

Safe Learning-based Tracking Control for Quadrotors under Wind Disturbances

Lei Zheng, Rui Yang, Jiesan Pan, and Hui Cheng*

Abstract—Enforcing safety on precise trajectory tracking is critical for aerial robotics subject to wind disturbances. In this paper, we present a learning-based safety-preserving cascaded quadratic programming control (SPQC) for safe trajectory tracking under wind disturbances. The SPQC controller consists of a position-level controller and an attitude-level controller. Gaussian Processes (GPs) are utilized to estimate the uncertainties caused by wind disturbances, and then a nominal Lyapunov-based cascaded quadratic program (QP) controller is designed to track the reference trajectory. To avoid unexpected obstacles when tracking, safety constraints represented by control barrier functions (CBFs) are enforced on each nominal QP controller in a way of minimal modification. The performance of the proposed SPQC controller is illustrated through numerical validations of (a) trajectory tracking under different wind disturbances, and (b) trajectory tracking in a cluttered environment with a dense time-varying obstacle field under wind disturbances.

I. INTRODUCTION

Safe trajectory tracking is an essential requirement for autonomous aerial vehicles, where an aerial vehicle is required to track a reference trajectory and avoid unexpected obstacles. To achieve high-accuracy tracking and avoid obstacles, the typical approach utilizes a high-level planner to generate a safe trajectory, and a low-level controller running at a much higher frequency than the planner to track the planned trajectory [1].

In practice, for high-speed quadrotor tracking in a clutter scenario with dense obstacles and limited sensing range, it is challenging for a planner to replan a safe trajectory in real-time. On the other hand, it is difficult to obtain an accurate model for the low-level tracking controller due to wind disturbances, which may cause the safety-critical quadrotor system to deviate from the planned trajectory and even collide with obstacles. These challenges show an urgent need for designing a safe tracking controller that ensures both obstacle avoidance and accurate trajectory tracking for a quadrotor with a limited sensing range and under uncertain wind disturbances.

In recent years, for safe trajectory tracking, the cascaded architecture [2], [3] using control barrier functions (CBFs) to ensure safety has shown great potential for addressing these challenges with practical feasibility. CBFs [4] have been widely used to enforce safety constraints, and have proved to be an effective strategy to avoid obstacles for

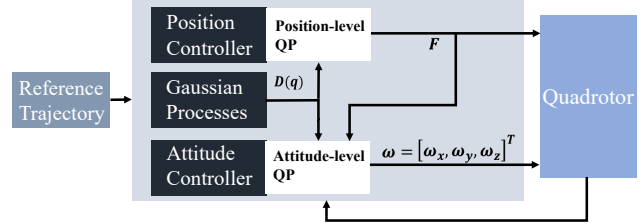


Fig. 1. The cascaded controller consists of a position-level QP and an attitude-level QP. Reference inputs are provided to position and attitude controllers. The GPs estimate the wind disturbances and generate a high confidence interval bound $D(q)$. With the $D(q)$, the position controller generates the desired thrust F , and the attitude controller generates the desired body rotational rates ω .

safety-critical systems. In [5], a learning-based CBF is presented to constrain a quadrotor within a static ellipsoid safe region under wind disturbances. In [3], a cascaded quadratic program (QP) safety-critical controller incorporating CBFs and control Lyapunov functions (CLFs) is proposed. In this work, the quadrotor with a limited sensing range can asymptotically track a reference trajectory by constructing stability constraints (represented by CLFs), and simultaneously avoid obstacles by constructing safety constraints (represented by CBFs). Although this controller can handle time-varying constraints, a precise model of the system is required to enforce the constraints.

Alternatively, Model Predictive Control (MPC) [6] is an effective finite-horizon optimal control method to handle constraints naturally in a safe tracking control problem. In [7], a Nonlinear MPC is proposed to track the desired trajectory while avoiding obstacles. While this controller can mediate the trade-off between safety and tracking, disturbances are not considered in its model. [8] proposes a Nonlinear MPC for safe trajectory tracking with obstacle avoidance capacity under constant disturbances. Although this method can efficiently deal with obstacles when tracking a trajectory, the considered obstacles are assumed to be static.

Considering the limitations of current approaches, a safe tracking controller is desirable to be developed for a quadrotor subject to a limited sensing range to ensure accurate trajectory tracking and enforce safety constraints under wind disturbances. Particularly, it should drive the quadrotor back to the predefined reference trajectory after safely avoiding unexpected static and dynamic obstacles.

In this paper, drawing inspiration from the cascaded architecture [3], we propose a learning-based safety-preserving cascaded QP control (SPQC) approach composed of a position-level controller and an attitude-level controller, as illustrated in Fig. 1. We utilize the Gaussian Process (GP)

L. Zheng is with the School of Electronics and Information Technology, Sun Yat-sen University, Guangzhou 510006, China.

R. Yang, J. Pan, H. Cheng are with the School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou 510006, China.

*Corresponding author: chengh9@mail.sysu.edu.cn

to estimate a high confidence interval $D(q)$ accounting for the model uncertainties caused by wind disturbances. Then, a local motion planning (LMP) method is designed based on the $D(q)$ to generate a reference orientation for the attitude controller. In each QP controller, a stability constraint based on CLF is formulated to develop a nominal Lyapunov-based cascaded tracking controller for accurate tracking. Considering safety, the Iterative Regional Inflation by Semidefinite programming (IRIS) algorithm [9] is adopted to construct a time-varying obstacle-free safe region and then use this safe region to design two CBFs to enforce safety constraints both on the position level and the attitude level for the nominal cascaded tracking controller. Finally, the safety constraints are enforced to minimally modify the Lyapunov-based tracking controller to formulate the SPQC controller.

