International Journal of Aeronautics and Astronautics
https://dergipark.org.tr/en/pub/ijaa

2022, VOL:3, ISS:1, 28-47 e-ISSN: 2757-6574

Research Article

Fault tolerant control of a quadrotor based on incremental nonlinear dynamic inversion

Karim Ahmadi*, Davood Asadi, Seyed-Yaser Nabavi-Chashmi

Department of Aerospace Engineering, Adana Alparslan Turkes Science and Technology University, Adana, Türkiye

Received	Revised	Accepted
December 6, 2021	February 7, 2022	**February 18, 2022**

Keywords
Fault Tolerant Control
Multirotor
Nonlinear Dynamic Inversion
Incremental

ABSTRACT

The multirotor unmanned aerial vehicles (UAVs) have rapidly attracted interest of the researchers since they play a unique role in a variety of areas including the military, agriculture, rescue, and mining. Actuator fault or failure is inevitable during multi-rotor's operations, which can endanger humans on the ground in addition to costly damage to the system itself. Therefore, this paper introduces a nonlinear controller algorithm for fault-tolerant control of a quadcopter with partial loss of actuator effectiveness. The introduced controller includes a cascade structure of the fast inner-loop dynamics and slow outer-loop dynamics. In the inner-loop part of the controller, an incremental nonlinear dynamic inversion controller is applied and a modified PID control algorithm is used in the outer-loop of the controller. Simulation results for different fault scenarios demonstrate that the proposed fault-tolerant controller approach can quickly adapt itself to the abrupt change due to the motor faults and tracks the desired inputs satisfactorily.

* Corresponding author, e-mail: kadastgerdi@atu.edu.tr

Authorship contribution statement for Contributor Roles Taxonomy
Karim Ahmadi, *has developed the theoretical analysis and the numerical simulations regarding the control algorithm and experiments as well as manuscript preparation.* **Davood Asadi,** *has supervised the theoretical development of control approaches, and experiments as well as helped in preparing the manuscript.* **Yaser Nabavi,** *has helped in manuscript writing.*

Conflicts of Interest: The authors declare no conflict of interest.

Citation: *Ahmadi, K., Asadi, D., Nabavi Chashmi, S. 2022. Fault tolerant control of a quadrotor based on incremental nonlinear dynamic inversion. International Journal of Aeronautics and Astronautics, 3(1), 28-47.*

1. Introduction

Multi-rotor Unmanned Aerial Vehicles (UAVs) have rapidly attracted the interest of researchers since they are being implemented in a variety of different applications including surveillance, reconnaissance, agriculture, rescue, and mining. 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]. The controllability of the multirotor in presence of motor fault is investigated in [6]. Fault-tolerant control (FTC) techniques have been proposed in [7–12] to recover the control of faulty vehicles. Nonlinear L1 adaptive control [7], robust adaptive control [8], adaptive sliding mode control [9], Linear Parametric Variable (LPV) sliding mode control [10], optimal adaptive control [11], and Model Reference Adaptive Control (MRAC) [12] are some instances of direct fault-tolerant control algorithms.

In addition to the direct methods, fault-detection and identification algorithms are also used in fault-tolerant control strategies [13]. 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 [14]. Since multi-rotors have a nonlinear, highly–coupled, and underactuated dynamic system, controlling them is a challenging problem. On the other hand, actuator fault is also a common problem in multi-rotors, which raises discussions about their reliability and safety.

In the event of a motor fault and/or failure, numerous studies are being conducted to recover multi-rotor vehicles. FTC researches corresponding to the multi-rotor's motor faults and failures can be classified into two groups of partial actuator fault and complete loss of actuator effectiveness or actuator failure. Some researches investigate the effect of partial fault on the rotor and propose fault-tolerant strategies while other researches have examined the effect of motor failure and appropriate fault-tolerant control strategies. Among the researches, some have applied fault detection algorithms as a part of the FTC strategy, while others apply direct fault-tolerant control algorithms to control the multirotor. Ref [15] 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 [2, 16, 17], which have tried to control the quadrotor in presence of partial fault [2]. The sliding mode control technique has been applied in Ref. [16] 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. [17] 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 [18,19].

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

https://doi.org/10.55212/ijaa.1033224

angles [20]. Various control methodologies have been addressed in literature the problem of complete loss of one or two rotors of the quadrotor [20–22].

A robust feedback linearization controller along with an H∞ loop shaping technique is adopted in Ref. [20] to achieve regulation of roll and pitch angles around the chosen working point. A nonlinear sensor-based fault-tolerant controller is developed in Refs. [22,23] to stabilize a quadrotor with failure of two opposing rotors in the high-speed flight condition.

Ref [24] 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 [25]. 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 [26]. 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. [27] 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 [18]. According to the literature and above discussion, full controllability of quadrotor (roll, pitch, and yaw) is not possible for complete loss of effectiveness of one motor (motor failure).

In a novel approach proposed here, the Incremental Nonlinear Dynamic Inversion (INDI) control algorithm is applied and augmented with a nonlinear robust adaptive controller to control the quadrotor in presence of a motor fault. The simulation results verify the perfect performance of the introduced architecture. Additionally, for trajectory tracking, a modified PID algorithm is applied in the third loop of the three-loop control strategy.

The remainder of this paper is organized as follows. The quadrotor's nonlinear dynamic equation of motion is derived in section (II). The controller architecture including the INDI algorithm, robust adaptive controller approach, and the PID controller is presented in section (III). Numerical results, controller performance, and the comparison are examined in section (IV), and finally, the conclusion section briefly discusses the key results.

2. Mathematical Model

In this section, the quadrotor model and equations of motion, disturbance due to unknown dynamics, the motor model, as well as the motor mixer equations are presented.

a) 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.

https://doi.org/10.55212/ijaa.1033224

Table 1. Quadrotor frame specifications

Quadrotor Parameters	Values
Mass, m	1.59 kg
Thrust Parameter, b	2.02×10^{-7} N/rpm^2
Drag Parameter, d	4.18×10^{-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	6.8×10^{-5} $kg.m^2$
Max motor speed, Ω_{max}	6250 rpm

b) Dynamic equations

The translational and rotational equations of the quadrotor in the body frame are presented in Eqs. (1) and (2), respectively. As depicted in Fig. 1, the quadrotor consists of four motors. Number one and two motors rotate counterclockwise 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.

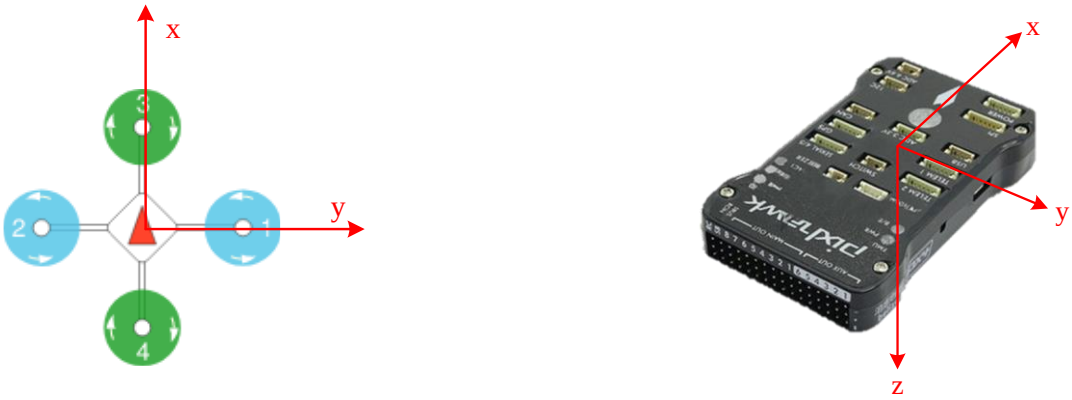

Fig. 1. Schematic representation of quadrotor

- **Translational dynamics**

$$\ddot{x} = -\frac{T}{m}(\sin\psi\sin\phi + \cos\psi\sin\theta\cos\phi) fx^{drag} + f_x^{\omega}$$
$$\ddot{y} = -\frac{T}{m}(-\cos\psi\sin\phi + \sin\psi\sin\theta\cos\phi) + f_y^{drag} + f_y^{\omega} \quad (1)$$
$$\ddot{z} = g - (cos\,\theta\,cos\,\emptyset)\frac{T}{m} + f_z^{drag} + f_z^{\omega}$$

https://doi.org/10.55212/ijaa.1033224

- **Rotational dynamics**

$$\begin{aligned}\dot{p} &= \frac{I_{yy}-I_{zz}}{I}qr + \frac{M_x}{I_{xx}} + \mathcal{T}_p^{drag} + \mathcal{T}_p^{gyro} + \mathcal{T}_p^{\text{wind}} \\ \dot{q} &= \frac{I_{zz}-I_{xx}}{I_{yy}}pr + \frac{M_y}{I_{yy}} + \mathcal{T}_q^{drag} + \mathcal{T}_q^{gyro} + \mathcal{T}_q^{wind} \\ \dot{r} &= \frac{I_{xx}-I_{yy}}{I_{zz}}pq + \frac{M_z}{I_{zz}} + \mathcal{T}_r^{drag} + \mathcal{T}_r^{gyro} + \mathcal{T}_r^{wind}\end{aligned} \quad \textbf{(2)}$$

- **Euler equations**

$$\begin{aligned}\dot{\varphi} &= p + q\,sin\,\varphi\,tan\,\theta + r\,cos\,\varphi\,tan\,\theta \\ \dot{\theta} &= q\,cos\,\varphi - r\,sin\,\varphi \\ \dot{\psi} &= \frac{1}{cos\,\theta}[q\,sin\,\varphi + r\,cos\,\varphi]\end{aligned} \quad \textbf{(3)}$$

