Fault Tolerant Control for Quad-rotor UAV by Employing Lyapunov-based Adaptive Control Approach

Xiaobing Zhang¹, Youmin Zhang²
Department of Mechanical and Industrial Engineering
Concordia University, Montreal, Canada, H3G 1M8

Abstract—In this paper, an Lyapunov-based adaptive control strategy has been proposed for fault tolerant control of a quad-rotor UAV. A nonlinear six-degree of freedom mathematic model of the quad-rotor UAV is derived first by applying Newton-Euler formalism, and the adaptive laws and control laws based on Lyapunov-based adaptive technique are then developed and applied for quad-rotor UAV in the presence of actuator partial loss of effectiveness faults. Simulation results show the effectiveness of developed adaptive control strategy. Performance comparisons of the quadrotor UAV for different levels of actuator faults under the conditions with/without system parameter uncertainties have been carried out by utilizing the Lyapunov-based adaptive control approach.

Nomenclature

\[ x, y, z \] = position information
\[ h \] = height control input
\[ u_\alpha, u_\beta, u_\gamma \] = roll, pitch, yaw control input respectively
\[ m \] = mass of quad-rotor
\[ \dot{\phi}, \dot{\theta}, \dot{\psi} \] = roll angular acceleration, pitch angular acceleration, yaw angular acceleration
\[ \phi, \theta, \psi \] = roll angle, pitch angle and yaw angle
\[ c_{df} \] = drag coefficients
\[ l \] = distance of motor from pivot centre
\[ J_x, J_y, J_z \] = Initial moment of \( x \)
\[ g \] = gravity acceleration

I. Introduction

Fault Tolerant Control Systems (FTCS) are known as the control systems that possess the ability to accommodate system component faults/failures automatically and to maintain overall system stability and acceptable performance in the event of such failures [1]. FTCS can be divided into Passive FTCS (PFTCS) and Active FTCS (AFTCS). The
differences between PFTCS and AFTCS are: PFTCS can tolerate the anticipated faults while retaining a fixed controller; meanwhile, AFTCS can tolerate both the anticipated and unanticipated faults and make real-time detection, decision-making and control reconfiguration [1, 2].

As an abnormal behavior of the system, faults can breakdown the entire system because they are hard to be observed. To resolve this problem, researchers have developed many schemes and tried to reduce or even eliminate the effects that faults bring. [3] presents a LQG (Linear Quadratic Gaussian) approach for linear system with sensor failures and compares the differences between the standard LQG and reliable LQG design. [4] utilizes LMI (Linear Matrix Inequality) technique to design reliable robust tracking controller in order to withstand actuator faults and control surface impairments in aircraft. [5] shows a unified discrete/continuous approach for observer-based controller cases. Recently, fault tolerant control based on adaptive control theory has gained significant attention. Developed methods include simultaneous adaptation and robust feedback [6], constrained adaptive backstepping against structural damage or control surface failures [7, 13], inverse optimal and tuning function adaptive backstepping technique [8], and adaptive sliding mode control [9]. Throughout the existing literatures, several adaptive backstepping controller parameters tuning strategies have been proposed, including \( \mathcal{F} \) modification approach [11, 12], B-spline neural network [13], and adaptive PID control [14].

Our previous work [10] employs the basic backstepping scheme with fixed controller parameters to design the control system against the preselected actuator failures of a quad-rotor UAV, satisfactory results have been obtained under different faults and system uncertainties. However, fixed controller gain cannot ensure acceptable performance when severe faults occur, which has been shown in the simulation part. This paper considers the Lyapunov-based adaptive control method to handle various faults.

This paper contributed to design a Lyapunov-based adaptive controller in the framework of FTCS for handling faults in quad-rotor UAV actuators. Simulations are conducted among cases under normal condition, with system parameter uncertainties, partial loss faults and combined fault conditions in different levels. Moreover, the simulation results have been analyzed and compared under each fault condition.

