Design of a Fuzzy-PID Controller for Controlling Quadrotor Attitude and Altitude

PREPARED BY,

MRINMOY SARKAR & BRIAN BAITY

ADVISOR,

DR. ABDOLLAH HOMAIFAR

Overview

- ✓Motivation
- ✓Challenges
- ✓Mathematical modeling of quadrotor
- ✓PID controller design
- ✓Fuzzy-PID controller design
- ✓Simulation
- ✓Conclusion

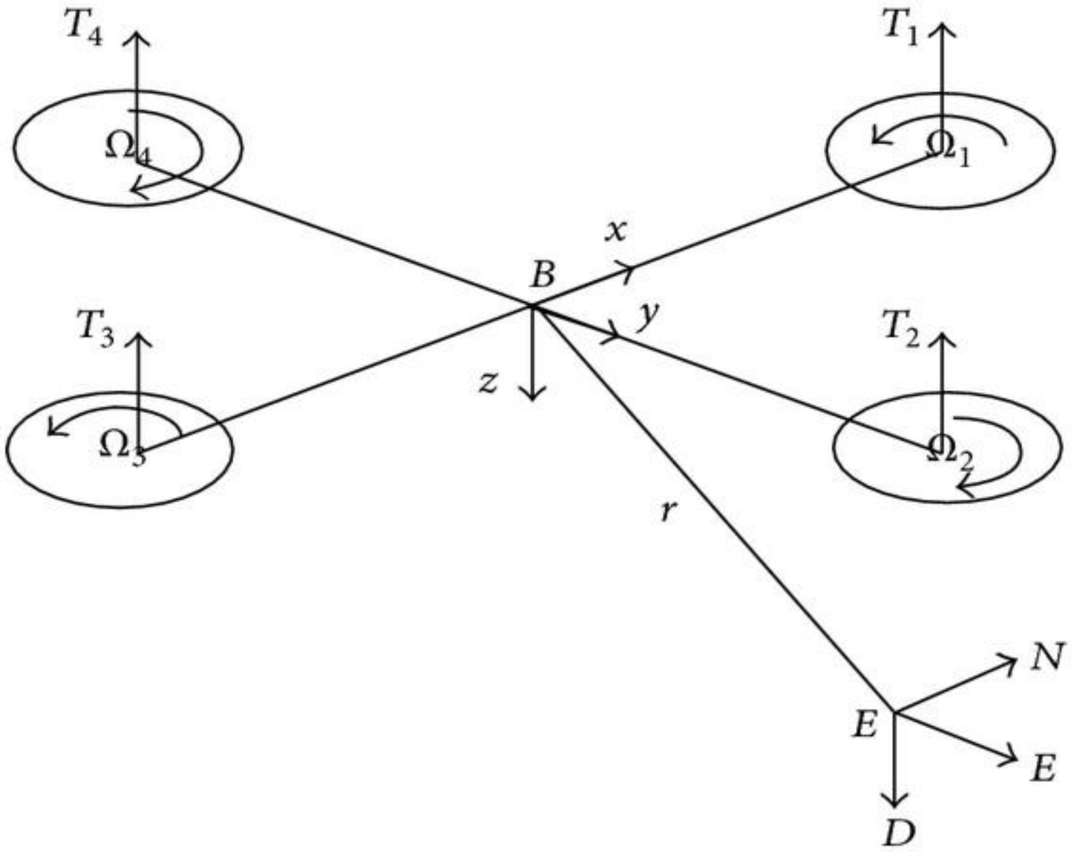
Motivation

- ✓ Fuzzy PID Controller solves the problem of fixed gain
- ✓ Fuzzy PID Controller is cheaper to develop
- ✓ Fuzzy PID Controller covers a wider range of operating conditions
- ✓ Fuzzy PID Controllers are more readily customizable
- ✓ Fuzzy PID Controller is robust in terms of uncertainty

Challenges

- ✓Nonlinear system
- ✓Under actuated system
- ✓Low on board processing capability
- ✓Low operation time
- ✓Low efficiency in power consumption

Mathematical modeling of quadrotor



Structure of a quadrotor

He, Z. and Zhao, L., 2014. A simple attitude control of quadrotor helicopter based on ziegler-nichols rules for tuning pd parameters. *The Scientific World Journal*, *2014*.

Cont.

$$R(x,\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix}$$

$$R(y,\theta) = \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix}$$

$$R(z,\psi) = \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$R_{E\to B} = R(x,\phi)\,R(y,\theta)\,R(z,\psi)$$

$$R_{B\to E} = R_{E\to B}^{T}.$$

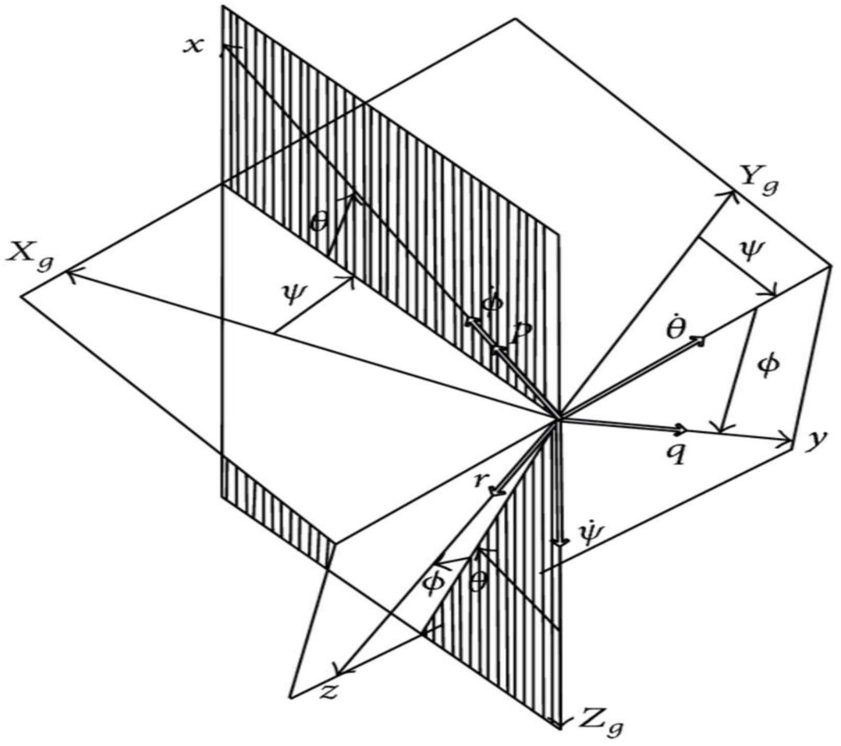
$$R_{B\to E} = \begin{bmatrix} c_\theta c_\psi & s_\phi s_\theta c_\psi - c_\phi s_\psi & c_\phi s_\theta c_\psi + s_\phi s_\psi \\ c_\theta s_\psi & s_\phi s_\theta s_\psi + c_\phi c_\psi & c_\phi s_\theta s_\psi - s_\phi c_\psi \\ -s_\theta & s_\phi c_\theta & c_\phi c_\theta \end{bmatrix}$$

where $c_\cdot = \cos(\cdot)$ and $s_\cdot = \sin(\cdot)$.

roll angle ϕ, pitch angle θ, and yaw angle ψ around x-, y-, and z-axes, respectively.

He, Z. and Zhao, L., 2014. A simple attitude control of quadrotor helicopter based on ziegler-nichols rules for tuning pd parameters. *The Scientific World Journal*, *2014*.

Cont.



The relationships between angular velocity components and the attitude angle change rate.

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = R_r \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}$$

where

$$R_r = \begin{bmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\phi & \sin\phi\cos\theta \\ 0 & -\sin\phi & \cos\phi\cos\theta \end{bmatrix}$$

He, Z. and Zhao, L., 2014. A simple attitude control of quadrotor helicopter based on ziegler-nichols rules for tuning pd parameters. *The Scientific World Journal*, *2014*.

Cont.

Dynamic Model: The dynamics model is composed of the rotational and translational motions. The rotational motion is fully actuated, while the translational motion is underactuated. In the body coordinate system, the rotational motion equations are derived according to the law of momentum theorem and gyroscopic effect of quadrotor, and they are given by,

$$J\dot{\omega} + \omega \times J\omega + \omega \times \begin{bmatrix} 0 & 0 & J_r\Omega_r \end{bmatrix} = M_B$$

$$F_i = b\Omega_i^2$$

$$M_i = d\Omega_i^2$$

where Ω_i $(i = 1, 2, 3, 4)$ represents the ith rotor speed.

$$M_B = \begin{bmatrix} l \cdot b\left(-\Omega_2^2 + \Omega_4^2\right) \\ l \cdot b\left(\Omega_1^2 - \Omega_3^2\right) \\ d\left(\Omega_1^2 - \Omega_2^2 + \Omega_3^2 - \Omega_4^2\right) \end{bmatrix} \qquad m\ddot{r} = \begin{bmatrix} 0 & 0 & mg \end{bmatrix}^T + RF_B \qquad F_B = \begin{bmatrix} 0 \\ 0 \\ -b\left(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2\right) \end{bmatrix}$$

He, Z. and Zhao, L., 2014. A simple attitude control of quadrotor helicopter based on ziegler-nichols rules for tuning pd parameters. *The Scientific World Journal*, *2014*.

Cont.

The Motion Equations of Quadrotor:

$$\ddot{\phi} = \dot{\theta}\dot{\psi}\left(\frac{I_y - I_z}{I_x}\right) - \frac{J_r}{I_x}\dot{\theta}\Omega_r + \frac{L}{I_x}U_2,$$

$$\ddot{\theta} = \dot{\phi}\dot{\psi}\left(\frac{I_z - I_x}{I_y}\right) + \frac{J_r}{I_y}\dot{\phi}\Omega_r + \frac{L}{I_y}U_3,$$

$$\ddot{\psi} = \dot{\phi}\dot{\theta}\left(\frac{I_x - I_y}{I_z}\right) + \frac{1}{I_z}U_4,$$

$$\ddot{x} = -\frac{U_1}{m}\left(\cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi\right),$$

$$\ddot{y} = -\frac{U_1}{m}\left(\cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi\right),$$

$$\ddot{z} = g - \frac{U_1}{m}\left(\cos\phi\cos\theta\right),$$

where U_1, U_2, U_3, and U_4 are the control input variables, which can be calculated by $U_1 = b(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2)$, $U_2 = b(-\Omega_2^2 + \Omega_4^2)$, $U_3 = b(\Omega_1^2 - \Omega_3^2)$, and $U_4 = d(\Omega_1^2 - \Omega_2^2 + \Omega_3^2 - \Omega_4^2)$, respectively.

He, Z. and Zhao, L., 2014. A simple attitude control of quadrotor helicopter based on ziegler-nichols rules for tuning pd parameters. *The Scientific World Journal*, *2014*.

