

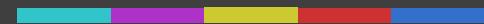
---



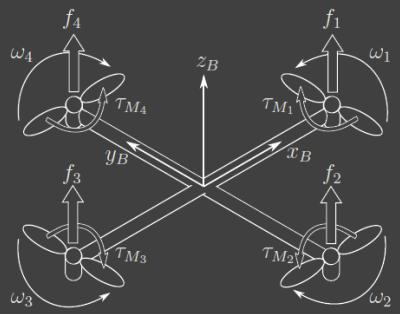
# Modelling and control of Quadcopter

Controlla's

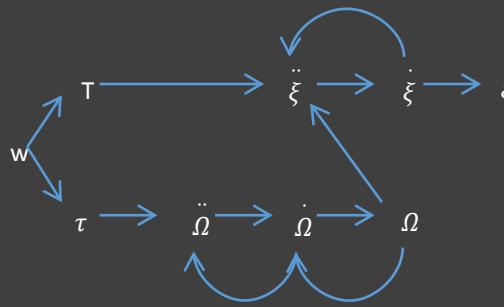
Yoon hyung chul



# Content



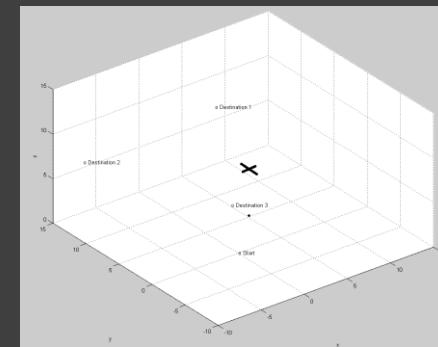
I . The quadcopter  
and interactions  
between states



II. Mathematical  
model of quadcopter



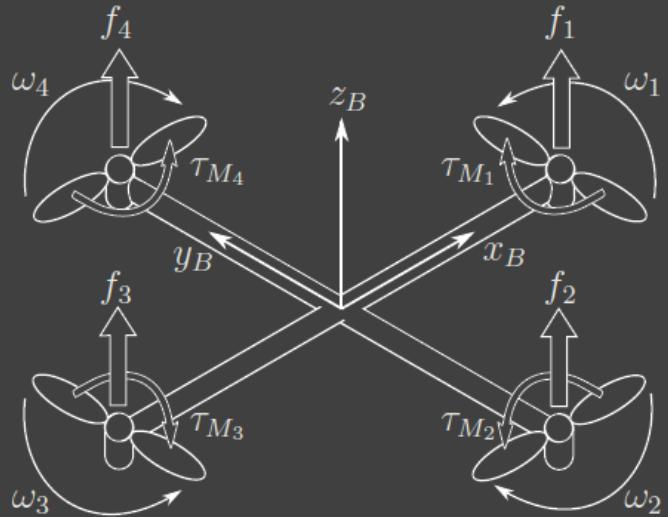
III. Stabilisation  
of quadcopter



VI. Trajectory control



# I . The quadcopter and interactions between states



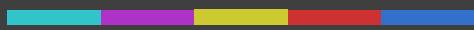
Angular velocities, torques and forces created by the four rotors.

$w_i$  is Rotor's angular velocities  
 $f_i$  is Thrust

\*  $f_i = k * w_i^2$  at  $k$  is lift constant.

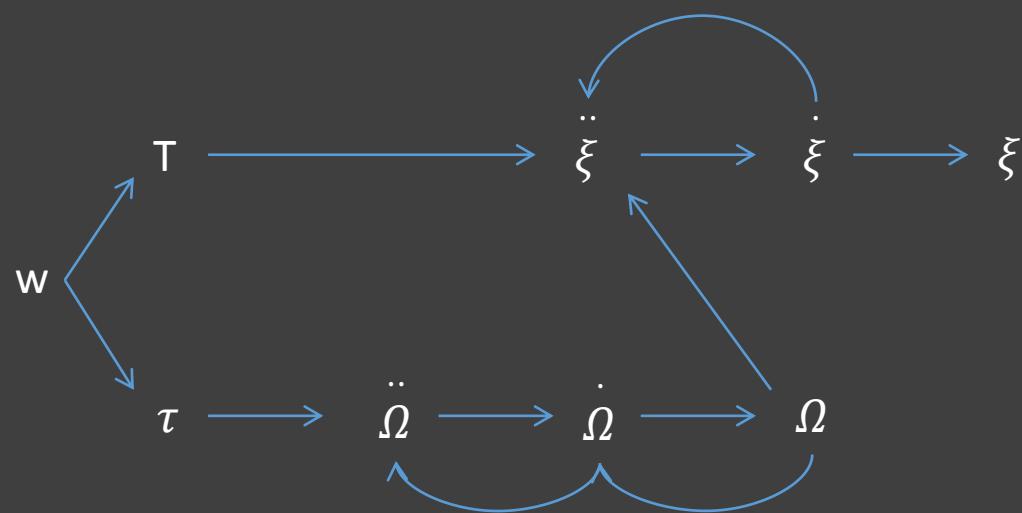
$\tau$  is Torque





# I . The quadcopter and interactions between states

Dynamics flowchart



This is Quadcopter dynamics flowchart.

When the values of the rotor's angular velocities are known, the Thrust and Torque can solved.



