

Barrier Functions in Cascaded Controller: Safe Quadrotor Control

Mouhyemen Khan, Munzir Zafar, Abhijit Chatterjee

Abstract—Safe control for inherently unstable systems such as quadrotors is crucial. Imposing multiple dynamic constraints simultaneously on the states for safety regulation can be a challenging problem. In this paper, we propose a quadratic programming (QP) based approach on a cascaded control architecture for quadrotors to enforce safety. Safety regions are constructed using control barrier functions (CBF) while explicitly considering the nonlinear underactuated dynamics of the quadrotor. The safety regions constructed using CBFs establish a non-conservative forward invariant safe region for quadrotor navigation. Barriers imposed across the cascaded architecture allow independent safety regulation in the quadrotor’s altitude and lateral domains. Despite barriers appearing in a cascaded fashion, we show preservation of safety for quadrotor motion in $SE(3)$. We demonstrate the feasibility of our method on a quadrotor in simulation with static and dynamic constraints enforced on the position and velocity spaces simultaneously.

I. INTRODUCTION

Safety is a critical component for today’s autonomous aerial systems [1], [2], [3]. Of particular interest are quadrotors due to their application in surveillance, agriculture, and acrobatic performances, see [4], [5], [6]. Thus accentuating the need for safety as an imperative requirement during flight operation. Control Barrier Functions (CBFs) have proven to be an effective strategy for guaranteeing safety in several applications [7],[8], including quadrotors [9]. The focus of this paper is to rectify the quadrotor’s nominal trajectory using a cascaded controller to ensure safety in 3D position and velocity spaces. We achieve this by independently imposing barriers across the cascaded hierarchy.

The underactuated and intrinsically unstable nature of quadrotor makes it challenging to generate safe trajectories [10]. CBFs, first implemented in adaptive cruise control [7] formed as an online quadratic program (QP), permit dynamically feasible constraints and ensure forward invariance. CBFs were used in collision avoidance for swarms of mobile ground robots [9] and quadrotors [11]. They

restricted their safety constraints to the position space only. CBFs were also used to learn quadrotor dynamics in the presence of wind disturbances [12]. This work uses a differential flatness model and a single CBF in its controller scheme with a focus on learning unmodeled dynamics. The works in [13] and [14] use sequential QP based methods augmenting the CBF with a Control Lyapunov Function (CLF) for obstacle avoidance resulting in a CLF-CBF-QP controller. They propose a sequential optimization scheme where virtual thrust is first computed using position level QP satisfying a CLF constraint, while CBF constraints are incorporated in the lower level QP to generate the control inputs. Their controller, therefore, results in a sequential CLF-CBF-QP formulation. The work in [13] and [14] impose safety explicitly in $SE(2)$ and $SE(3)$ respectively. In order to find a feasible solution for QP, input bounds are not considered in [14] (although quadrotors have input bounds, e.g. no reverse thrust).

Key contributions: In contrast to prior research, our contributions in this paper are threefold:

- Prior work has not merged the forward invariance of CBFs at *every level of the hierarchy* in cascaded controller to ensure safety for quadrotors. The cascaded control scheme is a popular architecture and imposing safety on such a controller is yet to be formulated. We present derivations for enforcing constraints across the hierarchy by considering the complete 3D quadrotor dynamics evolving in tangent bundle to $SE(3)$.
- CBFs are employed in a nonlinear cascaded controller with constraints *explicitly* imposed on the position and velocity spaces. For safety-critical tracking of complex trajectories, imposing safety on velocity along with position is critical. We have also empirically verified in simulation that our QP formulation with actuator bounds found feasible solution. To the best of our knowledge, this is the first time constraints on position and velocity spaces are explicitly formulated for safe 3D quadrotor control along with actuator bounds.
- Safety constraints are handled in the altitude and lateral

Mouhyemen Khan, Munzir Zafar, and Abhijit Chatterjee are with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA: {mouhyemen.khan, mzafar7, abhijit.chatterjee}@gatech.edu

domains of the quadrotor *independently*. We place separate CBF constraints on the outer loop and inner loop of the controller. This has two advantages: (i) decoupling the safety constraint results in a richer superlevel safe set and (ii) this allows independent regulation of quadrotor flight in altitude and lateral domains.

The rest of the paper is organized as follows. Section II introduces preliminaries on barrier functions and quadrotor dynamics. Cascaded controller with augmented QP design is discussed in Section III while safety barrier formulations across the hierarchy are derived in Section IV. Simulation results are provided in Section V, followed by conclusions in Section VI.

II. PRELIMINARIES: BARRIER FUNCTIONS AND QUADROTOR DYNAMICS

This section introduces CBFs along with its extension called Exponential CBFs (ECBFs) and dynamics of a quadrotor in 3D. These topics are well studied, hence for a more detailed discussion on CBFs, ECBFs, and quadrotor dynamics, we refer the reader to [7], [15], and [16] respectively.

A. Control Barrier Functions

Consider a general control affine dynamical system,

$$\dot{x} = f(x) + g(x)u, \quad x(t_0) = x_0, \quad (1)$$

where $x \in \mathcal{X} \subseteq \mathbb{R}^n$ is the state and $u \in \mathcal{U} \subseteq \mathbb{R}^m$ is the control input of the system. Both the drift and control vector fields, $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ and $g : \mathbb{R}^n \rightarrow \mathbb{R}^m$ respectively, are assumed to be Lipschitz continuous. Let the safe state space of the system be encoded as the superlevel set \mathcal{S} of a smooth function $h : \mathcal{X} \rightarrow \mathbb{R}$ as follows,

$$\mathcal{S} = \{x \in \mathbb{R}^n \mid h(x) \geq 0\}. \quad (2)$$

Definition 1 [7]: *The function $h(x) : \mathcal{X} \rightarrow \mathbb{R}$ is defined as a control barrier function (CBF), if \exists an extended class- κ function ($\kappa(0) = 0$ and strictly increasing) such that $\forall x \in \mathcal{S}$,*

$$\sup_{u \in \mathcal{U}} \left\{ L_f h(x) + L_g h(x)u + \kappa(h(x)) \right\} \geq 0, \quad (3)$$

where, $L_f h(x)$ and $L_g h(x)$ stand for the Lie derivative of $h(x)$ along the vector fields $f(x)$ and $g(x)$ respectively.