Cont. (State space model)

The state space model adopted by the control system is $\dot{X} = f(X,U)$, where X is the state vector and U is the control input vector. The state vector is chosen as $X = [x \;\; y \;\; z \;\; \dot{x} \;\; \dot{y} \;\; \dot{z} \;\; \phi \;\; \theta \;\; \psi \;\; \dot{\phi} \;\; \dot{\theta} \;\; \dot{\psi}]^T$. In the design of controller, the state variables are chosen as $x_1 = x$, $x_2 = y$, $x_3 = z$, $x_4 = \dot{x}_1 = \dot{x}$, $x_5 = \dot{x}_2 = \dot{y}$, $x_6 = \dot{x}_3 = \dot{z}$, $x_7 = \phi$, $x_8 = \theta$, $x_9 = \psi$, $x_{10} = \dot{x}_7 = \dot{\phi}$, $x_{11} = \dot{x}_8 = \dot{\theta}$, and $x_{12} = \dot{x}_9 = \dot{\psi}$.

Synthesizing the motion equations of quadrotor, the state vector, and the control input variables, the state equations can be described as

$$f(X,U) = \begin{bmatrix} x_4 \\ x_5 \\ x_6 \\ -\frac{U_1}{m}(\cos x_7 \sin x_8 \cos x_9 + \sin x_7 \sin x_9) \\ -\frac{U_1}{m}(\cos x_7 \sin x_8 \sin x_9 - \sin x_7 \cos x_9) \\ g - \frac{U_1}{m}(\cos x_7 \cos x_8) \\ x_{10} \\ x_{11} \\ x_{12} \\ x_{11}x_{12}\frac{I_y - I_z}{I_x} - \frac{J_r}{I_x}x_{11}\Omega_r + \frac{L}{I_x}U_2 \\ x_{10}x_{12}\frac{I_z - I_x}{I_y} + \frac{J_r}{I_y}x_{10}\Omega_r + \frac{L}{I_y}U_3 \\ x_{10}x_{11}\frac{I_x - I_y}{I_z} + \frac{1}{I_z}U_4 \end{bmatrix}$$

He, Z. and Zhao, L., 2014. A simple attitude control of quadrotor helicopter based on ziegler-nichols rules for tuning pd parameters. *The Scientific World Journal*, *2014*.

Underactuation

Underactuation is a technical term used in robotics and control theory to describe mechanical systems that cannot be commanded to follow arbitrary trajectories in configuration space. This condition can occur for a number of reasons, the simplest of which is when the system has a lower number of actuators than degrees of freedom. In this case, the system is said to be trivially underactuated.

The class of underactuated mechanical systems is very rich and includes such diverse members as automobiles, airplanes, and even animals.

$$\ddot{q} = f(q, \dot{q}, u, t)$$

Where:

$q \in \mathbb{R}^n$ is the position state vector

$u \in \mathbb{R}^m$ is the vector of control inputs

t is time.

Furthermore, in many cases the dynamics for these systems can be rewritten to be affine in the control inputs:

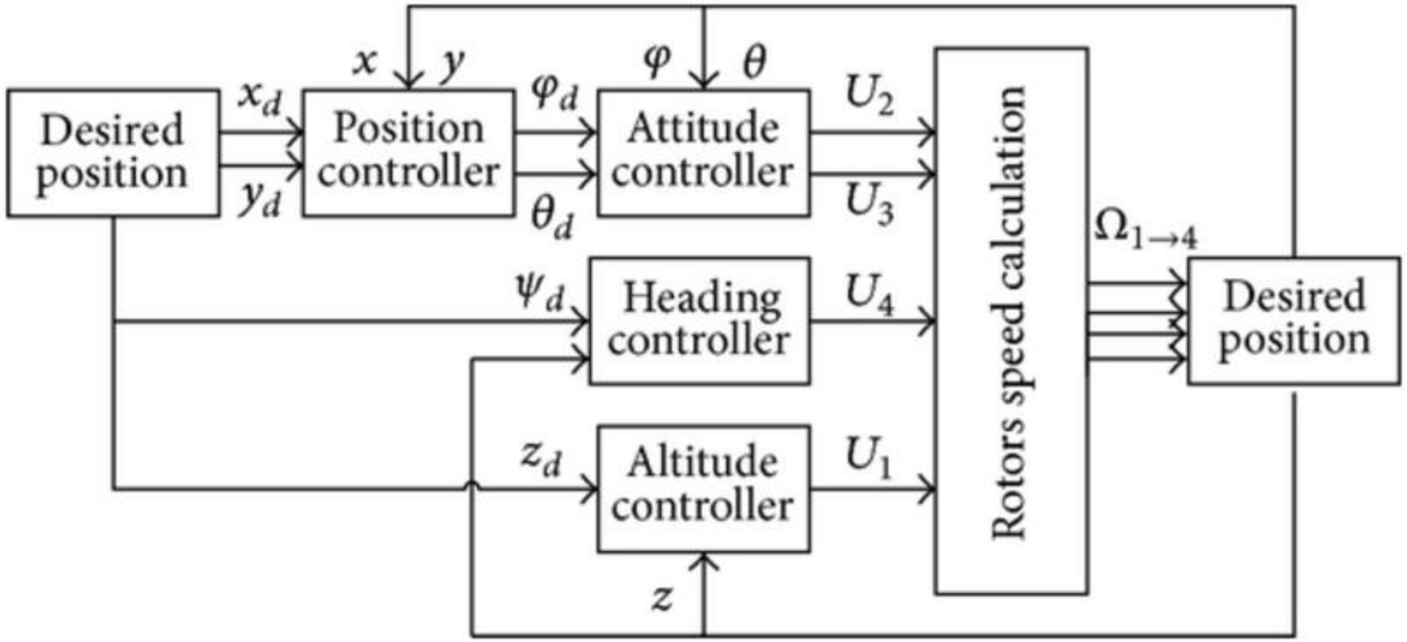
$$\ddot{q} = f_1(q, \dot{q}, t) + f_2(q, \dot{q}, t)u$$

When expressed in this form, the system is said to be underactuated if:[1]

$$rank[f_2(q, \dot{q}, t)] < dim[q]$$

When this condition is met, there are acceleration directions that can not be produced no matter what the control vector is.

PID Controller Design



Quadrotor control structure

He, Z. and Zhao, L., 2014. A simple attitude control of quadrotor helicopter based on ziegler-nichols rules for tuning pd parameters. *The Scientific World Journal*, *2014*.

Hovering Condition

For hovering,

$\Phi = 0$;

$\theta = 0$;

$\Psi = 0$;

$\Phi` = 0$;

$\Theta` = 0$;

$\Psi` = 0$;

$X` = 0$;

$Y` = 0$;

$Z` = 0$;

$\Phi`` = 0$;

$\Theta`` = 0$;

$\Psi`` = 0$;

$X`` = 0$;

$Y`` = 0$;

$Z`` = 0$;

So, for hovering Condition the control inputs are,

$U_1 = mg$

$U_2 = 0$

$U_3 = 0$

$U_4 = 0$

where $a_1 = (I_y - I_z)/I_x$, $a_2 = J_r/I_x$, $a_3 = (I_z - I_x)/I_y$, $a_4 = J_r/I_y$, $a_5 = (I_x - I_y)/I_z$, $b_1 = L/I_x$, $b_2 = L/I_y$, and $b_3 = 1/I_z$.

He, Z. and Zhao, L., 2014. A simple attitude control of quadrotor helicopter based on ziegler-nichols rules for tuning pd parameters. *The Scientific World Journal*, *2014*.

Feedback Linearization

$$\dot{x}_{10} = a_1 x_{11} x_{12} + b_1 U_2,$$
$$\dot{x}_{11} = a_3 x_{10} x_{12} + b_2 U_3,$$
$$\dot{x}_{12} = a_5 x_{10} x_{11} + b_3 U_4.$$

$$U_2 = f_2(x_{10}, x_{11}, x_{12}) + U_2^{\#},$$
$$U_3 = f_3(x_{10}, x_{11}, x_{12}) + U_3^{\#},$$
$$U_4 = f_4(x_{10}, x_{11}, x_{12}) + U_4^{\#},$$

$$a_1 x_{11} x_{12} + b_1 f_2(x_{10}, x_{11}, x_{12}) = K_2 x_{10},$$
$$a_3 x_{10} x_{12} + b_2 f_3(x_{10}, x_{11}, x_{12}) = K_3 x_{11},$$
$$a_5 x_{10} x_{11} + b_3 f_4(x_{10}, x_{11}, x_{12}) = K_4 x_{10},$$

$$\dot{x}_{10} = K_2 x_{10} + b_1 U_2^{\#},$$
$$\dot{x}_{11} = K_3 x_{11} + b_2 U_3^{\#},$$
$$\dot{x}_{12} = K_4 x_{12} + b_3 U_4^{\#}.$$

$$f_2(x_{10}, x_{11}, x_{12}) = \frac{1}{b_1}(K_2 x_{10} - a_1 x_{11} x_{12}),$$
$$f_3(x_{10}, x_{11}, x_{12}) = \frac{1}{b_2}(K_3 x_{11} - a_3 x_{10} x_{12}),$$
$$f_4(x_{10}, x_{11}, x_{12}) = \frac{1}{b_3}(K_4 x_{10} - a_5 x_{10} x_{11}).$$

where K_2, K_3, and K_4 are undetermined parameters.

He, Z. and Zhao, L., 2014. A simple attitude control of quadrotor helicopter based on ziegler-nichols rules for tuning pd parameters. *The Scientific World Journal*, *2014*.

