLF for system (A.1) if there exist constants $c_1 > 0, c_2 > 0, c_3 > 0$ such that for $\forall\boldsymbol{q} \in \mathbb{R}^{n_q}$, $c_1\|\boldsymbol{q}\|^2 \leq V(\boldsymbol{q}) \leq c_2\|\boldsymbol{q}\|^2$,

$$\inf_{\boldsymbol{u}\in U}(L_{f_o}V(\boldsymbol{q}) + L_{g_o}V(\boldsymbol{q})\boldsymbol{u} + c_3V(\boldsymbol{q})) \leq 0.$$