A study on the modeling of a hexacopter

2 authors, including:

Le Dang Khanh
Vietnam Maritime University
11 PUBLICATIONS   **35** CITATIONS

A study on the modeling of a hexacopter

Dang-Khanh Le[1] · Taek-Kun Nam[†]

Abstract: The purpose of this paper is to present the basic mathematical modeling of a hexacopter, which could be used to develop proper methods for stabilization and trajectory control. A hexacopter consists of six rotors with three pairs of counter-rotating fixed-pitch blades. This mechanism is an under-actuated, dynamically unstable, six-degrees-of-freedom system. The whole motion of this object consists of translational and rotational motion in three dimensions, where the translational motion is created by changing the direction and magnitude of the upward propeller thrust. The hexacopter is controlled by adjusting the angular velocities of the rotors, which are spun by electric motors. It is assumed to be a rigid body; thus, the differential equation of the hexacopter dynamics can be derived from the Newton‑Euler equation. The Euler-angle parametrization of the three-dimensional rotations contains singular points in the coordinate space that can cause failure of both the dynamical model and control. In order to avoid singularities, the rotations of the hexacopter are parametrized in terms of quaternions. This choice has been made considering the linearity of the quaternion formulation and their stability and efficiency. Further, control simulation of a hexacopter applying cascaded-PID control is also presented in this paper.

Keywords: Hexacopter, Newton-Euler model, Inertial moment, Gyroscopic, Cascaded PIDs.

1. Introduction

Currently, the design of multicopters with more than four rotors, i.e., a hexacopter and an octocopter, is developing thanks to increases in the total payload and the possibility of managing one or more engine failures [1][5][6]. In particular, the advantages of more rotors include more power, more lift, and greater safety in the case where one or two motors fail during flying.

In this work, a hexacopter is considered, whose six rotors are located at the vertices of a hexagon and are equidistant from the center of gravity; moreover, the propulsion system consists of three pairs of counter-rotating fixed-pitch propellers. In order to characterize the dynamic aircraft behavior, a mathematical model of the hexacopter is presented by considering all of its external and internal influences. It is well known that, assuming the hexacopter is a rigid body, the differential equations describing its dynamic behavior can be derived from the Newton‑Euler model equation, leading to equivalent mathematical models. Usually, the Euler-angle parameterization is used to describe the rotation of the hexacopter, but this formulation suffers from the presence of singular points in the coordinate space–the so-called gimbal lock–

that can cause failure of both the dynamical model and UAV control. These singularities are not present if the three-dimensional rotations are parameterized in terms of quaternions [2], [3]. The strength of the quaternions depends on the linearity of their formulation, on the easiness of their algebraic structure, and overall, on their stability and efficiency. Thus, the quaternion parameterization is proposed in this paper to obtain a mathematical model of a hexacopter, which allows for a more efficient and fast algorithm implementation for control.

The remainder of this paper is organized as follows. Section 2 presents the coordination system for hexacopter modeling. Section 3 presents the mathematical model of the hexacopter. Control simulation results are given in Section 4, and Section 5 concludes this work.

2. The modeling coordination system and calculus

2.1 Modeling coordination

To keep track of the hexacopter, two different coordinate frames are used to represent the position and orientation in three dimensions–the earth frame (Oxyz) and the body frame

† Corresponding Author (ORCID: http://orcid.org/0000-0003-0432-3504): Division of Marine Engineering, Mokpo National Maritime University 530-729, Korea, E-mail: tknam@mmu.ac.kr, Tel: 061-240-7225

1 Department of Marine Engineering, Mokpo National Maritime University, E-mail: ledangkhanhmtb@gmail.com, Tel: 061-240-7225

($O_bX_bY_bZ_b$), as seen in **Figure 1**. The earth frame is a fixed frame used as an unmoving reference. For example, if the user wants to define a path that the hexacopter should follow, then that path would be represented in the earth frame. The body frame axes are aligned with the sensors, which is convenient when reading sensor data and controlling the angular orientation (attitude). As seen in **Figure 1**, the body frame in this configuration is defined as having the X axis lie between the arms of motor 1 and motor 6, the Y axis lie along the arms of motor 2, and the Z axis pointing upward. The value represents the distance from a given motor to the axis of rotation and should be the same for every motor. The x axis is assumed to be the positive forward direction for vehicle movement. For clarity, the rotation conventions are shown in **Figure 2**.
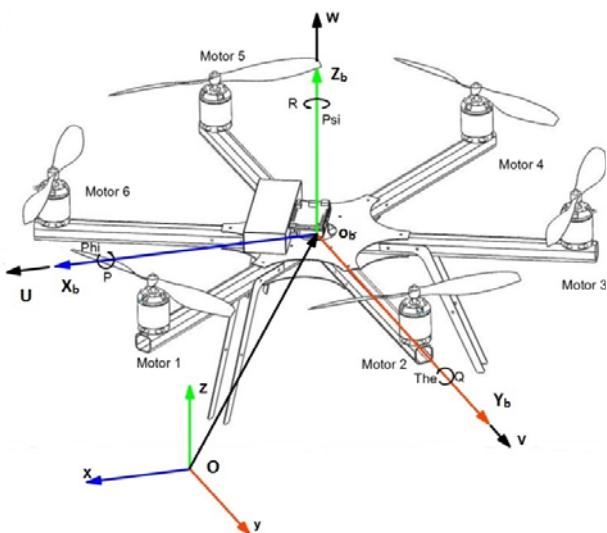

