Modeling and PD Control of a Quadrotor VTOL Vehicle

Bora Erginer and Erdinç Altuğ

Abstract—In this paper, we present a model of a four rotor vertical take-off and landing (VTOL) unmanned air vehicle known as quadrotor aircraft. And we explained its control architecture including vision based control. Quadrotors have generated considerable interest in both the control community due to their complex dynamics and military because of their advantages over regular air vehicles. The proposed dynamical model which comprises gyroscopic effects and its control strategies can be source for future works.

Keywords: Quadrotor, Unmanned Aerial Vehicle (UAV), modeling, aircraft control.

I. INTRODUCTION

An autonomous UAV provides tremendous advantages and is eligible for applications like rescue and research, remote inspection, surveillance, military applications, therefore saving human pilots from dangerous flight conditions. As a UAV quadrotors are very useful when the environment is inaccessible or hard to reach. When the flight is dangerous, monotonous or flight time is extended and flight is not possible even by a skilled pilot an unmanned quadrotor can provide great advantages. As a helicopter, quadrotors have evident advantages over other aircrafts since they can take-off and land in limited area and can easily hover above stable or moving targets. Furthermore, they have great maneuverability which makes quadrotors difficult to control. Additionally, having four rotors increases quadrotors carriage capacity however constrains it to consume more energy.

However control of a quadrotor is a new idea, the quadrotor itself is not new. A full-scale four rotor helicopter was built by De Bothezat in 1921 [1]. Other examples are the Mesicopter [2] and Hoverbot [3]. The studies in quadrotor modeling and control increased rapidly in recent years. Some examples of these studies can be summarized as following; T. Hamel et. al. modeled a quadrotor by incorporating the airframe and motor dynamics as well as aerodynamics and gyroscopic effects and controlled it separating the rigid body dynamics from the motor dynamics [4]. Altuğ et. al. modeled a quadrotor using Euler-Newton method and worked on vision based stabilization and output tracking control using cameras [5, 6, 7, 8]. N. Guernad et. al. and D. Suter et. al. also studied on image based visual servo control for quadrotors [9, 10]. A. Moktari et. al. presented a nonlinear dynamic model for a quadrotor with a state parameter control which is based on Euler angles and open-loop positions state observer [11]. K. M. Zemalache et. al. used natural features for vision based navigation of a quadrotor [12]. A feedback linearization based controller with a high order sliding mode observer running parallel applied to a quadrotor by A. Benallegue et. al. [13]. J. Dunfied et. al. created a neural networks controller for a quadrotor [14]. M.G Earl et. al. used a Kalman filter to estimate the attitude of a quadrotor [15]. In their studies, S. Salazar-Cruz and J. Escareno et. al., used a Lagrangian model and a controller based on Lyapunov analysis using nested saturation control algorithm and designed an embedded control architecture for a quadrotor to perform autonomous hover flight [16, 17]. S. Bouabdallah et. al. mechanically designed, dynamically modeled and used nonlinear control techniques in their works [18, 19]. L. Beji et. al. presented structure and control of a quadrotor where two rotors are bidirectional [20]. H. Romero et. al. used a simple vision system for a quadrotor’s local positioning and orientation in indoor flight [21]. P. Castillo et. al., used a Lagrangian model of the quadrotor and controlled it based on Lyapunov analysis [22, 23, 24]. Finally, A. Tayebi et. al., proposed a controller which is based upon the compensation of the Coriolis and gyroscopic torques and the use of PD\(^2\) feedback structure [25].

II. DYNAMIC MODELING

A quadrotor is an under actuated aircraft with fixed pitch angle four rotors as shown in Figure 1. Modeling a vehicle such as a quadrotor is not an easy task because of its complex structure. Our aim is to develop a model of the vehicle as realistically as possible.

