PROPORTIONAL DERIVATIVE ACTIVE FORCE CONTROL FOR “X” CONFIGURATION QUADCOPTER

By Ni’am Tamami
PROPORTIONAL DERIVATIVE ACTIVE FORCE CONTROL FOR "X" CONFIGURATION QUADCOPTER

Nia'ami T, Endra Pitowarno I, Gede Puja Astawa A

Electronics Engineering Polytechnic Institute of Surabaya
Kampus PENS, Jalan Raya ITS, Sukolilo, Surabaya 60111, Indonesia

Received 21 August 2014, revised 10 September 2014; accepted 16 September 2014
Published online 24 December 2014

Abstract

This paper presents a stabilization control method for "X" configuration quadcopter using PDAFC (Proportional Derivative Active Force Control). PDAFC is used to stabilize quadcopter, whereas AFC is used to reject disturbance uncertainty (e.g., wind) by estimating disturbance torque value of quadcopter. Simulation result shows that PDAFC is better than PD and AFC can minimize disturbance uncertainty effect. The sensitivity toward disturbance uncertainty can be set to zero. Constant disturbance simulation result shows that the best sensitivity constant (G0) is 0.15, the quadcopter maximum error is 0.125 radian and can stable in 5 seconds. Fluctuated disturbance simulation result shows that PDAFC with 0.18 sensitivity constant gives lowest RMS error value, there are 0.074 radian for sine disturbance, 0.055 radian for sawtooth disturbance, and 0.092 radian for square pulse disturbance.

Keywords: "X" configuration quadcopter, PD, AFC

I. INTRODUCTION

Unmanned aerial vehicles (UAVs) have been developed and used over the last few years. UAVs can be built not only for a hobby but also for performing important tasks such as area mapping, surveillance, disaster monitoring, air pollution monitoring, etc. They are capable to hover without an on-board pilot. UAVs become good choice because it has low operational cost and also is safe in important task where risk to pilot are high.

Quadcopter has a simple structure. It utilizes rotors which are directed upwards and placed at the end of a crossed frame. It is controlled by adjusting the angular velocities of each rotor. The quadcopter biggest advantage is that the blades do not have to be movable. A normal helicopter has blades that can be tilted up or down to vary lift. They have complex joints at the hub of the blade, which makes the blades hard to manufacture, difficult to maintain, and very dangerous if any failure occurs. Moreover, quadcopter can take off, land in limited spaces and hover above targets. These vehicles have certain advantages over conventional fixed-wing aircraft for surveillance and inspection tasks.

There are many researches about quadcopter control algorithms and uncertainty disturbance rejection. Bouabdallah et al. designed an LQ controller and PID controller then compared it [1]. The PID controller result is better than LQ controller. Jun Li and Yuntang Li designed PID controller to control angular and linear position, and succeeded to stabilize quadcopter [2]. Mokhtari and Benallegue applied state parameter control to quadcopter rotation angle [3]. By using state observer, quadcopter can measure external disturbance. Gupta et al. described that "X" configuration quadcopter is more stable than "I" configuration quadcopter because of the distribution of rotor force during hover [4]. Bora and Erdine have been controlling position of quadcopter using PD controller and combined by using a vision system [5]. Pounds et al. developed independent linear SISO controllers to regulate quadcopter using PID controller [6]. A. Tayebi et al. proposed a controller which is based upon the compensation of the Coriolis and gyroscopic torques and the use of PD+ feedback structure [7]. Sumantri et al. designed a sliding
mode control using a nonlinear sliding surface (NSS) to design a robust tracking controller for a quad-rotor helicopter [8]. Chen and Hameh used linear H∞ controller to achieve stabilization in angular rates, vertical velocity, longitudinal velocity, lateral velocity, yaw angle, and height of a quadcopter [9]. A linear H∞ controller can be designed to obtain stabilization and tracking performance if the control system.
force of each motor, and "m.g" arrow is weight force of quadcopter. From Li et al., the thrust and hub force for each rotor \((F_i, H_i)\) can be represented in equation (1) and (2) [2]. Thrust force is the resultant of vertical forces acting on all blade elements. Hub force is the resultant of the horizontal forces acting on all blade elements.

\[
F_i = \frac{1}{2} \rho C_T \Omega_i^2 \\
= k_i \Omega_i^2 \\

H_i = \rho C_d \Omega_i^2 \\
= \frac{2C_d}{C_T} F_i \\
= k_d F_i \\
= k_d \frac{1}{2} \rho C_T \Omega_i^2 \\
= k_d \Omega_i^2
\]

where \(\rho\) is air density; \(C_T\) is thrust constant that depends on polar lift slope, geometric blade, velocity through motor, the ratio of the surface area and rotor disk area [6]. \(C_d\) is drag constant, and \(\Omega_i\) is propeller rotation speed.