Figure 1: System coordination
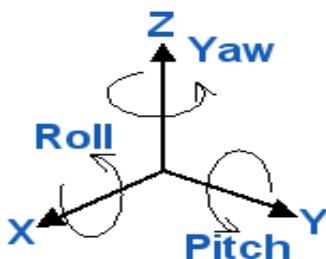
Figure 2: Axis labels and conventions

The angular position of the body frame with respect to the inertial one is usually defined by means of the Euler angles: the roll ϕ, pitch θ and yaw ψ angles. According to the aerospace rotation sequence, the rotation of an aircraft is described as a rotation about the z axis (yaw) and then a rotation about the y axis (pitch) followed by a rotation about the x axis (roll). Each rotation is made on the basis of a right-handed system and in a single plane.

For the sake of simplicity, let us denote the inertial position vector and the Euler angle vector by means of ξ and η, respectively.

$$\xi = \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \qquad \eta = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} \tag{1}$$

In the body frame, the linear velocities are determined by V_B and the angular velocities by υ, respectively.

$$V_B = \begin{bmatrix} V_{x,B} \\ V_{y,B} \\ V_{z,B} \end{bmatrix}, \qquad \upsilon = \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{2}$$

Using these three rotations, a composite rotation matrix can be created, which can transform the motion of the aircraft from the body frame to a new reference frame. The resulting rotation matrix transforms the rotations from the body frame with respect to the inertial frame and can be found using matrix multiplication. In **Equation (3)**, s and c represent the sine and cosine functions, respectively; R is the rotation matrix that transforms the rotations from the body frame to the inertial frame.

$$u^b = \begin{bmatrix} c\theta c\psi & c\psi s\theta s\phi - c\phi s\psi & c\phi c\psi s\theta + s\phi s\psi \\ c\theta s\psi & c\phi c\psi + s\theta s\psi s\phi & c\phi s\theta s\psi - c\psi s\phi \\ -s\theta & c\theta s\phi & c\theta c\phi \end{bmatrix} u^i \tag{3}$$

$$= Ru^i$$

As a consequence, the transformation matrix from the inertial frame to the body frame is $R^{-1} = R^T$.

As shown in **[1]**, the transformation matrix for the angular velocities from the inertial frame to the body frame is W_η.

$$W_\eta = \begin{bmatrix} 1 & 0 & -s\theta \\ 0 & c\phi & c\theta s\phi \\ 0 & -s\phi & c\theta c\phi \end{bmatrix} \tag{4}$$

Further, the transformation matrix from the body frame to the inertial frame is W_η^{-1}.

$$W_\eta^{-1} = \begin{bmatrix} 1 & s\phi t\theta & c\phi t\theta \\ 0 & c\phi & -s\phi \\ 0 & -s\phi/c\theta & c\phi/c\theta \end{bmatrix} \tag{5}$$

Then the transformation laws are

$$\upsilon = W_\eta \dot{\eta} \text{ and } \dot{\eta} = W_\eta^{-1} \upsilon, \tag{6}$$

2.2 Quaternions

In the above section, it is pointed out that the rotation of a rigid body in space could be represented by Euler angles;

however, the singularity of the transformation laws in **Equation (6)** leads to the adoption of a new parameterization–the quaternion–with the aim of describing the orientation of the aircraft with respect to the fixed earth frame **[2][3]**.

The quaternion representation is based on Euler's rotation theorem, which states that any rigid body displacement, where a point is fixed, is equivalent to a rotation. Therefore, if α is the rotation angle about the unit vector u, it is possible to define a quaternion as

$$q_u = (\cos\frac{\alpha}{2}, \sin\frac{\alpha}{2}u) \tag{7}$$

whose components describe the three-degree orientation. Unlike the Euler angles, quaternion rotations do not require a set of predefined rotation axes because they can change its single axis continuously. Owing to the fact that the method of rotation around an arbitrary direction has only one axis of rotation, the degrees of freedom cannot be lost; therefore, a gimbal lock cannot occur. Otherwise, a quaternion can be defined by four components describing a three-dimensional rotation. Thus, the quaternion elements must satisfy a constraint equation called the normality condition, i.e.,

$$q_u = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} \tag{8}$$

$$q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1 \tag{9}$$

where $q_i = q_i(t)$ is assumed for the sake of readability. The transformation of the translational velocity representation from the body frame to the inertial one can be expressed by

$$\dot{\xi} = QV_B \tag{10}$$

where Q is the following matrix:

$$Q = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 - q_0q_3) & 2(q_0q_2 + q_1q_3) \\ 2(q_1q_2 + q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_0q_1 + q_2q_3) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \tag{11}$$

