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 allows independent safety regulation in 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 position and velocity spaces simultaneously.

I. INTRODUCTION

Safety is a critical component for today’s aerial autonomous systems [1], [2], [3]. Of particular interest among aerial autonomous systems are quadrotors due to their application in surveillance, agriculture, acrobatic performances, and search and rescue, see [4], [5], [6]. Thus, accentuating the need for safety being an imperative component during flight operation. Moreover, given recent advances in design, control, planning, and sensing, quadrotors have gained wide interest. The focus of this paper is to rectify the nominal flight trajectory for a quadrotor using a cascaded controller in a minimally invasive manner to ensure safety in position and velocity spaces. We achieve this by independently imposing barriers in the altitude and lateral domains of the quadrotor using Control Barrier Functions (CBF).

The underactuated and intrinsically unstable nature of quadrotor makes it challenging to generate safe trajectories [7]. Constrained optimization based approaches such as Model Prediction Control [8] are formulated as finite-horizon problems. However, MPC is limited in its real-time scalability to more complex systems. Although, real time MPC on quadrotors has been demonstrated in [9], velocity constrained safety has not been addressed. CBFs [10], first used in adaptive cruise control, permit dynamically feasible constraints and ensure forward invariance. CBFs were used in collision avoidance for swarm of mobile robots [13] and quadrotors [14]. CBFs were also used to learn quadrotor dynamics in presence of wind disturbances [16]. The works in [11] and [12] uses a sequential-QP based approach augmented with CBFs for obstacle avoidance. The prior work used CBFs in quadrotor controllers designed using differential flatness [15] or Control Lyapunov Functions (CLF) [11], [12]. We differ from the aforementioned endeavors by imposing barriers in a cascaded control architecture in a minimally invasive approach. Prior work has not merged the forward invariance of CBFs with a nonlinear cascaded controller for quadrotors to ensure safety. Unlike CLF or differentially flat based controllers [15], cascaded controllers use PID regulators within nested loops operating at different frequencies, thereby reducing the need for a model-based controller. Moreover, [11], [12], [14] only imposed safety for position spaces, while we impose safety limits explicitly on both position and velocity spaces.

In summary, our key contributions in this paper are three-fold. First, barrier functions are employed on a cascaded controller in a minimally invasive way with constraints explicitly imposed on position and velocity spaces. Second, safety constraints are handled in the altitude and lateral domains of the quadrotor independently. Third, we present derivations for enforcing constraints across the hierarchy by considering the complete 3D underactuated dynamics of the quadrotor evolving in $TSE(3)$.

The rest of the paper is organized as follows. Section II introduces preliminaries on quadrotor dynamics and barrier functions. The cascaded controller with cascaded QP design is presented in Section III. Safety barrier formulations are shown in Section IV. Simulation results are provided in Section V, followed by conclusion in Section VI.
II. Preliminaries of Quadrotor Dynamics and Barrier Functions

This section introduces the dynamics of a quadrotor in 3D and barrier functions along with its invariance property. Due to dynamical nature of the quadrotor with high relative degree, an extended version of CBFs are used called Extended Control Barrier Functions (ECBF). For a more detailed discussion on quadrotor dynamics, CBFs, and ECBFs, we refer the reader to [17], [10], and [18] respectively.

A. Dynamics of 3D Quadrotor

Quadrotor is a dynamical system whose motion is described in the Lie Group $SE(3)$. Hence, it is described with six degrees of freedom: translational position $(x, y, z)$ in the inertial frame $W$ and attitude represented by Euler angles (roll $\phi$, pitch $\theta$, and yaw $\psi$) in the intermediate frames after yaw rotation with respect to the body-fixed frame $B$ [17]. A pictorial representation is illustrated in Figure 1.