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_{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 $\mathcal{T}$,M_x,M_y,cM_z, which are the total thrust force ($\mathcal{T}$) generated by propellers in *z*-direction and moments about *x*, *y*, *z* axes, respectively. The terms f_x^{drag}, f_y^{drag},f_z^{drag}, $\mathcal{T}_p^{drag}$,$\mathcal{T}_q^{drag}$, and $\mathcal{T}_r^{drag}$ are the drag forces and moments produced by the quadrotor's frame, which are expressed as $f_x^{drag} = -\frac{k_x}{m}\dot{x}$, $f_y^{drag} = -\frac{k_y}{m}\dot{y}$, $f_z^{drag} = -\frac{k_z}{m}\dot{z}$, $\mathcal{T}_p^{drag} = -\frac{k_\varphi}{I_{xx}}p^2$, $\mathcal{T}_q^{drag} = -\frac{k_\theta}{I_{yy}}q^2$,$\mathcal{T}_r^{drag} = -\frac{k_\psi}{I_{zz}}r^2$. 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 $\mathcal{T}_p^{gyro}$, $\mathcal{T}_q^{gyro}$, which are expressed as $\mathcal{T}_p^{gyro} = \frac{J_T}{I_{xx}}q\Omega$ and $\mathcal{T}_q^{gyro}, = -\frac{J_T}{I_{yy}}p\Omega$, in which J_T is the moment of inertia of each motor and Ω represents the propellers total speed as below:

$$\Omega = \Omega_1 - \Omega_2 + \Omega_3 - \Omega_4 \quad \textbf{(4)}$$

The terms f_x^w, f_y^w, f_z^w and $\mathcal{T}_p^{\text{wind}}$, $\mathcal{T}_q^{wind}$, $\mathcal{T}_r^{wind}$ 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. For the purpose of simulation, this paper considers the Dryden turbulence model, which is a stochastic model of the wind and is inherently dependent on the quadrotor's states (attitude, altitude, and velocity). Accordingly, in Dryden model the scale length and the probability of exceedance of high-altitude intensity are considered as 533.4*m* and 0.01, respectively and the low-altitude intensity is defined as 15 m/s.

c) Rotor dynamics

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

$$u_{i_c} = K\frac{\omega_0}{S+\omega_0}u_i \quad \textbf{(5)}$$

https://doi.org/10.55212/ijaa.1033224

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:

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

Where 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:

$$\boldsymbol{U} = \boldsymbol{K}_{\Omega 2U}\boldsymbol{\Omega} \quad \textbf{(7)}$$

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

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.

d) Motor mixer

The motor mixer determines the rotational speeds of each rotor corresponding to the intermediate autopilot outputs (**U**). accordingly, motor mixer expression can be reached by inversing Eq.(7) as below:

$$\boldsymbol{\Omega} = \boldsymbol{K}_{U2\Omega}\boldsymbol{U}, \boldsymbol{K}_{U2\Omega} = inv(\boldsymbol{K}_{\Omega 2U}) \quad \textbf{(8)}$$

e) Motor thrust and speed limitations

When converting the controller outputs to the motor inputs, the maximum constraint of the motors is applied to the motor mixer formulation. Based on the motor type, propeller size, battery specifications, the maximum speed of each motor (Ω_{max}) is found to be 6250 rpm. Accordingly, the maximum thrust of each motor is 7.89*N*.

$$\boldsymbol{T} = \boldsymbol{4b\Omega^2} \quad \Rightarrow \quad 0 \le T \le 31.56 \quad \textbf{(9)}$$

The maximum bounds of the control moments output before converting to the input of each motor is:

$$\begin{aligned} M_{x_{\max}} &= bl(\Omega_{max}^2) \quad \Rightarrow \quad -1.917 \le M_x \le 1.917 \\ M_{y_{\max}} &= bl(\Omega_{max}^2) \quad \Rightarrow \quad -1.917 \le M_y \le 1.917 \\ M_{z_{\max}} &= d(\Omega_{max}^2 + \Omega_{max}^2) \quad \Rightarrow \quad -0.33 \le M_z \le 0.33 \end{aligned} \quad \textbf{(10)}$$

f) Motor fault modeling

Degradation of motor performance or damage to the rotor can be considered as the partial fault on the actuator regarding the normal operative condition of the motor. Partial fault on the i-th actuator can lead to loss of thrust, which generates unwanted roll, pitch, and yawing moments. Accordingly, the effect of partial fault on the thrust force and moment of the faulty motor is considered as parametric uncertainty as follow:

$$\begin{aligned} T_{i_f} &= T_i + \Delta T_i = b{\Omega_i}^2 + \Delta b{\Omega_i}^2, \Delta b = -f_i b \\ Q_{i_f} &= Q_i + \Delta Q_i = d{\Omega_i}^2 + \Delta d{\Omega_i}^2, \Delta d = -f_i d \end{aligned} \quad \textbf{(11)}$$

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

https://doi.org/10.55212/ijaa.1033224

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

In the above equation, t_f is the time that fault occurs and $0 \leq f_i < 1$, in which $f_i = 0, f_i = 1$ represent the healthy and the fully failed actuator, respectively. Based on the maximum rotational speed of the motors (6250 rpm), each motor can generate the required thrust to keep the quadrotor to hover for about 50% partial fault. Therefore, the maximum partial fault is considered to be 50%.

3. Subsystem Level Operational Architecture

In this section, by applying the multiple-timescales approach, the rotational and translational dynamics are separated by assuming that the rotational dynamics are much faster than the translational dynamics. The block diagram of the controller system is shown in Fig. 2.

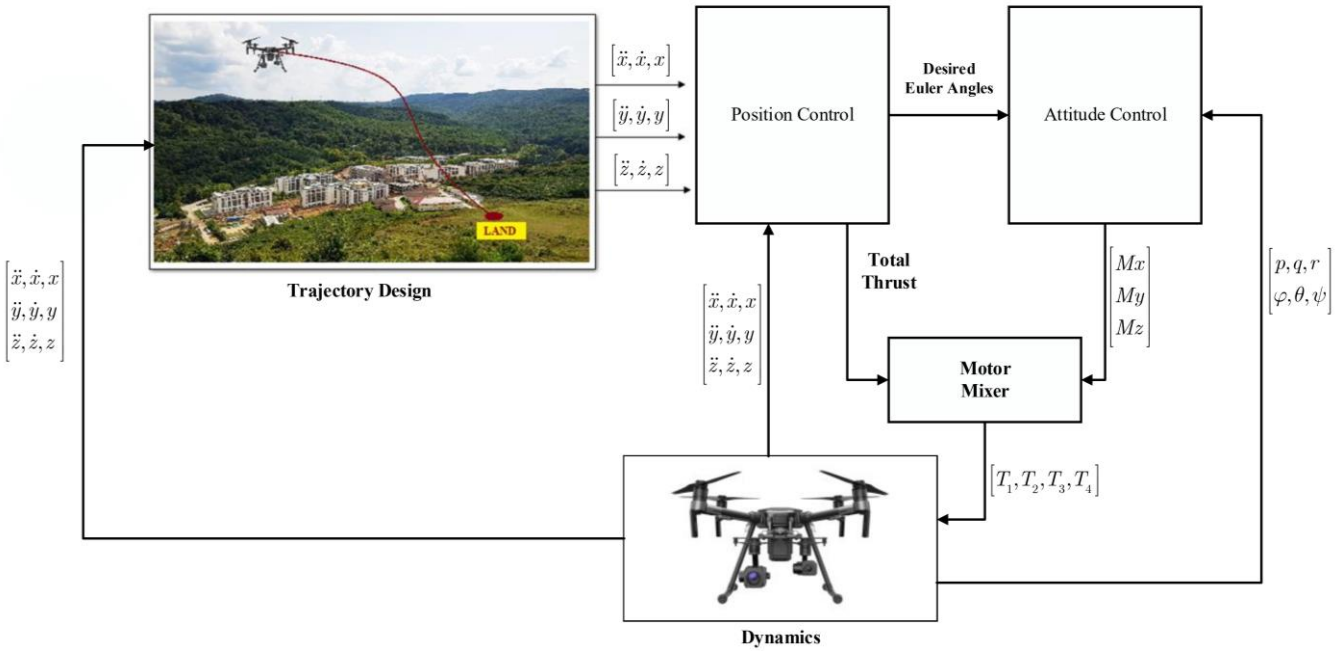

Fig. 2. Total controller architecture

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:

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{g}(\boldsymbol{x})\boldsymbol{U}_c \quad \textbf{(13)}$$

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

a) INDI controller design

Considering the rotational dynamics of the quadrotor based on Eq.(13), the Taylor series approach is applied to expand Eq. (2) while neglecting higher-order terms. Accordingly, Eq. (14) is obtained as below:

https://doi.org/10.55212/ijaa.1033224

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t-Ts)) + \boldsymbol{g}(\boldsymbol{x}(t-Ts))\boldsymbol{U}_c(t-Ts) + \frac{\partial}{\partial \boldsymbol{x}}\boldsymbol{f}(\boldsymbol{x})\Big|_{\substack{x=x(t-Ts)\\ U_c=U_c(t-Ts)}} (\boldsymbol{x}(t)-\boldsymbol{x}(t-Ts)) + \frac{\partial}{\partial \boldsymbol{x}}(g(\boldsymbol{x})\boldsymbol{U}_c)\Big|_{\substack{x=x(t-Ts)\\ U_c=U_c(t-Ts)}} (\boldsymbol{x}-\boldsymbol{x}(t-Ts)) + \frac{\partial}{\partial \boldsymbol{U}_c}\boldsymbol{f}(\boldsymbol{x})\Big|_{\substack{x=x(t-Ts)\\ U_c=U_c(t-Ts)}} (\boldsymbol{U}_c(t)-\boldsymbol{U}_c(t-Ts)) + \boldsymbol{g}(\boldsymbol{x}(t-Ts))(\boldsymbol{U}_c(t)-\boldsymbol{U}_c(t-Ts)) \quad \textbf{(14)}$$