## III. Mathematical model of quadcopter

Inertial frame

$$\xi = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

Position

$$\theta = \begin{bmatrix} \phi \\ \theta \\ \varphi \end{bmatrix}$$

Euler angle

Body frame

$$V_B = \begin{bmatrix} v_{x,B} \\ v_{y,B} \\ v_{z,B} \end{bmatrix}$$

$$\Omega = \begin{bmatrix} p \\ q \\ r \end{bmatrix}$$

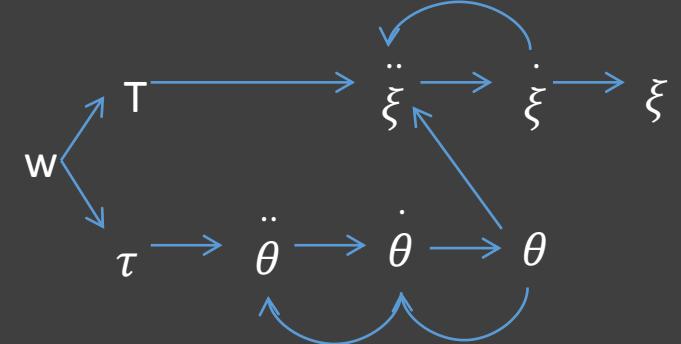
Angular velocities

Rotation matrix R. (roll->pitch->yaw)

$$R = \begin{bmatrix} C_\varphi C_\theta & C_\varphi S_\theta S_\phi - S_\varphi C_\phi & C_\varphi S_\theta C_\phi + S_\varphi S_\phi \\ S_\varphi C_\theta & S_\varphi S_\theta S_\phi + C_\varphi C_\phi & S_\varphi S_\theta C_\phi - C_\varphi S_\phi \\ -S_\theta & C_\theta S_\phi & C_\theta C_\phi \end{bmatrix}$$



## III. Mathematical model of quadcopter



Transformation matrix

$$W_\theta = \begin{bmatrix} 1 & 0 & -S_\theta \\ 0 & C_\phi & C_\theta S_\phi \\ 0 & -S_\phi & C_\theta C_\phi \end{bmatrix}$$

Inertial frame to body frame

$$W_\theta^{-1} = \begin{bmatrix} 1 & S_\phi T_\theta & C_\phi T_\theta \\ 0 & C_\phi & -S_\phi \\ 0 & S_\phi/C_\theta & C_\phi/C_\theta \end{bmatrix}$$

Body frame to Inertial frame

$$\dot{\Omega} = W_\theta \dot{\theta}, \quad \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -S_\theta \\ 0 & C_\phi & C_\theta S_\phi \\ 0 & -S_\phi & C_\phi C_\theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\varphi} \end{bmatrix}$$

Inertial frame to body frame

$$\dot{\theta} = W_\theta^{-1} \dot{\Omega}, \quad \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} 1 & S_\phi T_\theta & C_\phi T_\theta \\ 0 & C_\phi & -S_\phi \\ 0 & S_\phi/C_\theta & C_\phi/C_\theta \end{bmatrix} \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix}$$

Body frame to Inertial frame



## III. Mathematical model of quadcopter

Thrust and Torque by rotor

$$f_i = k * w_i^2$$

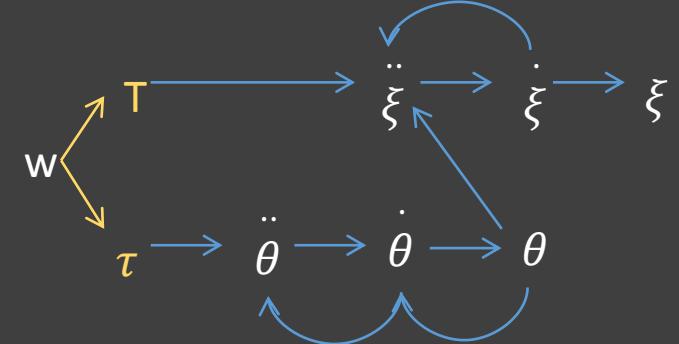
Thrust in the direction of the  
rotor axis.

$$T = \sum_{i=1}^4 f_i = k \sum_{i=1}^4 w_i^2 \quad T^B = \begin{bmatrix} 0 \\ 0 \\ T \end{bmatrix}$$

Thrust in the direction of the  
body z-axis.

$$\tau_B = \begin{bmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\varphi \end{bmatrix} = \begin{bmatrix} Lk(-w_2^2 + w_4^2) \\ Lk(-w_1^2 + w_3^2) \\ b(w_1^2 - w_2^2 + w_3^2 - w_4^2) \end{bmatrix}$$

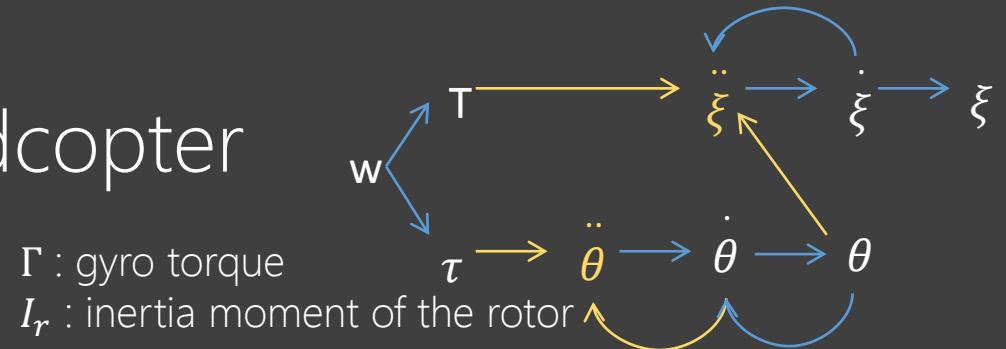
Torque in the direction of  
the corresponding body frame angles





## III. Mathematical model of quadcopter

Newton-Euler equation



Linear accelerations  $m\ddot{V}_B + \Omega \times (mV_B) = R^T G + T_B$

Angular accelerations  $I\ddot{\Omega} + \Omega \times (I\Omega) + \Gamma = \tau$

$$m\ddot{\xi} = G + RT_B$$

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = -g \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} + \frac{T}{m} \begin{bmatrix} C_\phi S_\theta C_\phi + S_\phi S_\theta \\ S_\phi S_\theta C_\phi - C_\phi S_\theta \\ C_\theta C_\phi \end{bmatrix}$$

$$\dot{\Omega} = I^{-1} \left( - \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} I_{xx}p \\ I_{yy}q \\ I_{zz}r \end{bmatrix} - I_r \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} w_F + \begin{bmatrix} \tau_\phi / I_{xx} \\ \tau_\theta / I_{yy} \\ \tau_\varphi / I_{zz} \end{bmatrix} \right)$$