As for the matrix R, Q is orthogonal; therefore, $Q^{-1} = Q^T$. For the angular velocities, the involved transformation can be written as

$$\dot{q}_u = Sv \tag{12}$$

where the matrix S depends on quaternion components as follows:

$$S = \frac{1}{2}\begin{bmatrix} -q_1 & -q_2 & -q_3 \\ q_0 & -q_3 & q_2 \\ q_3 & q_0 & -q_1 \\ -q_2 & q_1 & q_0 \end{bmatrix} \tag{13}$$

On the other hand, it is possible to consider the transformation matrix depending on the angular velocity components, thereby obtaining the link between quaternions and their derivatives with respect to time, which are

$$\begin{bmatrix} \dot{q}_0 \\ \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} = \frac{1}{2}\begin{bmatrix} 0 & -p & -q & -r \\ p & 0 & q_2 & -q \\ q & -r & 0 & p \\ r & q & -p & 0 \end{bmatrix}\begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} \tag{14}$$

In conclusion, it is remarked that the advantage of considering the quaternion reference is twofold because it avoids critical positions, and, thanks to the linearity of the coefficients of the transformation matrix, it is also numerically more efficient and stable compared to traditional rotation formulation.

2.3 Mass moment of inertia matrix

The mass moment of inertia of an object (J) plays a similar role in the rotational motion to the role that mass plays in translational motion: the mass moment of inertia determines how the rotational velocity is affected by the applied torque. This of course depends not only on the mass of the object but also on how the mass is distributed around the rotation axis. It is important to note here that the hexacopter is assumed to be perfectly symmetric about the x y and z axes and to have its center of mass at the geometric center of the arms. With these assumptions, the matrix J_b in **Equation (15)** becomes a diagonal matrix (note that this is related to our choice of the x- and y-axis positions). The J_{xx} and J_{yy} terms are also taken to be identical owing to this symmetry.

$$J_b = \begin{bmatrix} J_{xx} & 0 & 0 \\ 0 & J_{yy} & 0 \\ 0 & 0 & J_{zz} \end{bmatrix} \tag{15}$$

2.3 Thrust

The motors' thrust is the driving force behind all hexacopter maneuvers and is thus integral to control design and simulation. The thrust, T_S, provided by a single motor/propeller system can be calculated as **Equation (16)**

$$T_s = C_T\bar{\omega}^2 \tag{16}$$

where C_T is the lumped parameter thrust coefficient that pertains to the individual motor/prop system, and $\bar{\omega}$ is the an-

gular velocity of the rotor. The total thrust T provided by the motor/prop system is a force perpendicular to the X-Y plane of the body frame in the positive Z direction and is formulated by **Equation (17)**

$$T = C_T \sum_{i=1}^{6} \overline{\omega}_i^2 \tag{17}$$

2.4 Torque

In theory, an increase/decrease in the speed of the six rotors independently will create torques around the x y and z axes, thereby creating roll, pitch, and yaw rotations. The torque is force multiplied by a distance, and the rotors will affect the total rotation about a certain axis differently, depending on the distance from the center of gravity. **Figure 3** shows the lengths and angles of the arms to the relative distance from the center of gravity, which is the distance from the rotor to the axis of rotation.
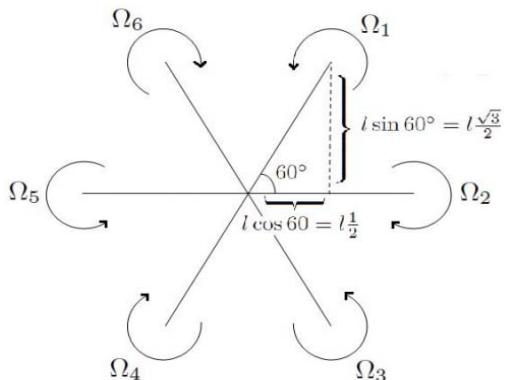

Figure 3: Hexacopter rotor distances to center of gravity

By decreasing $\omega_1, \omega_2, \omega_3$ and increasing $\omega_4, \omega_5, \omega_6$, a positive roll moment is produced as **Equation (18)**

$$\tau_\phi = l\,C_T(-\omega_2^2 + \omega_5^2 + \frac{1}{2}(-\omega_1^2 - \omega_3^2 + \omega_4^2 + \omega_6^2)) \tag{18}$$

By decreasing ω_1, ω_6 and increasing ω_3, ω_4, a positive roll moment is produced as **Equation (19)**.

$$\tau_\theta = l\,C_T \frac{\sqrt{3}}{2}(-\omega_1^2 + \omega_3^2 + \omega_4^2 - \omega_6^2) \tag{19}$$

In order to understand the effect of the motor on the yaw, the torque force of the motor/prop system must also be determined and can be done in a similar fashion to that of the thrust tests. The related lumped parameter is **Equation (20)**

$$\tau = C_\tau \overline{\omega}^2 \tag{20}$$

where τ is the torque created by the motor, and C_τ is the tor-

que coefficient for the motor/prop system. This torque provides a force that acts to yaw the system about the z axis.