The translational acceleration of the quadrotor depends on its attitude along the body frame’s $z_B$ axis and overall thrust produced by the four propellers [6]. In inertial frame $W$, this acceleration is given by,

$$\begin{bmatrix}
\ddot{x} \\
\ddot{y} \\
\ddot{z}
\end{bmatrix} = 0 - \mathbf{R} \begin{bmatrix}
0 \\
0 \\
f(t) \\
m
\end{bmatrix} \Leftrightarrow \ddot{r} = g z_w - \mathbf{R} z_w \frac{f(t)}{m} \quad (1)
$$

where $z_w = [0, 0, 1]^T$, $r = [x, y, z]^T$ is the position of center of mass of quadrotor in $W$, $m$ is its mass, $g$ is gravitational acceleration, and $f(t)$ is the total thrust produced by the four propellers. $\mathbf{R}$ is the rotation matrix from body frame $B$ to the inertial frame $W$ given by,

$$\mathbf{R} = \begin{bmatrix}
\csc^2 \theta \csc^2 \psi - \csc^2 \phi & \csc^2 \theta \csc \phi - \csc \theta \csc \psi & \csc^2 \phi \\
\csc^2 \theta \csc \phi - \csc \phi \csc \theta & \csc^2 \phi - \csc \theta \csc \phi & \csc \theta \\
-\csc \theta & s \csc \phi & c \csc \phi
\end{bmatrix}^T \quad (2)
$$

where $s$ and $c$ stand for $\sin$ and $\cos$ respectively. The evolution of the rotation matrix $\mathbf{R}$ is given by,

$$\dot{\mathbf{R}}(t) = \mathbf{R}(t) \begin{bmatrix}
0 & -r(t) & q(t) \\
r(t) & 0 & -p(t) \\
-q(t) & p(t) & 0
\end{bmatrix} = \mathbf{R}(t)[\Omega(t)] \times, \quad (3)
$$

where $\times$ is the overloaded operator for skew-symmetric representation of the angular velocity $\Omega = [p, q, r]^T$.

In the body frame, the angular acceleration of the body velocities is calculated using the following equation [17],

$$\begin{bmatrix}
\dot{p} \\
\dot{q} \\
\dot{r}
\end{bmatrix} = \begin{bmatrix}
\tau_x \\
\tau_y \\
\tau_z
\end{bmatrix} - \begin{bmatrix}
0 \\
p \\
r
\end{bmatrix} \times \begin{bmatrix}
p \\
q \\
r
\end{bmatrix} \Leftrightarrow \dot{\Omega} = \tau - \Omega \times \Omega, \quad (4)
$$

where $\mathbf{I}$ is the inertia matrix of the quadrotor vehicle, $\tau = [\tau_x, \tau_y, \tau_z]^T$ are the moments along each principal axis.

For roll, pitch, and yaw angles, their derivatives can be computed from quadrotor’s angular velocities $\Omega$ by [19],

$$\begin{bmatrix}
\dot{\phi} \\
\dot{\theta} \\
\dot{\psi}
\end{bmatrix} = \begin{bmatrix}
1 & \sin \theta \sin \psi & \cos \theta \\
0 & \cos \theta & -\sin \theta \\
0 & \sin \psi \cos \theta & \cos \psi
\end{bmatrix} \begin{bmatrix}
p \\
q \\
r
\end{bmatrix},
$$

where $t$ and $sc$ stand for $\tan$ and $secant$ respectively.

The actuator dynamics relates rotor rotational speeds with the desired thrust and moments. Each rotor produces a thrust in the positive $z_B$ direction, $F_i = k_i \omega_i^2$, where $k_i$ represents rotor thrust constant (see [20]), $\omega_i$ is the rotor $i$’s rotational speed, and $i \in I = \{1, 2, 3, 4\}$. A reaction torque is also produced by the rotors given by $M_i = k_i \omega_i^2$, and $M_j = -k_w \omega_j^2$, where $k_w$ is rotor torque constant, $i \in \{1, 3\}$, and $j \in \{2, 4\}$. The net thrust is given by $F = \sum_{i \in I} F_i$, and torque moments are given by $\tau_x = (F_1 - F_2 + F_3 - F_4)l$, $\tau_y = (F_1 + F_2 - F_3 - F_4)l$, $\tau_z = M_1 + M_2 + M_3 + M_4$.

The quadrotor system is control affine with its full state as $X = [x, y, z, \phi, \theta, \psi, \dot{x}, \dot{y}, \dot{z}, \dot{p}, \dot{q}, \dot{r}]^T$ and control input $u = [f, \tau_x, \tau_y, \tau_z]^T$.

