Journal of Engineering Technology and Applied Sciences, 2024 *e-ISSN:* 2548-0391
Received 1 September 2022, Accepted 28 August 2023 *Vol. 9, No. 1, 1-21*
Published online 29 April 2024, Research Article *doi: 10.30931/jetas.1169518*

Citation: Dastgerdi, K. A., Asadi, D., Chashmi, S. Y. N., Tutsoy, O., "Robust Adaptıve Control Based On Incremental Nonlınear Dynamıc Inversıon For A Quadrotor In The Presence Of Actuator Fault". Journal of Engineering Technology and Applied Sciences 9 (1) 2024 : 1-21.

ROBUST ADAPTIVE CONTROL BASED ON INCREMENTAL NONLINEAR DYNAMIC INVERSION FOR A QUADROTOR IN THE PRESENCE OF ACTUATOR FAULT[1]

Karim Ahmadi Dastgerdi[a], Davood Asadi[a*], Seyed Yaser Nabavi Chashmi[a], Onder Tutsoy[b]

[a*] *Department of Aerospace Engineering., Faculty of Aerospace Engineering, Adana Alparslan Turkes Science and Technology University, Adana, Turkey,*
*dasadihendoustani@gmail.com (*corresponding author), kadastgerdi@atu.edu.tr,*
synabavi @atu.edu.tr

[b] *Department of Engineering, Faculty of Electrical and Electronic Engineering*
Adana Alparslan Turkes Science and Technology University, Adana, Turkey, otutsoy@atu.edu.tr

Abstract

This paper presents a novel nonlinear robust adaptive trajectory tracking control architecture for stabilizing and controlling a quadrotor in the presence of actuator partial faults. The proposed control strategy utilizes an Incremental Nonlinear Dynamic Inversion (INDI) algorithm as the baseline controller in the inner loop and augments a nonlinear model reference adaptive controller in the outer loop to ensure robustness against unmodeled faults. Additionally, a modified PID controller is introduced in the most outer-loop to track the desired path. The effects of actuator faults are modeled by considering sudden variations in motor thrust and torques. To enhance the control algorithm's robustness, a projection operator is employed in the robust adaptive structure. Comparative performance evaluations with a previous successful algorithm implemented on a quadrotor model demonstrate that the proposed controller achieves full controllability of the faulty quadrotor in pitch, roll, and yaw channels in the presence of actuator partial faults up to 50%.

Keywords: Robust adaptive control, incremental nonlinear dynamic inversion, actuator fault

[1] This research is supported by the Scientific and Technological Research Council of Turkey (TÜBİTAK) under 3501 program, with project number [120M793].

1. Introduction

The use of Multi-rotor Unmanned Aerial Vehicles (UAVs) has generated significant interest among researchers due to their diverse applications across various fields such as surveillance, reconnaissance, agriculture, rescue operations, and mining. The versatility and capabilities of these UAVs have propelled them into the forefront of technological advancements. One of the outstanding research challenges in multirotor design is the requirement of a sophisticated control system that can cope with unexpected casualties like actuator failures [1,2]. Faults and failures are inevitable in complex systems like aircraft. Hence, scientists are working on fault-tolerant control strategies to safely land the aircraft in presence of faults or failures [3–5]. Fault-tolerant control (FTC) techniques have been proposed in several researches to recover the control of faulty aircraft [6–11]. Nonlinear L1 adaptive control [6], robust adaptive control [7], adaptive sliding mode control [8], Linear Parametric Variable (LPV) sliding mode control [9], optimal adaptive control [10], and Model Reference Adaptive Control (MRAC) [11] are some instances of direct fault-tolerant control algorithms. In addition to the direct methods, fault-detection and identification algorithms are also used in some references in the fault-tolerant control strategy [12]. Timely detection of the actuator failures and estimation of their severity plays an important role in avoiding crashes and leading to fast recovery for a safe landing. Fault-detection approaches can be categorized into model-based, signal-based, knowledge-based, and active diagnosis techniques [13]. Controlling multi-rotor Unmanned Aerial Vehicles (UAVs) poses a challenging task due to their nonlinear, highly-coupled, and underactuated dynamic systems. Moreover, the occurrence of actuator faults in multi-rotors raises concerns regarding their reliability and safety. Extensive research has been conducted to address the control and recovery of multi-rotors in the presence of motor faults or failures. These studies can be broadly categorized into two groups: partial actuator faults and complete loss of actuator effectiveness or actuator failure. Some studies focus on analyzing the impact of partial faults on rotor performance and propose fault-tolerant strategies, while others investigate the effects of motor failure and develop appropriate fault-tolerant control strategies. Within these research endeavors, certain studies integrate fault detection algorithms as part of their fault-tolerant control strategies, while others directly implement fault-tolerant control algorithms to govern the multi-rotor's behavior. The investigation of partial motor faults, which lead to a reduction in control effectiveness, has been a subject of interest in numerous research studies. Ref [14] introduced a fault-tolerant control strategy to control a quadcopter in case of a time-varying motor fault. The proposed fault-tolerant strategy includes fault detection and identification algorithm based on the controller outputs and the angular rates calculated by a discrete extended Kalman filter and a discrete nonlinear adaptive tracking controller. There are also several other researches, which have tried to control the quadrotor in presence of partial fault [2]. The sliding mode control technique has been applied in Ref. [15] as a passive fault-tolerant control method to control the quadrotor’s attitude considering partial rotor fault. An adaptive fuzzy system is used as a compensator to compensate for the estimation error of nonlinear functions and faulty parts. Ref. [16] applies a sliding mode disturbance observer inside the fault-tolerant sliding mode controller to control and improve the performance of the quadrotor with partial actuator fault. There are several researches regarding the controllability of multi-rotors in presence of rotor fault or failure, in which different configurations including quadrotor, hexarotor, and octarotors have been investigated to determine the status of controllability [17,18]. Among the aforementioned multi-rotors, quadrotors suffer more from rotor fault due to lack of actuator redundancy. Respecting the controllability of quadrotors, it is well known that failure of one rotor results in an uncontrollable system. Therefore, full attitude control of the quadrotor can be achieved for a maximum specific magnitude of the partial fault and is not achievable in presence of complete one rotor failure. In case of one rotor failure in quadrotors, controllability of the yaw state is scarified and the controller tries to control the roll and pitch angles of the quadrotor[19]. Various control methodologies have been addressed in literature the problem of complete loss of one or two rotors of the quadrotor [19–21]. A robust feedback linearization controller along with an H∞ loop shaping technique is adopted in Ref. [19] to achieve regulation of roll and pitch angles around the chosen working point. A nonlinear sensor-based fault-tolerant controller is developed in Refs. [21,22] to stabilize a quadrotor with failure of two opposing rotors in the high-speed flight condition. Ref [23] proposes a complete FTC design approach with fault detection and diagnosis (FDD) of a quadrotor in presence of a partial fault. Hexarotor seems to be more robust respecting motor failure because of having more actuators. Despite the higher numbers of motors concerning quadrotors, researchers demonstrated that standard hexarotors are not fully controllable in case of one motor failure, in which yaw control is lost if one engine is failed [24]. It is difficult to reach a controller that can cope with motor failures in the standard configurations, and most proposed controller algorithms in the literature are confined to reduced attitude control [25]. In the standard configuration of hexarotor (PNPNPN: P stands for rotation in the positive direction and N stands for rotation in the negative direction), all neighboring motors rotate in opposite directions. Non-standard configurations (PPNNPN) can maintain full controllability in presence of one rotor failure. Accordingly, Ref. [26] applies the composition of a Tau-observer and a disturbance based sliding mode controller on a non-standard configuration of hexarotor and investigated the fault detection and control of a hexarotor in presence of one and two motor failure with controlling the attitudes including the heading and keep the hovering flight to landing. It can be demonstrated that the non-standard configurations of hexarotor are fully controllable in 33% of up to two random motor failures [17]. In addition, some researches in the literature try to control the faulty multirotor to another appropriate configuration corresponding to the failure. A reconfiguration technique based on control allocation has been proposed to transform a quadcopter into a tricopter in Ref. [27]. The proposed approach can tolerate complete failure but requires an extra weight mounted on the opposite motor. In a similar strategy, Ref. [20] applied a backstepping control algorithm and transformed the quadrotor into a birotor for an emergency landing in presence of one motor failure.

Building upon the aforementioned discussions, this paper aims to explore the full controllability of a quadrotor (roll, pitch, and yaw) in the presence of partial actuator faults. It is well-known from the literature that achieving full controllability becomes challenging in the case of complete loss of one motor's effectiveness or motor failure. To address this issue, a novel approach is proposed, utilizing the Incremental Nonlinear Dynamic Inversion (INDI) control algorithm augmented with a nonlinear robust adaptive controller to effectively control the quadrotor in the presence of a motor fault. The simulation results validate the excellent performance of the introduced control architecture. Additionally, for trajectory tracking purposes, a modified PID algorithm is employed as the third loop in the three-loop control strategy.

The subsequent sections of this paper are organized as follows: Section (II) presents the derivation of the quadrotor's nonlinear dynamic equation of motion. Section (III) details the controller architecture, encompassing the INDI algorithm, the robust adaptive controller approach, and the application of the PID controller. In Section (IV), numerical results are provided, evaluating the controller's performance and conducting comparisons. Finally, the conclusion section offers a brief summary of the key findings obtained throughout the study.

2. Mathematical model

This section presents the quadrotor model and the corresponding equations of motion, taking into account disturbances arising from unknown dynamics. Additionally, the motor model and the motor mixer equations are described in detail.

2.1 Quadrotor frame

The S500 frame with the EMAX2212/ 820KV motors is selected as the plant model in this research. The quadrotor parameters, which are used in this paper are given in Table 1 [12].

Table 1. Quadrotor Frame Specifications

Quadrotor Parameters	Values
Mass, *m*	1.59 *kg*
Thrust Parameter, *b*	$2.02 \times 10^{-7}\ N/rpm^2$
Drag Parameter, *d*	$4.18 \times 10^{-9}\ Nm/rpm^2$
Moment arm (C.G to motor distance), *l*	0.243 *m*
Moment of Inertia about the *x*-axis, I_{xx}	$0.0213\ kg.m^2$
Moment of Inertia about the *y*-axis, I_{yy}	$0.0221\ kg.m^2$
Moment of Inertia about the *z*-axis, I_{zz}	$0.028\ kg.m^2$
Translational drag coefficients, k_x, k_y	$5.5e-4\,N/m/s$
Translational drag coefficients, k_z	$6.3e-4\,N/m/s$
Rotational drag coefficients, k_φ, k_θ	$5.5e-4\,N/rad/s$
Rotational drag coefficients, k_ψ	$6.35e-4\,N/rad/s$
Total rotational moment of inertia, J_T	$6.8\times 10^{-5}\ kg.m^2$

2.2 Dynamic equations

The translational and rotational equations of the quadrotor in the body frame are presented in Eqs. (1) and (2), respectively [28]. As depicted in **Figure 1**, the quadrotor consists of four motors. Number one and two motors rotate counter-clockwise with velocities Ω_1, Ω_2, respectively, whereas the other two motors (number 3 and 4) rotate in the opposite (clockwise) direction with velocities Ω_3, Ω_4.

