sensors

Article

Robust Fault Estimation Using the Intermediate Observer: Application to the Quadcopter

Ngoc Phi Nguyen [1], Tuan Tu Huynh [2,3], Xuan Phu Do [4], Nguyen Xuan Mung [1] and Sung Kyung Hong [1,*]

[1] Department of Aerospace Engineering, Sejong University, Seoul 143-747 (05006), Korea; npnguyen@sejong.ac.kr (N.P.N.); xuanmung1009@gmail.com (N.X.M.)

[2] Department of Electrical Engineering, Yuan Ze University, No. 135, Yuandong Road, Zhongli, Taoyuan 320, Taiwan; huynhtuantu@saturn.yzu.edu.tw

[3] Department of Electrical Electronic and Mechanical Engineering, Lac Hong University, No. 10, Huynh Van Nghe Road, Bien Hoa, Dong Nai 810000, Vietnam

[4] MediRobotics Laboratory, Department of Machatronics and Sensor Systems Technology, Vietnamese-German University, Binh Duong 820000, Vietnam; phu.dx@vgu.edu.vn

* Correspondence: skhong@sejong.ac.kr

Received: 20 May 2020; Accepted: 7 June 2020; Published: 31 August 2020

Abstract: In this paper, an actuator fault estimation technique is proposed for quadcopters under uncertainties. In previous studies, matching conditions were required for the observer design, but they were found to be complex for solving linear matrix inequalities (LMIs). To overcome these limitations, in this study, an improved intermediate estimator algorithm was applied to the quadcopter model, which can be used to estimate actuator faults and system states. The system stability was validated using Lyapunov theory. It was shown that system errors are uniformly ultimately bounded. To increase the accuracy of the proposed fault estimation algorithm, a magnitude order balance method was applied. Experiments were verified with four scenarios to show the effectiveness of the proposed algorithm. Two first scenarios were compared to show the effectiveness of the magnitude order balance method. The remaining scenarios were described to test the reliability of the presented method in the presence of multiple actuator faults. Different from previous studies on observer-based fault estimation, this proposal not only can estimate the fault magnitude of the roll, pitch, yaw, and thrust channel, but also can estimate the loss of control effectiveness of each actuator under uncertainties.

Keywords: fault diagnosis; quadrotor; UAVs; sliding mode observer; sensor fault; quadcopter

1. Introduction

Unmanned aerial vehicles (UAVs) have attracted attention over many years owing to their vital achievements and significant advantages in various applications, such as rescue [1,2], coastal surveillance [3,4], forest monitoring [5,6], military, defense [7,8], and robust control with uncertainties [9–11]. In comparison with manned aerial vehicles, the control of UAVs is more complex since all tasks are operated autonomously through an embedded flight controller or by a pilot.

Quadcopter, a type of UAV system, has been used and developed for various applications owing to its numerous advantages, such as simplicity, small size, indoor and outdoor operation, and agility, which have rendered it more famous than other types of UAV systems. Quadcopters have been widely investigated and developed for different tasks and environments, including tracking control, formation flight, object tracking, remote sensing, and fault-tolerant control (FTC).

In recent years, studies on FTC have received significant study in the scientific community, which could further increase the reliability of the quadcopter during flights. Normally, an FTC can be classified as a passive FTC (PFTC) or an active FTC (AFTC). PFTC techniques have been extensively used in previous studies [12–14], which have the advantage that fault identification is not required in controller design. However, these approaches have a lower fault-tolerance capability. To deal with this issue, AFTC approaches have been proposed to provide improved fault-tolerance capability and condition-based control for flight systems. In these schemes, fault diagnosis (FD) is an essential precondition to identify the magnitude and location of faults, which is combined with a nominal controller to accommodate the effect of faults. Moreover, through FD information, the control law allows for a landing action in an emergency case. As a result, FD is a major feature in the design of the AFTC system.

The problem with the FD approach has been addressed in several studies. In [15–17], a model-based FD was investigated to monitor the sensor fault and actuator fault, which was validated through a set of observer residuals; however, this algorithm is inaccurate and unsuitable for the isolation and identification of faults. In the same context, a polynomial observer was developed to estimate actuator faults. In [18], Aguilar-Sierra et al. proposed an FD technique to estimate actuator faults using a polynomial observer. In [19], an FD method based on a Kalman filter technique was investigated for actuator faults presenting experimental results; but this method lacks robustness against disturbance. In [20], a linear parameter varying technique was developed for a quadrotor helicopter to estimate the faults. Using this fault estimation information, a fault-tolerant controller was designed to accommodate faults. In [21,22], Cen et al. proposed an adaptive law for Thau observer to estimate the time-varying actuator faults in a quadrotor. The FD strategies presented in the above studies show either valuable simulation or experimental results. However, these methodologies do not examine the uncertainties in their mathematical models.

To address the above statement, several effective strategies, such as high-gain observer [23], neural network [24,25], and nonlinear descriptor observer [26], were previously proposed for nonlinear systems, but none of these methods concentrated on the quadcopter model or real experimental tests. Only a few studies investigated the fault diagnosis problem in quadcopter platforms. In [27], actuator fault diagnosis approaches based on fuzzy techniques were investigated in the quadcopter model showing simulation results. In [28], the adaptive observer based on H_∞ was proposed for fault estimation considering the simulation and experimental results. Other effective methods, such as sliding mode observers (SMOs) have been developed to enhance the robustness of fault identification and time convergence. In this sense, SMO based on a linear parametric varying technique was proposed for actuator fault reconstruction [29]. In [30], a sliding- mode observer combined with Thau observer was proposed to estimate the magnitude of the fault showing experimental works. However, the current studies on quadcopters are still based on assumptions or observer-matching conditions.

Motivated by the above challenges, this article proposes a fault estimation strategy for a quadcopter system with relaxing matching conditions. To be more specific, an intermediate variable combined with an intermediate observer has been improved to estimate both states and faults. The stability of the proposed algorithm was proved using the Lyapunov theory. It was shown that the error system is uniformly ultimately bounded. In contrast to previous results, the main contributions of this article are as follows:

- An intermediate estimator is provided in the quadcopter system to estimate states and faults in the presence of uncertainties. The proposed method aims to relax the matching conditions or equation constraints.
- Stability analysis is described to validate the convergence of the error system.
- The experimental works on the quadcopter are demonstrated to validate the proposed algorithm.

The remainder of this article is organized as follows. Section 2 provides the mathematical model of the quadcopter system. In Section 3, the proposed FD method is introduced. Experimental results are described in Section 4. The conclusions and further works are stated in Section 5.

2. Mathematical Model of the Quadcopter

The body frame B and inertial frame E are used to present the dynamics of the quadcopter, as shown in Figure 1. In the body frame, the XY plane is located at the surface, and the *Z*-axis follows the right-hand rule. In the quadcopter system, the center of gravity is located at the origin of the body frame. The rotation matrix is defined to transform from the body frame to the inertial frame as

$$R(\phi,\theta,\psi) = \begin{bmatrix} c\psi c\theta & c\psi s\theta s\phi - s\psi c\phi & c\psi s\theta c\phi + s\psi c\phi \\ s\psi c\theta & s\psi s\theta s\phi + c\psi c\phi & s\psi s\theta c\phi - s\phi c\psi \\ -s\theta & c\theta s\psi & c\theta c\theta \end{bmatrix} \tag{1}$$

where s and c denote sin and cos; ϕ, θ, ψ denote Euler angles.

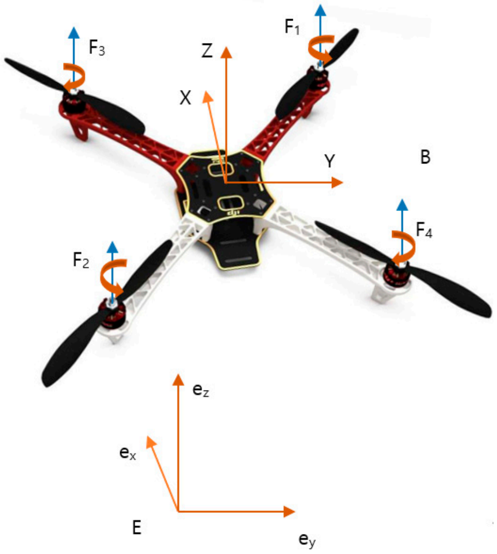

Figure 1. Quadcopter model.

The quadcopter includes two motors (1, 2) rotating in a counterclockwise direction and the remaining two motors rotating clockwise. The four control variables are expressed as:

$$\begin{cases} U_1 = F_1 + F_2 + F_3 + F_4 \\ U_2 = (-F_1 + F_2 + F_3 - F_4)L\sqrt{2}/2 \\ U_3 = (F_1 - F_2 + F_3 - F_4)L\sqrt{2}/2 \\ U_4 = \tau_1 + \tau_2 - \tau_3 - \tau_4 \end{cases} \tag{2}$$

where $\tau_i \approx K_d u_i$ and $F_i \approx K_{th} u_i$ denote the respective torques and forces produced by the *ith* motor; u_i is the pulse-width modulation; L is the arm length; U_1 is the thrust force; and U_2, U_3, U_4 are the torques in the directions ϕ, θ, ψ, respectively.