![Fig. 1 The quadrotor and its rotors turning directions.](image-url)
Having four rotors with fixed angles makes quadrotor has four input forces which are basically the thrust provided by each propellers as shown in Figure 2. Forward (backward) motion is maintained by increasing (decreasing) speed of front (rear) rotor speed while decreasing (increasing) rear (front) rotor speed simultaneously which means changing the pitch angle. Left and right motion is accomplished by changing roll angle by the same way. The front and rear motors rotate counter-clockwise while other motors rotate clockwise so yaw command is derived by increasing (decreasing) counter-clockwise motors speed while decreasing (increasing) clockwise motor speeds. The dynamical model and expressions in [25] used with a minor alteration. Let \( I = \{e_x, e_y, e_z\} \) signify an inertial frame, and \( A = \{e_x, e_y, e_z\} \) express a frame which is rigidly attached to the aircraft as shown in Fig.2.

\[
\begin{align*}
&\dot{\xi} = v \\
&\dot{v} = g e_z - \frac{1}{m} T R e_z \\
&R = RS(\Omega) \\
&I_i \dot{\Omega} = -\Omega \times I_j \Omega - G_a + \tau_a \\
&I_i \dot{\omega}_i = \tau_i - Q_i, \quad i \in \{1, 2, 3, 4\}
\end{align*}
\]

where the vector \( \xi = [x \ y \ z]^T \) denotes the position of the origin of the body fixed frame \( A \) with respect to the inertial frame \( I \), the vector \( v = [v_x \ v_y \ v_z]^T \) denotes the linear velocity of the origin of \( A \) expressed in \( I \), \( \Omega = [\Omega_1, \Omega_2, \Omega_3]^T \) denotes the angular velocity of the airframe expressed in the body fixed frame \( A \), \( \times \) denotes the vector cross product, \( m \) represents the mass of the airframe. Airframe torques are denoted by \( \tau_a = [\tau^1_a \ \tau^2_a \ \tau^3_a]^T \) and torques are,

\[
\begin{align*}
\tau^1_a &= db(\omega^2_a - \omega^2_1) \\
\tau^2_a &= db(\omega^2_2 - \omega^2_3) \\
\tau^3_a &= \kappa(\omega^2_1 + \omega^2_2 - \omega^2_3 - \omega^2_4)
\end{align*}
\]

where \( \kappa \) and \( b \) are parameters depending on the shape and structure of the blades, density of the air and other factors, \( d \) is the distance from the center of rotors to the center of mass of the quadrotor. \( G_a \) denotes the gyroscopic torques and derived as,

\[
G_a = \sum_{i=1}^{4} I_i (\Omega \times e_x)(-1)^{i+1} \omega_i
\]

\( T \) denotes the total thrust composed by rotor forces and is given by

\[
T = \sum_{i=1}^{4} |f_i|
\]

where \( f_i \) is the lift force of rotor \( i \) and derived by

\[
f_i = -b \omega^2_a e_3
\]

The orientation matrix \( R \in SO(3) \) is given by

\[
R = \begin{pmatrix}
C_\phi C_\psi & C_\phi S_\psi S_\theta - S_\phi C_\theta & C_\psi S_\phi C_\theta + S_\psi S_\theta \\
C_\psi S_\phi & C_\psi S_\theta + S_\phi C_\psi & -C_\psi S_\phi C_\theta + S_\psi S_\theta \\
-S_\theta & S_\phi C_\theta & C_\theta C_\phi
\end{pmatrix}
\]

and \( \zeta = [\phi \ \theta \ \psi]^T \) is the Euler angles vector, where \( \phi, \theta \) and \( \psi \) are denoting, respectively roll, pitch and yaw angles. \( g \) denotes the acceleration due to gravity, \( I_i \in \mathbb{R}^{3 \times 3} \) is a symmetric positive-definite constant inertia matrix of the airframe with respect to the frame \( A \). \( \omega_i \) denotes the speed of the rotor \( i \). \( I_i \) denotes the moment of inertia of one rotor. \( S(\Omega) \) is a skew-symmetric matrix and given by
Reactive torque $Q_i$, constituted by the rotor $i$ due to rotor drag, is given by
\[ Q_i = \kappa \omega_i^2, \tag{14} \]
and $\tau_i$ is the torque produced by rotor $i$.