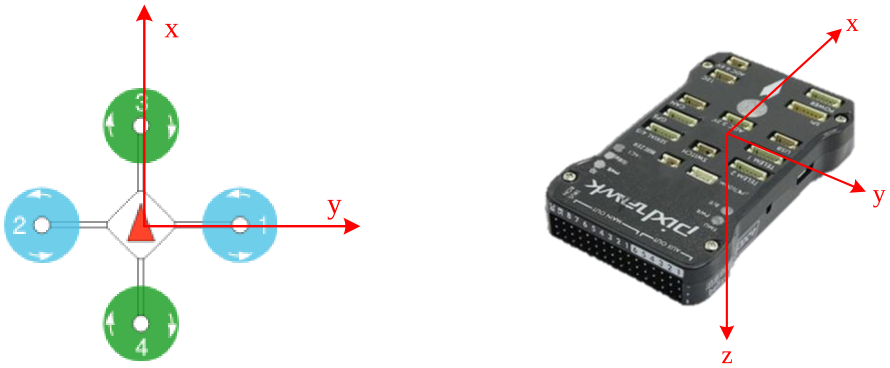


Figure 1 Schematic representation of quadrotor

- **Translational dynamics**

$$\begin{aligned}
\ddot{x} &= -\frac{T}{m}(\sin\psi\sin\phi + \cos\psi\sin\theta\cos\phi) + f_x^{drag} + f_x^{w} \\
\ddot{y} &= -\frac{T}{m}(-\cos\psi\sin\phi + \sin\psi\sin\theta\cos\phi) + f_y^{drag} + f_y^{w} \\
\ddot{z} &= +g - (\cos\theta\cos\phi)\frac{T}{m} + f_z^{drag} + f_z^{w}
\end{aligned} \tag{1}$$

- **Rotational dynamics**

$$\begin{aligned}
\dot{p} &= \frac{I_{yy} - I_{zz}}{I_{xx}} q\, r + \frac{M_x}{I_{xx}} + \tau_p^{drag} + \tau_p^{gyro} + \tau_p^{wind} \\
\dot{q} &= \frac{I_{zz} - I_{xx}}{I_{yy}} p\, r + \frac{M_y}{I_{yy}} + \tau_q^{drag} + \tau_q^{gyro} + \tau_q^{wind} \\
\dot{r} &= \frac{I_{xx} - I_{yy}}{I_{zz}} p\, q + \frac{M_z}{I_{zz}} + \tau_r^{drag} + \tau_r^{wind}
\end{aligned} \tag{2}$$

- **Euler Equations**

$$\begin{aligned}
\dot{\varphi} &= p + q\sin\varphi\tan\theta + r\cos\varphi\tan\theta \\
\dot{\theta} &= q\cos\varphi - r\sin\varphi \\
\dot{\psi} &= \frac{1}{\cos\theta}\left[q\sin\varphi + r\cos\varphi\right]
\end{aligned} \tag{3}$$

where *x*, *y*, and *z* are the position of quadrotor center of mass in the inertial frame and ψ, θ, φ are the Euler angles, which represent the body frame rotation concerning the inertial frame. *I* I_{xx}, I_{yy}, and I_{zz} are the moments of inertia in *x*, *y*, and *z-direction*, respectively, *m* is the system mass, *l* is the distance between the center of the mass and the motors, and *g* is the gravitational acceleration. The quadrotor inputs are represented by T , M_x , M_y , M_z , which are the total thrust force (T) generated by propellers in *z*-direction and moments about *x*, *y*, *z* axes, respectively. (See Part C.). The terms f_x^{drag} , f_y^{drag} , f_z^{drag} , τ_p^{drag} , τ_q^{drag} , and τ_r^{drag} are the drag forces and moments produced by the quadrotor's frame, which are expressed as $f_x^{drag} = -\frac{k_x}{m}\dot{x}$, $f_y^{drag} = -\frac{k_y}{m}\dot{y}$, $f_z^{drag} = -\frac{k_z}{m}\dot{z}$, $\tau_p^{drag} = -\frac{k_\varphi}{I_{xx}}p^2$, $\tau_q^{drag} = -\frac{k_\theta}{I_{yy}}q^2$, $\tau_r^{drag} = -\frac{k_\psi}{I_{zz}}r^2$. The constant parameters k_x, k_y, k_z are translational drag coefficients, and $k_\varphi, k_\theta, k_\psi$ are rotational drag coefficients, which are considered with values according to Table 1. Moments produced by the gyroscopic effect of the rotors around *x* and *y* axes are presented by τ_p^{gyro} , τ_q^{gyro} , which are expressed as $\tau_p^{gyro} = \frac{J_T}{I_{xx}} q\Omega$ and $\tau_q^{gyro} = -\frac{J_T}{I_{yy}} p\,\Omega$, in which J_T is the moment of inertia of each motor and Ω represents the propellers total speed as below:

$$\Omega = \Omega_1 - \Omega_2 + \Omega_3 - \Omega_4 \tag{4}$$

The terms f_x^{w} , f_y^{w} , f_z^{w} and τ_p^{wind} , τ_q^{wind} , τ_r^{wind} are the forces and moments, which are produced by the effect of wind. The wind model can be composed of different elements of the wind including the mean wind, wind gust, and turbulence [29]. For the purpose of simulation, this paper considers the Dryden turbulence model, which is the stochastic component of the wind and is inherently dependent on the quadrotor's states.

2.3 Rotor dynamics

The thrust generated by the motors is modeled as a first-order system to account for the rotor dynamics for variation of rotational speed:

$$u_{i_c} = K\frac{\omega_0}{S+\omega_0}u_i \tag{5}$$

where *S* is the Laplace variable, u_{i_c} is the *i*-th motor input which is the PWM reference signal to the motors, *K* is the motor gain, and ω_0 is the bandwidth of the motors. The motors' thrust force and torque depend on the rotational velocity, propeller diameter, as well as the aerodynamics characteristics of blades as below:

$$\begin{aligned} T_i &= C_t\rho\Omega_i^2 D^4 = b\Omega_i^2 \\ Q_i &= C_d\,\rho\Omega_i^2 D^5 = kT_i = d\,\Omega_i^2\,,\ k = 2.07e-2\ \ m \end{aligned} \tag{6}$$

Where *T_i* and *Q_i* are the *i*-th motor's aerodynamic thrust force and torque, respectively, C_t, C_d are thrust and drag coefficients, ρ is the air density, Ω_i is the rotational speed of each motor in *rpm*, and *D* is the propeller diameter. The numerical values of *b* and *d* are introduced in Table 1. Accordingly, the actuation inputs in the body frame are expressed based on the rotational speeds as follows:

$$\mathbf{U} = \mathbf{K}_{\Omega 2U}\,\mathbf{\Omega} \tag{7}$$

where $\mathbf{U} = \left[T, M_x, M_y, M_z\right]^T$, $\mathbf{K}_{\Omega 2U} = \begin{bmatrix} b & b & b & b \\ -bl & bl & 0 & 0 \\ 0 & 0 & bl & -bl \\ d & d & -d & -d \end{bmatrix}$, and $\mathbf{\Omega} = \left[\Omega_1^2, \Omega_2^2, \Omega_3^2, \Omega_4^2\right]^T$.

The autopilot outputs (**U**) must be translated into each motor inputs to send the signal to the quadrotor speed controls, then apply the related PWM signal to each quadrotor's motor.

a) Motor mixer

The motor mixer plays a crucial role in determining the rotational speeds of each rotor based on the intermediate autopilot outputs (**U**). By inverting Equation (7), the motor mixer expression can be derived as follows:

$$\mathbf{\Omega} = \mathbf{K}_{U2\Omega}\,\mathbf{U}, \mathbf{K}_{U2\Omega} = inv(\mathbf{K}_{\Omega 2U}) \tag{8}$$

b) Motor thrust and speed limitations

Based on the motor type, propeller size, battery specifications, the maximum speed of each motor ($\Omega_{\max}$) is found to be 6250 rpm. Accordingly, the maximum thrust of each motor is 7.89*N*, and therefore, the minimum rotational speed of the motors will be confined based on the hover rotational speed. Therefore, the minimum rotational speed ($\Omega_{\min}$) of motors in fault scenarios is specified based on the quadrotor weight and thrust of each motor, which is considered to be a little less than the rotational velocity of motors in hover conditions:

$$W = b\sum_{1}^{4}\Omega_i^{\,2} \Rightarrow \Omega_{i_{hover}} = \sqrt{\frac{15.39}{4\times 2.02e-7}} = 4364\ rpm\ \Rightarrow \Omega_{i_{\min}} = 4364\ rpm \tag{9}$$

The maximum and minimum bounds of the controller output before converting to the input of each motor is:

$$\begin{aligned}
&T = 4b\Omega^2 && \Rightarrow \quad 15.39 \le T \le 31.56 \\
&M_{x_{\max}} = bl(\Omega^2_{\max} - \Omega^2_{\min}) && \Rightarrow \quad -0.983 \le M_x \le 0.983 \\
&M_{y_{\max}} = bl(\Omega^2_{\max} - \Omega^2_{\min}) && \Rightarrow \quad -0.983 \le M_y \le 0.983 \\
&M_{z_{\max}} = d(\Omega^2_{\max} + \Omega^2_{\max} - \Omega^2_{\min} - \Omega^2_{\min}) && \Rightarrow \quad -0.167 \le M_z \le 0.167
\end{aligned} \tag{10}$$

2.4 Motor fault modeling

When there is a degradation in motor performance or damage to the rotor, it can be interpreted as a partial fault in the actuator compared to its normal operational condition. This partial fault on the i-th actuator results in a loss of thrust, consequently causing undesired roll, pitch, and yawing moments. Therefore, the impact of partial fault on the thrust force and moment is treated as a parametric uncertainty, which can be represented as follows:

$$\begin{aligned}
T_{if} &= T_i + \Delta T_i = b\Omega_i^2 + \Delta b\,\Omega_i^2,\ \Delta b = -f_i b \\
Q_{if} &= Q_i + \Delta Q_i = d\,\Omega_i^2 + \Delta d\,\Omega_i^2,\ \Delta d = -f_i d
\end{aligned} \tag{11}$$

where $\Delta b,\ \Delta d$ are bounded variation of motor effectiveness respecting its nominal values and can be represented as $-b \le \Delta b \le 0,\ -d \le \Delta d \le 0$ and f_i is the *i*-th motor fault. Therefore, the actual signal ($\mathbf{u}$) generated by the faulty actuator ($\mathbf{u}_f$) is as follows:

$$\begin{aligned}
&\mathbf{u}_f(t) = (1-\mathbf{\Gamma})\mathbf{u}(t)\ ,\ \mathbf{u}(t)^T = [T_1,\ T_2,\ T_3,\ T_4] \\
&\mathbf{\Gamma} = \begin{cases} 0 & t < t_f \\ diag(f_1,\ f_2,\ f_3,\ f_4) & t > t_f \end{cases}
\end{aligned} \tag{12}$$

In the above equation, t_f is the time that fault occurs and $0 \le f_i < 1$, in which $f_i = 0$, $f_i = 1$ represent the healthy and the fully failed actuator, respectively.