Theorem [7]: *Given a system defined by (1), with safe set $\mathcal{S} \subset \mathbb{R}^n$ defined by (2), and smooth CBF $h(x) : \mathcal{S} \rightarrow \mathbb{R}$ defined by (3), \forall Lipschitz continuous $u \in \mathcal{U}$ that satisfies, $\bar{\mathcal{U}} = \{u \in \mathcal{U} \mid L_f h(x) + L_g h(x)u + \kappa(h(x)) \geq 0\}$, $\forall x \in \mathcal{X}$, then the safe set \mathcal{S} is forward invariant for the system.*

CBFs are limited in their nature to systems with relative degree one, i.e., $\delta = 1$, where $\delta \in \mathbb{W}$ [15]. Depending on how one enforces barriers around state(s), relative degree can go above 1. Thus, CBFs cannot be directly applied for such barrier constraints. For $\delta > 1$, an extension of the CBF, called Exponential CBF (ECBF), is used to guarantee forward invariance property of \mathcal{S} [15].

Definition 2 [15]: *The smooth function $h(x) : \mathcal{X} \rightarrow \mathbb{R}$, with relative degree δ , is defined as an exponential control barrier function (ECBF), if $\exists \mathcal{K} \in \mathbb{R}^\delta$ such that $\forall x \in \mathcal{S}$,*

$$\sup_{u \in \mathcal{U}} \left\{ L_f^\delta h(x) + L_g L_f^{\delta-1} h(x)u + \mathcal{K}^\top \mathcal{H} \right\} \geq 0, \quad (4)$$

where $\mathcal{H} = [h(x), L_f h(x), L_f^2 h(x), \dots, L_f^{\delta-1} h(x)]^\top$ is the vector of Lie derivatives for $h(x)$, and $\mathcal{K} = [k_0, k_1, \dots, k_{\delta-1}]$ is the vector of coefficient gains for \mathcal{H} . The coefficient gain vector \mathcal{K} can be determined using pole placement technique on the closed-loop matrix $(F - GK)$ determined from $h(x) \geq C e^{F-GK} \mathcal{H}(x_0) \geq 0$, when $C = [1, 0, 0, \dots, 0]^\top \in \mathbb{R}^\delta$, $h(x_0) \geq 0$ [15]. Forward invariance is also satisfied for ECBFs and we refer the reader to [15] for detailed proofs.

B. Dynamics of 3D Quadrotor

3D Quadrotor is a dynamical system whose motion evolves in the Lie Group $SE(3)$. Hence, it is described with six degrees of freedom: translational position, $r = [x, y, z]^\top$, in the inertial frame and attitude represented by Euler angles, $\eta = [\phi, \theta, \psi]^\top$, in the body-fixed frame (see Figure 1). Quadrotor dynamics is well studied in the literature [16],[17], hence we only present equations governing dynamics.

The translational acceleration of the quadrotor is [16],

$$\ddot{r} = gz_w - \mathbf{R}z_w \frac{f(t)}{m}, \quad (5)$$

where $z_w = [0, 0, 1]^\top$, m is its mass, g is gravity, and $f(t) \in \mathbb{R}$ is the total thrust. $\mathbf{R} \in SO(3)$ is the rotation matrix of body-fixed frame \mathbf{B} and its evolution is given by [16],

$$\dot{\mathbf{R}}(t) = \mathbf{R}(t)[\Omega(t)]_\times, \quad (6)$$

where $[\cdot]_\times$ is the overloaded operator for skew-symmetric representation of the angular velocity $\Omega = [p, q, r]^\top$. In the body-fixed frame \mathbf{B} , the angular acceleration is given by [16],

$$\mathbf{I}\dot{\Omega} = \tau - \Omega \times \mathbf{I}\Omega, \quad (7)$$

where $\tau = [\tau_x, \tau_y, \tau_z]^\top$ are the moments along each principal axis and \mathbf{I} is the inertia matrix of the quadrotor. The quadrotor system is control affine with its full state as $x = [r, \eta, \dot{r}, \Omega]^\top$ and control input $u = [f, \tau_x, \tau_y, \tau_z]^\top$.

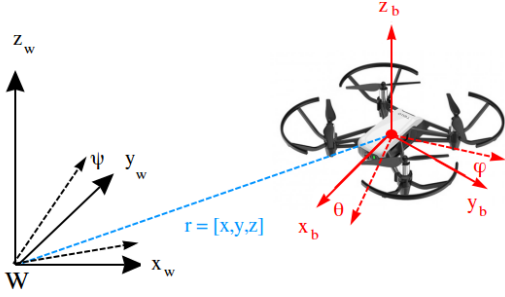


Fig. 1. World frame \mathbf{W} (black) and body frame \mathbf{B} (red) are shown along with Euler angles. The quadrotor's position vector in \mathbf{W} is also marked (blue dashed). DJI Tello Drone is used as base model for illustration.

III. NONLINEAR QP CASCADED ARCHITECTURE

We first discuss nonlinear cascaded controller for quadrotors [16] and present its design. We then discuss details for the barrier-enforced QP modification to the controller. Solving QP online is fast and has been implemented successfully on cruise control [7], bipeds [8], and quadrotors [9]-[12].

