

External Forces Resilient Safe Motion Planning for Quadrotor

Yuwei Wu^{1,2}, Ziming Ding¹, Chao Xu¹ and Fei Gao¹

Abstract—Adaptive autonomous navigation with no prior knowledge of extraneous disturbance is of great significance for quadrotors in a complex and unknown environment. The mainstream approach that considers external disturbance is to implement disturbance-rejected control and path tracking. However, the robust control to compensate for tracking deviations is not well-considered regarding energy consumption, and even the reference path will become risky and intractable with disturbance. As recent external forces estimation advances, it is possible to incorporate a real-time force estimator to develop more robust and safe planning frameworks. This paper proposes a systematic (re)planning framework that can resiliently generate safe trajectories under volatile conditions. Firstly, a front-end kinodynamic path is searched with force-biased motion primitives. Then we develop a nonlinear model predictive control (NMPC) as a local planner with Hamilton-Jacobi (HJ) forward reachability analysis for error dynamics caused by external forces. It guarantees collision avoidance by constraining the ellipsoid of the quadrotor body expanded with the forward reachable sets (FRSs) within safe convex polytopes. Our method is validated in simulations and real-world experiments with different sources of external forces.

Index Terms—Motion and Path Planning; Aerial Systems: Applications; Collision Avoidance

I. INTRODUCTION

Safe trajectory generation under real-world conditions where quadrotors are exposed to external disturbance and other sources of uncertainty has been a crucial concern, especially in complex and dynamic environments [1]–[3]. External forces caused by air turbulence, loaded objects, or other uncertain disturbance induce a considerable effect on the quadrotor system and challenge planning stability.

Existing attempts mainly focus on robust control with unpredictable disturbance to prevent the quadrotor from deviations to its desired trajectories. Disturbance-aware controllers for quadrotors like robust MPC [4] and tube-based MPC [5] aim to follow a predefined reference path with robustness guarantees and try to compensate for the influence of the disturbance. However, these methods either may fail to handle

Manuscript received: March 17, 2021; Revised: June 25, 2021; Accepted: August 18, 2021. This paper was recommended for publication by Editor Hanna Kurniawati upon evaluation of the Associate Editor and Reviewers' comments. This work was supported by National Natural Science Foundation of China under Grant 62088101 and Grant 62003299. (*Corresponding author: Fei Gao, Chao Xu*)

¹State Key Laboratory of Industrial Control Technology, Institute of Cyber-Systems and Control, Zhejiang University, Hangzhou 310027, China, and Huzhou Institute of Zhejiang University, Huzhou 313000, China.

²Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, PA 19104 USA

E-mail: yuweiwu@seas.upenn.edu, {zm_ding, cxu, fgaoaa}@zju.edu.cn

Digital Object Identifier (DOI): see top of this page.

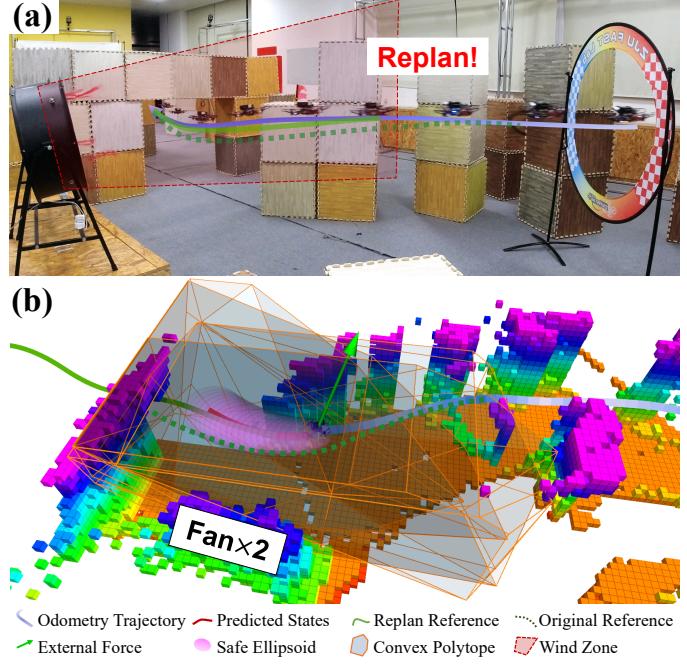


Fig. 1. An indoor flight when the quadrotor enters a wind zone. (a) The wind pushes the quadrotor to obstacles while our planner replans a new feasible reference (green dotted) to adapt. (b) Visualizations. Video is at: <https://youtu.be/nSKbzAM0v18>

tremendous forces because of the controller's limitations or generate too conservative trajectories because they consider all sources of uncertainty. Another work [6] addresses this issue by adding a receding horizon, corridor-based, local adaptive tracking layer between the planner and the controller. When the quadrotor encounters an external force, this method directly replans a new trajectory on the current state without any estimation of this force. Due to its lack of modeling on the external force, it can not handle sustained forces and fails to solve a feasible path. Other works partly solve this problem by planning under known constant wind [3] or with prior information of a force field [7]. These methods partially simplify the environments and need reliable predictions of the external force. As a result, it is impractical in real-world situations with dynamic and unexpected external disturbances.

In this work, we propose a systematic framework to achieve robust local planning, which accounts for the influence of extreme external disturbance¹. The presented method searches a dynamically feasible path by kinodynamic hybrid-state A* [8] as a rough reference with added nominal force estimated

¹The source code is available at https://github.com/ZJU-FAST-Lab/forces_resilient_planner

by the method in [9] to generate motion primitives. Then we formulate an optimal control problem (OCP) for trajectory generation based on the reference path, which encodes the external force with its uncertainty. This OCP can be solved by a corridor-based NMPC that exploits the full dynamics and geometrical shape of a quadrotor. We extend the ellipsoidal approximation of FRSs [10] which are the state sets of error dynamics in a planning horizon. Therefore, we can constrain the discrete-time FRSs in a safe flight corridor for collision avoidance. The real-world experiments illustrate the robustness of our method, and an indoor flight is shown in Fig. 1.

Our contributions can be summarized as:

- An efficient front-end kinodynamic path searching and collision checking method with external forces consideration.
- An online trajectory optimization algorithm that utilizes nonlinear model predictive control with safe ellipsoid boundaries constrained in a convex flight corridor to enforce reliable obstacle avoidance.
- A (re)planning framework for quadrotor that integrates the VID-Fusion external force estimator [9]. The onboard experiments demonstrate the performance of our system in volatile real-world environments.

II. RELATED WORK

A. Modeling of External Forces

Accurate estimation and prediction of external forces assist robust control [11] and offset-free trajectory tracking for a quadrotor. Some works attempt to present this special external force by modeling the disturbance as a predicted spatiotemporally varying wind field [7] or a Gaussian mixture field [12]. For more generic conditions with unknown types of external forces, different force estimators are employed. Extended Kalman Filter (EKF) is applied in [4] to estimate the external disturbance while utilizing the same model used in the inner-loop attitude dynamics to decrease the tracking errors. Other approaches like nonlinear observer [13], momentum-based estimator [11], and optimization-based methods [9, 14] are also widely used for visual-inertial odometry (VIO) systems and force estimation.

