Fault Tolerant Motion Planning for a Quadrotor subject to Complete Rotor Failure

Youngim Nam[a,**], Kangmin Lee[a,**], Hyo-Sang Shin[b,c], Cheolhyeon Kwon[a,*]

[a]Department of Mechanical Engineering, Ulsan National Institute of Science and Technology, Ulsan, South Korea
[b]Institute of Aerospace Sciences, Cranfield University, Cranfield, United Kingdom
[c]Cho Chun Shik Graduate School of Mobility, Korea Advanced Institute of Science and Technology, Daejeon, South Korea

---

Abstract

Assuring safety of a quadrotor subject to rotor failure has been heavily investigated at the control level in view of fault tolerant control (FTC) approach. Yet, the existing FTCs are often concerned with tracking the reference motion even when that reference may not be safely trackable due to the physical constraints of the quadrotor. This paper tackles the faulty quadrotor safety at the planner level, proposing a fault tolerant motion planner. Starting from the formal backward reachability problem formulation, the proposed motion planner generates the time trajectory of the coupled rotational and translational motions that safely guide the faulty quadrotor. The generated trajectory is theoretically guaranteed to be tracked by the embedded FTC without violating the physical constraints. Further, the trajectory is prescribed as an analytical closed-form expression and thus suitable for real-time emergency maneuvers. The effectiveness of the proposed motion planner is numerically validated in conjunction with the different FTC techniques and compared to the existing planning method. The simulation results clearly signify that the proposed planner can successfully complement the fault tolerance of quadrotor. The supplements including detailed proofs of the theorems and code implementations are available on GitHub repository: https://github.com/HMCL-UNIST/Fault-tolerant-motion-planner.

Keywords: Quadrotor, Fault tolerant control, Fault tolerant motion planning, Rotor failure, Rotor saturation.

---

1. Introduction

Quadrotor has proven its versatility over diverse applications, which is mainly attributed to simple structure, ease-of-operation, and cost-effectiveness compared to fixed-wing types and its multi-(hexa-/octo-)rotor relatives. However, such advantages come at the risk against potential faults and failures. In particular, having only four rotors to fly brings out great vulnerability as it is prone to crash even under a single rotor failure [1]. To address the safety of the quadrotor subject to rotor failure, the majority of research has been conducted to develop control algorithms, namely fault tolerant control (FTC), that can tolerate the rotor failure and eventually assure the safety of the faulty quadrotor [2]. While the fundamentals of FTC are quite mature, the FTC for quadrotors is yet challenging due to the highly nonlinear dynamics, under-actuated control mechanism, physical constraints imposed on the rotors, etc [3].

The existing FTC work toward the quadrotor applications can be roughly divided into passive and active approaches [4]. The passive FTCs proactively design controllers to be robust against the potential faults [5], whereas the active FTCs reconfigure the controller online in response to the detected fault information [6]. These two approaches have often been complemented with each other, such as gain scheduling techniques that enable switching (i.e., active) to the controller pre-tuned (i.e., passive) for the specific quadrotor fault [7]. Further, different control techniques have been employed to recover the faulty quadrotor, e.g., sliding mode control (SMC) [8], backstepping control (BSC) [9], adaptive control [10], linear quadratic regulator [11], etc. These works have considered the limited type of rotor failures, i.e., the partial loss, under which the faulty quadrotor is still controllable since the failed rotor can exert degenerate thrust. This is not the case when the rotor completely fails, as it makes the quadrotor not fully controllable [12]. Therefore, the complete rotor loss is arguably more daunting than the partial loss cases, needing substantially different FTC mechanisms, and thus considered to be the matter of this paper.

Several FTC studies have addressed the complete rotor loss by utilizing the three remaining rotors. The underlying idea is to discard controlling a portion of quadrotor states, for which the yaw state is usually abandoned [13]. This is due to the fact that the quadrotor yaw angle is relatively less significant in a hovering maneuver, the merest impact for safety. Based on this

scheme, the controller can appropriately reconfigure the thrust of three rotors thereby stabilizing the quadrotor through different FTC techniques, such as PD control [14], SMC [12], BSC [15], feedback linearization control (FLC) [16], etc. Besides, [17, 18] have further examined the stability under the complete rotor loss by establishing the periodic equilibrium points within the reduced attitude space. However, many FTC approaches for complete rotor loss may not be safe for practical deployment, often due to insufficient consideration of physical constraints, i.e., rotor thrust saturation. Without rigorous consideration of the saturation constraints in the remaining three rotors, the FTC tends to issue physically infeasible commands (e.g., reversing thrust to the downside). To address this issue, a few FTC studies have resorted to optimization-based methods such as model predictive control (MPC), which explicitly account for saturation constraints [19, 20]. Nevertheless, the optimization-based methods are computationally burdensome, making it difficult to handle real-time commands during emergent faulty situations.

Apart from the complication caused by the rotor saturation, the faulty quadrotor should perform more than just hovering in certain circumstances. Specifically in densely populated or obstacle-rich environments, the quadrotor is desired to safely return to the designated safe zone even when the complete loss of a rotor occurs. Using only three rotors, such a task is challenging if not impossible, and is somewhat beyond the controller level whose primary focus is to follow the predetermined trajectories [21, 22] (See the red dotted trajectory in Fig. 1). Accordingly, there needs an effort at the planner level, to plan the reference trajectory that guides the quadrotor to the safe zone and further eases the FTC with rotor saturation. This idea is well aligned with the reference governor [23, 24], which discusses the way to adjust the reference command with respect to the system dynamics and constraints. Building upon this, trajectory replanning strategies [25, 26] have been developed. However, these approaches typically rely on numerical optimization, their computation burden could be critical when necessitating real-time command at the emergent faulty situation. On the other hand, some recent approaches have attempted the integrated fault tolerant planning-control with reduced computation complexity, such as sampling-based [27, 28] or learning-based [29, 30] methods. Nonetheless, they lack the stringent guarantee of safety and have not been situated in the complete rotor loss of quadrotors.

In a nutshell, a stand-alone FTC may not suffice for the safe maneuver of the faulty quadrotor, and the appropriate planner can complement the FTC. Despite the rich literature for either the controller or planner level, to the best of the authors' knowledge, no apparent work has been carried out the fault tolerance of the quadrotor subject to the complete rotor loss, considering both the rotor saturation and the safety concerns in terms of the reachability to the safe zone. To this end, this paper proposes a fault tolerant planner that strives to simultaneously accomplish the two objectives: generating the reference trajectory that can be safely tracked by the embedded FTC even with the rotor saturation constraints; and eventually reach the designated goal, i.e., safe zone while staying away from the unsafe area (See the blue dotted trajectory in Fig. 1).
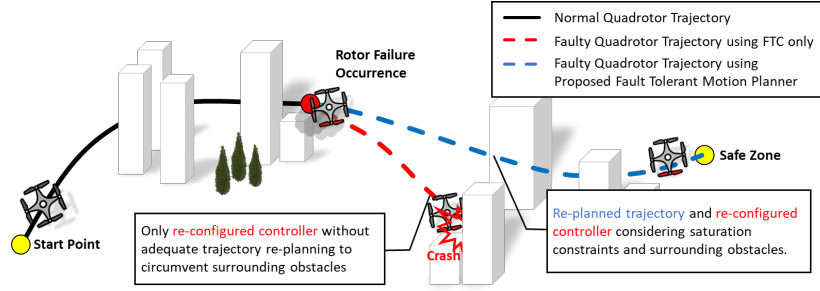

Figure 1: Illustration of faulty quadrotor maneuvers when navigating in obstacle-rich environments with and without the proposed motion planner.

Inspired by the conventional cascaded structure of the quadrotor control architecture, the proposed planner consists of two phases, rotational motion planning and translational motion planning, and thus is named as a fault tolerant *motion* planner. Then, each planning phase tackles the faulty quadrotor behavior through mathematically well-posed inequality constraints, and respectively generates the attitude and position trajectories that are trackable by the embedded FTC without violating the rotor saturation. Once the FTC tracking performance with respect to the reference trajectory is granted, one can fulfill the safety goal by simply setting the end of the trajectory into the safe zone. The feasibility of such trajectory generation is formally addressed based on backward reachability, whereby the condition for the existence of the safe trajectory is derived. This condition indeed implies the theoretical guarantee of the proposed motion planner assuring the safety of the quadrotor.

4

The proposed fault tolerant motion planner is the first-ever work that encompasses: (i) faulty quadrotor dynamics subject to a complete rotor loss; (ii) the capacity of the embedded FTC with rotor saturation constraints; and (iii) the common safety objective at planner level such as return to the safe zone and obstacle avoidance. The contributions of this paper are remarked as follows:

- We develop a fault tolerant motion planner based on the formalism of backward reachability and thereby theoretically verify the safety of the faulty quadrotor.
- The motion trajectory is implemented as an analytical closed-form solution, which is suitable for real-time computation in emergent quadrotor maneuvers.
- The proposed planner possesses great compatibility in the sense that it is agnostic to most existing FTC schemes and can readily be adjoined to other path planners.
- The effectiveness of the proposed planner is comprehensively validated through different simulations, including the comparison between different FTC schemes (e.g., PD, SMC, and BSC) with/without the proposed planner while demonstrating the different scenarios (e.g., hovering, waypoint tracking and obstacle avoidance).

The rest of this paper is organized as follows: Section 2 presents the preliminary backgrounds which include quadrotor dynamics, rotor failure and saturation constraints, and the baseline FTC scheme. Section 3 mathematically formulates the faulty quadrotor safety problem and additionally explains the stability characteristics of the faulty quadrotor. Based on this, Section 4 develops a fault tolerant motion planner which divides into rotational motion and translational motion, respectively in Subsections 4.1 and 4.2. The numerical simulations and comparative analyses of the developed planner are carried out in Section 5. The conclusions and future work are given in Section 6.

2. Preliminaries

2.1. Dynamical Model of Quadrotor

The typical configuration of a quadrotor that consists of four rotors, each denoted by R_i, $i \in \{1, 2, 3, 4\}$, is illustrated in Fig. 2. Here, two diagonally opposite rotors R_1 and R_3 (R_2 and R_4)
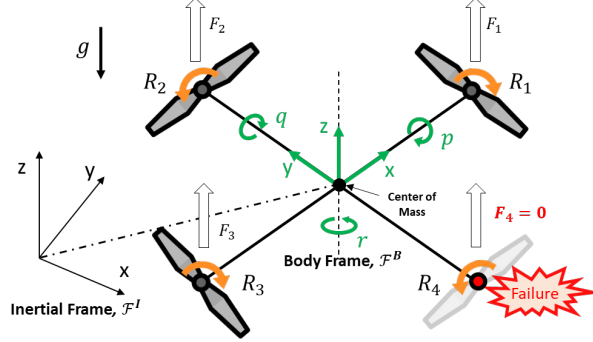
5

Figure 2: Typical quadrotor configuration with rotor failure.

rotate in the clockwise (counterclockwise) direction, each generating the upward thrust F_1 and F_3 (F_2 and F_4) respectively.

Two reference frames are considered to describe the motion of a quadrotor: the inertial frame $\mathcal{F}^I$ fixed to the world map, and the body frame $\mathcal{F}^{\mathcal{B}}$ fixed to the quadrotor's center of gravity. The translation of the quadrotor is described by its center position, denoted by $\{x(t), y(t), z(t)\}$, with respect to the inertial frame $\mathcal{F}^I$. For the rotation of the quadrotor, the orientation of the body frame $\mathcal{F}^{\mathcal{B}}$ is represented by Euler angles $\{\phi(t), \theta(t), \psi(t)\}$, respectively representing the roll, pitch, and yaw motions of the quadrotor. For brevity, time index t will be omitted from variables when referring to function values, unless it causes confusion. Then, the rotation of $\mathcal{F}^{\mathcal{B}}$ with respect to $\mathcal{F}^I$ is expressed by the rotation matrix $\mathcal{R}_{\mathcal{B}}^{I} \in \mathcal{SO}(3)$, given by:

$$\mathcal{R}_{\mathcal{B}}^{I} := \begin{bmatrix} C_\theta C_\psi & S_\phi S_\theta C_\psi - C_\phi S_\psi & C_\phi S_\theta C_\psi + S_\phi S_\psi \\ C_\theta S_\psi & S_\phi S_\theta S_\psi + C_\phi C_\psi & C_\phi S_\theta S_\psi - S_\phi C_\psi \\ -S_\theta & S_\phi C_\theta & C_\phi C_\theta \end{bmatrix} \tag{1}$$

where $S_{(\cdot)}$ and $C_{(\cdot)}$ denote $\sin(\cdot)$ and $\cos(\cdot)$, respectively. By this rotation matrix, the angular velocity with reference to the body frame $\mathcal{F}^{\mathcal{B}}$, denoted by $\{p, q, r\}$, is related to the rate of Euler

6

angles $\{\dot{\phi}, \dot{\theta}, \dot{\psi}\}$ as follows:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin(\phi)\tan(\theta) & \cos(\phi)\tan(\theta) \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \frac{\sin(\phi)}{\cos(\theta)} & \frac{\cos(\phi)}{\cos(\theta)} \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix}. \tag{2}$$

Based on the kinematics (1) and (2), the following nonlinear quadrotor dynamics is obtained using the Newton-Euler formula [16]:

$$
\begin{aligned}
\dot{x} &= v_x, & \dot{v}_x &= (-k_t v_x + (\cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi)u_f)/m, \\
\dot{y} &= v_y, & \dot{v}_y &= (-k_t v_y + (\cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi)u_f)/m, \\
\dot{z} &= v_z, & \dot{v}_z &= (-k_t v_z - mg + (\cos\phi\cos\theta)u_f)/m, \\
\dot{\phi} &= p + q\sin\phi\tan\theta + r\cos\phi\tan\theta, & \dot{p} &= (-(I_z - I_y)qr + u_p)/I_x, \\
\dot{\theta} &= q\cos\phi - r\sin\phi, & \dot{q} &= (-(I_x - I_z)pr + u_q)/I_y, \\
\dot{\psi} &= (q\sin\phi + r\cos\phi)/\cos\theta, & \dot{r} &= (-k_r r + u_r)/I_z
\end{aligned}
\tag{3}
$$

where m represents the mass of quadrotor; g stands for the gravitational acceleration; and I_x, I_y, and I_z are the moments of inertia for the corresponding axis of the body frame. Their values are usually set to $I_z > I_x = I_y$ according to the common symmetric shape of the quadrotor. Considering air resistance, k_t and k_r are the drag coefficients associated with the translation and the rotation of the quadrotor. To manifest the proposed idea, we consider the deterministic system dynamics in this paper, assuming that k_t is identical along all translation directions. The rotational drags are negligible except for the yaw motion, and the uncertainties such as disturbance noise are disregarded [31]. Extension to stochastic dynamics with realistic uncertainty factors will be included in the future work. As seen in (3), the quadrotor is an under-actuated system controlled by four inputs, i.e., the torques $u_p, u_q,$ and u_r for each rotation axis of $\mathcal{F}^{\mathcal{B}}$, and the total thrust u_f to the upward direction of $\mathcal{F}^{\mathcal{B}}$. These inputs are governed by the individual rotor

thrusts through the following control allocation matrix:

$$
\begin{bmatrix} u_f \\ u_p \\ u_q \\ u_r \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & l & 0 & -l \\ -l & 0 & l & 0 \\ b & -b & b & -b \end{bmatrix} \begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{bmatrix} \tag{4}
$$

where l is the distance between the rotor and the center of quadrotor; and b is the ratio between the drag and thrust coefficients of the propeller.

2.2. Rotor Failure and Saturation Constraints

Different quadrotor failures can be thought of depending on the type of rotor failure (partial loss [6] or complete loss [14]) and the number of failed rotors (from one to four). In this paper, we focus on the 'complete loss' occurring on a single rotor which is in general more challenging than the partial loss case. Furthermore, the partial loss of effectiveness can be treated as the complete loss by simply turning off the failed rotor, allowing the remedies for complete loss to be applicable for the partial loss cases as well. Simultaneous failures on the multiple rotors are more delicate than the single rotor failure, and thus are included in our future work.