The **main contributions** of this paper are presented as follows:

- A learning-based SPQC controller consists of a position-level and an attitude-level controller is presented for quadrotors to safely track the reference trajectory with obstacle avoidance capacity under wind disturbances. An LMP algorithm is designed for the attitude-level controller for accurate tracking under wind disturbances.
- Two CBFs are designed to enforce safety constraints on the position and attitude level for the quadrotor to minimally modify the controls generated by the nominal tracking controller via constrained QPs.
- The proposed SPQC controller is validated through numerical simulations on a quadrotor in a cluttered environment under varying wind disturbances.

This paper is organized as follows: Section II presents the preliminaries used in this paper. Section III illustrates the proposed SPQC controller. Section IV shows simulation results to validate and clarify our SPQC controller. Finally, conclusions are drawn in Section V.

II. PRELIMINARIES

In this section, we briefly provide a review of CBFs [4] and quadrotor dynamics to formulate the safe learning-based tracking control problem for Quadrotors.

A. Control Barrier Function

Consider a control affine system

$$\dot{x} = f(x) + G(x)u, \quad (1)$$

where $x \in \mathcal{X} \subseteq \mathbb{R}^n$, $u \in \mathcal{U} \subseteq \mathbb{R}^m$ denote the state and the control input of the system, respectively. The function $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ and $G : \mathbb{R}^n \rightarrow \mathbb{R}^{n \times m}$ are Lipschitz continuous.

The *safety set* \mathcal{S} of the system can be defined by

$$\mathcal{S} := \{x \in \mathcal{X} | h(x) \geq 0\}, \quad (2)$$

where $h : \mathbb{R}^n \rightarrow \mathbb{R}$ is a continuously differentiable function related to state constraints.

Definition 2. The set \mathcal{S} is called *forward invariant*, if for every $x_0 \in \mathcal{S}$, $x(t, x_0) \in \mathcal{S}$ for all $t \in \mathbb{R}_0^+$.

To ensure forward invariance of \mathcal{S} , e.g. quadrotors stay in the collision-free safety set at all times, we consider the following definition.

Definition 3 (Definition 5 of [4]). For the dynamical system (1), given a set $\mathcal{S} \subset \mathbb{R}^n$ defined by (2) for a continuously differentiable function $h : \mathbb{R}^n \rightarrow \mathbb{R}$, the function h is called a *Zeroing Control Barrier Function (ZCBF)* defined on the set \mathcal{D} with $\mathcal{S} \subseteq \mathcal{D} \subset \mathbb{R}^n$, if there exists an extended class \mathcal{K} function ($\kappa(0) = 0$ and strictly increasing) α such that

$$\sup_{u \in \mathcal{U}} [L_f h(x) + L_g h(x)u + \alpha(h(x))] \geq 0, \forall x \in \mathcal{D}, \quad (3)$$

where L represents the Lie derivatives. To be more specific:

$$L_f h(x) = \frac{\partial h(x)}{\partial x} f(x), \quad L_g h(x) = \frac{\partial h(x)}{\partial x} g(x). \quad (4)$$

ZCBF is a special control barrier function that comes with asymptotic stability [10]. The existence of a ZCBF implies the asymptotic stability and forward invariance of \mathcal{S} as proved in [10].

B. Dynamics of Quadrotor

The motion of the quadrotor can be described by six degrees of freedom: The translational position (x, y, z) in the inertial coordinate system \mathcal{O} and attitude represented by Euler angles (roll ϕ , pitch θ , and yaw ψ). The vehicle attitude is defined by the rotation matrix $R \in SO(3)$ [8] from the vehicle coordinate system \mathcal{V} to the inertial coordinate system \mathcal{O} .

Considering the unknown wind force d_w caused by wind disturbances [11], we adopt the following model to describe the nonlinear quadrotor dynamics,

$$\dot{p} = v, \quad (5)$$

$$m\dot{v} = mge_3 + Re_3F + d_w, \quad (6)$$

$$\dot{R} = RS(\omega), \quad (7)$$

where $e_3 = [0, 0, 1]^T$ is the unit vector. $m > 0$ and g denote the mass and the gravity acceleration, respectively. $p = [x, y, z]^T$ and $v = [v_x, v_y, v_z]^T$ denotes the translational position and velocity in \mathcal{O} , respectively. F denotes the total thrust generated from the four rotors in \mathcal{V} , and $S(\cdot)$ is skew-symmetric mapping. The control inputs of the quadrotor are denoted by the body rotational rates $\omega = [\omega_x, \omega_y, \omega_z]^T$ and the thrust force $F \in \mathbb{R}$ along the body axis in the vehicle coordinate system \mathcal{V} . $d_w = [W_x, W_y, W_z]^T$ is unknown wind force acting on the quadrotor.