By decreasing $\omega_1, \omega_3, \omega_5$ and increasing $\omega_2, \omega_4, \omega_6$, a positive roll moment is produced as **Equation (21)**.

$$\tau_\psi = C_\tau(-\omega_1^2 + \omega_2^2 - \omega_3^2 + \omega_4^2 - \omega_5^2 + \omega_6^2) \tag{21}$$

The rotation of the propellers produces a gyroscopic effect that is expressed as **Equation (22)**.

$$\tau_{gyro} = \begin{bmatrix} -J_r \dot{q}\omega_r \\ J_r \dot{p}\omega_r \\ 0 \end{bmatrix} \tag{22}$$

where J_r is the rotational inertia of the propeller, and $\omega_r = -\omega_1 + \omega_2 - \omega_3 + \omega_4 - \omega_5 + \omega_6$ is the overall propeller speed.

3. Mathematical model of the hexacopter

As a basis for simulation, estimation, and control, a model describing the hexacopter and its dynamics is developed. For this, the well-known Newton-Euler formalism is used to describe the dynamics of a rigid body affected by external forces and torques. The motion of a rigid body can be decomposed into translational and rotational components.

3.1 Translational dynamics

The force acting on the hexacopter is provided in the body frame, and the force required for the acceleration of mass and the centrifugal force are equal to the gravity and the total thrust of the rotors as follows: **Equation (23)**.

$$m \dot{V}_B + v \times (m\,V_B) = R^T G + T \tag{23}$$

where $R^T G$ is the gravitational force acting in the body frame, and $G = [0\,0\,-g]^T$ is gravitational force in the inertial frame.

In the inertial frame, the centrifugal force is nullified. Thus, only the gravitational force and the magnitude and direction of the thrust contribute to the acceleration of the hexacopter as **Equation (24)**.

$$m\ddot{\xi} = G + R^T T \tag{24}$$

From **Equations (1)**, **(3)**, and **(17)**, **Equation (24)** is equivalent to **Equation (25)**.

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = -g \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} + \frac{T}{m} \begin{bmatrix} c\psi s\theta c\phi + s\psi s\phi \\ s\psi s\theta c\phi - s\psi s\phi \\ c\theta c\phi \end{bmatrix} \tag{25}$$

or in quartation form

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = -g \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} + \frac{T}{m} \begin{bmatrix} 2(q_0q_2+q_1q_3) \\ 2(q_2q_3-q_0q_1) \\ q_0^2-q_1^2-q_2^2+q_3^2 \end{bmatrix} \quad (26)$$

3.2 Rotational dynamics

In the body frame, the external torque, $\tau_B = \begin{bmatrix} \tau_\phi\,\tau_\theta\,\tau_\psi \end{bmatrix}^T$, is equal to the torques due to the angular acceleration of the inertia, $J_b\dot{v}$; the centripetal forces, $v \times (J_b v)$; and the gyroscopic forces, τ_{gyro} as **Equation (27)**.

$$J_b\dot{v} + v \times (J_b v) + J_{gyro} = \tau_B \quad (27)$$

After some algebra, the following is obtained:

$$\dot{v} = J_b^{-1}(-\begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} J_{xx}p \\ J_{yy}q \\ J_{zz}r \end{bmatrix} - J_r \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \omega_r + \tau_B) \quad (28)$$

Equation (28) can be written as **Equation (29)**

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} (J_{yy}-J_{zz})qr/J_{xx} \\ (J_{zz}-J_{xx})pr/J_{yy} \\ (J_{xx}-J_{yy})pq/J_{zz} \end{bmatrix} - J_r \begin{bmatrix} q/J_{xx} \\ -p/J_{yy} \\ 0 \end{bmatrix} \overline{\omega} + \begin{bmatrix} \tau_\phi/J_{xx} \\ \tau_\theta/J_{yy} \\ \tau_\psi/J_{zz} \end{bmatrix} \quad (29)$$

Once the angular velocity has been evaluated, the angular acceleration in the inertial frame can be easily deduced as

$$\ddot{q}_u = \frac{d}{dt}(Sv) \quad (30)$$

Finally, by combining **Equation (26)** and **Equation (30)**, the differential dynamics equation of the hexacopter system can be obtained as

$$\begin{cases} \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = -g \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} + \frac{T}{m} \begin{bmatrix} 2(q_0q_2+q_1q_3) \\ 2(q_2q_3-q_0q_1) \\ q_0^2-q_1^2-q_2^2+q_3^2 \end{bmatrix} \\ \ddot{q}_u = \frac{d}{dt}(Sv) \end{cases} \quad (31)$$

4. Control simulation

In this section, a typical PID controller tuned by an available tool in Matlab **[4][7]** will be used to control the above hexacopter model to achieve the control purpose of reaching a given set point.