Suppose the complete loss occurs at the fourth rotor R_4 so that $F_4 = 0$ (See Fig. 2). The control allocation mechanism (4), with only three healthy rotors, cannot carry out the full controllability of the quadrotor. From the stability point of view, the yaw motion of quadrotor is relatively less significant than the roll and pitch motions, so the control allocation matrix is usually reconfigured into the following reduced control space [13]:

$$
\begin{bmatrix} u_f \\ u_p \\ u_q \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & l & 0 \\ -l & 0 & l \end{bmatrix} \begin{bmatrix} F_1 \\ F_2 \\ F_3 \end{bmatrix}. \tag{5}
$$

The yaw torque u_r is then determined by the following combination of u_f and u_p:

$$u_r = b(u_f - \frac{2}{l} u_p). \tag{6}$$

In practice, the remaining three rotors R_i, $i \in \{1, 2, 3\}$ are further subject to physical constraints on their maximum and minimum thrust values, such as $0 \leq F_i \leq \overline{F}$, $i \in \{1, 2, 3\}$. Applying these constraints to (5) forms the following saturation constraints on the total thrust and torques:

$$0 \leq u_p \leq l\overline{F}, \tag{7a}$$

$$u_p \leq u_q + lu_f \leq u_p + 2l\overline{F}, \tag{7b}$$

$$-u_p - 2l\overline{F} \leq u_q - lu_f \leq -u_p. \tag{7c}$$

In reflection of (5), (6) and (7) into (3), the faulty quadrotor dynamics can be compactly written by:

$$\dot{\mathbf{x}} = \mathcal{F}(\mathbf{x}, \mathbf{u}), \quad \mathbf{x} \in \mathbf{X}, \quad \mathbf{u} \in \mathbf{U} \tag{8}$$

where $\mathbf{x} = [x\ y\ z\ v_x\ v_y\ v_z\ \phi\ \theta\ \psi\ p\ q\ r]^{\mathrm{T}}$ and $\mathbf{u} = [u_f\ u_p\ u_q]^{\mathrm{T}}$. Here, $\mathbf{X} \subset \mathbb{R}^{12}$ and $\mathbf{U} \subset \mathbb{R}^3$ are respectively the set of all the feasible states and the set of all the admissible system input complying with the saturation constraints (7).

2.3. Fault Tolerant Control against Complete Rotor Loss

The primal goal of the FTC for the complete loss of a rotor is to stabilize the roll, pitch, and altitude of the faulty quadrotor, i.e., ϕ, θ, and z. Subsequently, the existing FTC work mostly focuses on the subsystem of (8) including only attitude and altitude states, which can be written by the following control-affine nonlinear dynamics:

$$\dot{X} = f(X) + h(X)\mathbf{u} \tag{9}$$

9

where $\mathcal{X} = [\phi \; \theta \; p \; q \; r \; z \; v_z]^{\mathrm{T}}$, and f and h are respectively defined by:

$$f(\mathcal{X}) := \begin{bmatrix} p + q \sin\phi \tan\theta + r\cos\phi \tan\theta \\ q\cos\phi - r\sin\phi \\ -(I_z - I_y)qr/I_x \\ -(I_x - I_z)pr/I_y \\ -k_r r/I_z \\ v_z \\ (-k_t v_z)/m - g \end{bmatrix}, \; h(\mathcal{X}) := \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1/I_x & 0 \\ 0 & 0 & 1/I_y \\ b/I_z & -2b/(lI_z) & 0 \\ 0 & 0 & 0 \\ (\cos\phi\cos\theta)/m & 0 & 0 \end{bmatrix}.$$

Let us further define the function L for $\{\dot{\phi}, \dot{\theta}, \dot{z}\}$,

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{z} \end{bmatrix} = L(\mathcal{X}) := \begin{bmatrix} p + q\sin\phi\tan\theta + r\cos\phi\tan\theta \\ q\cos\phi - r\sin\theta \\ v_z \end{bmatrix}. \tag{10}$$

Using (9) and (10), the second-order time derivative of $\{\phi, \theta, z\}$ can be expressed by:

$$\begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{z} \end{bmatrix} = \frac{dL(\mathcal{X})}{dt} = \frac{\partial L(\mathcal{X})}{\partial \mathcal{X}} f(\mathcal{X}) + \frac{\partial L(\mathcal{X})}{\partial \mathcal{X}} h(\mathcal{X})\mathbf{u}. \tag{11}$$

Now, let $\mathbf{u}^{\mathrm{FTC}}$ be the system input designed by the common FTC law in light of nonlinear dynamic inversion (NDI) [21]. Its underlying expression is given by:

$$\mathbf{u}^{\mathrm{FTC}} = [J(\mathcal{X})h(\mathcal{X})]^{-1}(-J(\mathcal{X})f(\mathcal{X}) + v) \tag{12}$$

where $J(\mathcal{X}) := \frac{\partial L(\mathcal{X})}{\partial \mathcal{X}}$ is the Jacobian matrix. The NDI approaches are particularly effective in handling the nonlinearities and compatible with most existing control laws for quadrotors, and thus widely adopted in the quadrotor FTC techniques such as PD control [14], SMC [12], and BSC [15]. v is the auxiliary control input designed by these FTC schemes and the base structure

10

of ν can be represented by:

$$
\nu = \begin{bmatrix} \ddot{\phi}_d \\ \ddot{\theta}_d \\ \ddot{z}_d \end{bmatrix} + K_1 \begin{bmatrix} \dot{\phi}_d - \dot{\phi} \\ \dot{\theta}_d - \dot{\theta} \\ \dot{z}_d - \dot{z} \end{bmatrix} + K_2 \begin{bmatrix} \phi_d - \phi \\ \theta_d - \theta \\ z_d - z \end{bmatrix} \tag{13}
$$

where $\{\phi_d, \theta_d, z_d\}$ are the desired reference motion and K_1, $K_2 > 0$ are appropriate gains. In most practice, $J(X)h(X)$ is invertible and (12) gives a nonsignular solution [16].

Lemma 1. *Let* $e_\phi := \phi_d - \phi$, $e_\theta := \theta_d - \theta$, $e_z := z_d - z$ *be the tracking error for the faulty quadrotor* (8). *Suppose the FTC is implemented according to* (12) *and* (13). *Then, given a smooth[1] time trajectory of* $\{\phi_d(t), \theta_d(t), z_d(t)\}$, *the resulting tracking error vector* $\mathbf{e} := [e_\phi\ e_\theta\ e_z]^{\mathrm{T}}$ *is asymptotically stable in the sense of Lyapunov if* $\mathbf{u}^{\mathrm{FTC}}(t) \in \mathbf{U}$, $\forall t \geq 0$.

Proof. See Appendix A in our supplementary document [32].

Note that the saturation constraints (7) are not respected in (12), and thus the generated $\mathbf{u}^{\mathrm{FTC}}$ often violates (7), i.e., $\mathbf{u} \neq \mathbf{u}^{\mathrm{FTC}}$ which may cause instability of the faulty quadrotor. This motivates the appropriate plan for $\{\phi_d, \theta_d, z_d\}$.

3. Problem Formulation

Our planning goal is not just stabilizing but safely guiding the faulty quadrotor to the designated safe zone. This goal can be expounded through the formalism of the backward reachability idea, which is instrumental in formulating our motion planning problem, as detailed in the following section.

---

[1]Without loss of generality, we refer smooth trajectory to those having Lipschitz continuous second order derivatives throughout the whole time interval.

3.1. Backward Reachability Problem

Suppose, for a time horizon $[t_0, t_f]$, the admissible control profile $\mathbf{u}(\cdot)$ is drawn from the set of measurable functions:

$$\mathbf{u}(\cdot) \in \mathbb{U} = \left\{ \mathcal{U} : [t_0, t_f] \to \mathbf{U} \mid \mathcal{U}(\cdot) \text{ is measurable} \right\}.$$

Since $\mathbf{X}$ and $\mathbf{U}$ are compact sets and $\mathcal{F} : \mathbf{X} \times \mathbf{U} \to \mathbb{R}^{12}$ is bounded Lipschitz continuous in most practice, there exists a unique solution to (8) for a given $\mathbf{u}(\cdot)$. This solution, equivalently said the trajectory of (8) starting from initial condition $\mathbf{x}_0$ at time t_0 under $\mathbf{u}(\cdot)$, is denoted by $\xi(t; \mathbf{x}_0, t_0, \mathbf{u}(\cdot)) : [t_0, t_f] \to \mathbf{X}$ and satisfying the following differential equation almost everywhere:

$$\frac{d}{dt} \xi(t; \mathbf{x}_0, t_0, \mathbf{u}(\cdot)) = \mathcal{F}\left(\xi(t; \mathbf{x}_0, t_0, \mathbf{u}(\cdot)), \mathbf{u}\right),$$

$$\xi(t_0; \mathbf{x}_0, t_0, \mathbf{u}(\cdot)) = \mathbf{x}_0.$$

As per the safety of the faulty quadrotor, let $\mathbf{X}_{\text{safe}} \subset \mathbf{X}$ be a safe zone to which the faulty quadrotor is desired to reach. Then, we are interested in whether $\mathbf{X}_{\text{safe}}$ can be reached from $\mathbf{x}_0$ by admissible system input, and if so, seeking the control profile that drives the faulty quadrotor into $\mathbf{X}_{\text{safe}}$. These can be addressed by the following problems.

Problem 1. *For a time horizon $[t_0, t_f]$ of duration $T = t_f - t_0$, compute the set of all possible initial states $\mathbf{x}_0$ such that the trajectories starting from this set can reach $\mathbf{X}_{\text{safe}}$ within T, which is represented by the following backward reachable set $\mathcal{G}(T) \subseteq \mathbf{X}$:*

$$\mathcal{G}(T) := \{\mathbf{x} \in \mathbf{X} \mid \exists \mathbf{u}(\cdot) \in \mathbb{U}, \xi(t_f; \mathbf{x}, t_0, \mathbf{u}(\cdot)) \in \mathbf{X}_{\text{safe}}\}. \tag{14}$$

Problem 2. *When $\mathbf{x}_0 \in \mathcal{G}(T)$, find $\mathbf{u}(\cdot) \in \mathbb{U}$ for $[t_0, t_f]$ such that:*

$$\xi(t_f; \mathbf{x}_0, t_0, \mathbf{u}(\cdot)) \in \mathbf{X}_{\text{safe}}. \tag{15}$$

Problem 1 solves the existence of the trajectory ξ satisfying (15) by simply checking $\mathbf{x}_0 \in \mathcal{G}(T)$. Yet, the reachable set for the nonlinear dynamical system (8) is computationally demand-

12

ing, for which a number of approximation techniques have been studied [33, 34]. On the other hand, Problem 2 can be solved by numerical optimization [25, 26] or sampling-based method [27, 28]. However, these methods are either computationally expensive or prone to fail in finding a feasible solution.

To simplify computations, we recast the original backward reachability problem (Problem 1 and 2) specific to the motion trajectory being compatible with the embedded FTC law. Let $\xi_d : [t_0, t_f] \rightarrow \mathbf{X}$, $\xi_d(t_0) = \mathbf{x}_0$, be a smooth trajectory. Then, based on FTC law in (12), we draw the corresponding control profile which can be expressed by the following non-anticipative functional γ that maps ξ_d to $\mathbf{u}^{\text{FTC}}$ [35]:

$$\gamma[\xi_d](\cdot) := \mathbf{u}^{\text{FTC}}(\cdot).$$

Note that $\gamma[\xi_d](\cdot)$ does not necessarily belong to $\mathbb{U}$ since $\mathbf{u}^{\text{FTC}}$ may violate the saturation constraints. But when $\gamma[\xi_d](\cdot) \in \mathbb{U}$, one can expect $\xi(t) \approx \xi_d(t)$ for sufficiently large t by Lemma 1. Utilizing this fact, we can associate the original problems with the desired reference trajectory $\xi_d(\cdot)$ instead of control profile $\mathbf{u}(\cdot)$.

Problem 3. *For a time horizon $[t_0, t_f]$ of duration $T = t_f - t_0$, compute the backward reachable set $\mathcal{G}^{\text{FTC}}(T) \subseteq \mathcal{G}(T)$ defined as follows:*

$$\mathcal{G}^{\text{FTC}}(T) := \{\mathbf{x} \in \mathbf{X} \mid \exists \xi_d(\cdot), \gamma[\xi_d](\cdot) \in \mathbb{U}, \xi(t_f; \mathbf{x}, t_0, \gamma[\xi_d](\cdot)) \in \mathbf{X}_{\text{safe}}\}. \tag{16}$$

Problem 4. *When $\mathbf{x}_0 \in \mathcal{G}^{\text{FTC}}(T)$, find $\xi_d(\cdot)$ for $[t_0, t_f]$ such that:*

$$\xi(t_f; \mathbf{x}_0, t_0, \gamma[\xi_d](\cdot)) \in \mathbf{X}_{\text{safe}}, \quad \text{where} \quad \gamma[\xi_d](\cdot) \in \mathbb{U}. \tag{17}$$

Problem 3 and 4 search for the specific trajectories that can be tracked by the FTC without violating the saturation constraints, i.e., $\gamma[\xi_d](t) = \mathbf{u}^{\text{FTC}}(t)$, $\forall t \in [t_0, t_f]$. As will be presented in the next section, the proposed motion planner does not directly solve for the backward reachable set computation (solution to Problem 3), rather indirectly provides the feasible conditions on

13

the closed-form trajectory formula (solution to Problem 4) to attain the backward reachability goal. Compared to direct control profile design, this would result in a conservative solution since $\exists \mathbf{x} \in \mathcal{G}^{\text{FTC}}(T)^c \cap \mathcal{G}(T)$ (See Fig. 3). Nonetheless, we argue the trajectory design is more relevant to incorporate factors from different levels (faulty quadrotor dynamics, embedded FTC capacity, global goal point, etc.), and thus better fit to our motion planning objective.
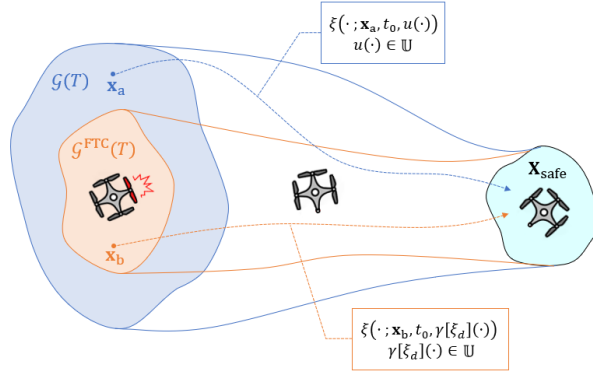

Figure 3: Comparison of the backward reachable sets for the faulty quadrotor maneuver: direct control profile design (blue) vs. trajectory design with the embedded FTC law (orange).

Remark 1. *Note that Lemma 1 is not a necessary condition but a sufficient condition for the asymptotic stability of FTC. This implies that it is still plausible that the faulty quadrotor exhibits the stable maneuver even under $\mathbf{u} \neq \mathbf{u}^{\text{FTC}}$. In such a case, however, we no longer theoretically guarantee the tracking error convergence nor stability of the embedded FTC. To assure the tracking performance of FTC, Problem 3 and 4 refrain these cases at the planning level.*

Remark 2. *One may wish to directly seek $\mathbf{u}(\cdot)$ while taking into account the saturation constraints at the control level, using optimization-based control like MPC [19]. However, the faulty quadrotor dynamics in our problem is not fully controllable, resulting in highly non-convex optimization which is computationally prohibitive to obtain a feasible solution. Stabilizing the dynamical system with bounded system input can be posed by control Lyapunov function and/or Sontag's universal formula [36]. This method, however, requires to find the specific Lyapunov function which is tricky under the simultaneous presence of complete rotor loss and rotor saturation. For this reason, the existing Lyapunov-based FTC approaches rarely account for rotor*

14

saturation [12, 15].