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

The other part of Eq. (14) $\left[\frac{\partial}{\partial \boldsymbol{x}}\boldsymbol{f}(\boldsymbol{x})\Big|_{\substack{x=x(t-Ts)\\ U_c=U_c(t-Ts)}} (\boldsymbol{x}(t)-\boldsymbol{x}(t-Ts)) + \frac{\partial}{\partial \boldsymbol{x}}(g(\boldsymbol{x})\boldsymbol{U}_c)\Big|_{\substack{x=x(t-Ts)\\ U_c=U_c(t-Ts)}} (\boldsymbol{x}-\boldsymbol{x}(t-Ts)) + \frac{\partial}{\partial \boldsymbol{U}_c}\boldsymbol{f}(\boldsymbol{x})\Big|_{\substack{x=x(t-Ts)\\ U_c=U_c(t-Ts)}} (\boldsymbol{U}_c(t)-\boldsymbol{U}_c(t-Ts))\right]$ can be neglected if the sampling time T_s is small. Thus, Eq.(14) can be rewritten as Eq.(15).

$$\dot{\boldsymbol{x}}(t) = \dot{\boldsymbol{x}}(t-Ts) + \boldsymbol{g}(\boldsymbol{x}(t-Ts))(\boldsymbol{U}_c(t)-\boldsymbol{U}_c(t-Ts)) \quad \textbf{(15)}$$

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

$$\dot{p}(t) = \dot{p}(t-Ts) + \frac{1}{I_{xx}}M_x(t) - M_x(t-Ts)$$

$$\dot{q}(t) = \dot{q}(t-Ts) + \frac{1}{I_{yy}}M_y(t) - M_y(t-Ts) \quad \textbf{(16)}$$

$$\dot{r}(t) = \dot{r}(t-Ts) + \frac{1}{I_{zz}}M_z(t) - M_z(t-Ts)$$

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. 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 $\omega_n = 50 rad/s$ and $\zeta = 0.55$. For the same application, other low-pass filters like the Butterworth filter can also be implemented [28].

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

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:

https://doi.org/10.55212/ijaa.1033224

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

where:

$$\dot{\boldsymbol{x}}_f = L^{-1}(\dot{\boldsymbol{x}}(t - Ts)C(s)) \quad \textbf{(19)}$$

Where L^{-1} is the Laplace inverse operator, $\dot{\boldsymbol{x}}_f$ 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 Fig. 3.

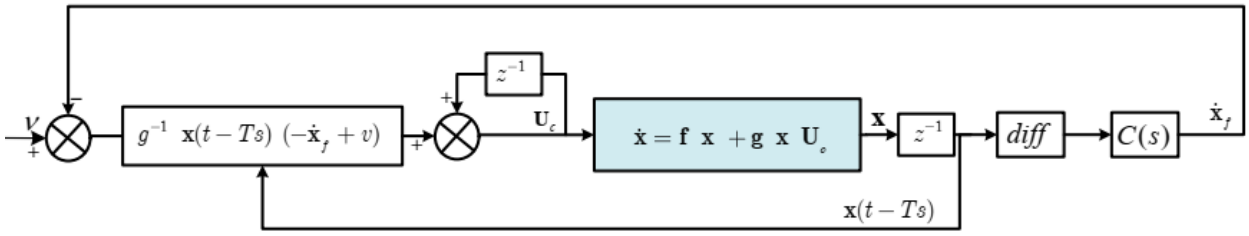

Fig. 3 INDI controller architecture

b) Motor fault modeling

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:

$$\dot{\boldsymbol{x}}_m = \boldsymbol{A}_m\boldsymbol{x}_m + \boldsymbol{B}_m\boldsymbol{R} \quad \textbf{(20)}$$

where $A_m \in R^{n\times n}$ is known desired Hurwitz closed-loop system dynamics and $B_m \in R^{n\times m}$ 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:

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{A}_m\boldsymbol{x}_m + \boldsymbol{B}_m(\omega\boldsymbol{u}_{\boldsymbol{ad}} + \theta\|\boldsymbol{x}\|_\infty + \sigma) + \Delta_1(\boldsymbol{x},\boldsymbol{u}), \boldsymbol{x}(0) = \boldsymbol{x}_0 \\ \boldsymbol{y}(t) &= \boldsymbol{c}^T\boldsymbol{x}(t) \end{aligned} \quad \textbf{(21)}$$

where $c\|.\|_\infty$ is the infinity norm; $cx(t) \in R^n$is the measured system state; $c \in R^{n\times n}$is a known constant vector; $\boldsymbol{u}_{\boldsymbol{ad}}(t) \in R^m$is the control input; ω, **θ, σ** are unknown constant parameters with known signs and lower and upper bounds; $\boldsymbol{\Delta_1}(\boldsymbol{x},\boldsymbol{u})$:$R \times R^n \to R^n$ 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):

$$\begin{aligned} \dot{\hat{\boldsymbol{x}}} &= \boldsymbol{A}_m\hat{\boldsymbol{x}}_m + \boldsymbol{B}_m(\hat{\omega}\boldsymbol{u}_{\boldsymbol{ad}} + \hat{\theta}\|\boldsymbol{x}\|_\infty + \hat{\sigma}), \boldsymbol{x}(0) = \boldsymbol{x}_0 \\ \hat{\boldsymbol{y}}(t) &= \boldsymbol{c}^T\hat{\boldsymbol{x}}(t) \end{aligned} \quad \textbf{(22)}$$

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

Adaptive law: The adaption laws governing the adaptive estimates are as follows [7]:

$$\begin{aligned} \dot{\hat{\theta}}(t) &= \Gamma Proj\dot{\theta}(t), -\tilde{x}^T(t)Pb\|x(t)\|_\infty, & \hat{\theta}(0) &= \hat{\theta}_0, \\ \dot{\hat{\sigma}}(t) &= \Gamma Proj\dot{\sigma}(t), -\tilde{x}^T(t)Pb, & \hat{\sigma}(0) &= \hat{\sigma}_0, \\ \dot{\hat{\omega}}(t) &= \Gamma Proj\dot{\omega}(t), -\tilde{x}^T(t)Pb, & \hat{\omega}(0) &= \hat{\omega}_0, \end{aligned} \quad \textbf{(23)}$$

https://doi.org/10.55212/ijaa.1033224

In Eq.(23), the term Γ is the adaptation gain and $\hat{\theta}_0$, $\hat{\omega}_0$, c$\hat{\sigma}_0$ are the initial values of pertinent variables, which are guessed for initialization of the algorithm. Large values of G, increases 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 $\tilde{x}(t) = \hat{x}(t) - x(t)$ is the error function.

The projection operator, which is denoted by Proj.,. guarantees estimated parameters boundedness according to Ref [7]. The projection operator is defined as below:

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

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

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 Fig. 4.

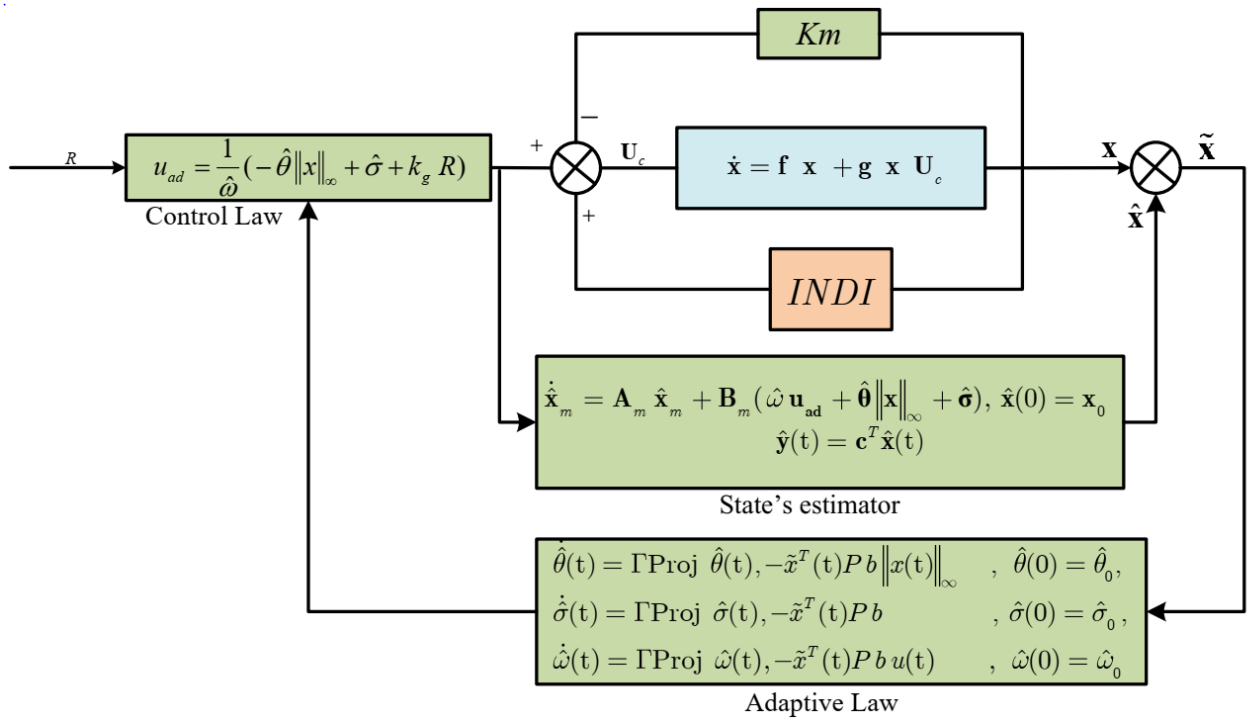

Fig. 4. Model reference robust adaptive controller with INDI Algorithm

c) Outer loop controller design

For the outer-loop position control, a PID control algorithm is applied. Based on the desired trajectory and their first and second derivatives, the dynamics of the position error can be derived as:

https://doi.org/10.55212/ijaa.1033224

$$\ddot{\boldsymbol{P}}_e + K_d\dot{\boldsymbol{P}}_e + K_p\boldsymbol{P}_e + K_I\int \boldsymbol{P}_e\,dt = 0$$
$$\boldsymbol{P}_d = [x_d, y_d, z_d], \boldsymbol{P}_e = \boldsymbol{P}_d - \boldsymbol{P} \quad \textbf{(26)}$$

where $\boldsymbol{P}_d$ is the desired position with bounded first and second derivatives, $\boldsymbol{P}_e$ 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:

$$\ddot{\boldsymbol{P}} = \ddot{\boldsymbol{P}}_d + K_d\dot{\boldsymbol{P}}_e + K_p\boldsymbol{P}_e + K_I\int \boldsymbol{P}_e\,dt = 0 \quad \textbf{(27)}$$