To further include the external force into a quadrotor system, Kamel et al. [4] directly add an estimated force to the dynamic model. Other methods model the disturbance as a zero-mean, bounded disturbance [10] with handcrafted bound modification when the drone encounters sudden winds, or as Gaussian uncertainty distributions propagated with $3-\sigma$ confidence ellipses [15]. These approaches can guarantee the safety and robustness of a quadrotor system but lack the flexibility to handle excessive disturbance. With an external force estimated in [9], we can apply a tighter bounded noise as the uncertainty of external force into the model.

B. Robust Planning With Disturbance

Researchers have focused on OCP of tracking a given reference path with robustness consideration to handle uncertainties and disturbances. MPC is an efficient tool to generate optimal local trajectories while encodes uncertainty and obstacle

avoidance. Although linear MPC demonstrates its excellent performance on trajectory generation and navigation [6, 16], nonlinear MPC of full dynamics is better regarding disturbance rejection, especially for aggressive trajectory tracking [4, 15].

The effects of external forces require a quadrotor system to apply higher-level strategies to resist disturbances and plan a safe and energy-efficient path. Among all attempts for robust path planning, Singh et al. [5] generate a conservative trajectory subject to bounded disturbances by pre-computing a globally valid invariant tube, while others [1, 12] conduct global planning assuming that a quadrotor is flying in a region of known time-varying winds. However, such methods rely on the accuracy of disturbance predictions and modeling, which are not practicable for real-world planning. To solve these limitations, we incorporate kinodynamic path searching and NMPC local planning to generate safe trajectories with real-time estimated force. In terms of obstacle avoidance and free space representation, the common method represents the obstacles by flexible parameterization [10] and incorporates cost functions with the distance to the nearest obstacle. Another representation of free space in an environment generates a safe flight corridor of several convex polytopes [17]–[19] to constrain the trajectory. Our method also exploits such representation [19] to generate large convex polytopes directly on an occupancy map.

III. PLANNING WITH EXTERNAL FORCES

A. Quadrotor Dynamic Model

We consider the dynamic model with accurate control of Euler angles, which assumes that its rate can accurately track the desired command. The system state is $\mathbf{x} = [\mathbf{p}, \mathbf{v}, \phi, \theta, \psi]^T \in \mathcal{X} \subset \mathbb{R}^{n_x}$, where $\mathbf{p} = [p_x, p_y, p_z]^T$, $\mathbf{v} = [v_x, v_y, v_z]^T \in \mathbb{R}^3$ denote the position and velocity of the quadrotor. $\phi, \theta, \psi \in \mathbb{R}$ are the roll, pitch and yaw angles. The control input is $\mathbf{u} = [\dot{\phi}_c, \dot{\theta}_c, \dot{\psi}_c, T_c]^T \in \mathcal{U} \subset \mathbb{R}^{n_u}$ in which $\dot{\phi}_c, \dot{\theta}_c, \dot{\psi}_c \in \mathbb{R}$ are the command rates of the Euler angles, $T_c \in \mathbb{R}$ is thrust command of the quadrotor in body frame. We provide a full-body nonlinear dynamic model as follows:

$$\dot{\mathbf{p}} = \mathbf{v}, \quad (1a)$$

$$\dot{\mathbf{v}} = \frac{1}{m} (\mathbf{R} \begin{bmatrix} 0 \\ 0 \\ T_c \end{bmatrix} - \mathbf{F}_{drag} + \mathbf{F}_{ext}) - \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix}, \quad (1b)$$

$$\dot{\phi} = \dot{\phi}_c, \quad (1c)$$

$$\dot{\theta} = \dot{\theta}_c, \quad (1d)$$

$$\dot{\psi} = \dot{\psi}_c, \quad (1e)$$

where $\mathbf{R} \in \mathbb{R}^{3 \times 3}$ is the rotation matrix parameterized by the Euler angles, $g \in \mathbb{R}$ is the magnitude of gravitational acceleration, $m \in \mathbb{R}$ is the mass of the quadrotor. We can represent the nonlinear dynamics as $\dot{\mathbf{x}} = f(\mathbf{x}(t), \mathbf{u}(t), \mathbf{F}_{ext}(t))$. Components of the rotor drag and drag like effects are complicated to model and correlated with quadrotor's velocity, we consider it in the system to compensate its interference to disturbance bound [20]. To further simplify its aerodynamics, we apply a first-order drag model by adding the drag force \mathbf{F}_{drag} in the x–y body frame. We define $\mathbf{K}_{drag} := diag\{k_d, k_d, 0\}$, where

k_d is the drag coefficient constant, then the drag force in the world frame is

$$\mathbf{F}_{drag} = \mathbf{R}\mathbf{K}_{drag}\mathbf{R}^T\mathbf{v}. \quad (2)$$

Instead of predicting the external forces or modeling the field of special force like winds, we generally obtain a real-time force estimation and then adjust the planning strategy when the force surpasses its bound. The disturbance of external forces can be expressed as a nominal value with an additive bounded noise, which makes it possible to design an NMPC that satisfies the constraints under bounded disturbance. The nominal force \mathbf{b}_{ext} can be treated as a constant value calculated in [9] for a sufficiently short duration. The external force is defined as $\mathbf{F}_{ext} := \mathbf{b}_{ext} + \mathbf{w}_{ext}$, where the bounded noise $\mathbf{w}_{ext} \in \mathbb{W} = \{\mathbf{w} \in \mathbb{R}^{n_w} : \|\mathbf{w}\|_\infty \leq w_m\}$, w_m is the maximum bound.

B. Collision Avoidance Constraints

1) *Error Dynamics*: We construct a linearized system for closed-loop dynamics of error state $e(t) = \mathbf{x}(t) - \underline{\mathbf{x}}(t)$ around nominal states $\underline{\mathbf{x}}(t)$, inputs $\underline{\mathbf{u}}(t)$ and nominal external force \mathbf{b}_{ext} . Assuming that the linearization error can be neglected around nominal states with the condition that the feedback control policy can be expressed as $\mathbf{u}(t) = \mathbf{K}(t)\mathbf{e}(t) + \underline{\mathbf{u}}(t)$, $\mathbf{K}(t)$ is the feedback gain. We can get the error dynamics as

$$\dot{\mathbf{e}}(t) = \Gamma(t)\mathbf{e}(t) + \mathbf{D}(t)\mathbf{F}_{ext}(t), \quad (3)$$

where $\Gamma(t) = \mathbf{A}(t) + \mathbf{B}(t)\mathbf{K}(t)$, $\mathbf{A}(t) := \partial f / \partial \mathbf{x}|_{(\underline{\mathbf{x}}, \underline{\mathbf{u}}, \mathbf{b}_{ext})}$, $\mathbf{B}(t) := \partial f / \partial \mathbf{u}|_{(\underline{\mathbf{x}}, \underline{\mathbf{u}}, \mathbf{b}_{ext})}$, $\mathbf{D}(t) := \partial f / \partial \mathbf{F}_{ext}|_{(\underline{\mathbf{x}}, \underline{\mathbf{u}}, \mathbf{b}_{ext})}$.