Equation (2) can be rewritten as

$$\begin{bmatrix} U_2 \\ U_3 \\ U_4 \\ U_1 \end{bmatrix} = \begin{bmatrix} -K_{th}L\sqrt{2}/2 & K_{th}L\sqrt{2}/2 & K_{th}L\sqrt{2}/2 & -K_{th}L\sqrt{2}/2 \\ K_{th}L\sqrt{2}/2 & -K_{th}L\sqrt{2}/2 & K_{th}L\sqrt{2}/2 & -K_{th}L\sqrt{2}/2 \\ K_d & K_d & -K_d & -K_d \\ K_{th} & K_{th} & K_{th} & K_{th} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} \tag{3}$$

The body dynamics are described using the Newton–Euler equation as [31]:

$$\begin{cases} m\ddot{r} = R\begin{bmatrix} 0 \\ 0 \\ U_1 \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} - \dot{\omega} \times m\ddot{r} \\ I\ddot{\omega} = \begin{bmatrix} U_2 \\ U_3 \\ U_4 \end{bmatrix} - \dot{\omega} \times I\ddot{\omega} \end{cases} \tag{4}$$

where m is the total mass; g is the gravity; ω and I are the angular velocity and the inertia vector, respectively; r is the position in an inertial frame.

The dynamics of the quadcopter can be presented through six equations [32]:

$$\begin{cases} \ddot{x} = \left\{U_1(\cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi) - K_x\dot{x}\right\}/m \\ \ddot{y} = \left\{U_1(\cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi) - K_y\dot{y}\right\}/m \\ \ddot{z} = -g + \left\{U_1(\cos\phi\cos\theta) - K_z\dot{z}\right\}/m \\ \ddot{\phi} = \left(U_2 + (I_2 - I_3)\dot{\theta}\dot{\psi} - K_\phi\dot{\phi}\right)/I_1 \\ \ddot{\theta} = \left(U_3 + (I_3 - I_1)\dot{\phi}\dot{\psi} - K_\theta\dot{\theta}\right)/I_2 \\ \ddot{\psi} = \left(U_4 + (I_1 - I_2)\dot{\phi}\dot{\theta} - K_\psi\dot{\psi}\right)/I_3 \end{cases} \tag{5}$$

where $I_1,\ I_2,\ I_3$ represent the moment inertia along the $x,\ y,\ z$ axes; and $K_\phi,\ K_\theta,\ K_\psi,\ K_x,\ K_y$ represent drag coefficients.

Let $x^T = \begin{bmatrix} \phi & \theta & \psi & \dot{\phi} & \dot{\theta} & \dot{\psi} \end{bmatrix}$ be the state vector, $u^T = \begin{bmatrix} U_\phi & U_\theta & U_\psi & U_T \end{bmatrix}$ be the control input vector, and $y = \begin{bmatrix} \phi & \theta & \psi & \dot{\phi} & \dot{\theta} & \dot{\psi} \end{bmatrix}$ be the output vector. Equation (4) is rewritten as

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) + N\xi(t, x(t)) + G \\ y(t) = Cx(t) \end{cases} \tag{6}$$

where $N = I_{4\times4}$ is the distribution matrix $d(t)$ denotes the disturbance vector; $A = \begin{bmatrix} 0_{4\times4} & I_{4\times4} \\ 0_{4\times4} & 0_{4\times4} \end{bmatrix}$,
$C = I_{8\times8}$, $B = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}^T$, $\xi(t, x(t)) = \begin{bmatrix} (\dot{\theta}\dot{\psi}(I_2 - I_3) - \bar{J}_m\dot{\theta}\Omega)/I_1 \\ (\dot{\phi}\dot{\psi}(I_3 - I_1) - \bar{J}_m\dot{\phi}\Omega)/I_2 \\ \dot{\phi}\dot{\theta}(I_1 - I_2)/I_3 \\ 0 \end{bmatrix}$;
$G = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & -g \end{bmatrix}^T$; $U_\phi = U_2/I_1;\ U_\theta = U_3/I_2;\ U_\psi = U_4/I_3;\ U_T = U_1/m.$

3. Methodology

The aim of this section is to provide the fault estimation technique based on the intermediate observer and reducing fault estimation error method for the quadcopter system. In detail, a robust intermediate observer is presented in Section 3.1 to estimate the fault magnitude of the roll, pitch, yaw, and thrust motion. In Section 3.2, a magnitude order balance method is applied to reduce the fault estimation error, which is caused by an imbalance magnitude problem of the yaw motion. Moreover, this section also presents a technique to obtain the loss of control effectiveness (LoCE) of each actuator through the presented method in Section 3.1.

3.1. Design of the Intermediate Observer

When a fault occurs, Equation (6) can be written as:

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) + N\xi(t, x(t)) + G + E_a f_a(t) \\ y(t) = Cx(t) \end{cases} \tag{7}$$

where E_a is the fault matrix, $f_a(t)$ is the fault vector.

To design a robust intermediate observer [33], some assumptions need to be considered.

Assumption 1. *The unknown disturbance is norm-bounded with constant* θ_1 *, i.e.,* $\|\xi(t, x(t))\| \leq \theta_1$ *, where* $0 \leq \theta_1 \leq \infty$.

Assumption 2. *The first derivative of* $f_a(t)$ *satisfies* $\|\dot{f}_a(t)\| \leq \theta_2$ *, with* $0 \leq \theta_2 \leq \infty$.

Assumption 3. E_a *has a full column rank.*

Assumption 4. *There exists a complex number* λ *with non-negative real part that satisfies*

$$rank\begin{bmatrix} A - \lambda I & E_a \\ C & 0 \end{bmatrix} = n + rank(E_a) \tag{8}$$

An auxiliary variable is constructed as

$$\tau = f_a(t) - Sx(t) \tag{9}$$

where $S = \alpha {E_a}^T$; α is a constant that needs to be chosen.

From Equations (7) and (9), we obtain

$$\dot{\tau} = \dot{f}_a - S(Ax(t) + Bu + E_a\tau + E_aSx(t) + N\xi(t, x(t) + G)) \tag{10}$$

The intermediate estimator is introduced as:

$$\dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t) + E_a\hat{f}_a(t) + G + L(y(t) - \hat{y}(t)) \tag{11}$$

$$\dot{\hat{\tau}}(t) = -S(A\hat{x}(t) + Bu(t) + E_a\hat{\tau}(t) + E_aS\hat{x}(t) + G) \tag{12}$$

$$\hat{y}(t) = C\hat{x}(t) \tag{13}$$

$$\hat{f}_a(t) = \hat{\tau}(t) + S\hat{x}(t) \tag{14}$$

where $\hat{x}(t)$, $\hat{\tau}(t)$, $\hat{y}(t)$, $\hat{f}_a(t)$ are the state observer vector, intermediate estimator, output observer vector, and fault estimation of $f_a(t)$, respectively.

Let $e_x = x(t) - \hat{x}(t)$ and $e_f(t) = f_a(t) - \hat{f}_a(t)$. The error system is described as

$$\dot{e}_x = (A - LC)e_x + E_a e_f + N\xi(t, x(t)) \tag{15}$$

$$\dot{e}_\tau = \dot{f}_a - \alpha E^T(Ae_x + E_a e_\tau + E_a S e_x + N\xi(t, x(t))) \tag{16}$$

Theorem 1. *If we apply the intermediate estimator Equations (11)–(14) to state-space model (7) and using Assumptions 1–4, and there exist scalars* $\alpha > 0$, $\varepsilon > 0$ *and matrix L such that*

$$\begin{bmatrix} \Pi_{11} & \Pi_{12} \\ * & \Pi_{22} \end{bmatrix} < 0 \tag{17}$$

where $\Pi_{11} = (A-LC)^T + (A-LC) + 2\alpha E_a E_a^T + \frac{1}{\varepsilon}NN^T$, $\Pi_{12} = E - \mu\alpha A^T E_a - \mu\alpha^2 E_a E_a^T E_a$, *then* $\Pi_{22} = \frac{\mu}{\varepsilon} - 2\mu\alpha E_a^T E_a + \frac{\mu\alpha}{\varepsilon} E_a^T NN^T E_a$.

Then, the error systems (15) and (16) are uniformly ultimately bounded.