Organization of the paper is as follows: Section II introduces the principle of quad-rotor operation and derives the equations of mathematic model for quad-rotor UAV. Section III presents the controller design procedures. Section IV shows the simulation results demonstrating the performance of control laws under different situations. The last section V concludes the work done in this paper.

II. Quad-rotor UAV Model

Quad-rotor is an under-actuated system because it has six degrees of freedom while it has only four inputs. The collective input (or throttle input) is the sum of the thrusts of each motor. The four rotors have been divided into front and back rotors (3 & 1) and left and right rotors (2 & 4). The front and back rotors rotate in counter-clockwise direction while the other two in clockwise direction. All of the movements can be controlled by the changes of each rotor speed. Vertical flight is achieved by increasing all of rotors’ speed to move up or decreasing the speed to go down. Roll motion can be controlled by decreasing (increasing) the left rotor speed while increasing (decreasing) the right rotor speed to make the quad-rotor roll left (right). Pitch motion can be controlled by decreasing (increasing) the front rotor speed while increasing (decreasing) the rear rotor speed to make the quad-rotor up (down). Yaw moment is a little different, which depends on all rotors’ speed. When front and rear pair spins slower (faster) than left and right pair, the quad-rotor will move in positive (negative) direction (counter-clockwise/clockwise direction).

Figure. 1 shows the structure of a quad-rotor UAV. The inertial frame \( E = \{x_E, y_E, z_E\} \) is fixed with the earth, \( B = \{x_B, y_B, z_B\} \) represents the body frame fixed with quad-rotor body, \( P = \{x, y, z\} \) is the position of the quad-rotor mass centre expressed in the inertial frame. \( F_1, F_2, F_3, \) and \( F_4 \) is thrust of each propeller respectively, and
\( m \) is the mass of the quadrotor. Meanwhile, the Euler angles are roll angle \((\phi \in (-\frac{\pi}{2}, \frac{\pi}{2}))\), pitch angle \((\theta \in (-\frac{\pi}{2}, \frac{\pi}{2}))\) and yaw angle \((\psi \in (-\frac{\pi}{2}, \frac{\pi}{2}))\), respectively. The rotation matrices from body frame to earth frame can be obtained as:

\[
L_{b,e} = \begin{bmatrix}
\cos \psi \cos \theta & \sin \phi \sin \theta \cos \psi - \cos \phi \sin \psi & \cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi \\
\cos \theta \sin \psi & \sin \phi \sin \theta \sin \psi + \cos \phi \cos \psi & \cos \phi \sin \theta \sin \psi - \sin \phi \cos \psi \\
-\sin \theta & \sin \phi \cos \theta & \cos \phi \cos \theta
\end{bmatrix}
\]  

(1)

According to Newton-Euler equation [15]:

\[
\begin{align*}
    m \ddot{v}_b + \omega_b \times m v_b &= F_b \\
    J \ddot{\omega}_b + \omega_b \times J \omega_b &= \mathbf{T}_b
\end{align*}
\]  

(2)

where \(\omega_b = \begin{bmatrix} P \\ Q \\ R \end{bmatrix}\) is quad-rotor’s angular velocity, \(v_b\) is body velocity, \(\mathbf{T}_b\) denotes the quad-rotor’s body moment,

\[
J = \begin{bmatrix}
J_x & 0 & 0 \\
0 & J_y & 0 \\
0 & 0 & J_z
\end{bmatrix}
\]

is inertial matrix of the rigid body.