In order to propagate the uncertainty of external forces on each nominal state, we apply the analytic solution based on Hamilton-Jacobi (HJ) reachability analysis that quantitatively expresses the error FRS $\mathcal{E}(t)$ as approximate ellipsoids [10]. With a more precious estimation of the external forces by VID-Fusion [9], a less conservative error FRSs could be obtained by employing a tighter variance bound.

2) *Propagation of Safe Ellipsoid Boundary*: To further solve the problem, we discrete the dynamics with a sampling time t_s over N time steps, as $\mathbf{x}^{k+1} = f_d(\mathbf{x}^k, \mathbf{u}^k, \mathbf{F}_{ext}^k)$. For convenience, we denote the discrete-time state as $\mathbf{x}^k = [\mathbf{p}^k, \mathbf{v}^k, \phi^k, \theta^k, \psi^k]^T$ at stage k , $\forall k \in \{0, 1, \dots, N-1\}$. The rotation matrix is \mathbf{R}^k , the error FRS is \mathcal{E}^k , and the external force is denoted as $\mathbf{F}_{ext}^k = \mathbf{b}_{ext} + \mathbf{w}_{ext}^k$.

We inflate the geometrical shape of a quadrotor to ensure the safety of our generated trajectory. The quadrotor is modeled as an ellipsoid with a radius r and a height h . Therefore, we can represent ego geometrical ellipsoid at stage k as

$$\xi(\mathbf{p}^k, \mathbf{Q}_{ego}^k) := \{\mathbf{p} \in \mathbb{R}^3 : (\mathbf{p} - \mathbf{p}^k)^T (\mathbf{Q}_{ego}^k)^{-1} (\mathbf{p} - \mathbf{p}^k) \leq 1\}, \quad (4)$$

which is centered on quadrotor's position \mathbf{p}^k with a shape matrix as

$$\mathbf{Q}_{ego}^k = \mathbf{R}^k \text{diag}\{r^2, r^2, h^2\} (\mathbf{R}^k)^T \in \mathbb{S}_+^3. \quad (5)$$

As shown in Fig. 2, the safe ellipsoid boundary $\xi(\mathbf{p}^k, \mathbf{Q}^k)$ is an outer ellipsoid that includes the shape of a quadrotor $\xi(\mathbf{p}^k, \mathbf{Q}_{ego}^k)$ and ellipsoidal approximation of error FRS

$\mathcal{E}^k = \xi(\mathbf{p}^k, \mathbf{Q}_{ext}^k)$, where $\mathbf{Q}^k, \mathbf{Q}_{ext}^k \in \mathbb{S}_+^3$ are the shape matrices accordingly. Given two ellipsoids centered on the same point, the optimal outer ellipsoidal approximation which is guaranteed to contain the Minkowski sum [21] of these ellipsoids is denoted by

$$\mathbf{Q}_1 \boxplus \mathbf{Q}_2 = (1 + \beta)\mathbf{Q}_1 + (1 + \frac{1}{\beta})\mathbf{Q}_2, \quad (6)$$

where $\beta = \sqrt{\frac{\text{tr}(\mathbf{Q}_2)}{\text{tr}(\mathbf{Q}_1)}}$, $\mathbf{Q}_1, \mathbf{Q}_2 \in \mathbb{S}_+^3$ are the shape matrices of ellipsoids, \mathbb{S}_+^3 denotes the real symmetric positive definite matrices.

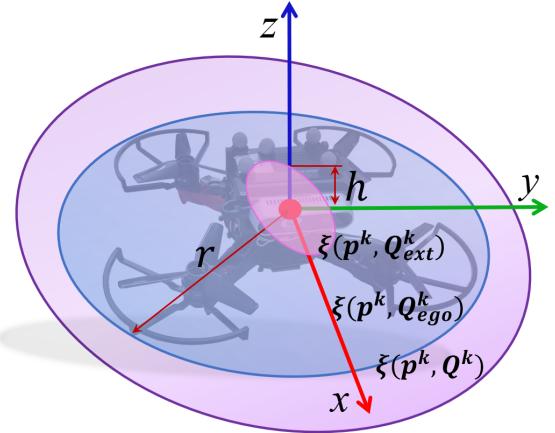


Fig. 2. The inner ellipsoid (in pink) is the ellipsoidal approximation of the error forward reachable set, the middle one (in blue) is the geometrical modeling of the quadrotor. The outer ellipsoid (in purple) is the approximation of ellipsoidal sum.

Therefore, the shape matrix $\mathbf{Q}^k \in \mathbb{S}_+^3$ of the safe ellipsoid boundary is obtained by $\mathbf{Q}^k = \mathbf{Q}_{ego}^k \boxplus \mathbf{Q}_{ext}^k$. Because we use the discrete representation of the model, we propagate \mathbf{Q}_{ext}^k in each stage with current error initial shape matrix \mathbf{Q}_0^k and system matrix Γ^k of quadrotor rather than using the same initial state during a sampling time t_s . The max outer ellipsoid propagation law of error FRS can be computed by

$$\mathbf{Q}_{ext}^k = [\exp(\Gamma^k t_s)(\mathbf{Q}_0^k \boxplus \mathbf{Q}_d^k) \exp(\Gamma^{k,T} t_s)]_{3 \times 3}. \quad (7)$$

Note that we only concern the uncertainty of quadrotor's position in the total error ellipsoid for further obstacle avoidance, so we directly block the shape matrix to the first 3-dimension matrix, where $[\cdot]_{3 \times 3}$ denotes the first 3×3 block of the matrix. The maximal ellipsoidal reachable set \mathbf{Q}_d^k of error dynamics is individually computed by [10] with the converted discrete-time model. The error initial shape matrix \mathbf{Q}_0^{k+1} is updated with the outer approximation as $\mathbf{Q}_0^{k+1} = \mathbf{Q}_0^k \boxplus \mathbf{Q}_d^k$.

3) *Safe Corridor Constraints*: With the propagated safe ellipsoid boundary in each stage, we can encode collision avoidance constraints by restricting the ellipsoid in a convex collision-free corridor. To ensure sufficient freedom for NMPC to get a better and refined trajectory, we use the method in [19] to rapidly generate a series of convex polytopes to cover reference waypoints in the current planning horizon, as shown in Fig. 3. The convex polyhedron can be represented as linear constraints with $\mathbf{A}_i \in \mathbb{R}^{m \times 3}$ and $\mathbf{b}_i \in \mathbb{R}^m$, $\forall i \in \{1, 2, \dots, m\}$, pre-assigned with the reference waypoints.