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} (I_{yy} - I_{zz})q \ r / I_{xx} \\ (I_{zz} - I_{xx})p \ r / I_{yy} \\ (I_{xx} - I_{yy})p \ q / I_{zz} \end{bmatrix} - I_r \begin{bmatrix} q / I_{xx} \\ -q / I_{yy} \\ 0 \end{bmatrix} w_F + \begin{bmatrix} \tau_\phi / I_{xx} \\ \tau_\theta / I_{yy} \\ \tau_\varphi / I_{zz} \end{bmatrix}$$



## III. Mathematical model of quadcopter

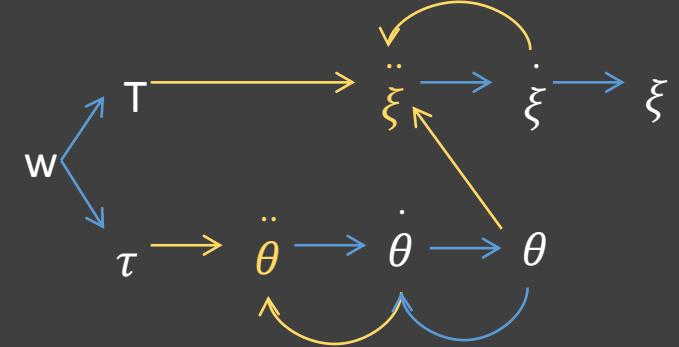
Aerodynamical effects

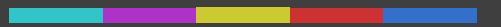
$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = -g \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} + \frac{T}{m} \begin{bmatrix} C_\phi S_\theta C_\phi + S_\phi S_\theta \\ S_\phi S_\theta C_\phi - C_\phi S_\theta \\ C_\theta C_\phi \end{bmatrix}$$

To enforce more realistical behaviour of the quadcopter, drag force generated by the air resistance is included. For example, dependence of thrust on angle of attack, blade flapping and airflow disruptions.

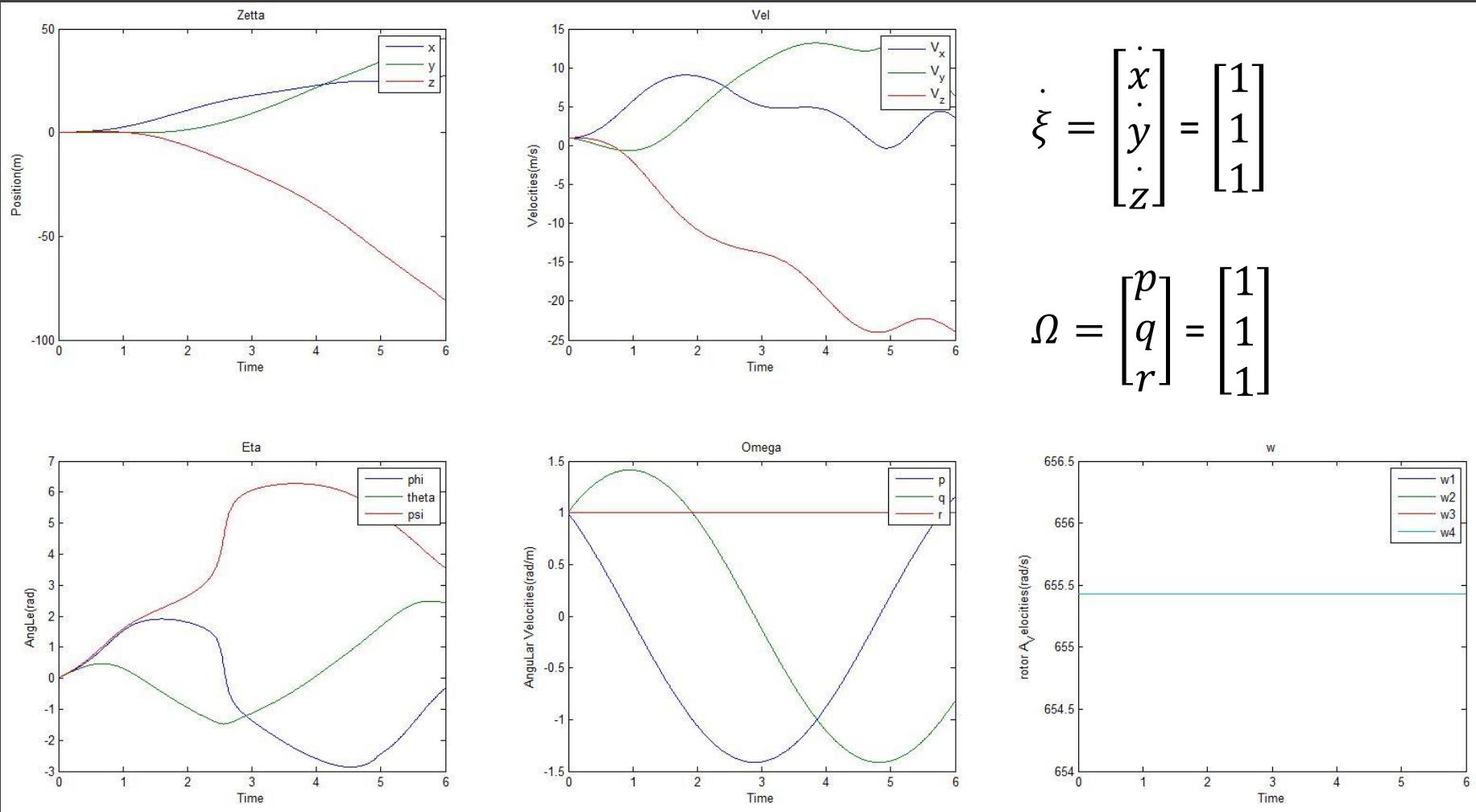
Thus, these effects are excluded from the model and the presented simple model is used.  $A_i$  is drag force coefficients for velocities in the corresponding directions of the inertial frame.