Proof. Choosing the Lyapunov function as:

$$V = e_x^T e_x + \mu e_\tau^T e_\tau \tag{18}$$

The first derivative of $V(t)$ is given as:

$$\begin{aligned}
\dot{V} \quad &= \dot{e}_x^T e_x + e_x^T \dot{e}_x + \dot{e}_\tau^T e_\tau + e_\tau^T \dot{e}_\tau \\
&= \Big[(A-LC)e_x + E_a e_f + N\xi(t,x(t))\Big]^T e_x \\
&+ e_x^T \Big[(A-LC)e_x + E_a e_f + N\xi(t,x(t))\Big] \\
&+ \mu \Big[\dot{f}_a - S(Ae_x + E_a e_\tau + E_a S e_x + N\xi(t,x(t))\Big]^T e_\tau \\
&+ \mu e_\tau^T \Big[\dot{f}_a - S(Ae_x + E_a e_\tau + E_a S e_x + N\xi(t,x(t))\Big] \\
&= e_x^T \Big[(A-LC)^T + (A-LC)\Big] e_x + 2e_x^T E_a e_f \\
&+ 2e_x^T N\xi(t,x(t)) \ + 2\mu e_\tau^T \dot{f}_a - 2\mu\alpha e_\tau^T E_a^T (A + \alpha E_a E_a^T) e_x \\
&- 2\mu\alpha e_\tau^T E_a^T E_a e_\tau \ - 2\mu\alpha e_\tau^T E^T N\xi(t,x(t))
\end{aligned} \tag{19}$$

Using the Lemma, we obtain

$$2e_x^T N\xi(t,x(t)) \leq \frac{1}{\varepsilon} e_x^T NN^T e_x + \varepsilon\theta_1^2 \tag{20}$$

$$2e_\tau^T \dot{f} \leq \frac{1}{\varepsilon} e_\tau^T e_\tau + \varepsilon\overline{\theta}_2^2 \tag{21}$$

$$-2e_\tau^T E_a^T N\xi(t,x(t)) \leq \frac{1}{\varepsilon} e_\tau^T E_a^T NN^T E_a + \varepsilon\overline{\theta}_1^2 \tag{22}$$

From (19)–(22) and $e_f = e_\tau + Se_x = e_\tau + \alpha E^T e_x$, we obtain

$$\begin{aligned}
\dot{V} \quad &= e_x^T \Big[(A-LC)^T + (A-LC)\Big] e_x + 2e_x^T E_a e_\tau + 2\alpha e_x^T E_a E_a^T e_x + \tfrac{1}{\varepsilon} e_x^T NN^T e_x \\
&+ \tfrac{\mu}{\varepsilon} e_\tau^T e_\tau - 2\mu\alpha e_\tau^T E_a^T (A + \alpha E_a E_a^T) e_x - 2\mu\alpha e_\tau^T E_a^T E_a e_\tau \\
&+ \tfrac{\mu\alpha}{\varepsilon} e_\tau^T E_a^T NN^T E_a e_\tau + \varepsilon(1+\mu\alpha)\overline{\theta}_1^2 + \mu\varepsilon\overline{\theta}_2^2
\end{aligned} \tag{23}$$

Let us define $\widetilde{e}(t) = \begin{bmatrix} e_x^T & e_\tau^T \end{bmatrix}^T$, then, Equation (23) becomes

$$\dot{V} \leq \widetilde{e}^T \Sigma_1 \widetilde{e}^T + \varepsilon(1+\mu\alpha)\overline{\theta}_1^2 + \mu\varepsilon\overline{\theta}_2^2 \tag{24}$$

where

$$\Sigma_1 = \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ * & \Sigma_{22} \end{bmatrix} \tag{25}$$

with

$$\Sigma_{11} = (A-LC)^T + (A-LC) + 2\alpha E_a E_a^T + \frac{1}{\varepsilon}NN^T \tag{26}$$

$$\Sigma_{12} = E_a - \mu\alpha A^T E_a - \mu\alpha^2 E_a E_a^T E_a \tag{27}$$

$$\Sigma_{22} = \frac{\mu}{\varepsilon} I_n - 2\mu\alpha E_a^T E_a + \frac{\mu\alpha}{\varepsilon} E_a^T NN^T E_a \tag{28}$$

If the following inequality holds,

$$\begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ * & \Sigma_{22} \end{bmatrix} < 0 \tag{29}$$

then, one can achieve $\dot{V}(t) < -\sigma\|\widetilde{e}(t)\|^2 + \eta$, where $\eta = \varepsilon(1+\mu\alpha)\overline{\theta}_1^2 + \mu\varepsilon\overline{\theta}_2^2$ and $\sigma = \lambda_{\min}(-\Sigma_1)$. It follows that $\dot{V}(t) < 0$ for $\sigma\|\widetilde{e}(t)\|^2 > \eta$, which indicates that $(e_x(t), e_\tau(t))$ converges to a small set according to the Lyapunov theory. Therefore, the system error is uniformly ultimately bounded. □

3.2. Magnitude Order Balance and Fault Estimation of Each Actuator

The control input vector $u = \begin{bmatrix} U_\phi & U_\theta & U_\psi & U_T \end{bmatrix}^T$ is obtained from thrusts and torques in Equation (6), and is used to estimate the actuator fault $f_a = \begin{bmatrix} f_\phi & f_\theta & f_\psi & f_T \end{bmatrix}^T$ using the intermediate observer. However, the fault estimation from the yaw motion has a large error compared with the roll, pitch, and thrust motions due to the imbalance of the magnitude order [21]. To overcome this issue, the magnitude order of the four channels should be adjusted in the same range by using the following modified control input vector

$$u = \begin{bmatrix} U_\phi & U_\theta & vU_\psi & U_T \end{bmatrix}^T \tag{30}$$

where v are adjustment gains.

Theorem 2. *Assume that the new estimation values $\hat{f}_\psi^*$ are obtained from the intermediate observer and the estimation values of f_ψ do not change. If the former fault estimation of yaw motion is denoted by $\hat{f}_\psi$, then the following relationship should hold:*

$$\hat{f}_\psi^* = v f_\psi \tag{31}$$

Proof. Recall the intermediate observer design

$$\dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t) + E_a\hat{f}_a(t) + L(y(t) - \hat{y}(t)) \tag{32}$$

If the error system converges to a small set as mentioned in Section 3.1, and the following requirement is satisfied: $e_y = 0$, $\dot{\hat{x}}(t) = 0$ and $\hat{x}(t) = 0$. The fault estimation vector $\hat{f}_a(t)$ can be obtained from

$$Bu(t) + E_a\hat{f}_a(t) = 0 \tag{33}$$

For the yaw motion, Equation (12) is described as

$$U_\psi + \hat{f}_\psi(t) = 0 \tag{34}$$

If $U_\psi^* = vU_\psi$ then $\hat{f}_\psi^* = v\hat{f}_\psi$.

When the new fault estimation algorithm is achieved, the desired real fault estimation can be achieved as

$$\hat{f}_\psi = \hat{f}_\psi^*/v \tag{35}$$

□

Remark 1. *The modification of the control input vector using the adjustment gain factor technique does not affect the intermediate observer. However, it is applied to handle the magnitude order unbalance issue and reduce the fault estimation errors, which is discussed in more detail in the experimental section described in Section 4.*

To estimate the fault magnitude of each motor, the dynamic system (7) is rewritten as

$$\begin{cases} \dot{x}(t) = Ax(t) + B_1 u^*(t) - B_1 \Gamma \Lambda + N\xi(t, x(t)) + G \\ y(t) = Cx(t) \end{cases} \tag{36}$$

where $B_1 = \begin{bmatrix} 0 & 0 & 0 & 0 & -\frac{K_{th}L\sqrt{2}}{2I_1} & \frac{K_{th}L\sqrt{2}}{2I_2} & \frac{K_d}{I_3} & K_{th} \\ 0 & 0 & 0 & 0 & \frac{K_{th}L\sqrt{2}}{2I_1} & -\frac{K_{th}L\sqrt{2}}{2I_2} & \frac{K_d}{I_3} & K_{th} \\ 0 & 0 & 0 & 0 & \frac{K_{th}L\sqrt{2}}{2I_1} & \frac{K_{th}L\sqrt{2}}{2I_2} & -\frac{K_d}{I_3} & K_{th} \\ 0 & 0 & 0 & 0 & -\frac{K_{th}L\sqrt{2}}{2I_1} & -\frac{K_{th}L\sqrt{2}}{2I_2} & -\frac{K_d}{I_3} & K_{th} \end{bmatrix}^T$; $\Gamma = \begin{bmatrix} u_1 & 0 & 0 & 0 \\ 0 & u_2 & 0 & 0 \\ 0 & 0 & u_3 & 0 \\ 0 & 0 & 0 & u_4 \end{bmatrix}$; $u^* = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix}$;

$\Lambda = \begin{bmatrix} \gamma_1 & \gamma_2 & \gamma_3 & \gamma_4 \end{bmatrix}^T$ is the control effectiveness matrix with $0 \leq \gamma_i \leq 1$; $i = 1, 2, 3, 4$; $\gamma_i = 0$ and $\gamma_i = 1$ indicate that the ith actuator is fault-free and fully damaged, respectively.

Because $Bu(t) = B_1 u^*(t)$ and Equation (36) can be written as:

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) - B\Pi\Gamma\Lambda + N\xi(t, x(t)) + G \\ y(t) = Cx(t) \end{cases} \tag{37}$$

where $\Pi = \begin{bmatrix} -K_{th}L\sqrt{2}/2 & K_{th}L\sqrt{2}/2 & K_{th}L\sqrt{2}/2 & -K_{th}L\sqrt{2}/2 \\ K_{th}L\sqrt{2}/2 & -K_{th}L\sqrt{2}/2 & K_{th}L\sqrt{2}/2 & -K_{th}L\sqrt{2}/2 \\ K_d & K_d & -K_d & -K_d \\ K_{th} & K_{th} & K_{th} & K_{th} \end{bmatrix}$.