A. Motivation

The cascaded controller is a popular control architecture with practical feasibility and satisfactory performance [6], [10], [16], [17]. The design is intuitive and is commonly used by students, developers, and/or hobbyists. In prior work, CBFs have been successfully deployed on several controllers such as differential flatness and CLF-CBF-QP [14],[18],[12]. However, augmentation of CBFs to enforce safety in a cascaded control framework, to the best of our knowledge, has not been investigated. By enforcing CBFs in this framework, we aim to provide safety-critical control using a cascaded online QP controller.

B. Controller Design

The cascaded terminology is due to the hierarchical approach taken while designing the controllers (see Figure 2). At the highest level is the position controller, which is further decomposed into altitude and lateral controllers. The next level controls the quadrotor's attitude. We make the following assumptions as inputs for our controller design:

- A smooth reference is given: $r_d(t) = [x_d, y_d, z_d]^\top$.
- A yaw reference trajectory is given: $\psi_d(t)$.
- Observability of states: $x(t) = [r, \eta, \dot{r}, \Omega]^\top$.

Our controller framework is modeled after [17]. The position controller's commanded accelerations are computed as a second-order system: $\ddot{r}_{cmd}(t) = \ddot{r}_d(t) + K_p e_r(t) + K_d \dot{e}_r(t)$, where $e_r(t) = r(t) - r_d(t)$, K_p and K_d are positive definite proportional and derivative gain matrices. Using (5) and

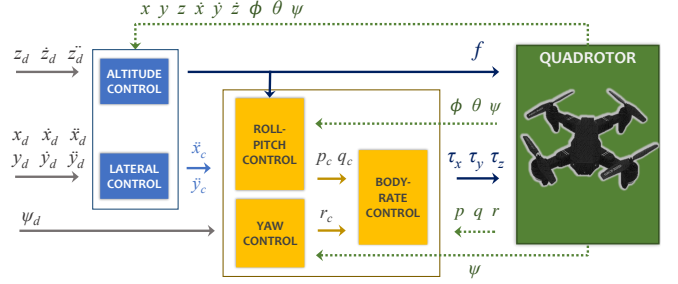


Fig. 2. The cascaded controller has a position loop (blue boxed) and attitude loop (yellow boxed). Reference inputs (grey solid) are provided to position and yaw controllers. The altitude controller generates **desired thrust**. The attitude loop orients roll-pitch and, separately, yaw with the body-rate controller generating **desired torques**. State measurements (green dotted) to controllers and control inputs (black solid) to quadrotor are shown.

altitude commanded acceleration \ddot{z}_{cmd} , we get the thrust,

$$f(t) = \frac{m}{R_{33}}(g - \ddot{z}_{cmd}), \quad (8)$$

where R_{33} is an entry of \mathbf{R} . The attitude controller computes commanded angular velocities using \mathbf{R} and (6),

$$\begin{bmatrix} p_{cmd} \\ q_{cmd} \end{bmatrix} = \frac{1}{R_{33}} \begin{bmatrix} R_{21} & -R_{11} \\ R_{22} & -R_{12} \end{bmatrix} \begin{bmatrix} \dot{R}_{cmd}^{13} \\ \dot{R}_{cmd}^{23} \end{bmatrix} \quad (9)$$

The yaw controller computes r_{cmd} separately since rotations around the quadrotor's z_B axis do not affect the dynamics of roll and pitch. A proportional regulator is used for determining r_{cmd} along z_B : $r_{cmd}(t) = k_\psi^p(\psi(t) - \psi_d(t))$.

Finally, the body-rate controller in the attitude loop computes $\dot{p}_{cmd}(t), \dot{q}_{cmd}(t), \dot{r}_{cmd}(t)$ using proportional regulators. The moments, τ_x, τ_y, τ_z , are then computed using (7).

C. Barrier-enforced QP Cascaded Controller

Given the nominal controller $\hat{u} = [f, \tau_x, \tau_y, \tau_z]^\top$ developed above, safety barriers are enforced across the cascaded architecture. This “modifies” the nominal control ensuring the system is always safe. Inspired by cascaded design, we deploy two separate QPs subjected to their own safety constraints, thus achieving independent safety regulation in the altitude and lateral domains.

The high-level QP responsible for altitude domain is:

High-level QP: *Thrust modification*

$$F^* = \arg \min_{F \in \mathbb{R}} \frac{1}{2} \|F - \hat{u}_1\|^2 \quad (10)$$

$$\text{s.t. } \left\{ L_f^\delta h(x) + L_g L_f^{\delta-1} h(x) F + \mathcal{K}^\top \mathcal{H} \right\} \geq 0 \quad (11)$$

$$0 \leq F \leq F_{max},$$

where $\hat{u}_1 = f$ is the nominal thrust computed using (8), F^*

is the computed thrust from QP. Hence, the modified control F^* follows the nominal control \hat{u}_1 as closely as possible subject to ensuring safety requirements, thereby, relaxing strict tracking if needed to meet safety requirements.

For ensuring safe control in lateral domain, a low-level QP is designed subject to another set of safety constraints:

Low-level QP: *Torque modification*

$$M^* = \arg \min_{M \in \mathbb{R}^2} \frac{1}{2} \|M - \hat{u}_2\|^2 \quad (12)$$

$$\text{s.t. } \left\{ L_f^\delta h(x) + L_g L_f^{\delta-1} h(x) M + \mathcal{K}^\top \mathcal{H} \right\} \geq 0 \quad (13)$$

$$|M| \leq M_{max},$$

where $\hat{u}_2 = [\tau_x, \tau_y]^\top$ are the nominal torques computed and M^* is the resultant torque vector from QP. Construction of Lie derivatives $L_f^\delta h(x)$ and $L_g L_f^{\delta-1} h(x)$ for high-level and low-level are discussed below in IV-B and IV-C respectively.