Inspired by the computation of the maximum volume inscribed ellipsoid in a polytope [22], we have

$$\{\mathbf{p} : \mathbf{p} \in \xi(\mathbf{p}^k, \mathbf{Q}^k)\} \in \{\mathbf{p} : \mathbf{A}_i \mathbf{p} \leq \mathbf{b}_i\}. \quad (8)$$

It is equal to solve the problem

$$h(\mathbf{p}^k, \mathbf{Q}^k) = \max \mathbf{A}_i \mathbf{p} \leq \mathbf{b}_i, \quad (9a)$$

$$\text{s.t. } (\mathbf{p} - \mathbf{p}^k)^T (\mathbf{Q}^k)^{-1} (\mathbf{p} - \mathbf{p}^k) \leq 1. \quad (9b)$$

This optimization problem can be easily solved analytically, so that the collision avoidance constraint can be obtained by

$$h(\mathbf{p}^k, \mathbf{Q}^k) = \|(\mathbf{Q}^k)^{\frac{1}{2}} \mathbf{A}_i^T\| + \mathbf{A}_i \mathbf{p}^k \leq \mathbf{b}_i. \quad (10)$$

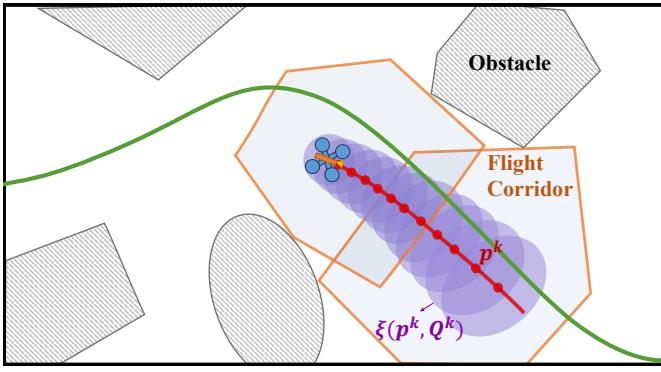


Fig. 3. Safety constraints by restricting the safe ellipsoids within a flight corridor. In the planning horizon, the safe ellipsoid boundary at stage k is $\xi(\mathbf{p}^k, \mathbf{Q}^k)$, as shown in Fig. 2.

C. Objective Functions

For robust reference tracking and yaw control, while reducing the control efforts, we penalize the cost with the following three terms.

1) *Navigation Cost*: In each stage, we minimize the tracking deviation from the reference position to the predicted one. Other references as velocity and acceleration are relaxed during the optimization. A yaw angle sequence is pre-computed according to the orientation of reference velocity of the front-end path. We define the stage cost as

$$J_{\mathbf{x}}^k(\mathbf{x}^k) = \|\mathbf{p}^k - \mathbf{p}_{ref}^k\|_{l_p} + \|\psi^k - \psi_{ref}^k\|_{l_\psi}, \quad (11)$$

where $\|\cdot\|_{l_p}$ and $\|\cdot\|_{l_\psi}$ are weighted squared norm induced by the l_p and l_ψ matrices. The terminal cost is

$$J^N(\mathbf{x}^N) = \|\mathbf{p}^N - \mathbf{p}_{ref}^N\|_{l_p^N} + \|\psi^N - \psi_{ref}^N\|_{l_\psi^N}, \quad (12)$$

where l_p^N and l_ψ^N are the terminal weighting matrices. When the remained reference state is beyond its planning horizon, a velocity penalty term $\|\mathbf{v}^N\|_{l_v^N}$ is added in the final stage to approach the terminal condition of rest.

2) *Control Input Cost*: In the aspect of control efforts, the penalty of the control input should be added as follows:

$$J_{\mathbf{u}}^k(\mathbf{u}^k) = \|\mathbf{u}^k\|_{l_u^k}, \quad (13)$$

l_u^k is the weighting matrix of control input cost.

3) *Control Input Rate Cost*: For the smoothness of the control input, we also penalize the variations between current and previous inputs for avoiding oscillatory trajectories, as

$$J_{\Delta \mathbf{u}}^k(\mathbf{u}^{k+1}, \mathbf{u}^k) = \|\mathbf{u}^{k+1} - \mathbf{u}^k\|_{l_{\Delta \mathbf{u}}^k}, \quad (14)$$

$l_{\Delta \mathbf{u}}^k$ is the weighting matrix of the smoothness cost.

D. External Forces Resilient NMPC

Finally, we can formulate the receding horizon optimization problem with nominal external force as

$$\min_{x,u} J^N(\mathbf{x}^N) + \sum_{k=0}^{N-1} J_{\mathbf{x}}^k(\mathbf{x}^k) + J_{\mathbf{u}}^k(\mathbf{u}^k) + J_{\Delta \mathbf{u}}^k(\mathbf{u}^{k+1}, \mathbf{u}^k), \quad (15a)$$

$$\text{s.t. } \mathbf{x}^{k+1} = f_d(\mathbf{x}^k, \mathbf{u}^k, \mathbf{b}_{ext}), \quad (15b)$$

$$\mathbf{x}^0 = \mathbf{x}_0, \quad (15c)$$

$$h(\mathbf{p}^k, \mathbf{Q}^k) \leq \mathbf{b}_i, \quad (15d)$$

$$\mathbf{Q}^k = \mathbf{Q}_{ego}^k \boxplus \mathbf{Q}_{ext}^k, \quad (15e)$$

$$\mathbf{u}^k \in \mathcal{U}, \quad \mathbf{x}^k \in \mathcal{X}, \quad (15f)$$

where $\mathbf{x}_0 \in \mathcal{X}$ is the initial state, \mathbf{Q}_{ext}^k and \mathbf{Q}_{ego}^k in (15e) are computed and propagated by equation (5) - (7). To reduce the problem's complexity, \mathbf{Q}^k is pre-computed before each optimization using the last loop planned series of Euler angles. Upper and lower bounds for control inputs and states should also be considered in (15f).

E. Path Searching With Nominal Force

The reference collision-free path we mention above is generated based on a kinodynamic hybrid-state A* proposed in [8], which searches a kinodynamic feasible trajectory in an occupancy map that minimizes time duration and control cost. When expanding the motion primitives, we extend the simplified kinematic model with added nominal external force, which is better for computational efficiency compared with full nonlinear dynamics. This makes reference local path reliable under the specific external force instead of searching a global path which may collide if the system suffers such force. The front-end trajectory can be represented in three dimensions as time-parameterized piece-wise polynomials, $\Phi(t) = [\Phi_x(t), \Phi_y(t), \Phi_z(t)]^T$. The state is $\mathbf{s}(t) := [\Phi(t)^T, \dot{\Phi}(t)^T]^T$, and the control input is $\mathbf{r}(t) = \ddot{\Phi}(t)$. Hence, the state-space model is

$$\dot{\mathbf{s}} = \begin{bmatrix} 0 & \mathbf{I}_3 & 0 \\ 0 & 0 & \mathbf{I}_3 \\ 0 & 0 & 0 \end{bmatrix} \mathbf{s} + \begin{bmatrix} 0 \\ 0 \\ \mathbf{I}_3 \end{bmatrix} (\mathbf{r} + \frac{1}{m} \mathbf{b}_{ext}). \quad (16)$$

The collision-free path in the planning phase is naturally close to the obstacles and may become unsafe due to the variance of the external force. Therefore, rather than the exact path following, we only consider the feasible path as a rough reference for corridor generation and orientation guiding.