From Equations (7) and (37), we can obtain $E_a f_a = -B\Pi\Gamma\Lambda$. In the quadcopter model, we can design $E_a = B$. Therefore, LoCE of each actuator (fault estimation of each actuator) can be obtained as

$$\Lambda = -\Gamma^{-1}\Pi^{-1} f_a \tag{38}$$

It should be noted that this method not only estimates the fault magnitude of the roll, pitch, yaw, and thrust motion through Equation (14) but also presents the LoCE of each actuator through Equation (38). The results of these equations are discussed in Section 4.

4. Experimental Results and Discussion

4.1. Experiment Setup

The DJI450 quadcopter frame is used for flight tests in an outdoor environment. The intermediate observer algorithm in the previous section was implemented and tested on a Pixhawk2 using a C++ program executed on the Eclipse environment [34]. The firmware version 3.5.7 is used as an open-source software. The implementation strategy of the proposed fault estimation is presented in Table 1. In the first step, the control inputs from the rotation speeds of the actuator and system matrices in (5) should be obtained. Next, the positive parameter μ is chosen with small values at the beginning to reduce the overestimation problem. Then, the linear matrix inequality (LMI) matrix toolbox from Matlab software is used to find the observer matrix L and constant value of α. After that, the fault estimation of roll, pitch, yaw, and thrust motion in (14) is implemented in the Pixhawk2 flight controller using the result from step 2. If there are some magnitude order imbalance problems, the adjustment gain υ should be chosen in step 4. Finally, LoCE of each actuator in (38) is obtained through (14).

The experimental procedure of the flight test is summarized in Figure 2. First, the fault estimation based on the intermediate observer is implemented on the Pixhawk2 flight controller. Then, to demonstrate the fault scenario, a remote control is used to switch between position hold mode and fault modes, which allows us to apply faults by reducing the pulse width modulation (PWM) of actuators. The Mission Planner software was used to interface the ground station and quadcopter through Xbee telemetry communication [35], which can monitor all system states. The DJI F450 quadcopter parameter and the numerical values of the observer design are presented in Table 2.

Table 1. Fault estimation procedure.

Input: The control inputs and system matrices in (5) satisfying Assumptions 1–4.
Step 1: Choose small positive parameters of μ. **Step 2:** Solving the LMI matrix to obtain matrix L and α. **Step 3:** Fault estimation for roll, pitch, yaw, thrust motion in (14). **Step 4:** Choose adjustment gain υ for magnitude order balance. **Step 5:** Loss of control effectiveness of each actuator (Fault estimation of each actuator) in (38) is obtained.

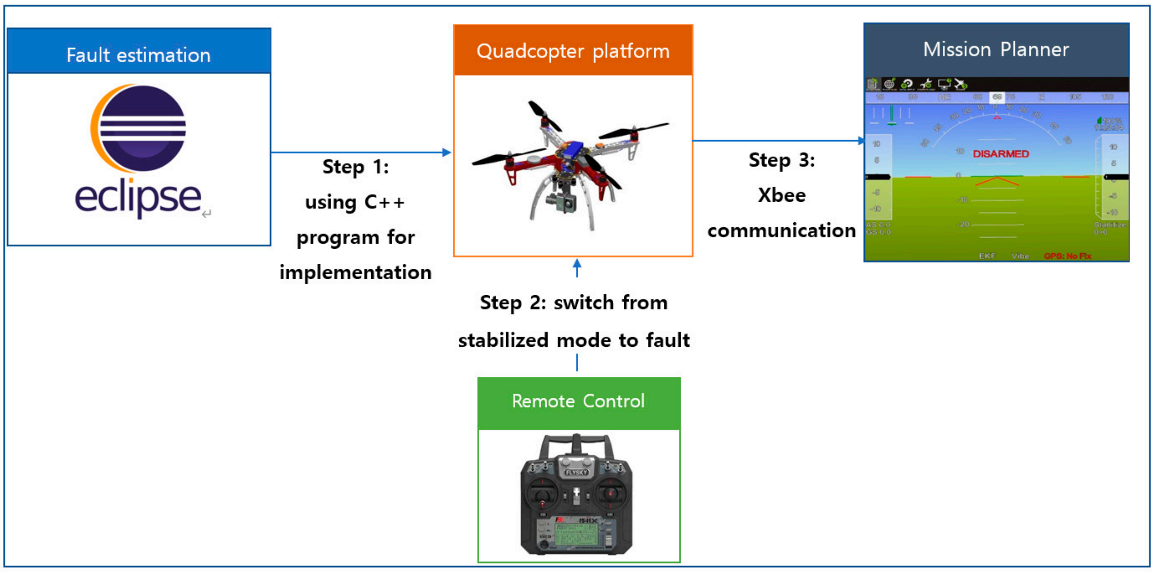
Fault estimation
eclipse
Step 1: using C++ program for implementation
Quadcopter platform
Step 3: Xbee communication
Mission Planner
DISARMED
Step 2: switch from stabilized mode to fault
Remote Control

Figure 2. Flight test procedure.

Table 2. DJI F450 quadcopter parameters and observer design values.

Parameter	**Description**	**Value**
L	Arm length	0.225 m
K_{th}	Thrust coefficient	0.0087
K_d	Drag coefficient	0.0055×10^{-2}
m	Total mass	1.776 kg
$I_x;\ I_y;\ I_z$	Moments of inertia	0.0035; 0.0035; 0.0055 kg.m^2
J_r	Rotor inertia	2.8×10^{-6} kg.m^2
υ	Adjustment gain	10
α	Constant	1
ε	Constant	0.02
η	Constant	1

The fault in each motor is modeled as the following function:

$$F_i = (1 - \gamma_i) K_{th} u_i \tag{39}$$

where $\gamma_i = 0$ indicates the fault-free condition and $\gamma_i = 1$ indicates the complete motor failure of the ith motor.

4.2. Experimental Results

4.2.1. LoCE in One Actuator without Using Magnitude Order Unbalance

The quadcopter system hovered at the height of 4 m. Faults with different magnitudes were injected into three actuators at time t = 68.5 s by 20% LoCE in the third motor. Figure 3 shows that the faults affect the horizontal movement. In detail, the x-direction deviates from the desired position by 60 cm between time t = 68.5 s and t = 76 s, whereas the y-direction deviates from the desired point by 30 cm. Subsequently, the system is recovered to the desired point due to the PID controller. In the vertical movement, the z-direction has a small deviation from the desired position because the fault magnitude injected in the third direction is small. Figure 4 describes the PWM inputs of all motors. It is shown that, before the fault is injected at t = 68.5 s, all PWM inputs have similar values. After faults occur in the third motor, the PID controller can recover the system by increasing the PWM input of the third actuator.

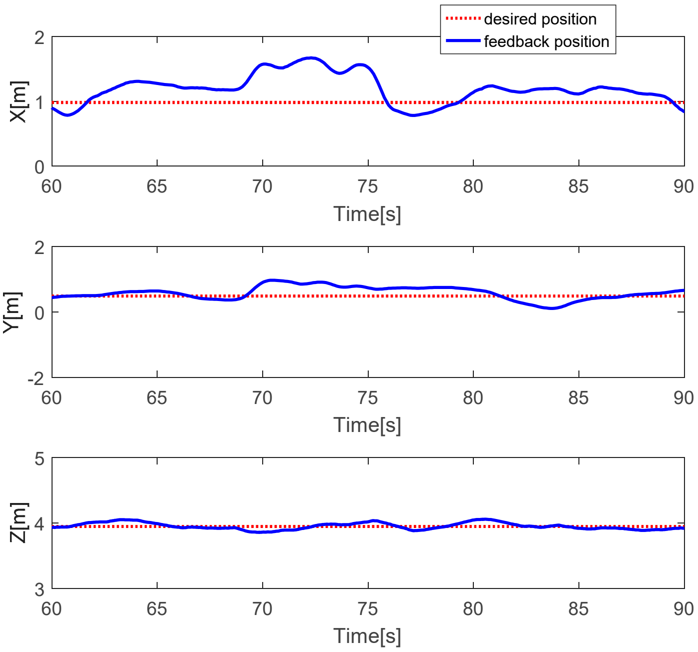

Figure 3. Horizontal and vertical movements of the quadcopter.

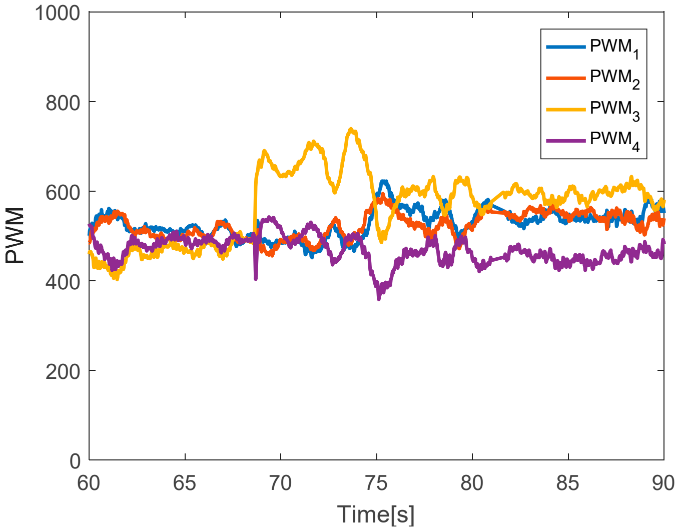

Figure 4. Pulse width modulation (PWM) inputs.