The formulation is thus a cascaded online QP controller that is decoupled in its safety objectives. One layer enforces safety at the high level for altitude domain modifying input f inside the *altitude controller*. The second layer enforces safety at the lower level for lateral domain modifying control inputs $[\tau_x, \tau_y]$ inside the *body-rate controller* (Figure 2).

Unlike [13], [14] which employ a sequential QP based design, and imposing barriers only at the lower level, we place separate barrier constraints at both levels of the cascaded scheme. Moreover, [14] does not consider input bounds while solving for F and M to find feasible solutions for their QP. We have empirically verified in simulation that our QP formulation with actuator bounds found feasible solution. Quadrotors already employed with cascaded controllers, with very minimal modification, can incorporate safety through our method.

IV. FORMULATION OF SAFETY BARRIERS

To ensure quadrotor's safety, we impose limits on position and velocity states using rectellipsoidal safety regions. Inclusion of velocity based constraints explicitly alongside position is imperative as it prevents aggressive braking.

A. Rectellipsoidal Safety Barrier Regions

The forward invariance property and ellipsoidal model of a safety region is illustrated in Figure 3. Inside the safe region, the system's states are allowed to evolve and approach the boundary. Outside the safe region, the control barrier function ensures the system asymptotically approaches the

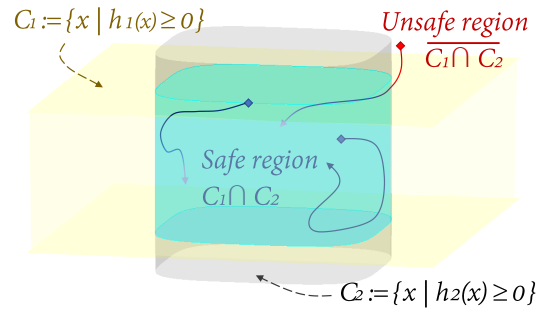


Fig. 3. Safety barrier regions at altitude and lateral domains ensure forward invariance of states using CBFs. The curves represent state evolution and diamonds represent initial states. Due to imposing separate barriers across the cascaded hierarchy, the intersected volume of safe region $C_1 \cap C_2$ results in a richer superlevel set.

safe region due to CBF constraints (see [7] for proof). In our work, the safety barrier region is modeled as,

$$h(x_i, \dots, x_n) = 1 - \left[\frac{x_i - c_i}{p_i} \right]^r + \dots + \left[\frac{x_n - c_n}{p_n} \right]^r \geq 0, \quad (14)$$

where r is the *curve* of the ellipse, x_i is the state of interest, c_i is the ellipse's center, and p_i is the limit enforced on the state. We choose $r = 4$, which is called *rectellipse*, resulting in a richer superlevel set and hence granting greater freedom for determining the safe region (Figure 3). Inspired by the work in [12], where ellipsoidal safe regions ($r = 2$) were used to learn quadrotor dynamics using CBFs in the presence of wind disturbances, we also use a similar safety region for ensuring safety of the quadrotor's state space.

B. High-level Altitude Domain Safety Objective

We now look at the high-level safety objective. The overall thrust for the quadrotor is generated by the altitude controller, thereby affecting the quadrotor's altitude position and velocity. In order to enforce limits on the altitude state(s), the following safety barrier region is used,

$$h(z) = 1 - \left[\frac{z - c_z}{p_z} \right]^4 \geq 0. \quad (15)$$

We then compute its Lie derivatives until the control input $u_1 = f(t)$ appears resulting in a relative degree $\delta = 2$. The Lie derivatives are given by,

$$\begin{aligned} \diamond L_f h(z) &= \frac{-4(z - c)^3}{p_z^4} \dot{z} \\ \diamond L_g L_f h(z) &= \frac{4(z - c)^3 R_{33}}{p_z^4 m} \\ L_f^2 h(z) &= \frac{-4(z - c)^3 g}{p_z^4} - \frac{12(z - c)^2}{p_z^4} \dot{z}, \end{aligned}$$

where (5) is substituted for \ddot{z} . Since relative degree $\delta = 2$, ECFBs are used. The Lie derivatives derived serve as the constraints in (10). A single safety region is constructed to

handle both position and velocity spaces in altitude domain, with results discussed in Section V. The barrier function is,

$$\diamond h(z, \dot{z}) = 1 - \left[\frac{z - c_z}{p_z} \right]^4 - \left[\frac{\dot{z}}{v_z} \right]^4 \quad (16)$$

Note that for (16), $\delta = 1$, hence CBFs are used as opposed to ECBFs for (15). We only present the Lie derivatives for the position space since it has a higher relative degree than the velocity space and the derivation is similar.

C. Low-level Lateral Domain Safety Objective

The lower-level safety objective allows enforcing safety limits for movement in the lateral space. The safety limits enforced on lateral positional states x and y are given by,

$$h(x, y) = 1 - \left[\frac{x - c_x}{p_x} \right]^4 - \left[\frac{y - c_y}{p_y} \right]^4 \geq 0. \quad (17)$$

Unlike the altitude domain, where the control input $f(t)$ appears directly by computing Lie derivatives, the motion in the lateral plane is affected through the moments τ_x and τ_y . This involves the effect of roll and pitch to induce this lateral motion. We present the derivation of the dynamic constraints for the low-level QP based controller.