Laplace Transformation of the Linear Model and PID Controller

$$G_1(s) = \frac{X_7(s)}{U_2^{\#}(s)} = \frac{b_1}{s^2 - K_2 s},$$

$$G_2(s) = \frac{X_8(s)}{U_3^{\#}(s)} = \frac{b_2}{s^2 - K_3 s},$$

$$G_3(s) = \frac{X_9(s)}{U_4^{\#}(s)} = \frac{b_3}{s^2 - K_4 s},$$

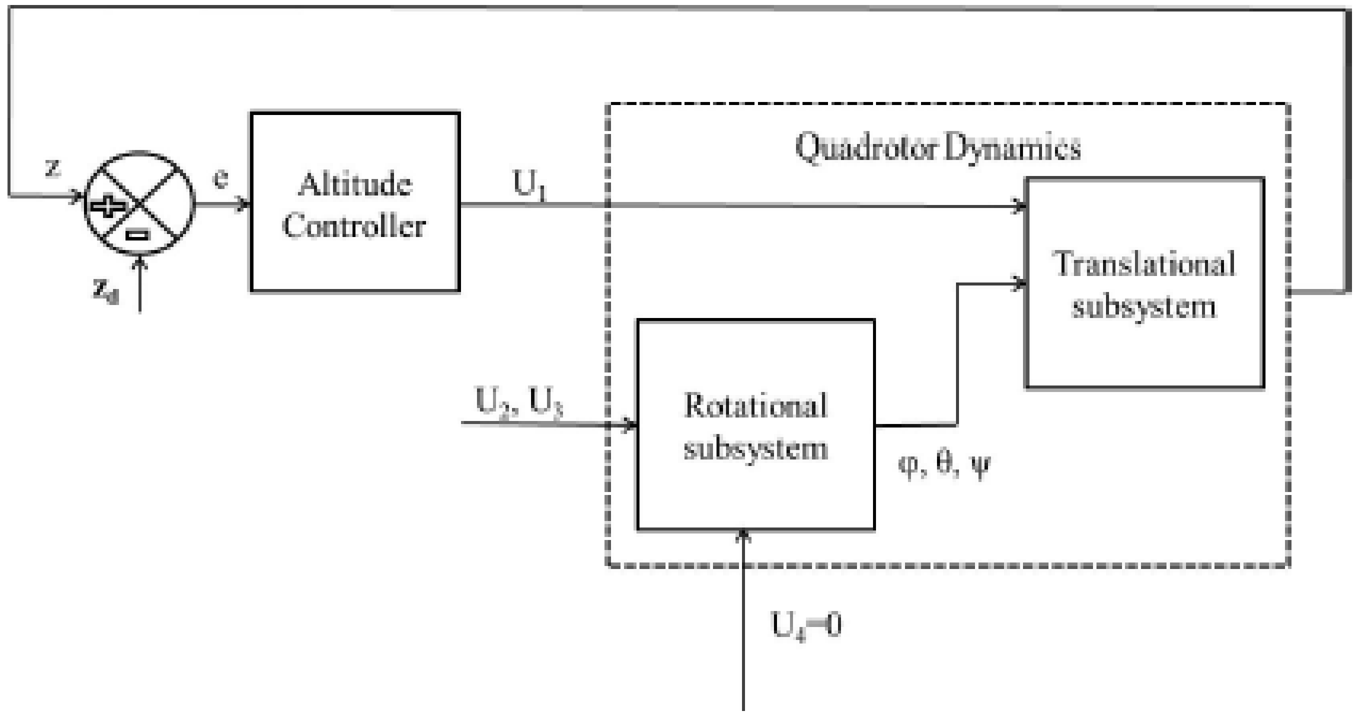
$$K(s) = k_p + \frac{k_i}{s} + k_d s = \frac{k_p s + k_i + k_d s^2}{s}$$

Where,

k_p = proportional gain
k_i = integral gain
k_d = differential gain

He, Z. and Zhao, L., 2014. A simple attitude control of quadrotor helicopter based on ziegler-nichols rules for tuning pd parameters. *The Scientific World Journal, 2014*.

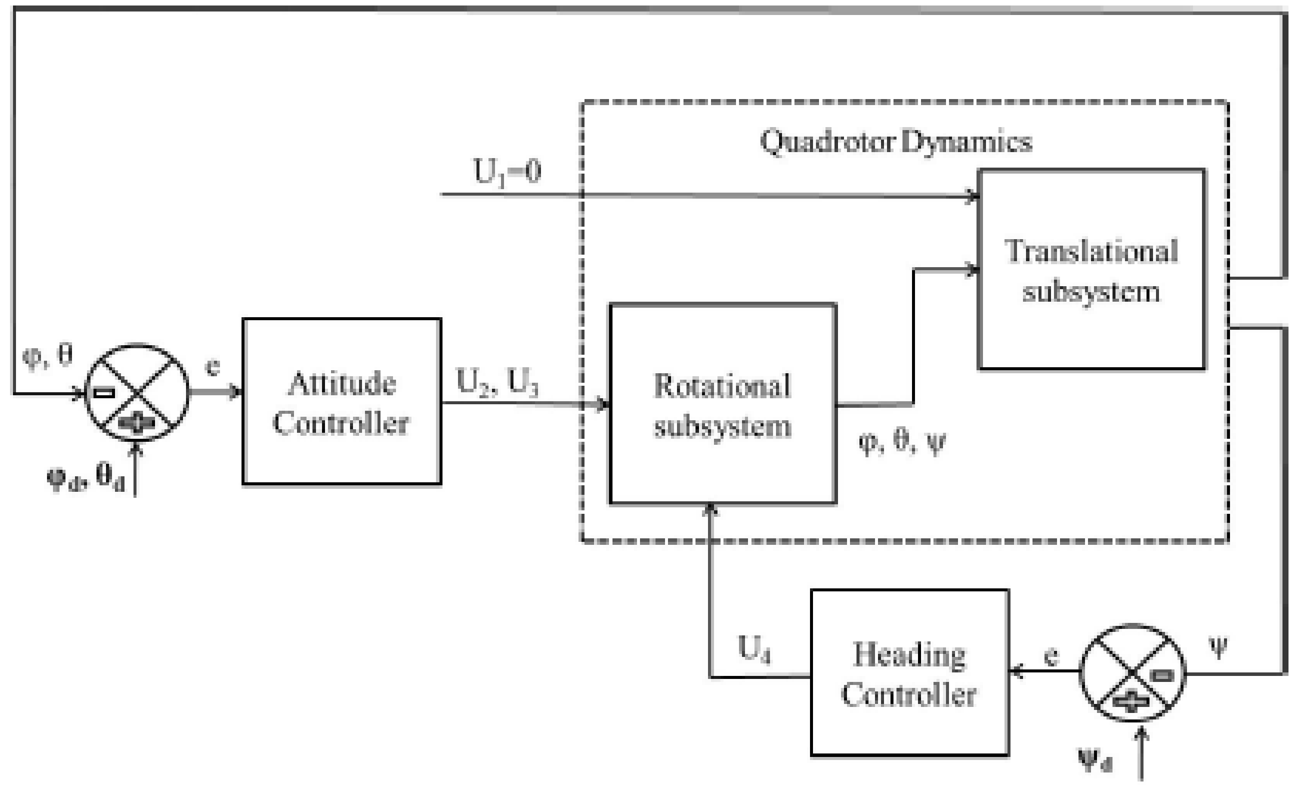
Altitude Controller



Block Diagram for Altitude Controller

Heba talla Mohamed Nabil ElKholy, 2014. Dynamic Modeling and Control of a Quadrotor Using Linear and Nonlinear Approaches, Thesis for degree of Master of Science in Robotics, Control and Smart Systems, *2014*.

Attitude and Heading Controller



Block Diagram for Attitude and Heading Controller

Heba talla Mohamed Nabil ElKholy, 2014. Dynamic Modeling and Control of a Quadrotor Using Linear and Nonlinear Approaches, Thesis for degree of Master of Science in Robotics, Control and Smart Systems, *2014*.

Position Controller

- Position can not be controlled directly for a quadcopter.

$$\ddot{x} = \frac{-U_1}{m}(\sin\phi_d \sin\psi + \cos\phi_d \sin\theta_d \cos\psi)$$
$$\ddot{y} = \frac{-U_1}{m}(\cos\phi_d \sin\theta_d \sin\psi - \sin\phi_d \cos\psi)$$

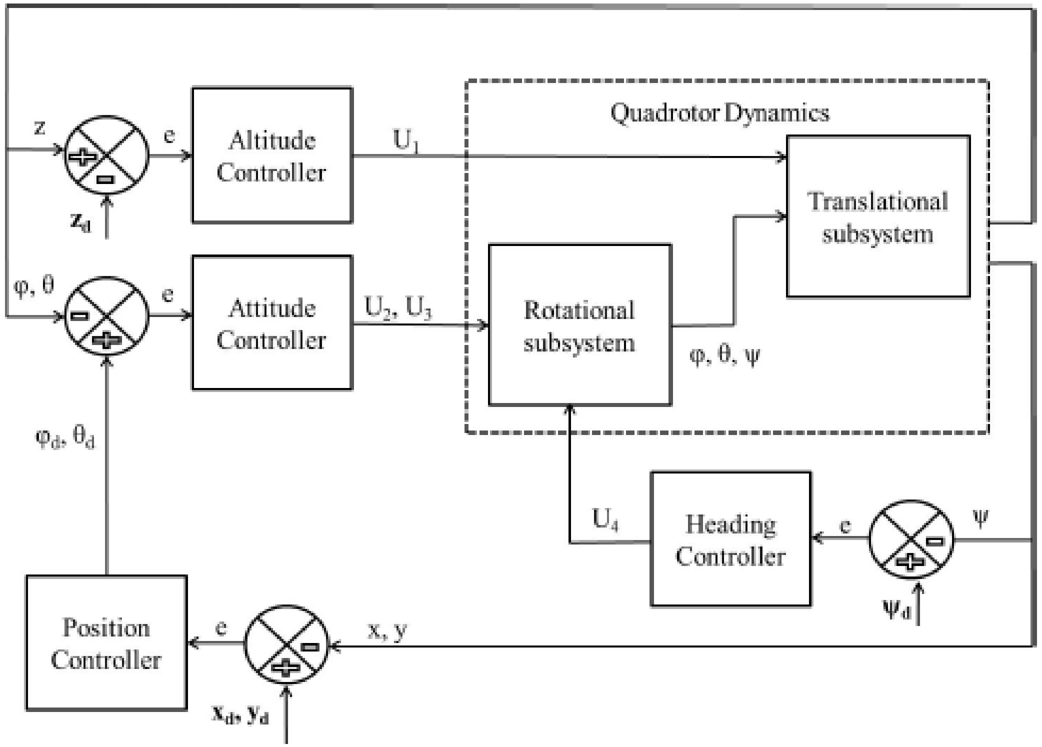
small angle assumption $(\sin\phi_d \equiv \phi_d, \sin\theta_d \equiv \theta_d$ and $\cos\phi_d = \cos\theta_d = 1)$

$$\ddot{x} = \frac{-U_1}{m}(\phi_d \sin\psi + \theta_d \cos\psi)$$
$$\ddot{y} = \frac{-U_1}{m}(\theta_d \sin\psi - \phi_d \cos\psi)$$

$$\begin{bmatrix}\phi_d\\ \theta_d\end{bmatrix} = \begin{bmatrix}-\sin\psi & -\cos\psi\\ \cos\psi & -sin\psi\end{bmatrix}^{-1} \frac{m}{U_1}\begin{bmatrix}\ddot{x}_d\\ \ddot{y}_d\end{bmatrix}$$
$$= \frac{m}{U_1}\begin{bmatrix}-\sin\psi & \cos\psi\\ -\cos\psi & -sin\psi\end{bmatrix}\begin{bmatrix}\ddot{x}_d\\ \ddot{y}_d\end{bmatrix}$$
$$= \frac{m}{U_1}\begin{bmatrix}-\ddot{x}_d \sin\psi + \ddot{y}_d \cos\psi\\ -\ddot{x}_d \cos\psi - \ddot{y}_d sin\psi\end{bmatrix}$$

Heba talla Mohamed Nabil ElKholy, 2014. Dynamic Modeling and Control of a Quadrotor Using Linear and Nonlinear Approaches, Thesis for degree of Master of Science in Robotics, Control and Smart Systems, *2014*.

Position Controller (cont.)