Remark 3. *The formulated trajectory planning problem shares a similar philosophy to the reference governor technique [23, 24]. However, the reference governor generally relies on solving optimization problems, which may not be adequate for timely providing the trajectory. Some variants such as the explicit reference governor obviate the need for online optimization, but require expensive computation efforts offline, e.g., pre-computing the invariant set, the returnable set, etc [37].*

In what follows, we present an analytical closed-form solution to Problem 4 and its existence guarantee as the solution to Problem 3, generating the safe motion trajectory of the faulty quadrotor in a real-time fashion.

4. Algorithm Development

The quadrotor dynamics is characterized by nonlinear, strongly coupled and under-actuated nature. The conventional control architecture tackles these complexities in a cascaded manner, i.e., attitude control in the inner loop and then position control in the outer loop [12, 13, 21]. Our fault tolerant motion planner adopts the similar structure consisting of two folds:

- Attitude Stabilizing Motion Planner (ASMP): Generates the reference motion trajectory from the initial condition to the stable equilibrium state.
- Position Stabilizing Motion Planner (PSMP): Generates the reference motion trajectory from the equilibrium state to the designated safe zone.

Unfortunately, the proposed planning strategy can not be immediately brought to bear, since the steady-state equilibrium point does not exist for the faulty quadrotor dynamics (3) subject to the complete rotor loss condition $F_4 = 0$. To this end, we resort to the notion of periodic equilibrium (PE) that can delicately bridge the transition between ASMP and PSMP.

The PE state for the faulty quadrotor dynamics can be defined by the equilibrium state in

15

subsystem (9) as follows [17, 18]:

$$\mathcal{X}_{\text{PE}} = [\phi_{\text{PE}} \ \theta_{\text{PE}} \ p_{\text{PE}} \ q_{\text{PE}} \ r_{\text{PE}} \ z_{\text{PE}} \ v_{z,\text{PE}}]^{\text{T}} \in \{\mathcal{X} \mid \dot{\mathcal{X}} = [0\ 0\ 0\ 0\ 0\ 0\ 0]^{\text{T}}\} \quad (18)$$

where the subscript $\cdot_{\text{PE}}$ denotes the state at the PE. For the sake of notation congruity, this subscript will be used in all the other states $\mathbf{x}_{\text{PE}}$ (e.g., x_{PE}) and their derivatives $\dot{\mathbf{x}}_{\text{PE}}$ (e.g., $\dot{\psi}_{\text{PE}}$), as well as system input $\mathbf{u}_{\text{PE}}$ (e.g., $u_{p,\text{PE}}$) and rotor thrusts (e.g., $F_{1,\text{PE}}$), indicating their values at the PE state. Fig. 4 illustrates the PE behavior of the faulty quadrotor that exhibits the circular position trajectory (Refer to [17] for details). The PE is not unique but infinitely many according to
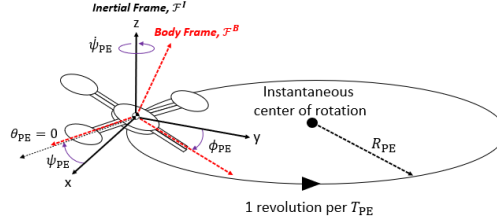

Figure 4: Circular motion trajectories of the faulty quadrotor at the periodic equilibrium.

different pairs of ϕ_{PE} and θ_{PE}. Without loss of generality, we consider the PE states with $\theta_{\text{PE}} = 0$ henceforth.[2] Then, the corresponding circular trajectories are expressed by the following orbital radius R_{PE} and period T_{PE}:

$$R_{\text{PE}} = \frac{g}{\dot{\psi}_{\text{PE}}^2} \tan(\phi_{\text{PE}}), \quad T_{\text{PE}} = \frac{2\pi}{|\dot{\psi}_{\text{PE}}|} \quad (19)$$

where the PE yaw rate $\dot{\psi}_{\text{PE}}$ is determined by:

$$\dot{\psi}_{\text{PE}} = r_{\text{PE}}(\tan(\phi_{\text{PE}})\sin(\phi_{\text{PE}}) + \cos(\phi_{\text{PE}})) \quad (20)$$

and the PE yaw rate r_{PE} is given by:

---

[2]The PE state with $\theta_{\text{PE}} = 0$ imposes two opposing propellers to produce equal thrust, making the PE derivation easy [18]. This setup is also useful for our safety concern since it minimizes the quadrotor's rotation rate and maximizes the saturation margin, i.e., the distance between the PE thrust and the thrust limit [17].

16

$$r_{\mathrm{PE}} = \begin{cases} \frac{bmg}{k_r} & \text{if } \phi_{\mathrm{PE}} = 0, \\[2ex] \frac{-\frac{k_r l}{b} + l\sqrt{(\frac{k_r}{b})^2 + \frac{8mg(I_z - I_y)\tan(\phi_{\mathrm{PE}})}{l\cos(\phi_{\mathrm{PE}})}}}{4(I_z - I_y)\tan(\phi_{\mathrm{PE}})} & \text{if } \phi_{\mathrm{PE}} \neq 0. \end{cases} \tag{21}$$

The full derivation of (21) is provided in Appendix B in our supplementary document [32].

The key observation is, under $\theta_{\mathrm{PE}} = 0$, the rest of states, system input, and rotor thrusts at the PE state is determined by ϕ_{PE} value. In our problem, on the other hand, ϕ_{PE} cannot be arbitrarily chosen but restricted by the rotor constraints (7) which imposes the allowable range of ϕ_{PE}, $\underline{\phi}_{\mathrm{PE}} \leq \phi_{\mathrm{PE}} \leq \overline{\phi}_{\mathrm{PE}}$, where $\underline{\phi}_{\mathrm{PE}}$ and $\overline{\phi}_{\mathrm{PE}}$ are respectively subject to the lower limit ($F_i \geq 0$) and upper limit ($F_i \leq \overline{F}$) of rotor thrust, as shown in Fig. 5. The details for computing $\underline{\phi}_{\mathrm{PE}}$ and $\overline{\phi}_{\mathrm{PE}}$ will be discussed in the following subsection.
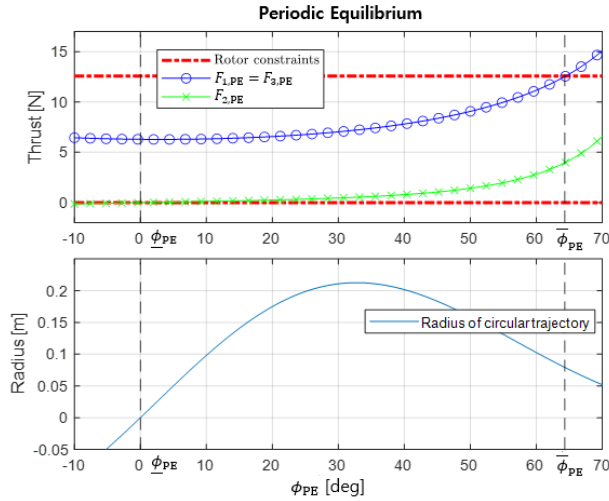

Figure 5: The PE of the quadrotor as a function of ϕ_{PE}, representing the orbital radius of the circular trajectory, and the thrusts, in the case of rotor R_4 failure. Allowable ϕ_{PE} range is restricted by rotor saturation constraints.

The proposed motion planner synthesizes the allowable PE states whereby the quadrotor is uniformly stable while not violating the saturation constraints. Specifically, the ASMP generates the reference trajectory from the initial state toward a certain PE state. Once the quadrotor has converged to the PE, the PSMP governs the reference trajectory to reach the safe zone while keeping it in the proximity of the PE. Considering the reference states fed into the embedded FTC, we only take care of $\{\phi_d, \theta_d, z_d\}$ rather than the whole states trajectory ξ_d. The overall

17

algorithm architecture is outlined in Fig. 6 where the pairs of ϕ_a, θ_a and ϕ_p, θ_p are respectively generated by the ASMP and the PSMP. Note that the altitude can be arbitrarily set regardless of ϕ_{PE} value. For simplicity, we set $z_d = z(t_0)$ to maintain the initial altitude.
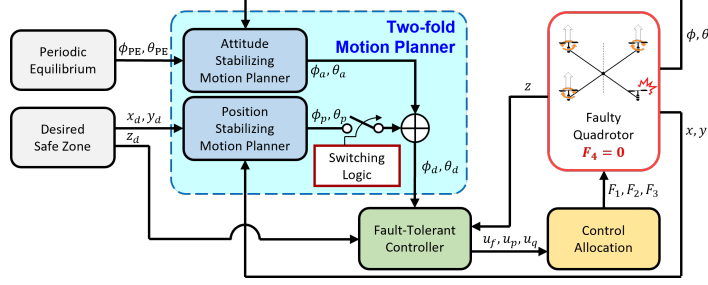

Figure 6: The algorithm structure of the proposed fault tolerant motion planner.

4.1. Attitude Stabilizing Motion Planner

The backward reachable set carried out by the ASMP is defined by:

$$\mathcal{G}_{PE}^{FTC}(T) := \{\mathbf{x} \in \mathbf{X} \mid \exists \xi_d(\cdot), \gamma\,[\xi_d]\,(\cdot) \in \mathbb{U}, \xi\left(t_f; \mathbf{x}, t_0, \gamma\,[\xi_d]\,(\cdot)\right) \in \mathbf{X}_{PE}\}$$

which explicates ξ_d that can be tracked by the embedded FTC without violating the saturation constraints and reach the PE state $\mathbf{X}_{PE} \subset \mathbf{X}$. To address the saturation constraints, the following assumption is made.

Assumption 1. *The rotor failure occurs when the quadrotor maneuvers with small attitude, i.e., around the steady-state where $v_x, v_y, v_z, \phi, \theta, p, q, r$ and their derivatives are approximately zeros.*

The above assumption is widely adopted in the quadrotor control techniques [8, 10] whose results are applicable beyond the steady-state in practice. Our trajectory design is also viable even when the quadrotor state is out of the assumed flight envelope, as will be demonstrated in Section 5. By Assumption 1, F_1 and F_3 produce equivalent thrusts under $\theta \approx 0$, which yields $u_q = l(F_3 - F_1) \approx 0$. Then, the constraint of u_f from (7) can be simplified as:

$$\overline{F} \le u_f \le 2\overline{F}. \tag{22}$$

Applying (22) and (7a) into the FTC law (12) yields the following inequalities, respectively for u_f and u_p:

$$
\begin{aligned}
\overline{F} &\leq \frac{m}{\cos(\phi)\cos(\theta)}\left(g + \ddot{z}_d + \frac{k_t \dot{z}}{m} + K_{1,z}(\dot{z}_d - \dot{z}) + K_{2,z}(z_d - z)\right) \leq 2\overline{F}, \\
0 &\leq -I_x bl\cos(\phi)\sin(\theta)\cos(\theta)u_f - \frac{I_x I_z l \cos(\phi)}{2I_x b \tan(\theta) - I_z l \cos(\phi)} \\
&\quad \left[\left(\frac{k_r r \cos(\phi)\tan(\theta)}{I_z} + \frac{pr\sin(\phi)\tan(\theta)(I_x - I_z)}{I_y} - \frac{\dot{\psi}\dot{\theta}}{\cos(\theta)} - \tan(\theta)\dot{\phi}\dot{\theta} + \frac{qr(I_z - I_y)}{I_x}\right.\right. \\
&\quad \left.+ \ddot{\phi}_d + K_{1,\theta}(\dot{\phi}_d - \dot{\phi}) + K_{2,\theta}(\phi_d - \phi)\right) - \tan(\phi)\tan(\theta)\left(\ddot{\theta}_d + \dot{\phi}\dot{\psi}\cos(\theta) + \frac{pr\cos(\phi)(I_x - I_z)}{I_y}\right. \\
&\quad \left.\left. - \frac{k_r r \sin(\phi)}{I_z} + K_{1,\theta}(\dot{\theta}_d - \dot{\theta}) + K_{2,\theta}(\theta_d - \theta)\right)\right] \leq l\overline{F}, \qquad \forall t \in [t_0, t_f].
\end{aligned}
\tag{23}
$$

As to ξ_d satisfying (23), the initial pitch $\theta(t_0)$ is close to the desired PE state $\theta_{PE} = 0$. Therefore, one can set the reference pitch to $\theta_a(t) = 0$, $\forall t \in [t_0, t_f]$, and as a result, all the other states of ξ_d are ruled by ϕ_a. If $\phi_a(\cdot)$ is smooth and satisfies (23), one can assure $\phi(t) = \phi_a(t)$ and $\theta(t) = \theta_a(t) = 0$ by Lemma 1. Conversely, when $\phi(t) = \phi_a(t)$, $\theta(t) = 0$ and $z_d(t) = z(t_0)$, $\forall t \in [t_0, t_f]$, (23) can be simplified into the following inequalities:

$$
\begin{aligned}
\overline{F} &\leq \frac{mg}{\cos(\phi_a)} \leq 2\overline{F}, \\
0 &\leq I_x\left(\frac{I_z - I_y}{I_x}r^2\tan(\phi_a) + \ddot{\phi}_a\right) \leq l\overline{F}, \qquad \forall t \in [t_0, t_f].
\end{aligned}
\tag{24}
$$

Among various trajectory design techniques, the polynomial trajectory is suitable for generating smooth curves in many robotic applications [38]. Here, we specifically exploit a quintic polynomial formula for $\phi_a(\cdot)$ [39]:

$$
\phi_a(t) = \sum_{k=0}^{5} a_k(t - t_0)^k, \qquad \forall t \in [t_0, t_f].
\tag{25}
$$

The primary concern of the ASMP is then the roll motion trajectory that obeys the following

boundary conditions:

$$
\begin{cases}
\phi_a(t_0) = \phi(t_0) \\
\dot{\phi}_a(t_0) = \dot{\phi}(t_0) \\
\ddot{\phi}_a(t_0) = \ddot{\phi}(t_0),
\end{cases}
\qquad
\begin{cases}
\phi_a(t_f) = \phi_{\mathrm{PE}} \\
\dot{\phi}_a(t_f) = 0 \\
\ddot{\phi}_a(t_f) = 0.
\end{cases}
\tag{26}
$$

The subsequent discussions are for the design of the coefficients a_k, $k = 0, 1, \cdots, 5$ to fulfill (26) and (24).

Proposition 1. *Over the time duration $T = t_f - t_0$, the reference roll trajectory (25) that complies with the boundary condition (26) can be designed with the following coefficients:*

$$
\begin{aligned}
a_0 &= \phi(t_0), \\
a_1 &= \dot{\phi}(t_0), \\
a_2 &= \frac{\ddot{\phi}(t_0)}{2}, \\
a_3 &= \frac{20(\phi_{\mathrm{PE}} - \phi(t_0)) - 12\dot{\phi}(t_0)T - 3\ddot{\phi}(t_0)T^2}{2T^3}, \\
a_4 &= \frac{-30(\phi_{\mathrm{PE}} - \phi(t_0)) + 16\dot{\phi}(t_0)T + 3\ddot{\phi}(t_0)T^2}{2T^4}, \\
a_5 &= \frac{12(\phi_{\mathrm{PE}} - \phi(t_0)) - 6\dot{\phi}(t_0)T - \ddot{\phi}(t_0)T^2}{2T^5}.
\end{aligned}
\tag{27}
$$

Proof. See Appendix C in our supplementary document [32].

With the coefficients (27), the designed reference trajectory smoothly leads to the desired PE state. The actual trajectory, on the other hand, may not align with the reference trajectory without ensuring (24) $\forall t \in [t_0, t_f]$. However, ensuring (24) is not straightforward since the actual yaw rate r and ϕ_a are nonlinearly interdependent. To circumvent such complication, we instead seek the upper and lower bounds of r over time.