In this paper, we consider the following state-space nonlinear control affine quadrotor system with partially unknown dynamics, i.e.,

$$\dot{q} = f(q) + G(q)u + d(q), \quad (8)$$

where $q = [x, y, z, \dot{x}, \dot{y}, \dot{z}, \phi, \theta, \psi]^T$ and $u = [F, \omega_x, \omega_y, \omega_z]^T$ denote the state and control input, respectively. $f : \mathbb{R}^9 \rightarrow \mathbb{R}^9$ and $G : \mathbb{R}^9 \rightarrow \mathbb{R}^{9 \times 4}$ compose a priori model representing our knowledge of the actual system, and $d : \mathbb{R}^9 \rightarrow \mathbb{R}^9$ is the model error representing the effects of wind disturbances acting on the quadrotor.

C. Problem Statement

The safe trajectory tracking problem considered herein can be described as follows. Let the quadrotor with a limited sensing range have an initial state (consisting of initial position x_0, y_0, z_0 , velocity $\dot{x}_0, \dot{y}_0, \dot{z}_0$, attitude θ_0, ϕ_0, ψ_0), and the reference yaw $\psi_d = 0$. The quadrotor should track

a given reference trajectory $p_d(t) = [x_d(t), y_d(t), z_d(t)]^T$ in a varying wind field while avoiding unexpected static and dynamic obstacles. If it deviates from the reference trajectory due to a sudden wind disturbance or after avoiding obstacles, it should rapidly converge to $p_d(t)$.

III. METHODOLOGY

In this section, we describe the proposed learning-based safety-preserving cascaded QP control (SPQC) for the system (8) that exploits the disturbance model to provide accurate tracking with safety and input constraints.

We follow a cascaded approach described in [3]. The cascaded controller consists of two QP controllers: a position-level QP controller and an attitude-level QP controller, as illustrated in Fig. 1. The position-level QP controller generates the desired thrust F , and the attitude-level QP controller utilizes this thrust together with the high confidence uncertainty interval $D(q)$ estimated via GPs to compute the desired body rotational rates $\omega = [\omega_x, \omega_y, \omega_z]^T$. Besides, a corresponding safety constraint is designed to minimally modify each nominal QP controller to keep the quadrotor within the obstacle-free region and avoid unexpected obstacles when the quadrotor is tracking the reference trajectory.

A. Gaussian Processes

We use the Gaussian process (GP) to learn the evolving model error $d(q)$ in (8) caused by uncertain wind disturbances d_w . A GP is a nonparametric regression method that can estimate complex functions and their uncertain distribution [12]. Given n observations $\mathbf{D}_n := \{q_i, \hat{d}(q_i)\}_{i=1}^n$, the mean and variance of $d(q_*)$ at the query state q_* can be given by

$$\mu(q_*) = \mathbf{k}_n^T (\mathbf{K} + \sigma^2 \mathbf{I})^{-1} \hat{\mathbf{d}}_n, \quad (9)$$

$$\sigma^2(q_*) = k(q_*, q_*) - \mathbf{k}_n^T (\mathbf{K} + \sigma^2 \mathbf{I})^{-1} \mathbf{k}_n, \quad (10)$$

respectively, where $\hat{\mathbf{d}}_n = [\hat{d}(q_1), \hat{d}(q_2), \dots, \hat{d}(q_n)]$ is the observed vector subject to a zero mean Gaussian noise $v \sim \mathcal{N}(0, \sigma^2)$. $\mathbf{K} \in \mathbb{R}^{n \times n}$ is the covariance matrix with entries, where $[\mathbf{K}]_{(i,j)} = k(q_i, q_j)$, $i, j \in \{1, \dots, n\}$, and $k(q_i, q_j)$ is the kernel function. $\mathbf{k}_n = [k(q_1, q_*), k(q_2, q_*), \dots, k(q_n, q_*)]$, and $\mathbf{I} \in \mathbb{R}^{n \times n}$ is the identity matrix.

With the system model error learned by GPs, a reliable high confidence interval $\mathcal{D}(q)$ on the uncertain dynamics $d(q)$ can be obtained by designing the constant c_δ [12].

$$\mathcal{D}(q) = \{d \mid \mu(x) - c_\delta \sigma(q) \leq d \leq \mu(q) + c_\delta \sigma(q)\}. \quad (11)$$

For instance, 95.5% and 99.7% confidence are achieved at $c_\delta = 2$ and $c_\delta = 3$, respectively.

B. Nominal Lyapunov-based Cascaded QP Controller

A nominal Lyapunov-based cascaded QP controller is proposed for trajectory tracking based on CLFs and GPs without the consideration of obstacle avoidance. We first construct a position-level QP controller to track the position of the quadrotor. Then, an attitude-level QP controller with a local motion planning strategy is designed to adjust the attitude of the quadrotor.

1) Position-level QP controller: Given a reference trajectory $p_d = [x_d, y_d, z_d]^T$, we can construct the following quadratic CLF:

$$V_p = \frac{1}{2} \lambda_1 e_p^T e_p + \frac{1}{2} \lambda_2 \dot{e}_p^T \dot{e}_p, \quad (12)$$

where $e_p = p - [x_d, y_d, z_d]^T$, $\dot{e}_p = \dot{p} - [\dot{x}_d, \dot{y}_d, \dot{z}_d]^T$, $p = [x, y, z]^T$ is the current position of the quadrotor, the $\lambda_1 > 0$ and $\lambda_2 > 0$.