The actuators generate the thrust of each rotor by [16]:

\[
F_i = \frac{P}{4} \omega_i^2 R^3 abc (\theta_i - \varphi_i)
\]  

(3)

where \(p\) is the air density, \(\omega_i\) is the rotor speed, \(R\) is the rotor radius, \(a\) is airfoil lift curve scope, \(b\) is the blade number of a rotor, \(c\) is the lift coefficient, \(\theta_i\) is the pitch angle at the blade tip, and \(\varphi_i\) is the inflow angle at the tip. Since the quadrotor UAV has fixed the pitch rotors, \(\theta_i\) can be seen as a constant. To simplify the system, \(\varphi_i\) has been ignored by setting it to zero when the airflow direction changes are resulted due to the quadrotor motions through the air. Since all of the parameters are constant, except the rotor speed \(\omega_i\), the actuator meant to generate the thrust can be simplified as:

\[
F_i = b_i \omega_i^2
\]  

(4)
where \( b_i \) is defined as the coefficient of each actuator. When faults happen, \( b_i \) in Eq. (4) has a reduced value by decreasing certain percentage corresponding to certain level of actuator partial fault or loss of its effectiveness. Finally, these partial losses have an effect on the entire quadrotor UAV system since the dynamic change induced by the fault.

**Dynamic Equations:**

\[
\begin{align*}
\dot{u} &= \frac{1}{m} (mg \sin \theta - k_{d1} \ddot{x}) + \ddot{\psi} v - \dot{\theta} w \\
\dot{v} &= \frac{1}{m} (-mg \cos \theta - k_{d2} \ddot{y}) + \ddot{\omega} - \ddot{\psi} u \\
\dot{w} &= \frac{1}{m} [(F_1 + F_2 + F_3 + F_4) - mg \cos \phi \cos \theta - k_{d3} \ddot{z}] + \dot{\theta} u - \dot{\phi} v
\end{align*}
\]

(Eq. 5)

**Navigation Equations:**

\[
\begin{align*}
\ddot{x} &= \frac{F_1 + F_2 + F_3 + F_4}{m} (\cos \psi \sin \theta \cos \phi + \sin \psi \sin \phi) - \frac{k_{d1} \ddot{x}}{m} \\
\ddot{y} &= \frac{F_1 + F_2 + F_3 + F_4}{m} (\sin \psi \sin \theta \cos \phi - \cos \psi \sin \phi) - \frac{k_{d2} \ddot{y}}{m} \\
\ddot{z} &= \frac{F_1 + F_2 + F_3 + F_4}{m} \cos \theta \cos \phi - \frac{k_{d3} \ddot{z}}{m} - g
\end{align*}
\]

(Eq. 6)

**Moment Equations:**

\[
\begin{align*}
\ddot{\phi} &= \frac{1}{J_x} [(F_2 - F_4)l - k_{d4} \dot{\phi} - \dot{\theta} \psi (J_z - J_y)] \\
\ddot{\theta} &= \frac{1}{J_y} [(F_1 - F_3)l - k_{d5} \dot{\theta} - \dot{\phi} \psi (J_z - J_x)] \\
\ddot{\psi} &= \frac{1}{J_z} [(F_1 - F_2 + F_3 - F_4)l - k_{d6} \ddot{\psi} - \dot{\phi} \dot{\theta} (J_y - J_x)]
\end{align*}
\]

(Eq. 7)

Therein, \( F_3 \) and \( F_1 \) are the thrust of the forward and rear rotors, \( F_4 \) and \( F_2 \) are the thrust of right and left rotors; \( k_{d4} \dot{\phi} , k_{d5} \dot{\theta} \) and \( k_{d6} \ddot{\psi} \) denote the moments caused by drag along body axis \( x_B \), \( y_B \), and \( z_B \) respectively, \( k_{d4} \), \( k_{d5} \), and \( k_{d6} \) are coefficients, and \( l \) represents distance from the rotor to the center of gravity (cg).