Definition 1. *Several critical time instances related to $\ddddot{\phi}_a(t)$ are defined as:*

$$t_{c1} := \{t \in [t_0, t_f] \mid \dddot{\phi}_a(t) = \max(\dddot{\phi}_a(\cdot))\},$$

$$t_{c2} := \{t \in (t_0, t_f) \mid \dddot{\phi}_a(t) = 0\}, \tag{28}$$

$$t_{c3} := \{t \in [t_0, t_f] \mid \dddot{\phi}_a(t) = \min(\dddot{\phi}_a(\cdot))\}.$$

Lemma 2. *Consider the faulty quadrotor dynamics* (9) *having yaw torque* (6) *and being controlled by the FTC* (12). *Suppose Assumption 1 holds and $\phi(t) = \phi_a(t), \forall t \in [t_0, t_f]$. Then the upper and lower bounds of $r(t)$, expressed by $\underline{r}(t) \leq r(t) \leq \bar{r}(t)$, can be obtained as follows:*

$$\bar{r}(t) = \frac{\bar{u}_r}{k_r}\left(1 - e^{-\frac{k_r}{I_z}t}\right), \quad \underline{r}(t) = \frac{\underline{u}_r}{k_r}\left(1 - e^{-\frac{k_r}{I_z}t}\right)$$

where

$$\bar{u}_r = \frac{bmg}{\cos\left(\max(|\phi_a(t)|)\right)},$$

$$\underline{u}_r = b\left(\frac{mg}{\cos\left(\min(|\phi_a(t)|)\right)} - \frac{2}{l}\bar{u}_p\right),$$

$$\bar{u}_p = I_x\left(\frac{I_z - I_y}{I_x}(\frac{\bar{u}_r}{k_r})^2 \tan\left(\max(\phi_a(t))\right) + \dddot{\phi}_a(t_{c1})\right).$$

Proof. See Appendix D in our supplementary document [32].

Using the bound information of *r*, the successful trajectory tracking can be ascertained by the following proposition.

Proposition 2. *Consider the attitude dynamics of the faulty quadrotor* (9) *controlled by the FTC* (12). *Suppose Assumption 1 holds. Then, one can guarantee $\phi(t) = \phi_a(t), \forall t \in [t_0, t_f]$ if $\phi_a(\cdot)$ designed by* (25) *with coefficients* (27) *satisfies the following inequality:*

$$0 \leq \phi_{\text{PE}} \leq \cos^{-1}\left(\frac{mg}{2\bar{F}}\right),$$

$$I_x\left(\frac{I_z - I_y}{I_x}(\frac{\bar{u}_r}{k_r})^2 \tan\left(\phi_{\text{PE}}\right) + \dddot{\phi}_a(t_{c1})\right) \leq l\bar{F}, \tag{29}$$

$$0 \leq \frac{I_z - I_y}{I_x}\underline{r}^2(t_{c2})\tan\left(\phi_a(t_{c2})\right) + \dddot{\phi}_a(t_{c3}).$$

Proof. See Appendix E in our supplementary document [32].

21

Based on the Propositions 1 and 2 results, the ASMP can design the reference roll trajectory that guides the faulty quadrotor toward the designated PE state; and can safely be tracked by the embedded FTC while adhering to the rotor saturation constraints (7), as illustrated in Fig. 7.
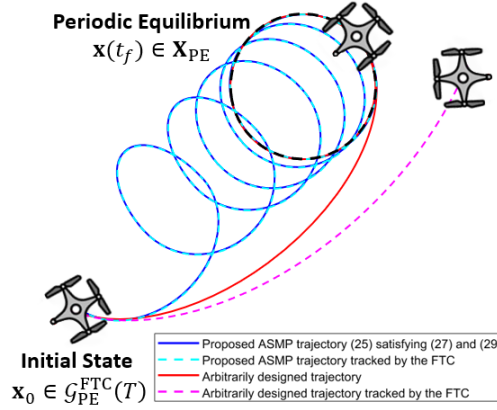

Figure 7: Comparison of the faulty quadrotor's motion on the horizontal plane when tracking the proposed ASMP trajectory and the arbitrary trajectory.

Remark 4. *Eq.* (29) *can be further deliberated from an optimality perspective, suggesting the safer reference trajectory. For instance, unlike the arbitrary ϕ_{PE} in [18], our ASMP can be advised to optimally set ϕ_{PE} in terms of saturation margin, i.e., the maximum distance between the PE thrust and the thrust limit.*

4.2. Position Stabilizing Motion Planner

Once the faulty quadrotor reaches the designated PE state by the ASMP, the PSMP is triggered to generate the trajectory $\xi_d(\cdot)$ that yields $\xi\left(t_f; \mathbf{x} \in \mathbf{X}_{\text{PE}}, t_0, \gamma\left[\xi_d\right](\cdot)\right) \in \mathbf{X}_{\text{safe}}$. Similar to the ASMP case, the quadrotor pitch conforms to the PE state by setting $\theta_d = \theta_p = \theta_{\text{PE}} = 0$ and the whole ξ_d can be governed by the reference roll trajectory ϕ_d during the PSMP phase. Consequently, the primary task of PSMP is to design ϕ_p that guides the faulty quadrotor towards the designated safe zone $\mathbf{X}_{\text{safe}}$ while obeying the rotor saturation constraints.

As depicted in Fig. 6, ϕ_p is added to ϕ_{PE} to produce ϕ_d when the PSMP is switched on, which

is mathematically written by:

$$\phi_d(t) = \begin{cases} \phi_{\text{PE}} & \text{if } t_0 \leq t < t_p \\ \phi_{\text{PE}} + \phi_p(t) & \text{if } t_p \leq t \leq t_f, \end{cases} \tag{30}$$

$$\theta_d(t) = 0, \qquad \forall t \in [t_0, t_f].$$

where $t_p \in [t_0, t_f]$ is the switching time specified later. Recalling (24), the saturation constraint for the PSMP is given by:

$$\overline{F} \leq \frac{mg}{\cos(\phi_d)} \leq 2\overline{F},$$

$$0 \leq I_x \left(\frac{I_z - I_y}{I_x} r^2 \tan(\phi_d) + \ddot{\phi}_d \right) \leq l\overline{F}, \quad \forall t \in [t_0, t_f]. \tag{31}$$

Like the ASMP, if $\phi_d(\cdot)$ is smooth and satisfies (31), the resulting actual trajectory would fulfill $\phi(t) = \phi_d(t)$, $\theta(t) = 0$ and $z_d(t) = z(t_0)$, $\forall t \in [t_0, t_f]$. Now, expanding the scope to the position of the faulty quadrotor, substituting $\theta = 0$ to (3) yields the following horizontal translation dynamics:

$$\dot{x} = v_x,$$

$$\dot{y} = v_y,$$

$$\dot{v}_x = \frac{1}{m}[-k_t v_x + u_f \sin\phi \sin\psi], \tag{32}$$

$$\dot{v}_y = \frac{1}{m}[-k_t v_y - u_f \sin\phi \cos\psi].$$

From (32), the horizontal motion depends on the roll and yaw states. Considering the cyclic yaw motion of the faulty quadrotor, the roll motion can correspondingly be designed as a sinusoid that oscillates near ϕ_{PE} as follows:

$$\phi_p(t) = a[\cos(\omega_p(t - t_p)) - 1], \quad \forall t \in [t_p, t_f] \tag{33}$$

where $a > 0$ is the amplitude and ω_p is the angular frequency of the designed ϕ_p trajectory. Note that (33) ensures smoothness of ϕ_d with any arbitrary $t_p \in [t_0, t_f]$.

23

The sinusoidal roll motion (33) influences the other motions to oscillate periodically. Accordingly, we introduce the following state definition during the PSMP phase.

Definition 2. *The subscript $\cdot_{\mathrm{PSMP}}$ denotes the mean value of the oscillating states subject to the roll motion* (33) *during the PSMP phase. For example, the mean of yaw rate is represented by:*

$$r_{\mathrm{PSMP}} := \int_{nT_{\mathrm{PSMP}}}^{(n+1)T_{\mathrm{PSMP}}} r(t)\, dt, \quad n = 0, 1, 2, \cdots$$

where $T_{\mathrm{PSMP}} = 2\pi/\omega_p$ is the period of oscillation during the PSMP phase.

When $\phi(t) = \phi_d(t)$, applying Definition 2 to (33) yields $\phi_{\mathrm{PSMP}} = \phi_{\mathrm{PE}} - a$. And substituting ϕ_{PSMP} for ϕ_{PE} in (20) and (21) respectively approximates $\dot{\psi}_{\mathrm{PSMP}}$ and r_{PSMP} as follows:

$$\dot{\psi}_{\mathrm{PSMP}} \approx r_{\mathrm{PSMP}}\left(\tan\left(\phi_{\mathrm{PSMP}}\right)\sin\left(\phi_{\mathrm{PSMP}}\right) + \cos\left(\phi_{\mathrm{PSMP}}\right)\right)$$

$$r_{\mathrm{PSMP}} \approx \begin{cases} \frac{bmg}{k_r} & \text{if } \phi_{\mathrm{PSMP}} = 0 \\[2mm] \frac{-\frac{k_r l}{b} + l\sqrt{\left(\frac{k_r}{b}\right)^2 + \frac{8mg(I_z - I_y)\tan(\phi_{\mathrm{PSMP}})}{l\cos(\phi_{\mathrm{PSMP}})}}}{4(I_z - I_y)\tan(\phi_{\mathrm{PSMP}})} & \text{if } \phi_{\mathrm{PSMP}} \neq 0. \end{cases}$$

In line with these set-ups, the following results present the design of parameters ω_p, t_p, a of (33) such that leads to the designated safe zone while fulfilling (31).

Proposition 3. *Consider the position dynamics of the faulty quadrotor* (32) *controlled by the FTC* (12). *Let (x_d, y_d) be the desired horizontal position inside the safe zone. Over the time horizon $[t_0, t_f]$, the reference roll trajectory that guides the faulty quadrotor from the PE state to (x_d, y_d) can be designed by* (33) *with the following parameters:*

$$\omega_p = \dot{\psi}_{\mathrm{PSMP}},$$
$$t_p = t_0 + \frac{1}{\dot{\psi}_{\mathrm{PE}}}\left[\tan^{-1}\left(\frac{y_d - y(t_0)}{x_d - x(t_0)}\right) - \psi(t_0) + \frac{\pi}{2}\right] + \frac{2\pi n}{\dot{\psi}_{\mathrm{PSMP}}}, \ n \in \{0, 1, 2, \cdots\}, \ \text{s.t. } t_p \geq t_0, \quad (34)$$
$$a = \frac{2k_t \cos^2(\phi_{\mathrm{PE}})}{mg(t_f - t_p)}\sqrt{\left(x_d - x(t_p)\right)^2 + \left(y_d - y(t_p)\right)^2}.$$

With the designed ω_p, t_p and a, the designed trajectory converges to the circular orbit centered

24

on x_d, y_d, with the following mean velocity represented in the polar coordinate:

$$|v_{\text{PSMP}}| = \frac{mga}{2k_t \cos^2(\phi_{\text{PE}})}, \quad \angle v_{\text{PSMP}} = \psi(t_p) - \frac{\pi}{2}.$$

Proof. See Appendix F in our supplementary document [32].

Fig. 8 geometrically visualizes the impact of each parameter design in the PSMP trajectory. The magnitude of the mean velocity is proportional to the parameter a, and the direction is determined by the time t_p. Besides, the mean trajectory cannot be straight without the appropriate selection of ω_p.
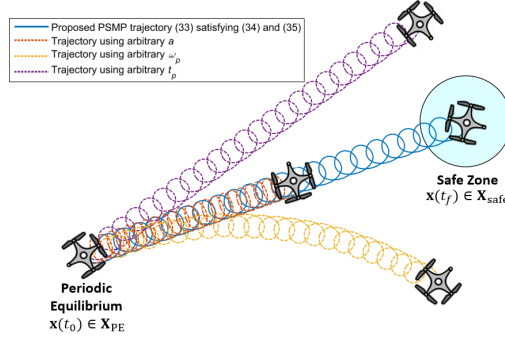

Figure 8: Comparison of the PSMP trajectory when using the proposed parameters (34) and arbitrary parameters.

Having the reference trajectory (33) with (34), the actual trajectory may still fail to track the reference due to the saturation constraints (31), entailing the following proposition.

Proposition 4. *Consider the position dynamics of the faulty quadrotor* (32) *controlled by the FTC* (12)*. Suppose the faulty quadrotor is initially at the PE state, and is tracking the reference trajectory designed by* (33) *with parameters* (34)*. Then, one can guarantee $\phi(t) = \phi_d(t), \forall t \in [t_0, t_f]$ if the designed parameter a satisfies:*

$$0 < a \leq \min\left(\frac{l\overline{F} - \beta_1}{I_x \dot{\psi}_{\text{PSMP}}^2}, \frac{\beta_1}{\beta_2 - I_x \dot{\psi}_{\text{PSMP}}^2} \right) \tag{35}$$

where $\beta_1 := (I_z - I_y) r_{\text{PSMP}}^2 \tan(\phi_{\text{PSMP}})$ and $\beta_2 := 2(I_z - I_y) r_{\text{PSMP}}^2 \sec^2(\phi_{\text{PSMP}})$.

25

Proof. See Appendix G in our supplementary document [32].

Proposition 3 and 4 grants the goal of the PSMP, enabling the reference roll trajectory to guide the faulty quadrotor from the PE state to the designated safe zone; and to be safely tracked by the embedded FTC while not violating the rotor saturation constraints (7).

Remark 5. *Trade-off exists in parameter a design respecting* (34) *and* (35). *Specifically, with large a, the quadrotor quickly converges to the desired position* (x_d, y_d), *whereas small a provides the quadrotor with a larger saturation margin.*

4.3. Main Result

We finally consolidate all the trajectory design schemes for the ASMP and PSMP into the main result of this paper.

Theorem 1. *Consider the quadrotor dynamics* (8) *that is subject to a faulty rotor* (5), (6), *rotor saturation constraints* (7), *and is controlled by the FTC* (12). *Suppose Assumption 1 holds and time horizon* $[t_0, t_f]$ *is sufficiently large. Then, the reference trajectory* $\xi_d(\cdot)$ *that guarantees* (17) *can be designed by consecutively implementing the ASMP* (25) *and the PSMP* (33), *which are respectively predicated upon the propositions. i.e., Propositions 1, 2 for the ASMP, and 3, 4 for the PSMP.*

Proof. Let us decompose the time horizon $[t_0, t_f]$ into $[t_0^A, t_f^A]$ and $[t_0^P, t_f^P]$ where $t_0^A \equiv t_0$, $t_f^A \equiv t_o^P$, and $t_f^P \equiv t_f$. The ASMP is then implemented during $[t_0^A, t_f^A]$, followed by the PSMP during $[t_0^P, t_f^P]$.

By Assumption 1, $\mathbf{x}_0 \in \mathcal{G}_{\mathrm{PE}}^{\mathrm{FTC}}(T^A)$ if $T^A = t_f^A - t_0^A$ is sufficiently large. Then, Proposition 1 carries out smooth $\xi_d(\cdot)$ from $\xi_d\left(t_0^A\right) = \mathbf{x}_0$ to $\xi_d(t_f^A) \in \mathbf{X}_{\mathrm{PE}}$. Likewise, Proposition 3 allows for smooth $\xi_d(\cdot)$ from $\xi_d(t_0^P) \in \mathbf{X}_{\mathrm{PE}}$ to $\xi_d(t_f^P) \in \mathbf{X}_{\mathrm{safe}}$. Combining these two trajectory segments by matching $\xi_d(t_f^A) = \xi_d(t_0^P)$, the resulting $\xi_d(\cdot)$ starts at $\xi_d(t_0) = \mathbf{x}_0$, ends with $\xi_d(t_f) \in \mathbf{X}_{\mathrm{safe}}$, and is smooth throughout $[t_0, t_f]$.

