Control System Design and Simulation
For a Quadrotor Helicopter

P. NIRUT, Ryosuke MASUDA, Hiroshi HIRATA
Graduate School of Computer and Information Engineering, Tokai University, Japan

1. Introduction
This paper studies to simulate the operation of aircraft with four propellers. It is known as the unmanned aerial vehicle (UAV) with quadrotor mechanism. A design methodology is introduced that blends the classical PID and the fuzzy controller in an intelligent way. The hybrid fuzzy PID controller (HFPID) [1] takes advantage of the nonlinear characteristics of the fuzzy and the accuracy near a setpoint is guaranteed by the classical PID. The paper also explains on the developments Unscented Kalman Filter (UKF) [2] estimation of the orientation from measurements of acceleration and magnetic field sensors. Many fusion techniques, mostly based on Kalman filter to improve the accuracy and robustness, it has been used in many applications such as in robotics, aerospace, under water vehicles, and others. The results of the simulations in the quadrotor are presented.

2. System description

2.1. Quadrotor Aerial Vehicle
In the quadrotor, which represent four input forces that are basically the thrust generated by each propeller as shown in Fig. 1. The collective input \( u \) is the sum of the thrusts of each motor. Translation along the x-axis is related to pitch angle (\( \theta \)). This angle can be obtained by changing rotor 1 or rotor 3 increasing (reducing) speed. On the other hand, translation along the y-axis is result from roll angle (\( \phi \)) by changing rotor 2 or rotor 4 increasing (reducing) speed. The yaw angle (\( \psi \)), the speed of rotor 1’s and 3 should not be same speed of rotor 2 and 4. To balance the moment, the direction of rotor 1 and 3 have to be opposite of rotor 2 and 4 at the same speed.

2.2 Dynamic Equation
In order to move the quadrotor model from the earth frame (\( E \)) to the body frame (\( B \)) in Fig.1. The mathematical design should depend on the direction cosine matrix.

\[
R_{xy} = \begin{bmatrix}
  c\theta & s\phi c\theta & -s\phi s\theta \\
  s\theta & c\phi c\theta & s\phi c\phi \\
  s\phi s\theta & -c\phi s\theta & c\phi c\phi
\end{bmatrix}
\]

Where: \( c \) and \( s \) represent cosine and sine respectively.

The dynamic model of the quadrotor can be obtained via a Lagrange approach and a simplified model is give as follow [3]. We will define the inputs to be

\[
\begin{align*}
  u_1 &= (F_1 + F_2 + F_3 + F_4) / m \\
  u_2 &= (\bar{F}_1 + F_1 - F_2 - F_4) / J_1 \\
  u_3 &= (F_1 + F_2 + F_3 - F_4) / J_1 \\
  u_4 &= C(F_1 - F_2 + F_3 - F_4) / J_3
\end{align*}
\]

Where: \( j_1 \) are the moment of inertia with respect to the axes and \( C \) is the force to moment scaling factor. The \( u_1 \) represents a total thrust in the z-axis, \( u_2 \) and \( u_3 \) are the pitch and roll inputs and \( u_4 \) is a yawing moment. A quadrotor has four input forces and six output state. The equations of motion be come

\[
\begin{align*}
  \dot{x} &= u_1 \cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi \\
  \dot{y} &= u_1 \sin \phi \sin \theta \cos \psi - \cos \phi \sin \psi \\
  \dot{z} &= u_1 \cos \theta \cos \psi - g \\
  \dot{\phi} &= u_4
\end{align*}
\]

3. Control algorithm

3.1 Hybrid fuzzy-PID control
Controller is proposed for position control of the Quadrotor system and its performance is tested by simulation studies. In Fig. 2: It has two main parts: the classical PID and Fuzzy controllers. PID control structure is given in (4).
\[ e_{i+1} = y_{ref} - y \]

\[ U_{i} = K_p e_{i} + K_i \int e_{i} dt + K_d \frac{de_{i}}{dt} \]  

(4)

Where \( e \) is the error between reference position \( y_{ref} \) and position \( y \). \( K_p \) is the proportional gain, \( K_i \) the integral gain, and \( K_d \) the derivative gain.