B. Exponential Control Barrier Functions

Consider a general control affine dynamical system,

$$\dot{x} = f(x) + g(x)u, \quad x(t_0) = x_0, \quad (5)$$

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 $S$ of a smooth function $h : \mathcal{X} \rightarrow \mathbb{R}$ as follows,

$$S = \{ x \in \mathbb{R}^n \mid h(x) \geq 0 \}. \quad (6)$$

**Definition 1 [10]:** The function $h(x) : \mathcal{X} \rightarrow \mathbb{R}$ is defined as a control barrier function (CBF), if $\exists$ an extended class-$\kappa$ function ($\kappa(0) = 0$ and strictly increasing) such that $\forall x \in S,$

$$\sup_{u \in \mathcal{U}} \{ L_f h(x) + L_g h(x)u + \kappa(h(x)) \} \geq 0. \quad (7)$$

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

**Theorem [10]:** Given a system defined by (5), safe set $S \subset \mathbb{R}^n$ defined by (6), and smooth CBF $h(x) : S \rightarrow \mathbb{R}$ defined in (7), $\forall$ Lipschitz continuous feedback control $u \in \mathcal{U}$ that satisfies, $\mathcal{U} = \{ u \in \mathcal{U} \mid L_f h(x) + L_g h(x) + \kappa(h(x)) \geq 0 \}, \forall x \in \mathcal{X},$ then the safe set $S$ is forward invariant for the system.

CBFs are limited in their nature to systems with relative degree ($\delta$) one, i.e., $\delta = 1,$ where $\delta \in \mathbb{N}$ [18]. Depending on how one enforces barrier around quadrotor’s state(s), the $\delta$ can go above 1. Thus, CBFs cannot be directly applied for such barrier constraints. For $\delta > 1,$ an extension of the CBF is used to guarantee forward invariance property of $S$ called the Exponential Control Barrier Functions (ECBF) [18].

**Definition 2 [18]:** The smooth function $h(x) : \mathcal{X} \rightarrow \mathbb{R},$ with relative degree $\delta,$ is defined as an exponential control barrier function (ECBF), if $\exists$ $K \in \mathbb{R}^\delta$ such that $\forall x \in S,$

$$\sup_{u \in \mathcal{U}} \{ L_f^\delta h(x) + L_g^\delta h(x)u + K^T \mathcal{H} \} \geq 0, \quad (8)$$

where $\mathcal{H} = [h(x), L_f h(x), L_f^2 h(x), ..., L_f^{(\delta - 1)} h(x)]^T$ is the vector of Lie derivatives for $h(x),$ and $K = [k_0, k_1, ..., k_{\delta - 1}]$ is vector of coefficient gains for $\mathcal{H}.$ The coefficient gain vector $K$ can be determined using linear control theory’s pole placement technique on the closed-loop matrix ($F - GK$) determined from $h(x) \geq Ce^{F - GK}h(x_0) \geq 0$ when $h(x_0) \geq 0,$ $C = [1, 0, 0, ..., 0]^T \in \mathbb{R}^\delta$ [18]. Akin to the forward invariance of CBFs, forward invariance is satisfied for ECBFs and we refer the reader to [18] for detailed proofs.

### III. Nonlinear Cascaded Control Architecture

In this section, we motivate our choice for the control architecture of the quadrotor called the nonlinear cascaded controller [17]. We then discuss details of controller design and barrier-enforced QP modification to the controller.

---

**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 attitude controller generates desired thrust. The attitude loop orientations 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.

**A. Motivation**

While there are many different controllers for a quadrotor [15], [16], [12], the cascaded controller is a popular control architecture demonstrated with practical feasibility and satisfactory performance [6], [7], [17], [21]. The architectural design is intuitive in its application and is commonly used in academic settings for students, developers, and/or hobbyists. Moreover, each sub-controller in this architecture uses a PID regulator thereby eliminating the need for a strict model based control. Prior to this work, augmentation of ECBFs on such a control framework has not been investigated.

**B. Controller Design**

The cascaded terminology is due to the hierarchical approach taken while designing the controllers. At the highest level of the hierarchy is the position controller, which is further separated into altitude and lateral position controllers. The next level controls the quadrotor’s attitude or orientation. At the lowest level of the design, with the highest bandwidth, is the motor controller responsible for converting commanded angular velocities to rotor rotational speeds [17]. These nested loops form a cascaded architecture and is shown in Figure 2. We make the following assumptions as inputs for our controller design:

- A smooth reference trajectory is given: $r_d(t),$ where $r = [x, y, z]^T.$
- A yaw reference trajectory is given: $\psi_d(t)$