As to trajectory tracking by the FTC under saturation constraints, Propositions 2 and 4 assure $\gamma\left[\xi_d\right](t) = \mathbf{u}^{\mathrm{FTC}}(t)$ over $[t_0^A, t_f^A]$ and $[t_0^P, t_f^P]$ respectively. Thus, $\gamma\left[\xi_d\right](\cdot) \in \mathbb{U}$ and this, together
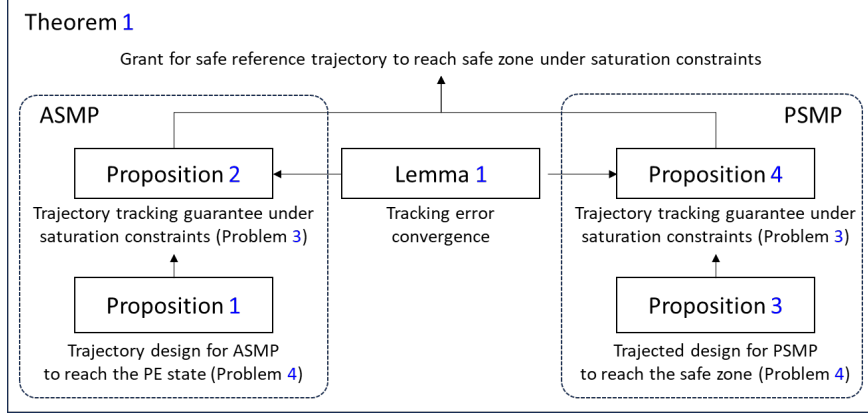
Figure 9: Overall schematic of theoretical results for the proposed fault tolerant motion planner.

with the smoothness of $\xi_d(\cdot)$, guarantees $\xi(t) \equiv \xi_d(t)$, $\forall t \in [t_0, t_f]$ by Lemma 1. This signifies that $\xi_d(\cdot)$ grants $\xi\left(t_f; \mathbf{x}_0, t_0, \gamma\left[\xi_d\right](\cdot)\right) \in \mathbf{X}_{\text{safe}}$ which completes the proof. ∎

Fig. 9 summarizes how each proposition and theorem interconnect and contribute to the main results. The online execution of the proposed fault tolerant motion planner is delineated in Algorithm 1.

Remark 6. *Algorithm 1 generates the new reference trajectory only if the actual trajectory significantly deviates from the previously generated reference trajectory. Such state-dependent and event-triggered trajectory update can save the computation resource, and enhance the reliability of the planning against the abrupt quadrotor maneuver under disturbance.*

5. Numerical Simulation

Simulation Setup. The proposed fault tolerant motion planning strategy is validated with numerical simulations. All the simulations are conducted in the MATLAB 2022a Simulink environment, running on a simulation PC equipped with an AMD Ryzen 7 3700X 8-Core processor and 16GB of RAM. The faulty quadrotor dynamics is modeled with the parameter values listed in Table 1 and the three FTC algorithms, PD control [14], SMC [12] and BSC [15] are implemented. Note that these control methods are all grounded in the NDI framework, and thus can

Algorithm 1 Fault Tolerant Motion Planning Algorithm

Given: $\mathbf{X}_{\text{safe}}$, $\mathcal{X}_{\text{PE}}$, T^A, T^P, quadrotor dynamics (8), and fault tolerant controller (12)
Initialize: $t_0 \leftarrow t$
Set $\phi_d(\tau) = \phi_{\text{PE}}$, $\theta_d(\tau) = 0$, $z_d(\tau) = z(t_0)$, $\forall \tau \geq t_0$
Generate $\xi_d(\cdot)$ over $[t_0, t_0 + T^A + T^P]$ based on $\phi_d(\cdot)$, $\theta_d(\cdot)$, and $z_d(\cdot)$ using (8) and (12)

1: **while** $\|\mathbf{X}_{\text{safe}} - \xi(t)\|_2 > \epsilon_1$ **do**
2:     **if** $\|\xi_d(t) - \xi(t)\|_2 > \epsilon_2$ or $t > t_0^A + T^A$ **then**
3:         **if** $\|\mathcal{X}_{\text{PE}} - \mathcal{X}(t)\|_2 > \epsilon_3$ **then**
4:             $t_0^A \leftarrow t$
5:             Compute $\phi_a(\cdot)$ over $[t_0^A, t_0^A + T^A]$ using (25) with (27) and (29)
6:             Set $\phi_d(\cdot) = \phi_a(\cdot)$ over $[t_0^A, t_0^A + T^A]$
7:             Generate $\xi_d(\cdot)$ over $[t_0^A, t_0^A + T^A]$ based on $\phi_d(\cdot)$, $\theta_d(\cdot)$, $z_d(\cdot)$ using (8) and (12)
8:         **else**
9:             $t_0^P \leftarrow t$
10:            Compute $\phi_p(\cdot)$ over $[t_0^P, t_0^P + T^P]$ using (33) with (34) and (35)
11:            Compute $\phi_d(\cdot)$ over $[t_0^P, t_0^P + T^P]$ using (30)
12:            Generate $\xi_d(\cdot)$ over $[t_0^P, t_0^P + T^P]$ based on $\phi_d(\cdot)$, $\theta_d(\cdot)$, $z_d(\cdot)$ using (8) and (12)
13:         **end if**
14:     **end if**
15:     $t \leftarrow t + \Delta t$
16:     Update $\xi(t)$
17: **end while**

---

where $\|\cdot\|_2$ is Euclidean norm, $\epsilon_1, \epsilon_2, \epsilon_3$ are sufficiently small positive values, and Δt is an execution time interval.

be seamlessly integrated with the proposed planner. The parameter values of individual FTCs are outlined in Table 2. Focusing on the control performance in tolerating fault, it is assumed the full state of the faulty quadrotor can be acquired timely and accurately.

Simulation Scenario. To verify the performance of the proposed algorithm in obstacle-rich environments, three simulation scenarios are considered: (i) hovering by the ASMP; (ii) waypoint tracking by the PSMP; and (iii) the collective behaviors to reach the safe zone while avoiding obstacles. Fig. 10 illustrates each of scenarios in detail. The hovering scenario evaluates the quadrotor's ability to maintain stable hovering despite the rotor failure. This can be attained by converging to the designated PE state, for which the attitude-stabilizing performance of the proposed ASMP is evaluated. The waypoint tracking scenario evaluates the quadrotor's ability to navigate toward a specified waypoint. This navigation is achieved through successive executions of the ASMP followed by the PSMP, from which the position stabilizing performance

is evaluated. Lastly, the collective behaviors scenario assesses the quadrotor's ability to reach a designated safe zone while avoiding obstacles. In this case, a higher-level path planner generates an obstacle-free sequence of waypoints, to which the proposed motion planner then follows for ensuring safe navigation.

Evaluation Methods. Ablation studies are performed to test various FTC schemes [12, 14, 15] with and without the proposed motion planner in Scenario I and II, followed by a comparative analysis against the existing planning method [26] in Scenario III. The following metrics are used to comprehensively assess the performance of the proposed planner: i) attitude/altitude tracking error over time; ii) maximum attitude/altitude tracking error; iii) maximum rotor thrust command exceeding rotor saturation; iv) average violation time of the rotor saturation; v) collision rate; and vi) execution time.
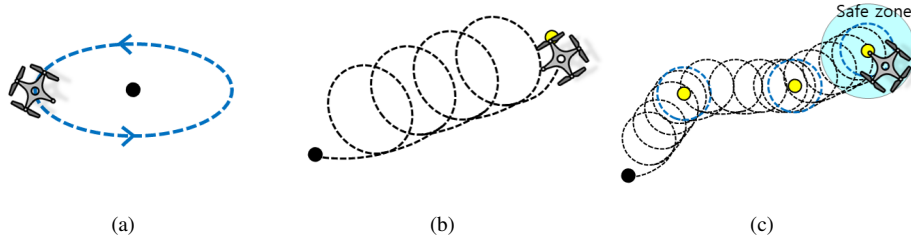

(a)   (b)   (c)

Figure 10: The faulty quadrotor scenario: (a) hovering, (b) waypoint tracking, and (c) collective behaviors. The blue dotted line represents the proposed ASMP trajectory, and the black dotted line represents the proposed PSMP trajectory. The black circle indicates the initial state, while the yellow circle marks the desired waypoint.

Parameter	Value	Unit
m	1.282	kg
g	9.81	m/s^2
l	0.1	m
I_x	$4.856 * 10^{-3}$	$kg \cdot m^2$
I_y	$4.856 * 10^{-3}$	$kg \cdot m^2$
I_z	$8.801 * 10^{-3}$	$kg \cdot m^2$
b	$6.6 * 10^{-3}$	$kg \cdot m^2$
k_t	0.5	$1/s$
k_r	0.02	$1/s$
$\overline{F}$	12.58	N

Table 1: Parameters of the quadrotor dynamics

Parameter	Value	Parameter	Value
$K_{\mathrm{P},\phi\theta}$	10	$K_{\mathrm{P},z}$	10
$K_{\mathrm{D},\phi\theta}$	10	$K_{\mathrm{D},z}$	1
$K_{\mathrm{SMC},\phi\theta}$	10	$K_{\mathrm{SMC},z}$	3
$\lambda_{\mathrm{SMC},\phi\theta}$	0.5	$\lambda_{\mathrm{SMC},z}$	1
$K_{\mathrm{BSC},\phi\theta}$	4	$K_{\mathrm{BSC},z}$	1

Table 2: FTC parameters

5.1. Scenario I: Hovering

In the first scenario, Monte Carlo (MC) simulations are carried out with 100 runs for the individual FTCs. In each simulation run, the rotor failure occurs at 1 second where $\theta(t_0^A)$ is set to $0°$ and $\phi(t_0^A)$ is randomly initialized according to Gaussian distribution with a mean of $20°$ and a standard deviation of $10°$. The results regarding altitude/attitude tracking error norm and rotor thrust profiles from the different FTCs over time are displayed in Fig. 11-13. Table 3 compares the average Root Mean Square Error (RMSE) of attitude to converge to the designed PE state across all MC simulations. These results demonstrate that the proposed ASMP successfully guides the faulty quadrotor to the PE state within a prescribed time horizon, with lower error than the stand-alone FTCs. More importantly, the exerted rotor thrusts keep distance from their limits all the time, which is well agreed with our theoretical results.

Further tests are conducted under a wide variation of the flight condition to validate the practical trajectory-tracking performance, different initial states $\phi(t_0^A)$ and $\theta(t_0^A)$ ranging between $-40°$ to $40°$. To examine the safe flight envelope, the maximum tracking errors are checked for individual simulations with different initial conditions. The maximum tracking error measure is useful to identify the worst possible maneuver of the quadrotor and thus is often adopted in robust control theory and safe control practices [40, 41], since it can be considered in hard constraints to ensure system stability and safety. The results are shown in Fig. 14 and 16 for maximum attitude tracking error, and Fig. 15 and 17 for the maximum altitude tracking error. Additionally, Fig. 18 compares the maximum excess rotor thrusts exerted by $\mathbf{u}^{\text{FTC}}$, and Fig. 19 compares the average time of $\mathbf{u}^{\text{FTC}}$ being violating the saturation constraints. Although the results sometimes fail when the fault condition overly exceeds the theoretical guarantee, they demonstrate that the proposed fault-tolerant motion planner ensures safety under a much broader range of fault conditions compared to the stand-alone FTCs.

5.2. Scenario II: Waypoint Tracking

In the second scenario, the faulty quadrotor is tasked to safely reach the designated safe zone, where the desired waypoint inside the safe zone is set to $x_d = 25m$, $y_d = -10m$, $z_d = 10m$. Prior
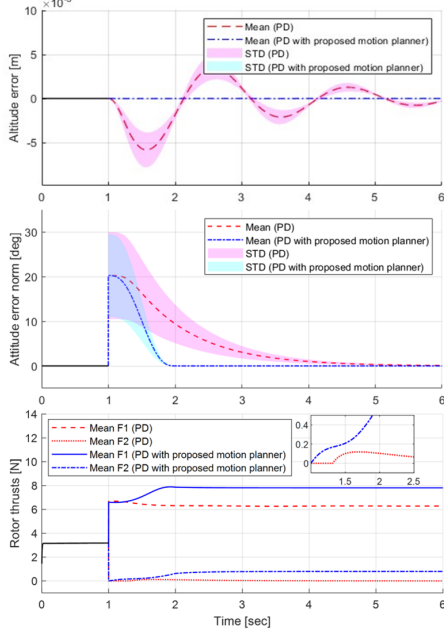
Figure 11: Comparison results of Monte Carlo simulation for the altitude/attitude tracking error norm and rotor thrusts, between PD control laws with and without the proposed motion planner.
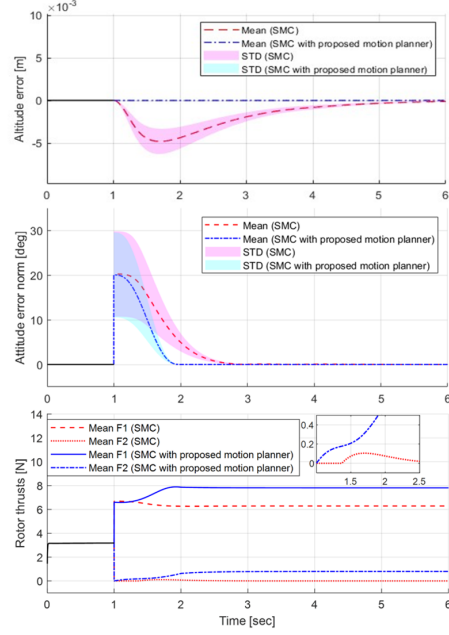
Figure 12: Comparison results of Monte Carlo simulation for the altitude/attitude tracking error norm and rotor thrusts, between SMC laws with and without the proposed motion planner.

to the rotor failure, the quadrotor flies along the x-axis with pitch angle $\phi = 10°$, starting from the initial position $x = 0m$, $y = 0m$, $z = 10m$. When the rotor failure occurs at 5 seconds, the initial fault condition is set to $\phi(t_0^A) = 10°$, $\theta(t_0^A) = 0°$, and $\psi(t_0^A) = 0°$.

Fig. 20 plots the traces of the faulty quadrotor position, each starting with the same fault condition but controlled by different FTCs. Apparently, when relying solely on the stand-alone FTCs, the quadrotor fails to carry out waypoint tracking and immediately crashes to the ground as soon as the fault occurs. This is clearly understood by Fig. 21, where the stand-alone FTCs frequently and severely violate the saturation constraints. The RMSE statistics are presented in Table 3. The results demonstrate that the proposed PSMP successfully guides the faulty quadrotor with higher tracking performance, regardless of the embedded FTC capability. Notably, the proposed planner keeps the exerted rotor thrusts below the specified limits, aligning with our theoretical results.
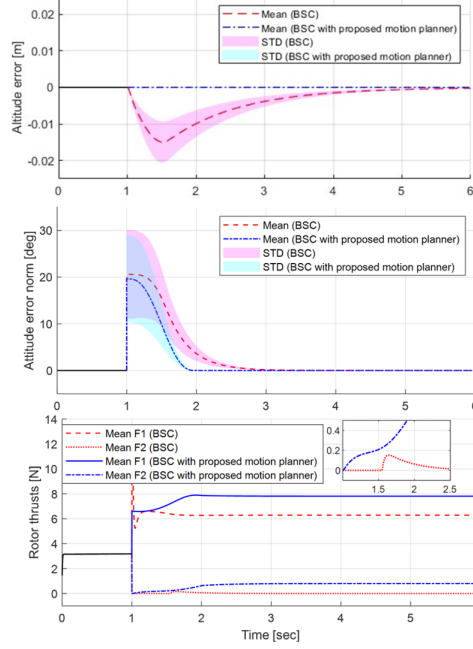
31

Figure 13: Comparison results of Monte Carlo simulation for the altitude/attitude tracking error norm and rotor thrusts, between BSC laws with and without the proposed motion planner.

5.3. Scenario III: Collective Behavior

Finally, the third scenario assesses the integrated fault tolerant motion planner when the faulty quadrotor is put in a general flight envelope surrounded by obstacles. To circumvent the obstacles, a higher-level path planner generates a sequence of obstacle-free waypoints, to which our motion planner is applied to generate the trajectory that trails the waypoints. Here, we specifically adopt RRT* algorithm considering its proven capability in rapid feasible path generation over obstacle-rich environment [42].