The real control inputs are angular velocity of each rotor. However, for the simplicity of the controller design, the virtual control inputs will be employed for the following controller designs, which are defined as:

\[
\begin{pmatrix}
  u_1 \\
  u_2 \\
  u_3 \\
  u_4
\end{pmatrix} =
\begin{pmatrix}
  F_1 + F_2 + F_3 + F_4 \\
  F_2 - F_4 \\
  F_1 - F_3 \\
  F_1 + F_3 - F_2 - F_4
\end{pmatrix} =
\begin{pmatrix}
  b_1 & b_2 & b_3 & b_4
\end{pmatrix}
\begin{pmatrix}
  \omega_1^2 \\
  \omega_2^2 \\
  \omega_3^2 \\
  \omega_4^2
\end{pmatrix}
\]

(Eq. 8)
III. Lyapunov-based Adaptive Control Design Considering Actuator Faults

The system controlled can be rewritten as a state space form:

\[ \dot{X} = \Psi + \Phi U_c \]

where \( X \) is the state space vector of the system and can be defined as:

\[ X = (z \quad \phi \quad \theta \quad \psi)^T \]

with the virtual control inputs

\[ U_c = \begin{bmatrix} u_{c1} \\ u_{c2} \\ u_{c3} \\ u_{c4} \end{bmatrix} \]

with

\[
\Psi = \begin{bmatrix}
-\dot{\psi} \\
-\dot{\theta} \\
-\dot{\phi}
\end{bmatrix}
\]

\[
\Phi = \begin{bmatrix}
\cos \theta \cos \phi & 0 & 0 & 0 \\
0 & \frac{l}{J_x} & 0 & 0 \\
0 & 0 & \frac{l}{J_y} & 0 \\
0 & 0 & 0 & \frac{l}{J_z}
\end{bmatrix}
\]

Considering the actuator partial loss \( U_c = B^*U \) shown in Figure 2, the controlled system can be represented as:

\[
\dot{X} = \begin{bmatrix}
-\dot{\psi} \\
-\dot{\theta} \\
-\dot{\phi}
\end{bmatrix}
+ \begin{bmatrix}
b_1 \cos \theta \cos \phi & 0 & 0 & 0 \\
0 & b_2 \frac{l}{J_x} & 0 & 0 \\
0 & 0 & b_3 \frac{l}{J_y} & 0 \\
0 & 0 & 0 & b_4 \frac{l}{J_z}
\end{bmatrix}\begin{bmatrix}
\dot{z} \\
\dot{\phi} \\
\dot{\theta} \\
\dot{\psi}
\end{bmatrix}
\]

(10)

Figure 2 Control diagram by utilizing the Lyapunov-based adaptive approach
where control effectiveness matrix $B^*$ is defined as

$$B^* = \begin{pmatrix} b_1^* & 0 & 0 & 0 \\ 0 & b_2^* & 0 & 0 \\ 0 & 0 & b_3^* & 0 \\ 0 & 0 & 0 & b_4^* \end{pmatrix}$$

with $b_i^* \in (0,1]$.

**Remark:** The value of $b_i^*$ ($i = 1, 2, 3, 4$) can reflect effects caused by the actuator partial loss. When the actuators work normally $b_i^* = 1$ ($i = 1, 2, 3, 4$), this situation has been addressed in Chapter 3. When the actuators work under partial loss, $0 < b_i^* < 1$ ($i = 1, 2, 3, 4$), the system performance will be affected due to the change for $b_i^*$ ($i = 1, 2, 3, 4$). In this chapter, considering the actual conditions, the values of $b_i^*$ ($i = 1, 2, 3, 4$) will be estimated by using adaptive control method, and the corresponding control approach will still ensure the robustness of the system.

In similarity with the Lyapunov-based control design, the following design procedure will utilize the altitude tracking control as an example [20].

The motion equation of the altitude is

$$\ddot{z} = \frac{b_1^*}{m} u_1 \cos \theta \cos \phi - g$$

The control objective is to design a control law for $u_1$ to force the system in $z$ to follow a specified desired trajectory $z_r$ i.e., $z \to z_r$ as $t \to \infty$. The desired trajectory satisfies the following assumption:

**Assumption 1:** The desired trajectory $z_r = [z_r, \dot{z}_r, \ddot{z}_r]^T$ is continuous and available, and $[z_r, \dot{z}_r, \ddot{z}_r]^T \in \Omega_d \subset \mathbb{R}^3$ with $\Omega_d$ is a compact set.