Quadcopter can change its position by combining translation and rotation [34]. Linear movement on the quadcopter can be produced by total thrust force of the four rotors in equation (3), whereas changes in the angle of rotation (roll, pitch, yaw) will cause a change in the direction of quadcopter translational movement. So, the total forces of the quadcopter can be decomposed into force elements in each axis \(F_x, F_y, F_z\). Figure 3 shows the illustration of force decomposition to each axis in body frame {B}.

\[
F_{\text{total}} = \sum_{i=1}^{4} F_i
\]

Equation (4) is rotation matrix of quadcopter. \(C, S\) are cosine and sine function respectively.

\[
R = \begin{bmatrix}
C\phi C\psi C\theta - S\psi S\theta & C\phi C\psi S\theta + S\psi C\theta & C\phi S\psi + C\phi C\theta S\psi \\
C\phi S\psi C\theta + S\phi S\theta & C\phi S\psi S\theta - C\psi C\theta & C\phi C\psi - S\phi C\theta S\psi \\
-\sin \psi & \cos \psi & 0
\end{bmatrix}
\]

The derived model of quadcopter translational movement can be represented as equation (5). Where \(\ddot{x}, \ddot{y}, \ddot{z}\) are linear acceleration in of quadcopter in each axis.

\[
m \begin{bmatrix}
\ddot{x} \\
\ddot{y} \\
\ddot{z}
\end{bmatrix} = R \begin{bmatrix}
r_{1x} & 0 & 0 \\
0 & r_{2y} & 0 \\
0 & 0 & r_{3z}
\end{bmatrix} \begin{bmatrix}
\dot{f}_x \\
\dot{f}_y \\
\dot{f}_z
\end{bmatrix}
\]

The model also contains a gyroscopic effect. Derived torque models of quadcopter are presented in equation (6), (7), and (8).

\[
\begin{bmatrix}
\tau_x \\
\tau_y \\
\tau_z
\end{bmatrix} = \begin{bmatrix}
F_1 \cos \psi \sin \theta & F_2 \cos \psi \sin \theta & F_3 \cos \psi \sin \theta \\
F_1 \cos \theta \cos \psi & F_2 \cos \theta \cos \psi & F_3 \cos \theta \cos \psi \\
F_1 \sin \theta & F_2 \sin \theta & F_3 \sin \theta
\end{bmatrix}
\]

\[
\begin{bmatrix}
\tau_x \\
\tau_y \\
\tau_z
\end{bmatrix} = R \begin{bmatrix}
I_{xx} & 0 & 0 \\
0 & I_{yy} & 0 \\
0 & 0 & I_{zz}
\end{bmatrix} \begin{bmatrix}
\dot{f}_x \\
\dot{f}_y \\
\dot{f}_z
\end{bmatrix}
\]

\[
\tau_x = I_{xx} \ddot{x} + \dot{\phi} \theta \ddot{y} - \dot{\phi} \psi \ddot{z} - \dot{\psi} \theta \ddot{x} - \dot{\phi} \psi \ddot{y} + \dot{\phi} \theta \ddot{z} + \tau_1 \\
\tau_y = I_{yy} \ddot{y} + \dot{\phi} \psi \ddot{z} - \dot{\psi} \theta \ddot{x} + \dot{\phi} \theta \ddot{z} + \tau_2 \\
\tau_z = I_{zz} \ddot{z} + \dot{\phi} \theta \ddot{x} - \dot{\phi} \psi \ddot{y} + \dot{\psi} \theta \ddot{x} + \dot{\psi} \theta \ddot{y} + \tau_3
\]

\[
\begin{bmatrix}
\tau_x \\
\tau_y \\
\tau_z
\end{bmatrix} = \begin{bmatrix}
\dot{u}_1 \\
\dot{u}_2 \\
\dot{u}_3
\end{bmatrix}
\]