To validate the effectiveness of the proposed motion planner, the comparative analysis is conducted with the existing planner planning method, namely reference governor [26]. Each algorithm is run 25 times under the same simulation conditions, with an added noise of $0.01m$ standard deviation to the position state x, y and z. As depicted in Fig. 22, the initial condition of the quadrotor at fault is set with position state $x(t_0^A) = 0m$, $y(t_0^A) = 0m$, $z(t_0^A) = 10m$ and attitude state $\phi(t_0^A) = 0°$, $\theta(t_0^A) = 0°$, $\psi(t_0^A) = 0°$, and the desired position inside the safe zone is set as
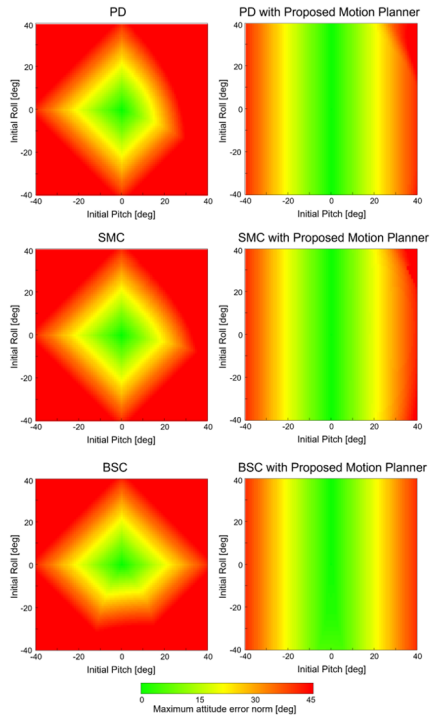
32

Figure 14: Comparison results of maximum attitude tracking error norm across a range of initial attitudes, between PD control, SMC, and BSC laws with and without the proposed motion planner.
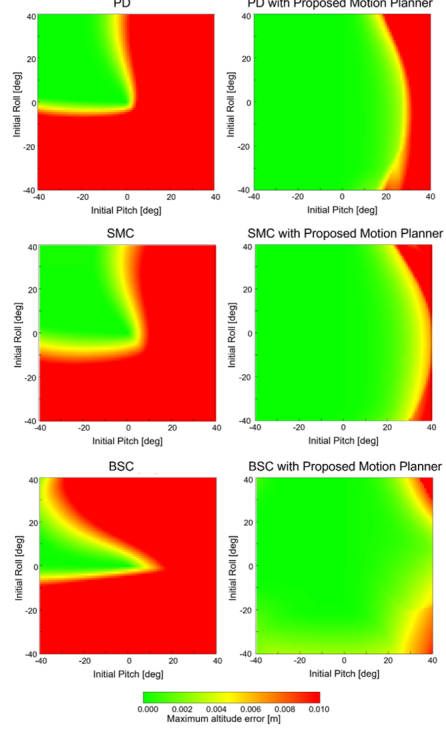

Figure 15: Comparison results of maximum altitude tracking error across a range of initial attitudes, between PD control, SMC, and BSC laws with and without the proposed motion planner.
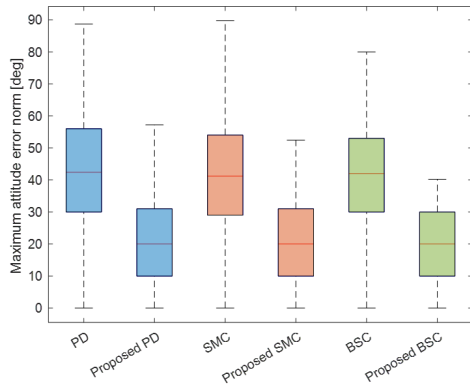

Figure 16: Comparison results of maximum attitude tracking error norm between PD control, SMC, and BSC laws with and without the proposed motion planner.
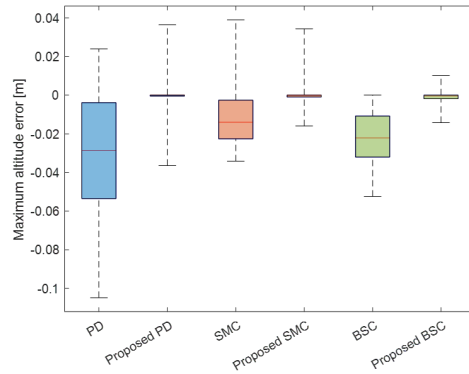

Figure 17: Comparison results of maximum altitude tracking error between PD control, SMC, and BSC laws with and without the proposed motion planner.

33

	Scenario I	Scenario II
PD	7.3386°	94.6234°
PD with proposed motion planner	5.7271°	0.8243°
SMC	6.7956°	7.1902°
SMC with proposed motion planner	5.7275°	0.6731°
BSC	6.3579°	3.5616°
BSC with proposed motion planner	5.7648°	0.1973°

Table 3: Comparison of average RMSE for attitude tracking error under different FTC laws with/without the proposed planner
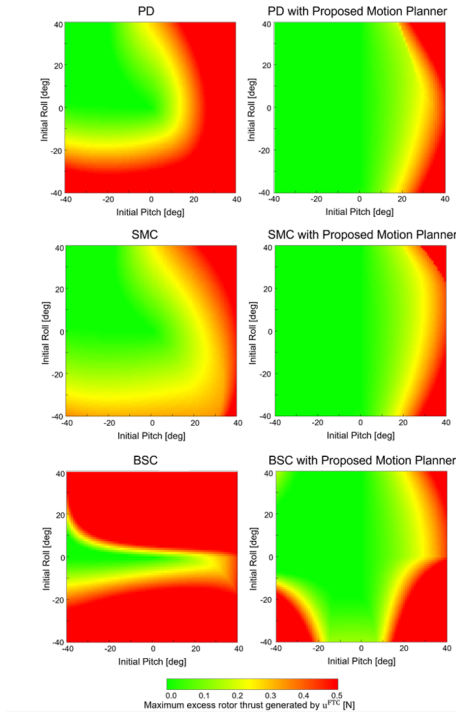

Figure 18: Comparison results of maximum excess rotor thrust generated by PD control, SMC, and BSC laws with and without the proposed motion planner.
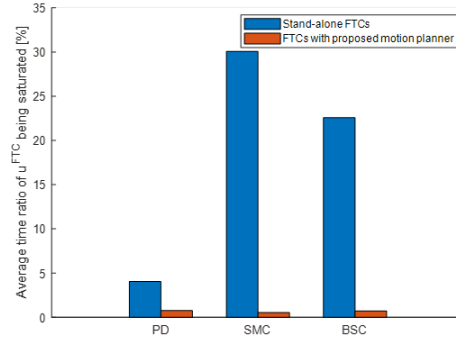
Figure 19: Comparison results of average time of $\mathbf{u}^{\text{FTC}}$ being saturated with and without the proposed motion planner.

$x_d = 40m$, $y_d = 44m$, $z_d = 10m$.

Fig. 22 displays the single-run simulation result for the each algorithm, and Table 4 compares the statistics of 25 runs with respect to different performance indices, including the collision rate, RMSE statistics for attitude tracking error, average execution time, and worst-case execution time. The average execution time measures the average duration the algorithm takes to

Figure 20: Comparison results of guidance to the desired safe zone between PD control, SMC, and BSC laws with and without the proposed motion planner.
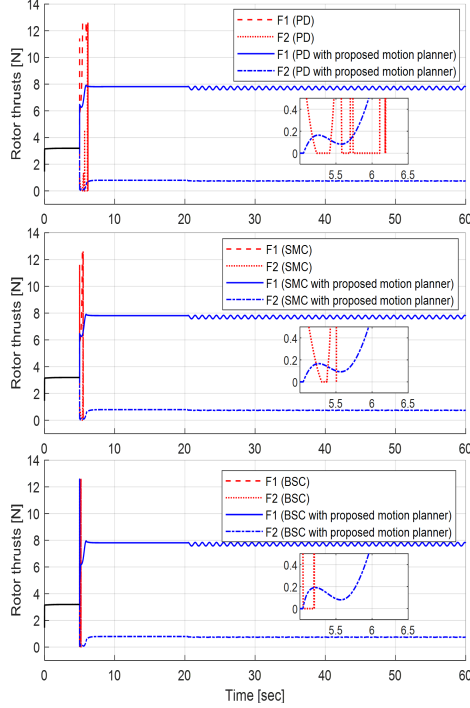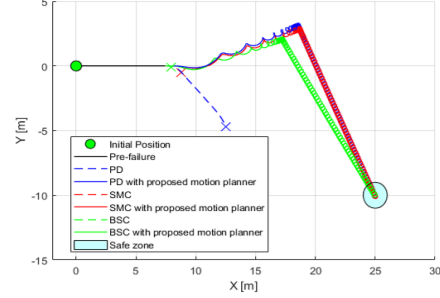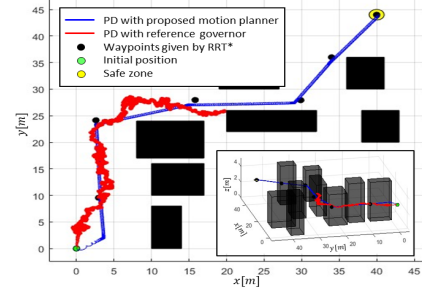

Figure 21: Comparison results of rotor thrusts when position stabilizing, between PD control, SMC, and BSC laws with and without the proposed motion planner.

Figure 22: Collision avoidance maneuver by integrating RRT* path planner into PD control with (blue) the proposed motion planner and (red) the reference governor.

compute the attitude commands per time step, whereas the worst-case execution time represents the maximum time taken across all time steps. The comparison results demonstrate that the proposed motion planner capably conducts waypoint flights while avoiding obstacles, assuring FTC tracking performance, and safely leading to the safe zone. Furthermore, the proposed planner maintains a lower execution time even in the worst case, confirming its excellence in real-time applications. This highlights the substantial computational advantages of the proposed analytical closed-form trajectory over optimization-based methods, and further underscores the robustness and efficiency of the proposed motion planner, demonstrating its reliability for maneuvering faulty quadrotors in dynamic and obstacle-rich environments.

Additionally, considering that most onboard computing processors in quadrotors are typically less powerful than desktop processors, we also evaluate the real-time computational efficiency

	Collision rate [%]	RMSE for attitude tracking error [°]	Average execution time [s]	Worst-case execution time [s]
Proposed motion planner	0	0.0136	2.6141×10^{-6}	1.2×10^{-3}
Reference governor	75	0.0974	4.068×10^{-3}	0.21

Table 4: The performance comparison between the proposed motion planner and the reference governor

	Computing frequency measured in simulation PC (AMD Ryzen 7 3700X)	Scale-down frequency in target onboard PC (STM32F405)
Proposed motion planner	382541Hz	17851Hz
Reference governor	246Hz	11.5Hz

Table 5: The comparison of the average computing frequency on the simulation PC and the corresponding scale-down frequency on the onboard PC (i.e., Crazyflie). The clock speed of the AMD Ryzen 7 3700X is 3.6GHz, while the STM32F405 processor used in Crazyflie operates at 168MHz

within the limited onboard computing resources for practical operation. Specifically, we appropriately associate the computing scales between the simulation PC and the onboard PC used in Crazyflie drones [43], and estimate the execution time of the algorithm as if it runs on an onboard PC [44]. In dynamic environments, quadrotors require a nominal flight operation frequency of more than 250Hz for stable maneuvering, even without fault [45]. We suppose this requirement become even more critical during emergency maneuvers, such as rotor failures. Table 5 clearly indicates that the proposed method significantly outperforms the existing method in terms of computation time, achieving a frequency well above the 250Hz required for stable quadrotor operation, whereas the existing method fails to fulfill this.

6. Conclusion

In this paper, we proposed a fault tolerant motion planner for the quadrotor subject to a complete rotor loss and rotor saturation constraints. Ensuring the safety of the faulty quadrotor at the planner level is a formidable task that encompasses different domain contexts such as nonlinear unstable dynamics, physical constraints, embedded FTC capability, etc., and thus has not been adequately deliberated in the existing work. The proposed motion planer sequentially designs the rotational and the translational motion trajectories that guide the faulty quadrotor to the safe zone in the presence of multiple constraints internally (a complete rotor loss and the saturation constraints of the remaining rotors) and externally (surrounding obstacles). The

36

motion trajectory is designed as an analytical closed-form expression and thus favorable for real-time computation, which is a crucial asset to emergent operation. The theoretical foundations for the feasibility of the trajectory were presented, and the numerical simulations further validated the practicality of the trajectory. The proposed motion planner is compatible with many existing FTC schemes that signify the promising potential of the synergy between the planner and the controller. In that regard, several future works are planned as follows.

- Further improving the safety or tracking performance of the proposed motion planner by addressing the optimization problem for the safest ϕ_{PE} (Remark 4) and/or the optimal design parameter a (Remark 5);

- Accounting for the different types of quadrotor faults in the planner, including the complete loss of two or more rotors, the mix of complete and partial loss of rotors, etc.;

- Cultivating the FTC mechanism such that more harmonizes with the proposed motion planner, thereby enhancing the overall fault tolerance capability, for example addressing extra robustness against the disturbance and noise, etc.;

- Compiling the planner and the FTC, together with other key algorithm modules such as state estimator and fault detector, into the fully integrated solution for tolerating the fault so that can be eventually deployed; and

- Conducting real-world experiments to validate the proposed planner, while properly considering the hardware factors that affect the execution of the planner, such as time delay in actuation.

Acknowledgements

Appendix A. Proof of Lemma 1

Proof. The FTC mechanism is based on appropriate design of ν followed by $\mathbf{u}^{\text{FTC}}$ stabilizing $\{\phi, \theta, z\}$ of the faulty quadrotor. Different forms of ν are presented by different FTC techniques

as listed in Table A.6.

Controller	Auxiliary control input ν
PD [14]	$[\ddot{\phi}_d \ \ddot{\theta}_d \ \ddot{z}_d]^{\mathrm{T}} + K_{\mathrm{D}}\dot{\mathbf{e}} + K_{\mathrm{P}}\mathbf{e}$
SMC [12]	$[\ddot{\phi}_d \ \ddot{\theta}_d \ \ddot{z}_d]^{\mathrm{T}} + \lambda_{\mathrm{SMC}}\dot{\mathbf{e}} + K_{\mathrm{SMC}}\mathrm{sign}[s_\phi, s_\theta, s_z]^{\mathrm{T}}$
BS [15]	$[\ddot{\hat{\phi}}_d \ \ddot{\hat{\theta}}_d \ \ddot{\hat{z}}_d]^{\mathrm{T}} + K_{\mathrm{BSC}}\hat{\mathbf{e}} + \mathbf{e}$

Table A.6: Auxiliary control designs by different FTC techniques

Here, K_{P}, K_{D}, K_{SMC}, λ_{SMC}, $K_{\mathrm{BSC}} > 0$ are gain parameters for the individual control designs. $s_\phi = \dot{e}_\phi + \lambda_{\mathrm{SMC}}e_\phi$, $s_\theta = \dot{e}_\theta + \lambda_{\mathrm{SMC}}e_\theta$, $s_z = \dot{e}_z + \lambda_{\mathrm{SMC}}e_z$ are the sliding surfaces, and $\mathrm{sign}(\cdot)$ is the signum function for SMC. The hat symbol $\hat{\cdot}$ denotes that the command filter is used for estimating the virtual control derivative in BSC. Evidently, all the listed auxiliary control designs are akin to (13), resulting in $\mathbf{u}^{\mathrm{FTC}}$ by (12). If $\mathbf{u}^{\mathrm{FTC}} \in \mathbf{U}$, we can substitute (12) for $\mathbf{u}$ in (11), which yields the following tracking error dynamics:

$$\ddot{\mathbf{e}} = -K_1\dot{\mathbf{e}} - K_2\mathbf{e}. \tag{A.1}$$

Now, let us define a positive definite function V of the tracking errors and their derivatives.

$$V(\mathbf{e}, \dot{\mathbf{e}}) := \frac{1}{2}(\epsilon K_1 + K_2)\mathbf{e}^{\mathrm{T}}\mathbf{e} + \epsilon\dot{\mathbf{e}}^{\mathrm{T}}\mathbf{e} + \frac{1}{2}\dot{\mathbf{e}}^{\mathrm{T}}\dot{\mathbf{e}}$$

where $1 \gg \epsilon > 0$ is small value. Then, the time derivative of V is:

$$\dot{V}(\mathbf{e}, \dot{\mathbf{e}}) = (\epsilon K_1 + K_2)\mathbf{e}^{\mathrm{T}}\dot{\mathbf{e}} + \epsilon\ddot{\mathbf{e}}^{\mathrm{T}}\dot{\mathbf{e}} + \epsilon\dot{\mathbf{e}}^{\mathrm{T}}\dot{\mathbf{e}} + \ddot{\mathbf{e}}^{\mathrm{T}}\dot{\mathbf{e}}. \tag{A.2}$$

And plugging (A.1) to (A.2) gives:

$$\dot{V}(\mathbf{e}, \dot{\mathbf{e}}) = -\epsilon K_2\mathbf{e}^{\mathrm{T}}\mathbf{e} - (K_1 - \epsilon)\dot{\mathbf{e}}^{\mathrm{T}}\dot{\mathbf{e}}.$$

Since is ϵ is sufficiently small such that $K_1 > \epsilon$, this immediately follows that:

$$\dot{V}(\mathbf{e}, \dot{\mathbf{e}}) < 0, \quad \forall \mathbf{e}, \dot{\mathbf{e}} \neq \mathbf{0}. \tag{A.3}$$

Thus, if $\mathbf{u}^{\mathrm{FTC}} \in \mathbf{U}$ so that $\mathbf{u} \equiv \mathbf{u}^{\mathrm{FTC}}$ and (A.1) hold, the existence of the positive definite Lyapunov

38

function V satisfying (A.3) guarantees the asymptotic stability of **e** by the Lyapunov theorem.