In presenting the developed adaptive control law, the following definitions are required:

$$\dot{\alpha}_i = \dot{\alpha}_i - \alpha_i$$

where $\dot{\alpha}_i$ is an estimate of $\alpha_i$, which is defined as

$$\Delta_i = \frac{1}{b_i^*}$$

Given the plant, the following control law is proposed

$$u_1 = \dot{\alpha}_i u_{e1}$$

with

$$u_{e1} = \frac{m}{\cos \theta \cos \phi} (-c_{12} y_{12} - y_{11} + g + \ddot{z}_r + \dot{\beta}_1)$$

where

$$y_{11}(t) = z(t) - z_r(t)$$

$$y_{12}(t) = \dot{z}(t) - \dot{z}_r(t) - \dot{\beta}_1$$

$$\dot{\beta}_1(t) = -c_{11} y_{11}(t)$$

and the parameter $\dot{\alpha}_i$ will be updated by the following adaptation law:
\[
\dot{\alpha}_i = -\gamma \frac{u_{i1}}{m} \cos \theta \cos \phi y_{i2}
\]  \hspace{1cm} (17)

where parameters \(c_{11}, c_{12}\) and \(\gamma\) are positive constants.

For the plant given in Eq. (11), subject to Assumption 1, the adaptive controller specified by (16) and (17) ensures that the signal \(z(t)\) is bounded and \(z(t) \to z_r(t)\) as \(t \to \infty\).

**Proof:** By utilizing the backstepping design procedure, the following Lyapunov candidate can be selected as

\[
V(t) = \frac{1}{2} y_{11}^2 + \frac{1}{2} y_{12}^2 + \frac{b_i^*}{2\gamma} \tilde{\alpha}_i^2
\]  \hspace{1cm} (18)

The derivative \(\dot{V}\) is given by

\[
\dot{V} = y_{11} \dot{y}_{11} + y_{12} \dot{y}_{12} + \frac{b_i^*}{\gamma} \tilde{\alpha}_i \dot{\tilde{\alpha}}_i
\]

\[
\leq y_{11} \dot{y}_{11} + y_{12} (\ddot{z}(t) - \ddot{z}_r(t) - \dot{\beta}_1) + \frac{b_i^*}{\gamma} \tilde{\alpha}_i \dot{\tilde{\alpha}}_i
\]  \hspace{1cm} (19)

\[
\leq y_{11} \dot{y}_{11} + y_{12} \left( \frac{b_i^*}{m} \cos \theta \cos \phi - g - \ddot{z}_r(t) - \dot{\beta}_1 \right) + \frac{b_i^*}{\gamma} \tilde{\alpha}_i \dot{\tilde{\alpha}}_i
\]

According to the definition of \(\tilde{\alpha}_i\) in Eq. (12), we have

\[
\tilde{\alpha}_i u_{i1} = (\tilde{\alpha}_i u_{i1} + \alpha_i u_{i1})
\]  \hspace{1cm} (20)

then the inequality (Eq. (19)) can be deduced by Eq. (20) as

\[
\dot{V} \leq y_{11} \dot{y}_{11} + y_{12} \left( \frac{b_i^*}{m} \tilde{\alpha}_i u_{i1} + \alpha_i u_{i1} \right) \cos \theta \cos \phi - g - \ddot{z}_r(t) - \dot{\beta}_1 + \frac{b_i^*}{\gamma} \tilde{\alpha}_i \dot{\tilde{\alpha}}_i
\]

\[
\leq y_{11} \dot{y}_{11} + y_{12} \left( \frac{b_i^*}{m} \alpha_i u_{i1} \right) \cos \theta \cos \phi - g - \ddot{z}_r(t) - \dot{\beta}_1 + \frac{b_i^*}{\gamma} \tilde{\alpha}_i \dot{\tilde{\alpha}}_i
\]  \hspace{1cm} (21)