Block Diagram for Position Controller

Heba talla Mohamed Nabil ElKholy, 2014. Dynamic Modeling and Control of a Quadrotor Using Linear and Nonlinear Approaches, Thesis for degree of Master of Science in Robotics, Control and Smart Systems, *2014*.

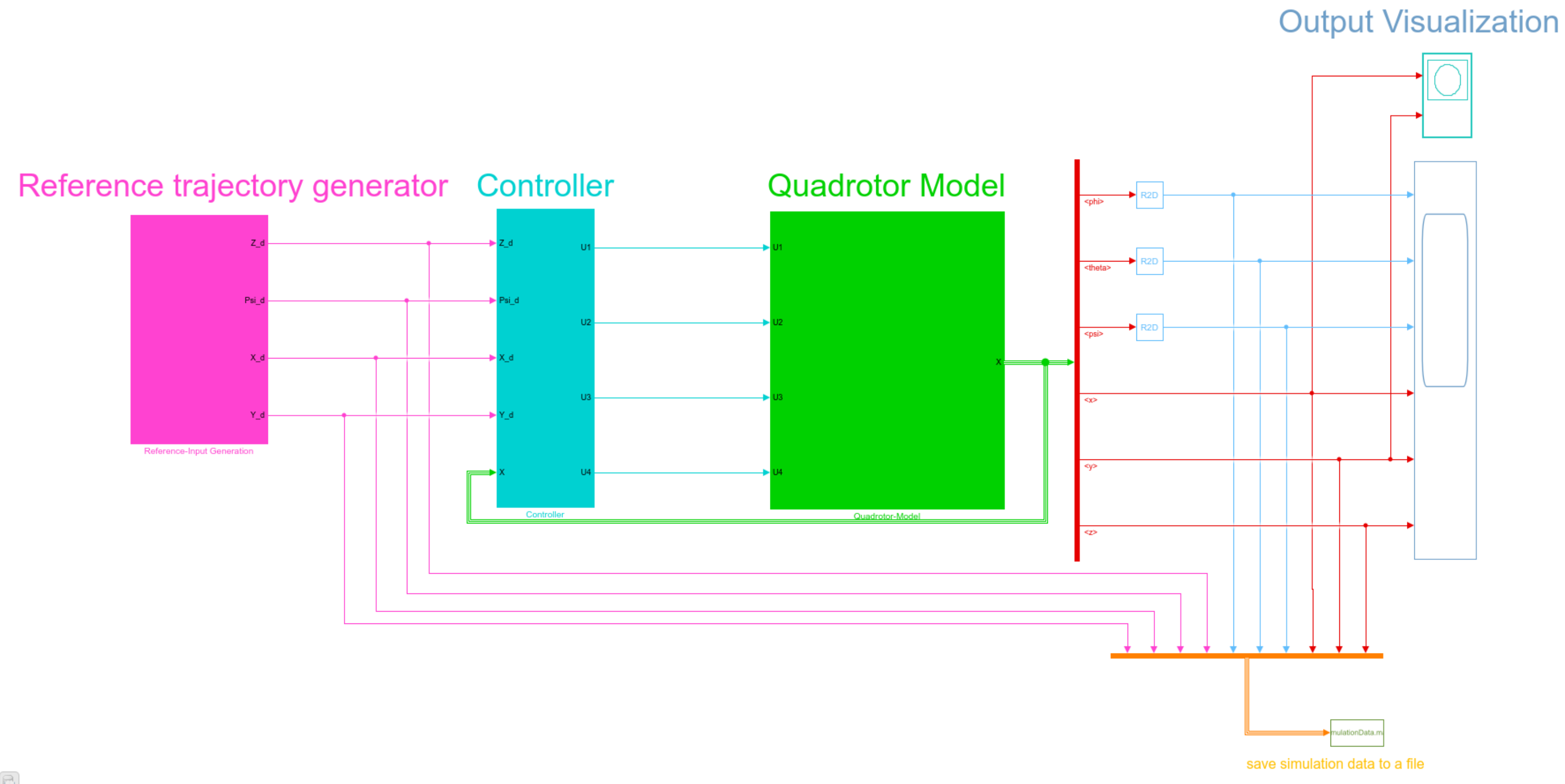
Simulation

Parameters	Description	Values
m	Quadrotor mass	0.4794 kg
g	Gravitational acceleration	9.8 m/s^2
l	Moment arm	0.225 m
I_x	MOI about body frame's x-axis	0.0086 $kg.m^2$
I_y	MOI about body frame's y-axis	0.0086 $kg.m^2$
I_z	MOI about body frame's z-axis	0.0172 $kg.m^2$
J_r	Rotor inertia	3.7404×10^{-5} $kg.m^2$
b	Aerodynamic force constant	3.13×10^{-5} $N.s^2$
d	Aerodynamic moment constant	9×10^{-7} $Nm.s^2$
K_2	Linearization constant	-14.6211
K_3	Linearization constant	-14.6211
K_4	Linearization constant	-32

He, Z. and Zhao, L., 2014. A simple attitude control of quadrotor helicopter based on ziegler-nichols rules for tuning pd parameters. *The Scientific World Journal, 2014*.

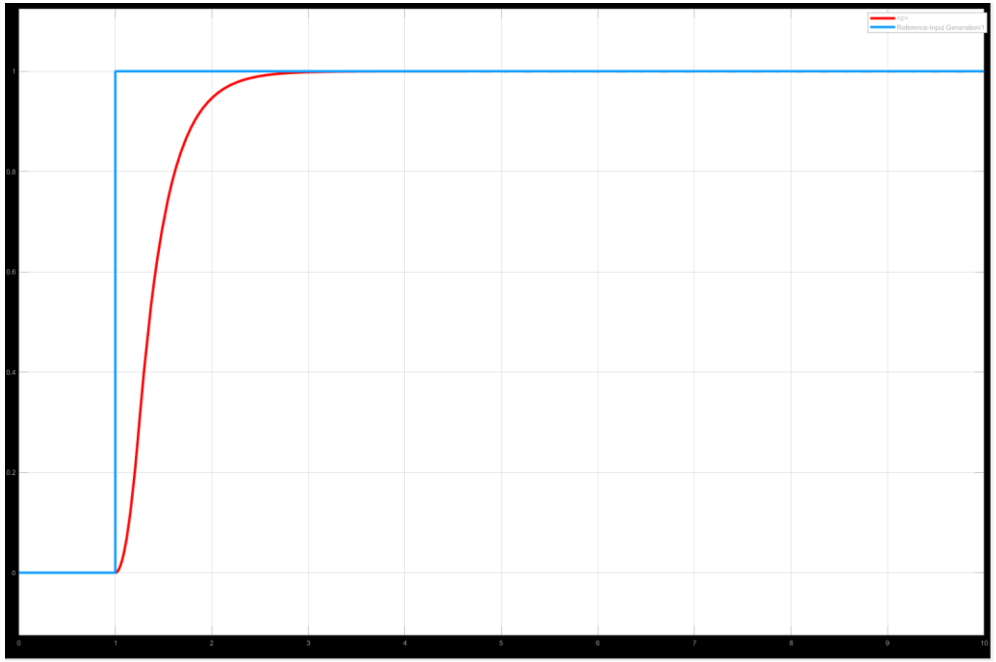
Simulink Model

Design of a Fuzzy-PID Controller for Controlling Quadrotor Attitude and Altitude

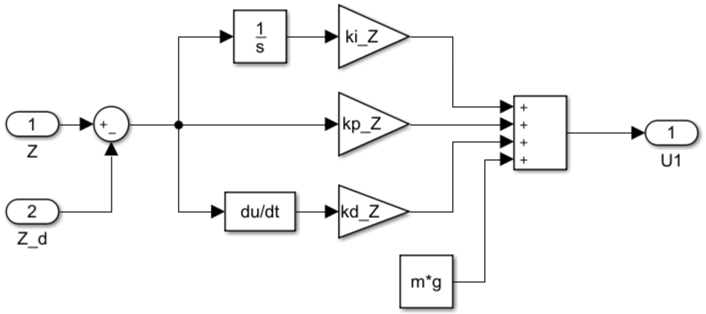


Altitude Controller(Simulation Result)

$$U_1 = k_p(z - z_d) + k_d(\dot{z} - \dot{z}_d) + k_i \int (z - z_d)dt$$

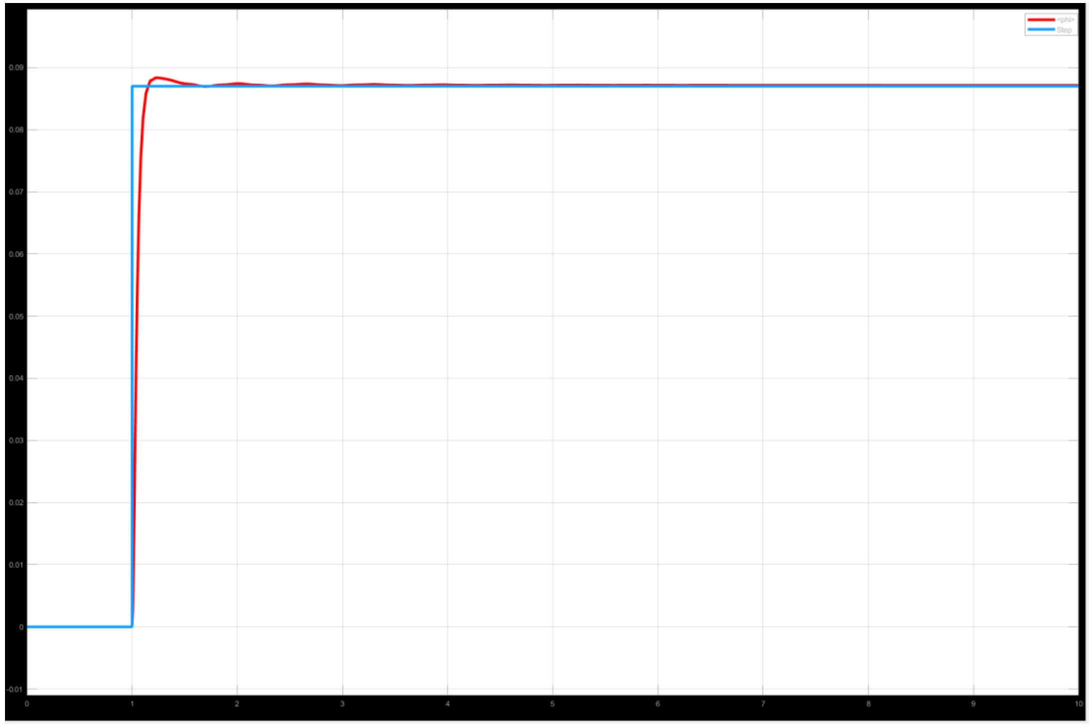


Parameter	Description	Value
k_p	Proportional gain	30
k_i	Integral gain	0
k_d	Derivative gain	10
t_r	Rising time	702.5 ms
t_s	Settling time	1.5 s
%OS	Overshoot	0