∎

Appendix B. Derivation of Eq. (21)

Proof. At the PE, the dynamics of yaw rate r from (3) can be rewritten as $(-k_r r_{\mathrm{PE}} + u_{r,\mathrm{PE}})/I_z = 0$ which gives the relationship between r_{PE} and $u_{r,\mathrm{PE}}$:

$$r_{\mathrm{PE}} = \frac{u_{r,\mathrm{PE}}}{k_r}. \tag{B.1}$$

Recalling (6), $u_{r,\mathrm{PE}}$ can be determined by:

$$u_{r,\mathrm{PE}} = b\left(u_{f,\mathrm{PE}} - \frac{2}{l}u_{p,\mathrm{PE}}\right) \tag{B.2}$$

where $u_{f,\mathrm{PE}}$ and $u_{p,\mathrm{PE}}$ at can be derived from (24):

$$u_{f,\mathrm{PE}} = \frac{mg}{\cos(\phi_{\mathrm{PE}})}, \qquad u_{p,\mathrm{PE}} = (I_z - I_y)r_{\mathrm{PE}}^2 \tan(\phi_{\mathrm{PE}}). \tag{B.3}$$

Substituting (B.2) and (B.3) to (B.1), we obtain a quadratic equation in terms of r_{PE}:

$$\frac{2}{l}(I_z - I_y)r_{\mathrm{PE}}^2 \tan(\phi_{\mathrm{PE}}) + \frac{k_r}{b}r_{\mathrm{PE}} - \frac{mg}{\cos(\phi_{\mathrm{PE}})} = 0. \tag{B.4}$$

Solving (B.4) and using the positive solution results:

$$r_{\mathrm{PE}} = \frac{-\frac{k_r l}{b} + l\sqrt{(\frac{k_r}{b})^2 + \frac{8mg(I_z - I_y)\tan(\phi_{\mathrm{PE}})}{l\cos(\phi_{\mathrm{PE}})}}}{4(I_z - I_y)\tan(\phi_{\mathrm{PE}})}. \tag{B.5}$$

One can notice that (B.5) has a singularity when $\phi_{\mathrm{PE}} = 0$. For this case, we set $\phi_{\mathrm{PE}} = 0$ into (B.3) resulting in:

$$u_{f,\mathrm{PE}} = mg, \qquad u_{p,\mathrm{PE}} = 0. \tag{B.6}$$

Substituting (B.2) and (B.6) to (B.1) yields the solution for r_{PE} when $\phi_{\mathrm{PE}} = 0$:

$$r_{\mathrm{PE}} = \frac{bmg}{k_r}. $$

∎

39

Appendix C. Proof of Proposition 1

Proof. Applying the boundary condition (26) to the trajectory (25) results in the following system of equations:

$$
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 2 & 0 & 0 & 0 \\
1 & T & T^2 & T^3 & T^4 & T^5 \\
0 & 1 & 2T & 3T^2 & 4T^3 & 5T^4 \\
0 & 0 & 2 & 6T & 12T^2 & 20T^3
\end{bmatrix}
\begin{bmatrix}
a_0 \\ a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5
\end{bmatrix}
=
\begin{bmatrix}
\phi(t_0) \\ \dot{\phi}(t_0) \\ \ddot{\phi}(t_0) \\ \phi_{\text{PE}} \\ 0 \\ 0
\end{bmatrix}.
\tag{C.1}
$$

Solving (C.1), the coefficients of (25) that comply with the boundary condition can be obtained.

∎

Appendix D. Proof of Lemma 2

Proof. To determine the upper and lower bounds of $r(t)$, namely $\bar{r}(t)$ and $\underline{r}(t)$, we turn to the dynamics of the yaw rate as described by (3). Specifically, these bounds can be extracted by employing the constant upper and lower limits of $u_r(t)$, represented as $\bar{u}_r$ and $\underline{u}_r$, respectively. This relationship is captured by the following differential equations:

$$
\dot{\bar{r}} = (-k_r \bar{r} + \bar{u}_r)/I_z, \quad \dot{\underline{r}} = (-k_r \underline{r} + \underline{u}_r)/I_z.
\tag{D.1}
$$

With the aforementioned bounds $\bar{u}_r$ and $\underline{u}_r$ computed, we can solve (D.1) to achieve:

$$
\bar{r}(t) = \frac{\bar{u}_r}{k_r}\left(1 - e^{-\frac{k_r}{I_z}t}\right), \quad \underline{r}(t) = \frac{\underline{u}_r}{k_r}\left(1 - e^{-\frac{k_r}{I_z}t}\right).
\tag{D.2}
$$

Utilizing (6) and (24), the upper bound of $u_r(t)$ can be computed as follows:

$$
\bar{u}_r = b\left(\max(u_f(t)) - \frac{2}{l}\min(u_p(t))\right) = \frac{bmg}{\cos\left(\max(|\phi_a(t)|)\right)}.
\tag{D.3}
$$

On the other hand, deducing the lower bound for $u_r(t)$ is more involved due to $\max(u_p(t))$ being unknown. Hence, the constant upper bound of $u_p(t)$ is obtained instead of $\max(u_p(t))$. By utilizing the second inequality condition from (24) and $\max(\bar{r}(t)) = \bar{u}_r/k_r$ from (D.2) and (D.3), the following upper bound is achieved:

$$\bar{u}_p = I_x\left(\frac{I_z - I_y}{I_x}(\frac{\bar{u}_r}{k_r})^2 \tan\big(\max(\phi_a(t))\big) + \ddot{\phi}_a(t_{c1})\right).$$

We can then proceed to compute the lower bound of $u_r(t)$:

$$\underline{u}_r = b\left(\min(u_f(t)) - \frac{2}{l}\bar{u}_p\right) = b\left(\frac{mg}{\cos\big(\min(|\phi_a(t)|)\big)} - \frac{2}{l}\bar{u}_p\right).$$

$$\blacksquare$$

Appendix E. Proof of Proposition 2

Proof. The second inequality condition (24) at the PE state yields $0 \le (I_z - I_y)r_{\text{PE}}^2 \tan\phi_{\text{PE}} \le l\bar{F}$ from which we can deduce the lower bound for the roll angle at the PE state:

$$0 \le \phi_{\text{PE}}. \tag{E.1}$$

By leveraging the inequality condition (E.1), along with Assumption 1, and the ASMP trajectory given by (25) with coefficients (27), we can obtain $\max(\phi_a(t)) = \phi_{\text{PE}}$. Drawing the first inequality condition from (24), we can deduce the upper bound of the roll angle at the PE:

$$\phi_{\text{PE}} \le \cos^{-1}(\frac{mg}{2\bar{F}}).$$

The second inequality condition in (24) is separated into upper and lower bounds by applying the bounds of $r(t)$ from Lemma 2:

$$I_x\left(\frac{I_z - I_y}{I_x}\bar{r}(t)^2 \tan(\phi_a(t)) + \ddot{\phi}_a(t_{c1})\right) \le l\bar{F},$$
$$0 \le \frac{I_z - I_y}{I_x}\underline{r}(t)^2 \tan(\phi_a(t)) + \ddot{\phi}_a(t_{c3}). \tag{E.2}$$

41

Given the ASMP trajectory with coefficients (27), the critical time incidents defined in (28) can be computed by:

$$t_{c1} = (3 - \sqrt{3})T/6,$$

$$t_{c2} = T/2,$$

$$t_{c3} = (3 + \sqrt{3})T/6.$$

To design the trajectory that always guarantees the first inequality condition from (E.2), the maximum values for each element are plugged in as:

$$I_x\left(\frac{I_z - I_y}{I_x}(\frac{\bar{u}_r}{k_r})^2 \tan(\phi_{\mathrm{PE}}) + \ddot{\phi}_a(t_{c1})\right) \leq l\bar{F}.$$

It is worth noting that $u_p(t)$ monotonically increases, so $\frac{I_z-I_y}{I_x}\underline{r}(t)^2 \tan(\phi_a(t))$ at t_{c2} must be greater than the minimum value of $\ddot{\phi}_a(t)$ to guarantee the second inequality condition from (E.2):

$$0 \leq \frac{I_z - I_y}{I_x}\underline{r}(t_{c2})^2 \tan(\phi_a(t_{c2})) + \ddot{\phi}_a(t_{c3}).$$

∎

Appendix F. Proof of Proposition 3

Proof. The yaw angle during $[t_p, t_f]$ can be approximated as:

$$\psi(t_p) \approx \psi(t_0) + \dot{\psi}_{\mathrm{PE}}(t_p - t_0), \quad \psi(t) \approx \psi(t_p) + \dot{\psi}_{\mathrm{PSMP}}(t - t_p). \tag{F.1}$$

Substituting (F.1) and u_f from (31) into (32) yields:

$$
\begin{aligned}
\dot{v}_x(t) &= -\frac{k_t}{m}v_x(t) + g\tan(\phi_d)\sin(\psi(t_p) + \dot{\psi}_{\mathrm{PSMP}}(t - t_p)), \\
\dot{v}_y(t) &= -\frac{k_t}{m}v_y(t) - g\tan(\phi_d)\cos(\psi(t_p) + \dot{\psi}_{\mathrm{PSMP}}(t - t_p)), \quad \forall t \in [t_p, t_f].
\end{aligned}
\tag{F.2}
$$

Due to the nonlinearity posed by $\tan(\phi_d)$, obtaining an exact analytical solution for the differential equation is difficult. To handle this, we employ the Taylor series expansion:

$$\tan(\phi_d) = \tan(\phi_{PE} + a[\cos(\omega_p(t - t_p)) - 1]) \approx \tan(\phi_{PE}) + \frac{a[\cos(\omega_p(t - t_p)) - 1]}{\cos^2(\phi_{PE})} \quad \text{(F.3)}$$

Substituting (F.3) into (F.2), the expanded differential equations can be obtained:

$$
\begin{aligned}
\dot{v}_x(t) = &-\frac{k_t}{m}v_x(t) + \left[g\tan(\phi_{PE}) - \frac{ga}{\cos^2(\phi_{PE})}\right]\sin(\dot{\psi}_{PSMP}(t - t_p) + \psi(t_p)) \\
&+ \frac{ga}{2\cos^2(\phi_{PE})}\left[\cos((\omega_p - \dot{\psi}_{PSMP})t - (\omega_p - \dot{\psi}_{PSMP})t_p - \psi(t_p) + \frac{\pi}{2})\right. \\
&\left. - \cos((\omega_p + \dot{\psi}_{PSMP})t - (\omega_p + \dot{\psi}_{PSMP})t_p + \psi(t_p) + \frac{\pi}{2})\right], \\
\dot{v}_y(t) = &-\frac{k_t}{m}v_y(t) - \left[g\tan(\phi_{PE}) - \frac{ga}{\cos^2(\phi_{PE})}\right]\cos(\dot{\psi}_{PSMP}(t - t_p) + \psi(t_p)) \\
&- \frac{ga}{2\cos^2(\phi_{PE})}\left[\sin((\omega_p - \dot{\psi}_{PSMP})t - (\omega_p - \dot{\psi}_{PSMP})t_p - \psi(t_p) + \frac{\pi}{2})\right. \\
&\left. + \sin((\omega_p + \dot{\psi}_{PSMP})t - (\omega_p + \dot{\psi}_{PSMP})t_p + \psi(t_p) + \frac{\pi}{2})\right], \quad \forall t \in [t_p, t_f].
\end{aligned}
\quad \text{(F.4)}
$$

Recalling (19), the orbital period of the faulty quadrotor at PSMP can be approximated as $T_{PSMP} \approx \frac{2\pi}{|\dot{\psi}_{PSMP}|}$. This implies that the macroscopic behavior of the faulty quadrotor will be straight and level flight only if the condition $\omega_p = \dot{\psi}_{PSMP}$ holds. Incorporating this condition, (F.4) can be reformulated as follows:

$$
\begin{aligned}
\dot{v}_x(t) = &-\frac{k_t}{m}v_x(t) + \left[g\tan(\phi_{PE}) - \frac{ga}{\cos^2(\phi_{PE})}\right]\sin(\dot{\psi}_{PSMP}(t - t_p) + \psi(t_p)) \\
&- \frac{ga}{2\cos^2(\phi_{PE})}\left[\cos(2\dot{\psi}_{PSMP}(t - t_p) + \psi(t_p) + \frac{\pi}{2}) + \cos(\psi(t_p) - \frac{\pi}{2})\right], \\
\dot{v}_y(t) = &-\frac{k_t}{m}v_y(t) - \left[g\tan(\phi_{PE}) - \frac{ga}{\cos^2(\phi_{PE})}\right]\cos(\dot{\psi}_{PSMP}(t - t_p) + \psi(t_p)) - \frac{ga}{2\cos^2(\phi_{PE})} \\
&\left[\sin(2\dot{\psi}_{PSMP}(t - t_p) + \psi(t_p) + \frac{\pi}{2}) + \sin(\psi(t_p) - \frac{\pi}{2})\right], \quad \forall t \in [t_p, t_f].
\end{aligned}
\quad \text{(F.5)}
$$

The solution to the differential equation (F.5) is shown as follows:

$$v_{\{x,y\}}(t) = \frac{1}{(A(A^2 + C^2)(A^2 + F^2))} \Big[A^5 v(t_p)e^{At} + A^4(He^{At} - E\cos(G + Ft) - B\sin(D + Ct) - H$$

$$+ Ee^{At}\cos(G) + Be^{At}\sin(D)) + A^3(C^2 v(t_p)e^{At} + F^2 v(t_p)e^{At} - BC\cos(D + Ct)$$

$$+ EF\sin(G + Ft)BCe^{At}\cos(D) - EFe^{At}\sin(G)) + A^2(-C^2 H - C^2 E\cos(G + Ft)$$

$$- BF^2\sin(D + Ct) - F^2 H + C^2 He^{At} + F^2 He^{At} + C^2 Ee^{At}\cos(G) + BF^2 e^{At}\sin(D))$$

$$+ A(C^2 EF\sin(G + Ft) + C^2 F^2 v(t_p)e^{At} - BCF^2\cos(D + Ct) - C^2 EFe^{At}\sin(G)$$

$$+ BCF^2 e^{At}\cos(D)) + C^2 F^2 He^{At} - C^2 F^2 H \Big], \quad \forall t \in [t_p, t_f].$$

with the coefficients for $v_x(t)$ and $v_y(t)$ defined as:

$$\begin{cases} A_x = -\dfrac{k_t}{m} \\[2mm] B_x = g\tan(\phi_{\text{PE}}) - \dfrac{ga}{\cos^2(\phi_{\text{PE}})} \\[2mm] C_x = \dot{\psi}_{\text{PSMP}} \\[2mm] D_x = -\dot{\psi}_{\text{PSMP}}t_p + \psi(t_p) \\[2mm] E_x = -\dfrac{ga}{2\cos^2(\phi_{\text{PE}})} \\[2mm] F_x = 2\dot{\psi}_{\text{PSMP}} \\[2mm] G_x = -2\dot{\psi}_{\text{PSMP}}t_p + \psi(t_p) + \dfrac{\pi}{2} \\[2mm] H_x = \dfrac{ga\cos\left(\psi(t_p) - \frac{\pi}{2}\right)}{2\cos^2(\phi_{\text{PE}})}, \end{cases} \qquad \begin{cases} A_y = -\dfrac{k_t}{m} \\[2mm] B_y = -\dfrac{ga}{2\cos^2(\phi_{\text{PE}})} \\[2mm] C_y = 2\dot{\psi}_{\text{PSMP}} \\[2mm] D_y = -2\dot{\psi}_{\text{PSMP}}t_p + \psi(t_p) + \dfrac{\pi}{2} \\[2mm] E_y = -g\tan(\phi_{\text{PE}}) + \dfrac{ga}{\cos^2(\phi_{\text{PE}})} \\[2mm] F_y = \dot{\psi}_{\text{PSMP}} \\[2mm] G_y = -\dot{\psi}_{\text{PSMP}}t_p + \psi(t_p) \\[2mm] H_y = \dfrac{ga\sin\left(\psi(t_p) - \frac{\pi}{2}\right)}{2\cos^2(\phi_{\text{PE}})}. \end{cases}$$