By using the control law Eq. (15) and the adaptation law (17), we have

\[
\dot{V} \leq y_{11} \dot{y}_{11} + y_{12} \left( \frac{b_i^*}{m} \cos \theta \cos \phi \tilde{\alpha}_i u_{i1} + \cos \theta \cos \phi \alpha_i u_{i1} \right)
\]

\[- g - \ddot{z}_r(t) - \dot{\beta}_1 + \frac{b_i^*}{\gamma} \tilde{\alpha}_i \dot{\tilde{\alpha}}_i \leq -c_{11} y_{11}^2 - c_{22} y_{12}^2\]  \hspace{1cm} (22)

Eq. (18) and Eq. (22) imply that \(V\) is nonincreasing. Hence, \(y_{i1}\) (\(i = 1, 2\)) are bounded. By applying the Lasalle-Yoshizawa theorem, it further follows that \(y_{i1}\) (\(i = 1, 2\)) \(\to 0\) as \(t \to \infty\), which implies that \(\lim_{t \to \infty} [z(t) - z_r(t)] = 0\).

Similarly, the control laws and adaptation laws for the roll angle, pitch angle and yaw angle can be obtained separately.

For roll angle, system plant is:

\[
\ddot{\phi} = \frac{1}{J_x} \left[ b_{z1}^* u_{z1} - \hat{\dot{\psi}} (J_z - J_y) \right]
\]  \hspace{1cm} (23)

the control law and adaptive law are designed as
\[ u_2 = \hat{\alpha}_2 u_{e2} \]  
(24)

with

\[ u_{e2} = \frac{J}{l} \left[ -c_{22} y_{22} - y_{21} + \dot{\beta}_2 + \dot{\phi}_r + \frac{1}{J_x} \dot{\theta}\psi(J_z - J_y) \right] \]  
(25)

where

\[ y_{21}(t) = \phi(t) - \phi_r(t) \]
\[ y_{22}(t) = \dot{\phi}(t) - \dot{\phi}_r(t) - \beta_2 \]
\[ \beta_2(t) = -c_{21} y_{21}(t) \]  
(26)

and the adaptive law for roll angle is:

\[ \dot{\hat{\alpha}}_2 = -\gamma_u \frac{u_{e2}}{J_x} y_{22} \]  
(27)

For pitch angle, the equation is expressed as below:

\[ \ddot{\theta} = \frac{1}{J_x} \left[ b_l\phi u_y - \dot{\phi}\psi(J_x - J_z) \right] \]  
(28)

the control law and adaptive law are designed as

\[ u_3 = \hat{\alpha}_3 u_{e3} \]  
(29)

with

\[ u_{e3} = \frac{J}{l} \left[ -c_{32} y_{32} - y_{31} + \dot{\beta}_3 + \dot{\theta}_r + \frac{1}{J_y} \dot{\phi}\psi(J_x - J_z) \right] \]  
(30)

where

\[ y_{31}(t) = \theta(t) - \theta_r(t) \]
\[ y_{32}(t) = \dot{\theta}(t) - \dot{\theta}_r(t) - \beta_3 \]
\[ \beta_3(t) = -c_{31} y_{31}(t) \]  
(31)

And the adaptive law for pitch angle is:

\[ \dot{\hat{\alpha}}_3 = -\gamma_3 \frac{u_{e3}}{J_y} y_{32} \]  
(32)