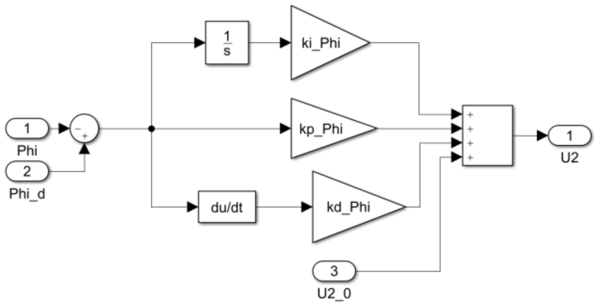


Roll Controller

$$U_2 = k_p(\phi_d - \phi) + k_d(\dot{\phi}_d - \dot{\phi}) + k_i \int (\phi_d - \phi)dt$$

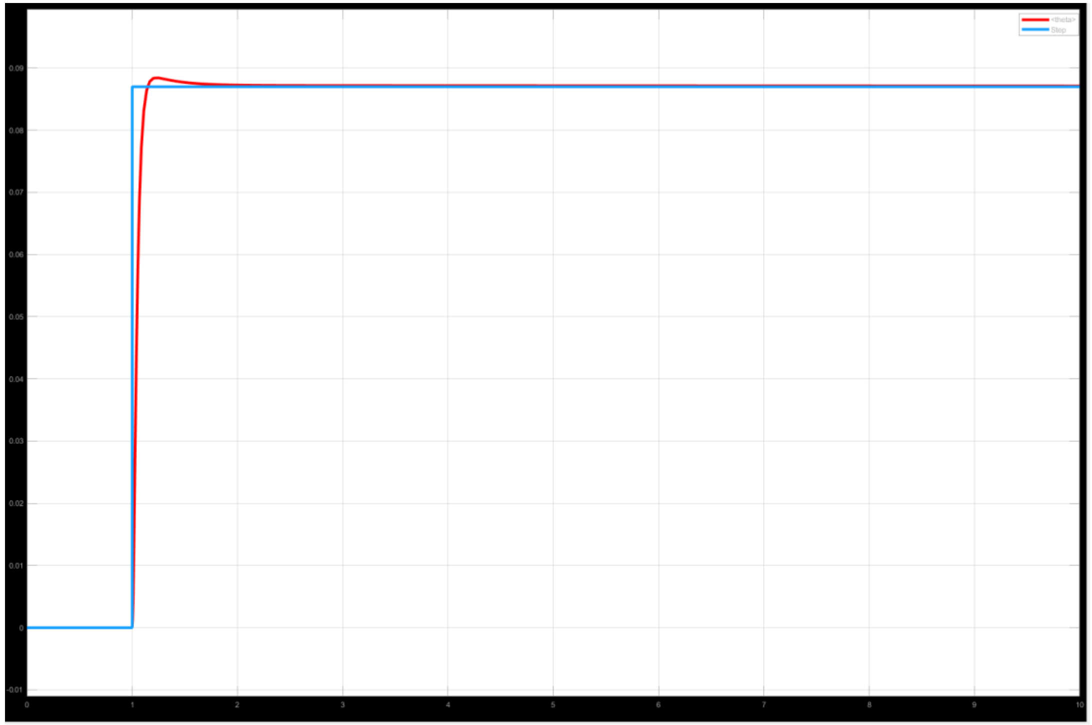


Parameter	Description	Value
k_p	Proportional gain	2.9996
k_i	Integral gain	0.17
k_d	Derivative gain	0.87
t_r	Rising time	110.1 ms
t_s	Settling time	597.5 ms
%OS	Overshoot	1.37

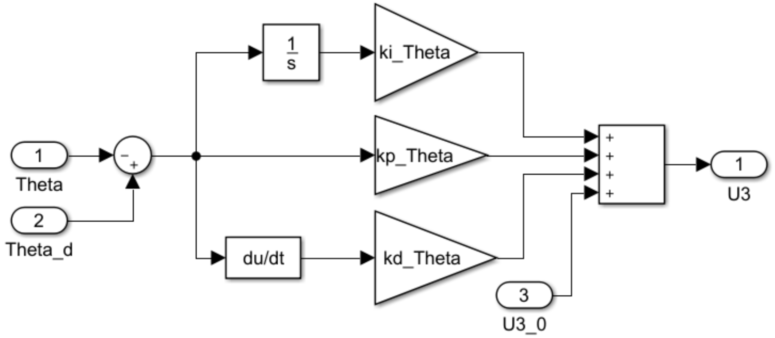


Pitch Controller

$$U_3 = k_p(\theta_d - \theta) + k_d(\dot{\theta}_d - \dot{\theta}) + k_i \int (\theta_d - \theta) dt$$

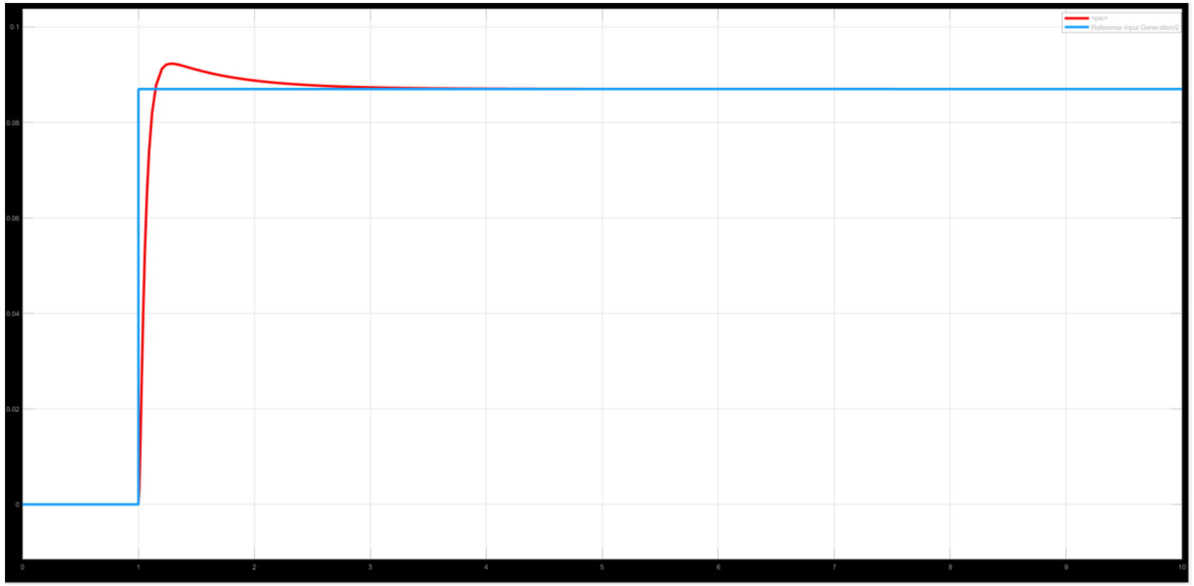


Parameter	Description	Value
k_p	Proportional gain	2.9996
k_i	Integral gain	0.17
k_d	Derivative gain	0.87
t_r	Rising time	110.1 ms
t_s	Settling time	597.5 ms
%OS	Overshoot	1.37

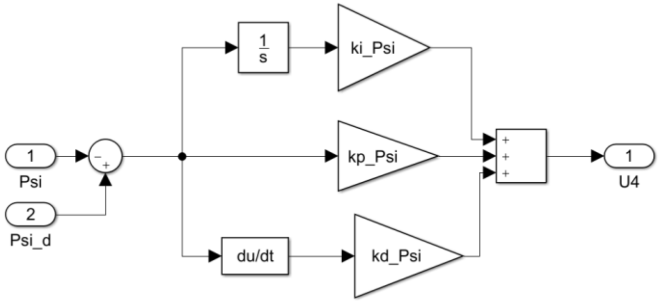


Yaw Controller

$$U_4 = k_p(\psi_d - \psi) + k_d(\dot{\psi}_d - \dot{\psi}) + k_i \int (\psi_d - \psi)dt$$

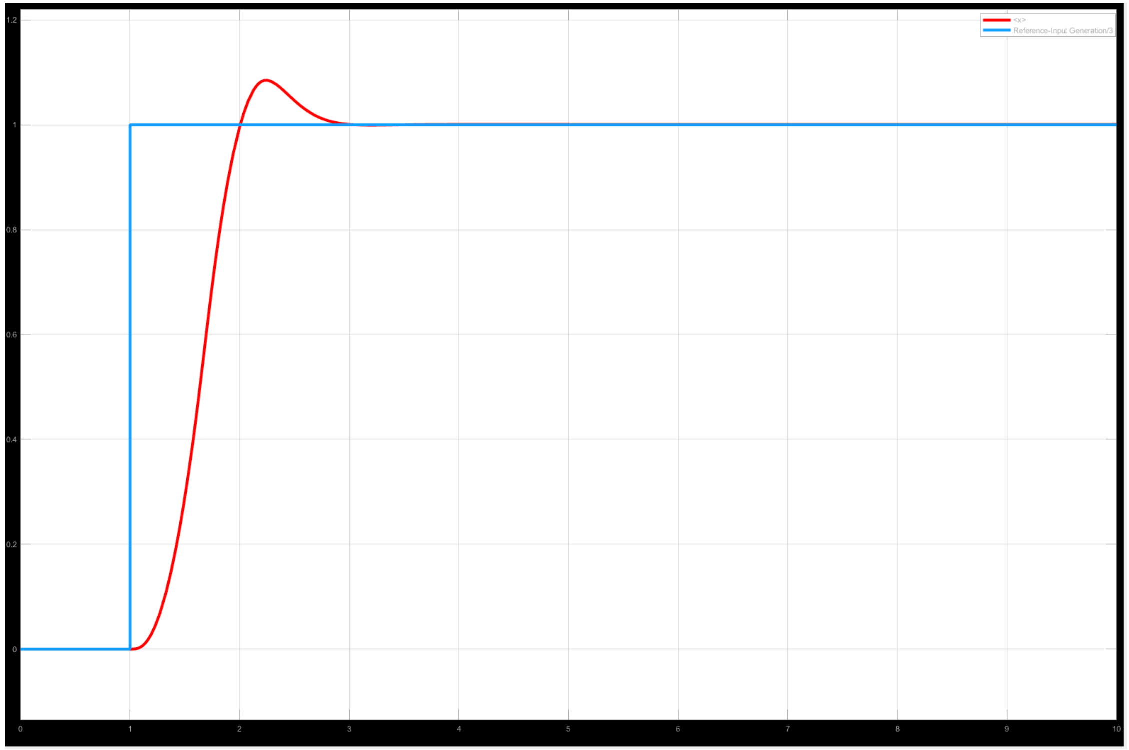


Parameter	Description	Value
k_p	Proportional gain	3.9
k_i	Integral gain	0
k_d	Derivative gain	2.57
t_r	Rising time	94 ms
t_s	Settling time	1.5s
%OS	Overshoot	6.09