This provides the mean velocity of the faulty quadrotor at the PE state:

$$v_{x,\text{PSMP}} = \frac{mga\cos\left(\psi(t_p) - \frac{\pi}{2}\right)}{2k_t\cos^2(\phi_{\text{PE}})}, \quad v_{y,\text{PSMP}} = \frac{mga\sin\left(\psi(t_p) - \frac{\pi}{2}\right)}{2k_t\cos^2(\phi_{\text{PE}})}. \tag{F.6}$$

Expressing (F.6) in polar coordinates yields:

$$|v_{\text{PSMP}}| = \frac{mga}{2k_t\cos^2(\phi_{\text{PE}})}, \quad \angle v_{\text{PSMP}} = \psi(t_p) - \frac{\pi}{2}. \tag{F.7}$$

Considering the fact that the quadrotor is initially at the PE state and its orbit radius is substan-

tially smaller than the distance that the quadrotor will traverse during the PSMP phase, we can approximate $x(t_p) \approx x(t_0)$ and $y(t_p) \approx y(t_0)$. Using (F.1) and the direction of the velocity in (F.7), the time t_p directing the faulty quadrotor toward the safe zone can be computed as follows:

$$t_p = t_0 + \frac{1}{\dot{\psi}_{\text{PE}}} \left[\tan^{-1}\left(\frac{y_d - y(t_0)}{x_d - x(t_0)} \right) - \psi(t_0) + \frac{\pi}{2} \right] + \frac{2\pi n}{\dot{\psi}_{\text{PSMP}}}, \quad n \in \{0, 1, 2, \cdots\}, \quad \text{s.t. } t_p \geq t_0$$

which satisfies $\angle v_{\text{PSMP}} = \tan^{-1}\left(\frac{y_d - y(t_0)}{x_d - x(t_0)} \right)$. Based on the magnitude of the velocity in (F.7) and the time period $[t_p, t_f]$, the parameter a can be computed such that the faulty quadrotor reaches the safe zone at t_f:

$$a = \frac{2k_t \cos^2(\phi_{\text{PE}})}{mg(t_f - t_p)} \sqrt{\left(x_d - x(t_p) \right)^2 + \left(y_d - y(t_p) \right)^2}.$$

∎

Appendix G. Proof of Proposition 4

Proof. Applying (30) $\forall t \in [t_p, t_f]$, the inequality condition regarding u_p in (31) can be rewritten as:

$$0 \leq I_x \left(\frac{I_z - I_y}{I_x} r_{\text{PSMP}}^2 \tan(\phi_{\text{PE}} + \phi_p) + \ddot{\phi}_p \right) \leq l\overline{F}, \quad \forall t \in [t_p, t_f]. \tag{G.1}$$

And the second-order time derivative of (33) is given by:

$$\ddot{\phi}_p(t) = -a\dot{\psi}_{\text{PSMP}}^2 \cos(\dot{\psi}_{\text{PSMP}}(t - t_p)). \tag{G.2}$$

Since (33) and (G.2) are sinusoidal signals, respectively having amplitudes a and $a\dot{\psi}_{\text{PSMP}}^2$ with offsets $-a$ and 0, their bounds are:

$$-2a \leq \phi_p \leq 0,$$
$$-a\dot{\psi}_{\text{PSMP}}^2 \leq \ddot{\phi}_p \leq a\dot{\psi}_{\text{PSMP}}^2. \tag{G.3}$$

Substituting the upper and lower bounds from (G.3) into (G.1), the upper and lower bounds of u_p at the PSMP phase can be determined as:

$$I_x \left(\frac{I_z - I_y}{I_x} r_{\text{PSMP}}^2 \tan(\phi_{\text{PE}}) + \dot{\psi}_{\text{PSMP}}^2 a \right) \le l\overline{F},$$
$$0 \le \frac{I_z - I_y}{I_x} r_{\text{PSMP}}^2 \tan(\phi_{\text{PE}} - 2a) - \dot{\psi}_{\text{PSMP}}^2 a. \tag{G.4}$$

However, it is not possible to solve for a in the second inequality of (G.4) as a closed form expression, due to the non-algebraic nature of the tangent function. Hence, the Taylor series expansion is applied as follows:

$$0 \le \frac{I_z - I_y}{I_x} r_{\text{PSMP}}^2 \left(\tan(\phi_{\text{PE}}) - 2a \sec^2(\phi_{\text{PE}}) \right) - \dot{\psi}_{\text{PSMP}}^2 a. \tag{G.5}$$

Rearranging the first inequalities of (G.4) and (G.5) with respect to the tuning parameter a gives:

$$a \le \frac{l\overline{F} - (I_z - I_y) r_{\text{PSMP}}^2 \tan(\phi_{\text{PSMP}})}{I_x \dot{\psi}_{\text{PSMP}}^2},$$
$$a \le \frac{(I_z - I_y) r_{\text{PSMP}}^2 \tan(\phi_{\text{PSMP}})}{2(I_z - I_y) r_{\text{PSMP}}^2 \sec^2(\phi_{\text{PSMP}}) - I_x \dot{\psi}_{\text{PSMP}}^2} \tag{G.6}$$

which can be rewritten as the inequality condition (35). ■

References

[1] H. Shraim, A. Awada, R. Youness, A survey on quadrotors: Configurations, modeling and identification, control, collision avoidance, fault diagnosis and tolerant control, IEEE Aerospace and Electronic Systems Magazine 33 (7) (2018) 14–33.

[2] G. K. Fourlas, G. C. Karras, A survey on fault diagnosis and fault-tolerant control methods for unmanned aerial vehicles, Machines 9 (9) (2021) 197.

[3] M. Saied, H. Shraim, C. Francis, A review on recent development of multirotor uav fault-tolerant control systems, IEEE Aerospace and Electronic Systems Magazine (2023).

[4] J. Jiang, X. Yu, Fault-tolerant control systems: A comparative study between active and passive approaches, Annual Reviews in control 36 (1) (2012) 60–72.

[5] R. C. Avram, X. Zhang, J. Muse, Nonlinear adaptive fault-tolerant quadrotor altitude and attitude tracking with multiple actuator faults, IEEE transactions on control systems technology 26 (2) (2017) 701–707.

[6] Y. Zou, K. Xia, Robust fault-tolerant control for underactuated takeoff and landing uavs, IEEE Transactions on Aerospace and Electronic Systems 56 (5) (2020) 3545–3555.

[7] A. Milhim, Y. Zhang, C.-A. Rabbath, Gain scheduling based pid controller for fault tolerant control of quad-rotor uav, in: AIAA Infotech@Aerospace 2010, 2010, p. 3530.

[8] X. Wang, S. Sun, E.-J. van Kampen, Q. Chu, Quadrotor fault tolerant incremental sliding mode control driven by sliding mode disturbance observers, Aerospace Science and Technology 87 (2019) 417–430.

[9] X. Zhang, Y. Zhang, C.-Y. Su, Y. Feng, Fault-tolerant control for quadrotor uav via backstepping approach, in: 48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition, 2010, p. 947.

[10] G. Xu, Y. Xia, D.-H. Zhai, W. Lyu, Adaptive finite-time attitude tracking control of quadrotor under actuator faults and external disturbances with guaranteed performance, International Journal of Adaptive Control and Signal Processing (2022).

[11] B. Yu, Y. Zhang, I. Minchala, Y. Qu, Fault-tolerant control with linear quadratic and model predictive control techniques against actuator faults in a quadrotor uav, in: 2013 Conference on Control and Fault-Tolerant Systems (SysTol), IEEE, 2013, pp. 661–666.

[12] Z. Hou, P. Lu, Z. Tu, Nonsingular terminal sliding mode control for a quadrotor uav with a total rotor failure, Aerospace Science and Technology 98 (2020) 105716.

[13] A.-R. Merheb, H. Noura, F. Bateman, Emergency control of ar drone quadrotor uav suffering a total loss of one rotor, IEEE/ASME Transactions on Mechatronics 22 (2) (2017) 961–971.

[14] W. Chung, H. Son, Fault-tolerant control of multirotor uavs by control variable elimination, IEEE/ASME Transactions on Mechatronics 25 (5) (2020) 2513–2522.

[15] J. Wang, F. Holzapfel, F. Peter, Comparison of nonlinear dynamic inversion and backstepping controls with application to a quadrotor, Council of European Aerospace Societies (CEAS), 2013, pp. 1245–1263.

[16] A. Lanzon, A. Freddi, S. Longhi, Flight control of a quadrotor vehicle subsequent to a rotor failure, Journal of Guidance, Control, and Dynamics 37 (2) (2014) 580–591.

[17] M. W. Mueller, R. D'Andrea, Relaxed hover solutions for multicopters: Application to algorithmic redundancy and novel vehicles, The International Journal of Robotics Research 35 (8) (2016) 873–889.

[18] G. P. S. Rible, N. A. A. Arriola, M. C. Ramos, Fail-safe controller architectures for quadcopter with motor failures, in: 2020 6th International Conference on Control, Automation and Robotics (ICCAR), IEEE, 2020, pp. 384–391.

[19] W. Jung, H. Bang, Fault and failure tolerant model predictive control of quadrotor uav, International Journal of Aeronautical and Space Sciences 22 (2021) 663–675.

[20] F. Nan, S. Sun, P. Foehn, D. Scaramuzza, Nonlinear mpc for quadrotor fault-tolerant control, IEEE Robotics and Automation Letters 7 (2) (2022) 5047–5054.

[21] Y. Beyer, M. Steen, P. Hecker, Incremental passive fault-tolerant control for quadrotors subjected to complete rotor failures, Journal of Guidance, Control, and Dynamics 46 (10) (2023) 2033–2042.

[22] C. Ke, K.-Y. Cai, Q. Quan, Uniform passive fault-tolerant control of a quadcopter with one, two, or three rotor failure, IEEE Transactions on Robotics (2023).

[23] M. Hosseinzadeh, I. Kolmanovsky, S. Baruah, B. Sinopoli, Reference governor-based fault-tolerant constrained control, Automatica 136 (2022) 110089.

[24] E. Garone, S. Di Cairano, I. Kolmanovsky, Reference and command governors for systems with constraints: A

survey on theory and applications, Automatica 75 (2017) 306–328.

[25] A. Chamseddine, Y. Zhang, C. A. Rabbath, C. Join, D. Theilliol, Flatness-based trajectory planning/replanning for a quadrotor unmanned aerial vehicle, IEEE Transactions on Aerospace and Electronic Systems 48 (4) (2012) 2832–2848.

[26] A. Chamseddine, D. Theilliol, Y. Zhang, C. Join, C.-A. Rabbath, Active fault-tolerant control system design with trajectory re-planning against actuator faults and saturation: application to a quadrotor unmanned aerial vehicle, International Journal of Adaptive Control and Signal Processing 29 (1) (2015) 1–23.

[27] B. Xie, J. Zhao, Y. Liu, Fault tolerant motion planning of robotic manipulators based on a nested rrt algorithm, Industrial Robot: An International Journal (2012).

[28] G. Zogopoulos-Papaliakos, G. C. Karras, K. J. Kyriakopoulos, A fault-tolerant control scheme for fixed-wing uavs with flight envelope awareness, Journal of Intelligent & Robotic Systems 102 (2) (2021) 1–33.

[29] Y. Song, L. He, D. Zhang, J. Qian, J. Fu, Neuroadaptive fault-tolerant control of quadrotor uavs: a more affordable solution, IEEE transactions on neural networks and learning systems 30 (7) (2018) 1975–1983.

[30] B. Gao, Y.-J. Liu, L. Liu, Adaptive neural fault-tolerant control of a quadrotor uav via fast terminal sliding mode, Aerospace Science and Technology (2022) 107818.

[31] M. Elfeky, M. Elshafei, A.-W. A. Saif, M. F. Al-Malki, Modeling and simulation of quadrotor uav with tilting rotors, International Journal of Control, Automation and Systems 14 (2016) 1047–1055.

[32] Y. Nam, K. Lee, H.-S. Shin, C. Kwon, Fault-tolerant-motion-planner, https://github.com/HMCL-UNIST/Fault-tolerant-motion-planner (2024).

[33] A. A. Kurzhanskiy, P. Varaiya, Ellipsoidal techniques for reachability analysis of discrete-time linear systems, IEEE Transactions on Automatic Control 52 (1) (2007) 26–38.

[34] M. Althoff, G. Frehse, A. Girard, Set propagation techniques for reachability analysis, Annual Review of Control, Robotics, and Autonomous Systems 4 (2021) 369–395.

[35] V. Bally, L. Caramellino, R. Cont, F. Utzet, J. Vives, Stochastic integration by parts and functional Itô calculus, Springer, 2016.

[36] Y. Lin, E. D. Sontag, A universal formula for stabilization with bounded controls, Systems & control letters 16 (6) (1991) 393–397.

[37] M. M. Nicotra, E. Garone, The explicit reference governor: A general framework for the closed-form control of constrained nonlinear systems, IEEE Control Systems Magazine 38 (4) (2018) 89–107.

[38] Y. Guan, K. Yokoi, O. Stasse, A. Kheddar, On robotic trajectory planning using polynomial interpolations, in: 2005 IEEE international conference on robotics and biomimetics-ROBIO, IEEE, 2005, pp. 111–116.

[39] M. Yue, X. Hou, X. Zhao, X. Wu, Robust tube-based model predictive control for lane change maneuver of tractor-trailer vehicles based on a polynomial trajectory, IEEE Transactions on Systems, Man, and Cybernetics: Systems 50 (12) (2018) 5180–5188.

[40] H. Hu, X. Feng, R. Quirynen, M. E. Villanueva, B. Houska, Real-time tube mpc applied to a 10-state quadrotor

model, in: 2018 American Control Conference (ACC), IEEE, 2018, pp. 3135–3140.

[41] M. Khan, M. Zafar, A. Chatterjee, Barrier functions in cascaded controller: Safe quadrotor control, in: 2020 American Control Conference (ACC), IEEE, 2020, pp. 1737–1742.

[42] S. Karaman, E. Frazzoli, Sampling-based algorithms for optimal motion planning, The international journal of robotics research 30 (7) (2011) 846–894.

[43] W. Giernacki, M. Skwierczyński, W. Witwicki, P. Wroński, P. Kozierski, Crazyflie 2.0 quadrotor as a platform for research and education in robotics and control engineering, in: 2017 22nd International Conference on Methods and Models in Automation and Robotics (MMAR), IEEE, 2017, pp. 37–42.

[44] D. A. Patterson, J. L. Hennessy, Computer organization and design ARM edition: the hardware software interface, Morgan kaufmann, 2016.

[45] L. Meier, D. Honegger, M. Pollefeys, Px4: A node-based multithreaded open source robotics framework for deeply embedded platforms, in: 2015 IEEE international conference on robotics and automation (ICRA), IEEE, 2015, pp. 6235–6240.