4.1 Altitude control

The goal of the altitude controller is to keep the hexacopter at a reference value by using altitude and altitude-velocity measurements. This is an attitude-command-only system. In other words, there is no control system to track

the position. Instead, the controller only tries to track the attitude (ϕ, θ, ψ) and altitude (z) commands using PID controllers.
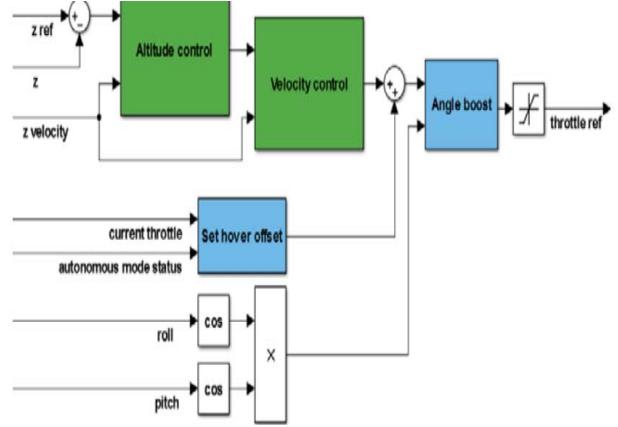
**Figure 4:** Altitude controller

A cascaded Position → Velocity PID control structure was implemented according to **Figure 4**, where the outer loop used a PID controller in which the controller's output signal was calculated on the basis of the altitude reference, altitude measurements, and velocity measurement as **Equation (32)**.

$$u_z(t) = K_p e_z(t) + K_i \int_0^t e_z(\tau)d\tau - K_d \frac{d}{dt}z(t) \quad (32)$$
$$e_z(t) = z_{ref}(t) - z(t)$$

This desired velocity was used by the inner loop, where a PI controller calculated the required change in the throttle as **Equation (33)**.

$$u_{\dot{z}}(t) = K_p e_{\dot{z}}(t) + K_i \int_0^t e_{\dot{z}}(\tau)d\tau \quad (33)$$
$$e_{\dot{z}}(t) = \dot{z}_{ref}(t) - \dot{z}(t)$$

The simulation conditions are summarized in **Tables 1-3**. The reference signal is listed in **Table 1**, in which (ϕ, θ, ψ) are in units of degrees and the altitude (z) in reported in meters. The initial conditions of given system and the PID controllers' parameters **[8][9]** are listed in **Table 2**. **Table 3** summarizes the parameters of a certain model of a hexacopter.

Table 1: Reference signals

	Step time	Initial value	Final value
ϕ	10	0	-10
θ	20	0	10
ψ	30	0	45
z	0	0	8

Dang-Khanh Le · Taek-Kun Nam
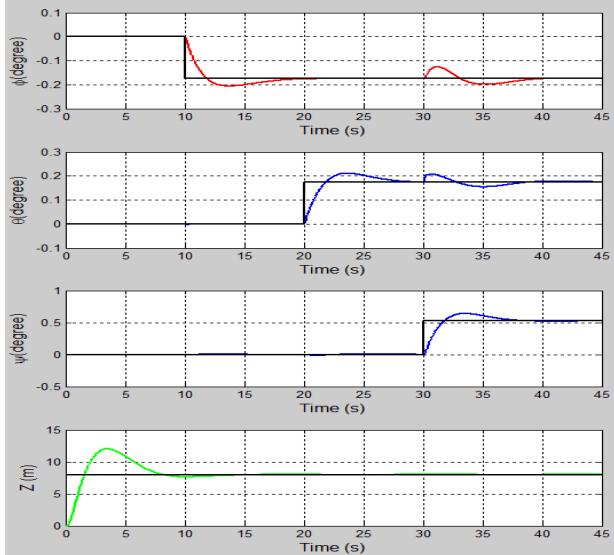
Table 2: Initial conditions and PIDs parameters

$\phi = \theta = \psi$	0^0	$K_{P\psi}$	3
ω_i	0	$K_{I\psi}$	1
$K_{P\phi}$	3	$K_{D\psi}$	3.5
$K_{I\phi}$	1	K_{Pz}	3
$K_{D\phi}$	3.3	K_{Iz}	1
$K_{P\theta}$	3	K_{Dz}	3.3
$K_{I\theta}$	1.2	x = y = z	0(m)
$K_{D\theta}$	3.5		