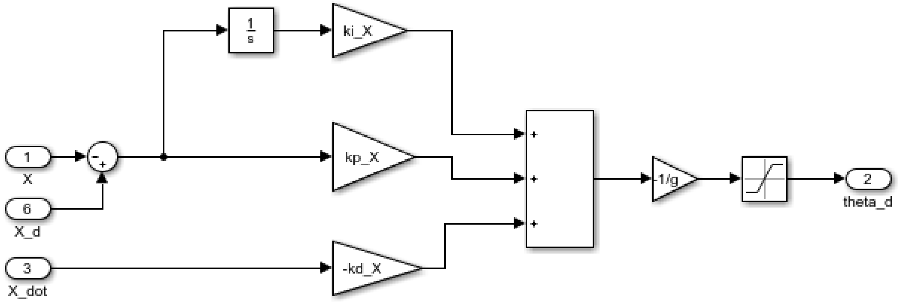


Position Controller (X)

$$\ddot{x}_d = k_p(x_d - x) + k_d(\dot{x}_d - \dot{x}) + k_i \int (x_d - x)dt$$

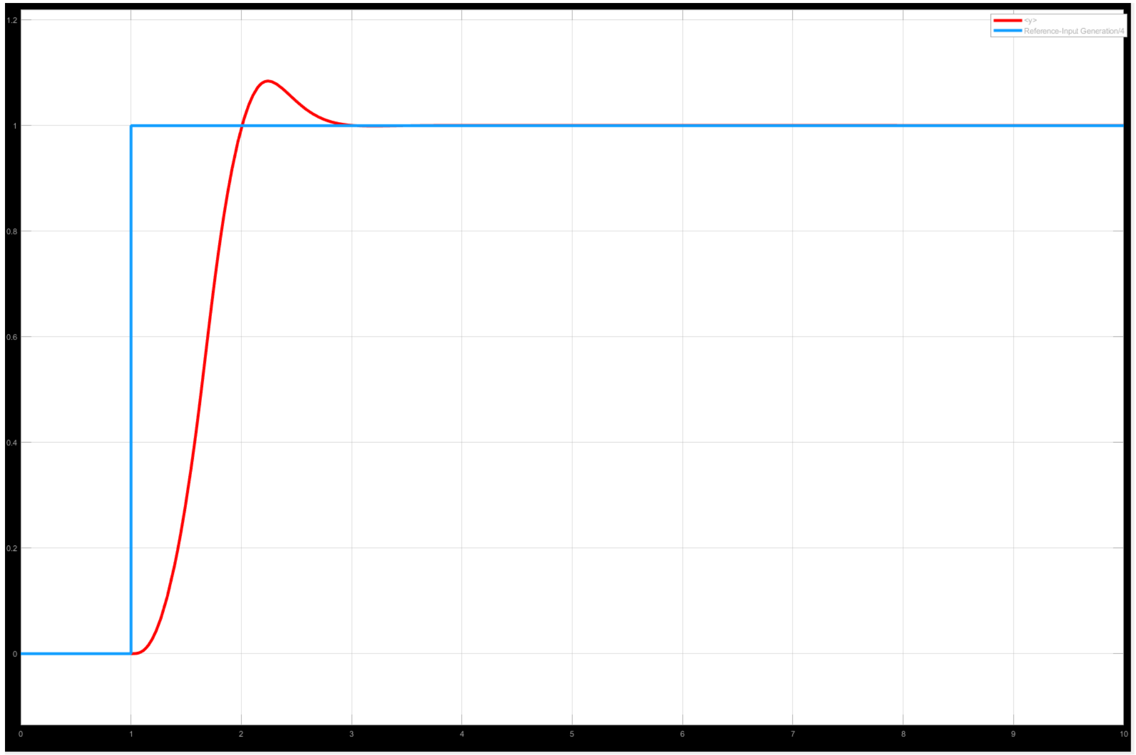


Parameter	Description	Value
k_p	Proportional gain	19.5
k_i	Integral gain	0
k_d	Derivative gain	5.7
t_r	Rising time	592.8 ms
t_s	Settling time	1.83 s
%OS	Overshoot	8.42

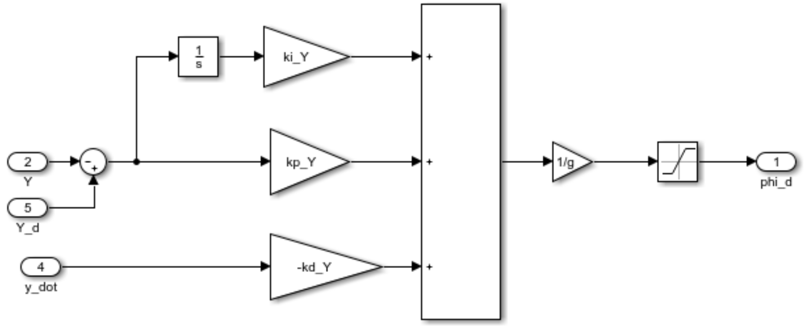


Position Controller (Y)

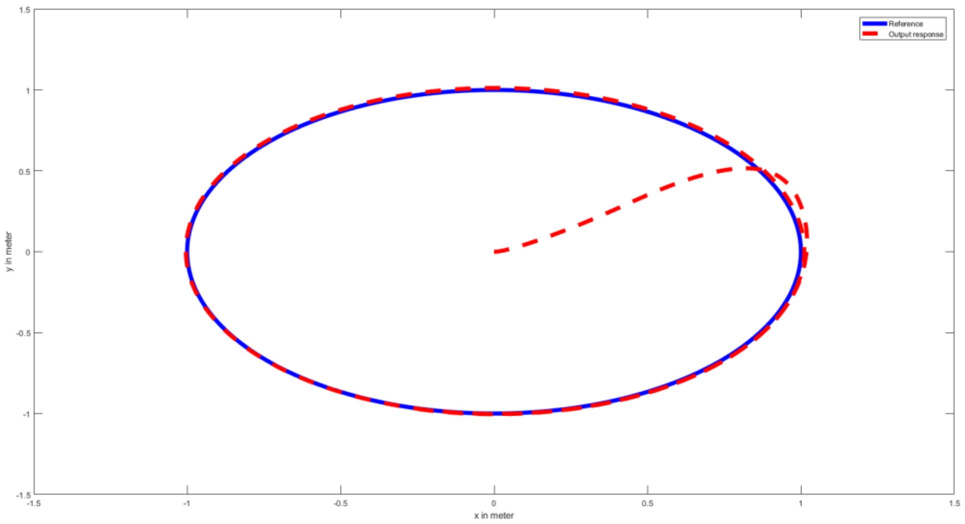
$$\ddot{y}_d = k_p(y_d - y) + k_d(\dot{y}_d - \dot{y}) + k_i \int (y_d - y)dt$$



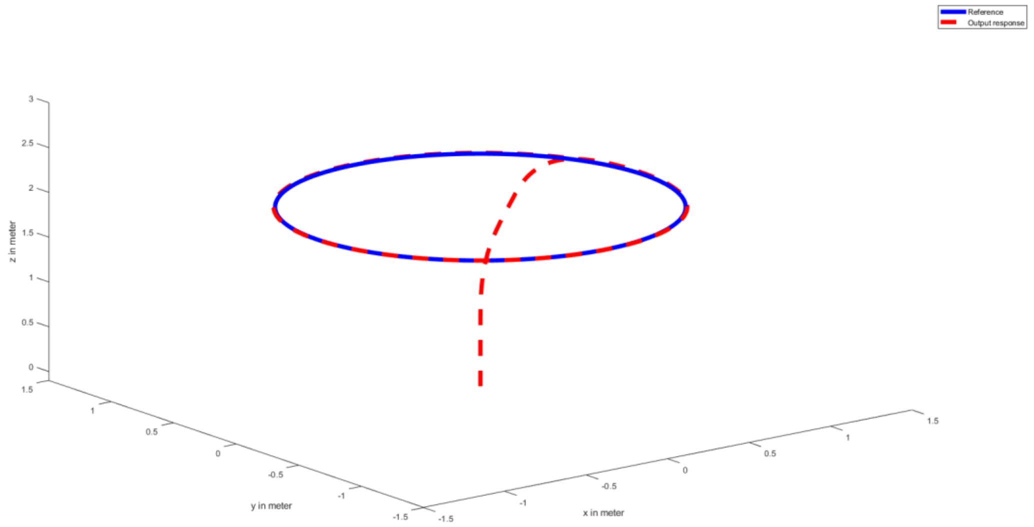
Parameter	Description	Value
k_p	Proportional gain	19.5
k_i	Integral gain	0
k_d	Derivative gain	5.7
t_r	Rising time	592.8 ms
t_s	Settling time	1.83 s
%OS	Overshoot	8.42