Considering wind disturbances, GPs are employed to estimate the model uncertainties in terms of the predicted mean $\mu(q)$ and variance $\sigma(q)$ of the model error $d(q)$ through (9) and (10). As shown by our previous work [13], stability constraints can be constructed based on the CLF V_p to ensure stable high tracking performance for the uncertain dynamical system (8). Hence, based on our previous work [13], we can construct a position-level QP controller to compute a nominal thrust F_n^* for tracking as follows:

$$F_n^* = \arg \min_{(F, \beta) \in \mathbb{R}^2} \frac{1}{2} H_1 F + K_\beta \beta^2 \quad (13)$$

$$\begin{aligned} \text{s.t. } & L_g V_p F + L_f V_p + L_\mu V_p \\ & + c_\delta |L_\sigma V_p| \leq -c_p V_p + \beta, \quad (\text{Stability constraints}) \\ & 0 \leq F \leq F_{max}, \quad (\text{Control constraints}) \end{aligned}$$

where $F_{max} \in \mathcal{U}$ is the upper bound of thrust, and c_p is a positive constant. $H_1 \in \mathbb{R}^+$, $K_\beta \in \mathbb{R}^+$, and β is a slack variable to ensure there is no conflict among the stability constraints and control constraints in (13). $L_\mu V_p$ and $L_\sigma V_p$ denote the Lie derivatives of V_p with respect to μ and σ , respectively.

Remark 1. Note that the optimization (13) is not sensitive to the parameter K_β . The violation of the stability constraints can be heavily penalized as long as the K_β is large enough (e.g. 10^{20}).

2) Local motion planning (LMP): During trajectory tracking, a quadrotor may deviate from the reference trajectory due to a sudden wind disturbance or unexpected obstacles. In this study, we proposed an LMP algorithm to obtain a reference attitude for the quadrotor to drive it back to its reference trajectory under wind disturbances. The LMP method is described from the view of discrete control with a control period of Δt .

Let $p(t_1) = [x(t_1), y(t_1), z(t_1)]^T$ and $p_d(t_1) = [x_d(t_1), y_d(t_1), z_d(t_1)]^T$ denote the position of a quadrotor and the target waypoint of reference trajectory at time t_1 , respectively. We use GPs to estimate the mean wind disturbances $\mu_p(p(t_1)) = [d_x, d_y, d_z]^T$ at position level, and $\mu_{\dot{p}}(p(t_1)) = [\dot{d}_x, \dot{d}_y, \dot{d}_z]^T$ at velocity level based on (9). Hence, we can get the next nominal position $\tilde{p}(t_2) = [\tilde{x}(t_2), \tilde{y}(t_2), \tilde{z}(t_2)]^T$ and velocity $\tilde{\dot{p}}(t_2) = [\tilde{\dot{x}}(t_2), \tilde{\dot{y}}(t_2), \tilde{\dot{z}}(t_2)]^T$ of the quadrotor based on the nominal thrust F_n^* generated by the position-level QP controller (13).

To generate the desired orientation of the quadrotor for tracking, we can adjust the orientation of thrust at time t_1 to obtain the desired thrust direction at position level as illustrated in Fig. 2. The desired direction of the thrust F at the velocity level at t_2 can be obtained in a similar way

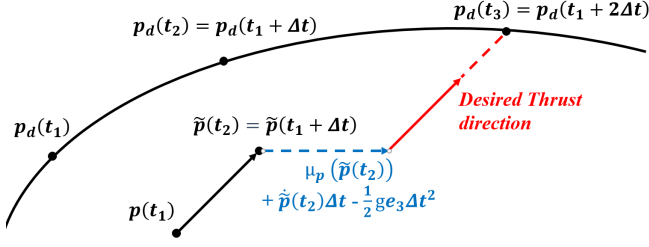


Fig. 2. **The local motion planning algorithm.** $p_d(t_1)$, $p_d(t_2)$ and $p_d(t_3)$ are three adjacent target waypoints in a reference trajectory. $\tilde{p}(t_2)$ and $\tilde{p}(t_2)$ are the next nominal position and velocity, respectively. The blue line is the offset vector of the quadrotor at position level at t_2 , and the red line is the desired thrust direction of the quadrotor at the position level.

to that of the desired thrust direction at the position level, and we do not show it in Fig. 2 due to lack of space.

The desired direction of the thrust F at position level at time t_2 can be denoted as follows:

$$\mathbf{R}_p e_3 = \frac{p_d(t_3) - \tilde{p}(t_2) - \mu_p(\tilde{p}(t_2)) - \dot{\tilde{p}}(t_2)\Delta t + \frac{g e_3}{2}(\Delta t)^2}{\|p_d(t_3) - \tilde{p}(t_2) - \mu_p(\tilde{p}(t_2)) - \dot{\tilde{p}}(t_2)\Delta t + \frac{g e_3}{2}(\Delta t)^2\|}. \quad (14)$$

The desired direction of the thrust F at velocity level at time t_2 can be denoted as follows:

$$\mathbf{R}_v e_3 = \frac{\dot{p}_d(t_3) - \dot{\tilde{p}}(t_2) - \mu_{\dot{p}}(\dot{\tilde{p}}(t_2)) + g e_3 \Delta t}{\|\dot{p}_d(t_3) - \dot{\tilde{p}}(t_2) - \mu_{\dot{p}}(\dot{\tilde{p}}(t_2)) + g e_3 \Delta t\|}. \quad (15)$$

Then, we can obtain the desired attitude $\Omega_{pd} = [\theta_{pd}, \phi_{pd}, \psi_d]^T$ at position level and the desired attitude $\Omega_{vd} = [\theta_{vd}, \phi_{vd}, \psi_d]^T$ at velocity level by solving (14) and (15). Thus, we can obtain the desired attitude Ω_d as follows:

$$\Omega_d = \lambda_3 \Omega_{pd} + (1 - \lambda_3) \Omega_{vd}, \quad (16)$$

where $\lambda_3 \in (0, 1)$ is a weight coefficient.