For yaw angle, the yaw angular acceleration equals to

\[ \ddot{\psi} = \frac{1}{J_z} \left[ b_l^y u_x - \dot{\theta}\psi(J_x - J_y) \right] \]  
(33)

the control law and adaptive law are designed as

\[ u_4 = \hat{\alpha}_4 u_{e4} \]  
(34)

with

\[ u_{e4} = \frac{J}{l} \left[ -c_{42} y_{22} - y_{41} + \dot{\beta}_4 + \ddot{\psi}_r + \frac{1}{J_z} \dot{\theta}\phi(J_y - J_x) \right] \]  
(35)

where
and the adaptive law for yaw angle is

$$\dot{\alpha}_4 = -\gamma_4 \frac{u_{e4}}{J_z} y_{42}$$

(37)

where $c_y(i = 1, 2, 3, 4, j = 1, 2)$ and $\gamma_i(i = 1, 2, 3, 4)$ are positive constants.

### IV. Simulations

The parameters of the quadrotor UAV used in dynamic modeling are given in Table 1 [19] below.

<table>
<thead>
<tr>
<th>symbol</th>
<th>Description</th>
<th>Value</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>$m$</td>
<td>Quad-rotor mass</td>
<td>0.6150</td>
<td>Kg</td>
</tr>
<tr>
<td>$i$</td>
<td>Distance from cg</td>
<td>0.305</td>
<td>M</td>
</tr>
<tr>
<td>$I_R$</td>
<td>Initial moment</td>
<td>0.0154</td>
<td>Kg*m</td>
</tr>
<tr>
<td>$I_y$</td>
<td>Initial moment</td>
<td>0.0154</td>
<td>Kg*m</td>
</tr>
<tr>
<td>$I_z$</td>
<td>Initial moment</td>
<td>0.0309</td>
<td>Kg*m</td>
</tr>
</tbody>
</table>

This section will demonstrate the methodology presented in previous Section III by using the quadrotor UAV nonlinear system. The control objective is to make the system state vector $X = (z \quad \phi \quad \theta \quad \psi)^T$ follow the desired Euler angle $(10, 10, 10)$ (degree) and the desired altitude $z_r = 1 + 0.1t$ under varying faults and uncertainties.

Choose the initial altitude $z = 0$ (meter), the initial Euler angle: $(30, 30, 30)$ (degree), and the initial condition of the derivative of the Euler angles and altitude as:

$$\dot{z}(t) = 0, \ \dot{\phi}(t) = 0, \ \dot{\theta}(t) = 0, \ \dot{\psi}(t) = 0$$

The control constants are chosen by iterative simulation instead of analytical strategy. The control constants employed for the adaptive laws and control laws are selected as:

$$\gamma_1 = 0.01, \ c_{11} = 302, \ c_{12} = 18$$

$$\gamma_2 = 0.015, \ c_{21} = 416, \ c_{22} = 14$$

$$\gamma_3 = 0.0146, \ c_{31} = 408, \ c_{32} = 15$$

$$\gamma_4 = 0.013, \ c_{41} = 390, \ c_{42} = 12.8$$

The following will illustrate the effectiveness of the Lyapunov-based adaptive control strategy gradually from normal situation, system parameter uncertainties, different partial loss for the quad rotors, and the quad rotor partial losses combined with system parameter uncertainties. Since the situations mentioned previously, from the system parameter uncertainties to combined case, have a very small effect on the Euler angles, the simulation results will focus on the altitude analysis.

**Simulation with Normal Case**
Figure 3 Altitude and altitude error with normal case

Figure 4 Roll angle and roll angle error with normal case

Figure 5 Pitch angle and pitch angle error with normal case
The tracking errors and outputs of the Euler angles and altitude in these figures achieve excellent performance, which can be seen from Figure 3 to Figure 6. They demonstrate that the proposed control scheme is executable under normal cases without considering system parameter uncertainties and partial loss.