The fault estimations of the control inputs are presented in Figure 5. This shows that the fault estimation values converge to the desired values in the roll, pitch, and thrust control inputs, but the fault estimation value in the yaw control has a larger error due to the magnitude unbalance problem. Figure 6 shows the fault estimation of each actuator. The fault estimation values of the first, second, and fourth actuators converge to zero. The fault estimation value of the third actuator cannot converge to 0.2 owing to the magnitude unbalance issue.

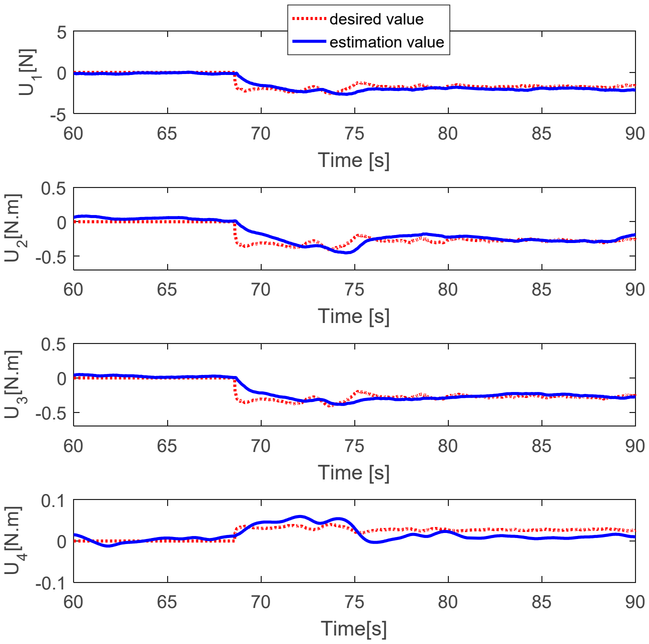

Figure 5. Fault estimations of control inputs.

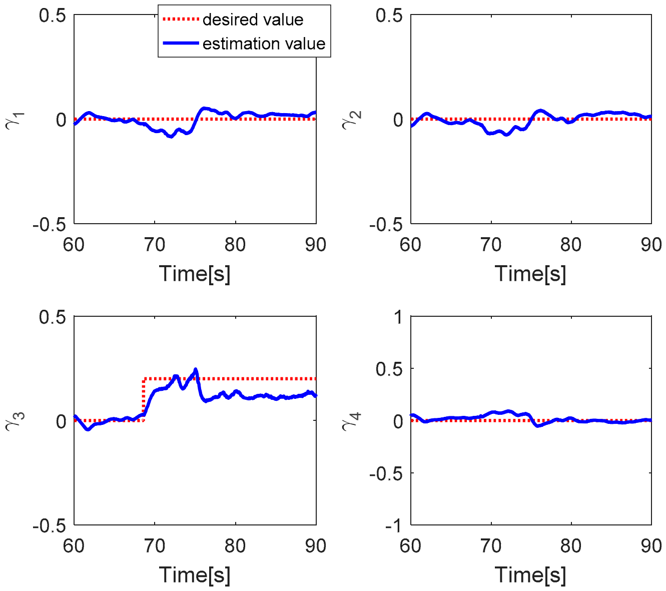

Figure 6. Fault estimation of each actuator.

Remark 2. *It is clear that the fault estimation in the yaw motion has a larger error compared with the other motion due to the magnitude order imbalance problem. To deal with this issue, the magnitude order balance method in Section 3.2 is applied to get a better performance, which is discussed in the next Sections 4.2.2–4.2.4.*

4.2.2. LoCE in One Actuator Using the Magnitude Order Balance Method

The quadcopter system hovers at a height of 2.4 m. The faults with different magnitudes are injected into three actuators at time t = 73.5 s by 20% LoCE in the third motor. In the Figure 7, the faults affect horizontal movement. In detail, the x-direction deviates from the desired position by 60 cm between t = 73.5 s and t = 77.5 s and the y-direction deviates from the desired point by 30 cm. After that, the system is recovered to the desired point due to the PID controller. In the vertical movement, the z-direction has a small deviation from the desired position because the fault magnitude injected in the third actuator is small. Figure 8 describes the PWM inputs of all motors. It was shown that before a fault is injected at t = 73.5 s, all PWM inputs have similar values. After the faults occur in the third motor, the PID controller can recover the system by increasing the PWM input of the third actuator.

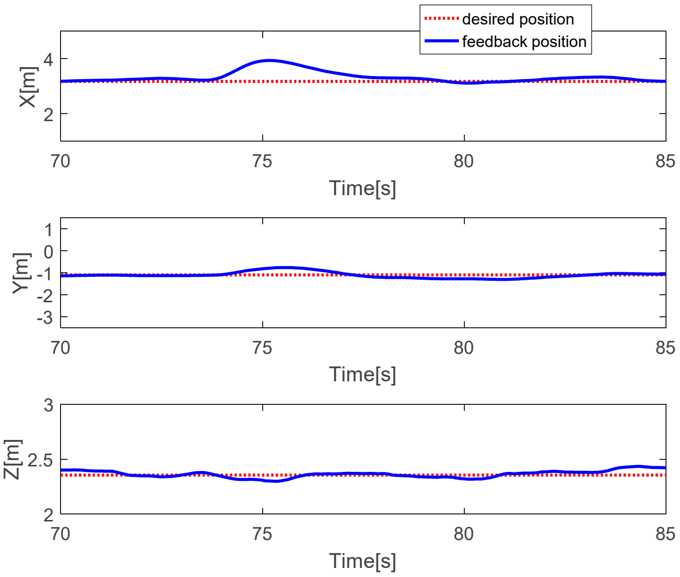

Figure 7. Horizontal and vertical movement of the quadcopter.

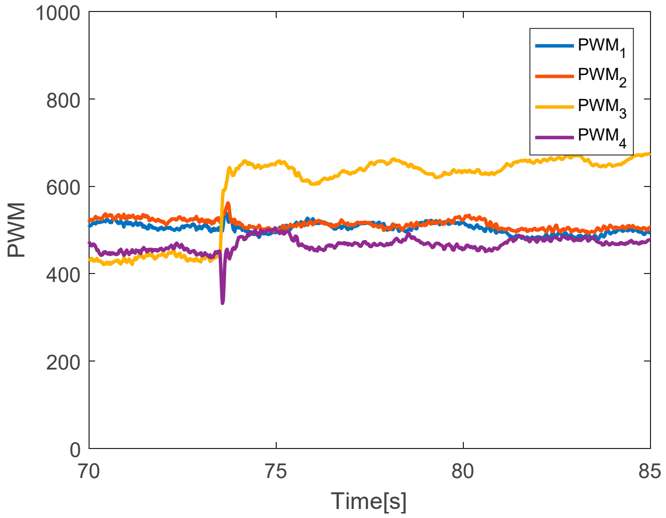

Figure 8. PWM inputs.

The fault estimations of the control inputs are presented in Figure 9. This shows that the fault estimation values converge to the desired values. Figure 10 shows the fault estimation of each actuator. The fault estimation values of the first, second, and fourth actuators converge smoothly to zero, while that of the third actuator converges to 0.2.

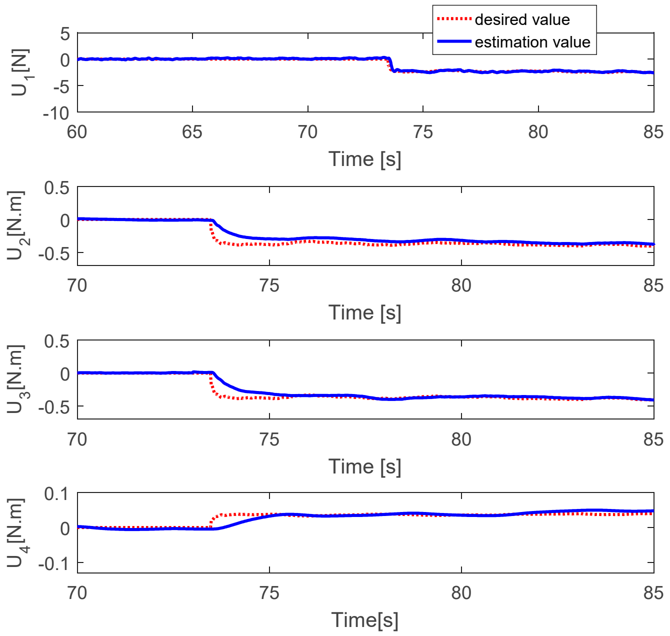

Figure 9. Fault estimation of control inputs.