Figure 2. Force distribution in quadcopter

Figure 3. Total forces illustration that decomposed into each axis
derived by substituting equation (1) to (3), \( u_2 \) is roll torque control input, \( u_3 \) is pitch torque control input, and \( u_4 \) is yaw torque control input can be derived by substituting equation (1) to (3) to (7) and equation (2) to (8). Where, \( \theta \) and \( k_d \) are constant values from equation (1) and (2).

\[
\begin{align*}
    u_1 &= f_{\text{total}} \\
    &= k_1 \sum_{i=1}^{4} \Omega_i^2 \\
    u_2 &= \tau_x \\
    &= k_1 \sum_{i=1}^{4} \Omega_i^2 \cos \left( \frac{\pi}{4} (i - 1) + \frac{\pi}{4} \right) \\
    u_3 &= \tau_y \\
    &= k_1 \sum_{i=1}^{4} \Omega_i^2 \sin \left( \frac{\pi}{4} (i - 1) + \frac{\pi}{4} \right) \\
    u_4 &= \tau_z \\
    &= k_1 \sum_{i=1}^{4} \Omega_i^2 (1) \Omega_i^2
\end{align*}
\]  

By substituting equation (9) into (5) to (8), the derived model of quadcopter in (10).

\[
\begin{align*}
    \dot{x} &= \left( S \dot{\phi} \dot{\psi} + C \dot{\phi} \dot{\psi} \dot{\phi} \right) u_2 \\
    \dot{y} &= \left( C \dot{\phi} \dot{\psi} \dot{\theta} - C \dot{\phi} \dot{\psi} \dot{\phi} \right) u_1 \\
    \dot{z} &= \left( C \dot{\phi} \dot{\psi} \dot{\phi} \right) u_1 - \dot{\theta} \\
    \dot{\phi} &= \frac{\dot{x} + \phi \dot{y} (1 + \psi)}{x} \\
    \dot{\psi} &= \frac{\dot{y} + \psi \dot{z} (1 + \phi)}{x} \\
    \dot{\theta} &= \frac{\dot{z} + \psi \dot{y} (1 + \phi)}{y}
\end{align*}
\]  

where \( \phi, \dot{\phi}, \dot{\psi}, \dot{\psi} \) are roll, pitch, yaw, angular acceleration at quadcopter body.

### III. Quadcopter Controller Design

In this section, the control algorithm of quadcopter is presented. The purpose is to combine PD and AFC as rotational controller to stabilize quadcopter. Figure 4 shows quadcopter control structure. Figure 5 shows the proposed rotational controller to stabilize quadcopter. In this simulation, translational movement are neglected. The controller design is focused to stabilize quadcopter toward disturbance. PD controller is used to stabilize quadcopter and AFC to reject uncertainty disturbance from environment. In this simulation, quadcopter get constant and fluctuated disturbance.

From Figure 4, the relationship of each input and each state can be represented as:

\[
\begin{align*}
    X &= AX + BU \\
    \dot{X} &= \left[ \phi \quad \theta \quad \psi \quad \dot{\phi} \quad \dot{\theta} \quad \dot{\psi} \right]^T \\
    U &= \left[ u_1 \quad u_2 \quad u_3 \quad u_4 \right]^T
\end{align*}
\]  

The system matrix (A) can be represented as:

\[
A = \begin{bmatrix}
0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0
\end{bmatrix}
\]  

The control matrix (B) can be represented as:

\[
B = \begin{bmatrix}
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix}
\]  

Figure 4. Quadcopter control structure
A. Disturbance Model

In this subsection, the model of disturbance will be presented. Figure 1 shows disturbance position of quadcopter, disturbance mass located at \((L_{DXY}, L_{DYZ})\) from the center of quadcopter in (X,B,Y) axis. State equation (11) can be written as follows:

\[
\dot{X} = AX + BU + Dis
\]  

(14)

The simulation disturbance is:

\[
Dis = [0 \quad 0 \quad 0 \quad Dis_{X} \quad Dis_{Y} \quad Dis_{Z} \quad 0 \quad 0]^{T}
\]  

(15)