Derivation: Recall (9), where angular velocities p and q are related to rotational rates,

$$\begin{bmatrix} p \\ q \end{bmatrix} = \frac{1}{R_{33}} \begin{bmatrix} R_{21} & -R_{11} \\ R_{22} & -R_{12} \end{bmatrix} \begin{bmatrix} \dot{R}_{13} \\ \dot{R}_{23} \end{bmatrix} = \frac{1}{R_{33}} W \begin{bmatrix} \dot{R}_{13} \\ \dot{R}_{23} \end{bmatrix}$$

For convenience, we define W as the 2×2 matrix of rotational entries and $A \triangleq [p \ q]^\top$. Rewriting in terms of angular velocities gives,

$$\begin{bmatrix} \dot{R}_{13} \\ \dot{R}_{23} \end{bmatrix} = R_{33} W^{-1} A = R_{33} V A, \quad W^{-1} \triangleq V \quad (18)$$

Now, computing the time derivative for (18) results in,

$$\begin{bmatrix} \ddot{R}_{13} \\ \ddot{R}_{23} \end{bmatrix} = \dot{R}_{33} V A + R_{33} \dot{V} A + R_{33} V \dot{A} \quad (19)$$

Since angular accelerations \dot{p} and \dot{q} are related to inputs τ_x and τ_y given in (7), substituting back in (19) gives,

$$\begin{aligned} \begin{bmatrix} \ddot{R}_{13} \\ \ddot{R}_{23} \end{bmatrix} &= \dot{R}_{33} V A + R_{33} \dot{V} A + R_{33} V \begin{bmatrix} \frac{I_y - I_z}{I_x} q r + \frac{\tau_x}{I_x} \\ \frac{I_z - I_x}{I_y} p r + \frac{\tau_y}{I_y} \end{bmatrix} \\ &= \underbrace{\dot{R}_{33} V A + R_{33} \dot{V} A + R_{33} V \begin{bmatrix} \frac{I_y - I_z}{I_x} q r \\ \frac{I_z - I_x}{I_y} p r \end{bmatrix}}_{\mathcal{J}} \\ &\quad + \underbrace{R_{33} V \begin{bmatrix} I_x^{-1} & 0 \\ 0 & I_y^{-1} \end{bmatrix}}_{\mathcal{L}} \begin{bmatrix} \tau_x \\ \tau_y \end{bmatrix} \\ &= \mathcal{J} + \mathcal{L} \begin{bmatrix} \tau_x \\ \tau_y \end{bmatrix}, \end{aligned} \quad (20)$$

where \mathcal{J} and \mathcal{L} are used for simplifying expressions. Since \ddot{x} and \ddot{y} are related to rotational entries R_{13} and R_{23} through (5), we need the fourth time derivative of x and y in order to obtain \ddot{R}_{13} and \ddot{R}_{23} , thus finally relating with τ_x and τ_y .

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \end{bmatrix} = -\frac{f}{m} \begin{bmatrix} R_{13} \\ R_{23} \end{bmatrix} \quad [\text{using (5)}] \quad (21)$$

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \end{bmatrix} = -\frac{f}{m} R_{33} V A \quad [\text{using (18)}] \quad (22)$$

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \end{bmatrix} = -\frac{f}{m} \mathcal{J} - \frac{f}{m} \mathcal{L} \begin{bmatrix} \tau_x \\ \tau_y \end{bmatrix} \quad [\text{using (20)}] \quad (23)$$

Thus, time derivatives of x and y relate to control inputs τ_x and τ_y with relative degree $\delta = 4$. We next compute the Lie derivatives for lateral safety barrier region (17),

$$\begin{aligned} \diamond L_f h(x, y) &= -4\eta_3^\top \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} \\ \diamond L_f^2 h(x, y) &= -4\eta_3^\top \begin{bmatrix} \ddot{x} \\ \ddot{y} \end{bmatrix} - 12\eta_2^\top \begin{bmatrix} \dot{x}^2 \\ \dot{y}^2 \end{bmatrix} \\ \diamond L_f^3 h(x, y) &= -4\eta_3^\top \begin{bmatrix} \ddot{x} \\ \ddot{y} \end{bmatrix} - 36\eta_2^\top \begin{bmatrix} \dot{x} & 0 \\ 0 & \dot{y} \end{bmatrix} \begin{bmatrix} \ddot{x} \\ \ddot{y} \end{bmatrix} \\ &\quad - 24\eta_1^\top \begin{bmatrix} \dot{x}^3 \\ \dot{y}^3 \end{bmatrix} \\ \diamond L_g L_f^3 h(x, y) &= \frac{4f}{m} \eta_3^\top \mathcal{L} \\ L_f^4 h(x, y) &= \frac{4f}{m} \eta_3^\top \mathcal{J} - 48\eta_2^\top \begin{bmatrix} \dot{x} & 0 \\ 0 & \dot{y} \end{bmatrix} \begin{bmatrix} \ddot{x} \\ \ddot{y} \end{bmatrix} \\ &\quad - 36\eta_2^\top \begin{bmatrix} \ddot{x}^2 \\ \ddot{y}^2 \end{bmatrix} - 144\eta_1^\top \begin{bmatrix} \dot{x} & 0 \\ 0 & \dot{y} \end{bmatrix} \begin{bmatrix} \ddot{x} \\ \ddot{y} \end{bmatrix} - 24\eta_0^\top \begin{bmatrix} \dot{x}^4 \\ \dot{y}^4 \end{bmatrix}, \end{aligned}$$

where $\eta_i = [(x - c_x)^i / p_x^4, (y - c_y)^i / p_y^4]^\top$, $i \in \{0, 1, 2, 3\}$ and (20) is substituted for $[\ddot{x}, \ddot{y}]^\top$. Due to the relative degree being four, ECBFs are once again employed to satisfy the QP constraints in (12). For the velocity space of the lateral motion, the following barrier function is used,

$$h(\dot{x}, \dot{y}) = 1 - \left[\frac{\dot{x}}{v_x} \right]^4 - \left[\frac{\dot{y}}{v_y} \right]^4 \geq 0. \quad (24)$$

Note that, although, higher time derivatives of x and y are present in the Lie derivatives, they are functions of the state $([r, \eta, \dot{r}, \Omega]^\top)$. *Numerical differentiation is not required for computing these higher derivatives.* Hence, the issue of having noisy and non-smooth signals that may arise due to differentiation, especially higher order derivatives, is averted.