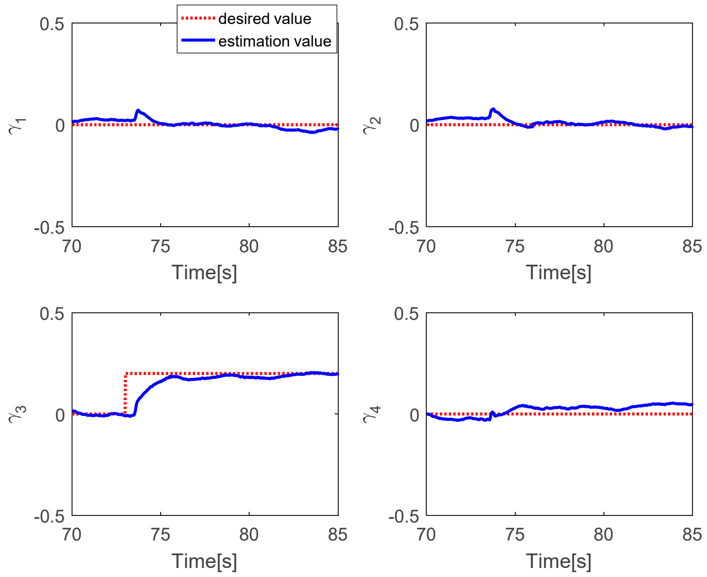

Figure 10. Fault estimation of each actuator.

4.2.3. LoCE in the Three Actuators Using Magnitude Order Unbalance

The quadcopter system hovers at an altitude of 2.4 m. Faults with different magnitudes are injected into the third actuator with 22% LoCE at $t = 73$ s, a fourth actuator with 16% LoCE at $t = 91$ s, and a second actuator with 13% LoCE at $t = 110$ s. In Figure 11, the faults affect the horizontal movement. From $t = 73$ s to $t = 80$ s, when the third actuator is faulty, the x-direction deviates from the desired position by 60 cm, while the y-direction deviates from the desired point by 25 cm. From $t = 90$ to 110 s, when the fourth actuator is faulty, the x-direction deviates from the desired position with 50 cm, while the y-direction experienced a small deviation from the desired point by 10 cm. From $t = 110$ to 1300 s, when the second actuator is faulty, the y-direction deviates from the desired position by 60 cm, whereas the x-direction deviates slightly from the desired point. Compared to the horizontal movement, the vertical movement has a very small deviation when faults occur in each actuator.

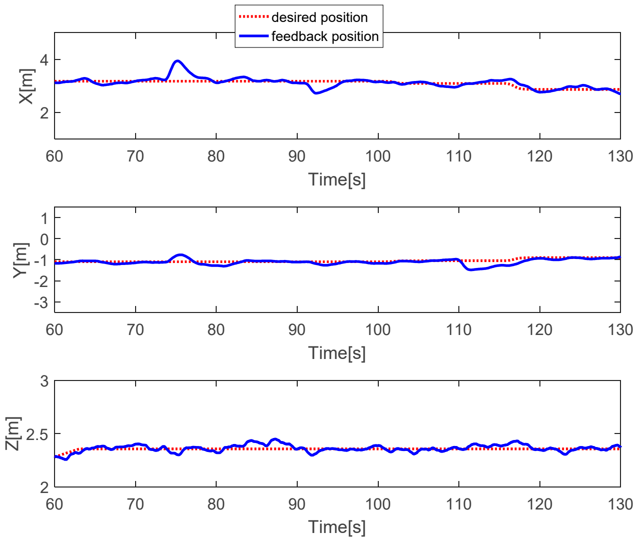

Figure 11. Horizontal and vertical movement.

Figure 12 shows the PWM inputs of all actuators. It is shown that before the fault is injected, all PWM inputs are almost the same. After faults occur in each actuator, the PID controller can recover the system by increasing the PWM input of the faulty actuator. Figure 13 reveals the fault estimations of the control inputs. This shows that the fault estimation values converge quickly to the desired values. Figure 14 describes the fault estimation of each actuator. The fault estimation values of the first, second, and fourth actuators converge to values of 0.22, 0.16, and 0.13, respectively, whereas the first actuator converges to zero.

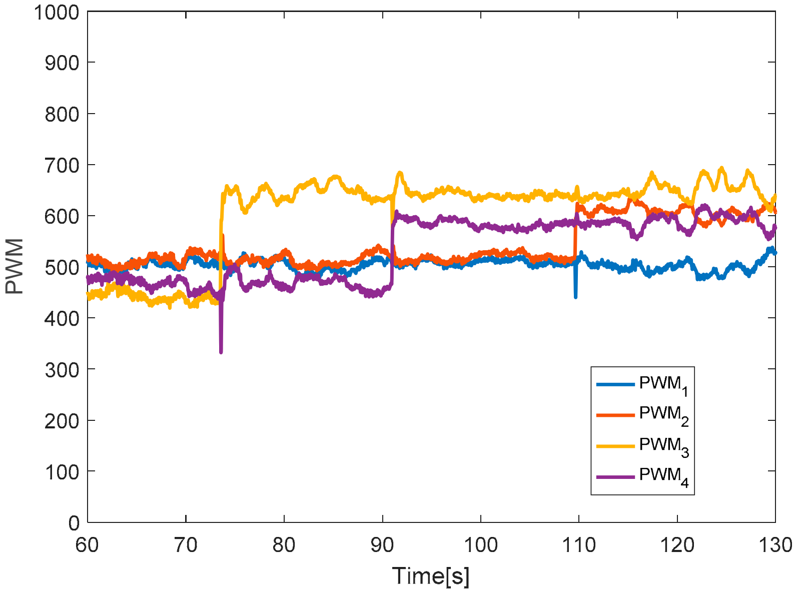

Figure 12. PWM inputs.

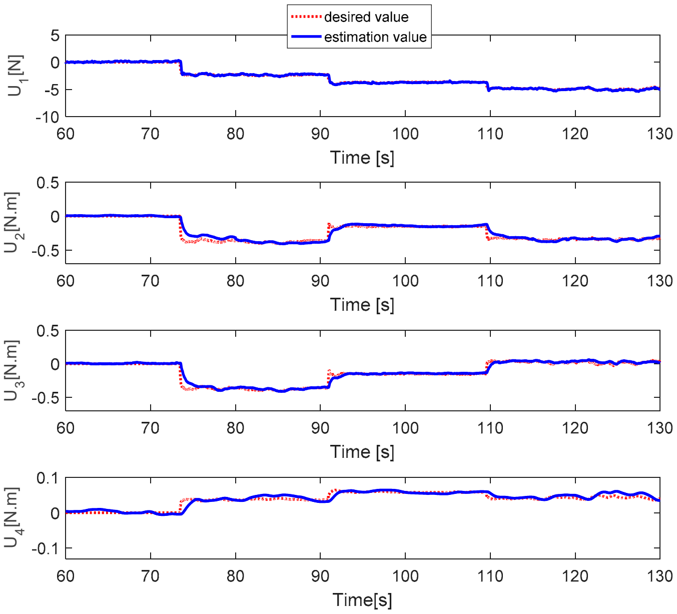

Figure 13. Fault estimation of control inputs.

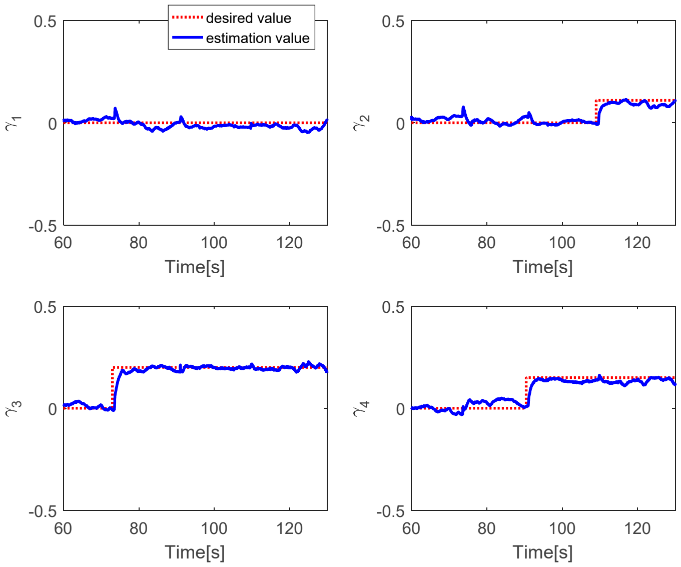

Figure 14. Fault estimation of each actuator.

4.2.4. Loss of Control Effectiveness in Four Motors Using Magnitude Order Unbalance

The quadcopter system hovers at the height of 3.8 m. The faults are injected into all actuators at time t = 50 s by 30% LoCE. As can be seen in Figure 15, the faults do not affect the horizontal movement, while the vertical movement drops to 0.4 m. The PWM inputs of all motors are shown in Figure 16. Before the faults are injected from time t = 0 to 50 s, all PWM inputs are almost the same. After faults occur in all motors, the PID controller can recover the system by increasing the PWM input of all motors.

Figure 17 presents the fault estimations of the control inputs. It shows that the fault estimation values converge quickly to the desired value. Figure 18 describes the fault estimation values of each actuator. It is shown that the fault estimation values of all motors converge to 0.3.