B. PD Controller Design

PD controller will be presented to stabilize quadcopter. The reason is this controller very simple and easy implemented. In this section, PD control algorithm is designed without disturbance parameter. The controller design is focused to stabilize quadcopter when hovering without get uncertainty disturbance. The model that presented at section 2 is completed by gyroscopic effect. Gyroscopic effect can be ignored because it does not have significant effect on quadcopter system [14]. The model can be simplified:

\[
\begin{align*}
\phi &= k_{\phi} \sum_{i=1}^{n} \omega_{i}^{2} \cos\left(\frac{\pi}{2}(i-1) + \frac{\pi}{4}\right) \\
\theta &= k_{\theta} \sum_{i=1}^{n} \omega_{i}^{2} \sin\left(\frac{\pi}{2}(i-1) + \frac{\pi}{4}\right) \\
\psi &= k_{\psi} \sum_{i=1}^{n} (-1)^{i} \omega_{i}^{2}
\end{align*}
\]  

(16)

\[
\begin{align*}
\phi &= \frac{k_{\phi} \sum_{i=1}^{n} \omega_{i}^{2} \cos\left(\frac{\pi}{2}(i-1) + \frac{\pi}{4}\right)}{I_{xx}} \\
\theta &= \frac{k_{\theta} \sum_{i=1}^{n} \omega_{i}^{2} \sin\left(\frac{\pi}{2}(i-1) + \frac{\pi}{4}\right)}{I_{yy}} \\
\psi &= \frac{k_{\psi} \sum_{i=1}^{n} (-1)^{i} \omega_{i}^{2}}{I_{zz}}
\end{align*}
\]  

(17)

From equation (17), the model is second order form, in order to make it possible to design multiple PD controllers for this system, one can neglect gyroscopic effects thus remove the cross coupling [1]. This is PD controller for each orientation angle.

\[
u_2, u_3, u_4 = P_{\phi, \theta, \psi} \phi, \theta, \psi + D_{\phi, \theta, \psi} \phi, \theta, \psi
\]  

(18)

where \(u_2, u_3, u_4\) are control input for roll, pitch, and yaw, respectively; \(P_{\phi, \theta, \psi}\) and \(D_{\phi, \theta, \psi}\) are proportional control for roll, pitch, and yaw respectively.

C. AFC Controller Design

AFC controller is designed to reject uncertainty disturbance from environment. Figure 6 shows AFC block diagram that used in simulation. This block has two inputs, they are measured angular velocity and applied propeller speed.

Let us define \(\gamma\) as rotation angle roll and pitch axis (\(\phi, \theta\)),

\[
\dot{\gamma} = \frac{\gamma}{\text{dt}}
\]  

(19)

\[
\begin{align*}
\Omega_{\phi} &= \dot{\Omega}_{\phi} + \Omega_{2} - \Omega_{3} - \Omega_{4} \\
\Omega_{\theta} &= \dot{\Omega}_{\theta} + \Omega_{2} - \Omega_{3} + \Omega_{4}
\end{align*}
\]  

(20)

\[
\gamma_{ref} = \frac{0.5 \rho C_{L} \Omega_{2} \text{\Omega}_{3}}{I_{xx,yy}}
\]  

(21)
\[ \ddot{y}_{AFC} = \ddot{y}_{ref} - \ddot{y} \]  
\[ \Omega_{AFC} = c_{sens} \left( \frac{\ddot{y}_{AFC} \times \dddot{y}_{ref}}{0.5 \Omega_{ref}} \right)^{0.5} \] 
with \( 0 \leq \ddot{y}_{AFC} \leq 1 \) 
\[ = k_{AFC} (\ddot{y}_{ref} - \ddot{y}) \]  

First input is measured angular velocity that differentiated into actual angular acceleration in equation (19). Second input is applied propeller speed that converted into angular acceleration reference in equation (21). \( \ddot{y}_{AFC} \) is estimated disturbance acceleration. To get estimated disturbance, actual angular acceleration is compared by angular acceleration reference in equation (22) [11]. Last, convert the disturbance acceleration into propeller speed in equation (23) then add the result with PD controller result. \( c_{sens} \) is a constant value to set AFC sensitivity output toward disturbance, then simplified to \( k_{AFC} \). \( \Omega_{AFC} \) in propeller speed calculation of AFC controller output.