V. SIMULATION RESULTS

We present our simulation results using cascaded QP controller from III-C and barrier regions constructed at both levels in IV-B and IV-C. The simulation was done in MATLAB 2018b with parameters as tabulated in I to model the quadrotor. References are generated using sinusoidal curves, $r_d(t) = [a_x \sin(\omega_x t), a_y \sin(\omega_y t), a_z \sin(\omega_z t)]^\top$ and $\psi_d(t) = \text{atan2}(y_d, x_d)$. The QP is solved online using MATLAB's built-in QUADPROG solver.

A. High-Level Altitude Domain Safety Behavior

We impose safety constraints only at the high-level and demonstrate our controller's efficacy in altitude domain. Through the safety barrier regions developed for high-level altitude domain in Section IV-B, we enforce constraints on altitude position (z) and altitude velocity (\dot{z}). With barrier limits of $\pm 2m$ for position and $\pm 0.75m/s$ for velocity, as shown in Figure 4, trajectory tracking is performed as long as the reference is within the barrier limits. Tracking is relaxed if the reference violates the safety limits.

Note that we subject the quadrotor's altitude reference velocity \dot{z}_d initially to be outside the safe region. The high-level safety objective ensures the quadrotor is first brought into the safe region and contained therein.

B. Low-Level Lateral Domain Safety Behavior

Now, we only impose constraints at the low-level QP responsible for lateral domain. Our cascaded formulation allows easy regulation of quadrotor motion in the lateral domain independent of the high-level constraint objectives since constraints are imposed only at the low-level QP now.

We test our method on both the lateral position (x, y) and velocity spaces (\dot{x}, \dot{y}) and illustrate the results in Figures

Variables	Definition	Value
g	Gravitational acceleration	9.81 kg m/s^2
m	Mass of quadrotor	0.45 kg
L	Distance between two rotors	0.24 m
I_x, I_y	Inertia about x_B -, y_B -axis	0.091 kg m^2
I_z	Inertia about z_B -axis	0.182 kg m^2
k_f	Motor's thrust constant	0.88 m
k_w	Motor's torque constant	1.00 m
f_{min}	Minimum rotor thrust	0.00 kg m/s^2
f_{max}	Maximum rotor thrust	36.00 kg m/s^2
$\tau_{min}^x, \tau_{min}^y$	Min. moment about x_B, y_B -axis	-20.0 Nm
$\tau_{max}^x, \tau_{max}^y$	Max. moment about x_B, y_B -axis	20.0 Nm

TABLE I

PARAMETERS FOR MODELING THE DYNAMICS OF THE QUADROTOR.

5 and 6. As seen from the two figures, the quadrotor relaxes trajectory tracking when faced with the obligation of upholding safety. This demonstrates that safety barriers are the top priority in regulating the control action.

We also change the velocity barriers mid-way during the flight as shown in Figure 6. For both \dot{x} and \dot{y} , initially the barriers were non-conservative values of $\pm 4m/s$ and $\pm 2m/s$ respectively. As seen in Figure 6, there is perfect velocity trajectory tracking. The barriers are then restricted to $\pm 1.25m/s$ and $\pm 0.9m/s$ for \dot{x} and \dot{y} respectively. The quadrotor reduces its lateral velocities mid-flight in order to respect the barrier constraints.

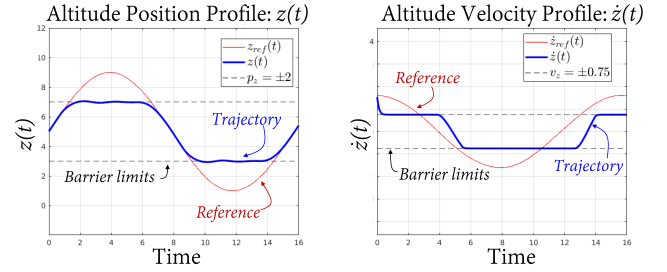


Fig. 4. (Left) Position barrier is enforced on state z with a limit of $\pm 2m$. (Right) Velocity barrier is placed on \dot{z} with $\pm 0.75m/s$. The rectified trajectory (blue) relaxes tracking the reference trajectory (red) to uphold safety limits (black dashed).

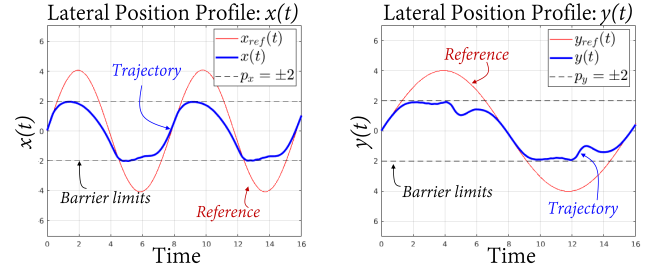


Fig. 5. Position barriers are placed on states x (top) and y (bottom) with limits ± 2.0 and $\pm 2.0m$ respectively. The actual trajectory (blue) compromises the reference trajectory (red) to uphold safety limits (dashed).