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:

$$\begin{cases} \theta c = \arcsin\left(\dfrac{m\ddot{x}}{\cos\psi_d}T\right) \\ \varphi_c = -\arcsin\left(\dfrac{m_{\ddot{y}}}{\cos\psi_d T}\right) \end{cases} \quad \textbf{(28)}$$

Where in the above equation $T = m\sqrt{\ddot{x} + \ddot{y} + (\ddot{z} + g)}$ 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 Fig. 5., the system output is used in the derivative part without accounting for the desired input, and an anti-windup filter [29] is applied in an integral part of the PID algorithm.

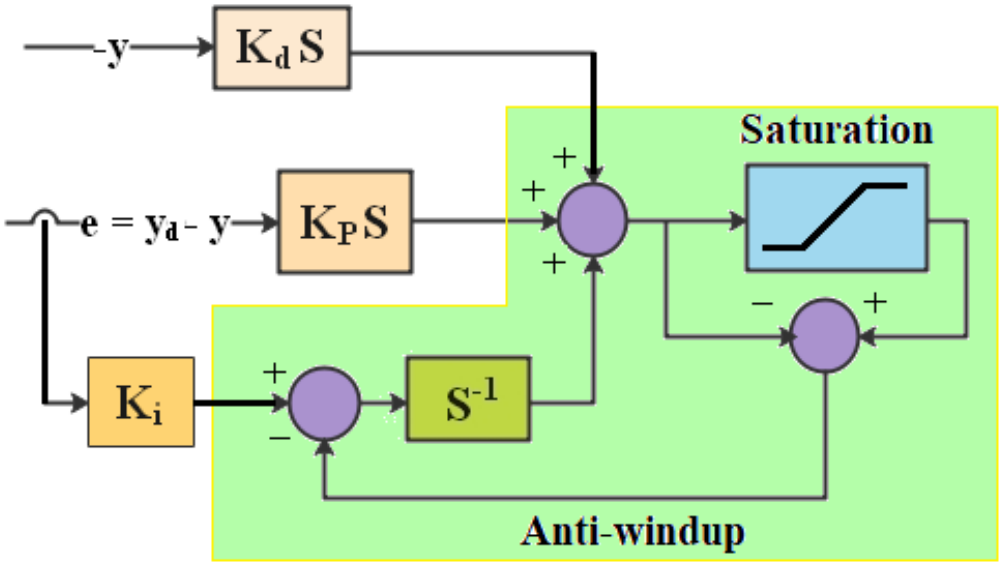

Fig. 5. Anti wind up PID controller architecture

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. In the first simulation scenario, the performance of the introduced controller is investigated for the case of the healthy (no-fault) quadrotor. Accordingly, Figs. 6-10 represent the parameters of the quadrotor, when tracking a helical trajectory. Figs. 6, 7

https://doi.org/10.55212/ijaa.1033224

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 Figs. 8, 9, respectively. Accordingly, the required rotation speed of the rotors to track the desired path are around 4400 RPM. Finally, the quadrotor position in 3D space and the trajectory tracking performance of the proposed controller has been illustrated in Fig. 10. As shown, the controller has a satisfactory tracking performance. 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 for different percentages of fault on the motor number one (number1) as illustrated in Figs. 11-25.

Figures 11-15 illustrate the quadrotor's parameters corresponding to 20% of rotor fault. Body rotational speed, Euler angles, motors' rotational speeds, moments generated by the motors, and the quadrotor position are all illustrated in all fault scenarios. All figures contain the curves of the desired values and the INDI outputs. Similar simulations are run for the case of 40% of fault on the number one rotor according to Fig. 16-Fig. 20. As previously discussed, the controller can maintain full controllability (roll, pitch, and yaw) of the quadrotor to maximum of 50% of rotor fault. Figs. 21-25 illustrate the controller performance in presence of 50% of the motor fault. As illustrated in Fig 23, the rotational speed of rotor number one approaches the saturation magnitude in presence of 50% of partial fault. The maximum rotation speed of the rotors is 6250 RPM, that's why for faults bigger than 50% the rotational speed of the faulty motor saturates and the controller performance degrades drastically.

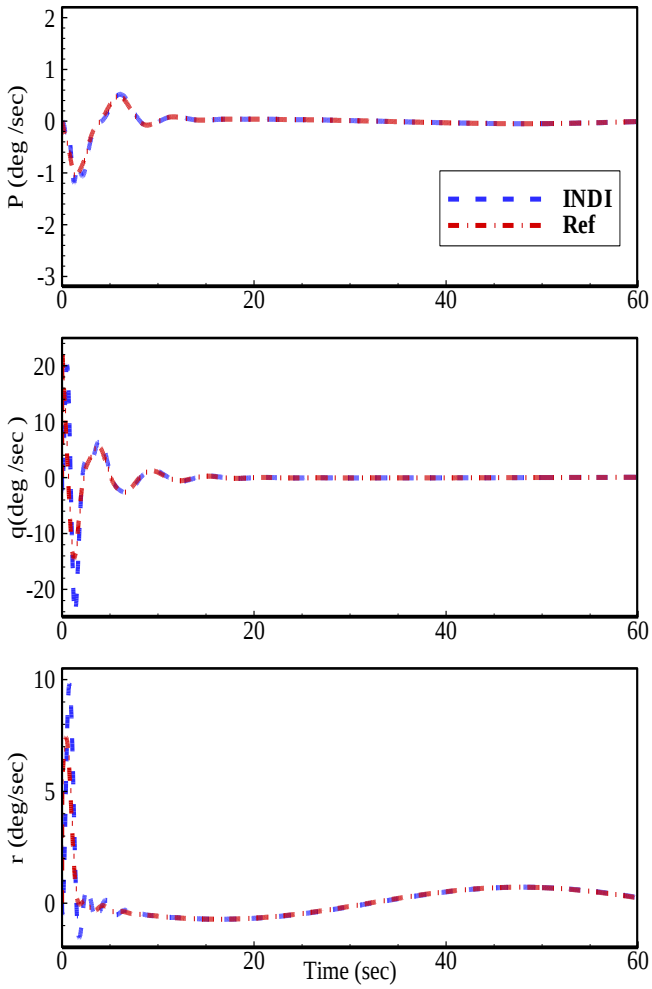

Fig. 6. Quadrotor body angular rates (deg/s)

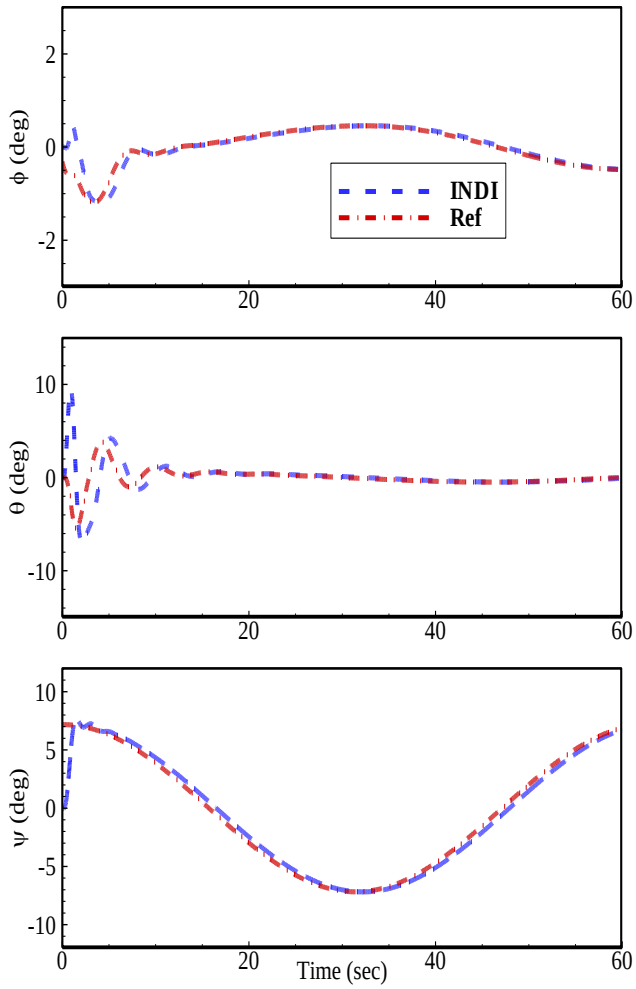

Fig. 7. Quadrotor Euler Angles (deg)