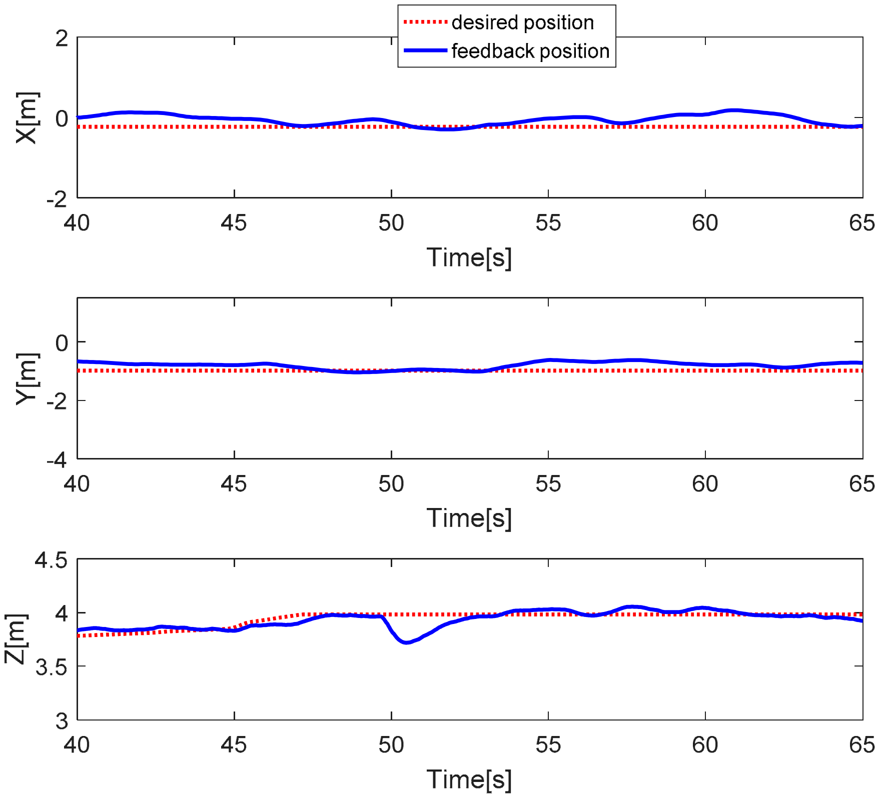

Figure 15. Horizontal and vertical movement.

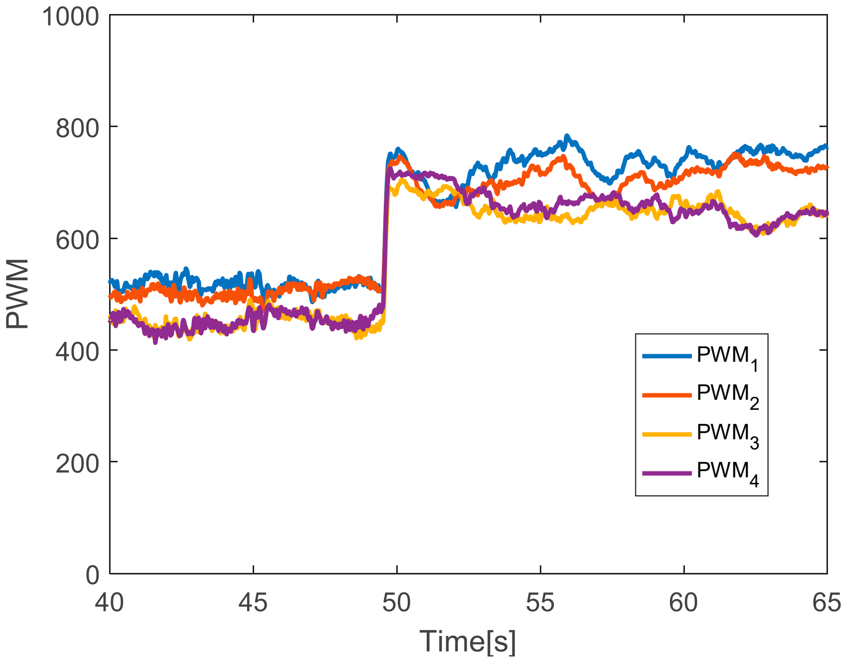

Figure 16. PWM inputs.

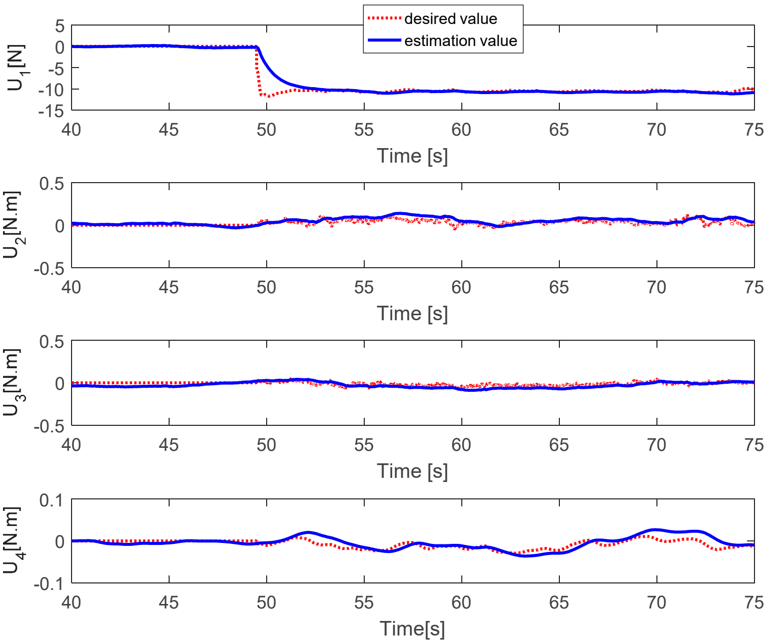

Figure 17. Fault estimation of control inputs.

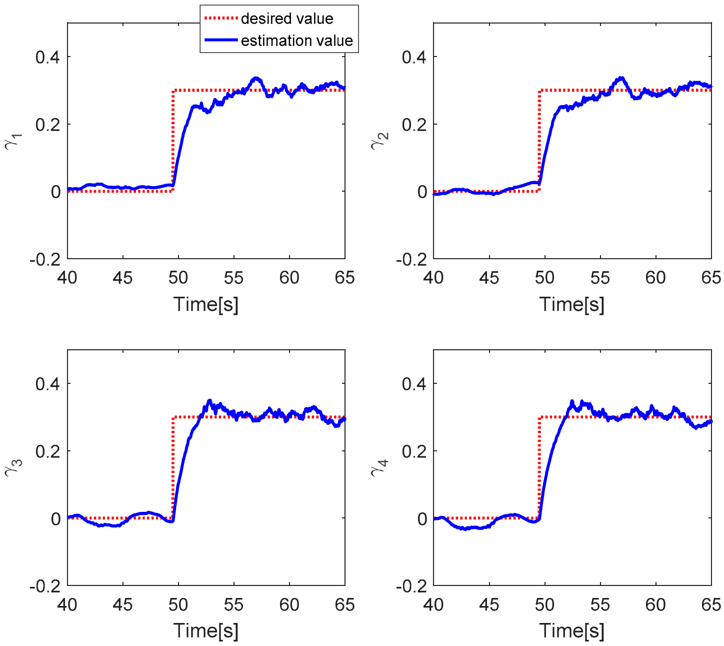

Figure 18. Fault estimation of each actuator.

Remark 3. *It should be mentioned that the previous observer-based fault diagnosis studies on quadcopter platform only consider the fault estimation of the roll, pitch, and yaw motion. In this article, the proposed method not only estimate the magnitude of the roll, pith, and yaw motion but also estimate the fault level of each actuator which is the differentiating point compared with other studies. The advantage of the presented fault estimation algorithm is the estimation of the actuator fault under model uncertainties.*

Remark 4. *The presented method was applied and tested in the quadcopter platform. For other multicopters (with more than four actuators), the matrix* Π *is not invertible. A pseudo-inverse matrix can be applied but it may cause inaccuracy in the fault estimation algorithm, which is a limitation of this study.*

5. Conclusions

In this study, an intermediate observer is investigated for a fault estimation scheme of a quadrotor under actuator faults. The improved fault estimation algorithm and magnitude order unbalance method were validated through a flight test on the DJI F450 quadcopter platform. Four experiments are presented. The first two scenarios present the effectiveness of the magnitude order balance method under a faulty third actuator. The remaining scenarios are shown to evaluate the reliability of the presented algorithm in the presence of multiple faults. The results reveal that the investigated method can accurately estimate the fault magnitude of the roll, pitch, yaw, and thrust motion. Different from other studies on observer-based fault estimation, this work can obtain the loss of control effectiveness of each actuator (fault estimation of each actuator) in the presence of uncertainties. However, the limitation of this article is that it does not provide a reconfiguration controller to give a completely active fault tolerant control system. Moreover, the drawback of this method is that it may not applied for multicopters with higher than four actuators due to the pseudo-inverse matrix mentioned in Remark 3. Future work should discuss and implement the reconfiguration controller for the quadcopter system using the proposed fault estimation algorithm. Furthermore, the difficulty with the pseudo-inverse matrix problem is also discussed for a general multicopter platform. The investigated algorithm needs the Assumptions 3–4, but some mathematical models may not meet these assumptions. Further studies will consider the relaxing technique for these assumptions.