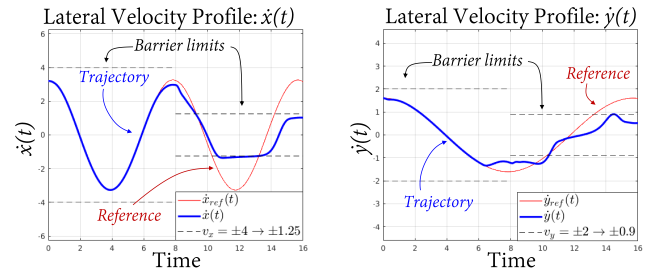


Fig. 6. Velocity barriers are enforced on states \dot{x} (top) and \dot{y} (bottom) with initial non-conservative limits of $\pm 4m/s$ and $\pm 2m/s$. Barrier limits change mid-flight to more conservative values modifying the controller inputs to respect safety constraints.

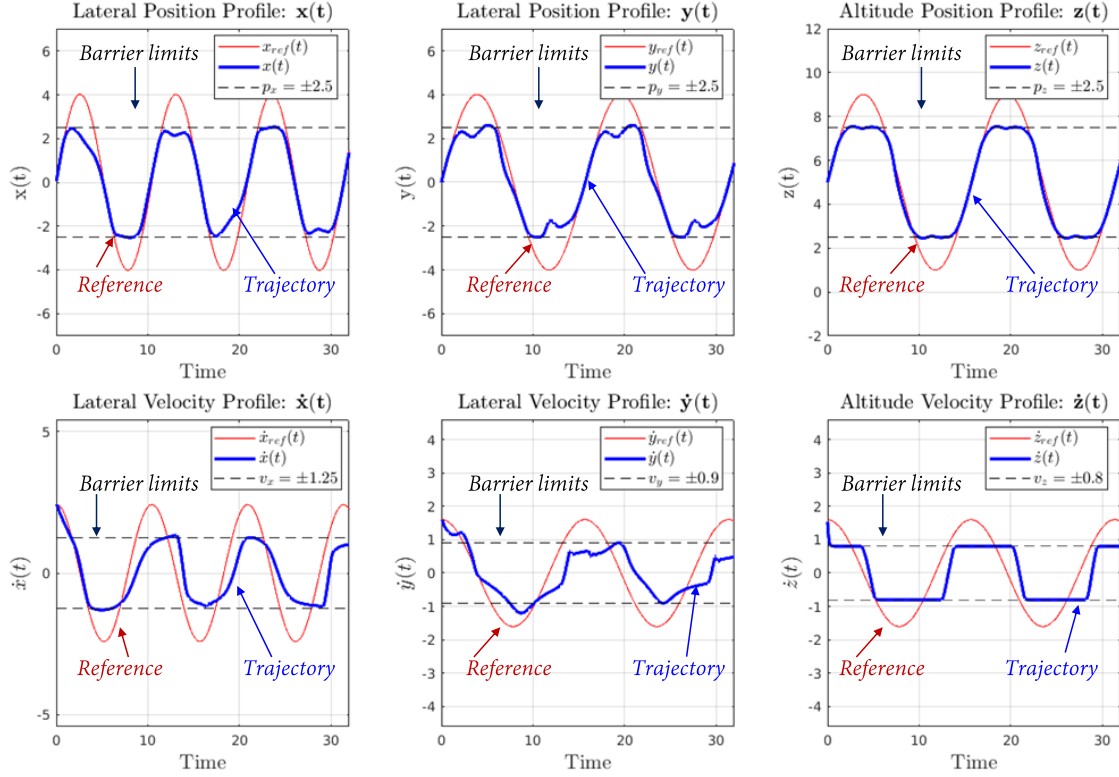


Fig. 7. Position barriers are placed on (x, y, z) (top) while velocity barriers are placed on $(\dot{x}, \dot{y}, \dot{z})$ (bottom). The actual trajectory (blue) is the modified flight behavior and the reference trajectory (red) tracking is compromised for respecting safe flight operation given by the barrier limits (black dashed).

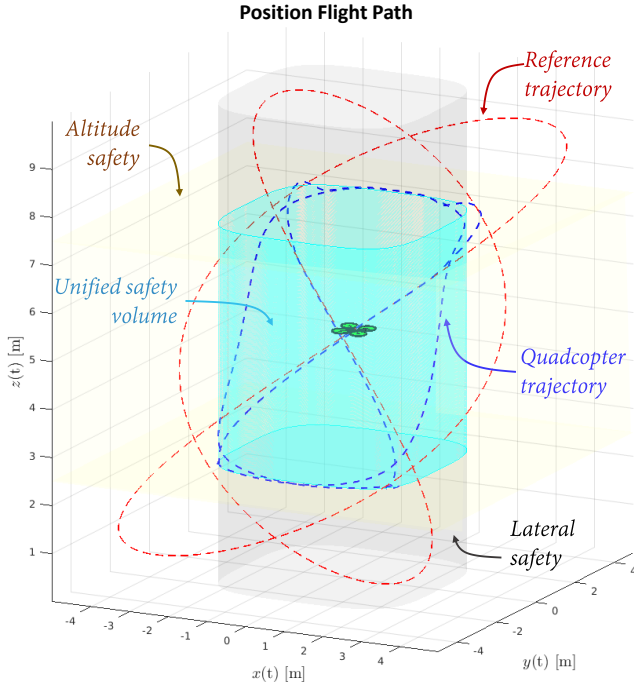


Fig. 8. The altitude (yellow) and lateral (gray) safety barrier functions jointly form the unified superellipsoid (cyan). The reference trajectory (dashed red) often violates the safety volume and goes outside. The cascaded QP controller regulating safety at both altitude and lateral domains constrains the quadrotor trajectory (dashed blue) inside the unified region.