Table 3: Hexacopter's information
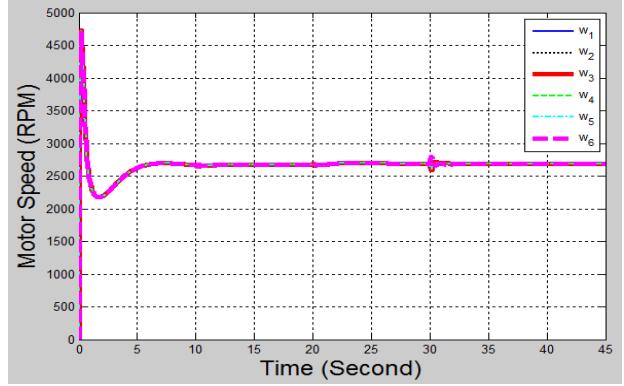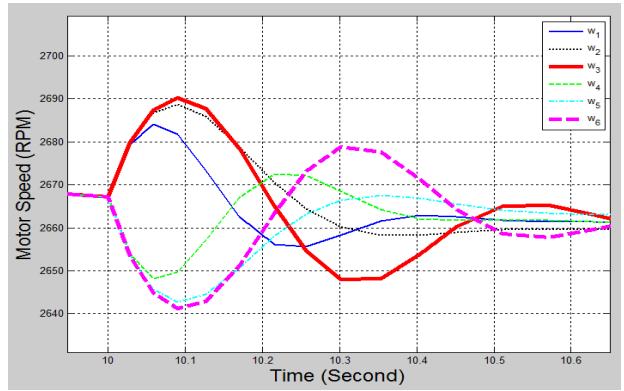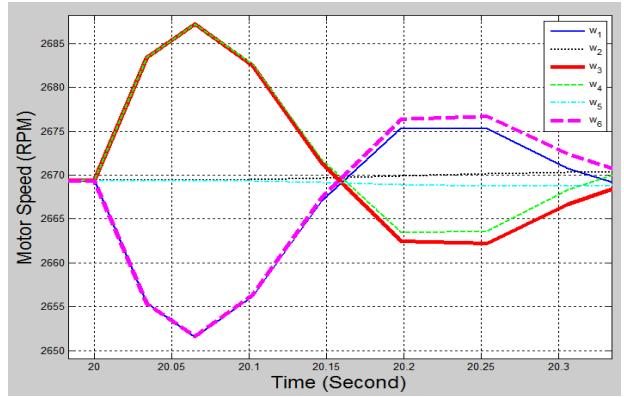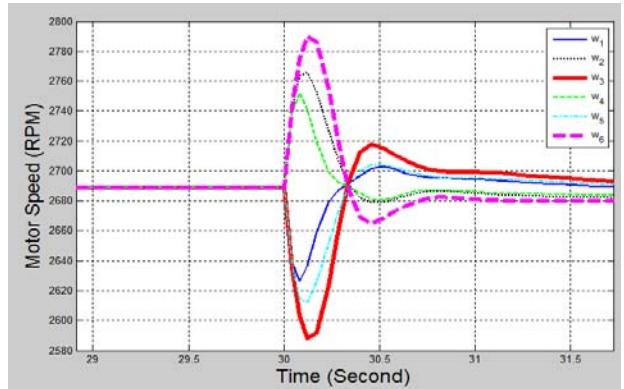
C_T	1.4865e-06	N/rpm^2
C_τ	2.925e-07	Nm/rpm^2
g	9.98	m/s^2
m	6.38	kg
I_{xx}	0.14822	kgm^2
I_{yy}	0.053208	kgm^2
I_{zz}	0.29239	kgm^2
l	0.3	m
J_r	3.357e-05	kgm^2

The simulation results for altitude control are shown in **Figure 5**.

Figure 5: Altitude control results

It is obvious that all altitude control purposes were achieved by the proposed PID controllers. All parameters consisted of roll, pitch, yaw, and the high flight of the hexacopter reached the set points after a controlling time. **Figure 6** shows the reaction of the motors during the simulation time. At 0 s, the speeds of all motors increased at same rate to generate the required thrust to lift hexacopter from the initial position of 0 m to a set point of 8 m. At 10 s, $\omega_1, \omega_2, \omega_3$ increased, and $\omega_4, \omega_5, \omega_6$ decreased to produce a negative roll moment, which decreased the roll to a set point of -10°, as seen in **Figure 7**.

Similarly, **Figures 8** and **9** show the behavior of the motors at points where the pitch and yaw and changed.