https://doi.org/10.55212/ijaa.1033224

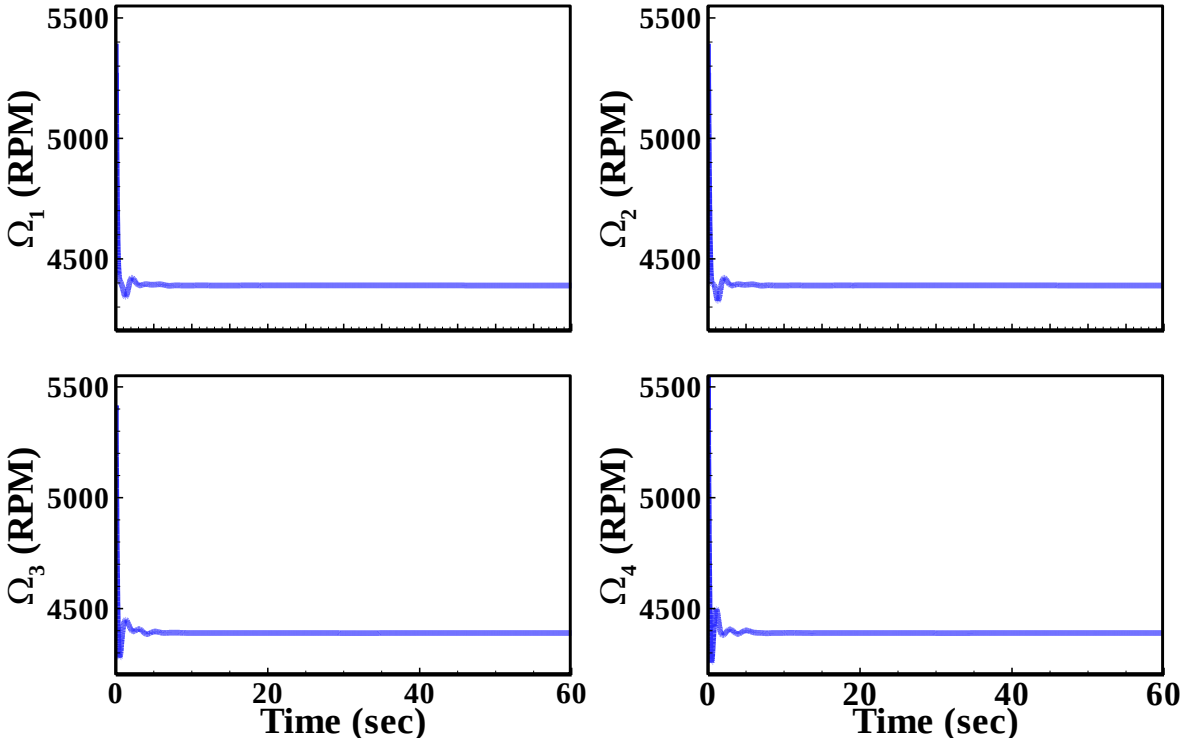

Fig. 8. Quadrotor rotational speeds of motors (RPM)