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = -g \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} + \frac{T}{m} \begin{bmatrix} C_\phi S_\theta C_\phi + S_\phi S_\theta \\ S_\phi S_\theta C_\phi - C_\phi S_\theta \\ C_\theta C_\phi \end{bmatrix} - \frac{1}{m} \begin{bmatrix} A_x & 0 & 0 \\ 0 & A_y & 0 \\ 0 & 0 & A_z \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix}$$



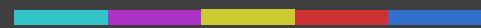


## II. Mathematical model of quadcopter



$$\dot{\xi} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

$$\Omega = \begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$



### III. Stabilisation of quadcopter

PID controller

$$e(t) = x_d(t) - x(t)$$

$$u(t) = K_P e(t) + K_I \int_0^t e(\tau) d\tau + K_D \frac{de(t)}{dt}$$

$x_d$  : destination position

$x$  : present position

$K_P$  : proportional gain

$K_I$  : integral gain

$K_D$  : derivative gain

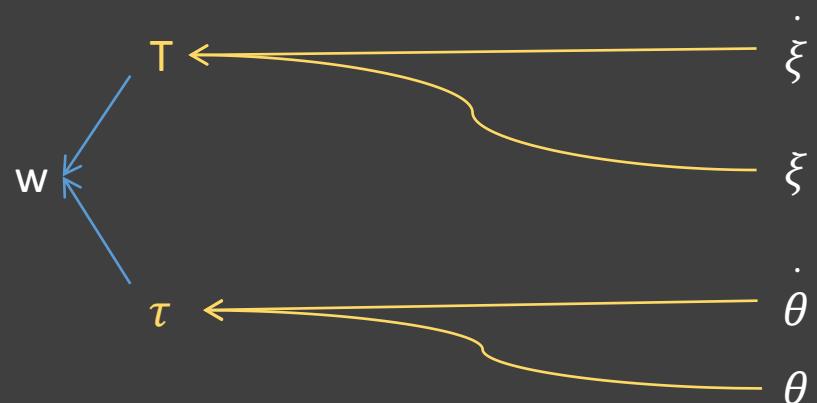
PD controller for the quadcopter

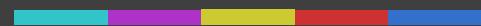
$$T = (g + K_{z,D}(\dot{z}_d - \dot{z}) + K_{z,P}(z_d - z)) \frac{m}{C_\phi C_\theta}$$

$$\tau_\phi = (K_{\phi,D}(\dot{\phi}_d - \dot{\phi}) + K_{\phi,P}(\phi_d - \phi)) I_{xx}$$

$$\tau_\theta = (K_{\theta,D}(\dot{\theta}_d - \dot{\theta}) + K_{\theta,P}(\theta_d - \theta)) I_{yy}$$

$$\tau_\varphi = (K_{\varphi,D}(\dot{\varphi}_d - \dot{\varphi}) + K_{\varphi,P}(\varphi_d - \varphi)) I_{zz}$$



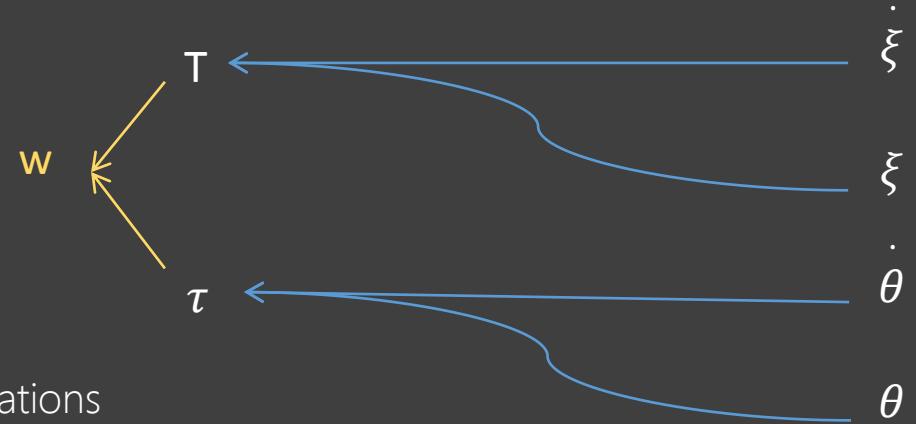


### III. Stabilisation of quadcopter

PD controller for the quadcopter

$$T = k \sum_{i=1}^4 w_i^2 = k(w_1^2 + w_2^2 + w_3^2 + w_4^2)$$

$$\tau_B = \begin{bmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\varphi \end{bmatrix} = \begin{bmatrix} lk(-w_2^2 + w_4^2) \\ lk(-w_1^2 + w_3^2) \\ b(w_1^2 - w_2^2 + w_3^2 - w_4^2) \end{bmatrix}$$



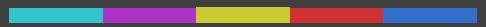
The angular velocities of rotors can be calculated from Equations

$$\begin{bmatrix} \ddot{z} \\ T \\ \tau_B \\ \tau_\varphi \end{bmatrix} = \begin{bmatrix} \ddot{z} \\ \tau_\phi \\ \tau_\theta \\ \tau_\varphi \end{bmatrix} = \begin{bmatrix} k & k & k & k \\ 0 & -Lk & 0 & Lk \\ -Lk & 0 & Lk & 0 \\ b & -b & b & b \end{bmatrix} \begin{bmatrix} w_1^2 \\ w_2^2 \\ w_3^2 \\ w_4^2 \end{bmatrix} \xrightarrow{\quad f \quad} u = Tf$$
$$f = \text{inv}(T)u$$