Hence, we can construct a quadratic CLF V_a based on the desired attitude Ω_d as follows:

$$V_a = \frac{1}{2} \lambda_4 e_{\Omega_p}^T e_{\Omega_p} + \frac{1}{2} \lambda_5 e_{\Omega_v}^T e_{\Omega_v}, \quad (17)$$

where $e_{\Omega_p} = \Omega - \Omega_{pd}$, $e_{\Omega_v} = \Omega - \Omega_{vd}$, $\Omega = [\theta, \phi, \psi]^T$, the $\lambda_4 > 0$ and $\lambda_5 > 0$.

3) Orientation-level QP controller: A procedure similar to design the position-level QP controller (13) can be used for the formulation of the attitude-level QP controller as follows:

$$\omega_n^* = \arg \min_{(\omega, \gamma) \in \mathbb{R}^4} \frac{1}{2} \omega^T H_2 \omega + K_\gamma \gamma^2 \quad (18)$$

$$\begin{aligned} \text{s.t. } & L_g V_a \omega + L_f V_a + L_\mu V_a \\ & + c_\delta |L_\sigma V_a| \leq -c_a V_a + \gamma, \quad (\text{Stability constraints}) \\ & -\omega_{max} \leq \omega \leq \omega_{max}, \quad (\text{Control constraints}) \end{aligned}$$

where ω_{max} is the upper bound of body rotational rates, $H_2 \in \mathbb{R}^{3 \times 3}$ is positive definite, c_a is a positive constant. γ is a slack variable for stability constraints, $K_\gamma \in \mathbb{R}^+$.

Remark 2. Note that the optimization (18) is not sensitive to the parameter K_γ . The violation of the stability constraints can be heavily penalized as long as the K_γ is large enough (e.g. 10^{20}).

With the Lyapunov-based cascaded QP controller composed of (13) and (18), the quadrotor can accurately track the reference trajectory $p_d(t)$ even under wind disturbances.

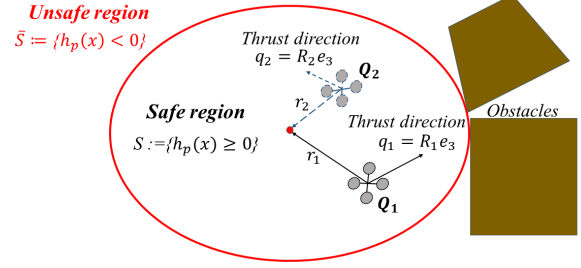


Fig. 3. **The illustration of quadrotor obstacle avoidance.** The success of quadrotor obstacle avoidance depends both on its position and orientation. The quadrotor is in the safe state S if it stays in the obstacle-free ellipsoid, and this can be captured through the condition: $h_p \geq 0$. The quadrotor Q2 is more capable of avoiding the obstacle than quadrotor Q1, and this can be captured through the condition: $\hat{h}_R := p_2 \cdot p_2 e_3 > 0 > h_R := p_1 \cdot p_1 e_3$.

C. Safety Barrier Scheme with a Limited Sensing Range

In practice, for quadrotor tracking in cluttered environments, the reference trajectory becomes unsafe when there exist unexpected obstacles in or near the trajectory. The quadrotor should sacrifice its tracking performance and adjust its attitude to avoid unexpected obstacles as shown in Fig. 3.

1) Safety obstacle-free region: An ellipsoid safe region can be obtained for the quadrotor with a limited sensing range based on the IRIS algorithm [9]. The IRIS algorithm is efficient in quickly computing the largest ellipsoidal regions of obstacle-free space through semidefinite programming. For a quadrotor tracking task, when there exist obstacles within its sensing region, a maximum ellipsoid in mathematical form can be constructed by utilizing the distance from the quadrotor to the obstacles in each direction.

Remark 3. Note that the quadrotor has a limited omnidirectional sensing range s_r , such that a point $\xi \in \mathbb{R}^{3 \times 1}$ on an obstacle \mathcal{B}_s is detected only when $\|p - \xi\| \leq s_r$, where $p = [x, y, z]^T$ is the current position of the quadrotor in the inertial coordinate system \mathcal{O} .

We represent the inscribed ellipsoid as an image of the unit ball: $\varepsilon(C, \zeta) = \{C o + \zeta \mid \|o\| = 1\}$, where $o \in \mathbb{R}^{3 \times 1}$, $o^T o = 1$ denotes a unit ball, $C \in \mathbb{R}^{3 \times 3}$ denotes the mapping matrix, and $\zeta \in \mathbb{R}^{3 \times 1}$ denotes the offset vector. The matrix C and ζ can be obtained for specific polyhedrons (e.g. obstacles) based on the IRIS algorithm. Then, we can get the following equation via equivalent transformation:

$$(\varepsilon - \zeta)^T C^{-1T} C^{-1} (\varepsilon - \zeta) = 1. \quad (19)$$

Finally, we can construct a ZCBF h_p at the position level by taking the inscribed ellipsoid as the safe region of the quadrotor:

$$h_p = 1 - (p - \zeta)^T C^{-1T} C^{-1} (p - \zeta), \quad (20)$$

where $p = [x, y, z]^T$ is the current position of quadrotor.