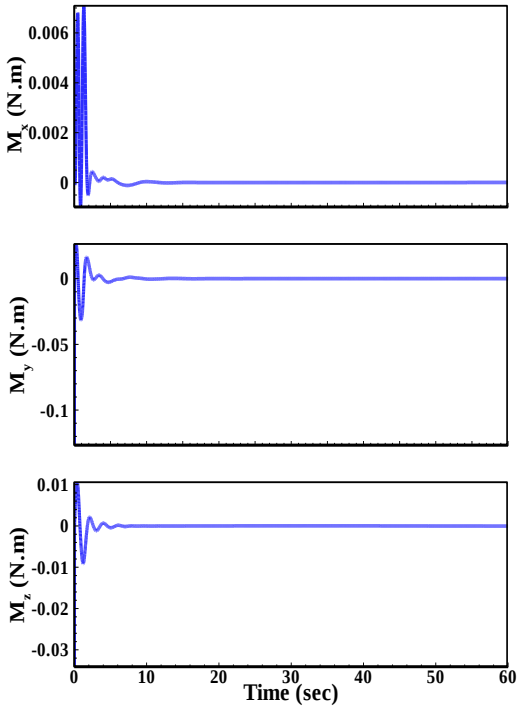

Fig. 9. Quadrotor Controller moments

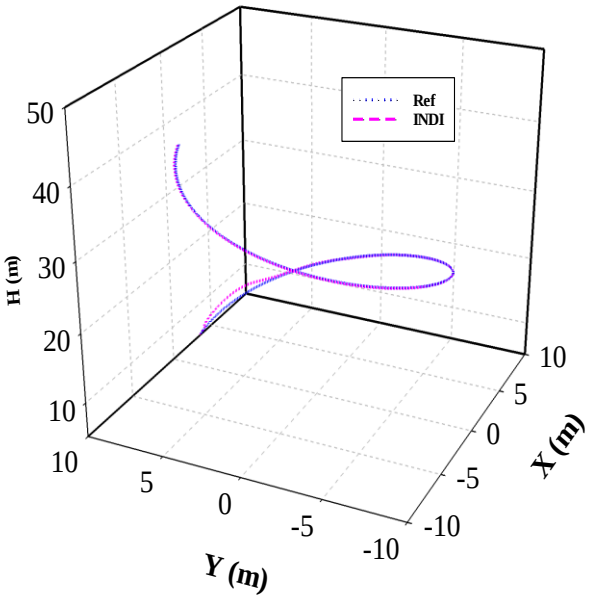

Fig. 10. Quadrotor position in 3D space

https://doi.org/10.55212/ijaa.1033224

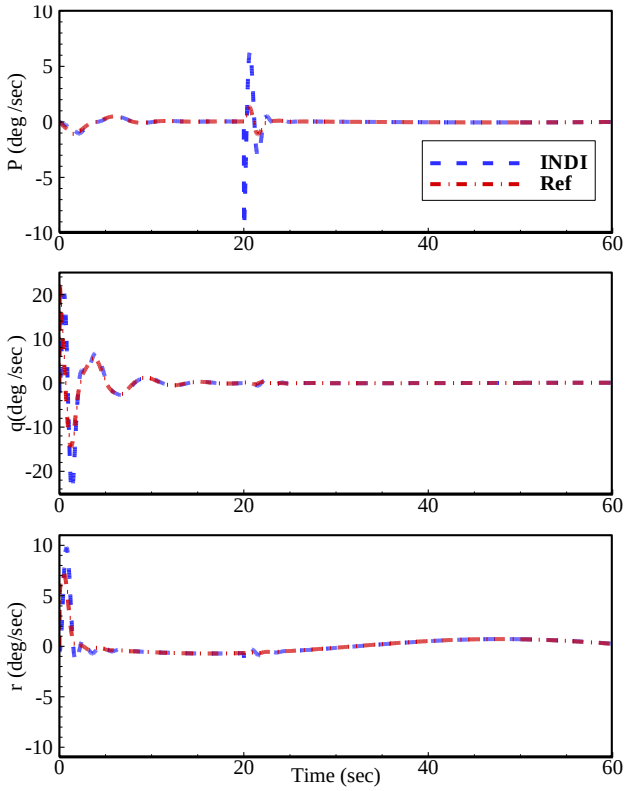

Fig. 11. Angular rates with 20% fault on motor1

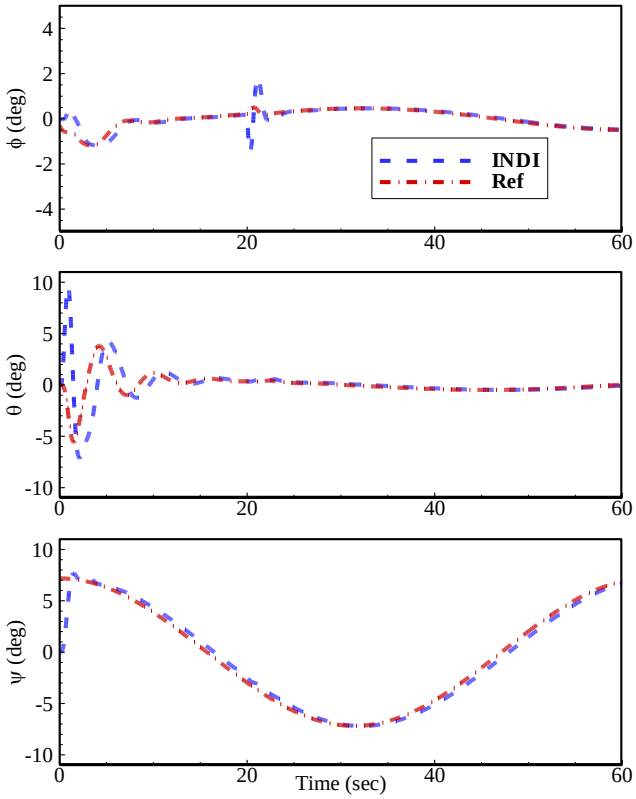

Fig. 12. Euler Angles with 20% fault on motor1

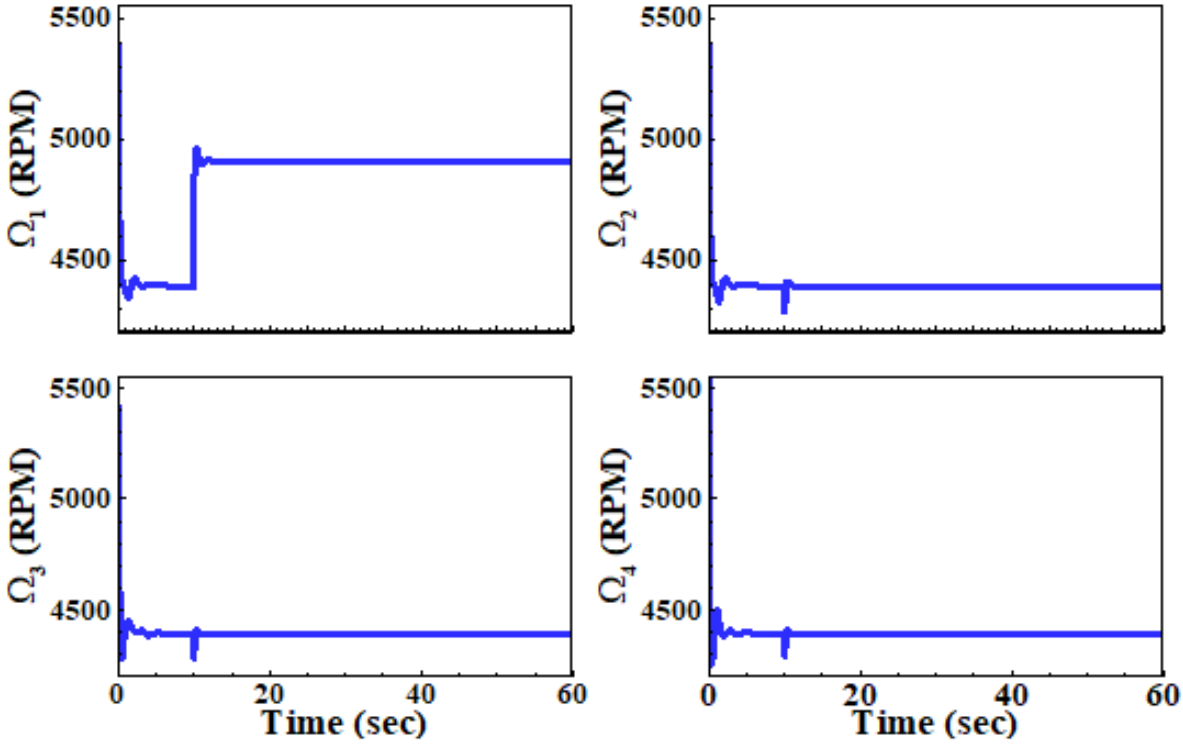

Fig. 13. Motors speeds with 20% fault on motor1

https://doi.org/10.55212/ijaa.1033224

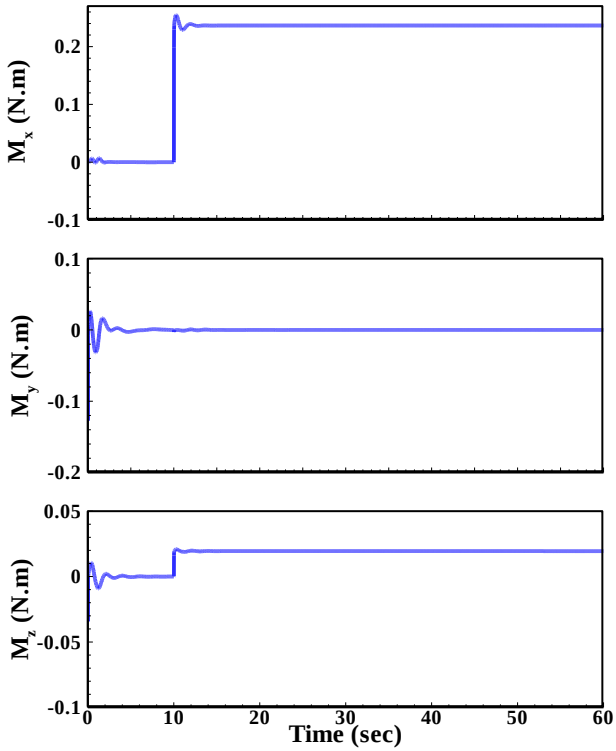

Fig. 14. Controller moments with 20% fault on motor1

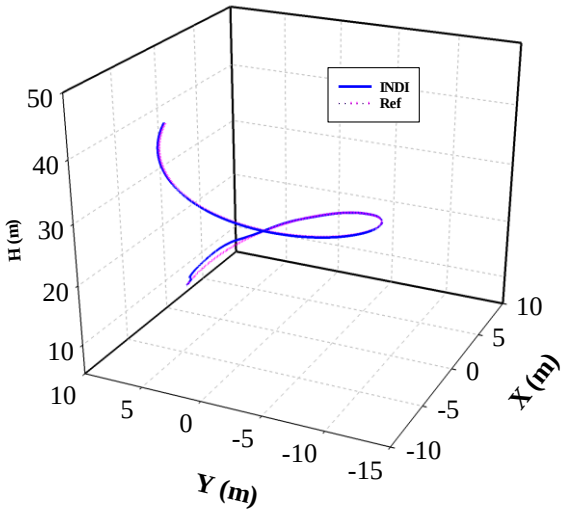

Fig. 15. Trajectory tracking with 20% fault on motor1

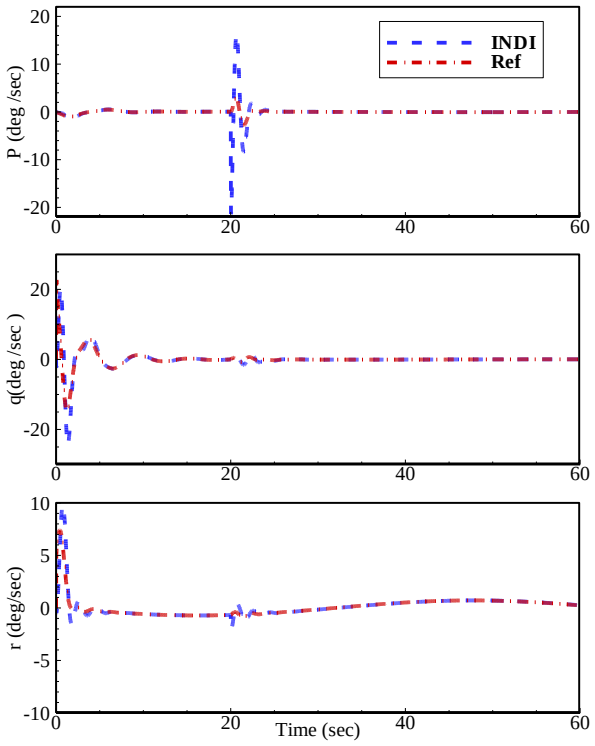

Fig. 16. Angular rates with 40% fault on motor1

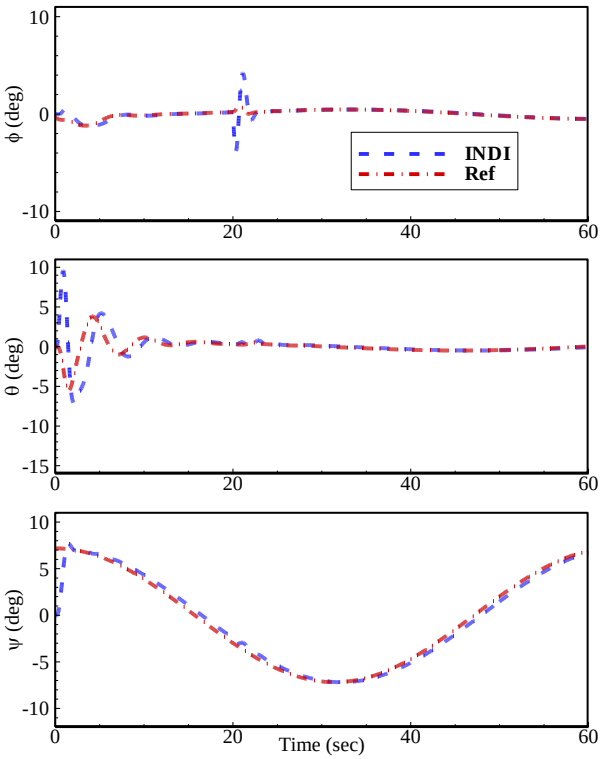

Fig. 17. Euler Angles with 40% fault on motor1

https://doi.org/10.55212/ijaa.1033224

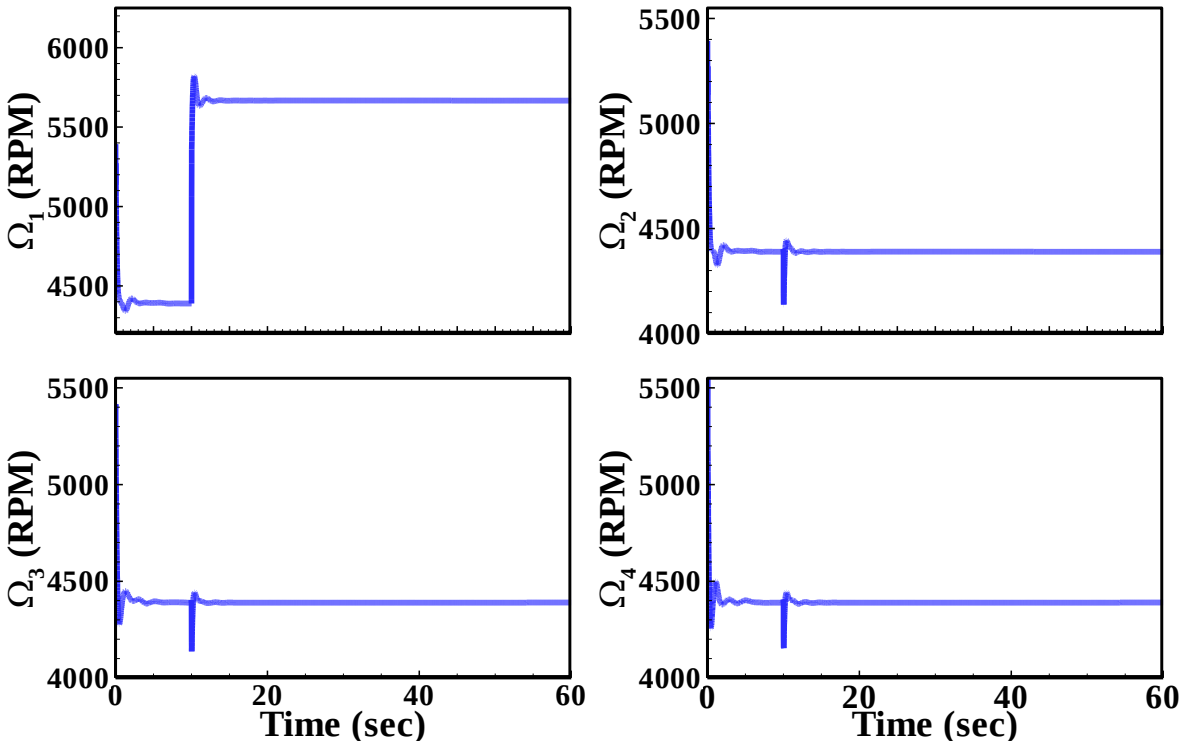

Fig. 18. Motors speeds with 40% fault on motor1

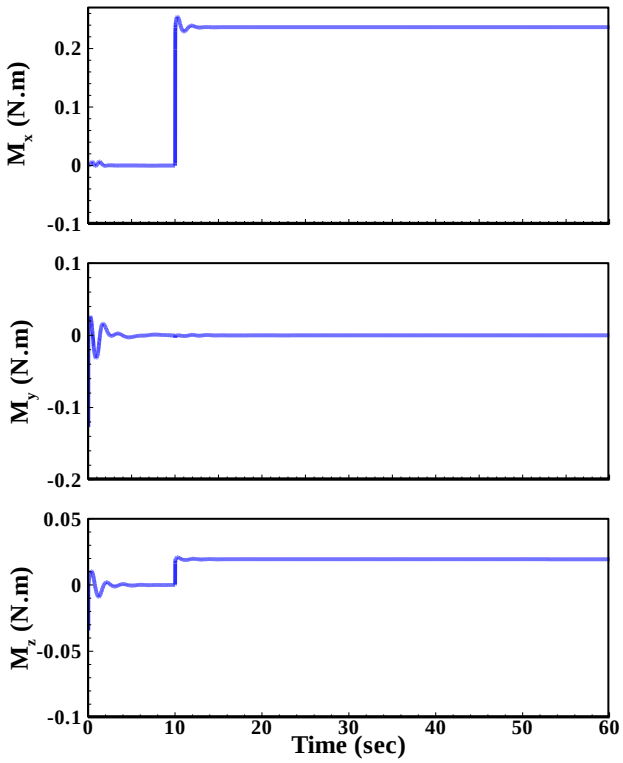

Fig. 19. Controller moments with 40% fault on motor1

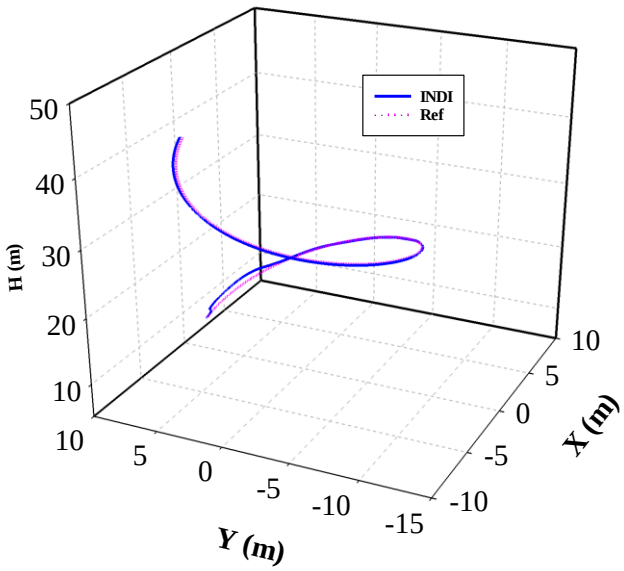

Fig. 20. Trajectory tracking with 40% fault on motor1

https://doi.org/10.55212/ijaa.1033224

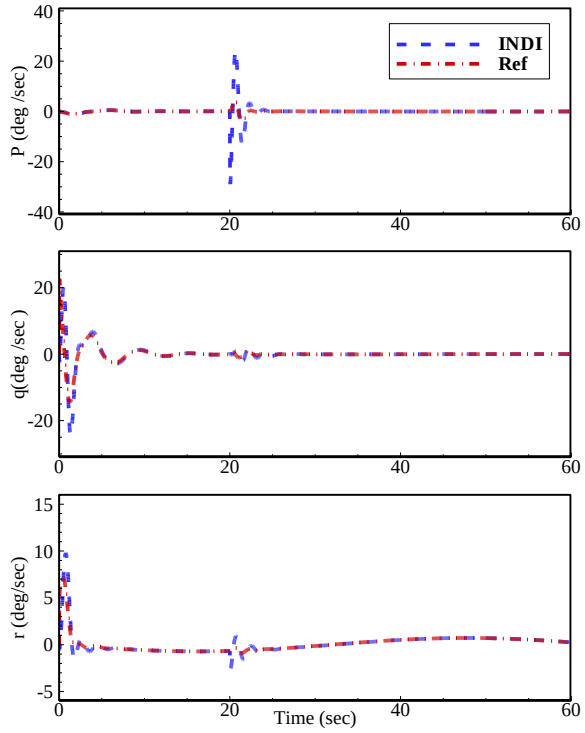

Fig. 21. Angular rates with 50% fault on motor1

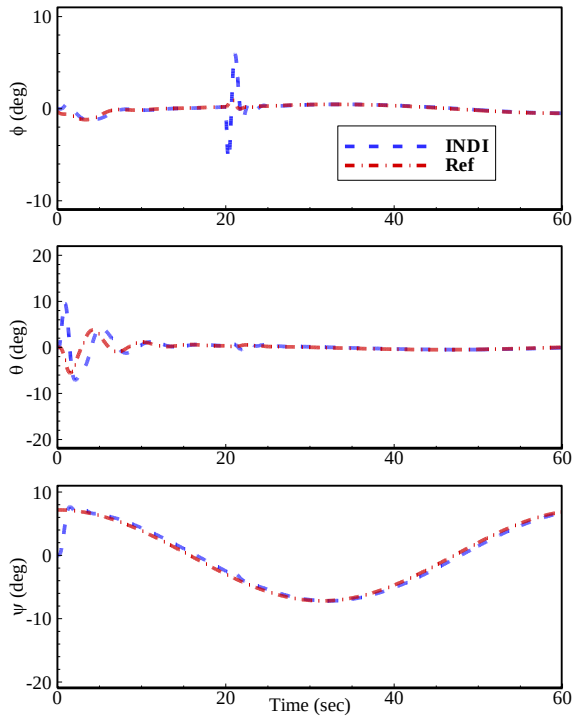

Fig. 22. Euler Angles with 50% fault on motor1

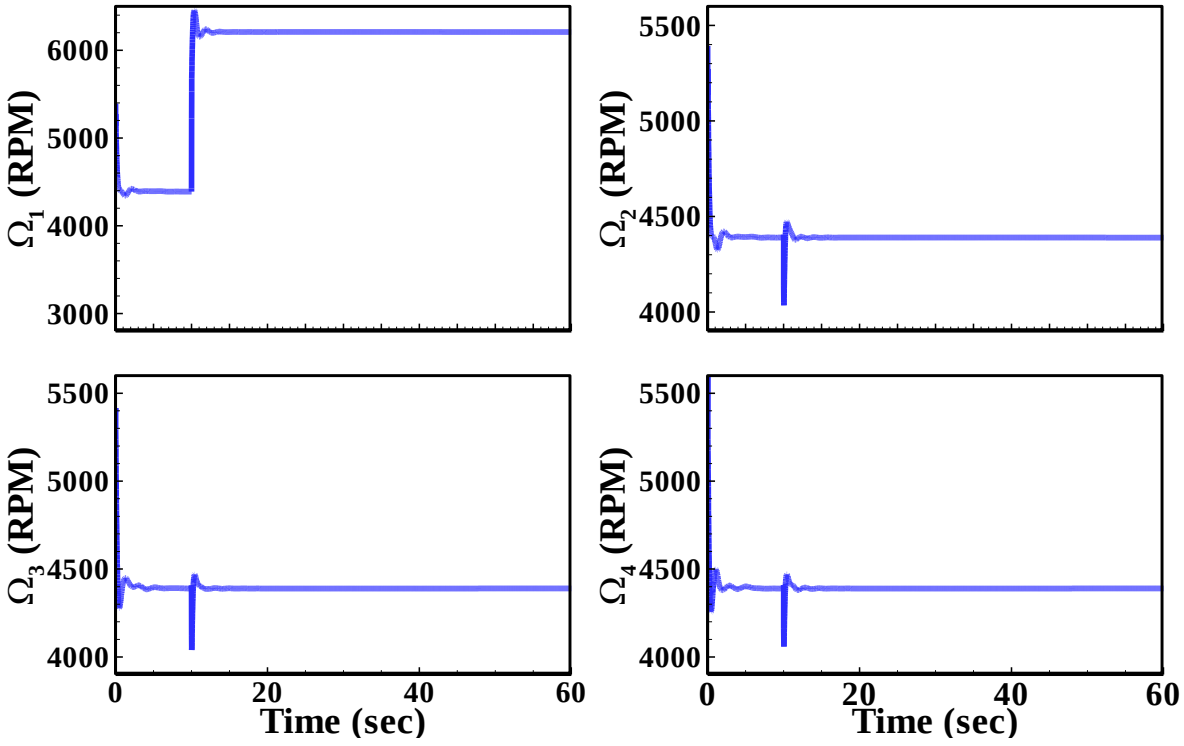

Fig. 23. Motors speeds with 50% fault on motor1

https://doi.org/10.55212/ijaa.1033224

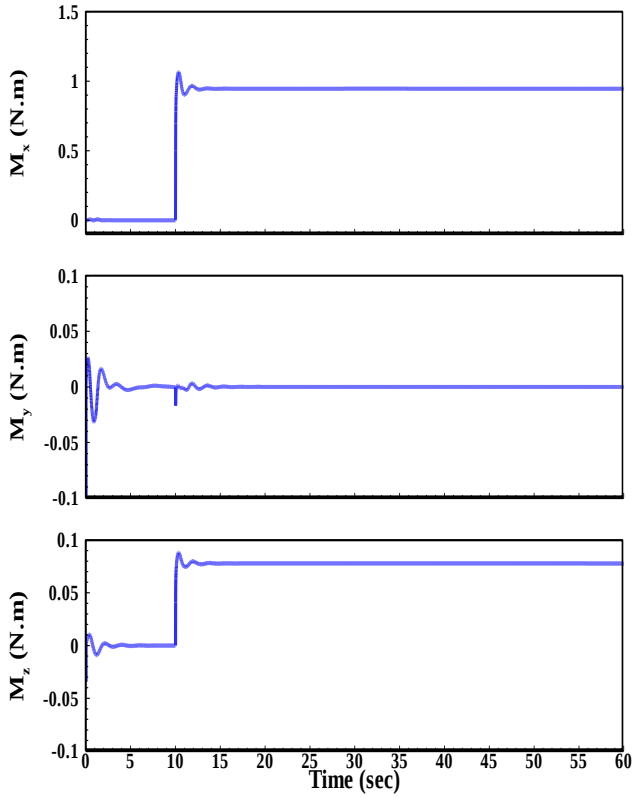

Fig. 24. Controller moments with 50% fault on motor1

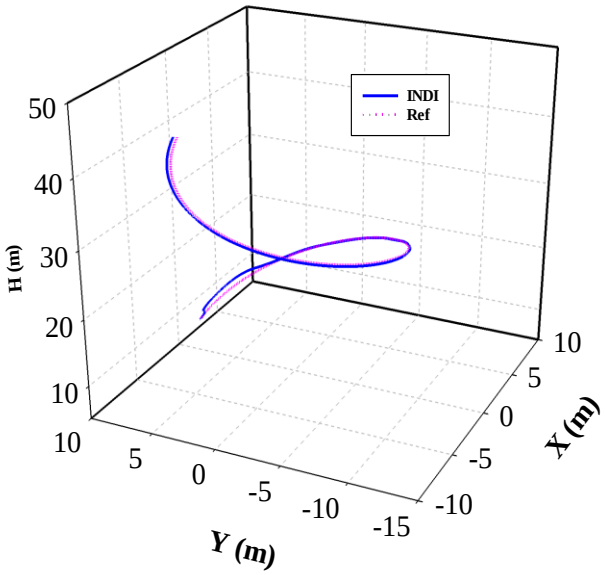

Fig. 25. Trajectory tracking with 50% fault on motor1

5. Conclusion

This paper presents a novel augmentation of Incremental Nonlinear Dynamic Inversion (INDI) as a baseline controller and a model reference robust adaptive algorithm to control and recover the quadrotor in presence of partial actuator fault. The robust adaptive algorithm is augmented to deal with the effect of the un-modeled fault due to the rotors. Different simulation scenarios are run to investigate the performance of the proposed control strategy. According to the simulation results, the proposed control strategy can maintain full controllability of the quadrotor in roll, pitch, and yaw channels in presence of partial faults of the actuator up to 50%. By providing full controllability in all channels, the quadrotor can track the desired trajectories in presence of a partial actuator fault. The results demonstrate that the performance of the INDI controller is desirable, while actuator fault affects the dynamics.

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