3. Fault tolerant control strategy

In this section, the multiple-timescales approach is employed to separate the rotational and translational dynamics. This is done by assuming that the rotational dynamics evolve much faster compared to the translational dynamics. The block diagram of the controller, illustrating this separation, is depicted in Figure 2.

It is clear that classical controller algorithms do not have appropriate performance in presence of motor failure. To deal with fail conditions a cascade control algorithm is applied to the quadrotor. The attitude control loop is a robust adaptive controller based on INDI and the position control loop is the PID algorithm.

Based on the rotational dynamics according to Eq. (2), the nonlinear model of quadrotor can be transformed into an affine control model as below:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{U}_c \tag{13}$$

where $\mathbf{x} \in \Re^3$ is the vector of rotational velocities ($\mathbf{x} = [p,\ q,\ r]$); $\mathbf{U}_c \in \Re^3$ is the controller output moments vector ($\mathbf{U}_c = [M_x, M_y,\ M_z]$); $\mathbf{f(x)} \in \Re^3$ and $\mathbf{g(x)} \in \Re^{3\times3}$ are differentiable matrices of state and input functions, respectively.

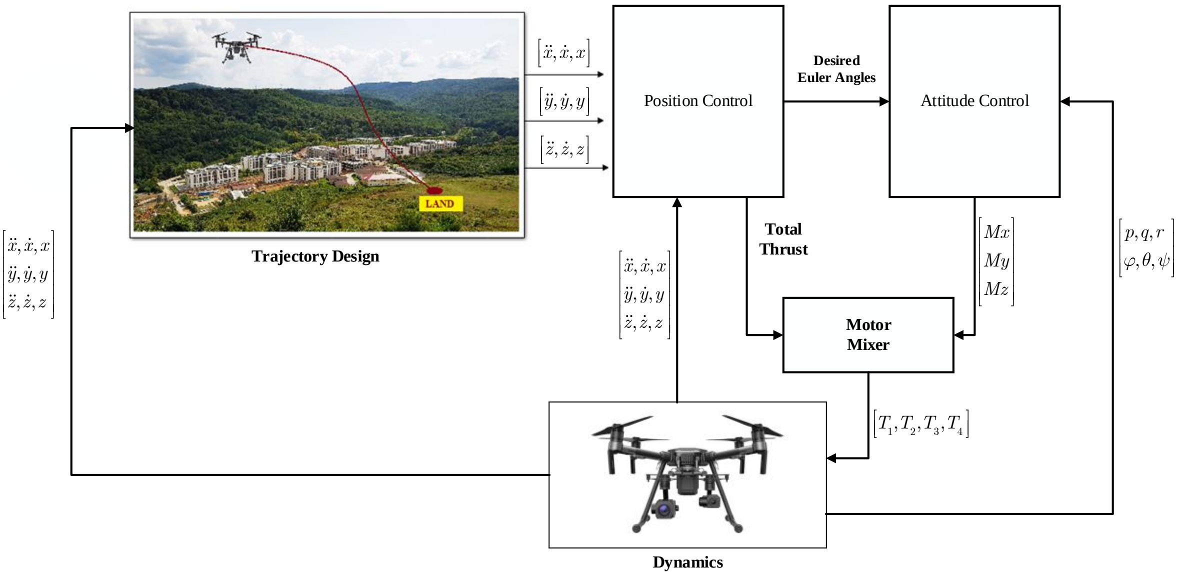


Figure 2 Total controller architecture

3.1 INDI controller design

Considering the rotational dynamics of the quadrotor based on Eq. (2), the Taylor series approach is applied to expand Eq. (2) while neglecting higher-order terms. Accordingly, Eq. (14) is obtained as below:

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{f}\ \ \mathbf{x}(t-Ts)\ \ + \mathbf{g}\left(\mathbf{x}(t-Ts)\right)\mathbf{U}_c(t-Ts)\\ &+ \frac{\partial}{\partial \mathbf{x}}\mathbf{f}(\mathbf{x})\Big|_{\substack{\mathbf{x}=\mathbf{x}(t-Ts)\\ \mathbf{U}_c=\mathbf{U}_c(t-Ts)}}\ \ \mathbf{x}(t)\text{ - }\mathbf{x}(t-Ts)\ \ + \frac{\partial}{\partial \mathbf{x}}\ \ g(\mathbf{x})\mathbf{U}_c\ \ \Big|_{\substack{\mathbf{x}=\mathbf{x}(t-Ts)\\ \mathbf{U}_c=\mathbf{U}_c(t-Ts)}}\ \ \mathbf{x}\text{ - }\mathbf{x}(t-Ts)\\ &+ \frac{\partial}{\partial \mathbf{U}_c}\mathbf{f}(\mathbf{x})\Big|_{\substack{\mathbf{x}=\mathbf{x}(t-Ts)\\ \mathbf{U}_c=\mathbf{U}_c(t-Ts)}}\ \ \mathbf{U}_c(t)\text{ - }\mathbf{U}_c(t-Ts)\ \ + \mathbf{g}\ \ \mathbf{x}(t-Ts)\ \ \mathbf{U}_c(t)\text{ - }\mathbf{U}_c(t-Ts)\end{aligned} \tag{14}$$

where T_s is the sampling time. The first part of Eq. (14), $\mathbf{f}\ \ \mathbf{x}(t-Ts)\ \ + \mathbf{g}\left(\mathbf{x}(t-Ts)\right)\mathbf{U}_c(t-Ts)$ is equal to $\dot{\mathrm{x}}(t-Ts)$. This part includes some terms, which can be calculated based on the onboard sensors at any instance of the flight time. The term $\dot{\mathrm{x}}(t-Ts)$ can be computed by taking derivative from rate gyros' outputs, which are the rotational speeds. In other words, the dynamic related terms $\mathbf{f}\ \ \mathbf{x}(t-Ts)\ \ + \mathbf{g}\left(\mathbf{x}(t-Ts)\right)\mathbf{U}_c(t-Ts)$ are replaced by the derivative of the sensor outputs. That is why this approach. i.e. INDI, is referred to as a sensor-based control strategy. The other part of Eq. (14)

$$\left[\frac{\partial}{\partial \mathbf{x}}\mathbf{f}(\mathbf{x})\Big|_{\substack{\mathbf{x}=\mathbf{x}(t-Ts)\\ \mathbf{U}_c=\mathbf{U}_c(t-Ts)}}\ \ \mathbf{x}(t)\text{ - }\mathbf{x}(t-Ts)\ \ + \frac{\partial}{\partial \mathbf{x}}\ \ g(\mathbf{x})\mathbf{U}_c\ \ \Big|_{\substack{\mathbf{x}=\mathbf{x}(t-Ts)\\ \mathbf{U}_c=\mathbf{U}_c(t-Ts)}}\ \ \mathbf{x}\text{ - }\mathbf{x}(t-Ts)\ \ + \frac{\partial}{\partial \mathbf{U}_c}\mathbf{f}(\mathbf{x})\Big|_{\substack{\mathbf{x}=\mathbf{x}(t-Ts)\\ \mathbf{U}_c=\mathbf{U}_c(t-Ts)}}\ \ \mathbf{U}_c(t)\text{ - }\mathbf{U}_c(t-Ts)\right]$$

can be neglected if the sampling time T_s is small. Thus, Eq. (14) can be rewritten as Eq.(15).

$$\dot{\mathbf{x}}(t) = \dot{\mathbf{x}}(t-Ts) + \mathbf{g}\ \ \mathbf{x}(t-Ts)\ \ \mathbf{U}_c(t)\text{ - }\mathbf{U}_c(t-Ts) \tag{15}$$

According to Eq.(15), the parameters in Eq. (2) can be rewritten as below:

$$\begin{aligned}\dot{p}(t) &= \dot{p}(t-Ts) + \frac{1}{I_{xx}}\ M_x(t) - M_x(t-Ts)\\ \dot{q}(t) &= \dot{q}(t-Ts) + \frac{1}{I_{yy}}\ M_y(t) - M_y(t-Ts)\\ \dot{r}(t) &= \dot{r}(t-Ts) + \frac{1}{I_{zz}}\ M_z(t) - M_z(t-Ts)\end{aligned} \tag{16}$$

As explained before, the angular acceleration terms are derived by taking derivatives from the angular rates. Since the sensor measurements from the gyroscope are naturally noisy due to disturbances induced by the vibrations of the motor or propeller on the vehicle's frame. Since differentiating the noisy signal amplifies the effect of noise on the output, the application of an appropriate filter is required. Accordingly, a second-order filter is adopted to be applied before differentiating the outputs of rate gyros [30]. The implemented filter in the form of a transfer function in the Laplace domain is given in Eq. (17). Satisfactory results are obtained from the filter with $\omega_n = 50\, rad/\sec$ and $\zeta = 0.55$. For the same application, other low-pass filters like the Butterworth filter can also be implemented.

$$C(s) = \frac{2500}{s^2 + 55s + 2500} \tag{17}$$

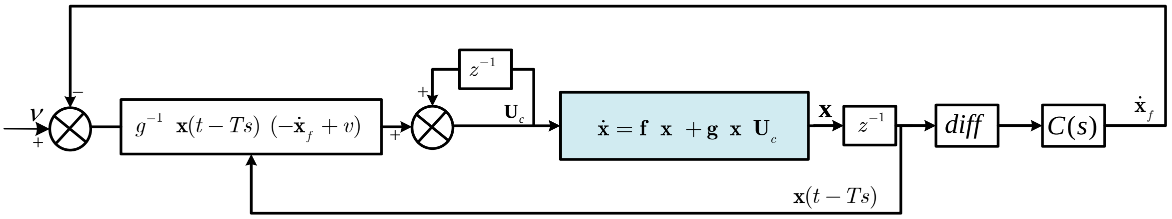
In the next step, the controller command should be computed corresponding to the INDI approach. Hence, by inversing Eq. (15) the control signal is obtained as below:

$$\mathbf{U}_c(t) = g^{-1}\ \mathbf{x}(t-Ts)\ (-\dot{\mathbf{x}}_f + v) + \mathbf{U}_c(t-Ts) \tag{18}$$

where:

$$\dot{\mathbf{x}}_f = L^{-1}(\dot{\mathbf{x}}(t-Ts)C(s)) \tag{19}$$

Where L^{-1} is the Laplace inverse operator, $\dot{\mathbf{x}}_f$ is the filtered derivatives of the angular rates, and v is the pseudo-control input, which is determined by the robust adaptive controller in the next section. The INDI controller architecture is illustrated in Figure 3.



v
+
−
$g^{-1}\ \mathbf{x}(t-Ts)\ (-\dot{\mathbf{x}}_f + v)$
+
+
z^{-1}
$\mathbf{U}_c$
$\dot{\mathbf{x}} = \mathbf{f}\ \mathbf{x}\ + \mathbf{g}\ \mathbf{x}\ \mathbf{U}_c$
$\mathbf{X}$
z^{-1}
diff
$C(s)$
$\dot{\mathbf{x}}_f$
$\mathbf{x}(t-Ts)$

Figure 3 INDI controller architecture

3.2 Robust-adaptive controller design

To enhance the INDI robustness, especially in presence of motor fault effect, a robust adaptive control algorithm is augmented to the INDI algorithm to generate the pseudo-control input (v). Integration of the INDI algorithm as the baseline controller and the model reference robust-adaptive controller as the outer-loop controller can improve the performance of the total controller. In the following, the design procedure and application of the robust adaptive algorithm, as an augmentation algorithm to the INDI controller is described.