III. CONTROL of the QUADROTOR HELICOPTER

In this section we present a controller design for the model of the quadrotor represented in Eqn’s 1-5. The controller design in [6, 7] used for the quadrotor model with slight modification. Let us define the control inputs of the model to be
\[ u_1 = -(f_1 + f_2 + f_3 + f_4) \]
\[ u_2 = (f_4 - f_2) \]
\[ u_3 = (f_3 - f_1) \]
\[ u_4 = (f_1 - f_2 + f_3 - f_4) C \tag{15} \]
where $C$ is the force-to-moment scaling factor, $u_1$ represents a total thrust on the body along z-axis, $u_2$ and $u_3$ are the roll and pitch inputs and $u_4$ is a yawing moment. The inputs can be represented in the matrix form as $u = Mf$. $M$ can be written as
\[ M = \begin{pmatrix} -1 & -1 & -1 & -1 \\ 0 & -d & 0 & d \\ d & 0 & -d & 0 \\ C & -C & C & -C \end{pmatrix} \tag{16} \]
Quadrotor actuates motors to turn propellers which generate the $f_i$ forces. As we use $f_i$ forces in our model, we need to derive the equation which shows how $U_i$ inputs constitute $f_i$ forces. $f_i$ forces can be represented as $f = M^{-1} U$.

A. Proportional Derivative Controllers

The motion along the x-axis and y-axis are related to the pitch and roll angles respectively. One can design proportional derivative controllers to control pitch and yaw angles in order to control x and y motions. We can control the motion along x axis with a PD controller given by
\[ u_3 = K_{p1}(\theta_d - \theta) + K_{d1}(\dot{\theta}_d - \dot{\theta}) \tag{17} \]
The desired pitch angle ($\theta_d$) can be written as
\[ \theta_d = \arcsin \left( K_p (x - x_d) + K_d (\ddot{x} - \ddot{x}_d) \right) \tag{18} \]
Derivative of pitch angle, gives the desired pitch angle velocity
\[ \dot{\theta}_d = \frac{K_p \ddot{x} + K_d x^2}{\sqrt{1 - K_p^2 \dot{x}^2 - 2K_p K_d \dot{x} \ddot{x} - K_d^2 \ddot{x}^2}} \tag{19} \]
Likewise, the desired roll angle ($\phi_d$) and desired roll angle velocity ($\dot{\phi}_d$), by the same way. PD controller for y motions is given by
\[ u_4 = K_{p2}(\phi_d - \phi) + K_{d2}(\dot{\phi}_d - \dot{\phi}) \tag{20} \]
Yaw angle ($\psi$) and altitude can be controlled by PD controllers given by
\[ u_4 = K_{p3}(z_d - z) + K_{d3}(\dot{z}_d - \dot{z}) \cos \theta \cos \phi \tag{22} \]
K coefficients which used for $u_i$ controllers derived by trial and error for best performance. The controllers have been simulated on MATLAB Simulink model. In this simulation shown in Figure 3, helicopter starts at (5, 4, 3) m position with 30 degrees yaw and zero degree pitch and roll angles. The desired position is origin with zero yaw, pitch and roll angles. Figure 4 shows helicopter’s this movement in x, y and z axes.

Fig. 3 PD Controller Simulation Results.

Each rotor of the quadrotor turns in one direction as it is shown in Fig.1, to reach from 5m altitude to zero, quadrotor
shuts down its motors and free falls to reach its target altitude as fast as possible. Hence, x and y motions and yaw, pitch and roll angles stay still up to quadrotor actuates its motors to provide its altitude target.