**Simulations with Uncertainty**

Define the uncertainty as system parameter (mass and inertial moments) alternations, the quadrotor UAV model can be:

\[
\begin{align*}
\ddot{z} &= \frac{u_z}{m} \cos \theta \cos \phi - g \\
\dot{\phi} &= \frac{1}{J_x}[u_x l - \dot{\theta} \dot{\psi} (J_z^l - J_y^l)] \\
\dot{\theta} &= \frac{1}{J_y}[u_y l - \dot{\phi} \dot{\psi} (J_z^l - J_x^l)] \\
\dot{\psi} &= \frac{1}{J_z}[u_z l - \dot{\theta} \dot{\phi} (J_y^l - J_x^l)] \\
\end{align*}
\]

(38)

When system uncertainty is 50%, it means that the system parameters (mass and inertial moments) are less by 50% of their normal values: 

\[m' = 0.5m, J_x' = 0.5J_x, J_y' = 0.5J_y, J_z' = 0.5J_z\]

where \(m, J_x, J_y, J_z\) are the system parameters under normal cases. The same method is used for 80% system parameter uncertainty. The system parameter uncertainty happens at 5 seconds.

Figure 7 shows system performance under situations which are uncertainty free, and uncertainty under 50% and 80% of system parameters reductions. The Lyapunov-based adaptive controller has strong robustness so that it can be fast to convergence to be stable. Moreover, higher uncertainty has almost the same performance as the uncertainty free instance, which is clearly shown in Figure 7 since 80% uncertainty performance overlaps with the one which has 50% uncertainty or the one which is uncertainty free. The excellent system performance shown in Figure 7 is not only its convergent speed, but also its steady-state error which is almost zero.
Simulations with Partial Loss Faults

The simulation here will discuss the quad rotor partial losses. Three working conditions will be discussed: no partial loss \( b^* = 1 \), partial loss decreases 50% \( b^* = 0.5 \) and 80% \( b^* = 0.2 \). Partial loss happens at 5 seconds.

As shown in Figure 8, the severe partial loss situation discussed is 80% quad rotor partial losses. When comparing the 80% partial loss with the fault free case, their altitude errors only show a little difference, which means that the adaptive controller has the strong ability to overcome the variable partial losses and reach a satisfactory performance.

Simulations with Partial Loss Fault Combined with Uncertainty

This part reveals the simulation results which are 50% uncertainty combined with 80% partial loss and 80% uncertainty combined with 80% partial loss respectively. Note that partial loss and system uncertainty will happen simultaneously at 5 seconds. The performances shown in Figure 9 illustrate that the Lyapunov-based adaptive control scheme has strong robustness to handle uncertainty and partial loss simultaneously.
V. Conclusion and Future Work

This paper has designed the Lyapunov-based adaptive controller and utilizes it for fault tolerant control of the quadrotor UAV. The simulations are implemented with normal case, uncertainty at different levels, and varying quad rotor partial losses. Furthermore, simulations of the severe fault scenarios, 80% partial loss combined with 50% uncertainty and 80% uncertainty respectively have been presented.

The simulation results in this chapter demonstrate that the Lyapunov-based adaptive controller has a very strong ability to overcome not only uncertainty and partial loss, but also the combined severe situations. When comparing the Lyapunov-based control with fixed controller gains in our previous work [10], the Lyapunov-based adaptive control approach clearly has advantages in achieving better fault-tolerant capability.

This research work mostly concentrate on the partial loss faults of the quadrotor UAV. It does not solve the stuck fault scenario of the quadrotor UAV since the stuck fault changes the quadrotor’s mathematical model, and the corresponding controller is required to be redesigned simultaneously. Therefore, future work will be as follows:

- Utilize the FTCS scheme and redesign the controller and the corresponding mathematical model to accommodate stuck faults;
- Extend the research work discussed in this thesis to other types of UAVs or aircrafts, in order to demonstrate the effectiveness of the proposed method;
- Test and evaluate the proposed control approaches in the physical quadrotor UAV test-bed.

References