In our proposed robust MRAC strategy, the dynamics of the reference model is considered as follows:

$$\dot{\mathbf{x}}_m = \mathbf{A}_m\mathbf{x}_m + \mathbf{B}_m\mathbf{R} \tag{20}$$

where $\mathbf{A}_m \in R^{n\times n}$ is known desired Hurwitz closed-loop system dynamics and $\mathbf{B}_m \in R^{n\times m}$ is an identity matrix (**I**) in our case. Applying simple feedback, the poles of the closed-loop system are set to the eigenvalues of the matrix **A***m.* Therefore, the differential equations of the plant's dynamics can be propagated as follows:

$$\begin{gathered}\dot{\mathbf{x}} = \mathbf{A}_m \mathbf{x}_m + \mathbf{B}_m(\omega \mathbf{u}_{\mathbf{ad}} + \boldsymbol{\theta}\left\|\mathbf{x}\right\|_\infty + \boldsymbol{\sigma}) + \boldsymbol{\Delta}_1 \;\; \mathbf{x},\mathbf{u} \;\;,\;\; \mathbf{x}(0) = \mathbf{x}_0 \\ \mathbf{y}(t) = \mathbf{c}^T \mathbf{x}(t)\end{gathered} \tag{21}$$

where $\|.\|_\infty$ is the infinity norm; $x(\mathrm{t}) \in R^n$ is the measured system state; $c \in R^{n\times n}$ is a known constant vector; $\mathbf{u}_{\mathbf{ad}}(\mathrm{t}) \in R^m$ is the control input; ω**, θ, σ** are unknown constant parameters with known signs and lower and upper bounds; $\boldsymbol{\Delta}_1(\mathbf{x},\mathbf{u}) : R \times R^n \to R^n$ is a continuous bounded unknown nonlinear argument due to INDI error and the effect of rotor fault.

The above system architecture is replicated by the use of state predictor which is given by Eq. (22):

$$\begin{gathered}\dot{\hat{\mathbf{x}}}_m = \mathbf{A}_m \hat{\mathbf{x}}_m + \mathbf{B}_m(\hat{\omega} \mathbf{u}_{\mathbf{ad}} + \hat{\boldsymbol{\theta}}\left\|\mathbf{x}\right\|_\infty + \hat{\boldsymbol{\sigma}}),\; \hat{\mathbf{x}}(0) = \mathbf{x}_0 \\ \hat{\mathbf{y}}(\mathrm{t}) = \mathbf{c}^T \hat{\mathbf{x}}(\mathrm{t})\end{gathered} \tag{22}$$

where $\hat{\omega}(\mathrm{t}) \in R$ is the estimate of $\omega \in \left[\omega_L, \omega_u\right]$, $\hat{\boldsymbol{\theta}}(\mathrm{t}) \in R^n$, and $\hat{\boldsymbol{\sigma}} \in R^n$ are adaptive estimates of the dynamic model parameters $\boldsymbol{\theta}(\mathrm{t})$ and $\boldsymbol{\sigma}(\mathrm{t})$, which are continuously differentiable and bounded as; $\left\|\dot{\boldsymbol{\theta}}\right\| \le \delta_1$, $\left\|\dot{\omega}\right\| \le \delta_2$, $\left\|\dot{\boldsymbol{\sigma}}\right\| \le \delta_3$.

Adaptive Law: The adaption laws governing the adaptive estimates are as follows [6]:

$$\begin{aligned}\dot{\hat{\theta}}(\mathrm{t}) &= \Gamma\mathrm{Proj}\;\; \hat{\theta}(\mathrm{t}), -\tilde{x}^T(\mathrm{t})P\,b\left\|x(\mathrm{t})\right\|_\infty \quad &&,\;\; \hat{\theta}(0) = \hat{\theta}_0, \\ \dot{\hat{\sigma}}(\mathrm{t}) &= \Gamma\mathrm{Proj}\;\; \hat{\sigma}(\mathrm{t}), -\tilde{x}^T(\mathrm{t})P\,b \quad &&,\; \hat{\sigma}(0) = \hat{\sigma}_0\,, \\ \dot{\hat{\omega}}(\mathrm{t}) &= \Gamma\mathrm{Proj}\;\; \hat{\omega}(\mathrm{t}), -\tilde{x}^T(\mathrm{t})P\,b\,u(\mathrm{t}) \quad &&,\;\; \hat{\omega}(0) = \hat{\omega}_0\end{aligned} \tag{23}$$

In Eq. (23), the term Γ is the adaptation gain and $\hat{\theta}_0, \hat{\omega}_0$, $\hat{\sigma}_0$ are the initial values of pertinent variables, which are guessed for initialization of the algorithm. Large values of Γ, increase the rate of adaptation for desirable performance without reducing the robustness properties. $P = P^T > 0$ and $Q = Q^T > 0$ are used in the Lyapunov function $A_m^T P + PA_m = -Q$, and $\tilde{x}(\mathrm{t}) = \hat{x}(\mathrm{t}) - x(\mathrm{t})$ is the error function.

The projection operator, which is denoted by $\mathrm{Proj}(.,.)$ guarantees estimated parameters boundedness according to Ref [6]. The projection operator is defined as below:

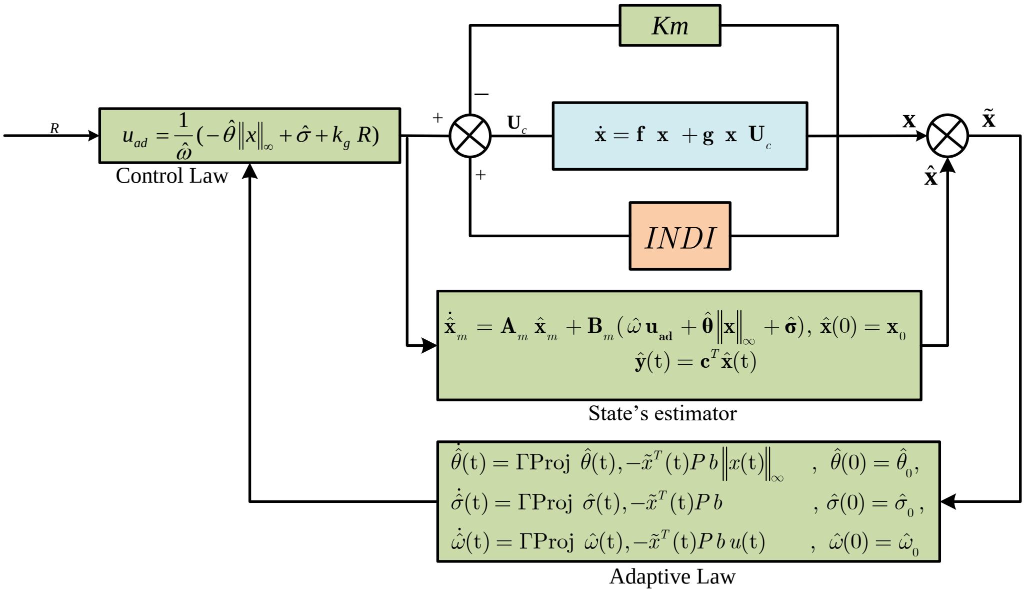
$$Proj(\varphi, z) = \begin{cases} z & \text{if } \; h(\varphi) < 0, \\ z & \text{if } \; h(\varphi) \ge 0 \quad and \quad \nabla h^T z \le 0, \\ z - \dfrac{\nabla h}{\left\|\nabla h\right\|}(\dfrac{\nabla h}{\left\|\nabla h\right\|}\,.\,z)h(\varphi) & \text{if } \; h(\varphi) \ge 0 \quad and \quad \nabla h^T z > 0. \end{cases} \tag{24}$$

where "." and ∇ represent the inner product and gradient, respectively and *h* is a convex function defined as $h(\varphi) = \dfrac{\left(\varepsilon_\varphi + 1\right)\varphi^T\varphi - \varphi_{\max}^2}{\varepsilon_\varphi \varphi_{\max}^2}$, and $\varepsilon_\varphi > 0$ is the projection tolerance bound, and $\varphi_{\max}$ is the norm bound forced on the vector φ, which is defined in a bounded convex as $\Omega_c = \left\{\varphi \in R^n \middle| h(\varphi) \le \mathrm{c}\right\},\; 0 \le c \le 1\cdot$

Control Algorithm: The robust model reference adaptive control algorithm signal is obtained as below:

$$u_{ad} = \frac{1}{\hat{\omega}}(-\hat{\theta}\|x\|_{\infty} + \hat{\sigma} + k_g\ R) \tag{25}$$

In the above control algorithm equation, k_g is selected to ensure a unity DC gain of the desired system corresponding to Eq. (20). The complete block diagram of the proposed controller including the INDI algorithm, the state estimator, adaptation law, and the adaptive control algorithm is illustrated in Figure 4.



Km
R
$u_{ad} = \frac{1}{\hat{\omega}}(-\hat{\theta}\|x\|_{\infty} + \hat{\sigma} + k_g\ R)$
Control Law
$+$
$-$
$+$
$\mathbf{U}_c$
$\dot{\mathbf{x}} = \mathbf{f}\ \ \mathbf{x}\ \ + \mathbf{g}\ \ \mathbf{x}\ \ \mathbf{U}_c$
$\mathbf{x}$
$\tilde{\mathbf{x}}$
$\hat{\mathbf{x}}$
$INDI$
$\dot{\hat{\mathbf{x}}}_m = \mathbf{A}_m\ \hat{\mathbf{x}}_m + \mathbf{B}_m(\hat{\omega}\,\mathbf{u}_{\mathbf{ad}} + \hat{\boldsymbol{\theta}}\|\mathbf{x}\|_{\infty} + \hat{\boldsymbol{\sigma}}),\ \hat{\mathbf{x}}(0) = \mathbf{x}_0$
$\hat{\mathbf{y}}(\mathrm{t}) = \mathbf{c}^T\hat{\mathbf{x}}(\mathrm{t})$
State's estimator
$\dot{\hat{\theta}}(\mathrm{t}) = \Gamma\mathrm{Proj}\ \ \hat{\theta}(\mathrm{t}), -\tilde{x}^T(\mathrm{t})P\,b\,\|x(\mathrm{t})\|_{\infty}\ \ ,\ \hat{\theta}(0) = \hat{\theta}_0,$
$\dot{\hat{\sigma}}(\mathrm{t}) = \Gamma\mathrm{Proj}\ \ \hat{\sigma}(\mathrm{t}), -\tilde{x}^T(\mathrm{t})P\,b\ \ ,\ \hat{\sigma}(0) = \hat{\sigma}_0,$
$\dot{\hat{\omega}}(\mathrm{t}) = \Gamma\mathrm{Proj}\ \ \hat{\omega}(\mathrm{t}), -\tilde{x}^T(\mathrm{t})P\,b\,u(\mathrm{t})\ \ ,\ \hat{\omega}(0) = \hat{\omega}_0$
Adaptive Law