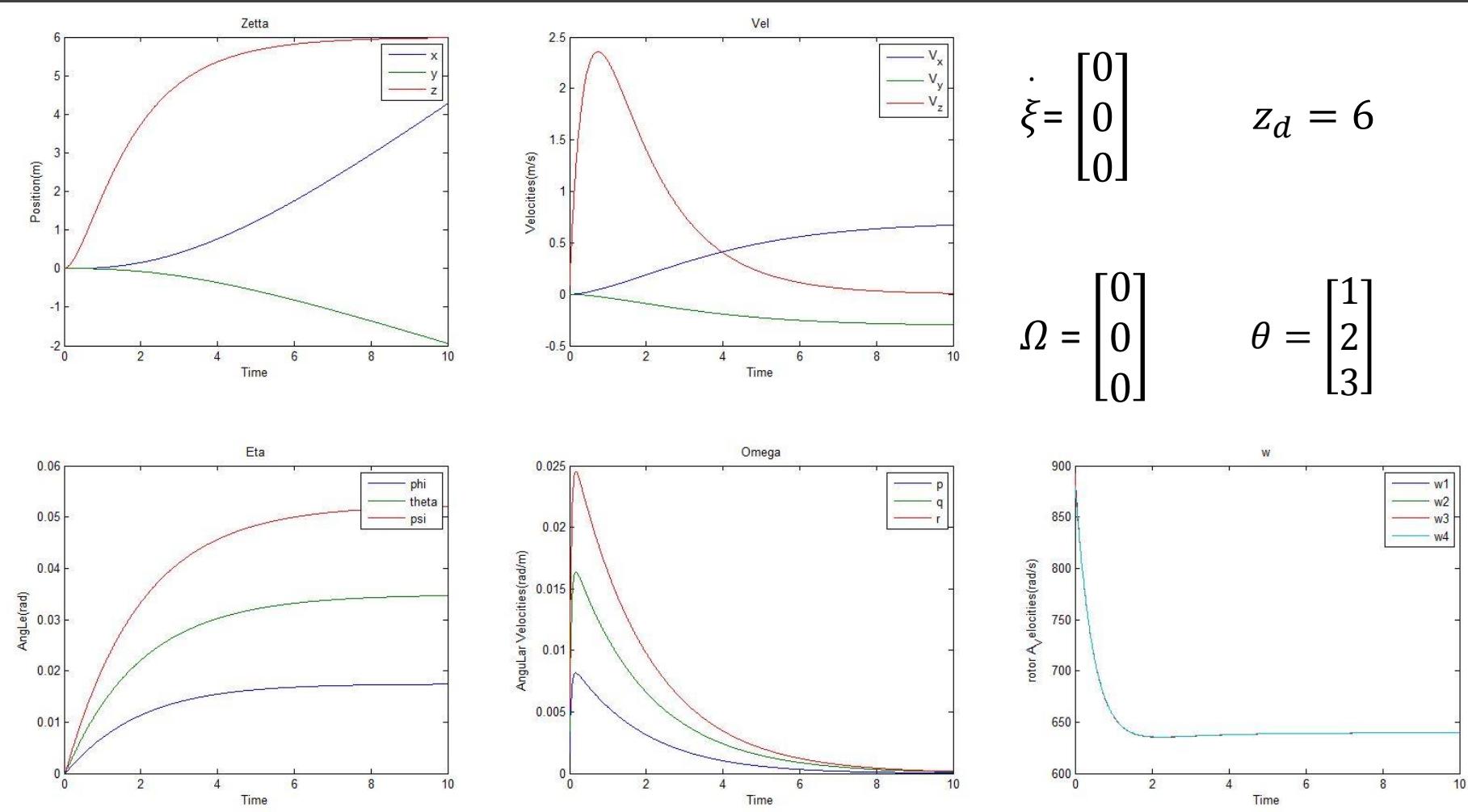
u

T

f



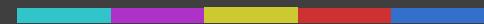
### III. Stabilisation of quadcopter



$$\xi = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

$$z_d = 6$$

$$\Omega = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad \theta = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$



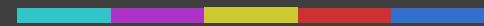
## VI. Trajectory control

Integrated PD controller

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = -g \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} + \frac{T}{m} \begin{bmatrix} C_\phi S_\theta C_\phi + S_\phi S_\theta \\ S_\phi S_\theta C_\phi - C_\phi S_\theta \\ C_\theta C_\phi \end{bmatrix} - \frac{1}{m} \begin{bmatrix} A_x & 0 & 0 \\ 0 & A_y & 0 \\ 0 & 0 & A_z \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix}$$

$$T_B = \begin{bmatrix} 0 \\ 0 \\ T \end{bmatrix} = R^T \left( m \left( \ddot{\xi} + \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \right) + \begin{bmatrix} A_x & 0 & 0 \\ 0 & A_y & 0 \\ 0 & 0 & A_z \end{bmatrix} \dot{\xi} \right)$$

$$\frac{1}{m} \begin{bmatrix} 0 \\ 0 \\ T \end{bmatrix} = \begin{bmatrix} C_\theta C_\phi & C_\theta S_\phi & -S_\theta \\ S_\theta C_\phi S_\phi - S_\phi C_\phi & S_\theta S_\phi S_\phi + C_\phi C_\phi & C_\theta S_\phi \\ S_\theta C_\phi C_\phi + S_\phi S_\phi & S_\theta S_\phi C_\phi - C_\phi S_\phi & C_\theta C_\phi \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \\ U_3 + g \end{bmatrix}$$



## VI. Trajectory control

Integrated PD controller

$$U_1 \cos\theta \cos\varphi + U_2 \cos\theta \sin\varphi - (U_3 + g) \sin\theta = 0$$

$$\begin{aligned} & U_1(\sin\theta \cos\varphi \sin\phi - \sin\varphi \cos\phi) + U_2(\sin\phi \sin\varphi \sin\phi + \cos\varphi \cos\phi) \\ & + (U_3 + g) \cos\theta \sin\phi = 0 \end{aligned}$$

$$\begin{aligned} & \checkmark U_1(\sin\theta \cos\varphi \cos\phi + \sin\varphi \sin\phi) + U_2(\sin\theta \sin\varphi \cos\phi - \cos\varphi \sin\phi) \\ & + (U_3 + g) \cos\theta \cos\phi = \frac{1}{m} T \end{aligned}$$

$$\therefore \quad \phi = \arcsin \left( \frac{U_1 \sin\varphi - U_2 \cos\varphi}{\sqrt{U_1^2 + U_2^2 + (U_3 - g)^2}} \right) \quad \theta = \arctan \left( \frac{U_1 \cos\varphi + U_2 \sin\varphi}{U_3 - g} \right)$$