**IV. Simulation Result**

The simulation test was performed using SIMULINK to evaluate the performance of the controller. The simulation model (10) was used in S-Function block. In this simulation, the model contain disturbance that has been modeled in section 3, there are constant and fluctuated disturbances.

Before doing some simulation process, the parameters of quadcopter must be collected from real data. This simulation used quadcopter data obtained from [16]. They are listed in Table 1. PD coefficients that used for simulations were derived by trial and error to get best performance, the PD parameter are listed in Table 2. First simulation compared PD and PDAFC performance when they constant disturbance.

**Table 1. Quadcopter simulation parameter**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Unit</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>M</td>
<td>kg</td>
<td>1.025</td>
</tr>
<tr>
<td>L</td>
<td>meter</td>
<td>0.270</td>
</tr>
<tr>
<td>k₁</td>
<td>Ns²</td>
<td>3.122e-06</td>
</tr>
<tr>
<td>k₀</td>
<td>Nms²</td>
<td>1.759e-08</td>
</tr>
<tr>
<td>Izz</td>
<td>kgm²</td>
<td>0.012</td>
</tr>
<tr>
<td>Iyy</td>
<td>kgm²</td>
<td>0.048</td>
</tr>
<tr>
<td>Disx</td>
<td>N Amp x WaveForm(Freq)</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0.2</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>0.2 x S(250,4)</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0.2 x sawtooth(0.4 Hz)</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>0.2 x square(0.4 Hz)</td>
<td></td>
</tr>
<tr>
<td>l₁x</td>
<td>mm</td>
<td>0</td>
</tr>
<tr>
<td>l₁yy</td>
<td>mm</td>
<td>190</td>
</tr>
</tbody>
</table>

**Table 2. PD coefficients simulation parameter**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>KP roll</td>
<td>0.697</td>
</tr>
<tr>
<td>KD roll</td>
<td>0.036</td>
</tr>
<tr>
<td>KP pitch</td>
<td>0.697</td>
</tr>
<tr>
<td>KD pitch</td>
<td>0.036</td>
</tr>
<tr>
<td>KP yaw</td>
<td>0.0001368</td>
</tr>
<tr>
<td>KD yaw</td>
<td>0.0000684</td>
</tr>
</tbody>
</table>
sensitivities constants \( C_{\text{sen}} \) in equation (23), they were 0.13, 0.15 and 0.18. By using PD, maximum error is 0.326 radian with RMS value is 0.060. PDAFC with 0.13 constant, maximum error is 0.153 radian and RMS value is 0.029. Then with 0.15 constant, maximum error is 0.125 radian and RMS value is 0.017, it can stable in 5 seconds. Last is PDAFC with 0.18 constant, maximum error is 0.090 radian and RMS value is 0.018, but still noisy because of the controller became more sensitive with disturbance.

Figure 8 shows second simulation result to compare PD method and PDAFC method with sine function disturbance. In this simulation, disturbance maximum amplitude was 0.2 with frequency 0.4 Hz. PDAFC was tested with three sensitivities constant \( C_{\text{sen}} \) in equation (23), which were 0.13, 0.15 and 0.18. PD maximum error is 0.394 radian with RMS value of 0.255. PDAFC with 0.13 constant, maximum error was 0.210 radian and RMS value is 0.121. Then with 0.15 constant, maximum error is 0.161 radian and RMS value is 0.098. Last is PDAFC with 0.18 constant, maximum error is 0.130 radian and RMS value is 0.074. PDAFC with 0.18 constant give lowest RMS error value.

Figure 9 shows third simulation by using sawtooth function disturbance. In this simulation, disturbance maximum amplitude is 0.2 with frequency 0.4 Hz. PDAFC was tested with three sensitivities constant \( C_{\text{sen}} \), they were 0.13, 0.15 and 0.18. PD maximum error is 0.241 radian with RMS value is 0.186. PDAFC with 0.13 constant, maximum error is 0.241 radian and RMS value is 0.092. Then with 0.15 constant, maximum error is 0.199 radian and RMS value is 0.073. Last is PDAFC with 0.18 constant, maximum error is 0.156 radian and RMS value is 0.055. PDAFC with 0.18 constant give lowest RMS error value.