References

[1] TURAN, V., AVŞAR, E., ASADİ, D., and AYDIN, E. A., 2021. "Image Processing Based Autonomous Landing Zone Detection for a Multi-Rotor Drone in Emergency Situations." *Turkish Journal of Engineering*, 5, (4).

[2] Asadi, D., 2021. "Partial Engine Fault Detection and Control of a Quadrotor Considering Model Uncertainty." *Turkish Journal of Engineering*, 6, (2), 106–117.

https://doi.org/10.55212/ijaa.1033224

[3] Asadi, D., and Atkins, E. M., 2018. "Multi-Objective Weight Optimization for Trajectory Planning of an Airplane with Structural Damage." *Journal of Intelligent and Robotic Systems: Theory and Applications*, 91, (3–4).

[4] Asadi, D., Sabzehparvar, M., Atkins, E. M., and Talebi, H. A., 2014. "Damaged Airplane Trajectory Planning Based on Flight Envelope and Motion Primitives." *Journal of Aircraft*, 51, (6), 1740–1757.

[5] Asadi, D., Sabzehparvar, M., and Talebi, H. A., 2013. "Damaged Airplane Flight Envelope and Stability Evaluation." *Aircraft Engineering and Aerospace Technology*, 85, (3), 186–198.

[6] Asadi, D., Ahmadi, K., Nabavi-chashmi, S., and Ö., T., 2021. "Controlability of Multi-Rotors under Motor Fault Effect." *Artıbilim: Adana Alparslan Türkeş Bilim ve Teknoloji Üniversitesi Fen Bilimleri Dergisi*, 4, (2), 24–43.

[7] Ahmadi, K., Asadi, D., and Pazooki, F., 2019. "Nonlinear L1 Adaptive Control of an Airplane with Structural Damage." *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 233, (1).

[8] Asadi, D., and Ahmadi, K., 2020. "Nonlinear Robust Adaptive Control of an Airplane with Structural Damage." *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 234, (14), 2076–2088.

[9] Asadi, D., and Bagherzadeh, S. A., 2017. "Nonlinear Adaptive Sliding Mode Tracking Control of an Airplane with Wing Damage." *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 232, (8), 1405–1420.

[10] Alwi, H., and Edwards, C., 2013. Fault Tolerant Control of an Octorotor Using LPV Based Sliding Mode Control Allocation.

[11] Navabi, M., Davoodi, A., and Mirzaei, H., 2021. "Trajectory Tracking of Under-Actuated Quadcopter Using Lyapunov-Based Optimum Adaptive Controller." *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*.

[12] Ahmadi Dastgerdi, K., Pazooki, F., and Roshanian, J., 2020. "Model Reference Adaptive Control of a Small Satellite in the Presence of Parameter Uncertainties." *Scientia Iranica*, 27, (6), 2933–2944.

[13] Asadi, D., Ahmadi, K., and Nabavi, S. Y., 2021. "Fault-Tolerant Trajectory Tracking Control of a Quadcopter in Presence of a Motor Fault." *International Journal of Aeronautical and Space Sciences*.

[14] Gao, Z., Cecati, C., and Ding, S. X., 2015. "A Survey of Fault Diagnosis and Fault-Tolerant Techniques—Part I: Fault Diagnosis with Model-Based and Signal-Based Approaches." *IEEE Transactions on Industrial Electronics*, 62, (6), 3757–3767.