Our controller framework is modeled after [21]. The position controller’s commanded accelerations are computed like a second-order system. The commanded accelerations are computed as:

$$\ddot{r}_{cmd}(t) = \ddot{r}_d(t) + K_p e_r(t) + K_d \dot{e}_r(t),$$

where $e_r = r - r_d.$
\[ e_r(t) = r(t) - \hat{r}_d(t), \quad \dot{e}_r(t) = \dot{r}(t) - \dot{r}_d(t), \quad K_p \text{ and } K_d \text{ are positive definite proportional and derivative gain matrices, and } r = [x, y, z]^T. \]

Using (1) and altitude commanded acceleration \( \ddot{z}_{cmd} \), we get the control input thrust,

\[ f(t) = \frac{m}{R_{33}}(g - \ddot{z}_{cmd}), \quad (9) \]

where \( R_{33} \) is rotational matrix entry. Using the lateral commanded accelerations, \( \ddot{x}_{cmd}(t), \ddot{y}_{cmd}(t) \), (1), and (9), commended rotational entries \( R_{cmd}^{13} \) and \( R_{cmd}^{23} \) are determined. Commanded rotational rates are then calculated using a simple proportional regulator resulting in \( \dot{R}_{cmd}^{13} \) and \( \dot{R}_{cmd}^{23} \). These commanded rotational rates in turn are used for calculating commanded angular velocities using current estimated attitude (2) and (3),

\[
\begin{bmatrix}
\dot{p}_{cmd} \\
\dot{q}_{cmd}
\end{bmatrix} = \frac{1}{R_{33}} \begin{bmatrix}
R_{21} & -R_{11} \\
R_{22} & -R_{12}
\end{bmatrix} \begin{bmatrix}
\dot{R}_{cmd}^{13} \\
\dot{R}_{cmd}^{23}
\end{bmatrix} \quad (10)
\]

The yaw controller can be separated from the roll-pitch controller since rotations around the quadrotor’s \( z_B \) axis does not affect the dynamics for determining roll and pitch. A proportional regulator is used for determining the commanded angular velocity \( r_{cmd} \) along \( z_B \); \( r_{cmd}(t) = k_p^\psi(\psi(t) - \psi_d(t)) \).

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. These are then used to compute the remaining control inputs, the torque moments \( \tau_x, \tau_y, \tau_z \), using (4).

C. Barrier-enforced QP Cascaded Controller

Given the nominal controller \( \hat{u} \) developed above, safety barriers are enforced across the cascaded architecture. Separate QP formulations are designed for altitude and lateral domains. This “modifies” the nominal control ensuring the system is always safe. The QP at each level (altitude and lateral) is constructed independently as follows:

\[
u^* = \arg\min_{u \in U} \mathcal{P}(u) = \frac{1}{2} ||u - \hat{u}||^2 \quad (11)
\]

s.t. \( \left\{ L_f^r h(x) + L_g L_f^{r-1} h(x)u + K^T \mathcal{H} \right\} \geq 0. \)

Hence, the modified control \( u^* \) tries to follow the nominal control \( \hat{u} \) as close as possible except when it comes to ensuring safety requirements at the expense of not strictly tracking the reference trajectory.

Motivated by the cascaded controller architecture, the formulation is thus a QP controller that is decoupled in its safety objectives. One layer of safety is enforced at the high level for altitude domain modifying control input \( f \) inside altitude controller. The second layer of safety is enforced at the lower level for the lateral domain modifying control inputs \( \tau_x, \tau_y \) inside body-rate controller (see Figure 2). This is unlike the works in [11], [12] which employ a sequential-QP based design, with the position level QP solving for a virtual tracking force and orientation level QP solving for the four control inputs. We decouple the QP objectives for \( f \) at the high level and \( \tau_x, \tau_y \) at the lower level. With a lot of quadrotors already employed with cascaded controllers, with minimal modification across the architecture, they can achieve safety through our method. This is in contrast to [11], [12] which designs a CLF-CBF-QP controller and replacing the existing controller can be expensive. Moreover, we enforce barriers explicitly pn position and velocity spaces, unlike [11], [12] which only dealt with position space.