IV. IMPLEMENTATION DETAILS

A. External Force Estimation

To ensure the robustness and accuracy of the estimation, a tightly-coupled Visual-Inertial-Dynamics estimator (VID-Fusion) [9] is used to optimize the pose and external force simultaneously via nonlinear optimization, which is suitable for our adaptive planning framework. We exploit the estimated force in [9] as the sum of nominal external force and drag force, and then redefine the external force \mathbf{F}_{ext} as a resultant force in the world frame except for rotor thrust, gravity, and drag force in our system.

B. System Overview of the (Re)Planning Framework

1) *External Forces Adaptation*: We apply kinodynamic path searching [8] with a constant nominal force to generate a front-end feasible path as reference. Because the update frequency of external force is much higher than replanning frequency, we introduce a noise bound to withstand the deviation of estimated external force in acceptable time duration. The back-end nonlinear MPC considers the error states as a series of ellipsoidal forward reachable sets, admitting external force to vary within this bound during the tracking phase. Therefore the followed waypoints and the optimized predicted states are guaranteed to be safe under such external forces.

2) *Replanning Activation*: The replanning strategy is event-triggered under the feasible consideration of the external force and reference path, which combines the real-time external force both in front-end and back-end planning, as shown in Fig. 4. When the variance of external force is upper the allowed bound, it is too fierce for NMPC to solve an acceptable solution on the current initial state. Then, the front-end reference needs to replan to fit with the current force. With the update of the occupancy map, if the global target or the reference path collides with obstacles, or the time-indexed reference path is hard to follow, then the replanning is triggered to generate a new path.

V. RESULTS

A. Setups

We use a code generation tool Forces Pro [23] to solve our NMPC problem and directly use its generated C library for speed-up. We apply a receding horizon with $t_s = 50$ ms and total time steps $N = 20$. The average solving time is around 5 ms on an Intel i7-10700 CPU computer for simulation tests and within 10 ms on our quadrotor platform. We define a mass-normalized external force $\mathcal{F}^e = [\mathcal{F}_x^e, \mathcal{F}_y^e, \mathcal{F}_z^e]^T$ obtained by VID-Fusion. The noise bound of \mathcal{F}^e is set as 0.5 m/s^2 according to our experiment data, to resist its allowable deviation during the planning phase.

The RotorS MAVs simulator [24] which includes physical engines, is employed for simulated flight tests. As shown in Fig. 5, we separately make two scenarios of benchmark comparison. The first comparison is tracking conducted with an elliptical reference trajectory, while the second compares the whole planning framework. An equivalent command acceleration that sends to the controller is defined as $a^e =$

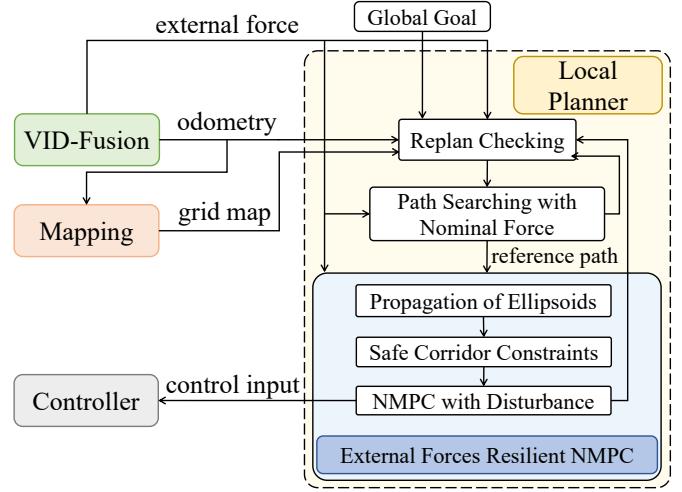


Fig. 4. The system overview diagram of the (re)planning framework. The VID-Fusion provides the odometry and estimated external force simultaneously, while our planning module checks this time-varying external force. If the force is within its allowed bound, the back-end NMPC is flexible to generate safe trajectories; otherwise, it triggers the replanning and reset reference path.

$[a_x^e, a_y^e, a_z^e]^T$ in the world frame for further evaluations. We set the maximum velocity of the quadrotor as 2.0 m/s.

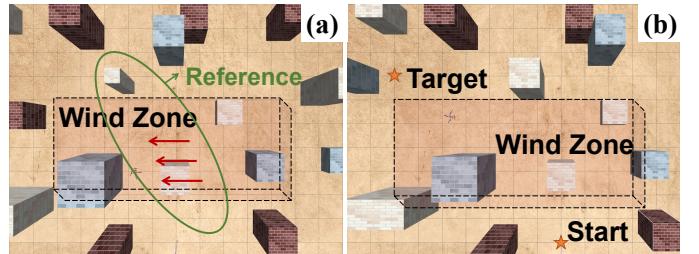


Fig. 5. The experiment settings for comparison testing. (a) The test for reference tracking. A predefined reference trajectory is provided, which is partially in the wind zone. (b) The test for local planning. With a global target, the drone aims to fly through the wind zone.

The real-world tests are presented in the unknown indoor and outdoor environments. We use the autonomous quadrotor platform in [9], with an onboard computer (i7-8550U), a stereo camera (Intel Realsense D435) for real-time mapping, a DJI N3 controller and a rotor speed measurement unit. The radius and height of our quadrotor is $r = 0.22 \text{ m}$ and $h = 0.13 \text{ m}$. The drag coefficient is identified as $k_d = 0.33$. The external force estimation setup follows the pipeline in [9].

B. Simulation Tests

1) *Comparison of Reference Tracking*: We firstly evaluate our external forces resilient NMPC with other MPCs [4, 10] with disturbance consideration under the same environments. Because other MPC-based methods require the predefined path and information of obstacles, for fairness to all MPCs, we provide a known global path for tracking and implement a disturbance observer for external force estimation. Method in [10] solves the MPC by unscented dynamic programming and needs more than 70 ms computation time for similar scale

