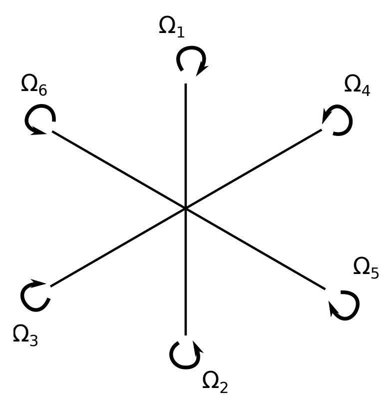
Modelling JDrones Hexacopter with Simulink

Roman Hofstetter

March 16, 2013

1 Model of the Copter

Note that the numbering of the blades is different in this model than in Arducopter.



1.1 Equations

The earth frame E has x to the front, y to the left and z up. First it is turned with ψ around the z-axis, next with θ around the new y-axis and finally with ϕ around the again new x-axis. Over Lagrange II one can get the following equations:

$$\tau_x = b\,l_C \sin(60^\circ)\left(\Omega_3^2 + \Omega_6^2 - \Omega_4^2 - \Omega_5^2\right)$$

$$\tau_y = b\,l_C \left[\Omega_2^2 - \Omega_1^2 + \cos(60^\circ)\left(\Omega_3^2 + \Omega_5^2 - \Omega_4^2 - \Omega_6^2\right)\right]$$

$$\tau_z = d\left(\Omega_1^2 + \Omega_3^2 + \Omega_5^2 - \Omega_2^2 - \Omega_4^2 - \Omega_6^2\right)$$

$$\Omega_r = \Omega_2 + \Omega_4 + \Omega_6 - \Omega_1 - \Omega_3 - \Omega_5$$

$$\ddot{\phi} = \frac{1}{I_{xx}}\left[\dot{\theta}\dot{\psi}(I_{yy} - I_{zz}) - J_r \Omega_r \dot{\theta} + \tau_x\right]$$

$$\ddot{\theta} = \frac{1}{I_{yy}}\left[\dot{\phi}\dot{\psi}(I_{zz} - I_{xx}) + J_r \Omega_r \dot{\phi} + \tau_y\right]$$

$$\ddot{\psi} = \frac{1}{I_{zz}} \left[\dot{\phi}\dot{\theta}(I_{xx} - I_{yy}) + \tau_z \right]$$

$$\ddot{x} = \frac{1}{m_C} \left[[\cos(\psi)\sin(\theta)\cos(\phi) + \sin(\psi)\sin(\phi)]\, b \sum_{i=1}^{6} \Omega_i^2 \right]$$

$$\ddot{y} = \frac{1}{m_C} \left[[\sin(\psi)\sin(\theta)\cos(\phi) - \cos(\psi)\sin(\phi)]\, b \sum_{i=1}^{6} \Omega_i^2 \right]$$

$$\ddot{z} = \frac{1}{m_C} \left[[\cos(\psi)\cos(\phi)]\, d \sum_{i=1}^{6} \Omega_i^2 \right] - g$$

2 Modelling the controller

First the three angles and the translation in z-axis should be stabilized. Looking at these four equations, one can sum up the input of the system as follows:

$$U_1 = b(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2 + \Omega_5^2 + \Omega_6^2) \tag{1}$$

$$U_2 = b \cdot l_C \cdot \frac{\sqrt{3}}{2} \left(\Omega_3^2 - \Omega_4^2 - \Omega_5^2 + \Omega_6^2 \right) \tag{2}$$

$$U_3 = b \cdot l_C \left(-\Omega_1^2 + \Omega_2^2 + \frac{1}{2} \left(\Omega_3^2 - \Omega_4^2 + \Omega_5^2 - \Omega_6^2 \right) \right) \tag{3}$$

$$U_4 = d(\Omega_1^2 - \Omega_2^2 + \Omega_3^2 - \Omega_4^2 + \Omega_5^2 - \Omega_6^2) \tag{4}$$

So now for instance the PID for controlling the pitch angle will return us U_3. To get the commands sent to the ESCs the above equations need to be inverted and solved for each Ω^2. In the Arducopter code this is done like this:

$$roll\_factor = \cos(angle{+}90^\circ) \quad pitch\_factor = \cos(angle) \quad yaw\_factor = direction$$

For the hexa setup this results in the following solution. Note that the equations are written in my model. This means, that my rotor six is Ardus number five and my pitch is positive for forward flight.

$$\begin{bmatrix} \Omega_1^2 \\ \Omega_2^2 \\ \Omega_3^2 \\ \Omega_4^2 \\ \Omega_5^2 \\ \Omega_6^2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & -1 & 1 \\ 1 & 0 & 1 & -1 \\ 1 & \frac{\sqrt{3}}{2} & \frac{1}{2} & 1 \\ 1 & -\frac{\sqrt{3}}{2} & -\frac{1}{2} & -1 \\ 1 & -\frac{\sqrt{3}}{2} & \frac{1}{2} & 1 \\ 1 & \frac{\sqrt{3}}{2} & -\frac{1}{2} & -1 \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} \tag{5}$$