[15] Nguyen, N. P., and Hong, S. K., 2019. "Active Fault-Tolerant Control of a Quadcopter against Time-Varying Actuator Faults and Saturations Using Sliding Mode Backstepping Approach." *Applied Sciences (Switzerland)*, 9, (19).

[16] Barghandan, S., Badamchizadeh, M. A., and Jahed-Motlagh, M. R., 2017. "Improved Adaptive Fuzzy Sliding Mode Controller for Robust Fault Tolerant of a Quadrotor." *International Journal of Control, Automation and Systems*, 15, (1), 427–441.

[17] Lanzon, A., Freddi, A., and Longhi, S., 2014. "Flight Control of a Quadrotor Vehicle Subsequent to a Rotor Failure." *Journal of Guidance, Control, and Dynamics*, 37, (2), 580–591.

[18] Schneider, T., Ducard, G., Rudin, K., and Strupler, P., 2012. *Fault-Tolerant Control Allocation for Multirotor Helicopters Using Parametric Programming*.

[19] Besnard, L., Shtessel, Y. B., and Landrum, B., 2012. "Quadrotor Vehicle Control via Sliding Mode Controller Driven by Sliding Mode Disturbance Observer." *Journal of the Franklin Institute*, 349, (2), 658–684.

[20] Quan, Q., 2017. *Introduction to Multicopter Design and Control*.

https://doi.org/10.55212/ijaa.1033224

[21] Lippiello, V., Ruggiero, F., and Serra, D., 2014. Emergency Landing for a Quadrotor in Case of a Propeller Failure: A Backstepping Approach.

[22] Sun, S., Wang, X., Chu, Q., and Visser, C. d., 2021. "Incremental Nonlinear Fault-Tolerant Control of a Quadrotor with Complete Loss of Two Opposing Rotors." *IEEE Transactions on Robotics*, 37, (1), 116–130.

[23] Sun, S., Cioffi, G., De Visser, C., and Scaramuzza, D., 2021. "Autonomous Quadrotor Flight Despite Rotor Failure with Onboard Vision Sensors: Frames vs. Events." *IEEE Robotics and Automation Letters*.

[24] Jung, W., and Bang, H., 2021. "Fault and Failure Tolerant Model Predictive Control of Quadrotor UAV." *International Journal of Aeronautical and Space Sciences*, 22, (3), 663–675.

[25] Du, G.-X., Quan, Q., and Cai, K.-Y., 2015. "Controllability Analysis and Degraded Control for a Class of Hexacopters Subject to Rotor Failures." *Journal of Intelligent & Robotic Systems*, 78, (1), 143–157.

[26] Lee, J., Choi, H. S., and Shim, H., 2016. "Fault Tolerant Control of Hexacopter for Actuator Faults Using Time Delay Control Method." *International Journal of Aeronautical and Space Sciences*, 17, 54–63.

[27] Nguyen, N. P., Xuan Mung, N., and Hong, S. K., 2019. Actuator Fault Detection and Fault-Tolerant Control for Hexacopter. *Sensors* .19, (21).

[28] Butterworth, S., 1930. "On the Theory of Filter Amplifiers." *Experimental Wireless and the Wireless Engineer*, 7.

https://doi.org/10.55212/ijaa.1033224