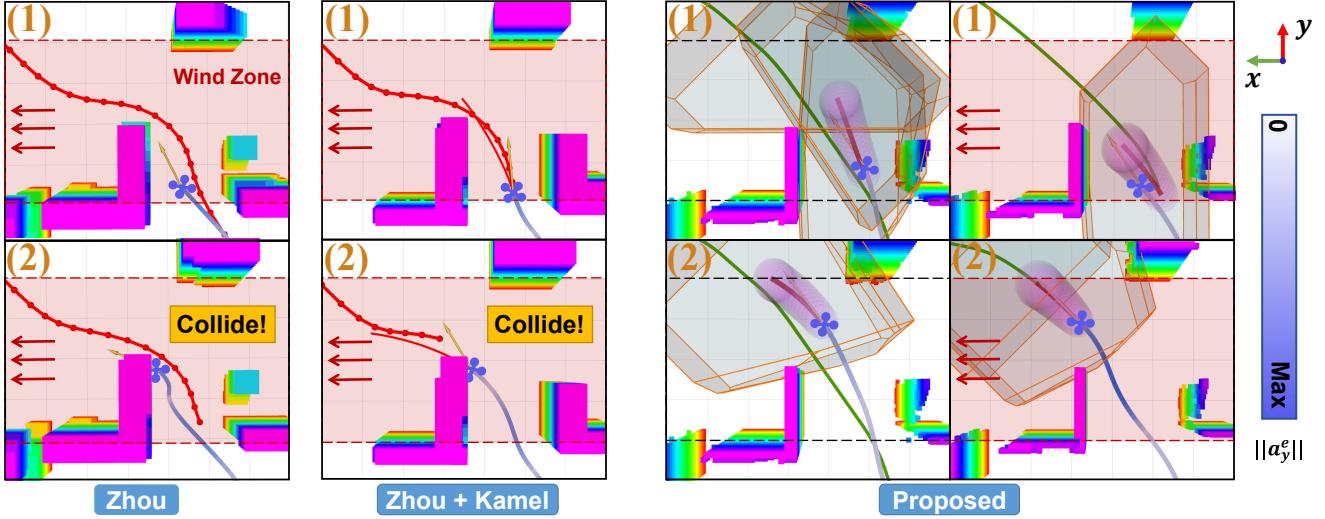


Fig. 6. The snapshots of the simulation comparison of different methods. The sub-figure(1) shows when the quadrotor enters the wind zone, while (2) is a snapshot of when the quadrotor leaves the zone. The odometry is colored with the norm of command acceleration in the y-axis a_y^e . When flying through the wind zone (the red area), the quadrotor suffers a constant mass-normalized external force $\mathcal{F}^e = [0.0, 2.0, 0.0]^T$. In Zhou's method, the quadrotor does not consider the deviation of its position to the original planning trajectory due to external force. Together with Kamel's NMPC controller, it smooths the impact of an external force, but the EKF-based estimator has some delay for the quadrotor to react. The right pictures are our method without and with a wind zone. It indicates that if the quadrotor encounters the external force, a_y^e will increase for disturbance resistance.

problems. Since they do not provide open source applicable code for real-time solving, we re-implement it using Forces Pro and provide available information of obstacles and wind zone for benchmark comparison. We define “success rate” as the rate of successful flights without collision and the control cost evaluated by squares of commanded accelerations. The simulation environment is set as Fig. 5(a) with a global path throughout a fixed wind zone. When the quadrotor enters the range of the wind zone, it suffers from a mass-normalized external force in the y-axis to simulate different extents of the winds.

TABLE I
COMPARISON OF REFERENCE TRACKING

\mathcal{F}^e (m/s ²)	Method	Succ. Rate	Traj. Time (s)	Ctrl. Cost (m ² /s ³)
$[0.0, 1.5, 0.0]^T$	Kamel et al. [4]	1.00	15.67	18.11
	Hoseong et al. [10]	1.00	15.73	19.03
	Proposed	1.00	14.89	16.34
$[0.0, 2.5, 0.0]^T$	Kamel	-	-	-
	Hoseong	-	-	-
	Proposed	0.60	15.02	27.38

The results are shown in Tab. I. All MPC methods can successfully finish the flight test if the wind is within the variance range and reaction window for tracking. When the external force surpasses the bound, their methods fail to handle the disturbance while following the trajectory.

2) *Comparison of Planning Framework:* Secondly, we compare our method against a state-of-the-art local planner by Zhou [25] under different external forces for real-time systems. We also test the disturbance-aware controller [4] using planning trajectory from Zhou's method as a reference to assess the robustness of this controller. The scenario is as

the map in Fig. 5 (2). Within the wind zone, the quadrotor is under a constant force with different values and directions.

TABLE II
COMPARISON OF PLANNING FRAMEWORK

\mathcal{F}^e (m/s ²)	Method	Succ. Rate	Traj. Time (s)	Ctrl. Cost (m ² /s ³)
$[0.0, 0.5, 0.0]^T$	Zhou et al. [25]	0.70	7.65	14.65
	Zhou + Kamel	0.50	10.93	19.18
	Proposed	1.00	8.33	13.71
$[0.0, 1.0, 0.0]^T$	Zhou	0.60	8.80	14.70
	Zhou + Kamel	0.40	12.67	19.33
	Proposed	1.00	8.17	16.81
$[2.0, 1.0, 0.0]^T$	Zhou	0.60	8.29	18.86
	Zhou + Kamel	0.45	9.95	20.01
	Proposed	1.00	8.27	16.37
$[0.0, 2.0, 0.0]^T$	Zhou	-	-	-
	Zhou + Kamel	0.25	9.04	19.13
	Proposed	0.90	8.86	12.78

As the Tab. II illustrates, when the external force is sufficiently small, the controller can resist such influence to a certain degree. As the external force increases, the local planner, without consideration of force disturbance, has a higher collision probability. A simulation comparison instance under mass-normalized external force equal to $[0.0, 2.0, 0.0]^T$ is shown in Fig. 6. The local planner, without force consideration, cannot recompense the trajectory offset in the y-axis resulting from an external force. Furthermore, with a disturbance-aware observer, the robust controller cannot give real-time feedback to the planner to adjust the trajectory because the current reference is already infeasible. The EKF based disturbance observer that is applied by the controller smooths the external force. Hence, the controller performs delayed reactions to suddenly imposed force and sometimes adversely increases the collision risk. For other cases in Tab.

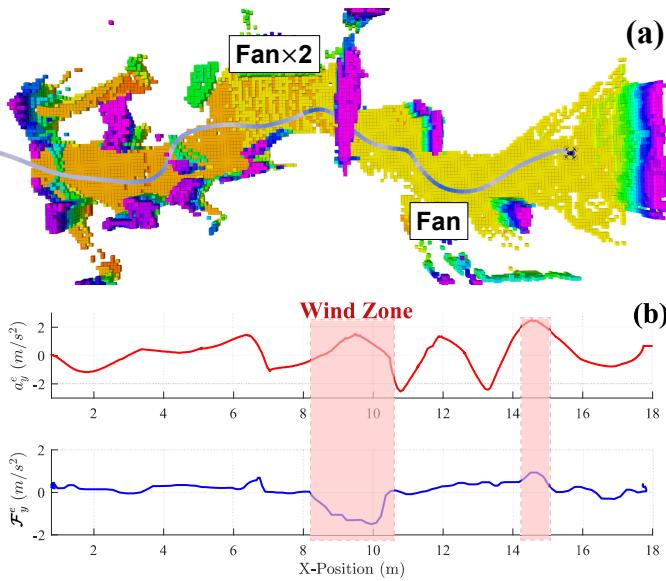


Fig. 7. Real-world indoor test. (a) The overall trajectory of the indoor flight experiment, colored with the norm of command acceleration in the y-axis. (b) The variance of \mathcal{F}_y^e and a_y^e with x-position.