## VI. Trajectory control

Integrated PD controller

$$\begin{aligned} T = & m(U_1(\sin\theta\cos\varphi\cos\phi + \sin\varphi\sin\phi) \\ & + U_2(\sin\theta\sin\varphi\cos\phi - \cos\varphi\sin\phi) \\ & + (U_3 + g)\cos\theta\cos\phi) \end{aligned}$$

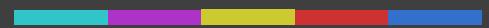


$$\begin{aligned} U_1 &= d_x \\ U_2 &= d_y \\ U_3 &= d_z \end{aligned}$$



$$\begin{aligned} \dot{\tau}_\phi &= (K_{\phi,D}(\dot{\phi}_c - \dot{\phi}) + K_{\phi,P}(\phi_c - \phi))I_{xx} \\ \dot{\tau}_\theta &= (K_{\theta,D}(\dot{\theta}_c - \dot{\theta}) + K_{\theta,P}(\theta_c - \theta))I_{yy} \\ \dot{\tau}_\varphi &= (K_{\varphi,D}(\dot{\varphi}_d - \dot{\varphi}) + K_{\varphi,P}(\varphi_d - \varphi))I_{zz} \end{aligned}$$

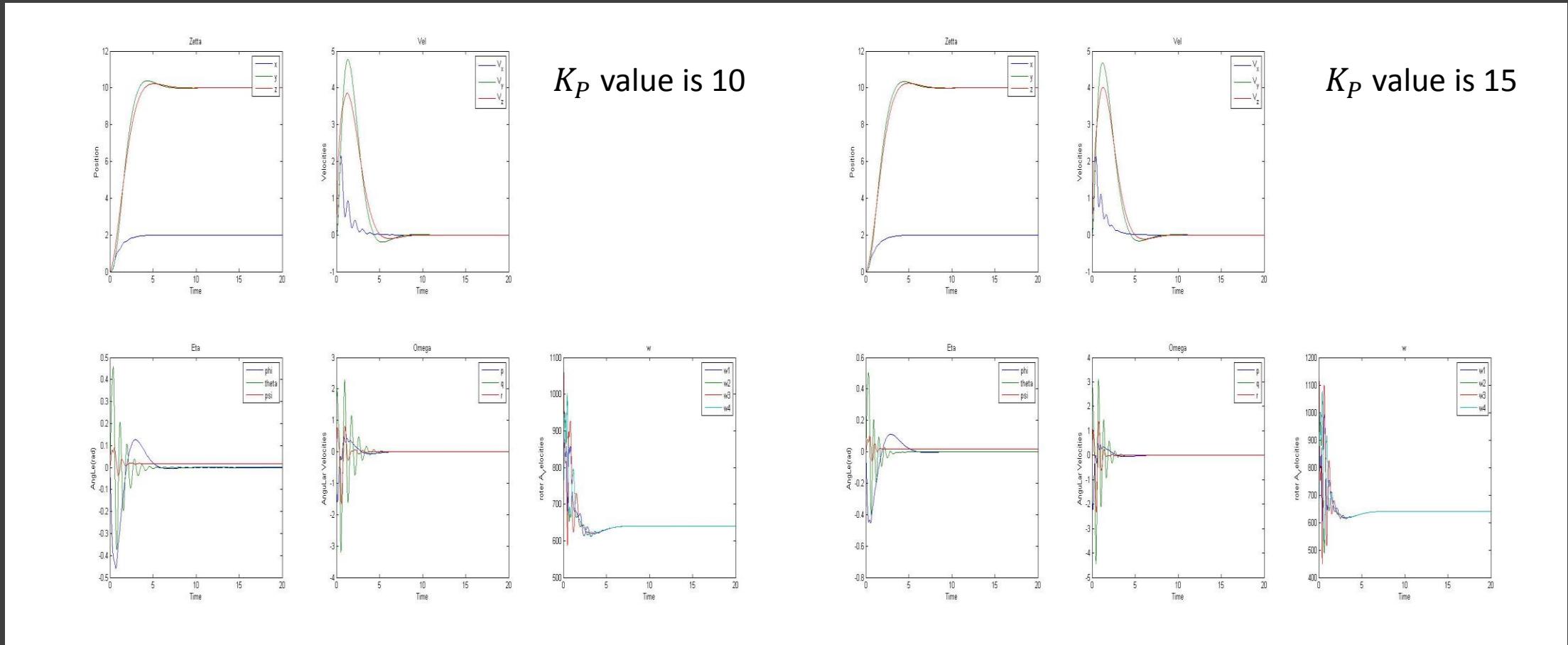
$$\begin{aligned} d_x &= K_{x,D}(\dot{x}_d - \dot{x}) + K_{x,P}(x_d - x) \\ d_y &= K_{y,D}(\dot{y}_d - \dot{y}) + K_{y,P}(y_d - y) \\ d_z &= K_{z,D}(\dot{z}_d - \dot{z}) + K_{z,P}(z_d - z) \end{aligned}$$