Figure 4 Model Reference Robust adaptive controller with INDI Algorithm

c) Outer loop controller design

In the outer-loop position control, a PID control algorithm is utilized. By considering the desired trajectory along with its first and second derivatives, the dynamics of the position error can be derived as follows:

$$\begin{aligned} &\ddot{\mathbf{P}}_e + K_d\dot{\mathbf{P}}_{\mathbf{e}} + K_p\mathbf{P}_e + K_I\int \mathbf{P}_e\,dt = 0 \\ &\mathbf{P}_d = [x_d, y_d, z_d]\ ,\ \ \mathbf{P}_e = \mathbf{P}_d - \mathbf{P} \end{aligned} \tag{26}$$

where $\mathbf{P}_d$ is the desired position with bounded first and second derivatives, $\mathbf{P}_e$ is the positions error, and the PID gains (K_p, K_d, K_i) are derived corresponding to the conditions of Routh-Hurwitz to exponentially converge the error to zero. According to the error dynamics, the following equation can be computed:

$$\ddot{\mathbf{P}} = \ddot{\mathbf{P}}_d + K_d\dot{\mathbf{P}}_e + K_p\mathbf{P}_e + K_I\int \mathbf{P}_e\,dt = 0 \tag{27}$$

Based on the desired positions and translational dynamics of Eq. (1), the desired Euler angles are derived as the command pitch and roll angles as follows:

$$\begin{cases} \theta_c = \arcsin\left(\dfrac{m\ddot{x}}{\cos\psi_d\ T}\right) \\ \varphi_c = -\arcsin\left(\dfrac{m\ddot{y}}{\cos\psi_d\ T}\right) \end{cases} \tag{28}$$

Where in the above equation $T = m\sqrt{\ddot{x}^2 + \ddot{y}^2 + (\ddot{z} + g)^2}$ and the desired heading angle (ψ_d) is imposed by the trajectory generation unit corresponding to the desired trajectory. The conventional PID control algorithm has two disadvantages; 1) sudden jump of the output of the derivative part of PID, which can saturate the actuator if the desired input is like a step function and 2) the problem of integral wind up when the integral value is high and the error switches its sign. To remove these problems, as shown in Figure 5, the system output is used in the derivative part without accounting for the desired input, and an anti-windup filter is applied in an integral part of the PID algorithm [31].

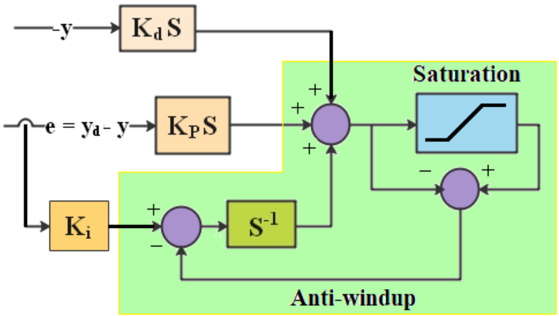


Figure 5 Anti wind up PID controller architecture

4. Simulation results

Several numerical simulations are considered in the presence of partial loss of motor effectiveness to verify the performance of the proposed three-loop robust adaptive fault-tolerant controller. The results of the implementation of the proposed algorithm are compared with a robust adaptive controller augmented with an NDI algorithm. In the first simulation scenario, the performance of the introduced controller is investigated for the case of the healthy (no-fault) quadrotor. Accordingly, Figures 6-10 represent the parameters of the quadrotor, when tracking a helical trajectory. Figures 6, 7 represent the quadrotor's attitude rates and Euler angles, respectively. The rotational speeds of the rotors along with the corresponding control moments are depicted in Figures 8, 9, respectively. Accordingly, the required rotation speed of the rotors to track the desired path is around 4400 RPM. Finally, the quadrotor position in 3D space and the trajectory tracking performance of the proposed controller has been illustrated in Figure 10. As shown, the controller has a satisfactory tracking performance. In addition to the proposed controller, the results are compared with an NDI controller augmented with the robust adaptive algorithm. Comparison of the results based on the proposed algorithm and the NDI controller shows the performance of both controllers are close in case of no-fault for the healthy quadrotor. Several fault scenarios are considered to investigate the performance and robustness of the controller algorithm. For this purpose, the performance of the controller is examined and compared with the NDI algorithm for different percentages of fault on the motor number1 as illustrated in Figures 11-25.

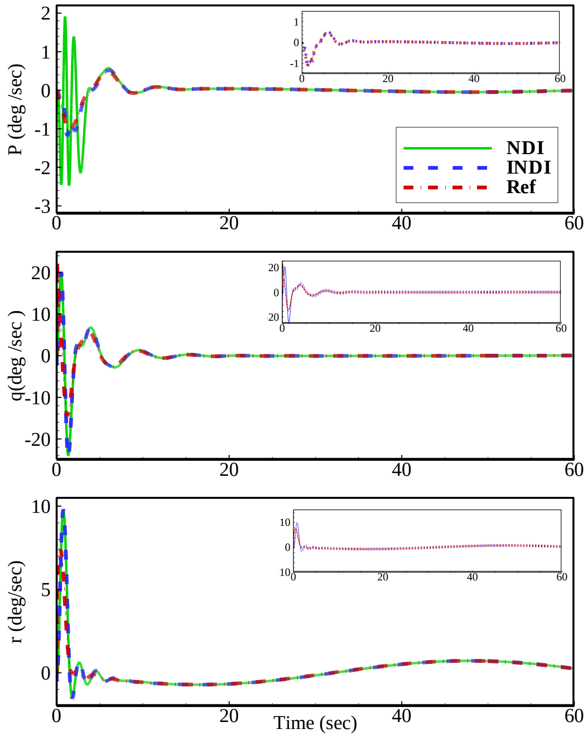


Figure 6 Quadrotor body angular rates

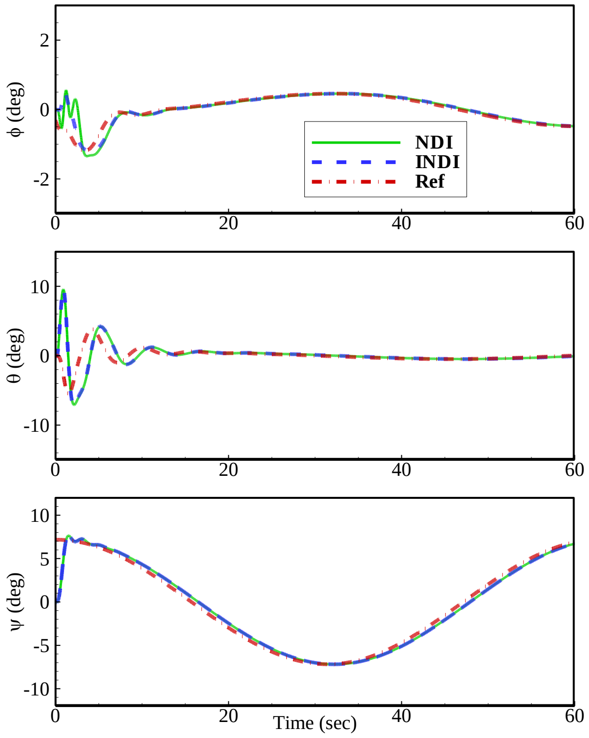


Figure 7 Quadrotor Euler Angles (deg)

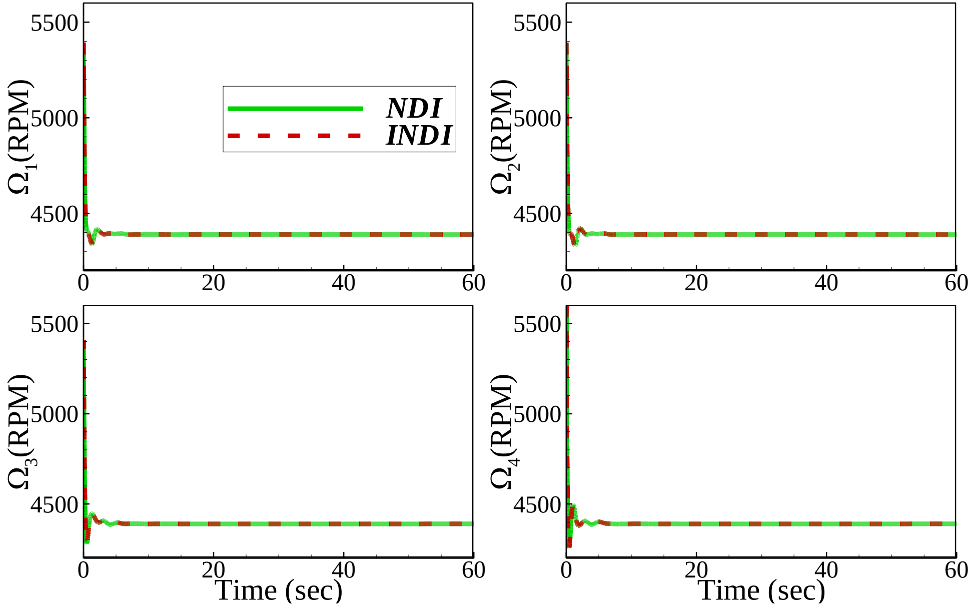
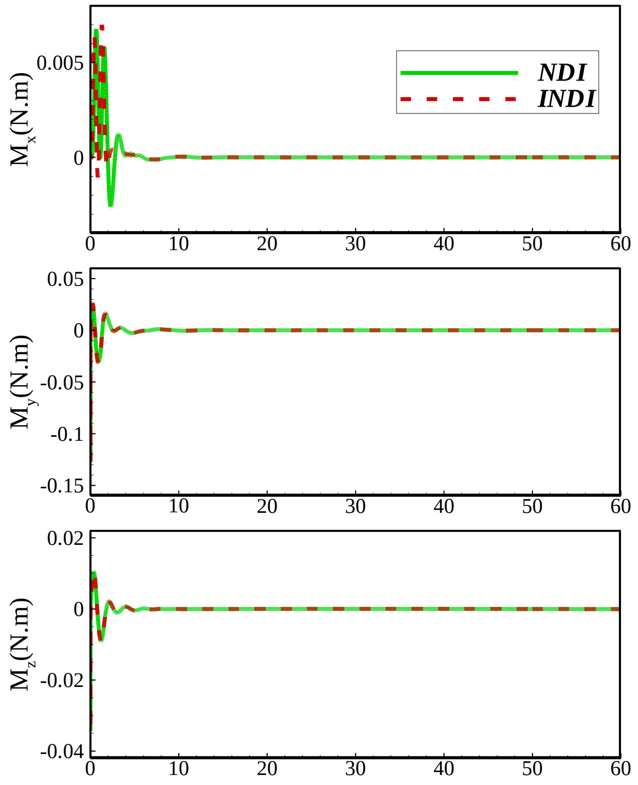


Figure 8 Quadrotor rotational speeds of motors (RPM)