II, when the direction of external force is not along with the main trends of the quadrotor’s motion, the safe and successful flight will cost more energy. Otherwise, the quadrotor can take advantage of the force.

C. Real-World Tests

1) *Indoor Flights Through Wind Zones*: We present several indoor experiments in cluttered environments with strong winds in Fig. 1. The quadrotor flies through an unknown narrow hallway fully autonomously with onboard computation and limited field of view (FOV) sensors. Several fans are set up near the possible flight paths, providing the wind zone for external disturbances. The speed of wind zones caused by fans ranges from 6.5 m/s to 7.4 m/s, measured by an anemometer. The wind causes the maximum mass-normalized external force at around 2.5 m/s² to the quadrotor based on experiment data, measured by VID-Fusion [9]. The trajectory of a flight test through two different wind zones is shown in Fig. 7. When the quadrotor enters wind zones, the increasing estimated external force triggers its replanning, which results in the variation of acceleration in the y-axis and adjustment of starting predicted state in back-end optimization.

2) *Indoor Benchmark Flights Comparison*: In this section, we test Zhou’s planner [25] in the same indoor flight environment mentioned above, where several wind zones are presented. The results of command positions and estimated positions during one flight experiment are shown in Fig. 8. After entering the wind zone, the planner fails to compensate position deviations and continues to follow an intractable trajectory, then collides with obstacles.

3) *Indoor Flights With an Unmodeled Load*: Flying with an object of unknown mass is also an essential application of delivery for quadrotors. We add a package to the quadrotor without modeling of complete dynamic model or measurement of a loaded object. A quadrotor (1.13 kg) with a loaded

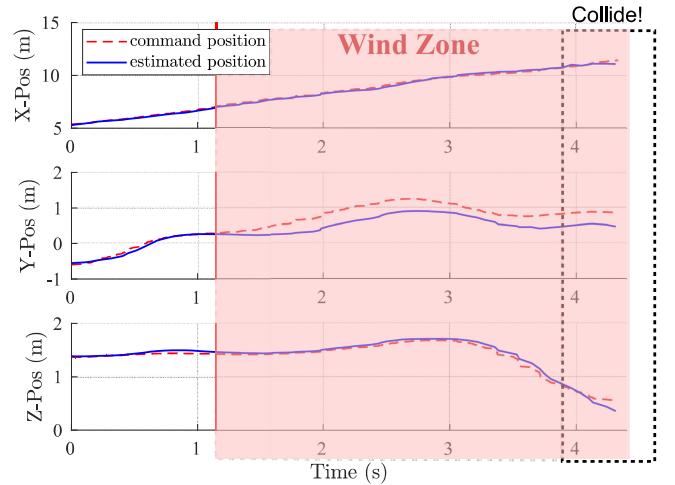


Fig. 8. Benchmark flight test for comparison. The blue curve is the estimated position by VINS, and the red dotted curve is the command send by Zhou’s planner over time. The red vertical marked that the quadrotor is entering a wind zone.

bottle (0.19 kg) is experimentally tested in a cluttered environment. When the quadrotor is hovering, the gravity of this load causes an equivalent mass normalized force acceleration $a_{load} = 1.65 \text{ m/s}^2$ in the z-axis.

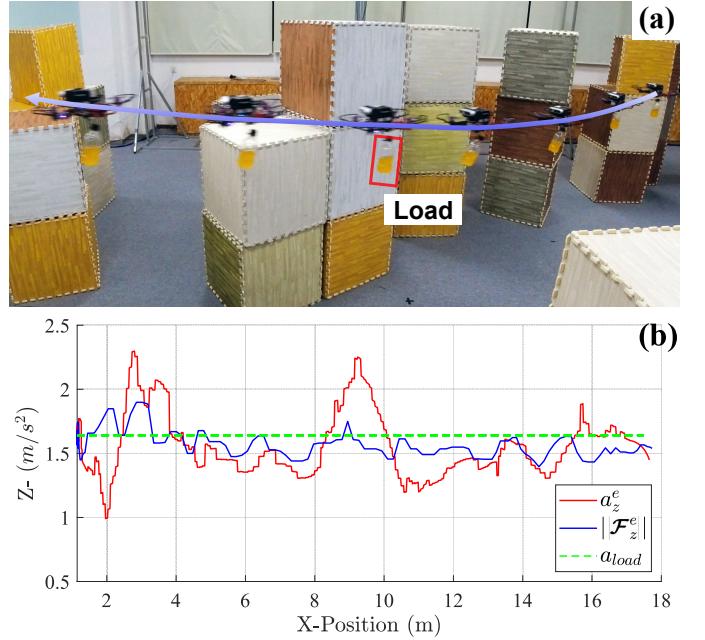


Fig. 9. Flight test with an unmodeled load. (a) Snapshots of indoor flight test with a loaded bottle. (b) The variance of \mathcal{F}_z^e and a_z^e with x-position, compared with gravity effect of the load.

The snapshot is shown in Fig. 9 (1), and the estimated external force and command acceleration in the z-axis of the quadrotor is shown Fig. 9 (2). It illustrates that the planning acceleration commands sent by our planner compensate for the influence of external force, which ensures a stable trajectory in the z-direction. The value of the external force matches the gravity effect of the loaded bottle and slightly varies during the whole flight.

In our experiment, the load is connected to the quadrotor with a short string so that the pendulum effect can be ne-

glected in path planning. If the string of a suspended load is considered, the planner needs to model its dynamics and adds obstacles avoidance of the load.

4) *Outdoor Flights Under Natural Winds:* To validate the robustness and efficiency of the proposed planning framework under natural winds, we conduct outdoor experiments in a cluttered forest with wind speed up to 3.1 m/s. Given several targets inside the forest, the drone generates a smooth trajectory to follow the targets against the natural wind gusts and other uncertain disturbances from leaves, as shown in Fig. 10. It reaches a maximum velocity around 2.5 m/s. The outdoor flight test shows that our proposed framework has the capability of autonomous navigation in a completely unknown and complex environment with natural disturbance.

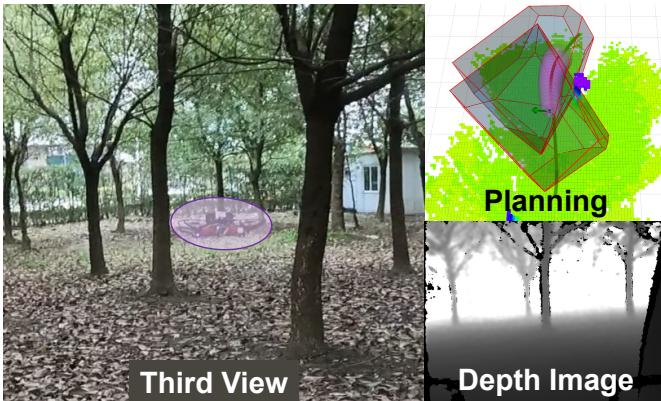


Fig. 10. The snapshot of the outdoor flight test.