Simulation of Quadrotor on a Circular Trajectory

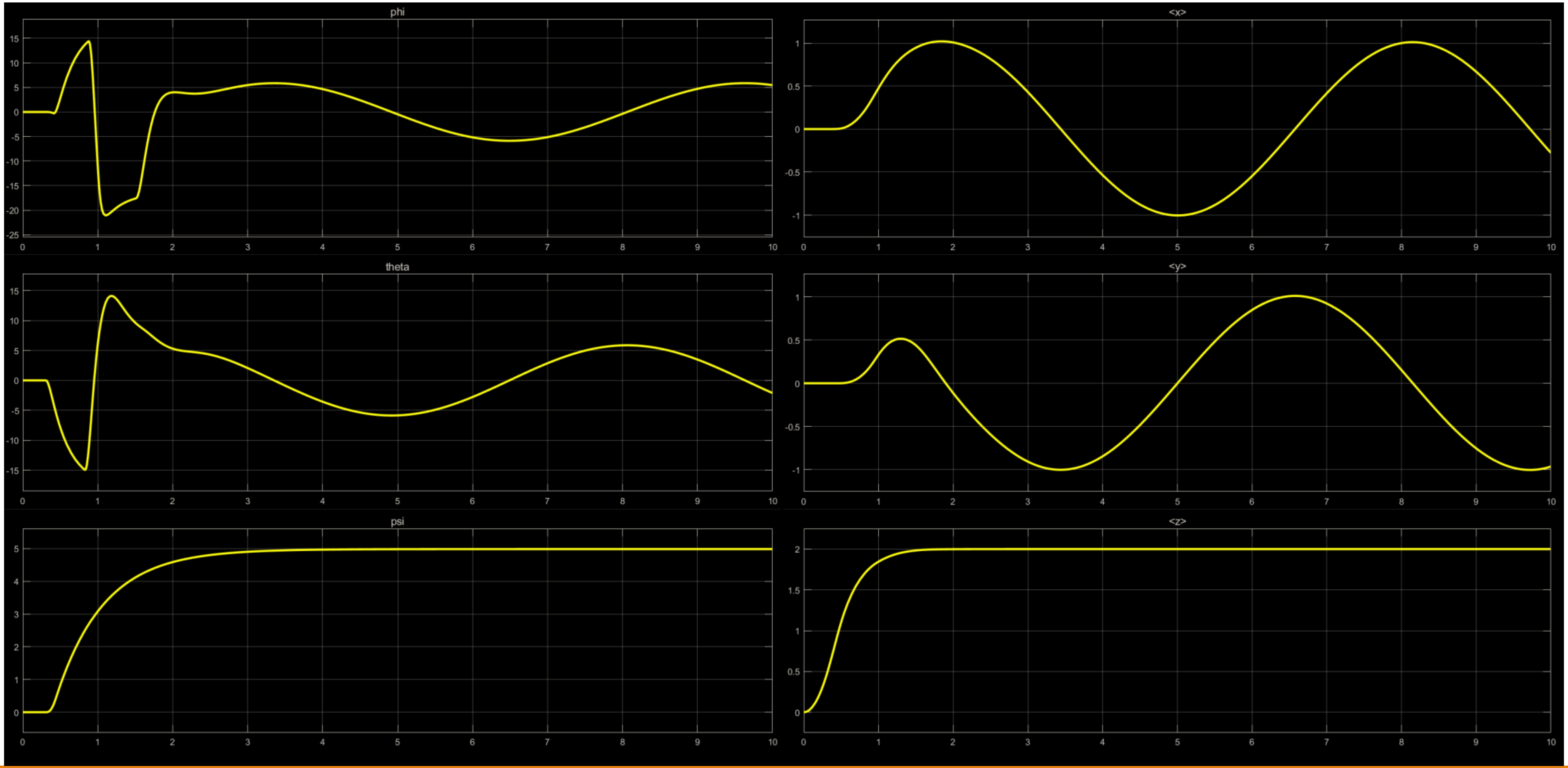


Plot of the Trajectory in 2d

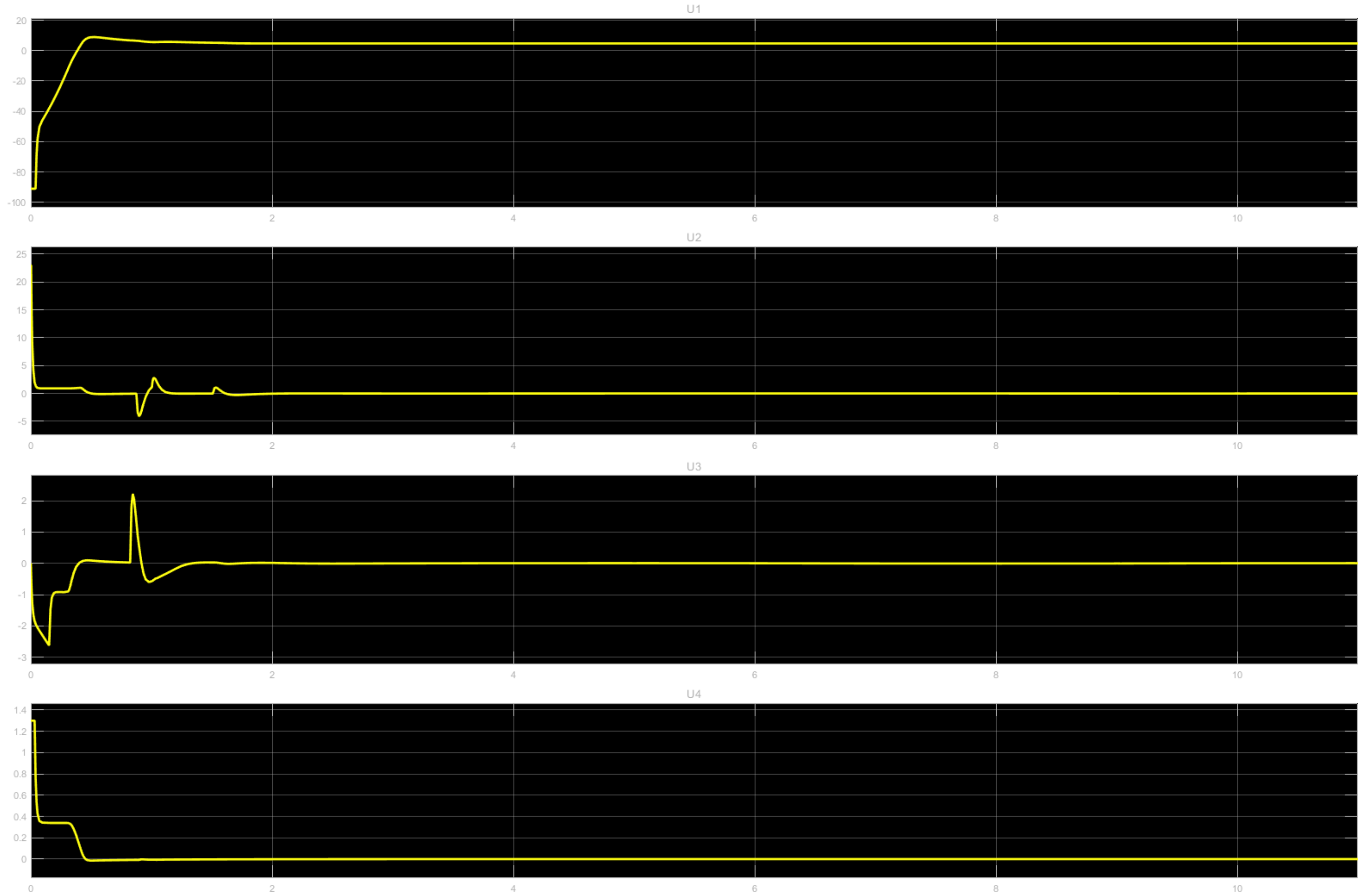


Plot of the Trajectory in 3d

Simulation of Quadrotor on a Circular Trajectory



Control Input Signals



Tuning of Conventional PID Controller

$$u(t) = K_p\ e(t) + K_i \int_0^t e(x).\,dx + K_d \frac{de(t)}{dt}$$

formulas based on
ultimate gain (K_u) and ultimate period (T_u)

Controller	Gain (K_P)	Integral time(T_i)	Derivative time(T_d)
P	$0.5K_u$	-	-
PI	$0.45K_u$	$0.8T_u$	-
PID	$0.6K_u$	$0.5T_u$	$0.125T_u$

Principle of Fuzzy Logic Controller (FLC)

FLC implementations require the following:

1) Fuzzification

2) Knowledge Base

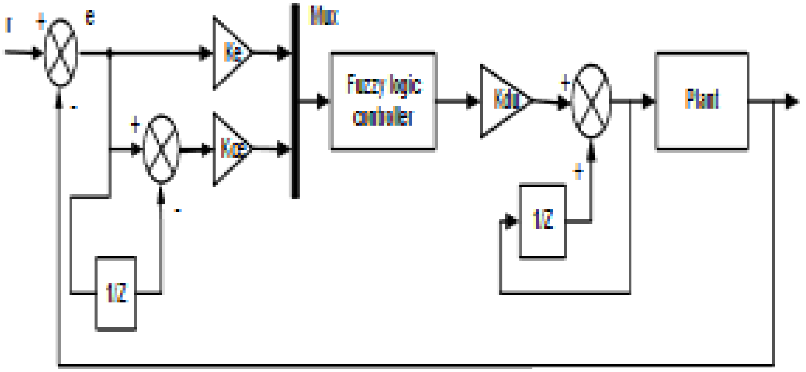
   a. Data Base

   b. Rule Base

3) Fuzzy inference system

4) Defuzzifcation

Design and Tuning Sample



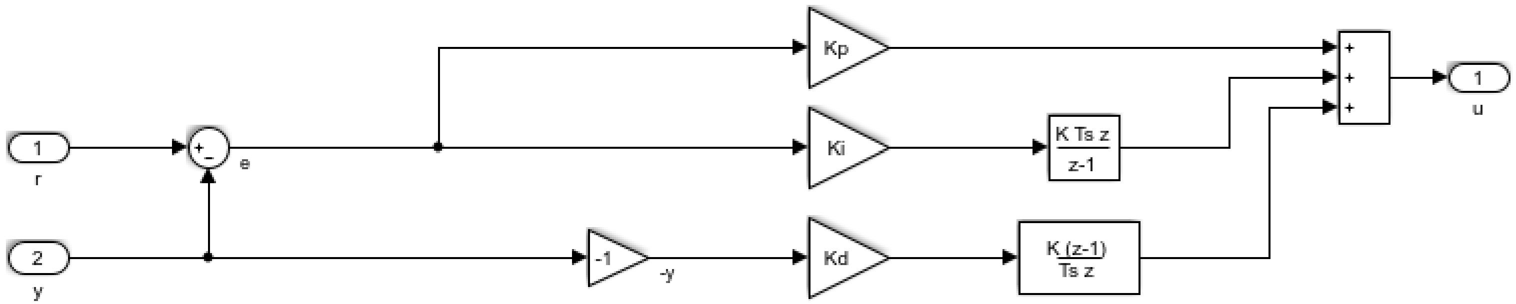
Basic rules table for fuzzy inference system

e \ ce	NB	NS	Z	PS	PB
NB	NVB	NB	NM	NS	Z
NS	NB	NM	NS	Z	PS
Z	NM	NS	Z	PS	PM
PS	NS	Z	PS	PM	PB
PB	Z	PS	PM	PB	PVB

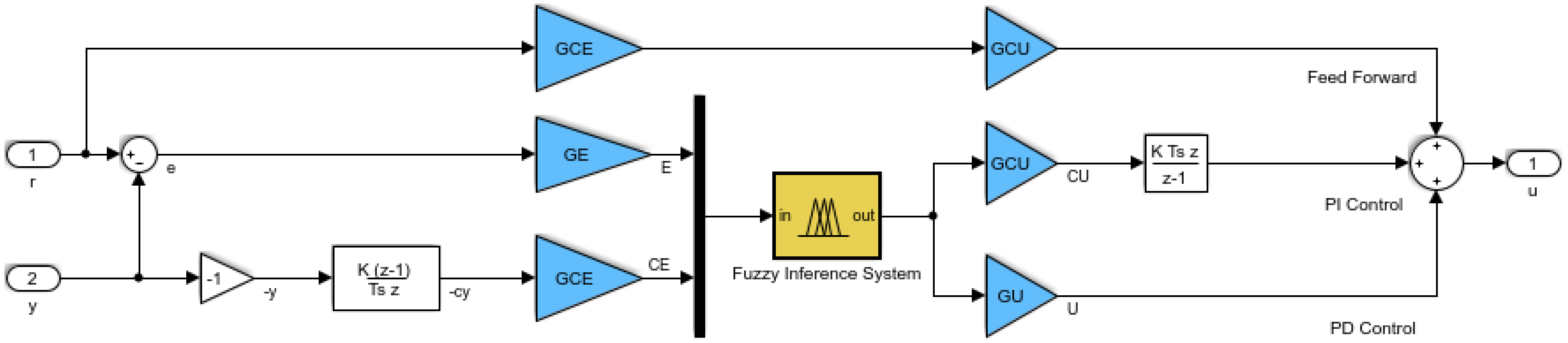
Meaning of the linguistic variables in the fuzzy inference system