# VI. Trajectory control

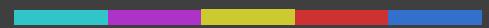
How find gain value

The figure,  $K_P$  value is 10 and 15



$K_P$  value is 10

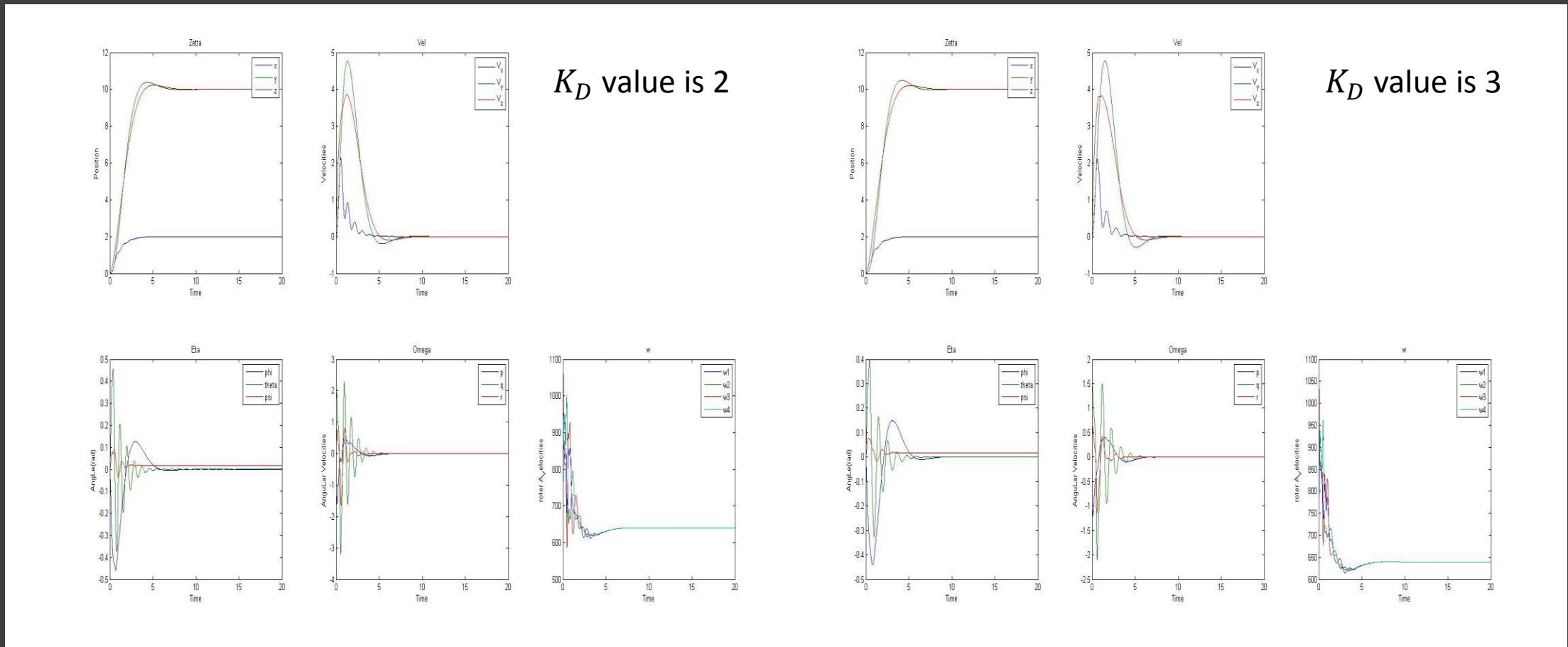
$K_P$  value is 15



# VI. Trajectory control

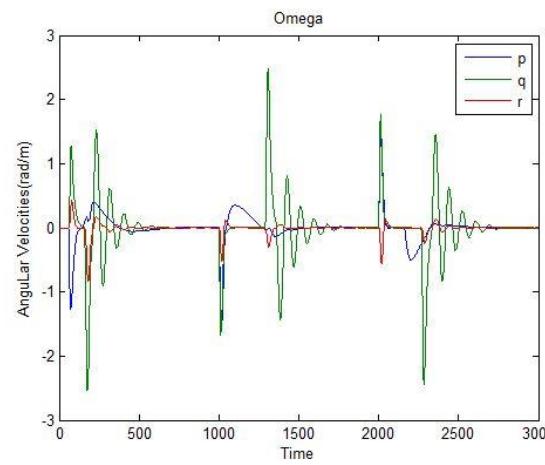
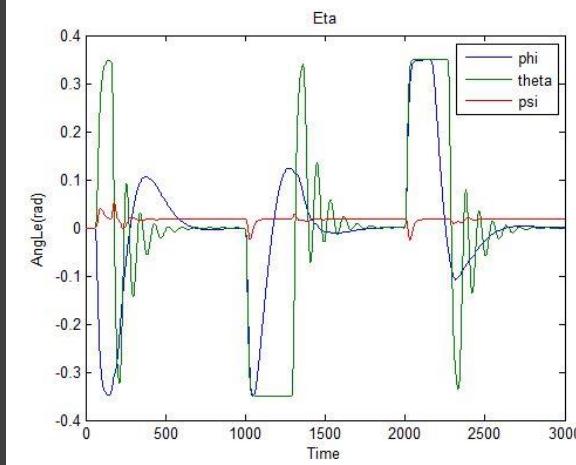
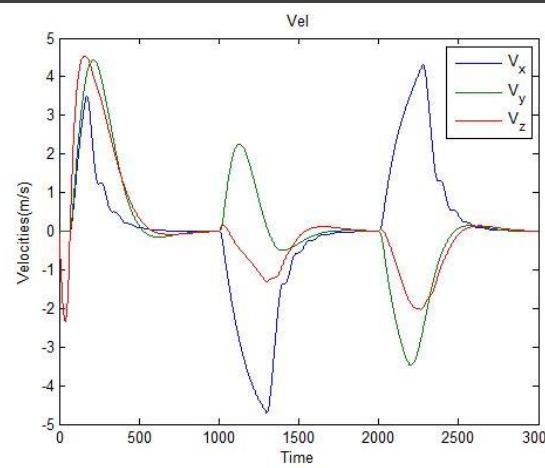
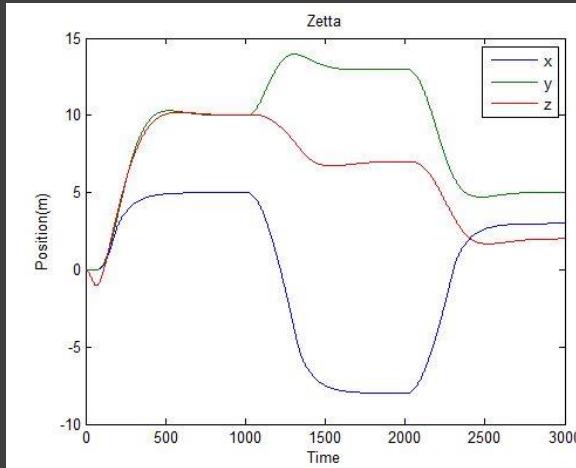
How find gain value