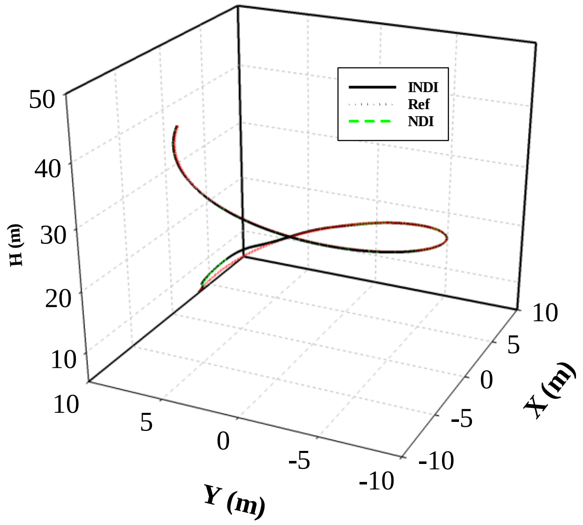
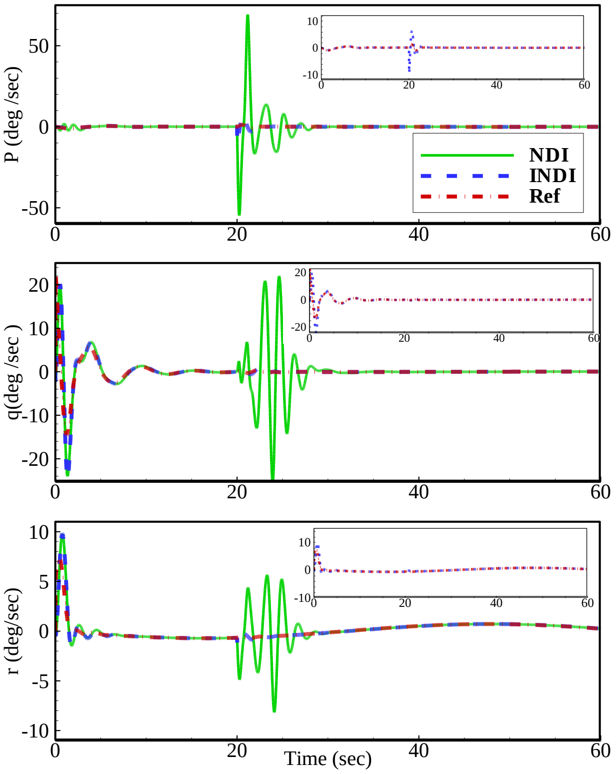