VI. CONCLUSION

This paper presents a systematic (re)planning framework for a quadrotor autonomous system in the presence of fierce external forces. We make a step forward to consistently include the external force in both the planning and tracking systems by incorporating the force in front-end path searching and back-end NMPC optimization. Using the ellipsoidal approximation of error FRSs and quadrotor's inflated geometrical shape, we can guarantee collision avoidance by constraining the safe ellipsoid boundaries into a flight corridor along the reference path. The proposed approach can achieve real-time adaptive planning without offline computation of FRSs or the pre-computed collision-free reference trajectory. The benchmark comparisons in both simulations and real-world tests illustrate the necessity and adaptability of the proposed framework with external force considerations. Furthermore, the indoor flights with different external forces and outdoor flights with natural winds validate the robustness of the proposed method under different environments.

REFERENCES

- [1] J. A. Guerrero, J. A. Escareno, and Y. Bestaoui, “Quad-rotor mav trajectory planning in wind fields,” in *Proc. of the IEEE Int'l. Conf. on Robot. and Autom.*, 2013, pp. 778–783.
- [2] K. Cole and A. M. Wickenheiser, “Reactive trajectory generation for multiple vehicles in unknown environments with wind disturbances,” *IEEE Trans. Robot.*, vol. 34, no. 5, pp. 1333–1348, 2018.
- [3] T. G. McGee and J. K. Hedrick, “Path planning and control for multiple point surveillance by an unmanned aircraft in wind,” in *Proc. of the American Control Conf.*, 2006, pp. 4261–4266.
- [4] M. Kamel, M. Burri, and R. Siegwart, “Linear vs nonlinear mpc for trajectory tracking applied to rotary wing micro aerial vehicles,” *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 3463–3469, 2017, 20th IFAC World Congress.
- [5] S. Singh, A. Majumdar, J. Slotine, and M. Pavone, “Robust online motion planning via contraction theory and convex optimization,” in *Proc. of the IEEE Int'l. Conf. on Robot. and Autom.*, 2017, pp. 5883–5890.
- [6] J. Ji, X. Zhou, C. Xu, and F. Gao, “Cmpec: Corridor-based model predictive contouring control for aggressive drone flight,” in *Proc. of the Int'l. Sym. on Exp. Robot.* Springer International Publishing, 2021, pp. 37–46.
- [7] M. Otte, W. Silva, and E. Frew, “Any-time path-planning: Time-varying wind field + moving obstacles,” in *Proc. of the IEEE Int'l. Conf. on Robot. and Autom.*, 2016, pp. 2575–2582.
- [8] B. Zhou, F. Gao, L. Wang, C. Liu, and S. Shen, “Robust and efficient quadrotor trajectory generation for fast autonomous flight,” *IEEE Robot. Autom. Lett.*, vol. 4, no. 4, pp. 3529–3536, 2019.
- [9] Z. Ding, T. Yang, K. Zhang, C. Xu, and F. Gao, “Vid-fusion: Robust visual-inertial-dynamics odometry for accurate external force estimation,” *arXiv preprint arXiv:2011.03993*, 2020.
- [10] H. Seo, D. Lee, C. Y. Son, C. J. Tomlin, and H. J. Kim, “Robust trajectory planning for a multirotor against disturbance based on hamilton-jacobi reachability analysis,” in *Proc. of the IEEE/RSJ Int'l. Conf. on Intell. Robots and Syst.*, 2019, pp. 3150–3157.
- [11] F. Ruggiero, J. Cacace, H. Sadeghian, and V. Lippiello, “Impedance control of vtol uavs with a momentum-based external generalized forces estimator,” in *Proc. of the IEEE Int'l. Conf. on Robot. and Autom.*, 2014, pp. 2093–2099.
- [12] L. Palmieri, T. P. Kucner, M. Magnusson, A. J. Lilenthal, and K. O. Arras, “Kinodynamic motion planning on gaussian mixture fields,” in *Proc. of the IEEE Int'l. Conf. on Robot. and Autom.*, 2017, pp. 6176–6181.
- [13] B. Yüksel, C. Secchi, H. H. Bülthoff, and A. Franchi, “A nonlinear force observer for quadrotors and application to physical interactive tasks,” in *IEEE/ASME Int'l. Conf. on Advanced Intelligent Mechatronics*, 2014, pp. 433–440.
- [14] T. Qin, P. Li, and S. Shen, “Vins-mono: A robust and versatile monocular visual-inertial state estimator,” *IEEE Trans. Robot.*, vol. 34, no. 4, pp. 1004–1020, 2018.
- [15] M. Kamel, J. Alonso-Mora, R. Siegwart, and J. Nieto, “Robust collision avoidance for multiple micro aerial vehicles using nonlinear model predictive control,” in *Proc. of the IEEE/RSJ Int'l. Conf. on Intell. Robots and Syst.*, 2017, pp. 236–243.
- [16] M. Bangura and R. Mahony, “Real-time model predictive control for quadrotors,” *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 11 773–11 780, 2014, 19th IFAC World Congress.
- [17] F. Gao, L. Wang, B. Zhou, X. Zhou, J. Pan, and S. Shen, “Teach-Repeat-Replan: A complete and robust system for aggressive flight in complex environments,” *IEEE Trans. Robot.*, vol. 36, no. 5, pp. 1526–1545, 2020.
- [18] S. Liu, M. Waterson, K. Mohta, K. Sun, S. Bhattacharya, C. J. Taylor, and V. Kumar, “Planning dynamically feasible trajectories for quadrotors using safe flight corridors in 3-d complex environments,” *IEEE Robot. Autom. Lett.*, vol. 2, no. 3, pp. 1688–1695, 2017.
- [19] X. Zhong, Y. Wu, D. Wang, Q. Wang, C. Xu, and F. Gao, “Generating large convex polytopes directly on point clouds,” *arXiv preprint arXiv:2010.08744*, 2020.
- [20] S. Kim, D. Falanga, and D. Scaramuzza, “Computing the forward reachable set for a multirotor under first-order aerodynamic effects,” *IEEE Robot. Autom. Lett.*, vol. 3, no. 4, pp. 2934–2941, 2018.
- [21] A. B. Kurzhanski, “Ellipsoidal calculus for estimation and feedback control,” in *Systems and Control in the Twenty-First Century*. Birkhäuser Boston, 1997, pp. 229–243.
- [22] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge University Press, 2004.
- [23] A. Domahidi and J. Jerez, “Forces professional,” Embotech AG, url=<https://embotech.com/FORCES-Pro>, 2014–2019.
- [24] F. Furrer, M. Burri, M. Achtelik, and R. Siegwart, *Robot Operating System (ROS): The Complete Reference (Volume 1)*. Springer International Publishing, 2016, ch. RotorS—A Modular Gazebo MAV Simulator Framework, pp. 595–625.
- [25] X. Zhou, Z. Wang, C. Xu, and F. Gao, “Ego-planner: An esdf-free gradient-based local planner for quadrotors,” *IEEE Robot. Autom. Lett.*, vol. 6, pp. 478–485, 2021.