Remark 4. Note that $h_p \geq 0$ shows that the quadrotor stays within the safety ellipsoid region S .

We can construct another ZCBF h_q to drive the direction of the quadrotor thrust toward the center of the obstacle-free ellipsoid to avoid obstacles as illustrated in Fig. 3:

$$h_R = r \cdot q, \quad (21)$$

where $r = d - p$, $q = R e_3$.

2) **Safety-preserving cascaded QP control (SPQC)**: A minimally modified SPQC controller can be designed to avoid obstacles based on the ZCBF h_p , h_q and the estimated high-confidence interval $\mathcal{D}(q)$ (11) accounting for wind disturbances. The SPQC is designed to modify the nominal tracking controller in a way of minimally modified. GPs are employed to estimate the wind disturbances in terms of the predicted mean $\mu(q)$ and variance $\sigma(q)$ of the model error $d(q)$ through (9) and (10).

As shown by our previous work [13], safety constraints can be enforced on the uncertain system (8) based on the ZCBF h_p to keep the quadrotor's position in the safe region S . Hence, the **position-level safety constraints** can be enforced as follows:

Position-level QP: *Minimally modification*

$$F^* = \arg \min_{(F, \eta) \in \mathbb{R}^{m+1}} \|F - F_n^*\|^2 + K_\eta \eta^2 \quad (22)$$

$$\begin{aligned} \text{s.t. } & L_f h_p + L_g h_p u + L_\mu h_p \\ & - c_\delta |L_\sigma h_p| + \alpha_1 (h_p) \geq -\eta, \quad (\text{Safety constraints}) \\ & F_{\min} \leq F \leq F_{\max}, \quad (\text{Control constraints}) \end{aligned}$$

where F_n^* is the thrust generated by the nominal position-level controller (13), $K_\eta \in \mathbb{R}^+$, and $\eta \in \mathbb{R}$ is a slack variable.

To avoid obstacles, a safe attitude-level QP based on the ZCBF h_R is designed to construct the **attitude-level safety constraints** as follows:

Attitude-level QP: *Minimally modification*

$$\omega^* = \arg \min_{(\omega, \epsilon) \in \mathbb{R}^{m+1}} \|\omega - \omega_n^*\|^2 + K_\epsilon \epsilon^2 \quad (23)$$

$$\begin{aligned} \text{s.t. } & L_f h_R + L_g h_R u + L_\mu h_R \\ & - c_\delta |L_\sigma h_R| + \alpha_2 (h_R) \geq -\epsilon, \quad (\text{Safety constraints}) \\ & -\omega_{\max} \leq \omega \leq \omega_{\max}, \quad (\text{Control constraints}) \end{aligned}$$

where ω_n^* is the body rotational rates generated by the nominal attitude-level controller (18), $K_\epsilon \in \mathbb{R}^+$, and $\epsilon \in \mathbb{R}$ is a slack variable.

Remark 5. Note that the solution $F^*(x)$ and $\omega^*(x)$ to the QP in (22) and (23) are always feasible because the slack variables η and ϵ can ensure no conflict among the safety and control input constraints. Furthermore, the weights K_η and K_ϵ are set large values (e.g. $K_\eta = 10^{30}$, $K_\epsilon = 10^{30}$) to penalize safety violation, and hence, the optimizations in (22) and (23) are not sensitive to the K_η and K_ϵ parameters.

Remark 6. Note that the safety constraints in (22) and (23) are enforced to minimally modify the nominal controls generated by the Lyapunov-based cascaded QP controller composed of (13) and (18) via constrained QPs.

IV. NUMERICAL VALIDATION

We built a simulator with Python 3.6 to numerically validate the performance of the proposed SPQC controller. The python library CVXOPT [14] is utilized to solve the QP problem. The simulation time is set to be 20 s, and the control frequency is 50 Hz. The quadrotor model is a crazyflie 2.0 with a two-meter sensing range. The mass of the quadrotor $m = 0.027 \text{ kg}$, the maximum thrust $F_{\max} = 0.6 \text{ N}$, and the body rotational rate $|\omega| \leq 10 \text{ rad s}^{-1}$.

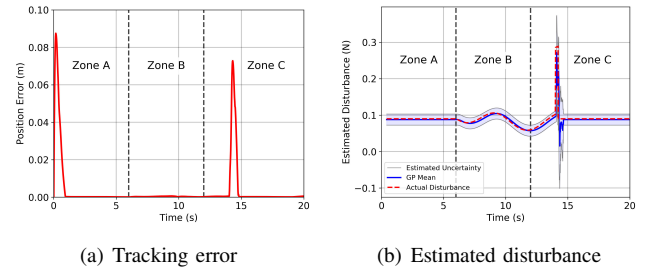


Fig. 4. Trajectory tracking result of the SPQC controller under wind disturbances. Note that the jumps in Zone C of 4(a) are caused by a sudden gust disturbance. The wind disturbance estimated via GPs is shown in 4(b).