Figure 10 Quadrotor position in 3D space

Figure 9 Quadrotor Controller moments



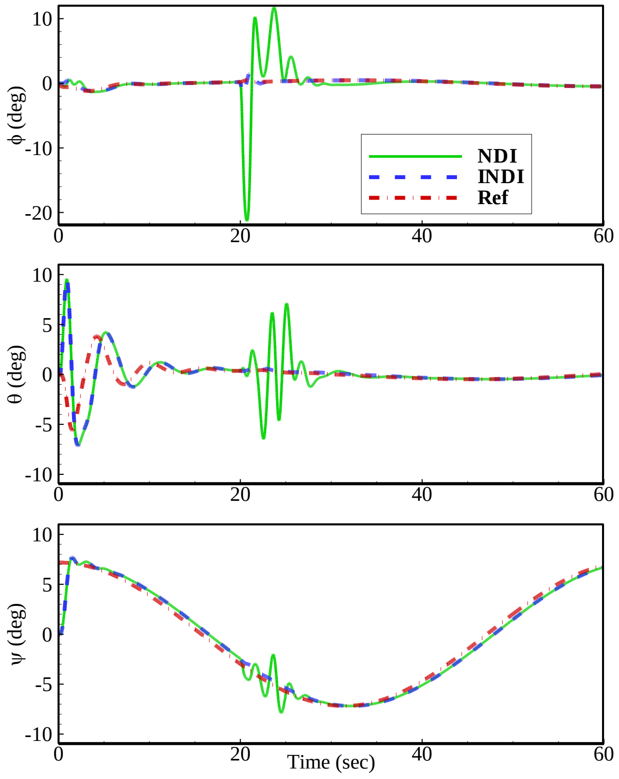


Figure 11 Angular rates with 20% fault on motor1

Figure 12 Euler Angles with 20% fault on motor1

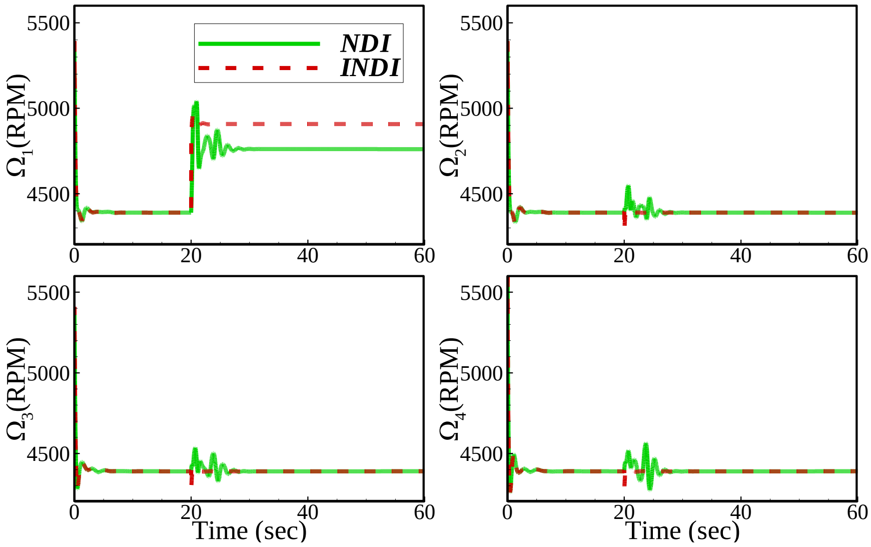


Figure 13 Motors speeds with 20% fault on motor1

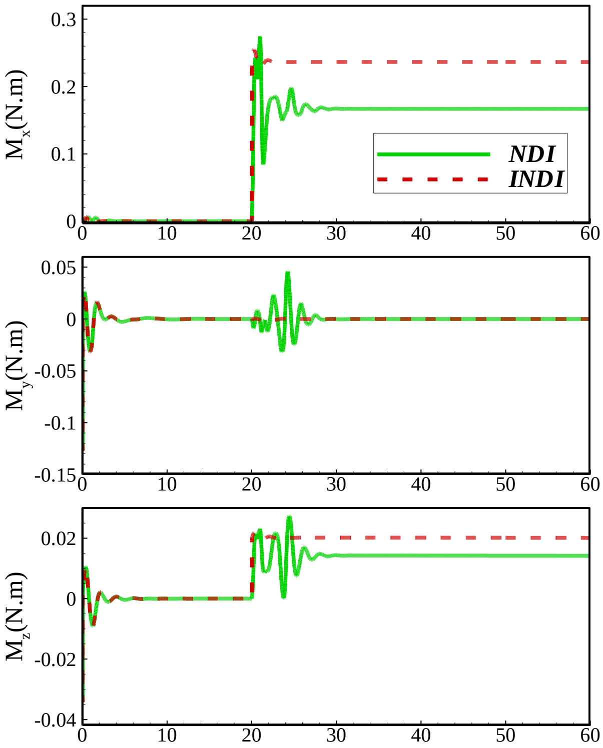


Figure 14 Controller moments with 20% fault on motor1

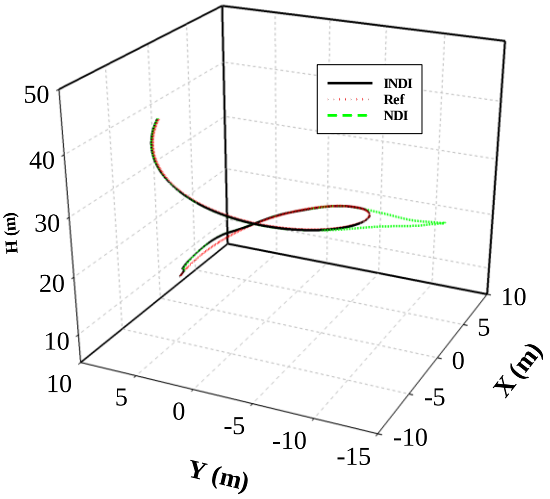


Figure 15 Trajectory tracking with 20% fault on motor1

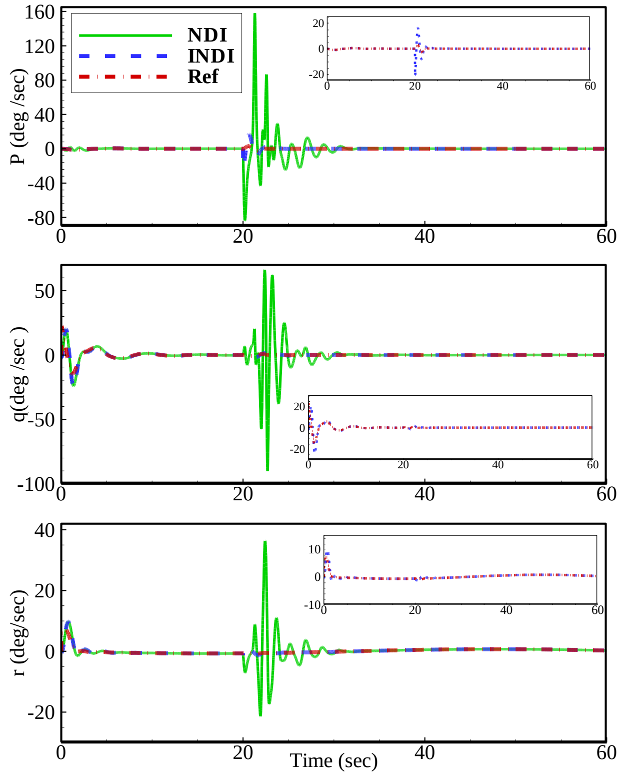


Figure 16 angular rates with 40% fault on motor1

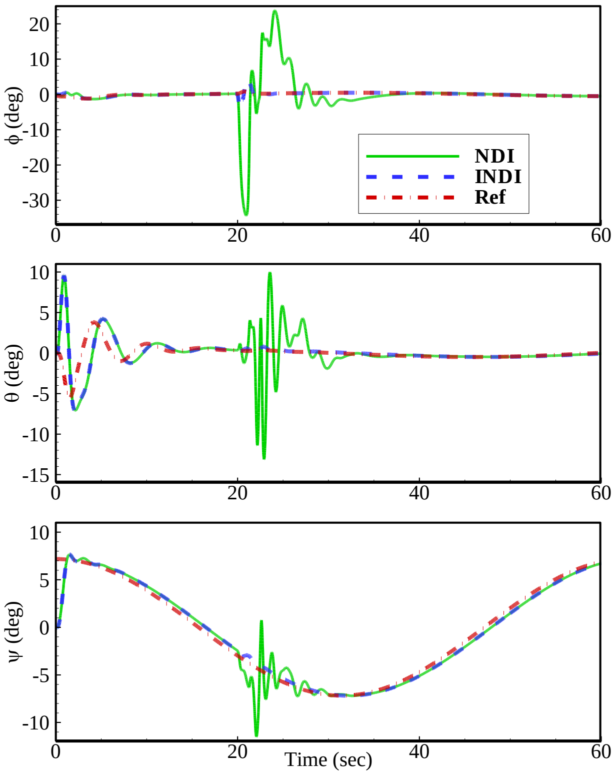


Figure 17 Euler Angles with 40% fault on motor1

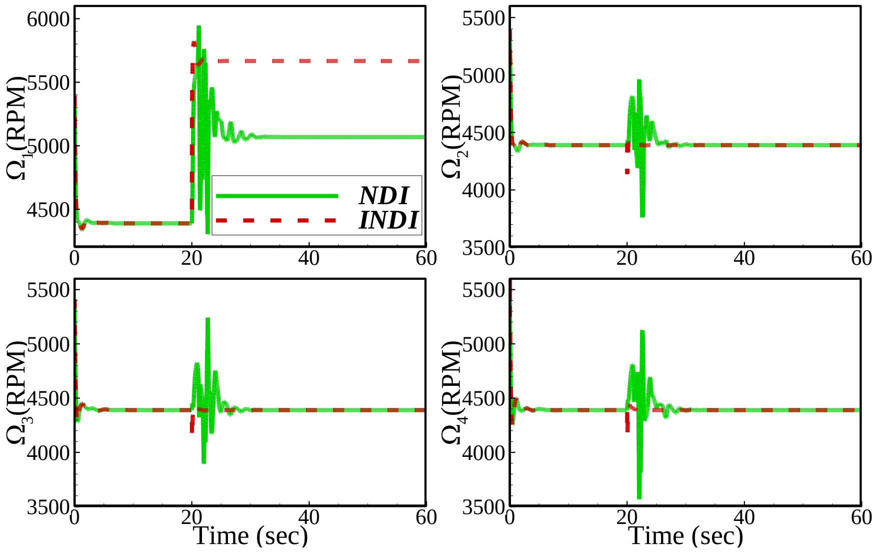


Figure 18 Motors speeds with 40% fault on motor1

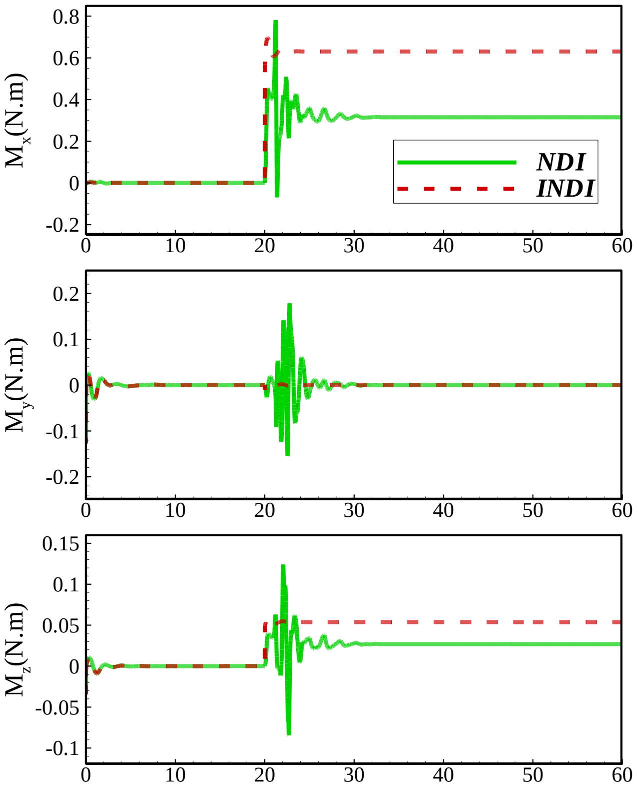


Figure 19 Controller moments with 40% fault on motor1

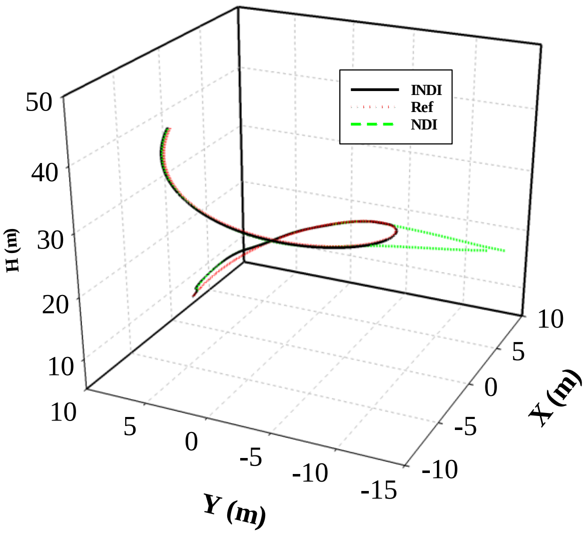
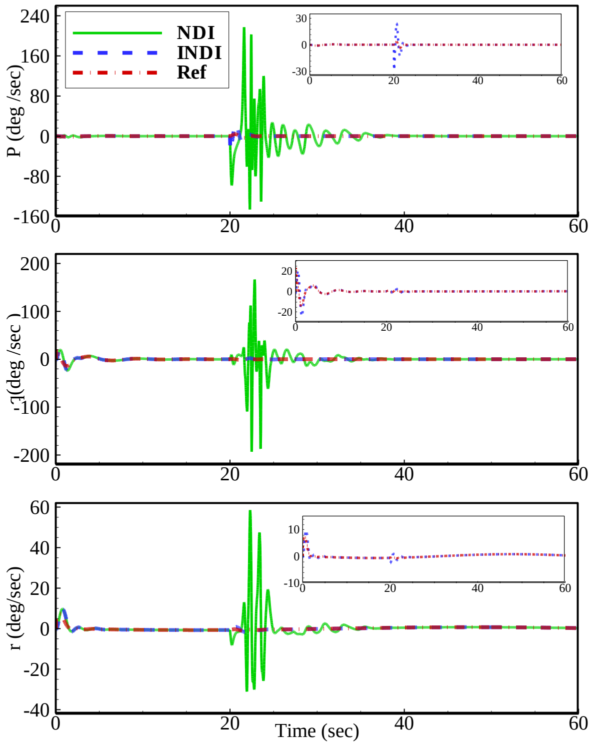