![Quadrotor movement in x, y and z axes.](image)

**Table 1: Parameters used for the controllers.**

<table>
<thead>
<tr>
<th>Kp</th>
<th>0.82</th>
<th>Kp2</th>
<th>80</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kd</td>
<td>1.5</td>
<td>Kd2</td>
<td>15</td>
</tr>
<tr>
<td>Kp1</td>
<td>3</td>
<td>Kp3</td>
<td>100</td>
</tr>
<tr>
<td>Kd1</td>
<td>0.4</td>
<td>Kd3</td>
<td>50</td>
</tr>
</tbody>
</table>

**IV. USING VISION TO CONTROL THE QUADROTOR HELICOPTER**

Vision can be used to identify and to estimate relative position of the objects with respect to a flying vehicle. Vision can also be used to estimate flying vehicle’s position and orientation if the reference feature’s position and orientation are known. This information is called as the pose of the helicopter. There are various methods to estimate the pose of the helicopter using features as well as without artificial features. This section will describe a control approach that uses artificial features (e.g. color blobs) placed at the ground and the control algorithms use the visual information to guide the vehicle.

**General approach to use vision for a flying vehicle has been described in Figure 5. The video stream of the features positioned on a landing pad processed with a feature extractor routine (onboard or off-board the vehicle) to identify the artificial features.**

When pose estimation is used, the image coordinates $u_i$ and $v_i$ for each of the features will be used with the camera parameters to identify the vehicle position and the orientation. The control signals will then use the pose information to drive the helicopter. Alternatively, one can avoid the pose estimation and use the image parameters directly in the control signals. The GPS and IMU sensors on flying vehicles provide some of the states of the helicopter. The vision sensor can be used to estimate the relative position and orientation of the goal point (e.g. landing pad).

Using such an approach, the altitude control can be performed using control signal $u_i$ as given in Equation 23. This signal is a function of image coordinates, desired altitude, pitch and roll angles.

$$u_i = \frac{g + K_p (z_d - \frac{f \cdot d}{l}) + K_d (\dot{z}_d - \frac{f \cdot d}{l})}{\cos \phi \cdot \cos \theta}$$

(23)

where $f$ is focal length, $l$ is the distance between blobs, and $d$, the distance between blobs in image plane, is given as

$$d = \sqrt{(u_2 - u_1)^2 + (v_2 - v_1)^2}.$$

Control along x and y motions can be achieved using control signals $u_2$ and $u_3$. When the desired x and y positions are the location of the landing pad, the following control signals can be used.

$$u_2 = K_p \cdot e_x + K_d \cdot \dot{e}_x$$

$$u_3 = K_p \cdot e_y + K_d \cdot \dot{e}_y$$

(24)

where $e_x = \frac{\sum u_i}{4} - O_x$ and $e_y = \frac{\sum v_i}{4} - O_y$ are the position errors on the image plane. In this equation the first
terms are the pattern centroid locations on the image plane, \( O_x \) and \( O_y \) are the camera center coordinates. Controllers will be working to drive the error to zero therefore, bringing the vehicle above the landing pad.

The yaw control can be achieved using the following control signal,

\[
u_4 = K_{p2} \cdot \left( \phi_d - a \cos \left( \frac{u_2 - u_1}{\sqrt{(u_2 - u_1)^2 + (v_2 - v_1)^2}} \right) \right)
\]  \hspace{1cm} (25)

where \( K_{p2} \) is the proportional constant, \( \phi_d \) is the desired yaw angle.

The controllers proposed in Equations 24, 25 and 26 have been implemented on a full dynamical quadrotor model in MATLAB Simulink simulation. In this simulation shown in Figure 6, image features located on ground projected to flying camera coordinates and the corresponding image values used as inputs for the controllers. Helicopter starts at point \((2,0,2)\) meters and the desired hover position is \((0,0,1)\) meters just above the landing pad. Controllers along x and y axes by controlling pitch and roll angles respectively. The mission of this controller is to control altitude of the helicopter by actuating thrusts and moves helicopter in x and y axes by controlling pitch and roll angles respectively. The controller we designed also controls the yaw angle of the helicopter independently. We simulated the system in MATLAB and showed the results. Finally we presented vision based control of the quadrotor and showed that the quadrotor with an on board camera can see the pattern on the ground and hover over on that pattern at a desired altitude.

**ACKNOWLEDGMENTS**

The authors would like to thank Prof. Levent Güvenç for his guidance and to the reviewers for their helpful and motivating comments.

This work is partially supported by HAGU project funded by DPT.

**REFERENCES**