\[ G(s) = \frac{K}{(rs + 1)} = \frac{K}{(0.5s + 1)} \]  

(5)

Table 1. Rule base for fuzzy logic control

<table>
<thead>
<tr>
<th>( de/e )</th>
<th>NB</th>
<th>NB</th>
<th>NM</th>
<th>NS</th>
<th>Z</th>
<th>PS</th>
<th>PM</th>
<th>PB</th>
</tr>
</thead>
<tbody>
<tr>
<td>NB</td>
<td>NB</td>
<td>NB</td>
<td>NB</td>
<td>NB</td>
<td>NM</td>
<td>NS</td>
<td>Z</td>
<td>Z</td>
</tr>
<tr>
<td>NM</td>
<td>NB</td>
<td>NB</td>
<td>NM</td>
<td>NS</td>
<td>NS</td>
<td>Z</td>
<td>PS</td>
<td>PS</td>
</tr>
<tr>
<td>NS</td>
<td>NB</td>
<td>NM</td>
<td>NS</td>
<td>NS</td>
<td>Z</td>
<td>PS</td>
<td>PM</td>
<td>PM</td>
</tr>
<tr>
<td>Z</td>
<td>NM</td>
<td>NS</td>
<td>NS</td>
<td>Z</td>
<td>PS</td>
<td>PM</td>
<td>PB</td>
<td>PB</td>
</tr>
<tr>
<td>PS</td>
<td>NM</td>
<td>NS</td>
<td>Z</td>
<td>PS</td>
<td>PM</td>
<td>PB</td>
<td>PB</td>
<td>PB</td>
</tr>
<tr>
<td>PM</td>
<td>Z</td>
<td>PS</td>
<td>PS</td>
<td>PM</td>
<td>PB</td>
<td>PB</td>
<td>PB</td>
<td>PB</td>
</tr>
</tbody>
</table>

Table 1. Rule base for fuzzy logic control

Fuzzy Logic controller (FLC) used in this paper is based on two input FLC structure. The input variables are defined as \( e = y_{ref} - y \) and \( de = v_{ref} - v \). Real interval of variables is obtained by using scaling factors which are \( G_e \), and \( G_d \). Fig.2. The fuzzy control rule is in the form of: IF \( e = E \) and \( de = D \) THEN \( \eta = \eta(E, D) \). These rules are written in a rule base, which is shown in Table 1. The rule base structure is Mamdani type [4].

3.2 Unscented Kalman Filter (UKF)

Kalman filter is quite a standard method in state estimation. Recently, the UKF is widely used for estimation of nonlinear systems instead of the extended Kalman filter (EKF). The main difference between the UKF and the EKF is the method to compute the propagation of process and measurement noise covariance in the UKF. The points around the mean, called sigma point, and propagate to next time step by general nonlinear dynamic function, while in the EKF which is based on Talor’s series approximation. Then the propagation is based on linearized model. In this paper we aim to design a UKF to estimate the sensor readings [5]. UKF algorithm proceeds as shown in Fig. 4.

4. Simulation result

Control systems are necessary to make the system response to the reference values as most as possible. This means the system can work properly and efficiently.

Simulation 1

To find out the actual parameters of the transfer function we needed to measure the angular rate of the motor by using the Micro-Tachometer. The dynamics of the motor are nonlinear therefore we have to make it to be a linear model to have implementation of first-order transfer functions. The model should be linearized around an adequate operation point. The System Identification tool GUI in MATLAB was used to estimate the respective time constant, with value 0.5s. In (5)

\[ G(s) = \frac{K}{(rs + 1)} = \frac{K}{(0.5s + 1)} \]  

Simulation 2

To test the effectiveness of the filter, the green lines are actual error. When the yellow lines are the UKF estimate error and the purple lines are the LPF estimate error.

Acknowledgements

The modelling of the motors was achieved due to the use of data collected for identification which originated from the Micro-Tachometer. It is possible to determine the angular speed of the motor propeller assembly. And the filter is able to track rotational motion in Fig. 6. The UKF estimate error is less than the actual error. When it compares with the LPF, the results show that the UKF is more accurate and faster than the LPF. Then we will implement to the real quadrotor system and make the experiment.

References