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 safe region due to CBF constraints. In our work, the safety barrier region is modeled as,

\[ h(x_1, ..., x_n) = 1 - \left[ \frac{x_1 - c_1}{p_1} \right]^r + ... + \left[ \frac{x_n - c_n}{p_n} \right]^r \geq 0, \quad (12) \]

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. In this work, we choose \( r = 4 \), which is called rectellipse, as it allows more freedom in the safe region. Inspired by the work in [16], where ellipsoidal safe regions \( (r = 2) \) were used to learn quadrotor dynamics using CBFs in 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.
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 $\tau_x$ and $\tau_y$. It involves the effect of roll and pitch to induce this lateral motion. We present the derivation required in order to derive constraints for the low-level QP-based controller.

**Derivation:** Recall (10), where angular velocities $p$ and $q$ are related to rotational rates,

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

For convenience, we define $W$ as the $2 \times 2$ matrix of rotational entries and $A \triangleq \begin{bmatrix} p & q \end{bmatrix}^\top$. Rewriting in terms of angular velocities gives,

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

Now, computing the time derivative for (16) results in,

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

Since angular accelerations $\ddot{p}$ and $\ddot{q}$ are related to inputs $\tau_x$ and $\tau_y$ given in (4), substituting it back in (17) gives,

$$
\begin{bmatrix}
\ddot{R}_{13} \\
\ddot{R}_{23}
\end{bmatrix} = \dot{R}_{33} \dot{V} A + R_{33} \dot{\dot{V}} A + R_{33} V \left[ \begin{array}{c}
I_x^{-1} - I_z q r + \frac{\tau_x}{\tau_x} \\
I_y^{-1} - I_z p r + \frac{\tau_y}{\tau_y}
\end{array} \right]
$$

$$
= R_{33} V A + R_{33} \dot{\dot{V}} A + R_{33} V \left[ \begin{array}{c}
I_x^{-1} - I_z q r \\
I_y^{-1} - I_z p r
\end{array} \right]
$$

$$
+ R_{33} V \left[ \begin{array}{ccc}
I_x^{-1} & 0 \\
0 & I_y^{-1}
\end{array} \right] \begin{bmatrix}
\tau_x \\
\tau_y
\end{bmatrix}
$$

$$
= \mathcal{J} + \mathcal{L} \begin{bmatrix}
\tau_x \\
\tau_y
\end{bmatrix}
$$

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 (1), we need the fourth time derivative of $x$ and $y$ in order to get $\dddot{R}_{13}$ and $\dddot{R}_{23}$, thus finally relating with $\tau_x$ and $\tau_y$.

$$
\begin{bmatrix}
\dddot{x} \\
\dddot{y}
\end{bmatrix} = \frac{f}{m} \begin{bmatrix}
R_{13} \\
R_{23}
\end{bmatrix} \quad \text{[using (1)]}
$$

$$
\begin{bmatrix}
\dddot{x} \\
\dddot{y}
\end{bmatrix} = \frac{f}{m} R_{33} V A \quad \text{[using (16)]}
$$

$$
\begin{bmatrix}
\dddot{x} \\
\dddot{y}
\end{bmatrix} = \frac{f}{m} \mathcal{J} - \frac{f}{m} \mathcal{L} \begin{bmatrix}
\tau_x \\
\tau_y
\end{bmatrix} \quad \text{[using (18)]}
$$
Thus, time derivatives of $x$ and $y$ relate to control inputs $\tau_x$ and $\tau_y$ with relative degree $\delta = 4$. We next compute the Lie derivatives for lateral safety barrier region (15),

$$
\begin{align*}
&\diamond L_f h(x, y) = -4\eta_3 \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} \\
&\diamond L_f^2 h(x, y) = -4\eta_3 \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} - 12\eta_2 \begin{bmatrix} \dot{x}^2 \\ \dot{y}^2 \end{bmatrix} \\
&\diamond L_f^3 h(x, y) = -4\eta_3 \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} - 36\eta_2 \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} \\
&\phantom{=} - 24\eta_1 \begin{bmatrix} \dot{x}^3 \\ \dot{y}^3 \end{bmatrix} \\
&\diamond L_f L_f^3 h(x, y) = \frac{4f}{m} \eta_3 L
\end{align*}
$$

where (18) was used to substitute for $[\dot{x}, \dot{y}]^T$, and $\eta_i = [(x - c_x)/\mu_x^i, (y - c_y)/\mu_y^i]^T$, $i \in \{0, 1, 2, 3\}$. Due to the relative degree being four, ECBFs are once again employed to satisfy the QP constraints in (11). For the velocity space of the lateral motion, the following barrier function is used,

$$
\begin{align*}
h(\dot{x}, \dot{y}) &= 1 - \left[\frac{\dot{x}}{v_x}\right]^4 - \left[\frac{\dot{y}}{v_y}\right]^4 \\
\geq 0. \quad (22)
\end{align*}
$$

V. SIMULATION RESULTS

With the QP cascaded controller developed in Section III-C and different barrier regions constructed at both layers of the architecture in Section IV, we present our simulation results. The simulation was done in MATLAB 2018b with parameters as tabulated in I to model the quadrotor.