The figure,  $K_D$  value is 2 and 3





## VI. Trajectory control



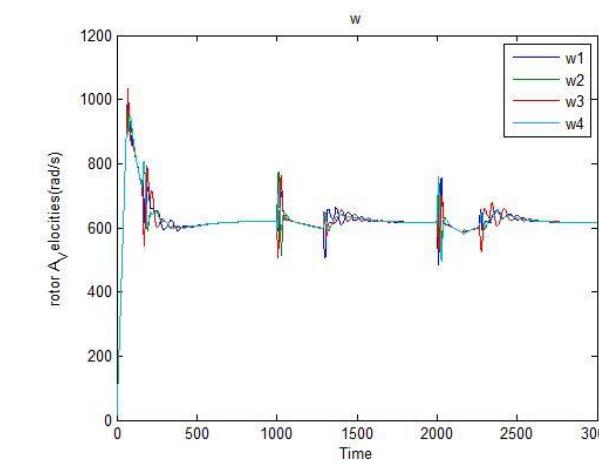
$$K_P = 10 \quad K_D = 2$$

$$\text{Theta Command} = [0 \quad 0 \quad 1]$$

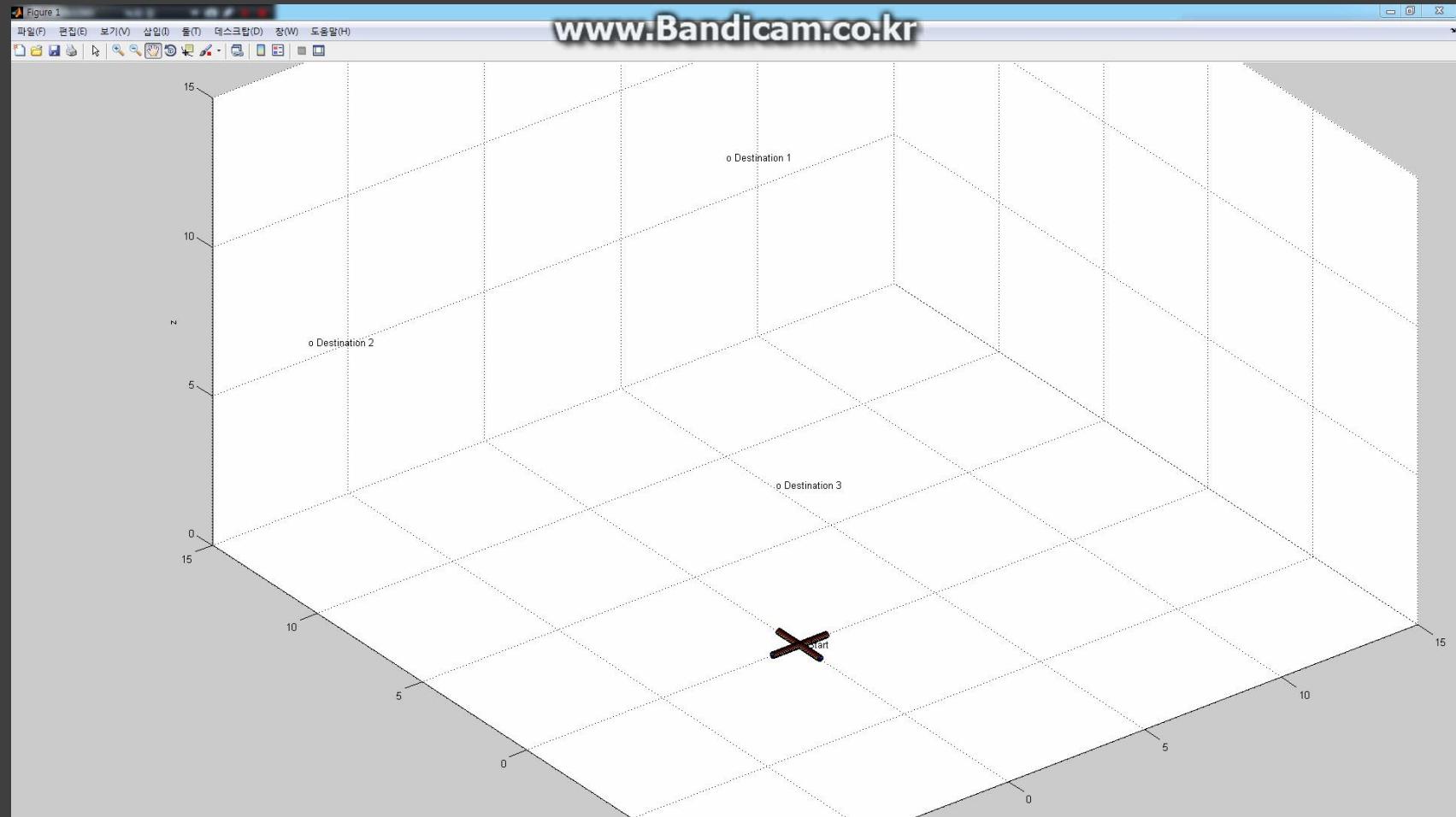
$$\text{Position Command 1} = [5 \quad 10 \quad 10]$$

$$\text{Position Command 2} = [-8 \quad 13 \quad 7]$$

$$\text{Position Command 3} = [3 \quad 5 \quad 2]$$



# VI. Trajectory control

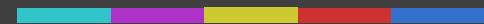


Theta Command = [0 0 1]

Position Command 1 = [5 10 10]

Position Command 2 = [-8 13 7]

Position Command 3 = [3 5 2]



# Future Plan