TABLE I

ROOT MEAN SQUARE (RMS) TRACKING ERROR (IN METER) IN EACH ZONE FOR DIFFERENT CONTROLLERS ON THE QUADROTOR

Uncertainty	Wind Disturbance	NMPC [15]	SPQC-N	SPQC
Zone A	Constant	0.0499	0.0724	0.0198
Zone B	Changing	0.0409	0.0883	0.0003
Zone C	Sudden	0.0485	0.0867	0.0137

A. Experimental setup

In the experiments, we assess two aspects of the proposed SPQC controller: (i) trajectory tracking performance under different wind disturbances, and (ii) trajectory tracking with obstacle avoidance under varying wind disturbances. The reference trajectory is given as a spiral curve $p_d(t) = [2\sin(0.5t), 2-2\cos(0.5t), 0.2t]^T$ and $\psi_d(t) = 0$. The initial state of the quadrotor is $q = [0 \text{ m}, 0 \text{ m}, 0 \text{ m}, 0 \text{ m/s}, 0 \text{ m/s}, 0 \text{ m/s}, 0 \text{ rad}, 0 \text{ rad}, 0 \text{ rad}]^T$.

We use 3 GPs to learn the uncertainty vector $d(q)$ (8). Each GP uses the same mixture of linear and radial basis function (RBF) kernels $k(x, x') = \sigma_f^2 \exp(-\frac{1}{2}(x - x')^T L^{-2}(x - x'))$ to capture the model uncertainties that result from wind disturbances, where $L = 10$ and $\sigma_f = 1$. Each GP uses the past $T_s = 20$ observations collected at 50 Hz. To generate high confidence intervals to estimate wind disturbances, we use $[\mu(x) - 3\sigma(q), \mu(x) + 3\sigma(q)]$ as high confidence intervals $\mathcal{D}(q)$ (11).

B. Trajectory Tracking under Different Wind Disturbances

As shown in Fig. 4(b), we define zones A, B and C with different wind disturbances. In zone A, the wind disturbance is constant: $d_w = [-0.06, 0.06, 0.03]^T (N)$. In zone B, a varying wind field $\Delta d_w = -0.05 \sin(p - p_0) n_w (N)$, where $p_0 = [0.28, 4, 0]^T (m)$ and $n_w = \text{diag}([0.6, -0.7, 0])$ is added to d_w to assess the adaptability to changing wind disturbances. In zone C, a large wind gust $\Delta d_w = [0.2, 0.18, 0.1]^T (N)$ is added to d_w on the system from $t = 14(s)$ to $t = 14.2(s)$ to push the quadrotor away from the trajectory to assess the robustness to irresistible disturbances.

It can be seen from Fig. 4(a), the position error converges to a small value in each zone. The sudden gust pushes the quadrotor away from the trajectory when the quadrotor travels in zone C, and the SPQC controller enables the quadrotor back to the trajectory after the gust. Fig. 4(a) shows that the actual wind disturbance lies within the uncertainty interval estimated via GPs when the quadrotor is in zone A and B. Besides, the actual wind disturbance lies within the uncertainty interval after the sudden wind blow in zone C.

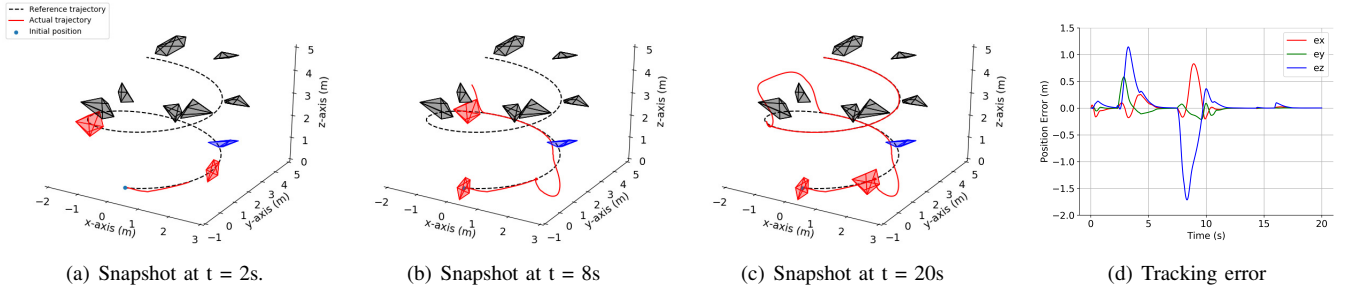


Fig. 5. Numerical validation of the quadrotor flight through a densely cluttered obstacle field under varying wind disturbances. Snapshots of the simulation are shown in 5(a), 5(b) and 5(c), respectively. The red solid line and the black dashed line denote the actual and reference trajectory, respectively. The red and blue obstacles denote the dynamic and static obstacle crossing the reference trajectory, respectively. The tracking error in each axis between the actual and reference trajectory is shown in 5(d). When the reference trajectory violates the safety constraint (by passing through an obstacle), the controller automatically relaxes trajectory tracking to strictly enforce safety constraints. Simulation video: <https://youtu.be/p-K3v9Z4OVa>.

We compare the SPQC controller with the NMPC [15] and an SPQC-N controller, where the SPQC-N is an ablation version of SPQC without estimated model uncertainties. The Root Mean Square Error (RMSE) of position tracking for each controller is shown in Table. I. In each zone, the SPQC method achieves a smaller tracking error than the NMPC and SPQC-N controller.