In order to ensure safe navigation for the quadrotor, position and velocity state constraints play an important role. To this end, we have constructed safety barrier regions for both the spaces. Through our cascaded QP formulation, we are independently able to satisfy safety constraints at the high-level altitude domain and the low-level lateral domain.

A. High-Level Altitude Domain Safety Behavior

Through the safety barrier regions developed for high-level altitude domain in Section IV-B, we enforce constraints on altitude position and altitude velocity. With limits of $\pm 2m$ for position and $\pm 0.55m/s$ for velocity, as shown in Figure 4, trajectory tracking is relaxed for upholding the limits enforced by the barrier regions.

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

B. Low-Level Lateral Domain Safety Behavior

While the high-level objective is responsible for regulating safety for the altitude domain, the low-level safety objective regulates it in the lateral domain. Our cascaded formulation allows easy regulation of quadrotor motion in the lateral domain independent of the high-level constraint objectives.

We test our method on both the position and velocity

| 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 motor thrust     | 0.00 kg m/s$^2$ |
| $f_{max}$ | Maximum motor thrust     | 36.00 kg m/s$^2$ |
| $\tau_{min}^x$, $\tau_{min}^y$, $\tau_{max}^x$, $\tau_{max}^y$ | Min. moment about $x_B$, $y_B$-axis | -20.0 Nm |
|           | Max. moment about $x_B$, $y_B$-axis | 20.0 Nm |

| TABLE I |
| Parameters for modeling the dynamics of the quadrotor. |
Fig. 5. Position barriers are placed on states $x$ (top) and $y$ (bottom) with limits $\pm 3.0$ and $\pm 2.5m$ 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 $\pm 3m/s$ 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.

C. Unified Safety Behavior

In this experimental section we demonstrate that the QP formulation in a cascaded architecture does not compromise safety in $SE(3)$. Safety is respected in a unified fashion for the quadrotor with each level independently 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 (14) and (15). 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.

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 Quadrocopters, pp. 73–105. 01 2014.
[7] 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.
[8] D. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, “Constrained model predictive control: Stability and optimality,” Automatica, vol. 36, pp. 789–814, 06 2000.
[9] M. Bangura, “Real-time model predictive control for quadrotors,” pp. 11773–11780, 08 2014.
[10] 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.
[11] G. Wu and K. Sreenath, “Safety-critical control of a planar quadrotor,” pp. 2252–2258, 07 2016.
[12] G. Wu and K. Sreenath, “Safety-critical control of a 3d quadrotor with range-limited sensing,” p. V001T05A006, 10 2016.
[13] 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.
[14] L. Wang, A. Ames, and M. Egerstedt, “Safe certificate-based maneuvers for teams of quadrotors using differential flatness,” 05 2017.
[15] D. Zhou and M. Schwager, “Vector field following for quadrotors using differential flatness,” pp. 6567–6572, 05 2014.
[16] L. Wang, E. A. Theodorou, and M. Egerstedt, “Safe learning of quadrotor dynamics using barrier certificates,” CoRR, vol. abs/1710.05472, 2017.
[17] 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.
[18] 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.
[19] D. Mellinger and V. Kumar, “Minimum snap trajectory generation and control for quadrotors,” in 2011 IEEE International Conference on Robotics and Automation, pp. 2520–2525, May 2011.
[20] J. Svacha, K. Mohta, and V. Kumar, “Improving quadrotor trajectory tracking by compensating for aerodynamic effects,” in Unmanned Aircraft Systems (ICUAS), 2017 International Conference on, pp. 860–866, IEEE, 2017.
[21] A. P. Schoellig, C. Wiltse, and R. D’Andrea, “Feed-forward parameter identification for precise periodic quadrocopter motions,” in Proc. of the American Control Conference (ACC), pp. 4313–4318, 2012.