Figure 10 shows fourth simulation by using square function disturbance. In this simulation, disturbance maximum amplitude was 0.2 with frequency 0.4 Hz. PDAFC was tested with three sensitivities constant \( C_{\text{sen}} \), they were 0.13, 0.15 and 0.18. PD maximum error is 0.575 radian, RMS value is 0.317. PDAFC with 0.13 constant, maximum error is 0.315 radian and
RMS value is 0.170. Then with 0.15 constant, maximum error is 0.272 radian and RMS value is 0.128. Last is PDAFC with 0.18 constant, maximum error is 0.190 radian and RMS value is 0.092. PDAFC with 0.18 constant give lowest RMS error value.

V. CONCLUSION

An “x” configuration quadcopter has been successfully modeled. Then, simulation results have been presented to show the controller performance. By adding PD with AFC, better results were obtained. From the simulation, PDAFC controller can minimize the effect of disturbance. Inconstant disturbance simulation, the best sensitivity constant ($C_{sens}$) was obtained when the value was 0.15, the quadcopter maximum error 0.125 radian and could stable in 5 seconds. In fluctuated simulation result, PDAFC with 0.18 constant gave lowest RMS error value, 0.074 radian for sine disturbance, 0.055 radian for sawtooth disturbance, and 0.092 radian for square pulse disturbance.

ACKNOWLEDGEMENT

The author would like to thank the Electronic Engineering Polytechnic Institute of Surabaya for giving laboratory facilities for the research.

REFERENCES


PROPORTIONAL DERIVATIVE ACTIVE FORCE CONTROL FOR “X” CONFIGURATION QUADCOPTER

ORIGINALITY REPORT

25% SIMILARITY INDEX

PRIMARY SOURCES


3. worldwidescience.org

4. hplusmagazine.com

5. ace.utsa.edu

6. www.idsc.ethz.ch

7. daedalus.scl.sztaki.hu

<table>
<thead>
<tr>
<th>No.</th>
<th>Source</th>
<th>Title</th>
<th>Authors/Rewriters</th>
<th>Year/Publication Details</th>
<th>Words</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>9</td>
<td>CrossCheck</td>
<td>Bora Erginer. &quot;Design and implementation of a hybrid fuzzy logic controller for a quadrotor VTOL vehicle&quot;</td>
<td>Bora Erginer</td>
<td>02/2012</td>
<td>27</td>
<td>1%</td>
</tr>
<tr>
<td>10</td>
<td>CrossCheck</td>
<td>P. Castillo. &quot;Stabilization of a mini rotorcraft with four rotors&quot;</td>
<td>P. Castillo</td>
<td>12/2005</td>
<td>26</td>
<td>1%</td>
</tr>
<tr>
<td>13</td>
<td>Internet</td>
<td>Jin, Taeseok. &quot;Dynamic Model and P-PD Control based Flight Performance Evaluation for Hexa-Rotor Type UAV&quot;</td>
<td>Jin, Taeseok</td>
<td>2015</td>
<td>22</td>
<td>1%</td>
</tr>
<tr>
<td>14</td>
<td>CrossCheck</td>
<td>Qiaokang Liang. &quot;A novel thin six-dimensional wrist force/torque sensor with isotropy&quot;</td>
<td>Qiaokang Liang</td>
<td>06/2009</td>
<td>21</td>
<td>1%</td>
</tr>
<tr>
<td>15</td>
<td>Internet</td>
<td>eprints.utm.my</td>
<td>eprints.utm.my</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Min Wang. "Fault tolerant control of feedback linearizable systems with stuck actuators", Asian Journal of Control, 01/2008


Darren M. Dawson. "Output Feedback Tracking
Control of an Underactuated Quad-Rotor UAV", 2007 American Control Conference, 07/2007


<table>
<thead>
<tr>
<th>No.</th>
<th>Source</th>
<th>Title</th>
<th>Conference/Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>34</td>
<td><a href="http://www.icrobotics.co.uk">www.icrobotics.co.uk</a></td>
<td>S.J. Kwon. &quot;RIC(Robust Internal-loop Compensator) Based Flight Control of a Quad-Rotor Type UAV&quot;</td>
<td>2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2005</td>
</tr>
</tbody>
</table>