C. Trajectory Tracking with Obstacles

In the second experiment, the quadrotor is commanded to track the same dynamic time-varying trajectory $p_d(t)$ under a varying wind disturbance. The wind disturbance $d_w = [0.08\cos(y-1), 0.08\cos(x), 0.05\sin(z-2)]^T(N)$. The reference trajectory $p_d(t)$ is surrendered by static obstacles. Two dynamic obstacles fly to the quadrotor along the reference trajectory $p_d(t)$ with a speed of $0.78m/s$ when the quadrotor is tracking the $p_d(t)$.

Figure 5(d) shows the tracking error between the quadrotor position and the reference trajectory $p_d(t)$. When there are no obstacles along the trajectory, the tracking error converges to a small value. Furthermore, at about 2.5, 7, 14, and 16 seconds in the simulation time, the tracking errors rise to high values which indicates that the quadrotor has changed its trajectory to avoid collision with the detected obstacles. Fig. 5 shows that the SPQC controller stably tracks the reference trajectory when the trajectory is safe, while the controller relaxes tracking the reference when avoiding obstacles. These simulation results indicate that the SPQC controller can mediate the trade-off between safety and tracking performance.

V. CONCLUSION

This paper presents a novel learning-based SPQC scheme that achieves high-accuracy tracking performance while guaranteeing safety for the quadrotor with a limited sensing range under wind disturbances. The proposed SPQC scheme consists of a nominal tracking controller and the safety constraints based on ZCBFs. In this control scheme, the GPs are exploited to estimate the uncertainties of the wind disturbances, and an LMP algorithm is designed to generate the desired tracking attitude. By minimally modifying the nominal tracking controls, the quadrotor can avoid unexpected obstacles while staying within the safety region represented by ZCBFs. The proposed SPQC algorithm was

shown to improve tracking performance compared with an NMPC [15] method under different disturbances. Numerical simulation results show that the SPQC controller was capable of performing the trajectory tracking task with obstacle avoidance capacity under varying wind disturbances.

REFERENCES

- [1] K. Leung, E. Schmerling, M. Zhang, M. Chen, J. Talbot, J. C. Gerdes, and M. Pavone, "On infusing reachability-based safety assurance within planning frameworks for human-robot vehicle interactions," *Int. J. Rob. Res.*, vol. 39, no. 10-11, pp. 1326-1345, 2020.
- [2] M. Khan, M. Zafar, and A. Chatterjee, "Barrier functions in cascaded controller: Safe quadrotor control," in *Proc. Amer. Control Conf*, 2020, pp. 1737-1742.
- [3] G. Wu and K. Sreenath, "Safety-critical control of a 3d quadrotor with range-limited sensing," in *Proc. Dyn. Syst. Control Conf*, vol. 50695. American Society of Mechanical Engineers, 2016, p. V001T05A006.
- [4] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs for safety critical systems," *IEEE Trans. Autom. Control*, vol. 62, no. 8, pp. 3861-3876, 2016.
- [5] L. Wang, E. A. Theodorou, and M. Egerstedt, "Safe learning of quadrotor dynamics using barrier certificates," in *Proc. IEEE int. Conf. Robot. Autom.*, 2018, pp. 2460-2465.
- [6] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789-814, 2000.
- [7] Castillo-Lopez and e. a. Manuel, "Model predictive control for aerial collision avoidance in dynamic environments," in *Proc. Mediterr. Conf. Control Autom.*, 2018, pp. 1-6.
- [8] J. C. Pereira, V. J. Leite, and G. V. Raffo, "Nonlinear model predictive control on se (3) for quadrotor trajectory tracking and obstacle avoidance," in *Proc. IEEE int. Conf. Robot. Autom.*, 2019, pp. 155-160.
- [9] R. Deits and R. Tedrake, "Computing large convex regions of obstacle-free space through semidefinite programming," in *Algorithmic foundations of robotics XI*, 2015, pp. 109-124.
- [10] X. Xu, P. Tabuada, J. W. Grizzle, and A. D. Ames, "Robustness of control barrier functions for safety critical control," *IFAC-PapersOnLine*, vol. 48, no. 27, pp. 54-61, 2015.
- [11] G. Shi, X. Shi, M. O'Connell, R. Yu, K. Azizadenesheli, A. Anandkumar, Y. Yue, and S.-J. Chung, "Neural lander: Stable drone landing control using learned dynamics," in *Proc. IEEE int. Conf. Robot. Autom.*, 2019, pp. 9784-9790.
- [12] C. E. Rasmussen and C. Williams, "Gaussian processes for machine learning the mit press," *Cambridge, MA*, 2006.
- [13] L. Zheng, R. Yang, J. Pan, H. Cheng, and H. Hu, "Learning-based safety-stability-driven control for safety-critical systems under model uncertainties," in *Proc. Int. Conf. Wirel. Commun. and Signal Process.*, 2020, pp. 1112-1118.
- [14] M. S. Andersen, J. Dahl, and L. Vandenbergh, "Cvxopt: A python package for convex optimization," *abel. ee. ucla. edu/cvxopt*, vol. 88, 2013.
- [15] M. Owis, S. El-Bouhy, and A. El-Badawy, "Quadrotor trajectory tracking control using non-linear model predictive control with ros implementation," in *Proc. Int. Con. Control Mechatron. Autom.*, 2019, pp. 243-247.