Author Contributions: Conceptualization, N.P.N. and S.K.H.; Methodology, N.P.N.; Software, N.P.N.; Validation, N.X.M.; Formal Analysis, N.P.N.; Investigation, N.P.N.; Resources, N.P.N.; Data Curation, N.P.N.; Writing—Original Draft Preparation, N.P.N.; Writing—Review and Editing, X.P.D., T.T.H.; Visualization, N.P.N.; Supervision, S.K.H.; Project Administration, S.K.H.; Funding Acquisition, S.K.H. All authors have read and agreed to the published version of the manuscript.

Funding: This paper received no external funding.

Acknowledgments: This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2020-2018-0-01423) supervised by the IITP (Institute for Information & communications Technology Promotion).

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

References

1. Gordon, O.; Arinze, O.; James, O.; Nnaemeka, O. Design and Implementation of a Real Time Wireless Quadcopter for Rescue Operations. *Am. J. Eng. Res.* **2016**, *5*, 130–138.
2. Eid, S.E.; Dol, S.S. Design and development of lightweight-high endurance unmanned aerial vehicle for offshore search and rescue operation. In Proceedings of the 2019 Advances in Science and Engineering Technology International Conference, Dubai, UAE, 26 March–10 April 2019.
3. Ian, L.T.; Mitchell, D.H.; Christopher, D.D. UAVs for coastal surveying. *Coast. Eng.* **2016**, *114*, 19–24.
4. Kingston, D.; Rasmussen, S.; Humphrey, L. Automated UAV tasks for search and surveillance. In Proceedings of the 2016 IEEE Conference on Control Applications Part of 2016 IEEE Multi-Conference on Systems and Control, Buenos Aires, Argentina, 19–22 September 2016.
5. Tenmuulen, S.; Jonathan, D.; Jason, M.; Joel, B.S. UAV lidar and hyperspectral fusion for forest monitoring in the southwestern USA. *Remote Sens. Environ.* **2017**, *195*, 30–43.
6. Krause, S.; Sanders, T.G.; Mund, J.-P.; Greve, K. UAV-Based Photogrammetric Tree Height Measurement for Intensive Forest Monitoring. *Remote Sens.* **2019**, *11*, 758. [CrossRef]
7. Zhang, M.; Li, H.; Xia, G.; Zhao, W.; Rem, S.; Wang, C. Research on the Application of Deep Learning Target Detection of Engineering Vehicles in the Patrol and Inspection for Military Optical Cable Lines by UAV. In Proceedings of the 2018 11th International Symposium on Computational Intelligence and Design (ISCID), Hangzhou, China, 8–9 December 2018.
8. Butt, A.; Shah, S.I.A.; Zaheer, Q. Weapon Launch System Design of Anti-Terrorist UAV. In Proceedings of the 2019 International Conference on Engineering and Emerging Technologies, Lahore, Pakistan, 21–22 February 2019.
9. Brossard, J.; Bensoussan, D.; Landry, R.; Hammami, M. Robustness studies on quadrotor control. In Proceedings of the 2019 International Conference on Unmanned Aircraft Systems, Atlanta, GA, USA, 11–14 June 2019.

10. Mystkowski, A. An application of mu-synthesis for control of a small air vehicle and simulation results. *J. Vibroeng.* **2012**, *14*, 79–86.
11. Tunik, A.A.; Nadsadnaya, O.I. A Flight Control System for Small Unmanned Aerial Vehicle. *Int. Appl. Mech.* **2018**, *54*, 239–247. [CrossRef]
12. Xu, D.; Whidborne, J.F.; Cooke, A. Fault tolerant control of a quadrotor using L1 adaptive control. *Int. J. Intell. Unmanned Syst.* **2016**, *4*, 1–20.
13. Ghandour, J.; Aberkane, S.; Ponsart, J.C. Feedback linearization approach for standard and fault tolerant control: Application to a quadrotor UAV testbed. *J. Phys. Conf. Ser.* **2014**, *570*, 082003. [CrossRef]
14. Li, T.; Zhang, Y.; Gordon, B.W. Nonlinear fault-tolerant control of a quadrotor UAV based on sliding mode control technique. In Proceedings of the 8th IFAC Symposium of Fault Detection, Supervision and Safety of Technical Processes, Mexico city, Mexico, 29–31 August 2012.
15. Freddi, A.; Longhi, S.; Monteriù, A. A diagnostic thau observer for a class of unmanned vehicles. *J. Intell. Robot. Syst.* **2012**, *67*, 1–13. [CrossRef]
16. Freddi, A.; Longhi, S.; Monteriù, A. A model-based fault diagnosis system for a mini-quadrotor. In Proceedings of the 7th Workshop on Advanced Control and Diagnosis, Zielona Gora, Poland, 19–20 November 2009.
17. Freddi, A.; Longhi, S.; Monteriu, A. Actuator fault detection system for a mini-quadrotor. In Proceedings of the 2010 IEEE International Symposium on Industrial Electronics, Bari, Italy, 4–7 July 2020.
18. Aguilar-Sierra, H.; Flores, G.; Salazar, S.; Lozano, R. Fault estimation for a quad-rotor MAV using a polynominal observer. In Proceedings of the 2013 International Conference on Unmanned Aircraft Systems, Atlanta, GA, USA, 28–31 May 2013.
19. Amoozgar, M.H.; Chamseddine, A.; Zhang, Y. Experimental test of a two-stage Kalman filter for actuator fault detection and diagnosis of an unmanned quadrotor helicopter. *J. Intell. Robot. Syst.* **2013**, *70*, 107–117. [CrossRef]
20. Liu, Z.; Yuan, C.; Zhang, Y. Active fault-tolerant control of unmanned quadrotor helicopter using linear parameter varying technique. *J. Intell. Robot. Syst.* **2017**, *88*, 415–436. [CrossRef]
21. Cen, Z.; Noura, H.; Susilo Younes, Y.A. Robust fault diagnosis for quadrotor UAVs using adaptive Thau observer. *J. Intell. Robot. Syst.* **2014**, *73*, 573–588. [CrossRef]
22. Cen, Z.; Noura, H. An Adaptive Thau Observer for estimating the time-varying LOE fault of quadrotor actuators. In Proceedings of the 2013 Conference on Control and Fault-Tolerant Systems (SysTol), Nice, France, 9–11 October 2013.
23. Ma, H.; Liu, Y.; Li, T.; Yang, G. Nonlinear High-Gain Observer-Based Diagnosis and Compensation for Actuator and Sensor Faults in a Quadrotor Unmanned Aerial Vehicle. *IEEE Trans. Ind. Inform.* **2019**, *15*, 550–562. [CrossRef]
24. Witczak, P.; Patan, K.; Witcak, M.; Mrugalski, M. A neural network approach to simultaneous state and actuator fault estimation under unknown input decoupling. *Neurocomputing* **2017**, *250*, 65–75. [CrossRef]
25. Cho, C.N.; Hong, J.T.; Kim, H.J. Neural Network Based Adaptive Actuator Fault Detection Algorithm for Robot Manipulators. *J. Intell. Robot. Syst.* **2019**, *95*, 137–147. [CrossRef]
26. Wang, Z.; Shen, Y.; Zhang, X. Actuator fault estimation for a class of nonlinear descriptor systems. *Int. J. Syst. Sci.* **2014**, *45*, 487–496. [CrossRef]
27. Mallavalli, S.; Fekih, A. A fault tolerant tracking control for a quadrotor UAV subject to simultaneous actuator faults and exogenous disturbances. *Int. J. Control* **2018**, *93*, 655–668. [CrossRef]
28. Chen, F.; Lei, W.; Tao, G.; Jiang, B. Actuator fault estimation and reconfiguration control for quad-rotor helicopter. *Int. J. Adv. Robot. Syst.* **2016**, *13*, 33. [CrossRef]
29. Bharani, K.P.; Alwi, H.; Edwards, C. Fault reconstruction for a quadrotor using an LPV sliding mode observer. In Proceedings of the 9th IFAC Symposium on Fault Detection Supervision and Safety for Technical Processes SAFEPROCESS 2015, Paris, France, 2–4 September 2015.
30. Nguyen, N.P.; Hong, S.K. Sliding Mode Thau Observer for Actuator Fault Diagnosis of Quadcopter UAVs. *Appl. Sci.* **2018**, *8*, 1893. [CrossRef]
31. Dong, W.; Gu, G.-Y.; Zhu, X.; Ding, H. Modeling and Control of a Quadrotor UAV with Aerodynamics Concepts. *World Acad. Sci. Eng. Technol.* **2013**, *7*, 901–906.
32. Zhang, Y.; Chamseddine, A. Fault tolerant flight control techniques with application to a quadrotor UAV testbed. In *Automatic Flight Control Systems-Latest Developments*; Thomas Lombaerts, IntechOpen: London, UK, 2012. [CrossRef]

33. Zhu, J.-W.; Yang, G.-H.; Wang, H.; Wang, F. Fault estimation for a class of nonlinear systems based on intermediate estimator. *IEEE Trans. Autom. Control* **2016**, *61*, 2518–2524. [CrossRef]
34. Editing/Building with Eclipse on Windows. Available online: http://ardupilot.org/dev/docs/editing-the-code-with-eclipse.html (accessed on 31 August 2018).
35. Telemetry. Available online: http://ardupilot.org/copter/docs/common-telemetry-landingpage.html (accessed on 31 August 2018).