Figure 20 Trajectory tracking with 40% fault on motor1



00

Figure 21 angular rates with 50% fault on motor1

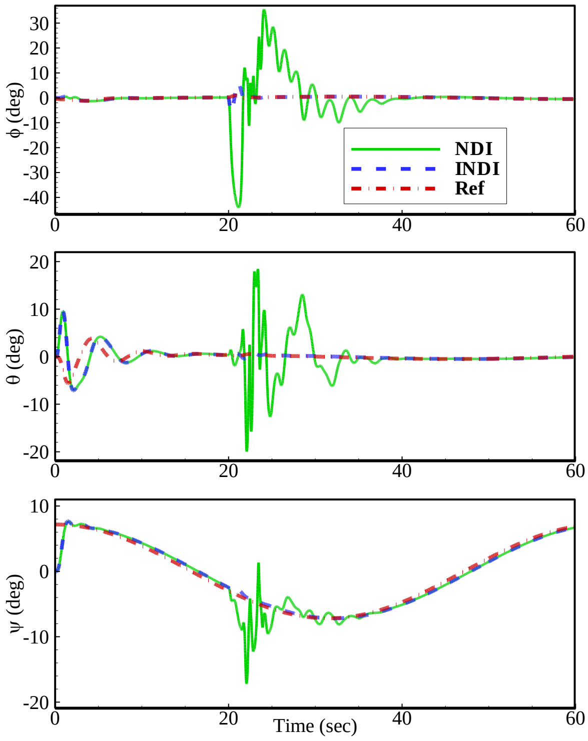


Figure 22 Euler Angles with 50% fault on motor1

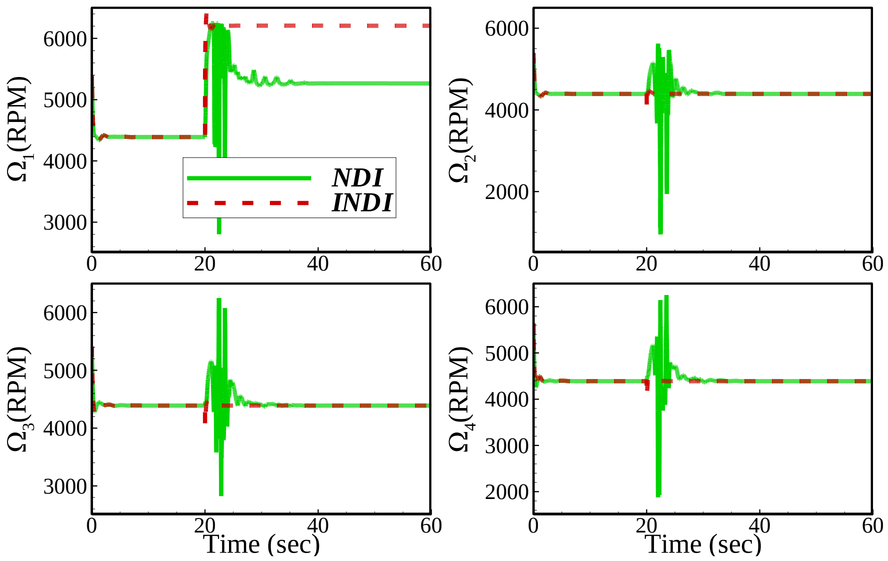


Figure 23 Motors speeds with 50% fault on motor1

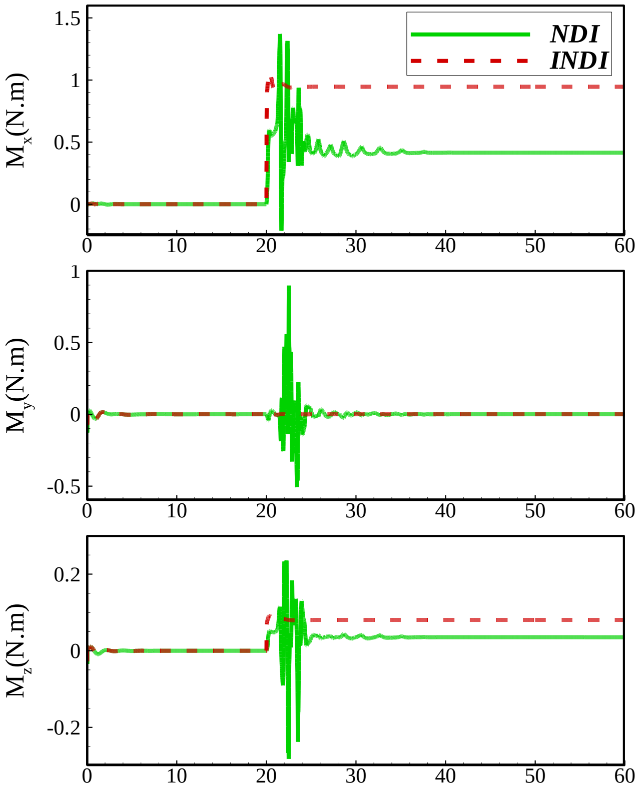


Figure 24 Controller moments with 50% fault on motor1

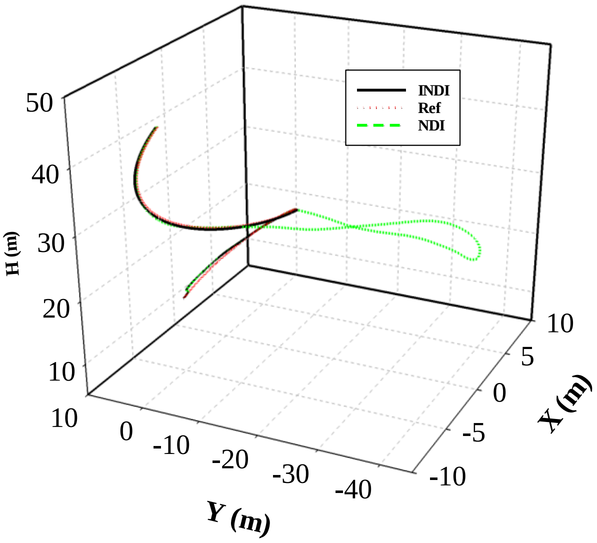


Figure 25 Trajectory tracking with 50% fault on motor1

Figures 11-15 present the quadrotor's parameters under a 20% rotor fault scenario. These figures include the body rotational speed, Euler angles, motors' rotational speeds, moments generated by the motors, and the quadrotor's position. Each figure displays the curves of the desired values, the NDI outputs, and the proposed controller outputs. As depicted in Figure 11 and Figure 12, the proposed controller successfully restores the rotational rates and Euler angles of the quadrotor at a much faster rate compared to the NDI algorithm. Additionally, Figure 15 confirms that the trajectory tracking performance of the proposed strategy surpasses that of the NDI algorithm.

Similar simulations are conducted for a 40% fault on the number one rotor, as shown in Figure 16 to Figure 20. Despite the excellent trajectory tracking performance of the proposed controller (Figure 20), there is a significant deviation in the trajectory of the NDI controller. As previously discussed, the controller maintains full controllability (roll, pitch, and yaw) of the quadrotor up to a maximum of 50% rotor fault. Figures 21-25 illustrate the performance of the controller in the presence of a 50% motor fault. In Figure 25, the proposed controller tracks the desired inputs and trajectory perfectly, while the NDI algorithm exhibits significant variations and oscillations due to the effect of the fault. Notably, the roll angle increases to approximately 50 degrees when applying the NDI algorithm, leading to a significant deviation from the desired trajectory (Figure 22). Figure 23 demonstrates that the rotational speed of rotor number one approaches saturation magnitude under a 50% partial fault. The maximum rotation speed of the rotors is 6250 RPM, which explains why the rotational speed of the faulty motor saturates and the controller's performance deteriorates drastically for faults greater than 50%.

5. Conclusion

This paper introduces a novel approach by augmenting the Incremental Nonlinear Dynamic Inversion (INDI) algorithm with a model reference robust adaptive algorithm to effectively control and recover a quadrotor in the presence of partial actuator faults. The robust adaptive algorithm is designed to handle the impact of unmodeled faults arising from the rotor system. Through various simulation scenarios, the performance of the proposed control strategy is thoroughly evaluated. Furthermore, a comprehensive comparison is conducted between the introduced controller and the NDI algorithm augmented with the robust adaptive algorithm to validate the effectiveness of the proposed approach.

The simulation results demonstrate that the proposed control strategy ensures full controllability of the quadrotor in the roll, pitch, and yaw channels even when partial actuator faults occur, with a fault tolerance capability of up to 50%. By achieving full controllability across all channels, the quadrotor is capable of accurately tracking desired trajectories despite the presence of partial actuator faults. A comparative analysis between the introduced algorithm and the NDI controller reveals that the INDI controller exhibits significantly superior performance, particularly when the dynamics are affected by actuator faults. While the NDI controller can still control the faulty quadrotor, it leads to notable deviations from the desired path at the moment of failure occurrence.

References

[1] Turan V., Avşar E., Asadi D., Aydin E.A., "Image Processing Based Autonomous Landing Zone Detection for a Multi-Rotor Drone in Emergency Situation”, Turkish

Journal Engineering 5 (2021): 193-200.

[2] Asadi D., “Partial engine fault detection and control of a quadrotor considering model uncertainty”, Turkish Journal Engineering 6 (2021):106-117.

[3] Nabavi Y., Asadi D., Ahmadi K., “Image-based UAV position and velocity estimationsing a monocular camera”, Control Engineering Practice 134 (2023).

[4] Asadi D., Sabzehparvar M., Atkins E.M., Talebi H.A., “Damaged airplane trajectory planning based on flight envelope and motion primitives”, Journal of Aircraft 51 (2014): 1740-1757.

[5] Asadi D., Sabzehparvar M., Talebi H.A., “Damaged airplane flight envelope and stability evaluation, Aircraft Engineering and Aerospace Technology”, 85 (2013) : 186-198.

[6] Ahmadi K., Asadi D., Pazooki F., “Nonlinear L1 adaptive control of an airplane with structural damage”, Proc. Inst. Mech. Eng. Part G J. Aerosp. Eng. 233 (2019) : 341-353.

[7] Asadi D., Ahmadi K., “Nonlinear robust adaptive control of an airplane with structural damage”, Proc. Inst. Mech. Eng. Part G J. Aerosp. Eng. 234 (2020) : 2076-2088.

[8] Asadi D., Bagherzadeh S.A., “Nonlinear adaptive sliding mode tracking control of an airplane with wing damage”, Proc. Inst. Mech. Eng. Part G J. Aerosp. Eng. 232 (2017): 1405–1420.

[9] Alwi H., Edwards C., “Fault-tolerant control of an octorotor using LPV based sliding mode control allocation”, American Control Conference, Washington, DC, USA, (2013) : 6505-6510.

[10] Navabi M., Davoodi A., Mirzaei H., “Trajectory tracking of an under-actuated quadcopter using Lyapunov-based optimum adaptive controller”, Proc. Inst. Mech. Eng. Part G J. Aerosp. Eng. 236(1) (2021) : 202-215.

[11] Ahmadi K., Asadi D., Nabavi Y., Tutsoy O., “Modified adaptive discrete-time incremental nonlinear dynamic inversion control for quad-rotors in the presence of motor faults”, Mechanical Systems and Signal Processing 188 (2023) : 109989.

[12] Asadi D., Ahmadi K., Nabavi Y., “Fault-tolerant Trajectory Tracking Control of a Quadcopter in Presence of a Motor Fault”, Int. J. Aeronaut. Sp. Sci. 23 (2022) : 129-142.

[13] Gao Z., Cecati C., Ding S.X., “A Survey of Fault Diagnosis and Fault-Tolerant Techniques—Part I: Fault Diagnosis With Model-Based and Signal-Based Approaches”, IEEE Trans. Ind. Electron. 62 (2015) : 3757-3767.

[14] Ahmadi K., Asadi D., Merheb A., Nabavi Y., Tutsoy O., “Active fault-tolerant control of quadrotor UAVs with nonlinear observer-based sliding mode control validated through hardware in the loop experiments”, Control Engineering Practice, 137 (2023).

[15] Barghandan S., Badamchizadeh M.A., Jahed-Motlagh M.R., “Improved adaptive fuzzy sliding mode controller for robust fault-tolerant of a Quadrotor”, Int. J. Control. Autom. Syst. 15 (2017): 427-441.

[16] Lanzon A., Freddi A., Longhi S., “Flight Control of a Quadrotor Vehicle Subsequent to a Rotor Failure”, J. Guid. Control. Dyn. 37 (2014): 580-591.

[17] Khaneghaei M., Asadi D., Tutsoy O., “Software in the Loop (SIL) Simulation for an Autonomous Multirotor Flight Planning and Landing with ROS and Gazebo”, 2023 7th International Symposium on Innovative Approaches in Smart Technologies (ISAS), Istanbul, Turkiye (2023): 1-10.

[18] Besnard L., Shtessel Y.B., Landrum B., “Quadrotor vehicle control via sliding mode

controller driven by sliding mode disturbance observer", J. Franklin Inst. 349 (2012) : 658-684.

[19] Quan Q., "Introduction to multicopter design and control", Springer, (2017).

[20] Haghighi H., Delahaye D., Asadi D., "Performance-based emergency landing trajectory planning applying meta-heuristic and Dubins paths, Applied Soft Computing", 117 (2022) : 108453.

[21] Sun S., Wang X., Chu Q., Visser C., "Incremental Nonlinear Fault-Tolerant Control of a Quadrotor With Complete Loss of Two Opposing Rotors", IEEE Trans. Robot. 37 (2021) : 116-130.

[22] Sun S., Cioffi G., De Visser C., Scaramuzza D., "Autonomous Quadrotor Flight Despite Rotor Failure With Onboard Vision Sensors: Frames vs. Events", IEEE Robot. Autom. Lett. 6(2) (2021) : 580-587.

[23] Asadi D., "Actuator Fault Detection, Identification, and Control of a Multirotor Air Vehicle Using Residual Generation and Parameter Estimation Approaches. Int. J. Aeronaut. Space Sci. 25, 176-189 (2024).

[24] Asadi D., Ahmadi K., Nabavi Y., Tutsoy O., "Controllability of multi-rotors under motor fault effect", Artıbilim: Adana Alparslan Türkeş Bilim ve Teknoloji Üniversitesi Fen Bilimleri Dergisi 4 (2021): 24-43.

[25] Lee J., Choi H.S., Shim H., "Fault Tolerant Control of Hexacopter for Actuator Faults using Time Delay Control Method", Int. J. Aeronaut. Sp. Sci. 17 (2016): 54-63.

[26] Asadi, D., "Model-based Fault Detection and Identification of a Quadrotor with Rotor Fault", Int. J. Aeronaut. Space Sci. 23 (2022): 916-928. 10.1007/s42405-022-00494-z.

[27] Merheb A., Noura H., Bateman F., "Emergency Control of AR Drone Quadrotor UAV Suffering a Total Loss of One Rotor", IEEE/ASME Trans. Mechatronics. 22 (2017) : 961-971.

[28] Lee D., "A Linear Acceleration Control for Precise Trajectory Tracking Flights of a Quadrotor UAV Under High-wind Environments", Int. J. Aeronaut. Sp. Sci. 22 (2021) : 898-910.

[29] Bouabdallah S., Siegwart R., "Full control of a quadrotor", IEEE/RSJ International Conference on Intelligent Robots and Systems, San Diego, CA, USA, (2007) :153-158.

[30] Tutsoy O., Asadi D., Ahmadi K., Nabavi Y., "Robust Reduced-Order Thau Observer With the Adaptive Fault Estimator for the Unmanned Air Vehicles", in IEEE Transactions on Vehicular Technology, 72 (2023) : 1601-1610.

[31] Ermeydan A., Kiyak E., "Fault tolerant control against actuator faults based on enhanced PID controller for a quadrotor", Aircr. Eng. Aerosp. Technol. 89 (2017) : 468-476.