C. Unified Safety Behavior

In this section, we demonstrate that by applying barriers across the hierarchy, the cascaded QP formulation does not compromise safety in $SE(3)$ and its tangent bundle. Safety is respected in a unified fashion for the quadrotor with each level meeting their safety objectives. The quadrotor is subjected to barrier constraints at both the high-level and low-level domains with safety barrier regions encoded in (16) and (17). By enforcing barriers at both levels, we regulate and ensure safety for the quadrotor's motion in $SE(3)$ domain.

The quadrotor is enforced with different limits for both position (x, y, z) and velocity $(\dot{x}, \dot{y}, \dot{z})$ states. Moreover, for testing the robustness of meeting the safety objectives at two different levels, quadrotor's initial \dot{x} and \dot{z} velocities are outside their respective safety regions. The trajectory rectification for the different states is illustrated in Figure 7. As seen in the figure, for each barrier-enforced state, the safety objectives are respected. Even if a particular state is outside the safety region, the constraints ensure the quadrotor asymptotically enter the safety region.

The flight path of the quadrotor is depicted in Figure 8. The safe set is the intersection of the two safe sets, namely,

altitude safety set and *lateral safety set*. The intersection of these two safe sets results in a richer superlevel safe set, *unified safety set*. The quadrotor's trajectory is constrained inside the *unified safe set* despite the reference trajectory going outside.

VI. CONCLUDING REMARKS

In this paper, we demonstrate the augmentation of (exponential) control barrier functions on a nonlinear cascaded control architecture for a quadrotor. We provide separate QP formulations in a cascaded architecture with the high-level safety objective regulating the altitude domain while the low-level safety objective regulating the lateral domain. Despite decoupling the objectives, safety is still preserved in a unified manner for the quadrotor navigation. We demonstrate the effectiveness of our strategy on position and velocity spaces for the quadrotor with both static and dynamic barrier limits.

Despite the effectiveness of our approach, we would like to add some closing remarks on the drawbacks we experienced. Depending on the nature of the barrier region and saturation constraints placed on thrust and moments, there is a possibility for infeasible solutions, thus rendering the QP-based cascaded controller ineffective. We have not found a way to counteract this issue yet. We believe this will be an interesting research direction to investigate further. In the future, we would also like to extend the notion by composing several safety barrier regions encapsulating an overall safe volume of space for the quadrotor to navigate.

VII. ACKNOWLEDGEMENTS

This work is supported in part by National Science Foundation under grant CISE:S&AS:1723998.

REFERENCES

[1] F. Berkenkamp, R. Moriconi, A. Schoellig, and A. Krause, "Safe learning of regions of attraction for uncertain, nonlinear systems with gaussian processes," pp. 4661–4666, 12 2016.

[2] R. Debouk, B. Czerny, and J. D'Ambrosio, "Safety strategy for autonomous systems," 08 2011.

[3] N. Hovakimyan, C. Cao, E. Kharisov, E. Xargay, and I. Gregory, "Adaptive control for safety-critical systems," *Control Systems, IEEE*, vol. 31, pp. 54 – 104, 11 2011.

[4] M. Valenti, D. Dale, J. How, and J. Vian, "Mission health management for 24/7 persistent surveillance operations," vol. 2, 08 2007.

[5] C. Zhang and J. M. Kovacs, "The application of small unmanned aerial systems for precision agriculture: a review," *Precision Agriculture*, vol. 13, pp. 693–712, Dec 2012.

[6] A. Schoellig, H. Siegel, F. Augugliaro, and R. D'Andrea, *So You Think You Can Dance? Rhythmic Flight Performances with Quadcopters*, pp. 73–105. 01 2014.

[7] 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, pp. 3861–3876, Aug 2017.

[8] S.-C. Hsu, X. Xu, and A. D. Ames, "Control barrier function based quadratic programs with application to bipedal robotic walking," in *2015 American Control Conference (ACC)*, pp. 4542–4548, IEEE, 2015.

[9] U. Borrmann, L. Wang, A. Ames, and M. Egerstedt, "Control barrier certificates for safe swarm behavior," *IFAC-PapersOnLine*, vol. 48, pp. 68–73, 12 2015.

[10] A. P. Schoellig, F. L. Mueller, and R. D'Andrea, "Optimization-based iterative learning for precise quadcopter trajectory tracking," *Autonomous Robots*, vol. 33, pp. 103–127, Aug 2012.

[11] L. Wang, A. Ames, and M. Egerstedt, "Safe certificate-based maneuvers for teams of quadrotors using differential flatness," 05 2017.

[12] L. Wang, E. A. Theodorou, and M. Egerstedt, "Safe learning of quadrotor dynamics using barrier certificates," *CoRR*, vol. abs/1710.05472, 2017.

[13] G. Wu and K. Sreenath, "Safety-critical control of a planar quadrotor," pp. 2252–2258, 07 2016.

[14] G. Wu and K. Sreenath, "Safety-critical control of a 3d quadrotor with range-limited sensing," p. V001T05A006, 10 2016.

[15] Q. Nguyen and K. Sreenath, "Exponential control barrier functions for enforcing high relative-degree safety-critical constraints," in *2016 American Control Conference (ACC)*, pp. 322–328, July 2016.

[16] R. Mahony, V. Kumar, and P. Corke, "Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor," *Robotics Automation Magazine, IEEE*, vol. 19, pp. 20 –32, 09 2012.

[17] A. P. Schoellig, C. Wiltscbe, and R. D'Andrea, "Feed-forward parameter identification for precise periodic quadcopter motions," in *Proc. of the American Control Conference (ACC)*, pp. 4313–4318, 2012.

[18] D. Zhou and M. Schwager, "Vector field following for quadrotors using differential flatness," pp. 6567–6572, 05 2014.