Figure 6: Speed of motors

Figure 7: Speed of motors when changing roll

Figure 8: Speed of motors when changing pitch

Figure 9: Speed of motors when changing yaw

4.2 Horizontal control

The horizontal controller used the roll and pitch angles to change the horizontal position and horizontal velocity.
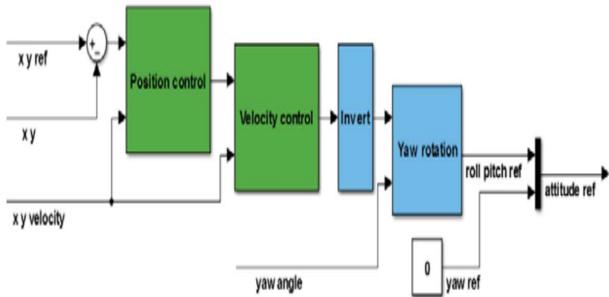
Figure 10: Horizontal controller

Similar to altitude control, a cascaded Position → Velocity controller was used again, as seen in **Figure 10**. This time, the outer position loop used a PID controller that outputs a desired velocity on the basis of the current position error and velocity measurement as **Equation (34)**.

$$\begin{bmatrix} u_x(t) \\ u_y(t) \end{bmatrix} = K_p e_z(t) + K_i \int_0^t e_z(\tau)d\tau - K_d \frac{d}{dt} y_{xy}(t) \qquad (34)$$

$$e_z(t) = \begin{bmatrix} x_{ref}(t) - x(t) \\ y_{ref}(t) - y(t) \end{bmatrix}, \; y_{xy}(t) = \begin{bmatrix} x(t) \\ y(t) \end{bmatrix}$$

The inner velocity loop uses a PD controller to calculate the desired roll and pitch angles from the velocity error using **Equation (25)**. Here, the PD controller is used for the sensitive response to the possible future values of the error based on its current rate of change.

$$\begin{bmatrix} u_{\dot{x}}(t) \\ u_{\dot{y}}(t) \end{bmatrix} = K_p e_{\dot{z}}(t) + K_D \frac{d}{dt} e_{\dot{z}}(t) \qquad (35)$$

$$e_{\dot{z}}(t) = \begin{bmatrix} \dot{x}_{ref}(t) - \dot{x}(t) \\ \dot{y}_{ref}(t) - \dot{y}(t) \end{bmatrix}.$$

The position and velocity control utilizes the earth frame; thus, the roll and pitch set points have to be rotated by the yaw rotation matrix because the hexacopter operates in the body frame.

In this simulation case, the information of the hexacopter model and the initial conditions were the same as previous simulation work. The reference signals in this simulation were a point in the inertial frame, which is summarized in **Table 4**. **Table 5** summarizes the parameters of the PID controllers, which are proposed for this case of horizontal control. The parameters applied to the inner-loop PD controllers are listed in **Table 6**.

Table 4: Reference point information

	Step time	Initial value	Final value
x	0	0	10
y	0	0	5
z	0	0	5

Table 5: outer PIDs parameters

$K_{P\phi}$	6	$K_{P\psi}$	6
$K_{I\phi}$	0.8	$K_{I\psi}$	0.8
$K_{I\phi}$	3.5	$K_{I\psi}$	3.5
$K_{I\theta}$	6	K_{Pz}	4.8
$K_{I\theta}$	0.8	K_{Iz}	1.4
$K_{D\theta}$	3.5	K_{Iz}	4.3

Table 6: Inner PD parameters

K_{Px}	0.2	K_{Py}	0.3
K_{Dx}	0.1	K_{Dy}	0.2

Figure 11 shows the tracking reference-point simulation results. After 10 s, the hexacopter almost reached the set point (10, 5, 5) in the inertial frame, meaning that the tracking task was achieved. **Figure 12** shows the tracking error during the simulation time. The tracking error decreased with time and almost reached zero around 10 s when the tracking task was achieved.
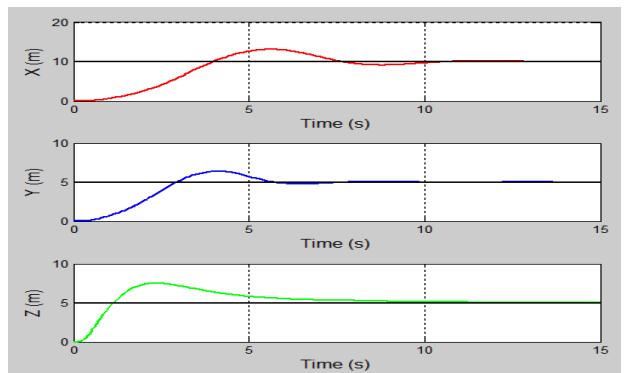

Figure 11: Tracking of reference-point
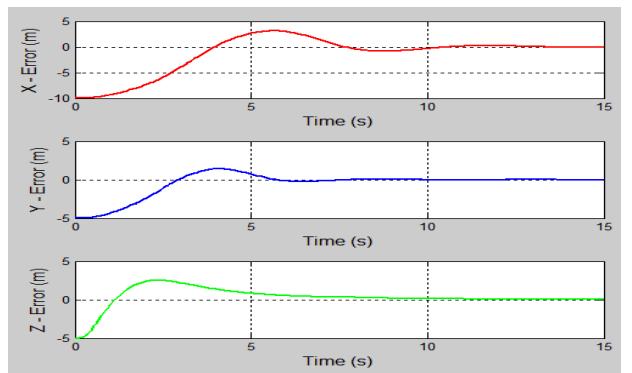
Figure 12: Tracking error

The behavior of the motors during tracking is shown in **Figure 13**. From the initial point, the horizontal controller sent a command to increase the speeds of all six motors to generate thrust to lift the hexacopter and reach the set point. The hexacopter changed its Euler angles to move from the initial point to the set point. **Figure 14** shows the details of the motors' reaction. In this figure, the hexacopter changes its roll angle by increasing or decreasing the speed of motors 1, 2, or 3 and motors 4, 5, or 6 respectively. These simulation results demonstrated that horizontal control is achieved with the mentioned PD and PID controllers.
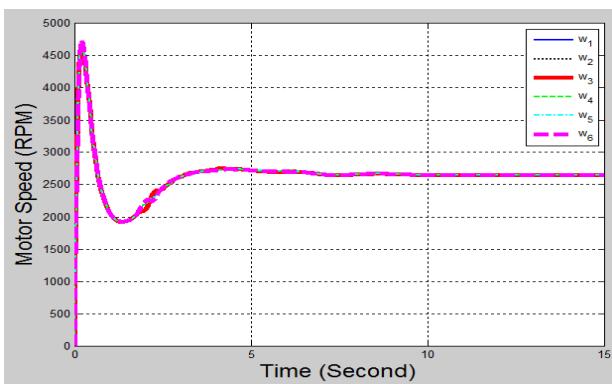

**Figure 13:** Speed of motors during trajectory tracking
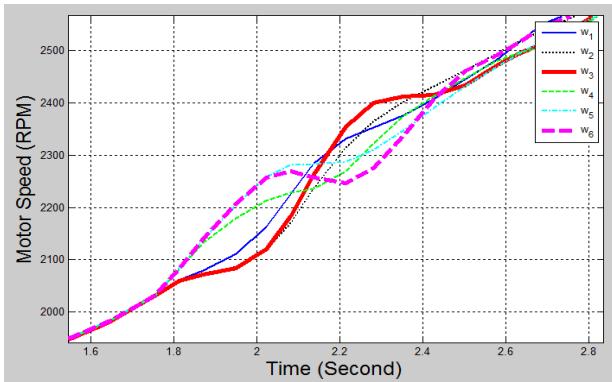
Figure 14: React of motors in some interval during tracking time

5. Conclusion

In this paper, a mathematical model of a hexacopter has been presented. The equations of motion have been defined by quaternions since, unlike the Euler angles, they do not suffer from a gimbal lock. They are also more efficient in terms of numerical computation. In order to avoid singularities, the rotations of the hexacopter were parametrized in terms of quaternions. All of the equations for the force and moment calculations were mentioned in detail. The equations of motion has been defined on the basis of the Newton‑Euler model. Altitude and horizontal controllers were proposed, in which the PD and PID controllers were implemented to generate control

signals. A Matlab Simulink program was created for the simulation work. The simulation results for both altitude and horizontal control were presented and analyzed in detail.

Future work will extend in several directions. The model presented in this paper is a simplification of more complex dynamics; indeed, aerodynamic effects have been neglected. Thus, the next step will involve improvements in the hexacopter model with more realistic features and the application of more accurate control laws in order to be applied to real flights.

References

[1] J. Fogelberg Navigation and Autonomous Control of a Hexacopter in Indoor Environments, Lund University, ISSN 0280-5316, 2013.

[2] J. M. Rico-Martinez and J. Gallardo-Alvarado, "A simple method for the determination of angular velocity and acceleration of a spherical motion through quaternions," Meccanica 35, pp. 111‑118, 2000.

[3] A. P. Yefremov, Quaternions: Algebra, Geometry and Physical Theories, Hypercomplex Numbers in Geometry and Physics 1, 2004.

[4] N. Ohlsson and M. Stahl, A Model-Based Approach to Computer Vision and Automatic Control using Matlab Simulink for an Autonomous Indoor Multirotor UAV, Department of Signals and Systems, CHALMERS UNIVERSITY OF TECHNOLOGY, Sweden, 2013.

[5] T. Luukkonen, Modelling and control of quadcopter, Aalto University, School of Science, 2011.

[6] B. W. McCormick, Aerodynamics, Aeronautics, and Flight Mechanics, John Wiley & Sons, 1995.

[7] S. S. Shin, J. H. Noh, and J. H. Park, "A study on the optimal tuning of the hydraulic motion driver parameter by using RCGA," Journal of the Korean Society of Marine Engineering, vol. 38, no. 1, pp. 39-47, 2014.

[8] Y. H. Lee, G. G. Jin, and M. O. So, "Level control of single water tank system using fuzzy-PID technique," Journal of the Korean Society of Marine Engineering, vol. 38, no. 5, pp. 550-556, 2014.

[9] S. S. Shin, J. H. Noh, and J. H. Park, "A study on the stabilization and controller design for directional Pan-tilt system," Journal of the Korean Society of Marine Engineering, vol. 37, no. 2, pp. 192-198, 2013.