NVB	Negative very big
NB	Negative big
NM	Negative medium
NS	Negative small
Z	Zero
PS	Positive small
PM	Positive medium
PB	Positive big
PVB	Positive very big

Conventional PID vs Fuzzy PID (Structure)



Conventional PID structure



Fuzzy PID structure

Relation between Conventional PID gain and Fuzzy PID gain

$$GE = \frac{1}{max.error}$$

$$GU = \frac{K_{df}}{GCE}$$

$$GCE = GE * (K_{pf} - \sqrt{K_{pf}^2 - 4K_{if}K_{df}}\frac{K_{if}}{2})$$

$$GCU = \frac{K_{if}}{GE}$$

Where,

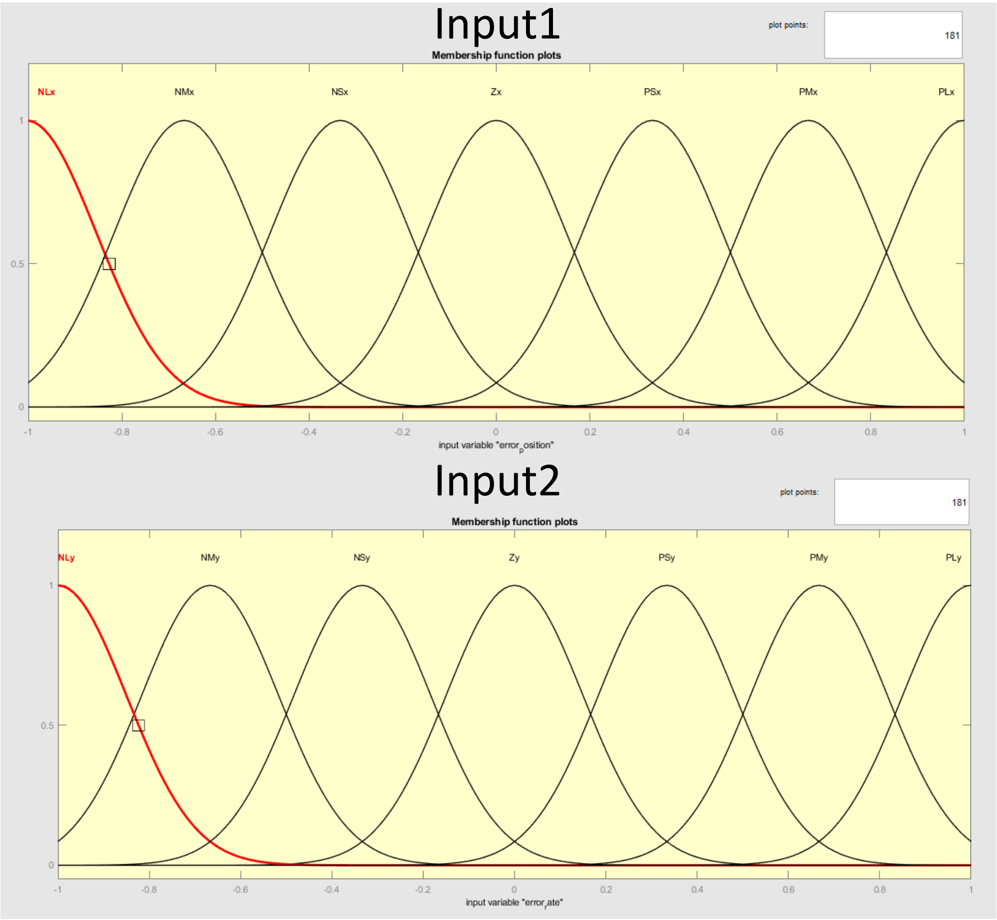
GE -> Error Normalization factor

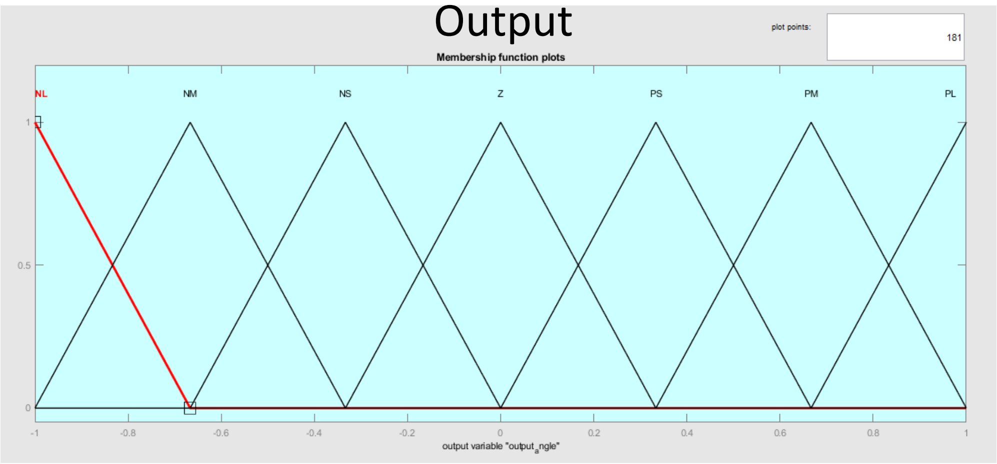
GCE -> Change in measurement normalization factor

GU -> response de-normalization factor

GCU -> change in the response de-normalization factor

Fuzzy Membership Function





Fuzzy Inference Rules

In1/In2	NLy	NMy	NSy	Zy	PSy	PMy	PLy
NLx	NL	NL	NL	NL	NM	NS	Z
NMx	NL	NL	NL	NM	NS	Z	PS
NSx	NL	NL	NM	NS	Z	PS	PM
Zx	NL	NM	NS	Z	PS	PM	PL
PSx	NM	NS	Z	PS	PM	PL	PM
PMx	NS	Z	PS	PM	PL	PL	PL
PLx	Z	PS	PM	PL	PL	PL	PL

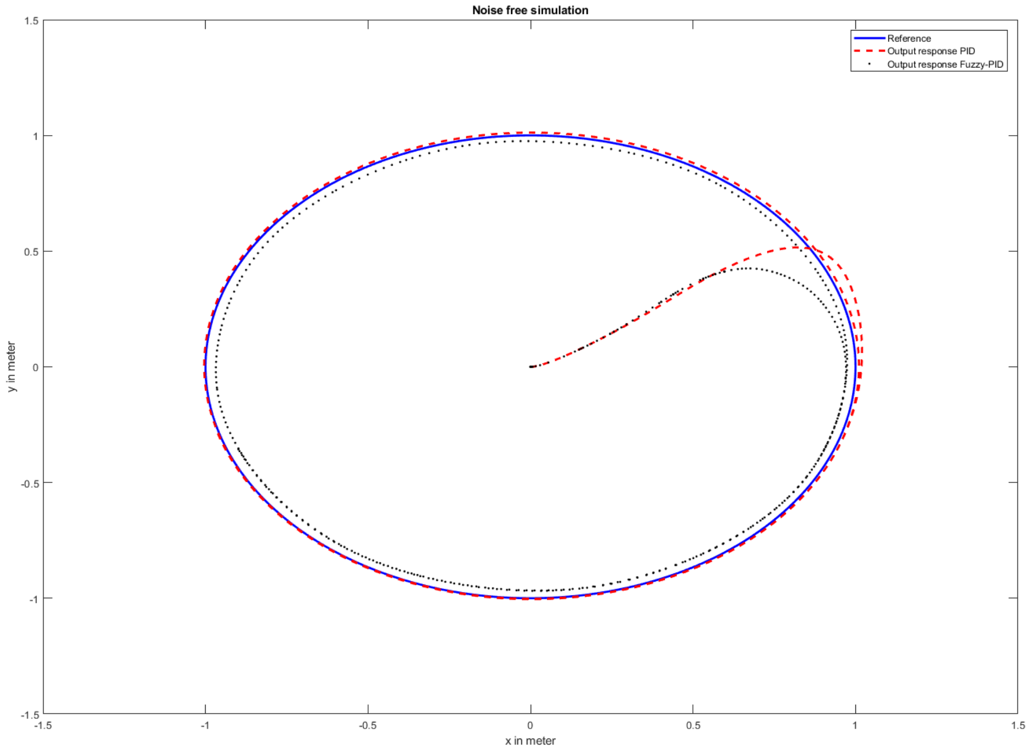
In1 = derivative value/error position; In2 = error rate

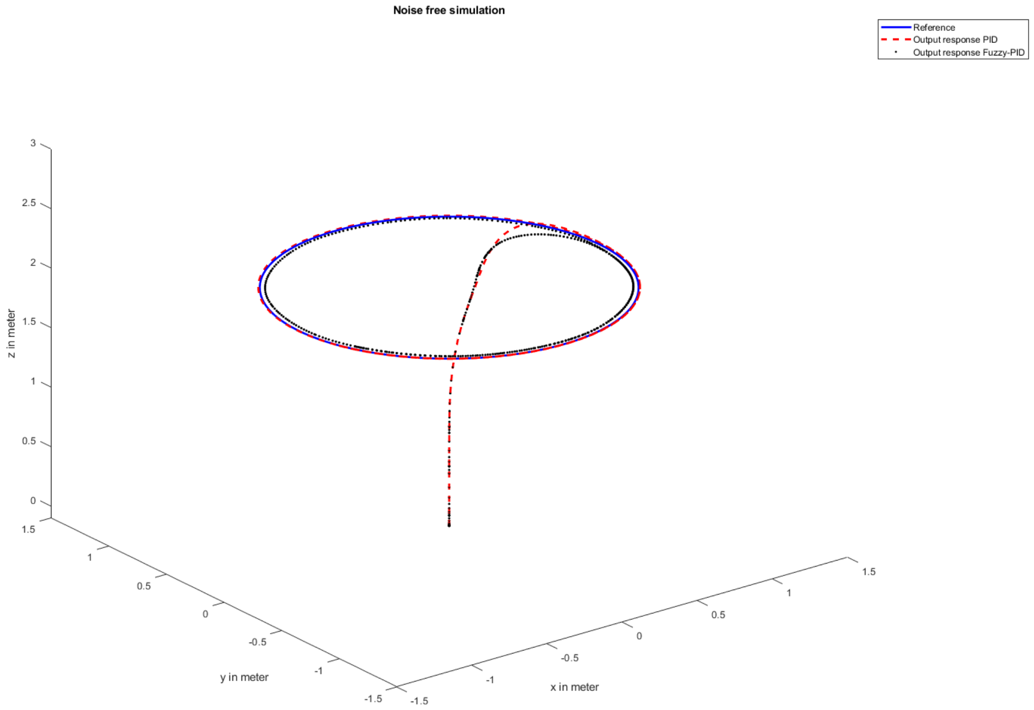
Linguistic Variable	Meaning
NL	Negative Large
NM	Negative Medium
NS	Negative Small
Z	Zero
PS	Positive Small
PM	Positive Medium
PL	Positive Large

Parameters for fuzzy-PID controller

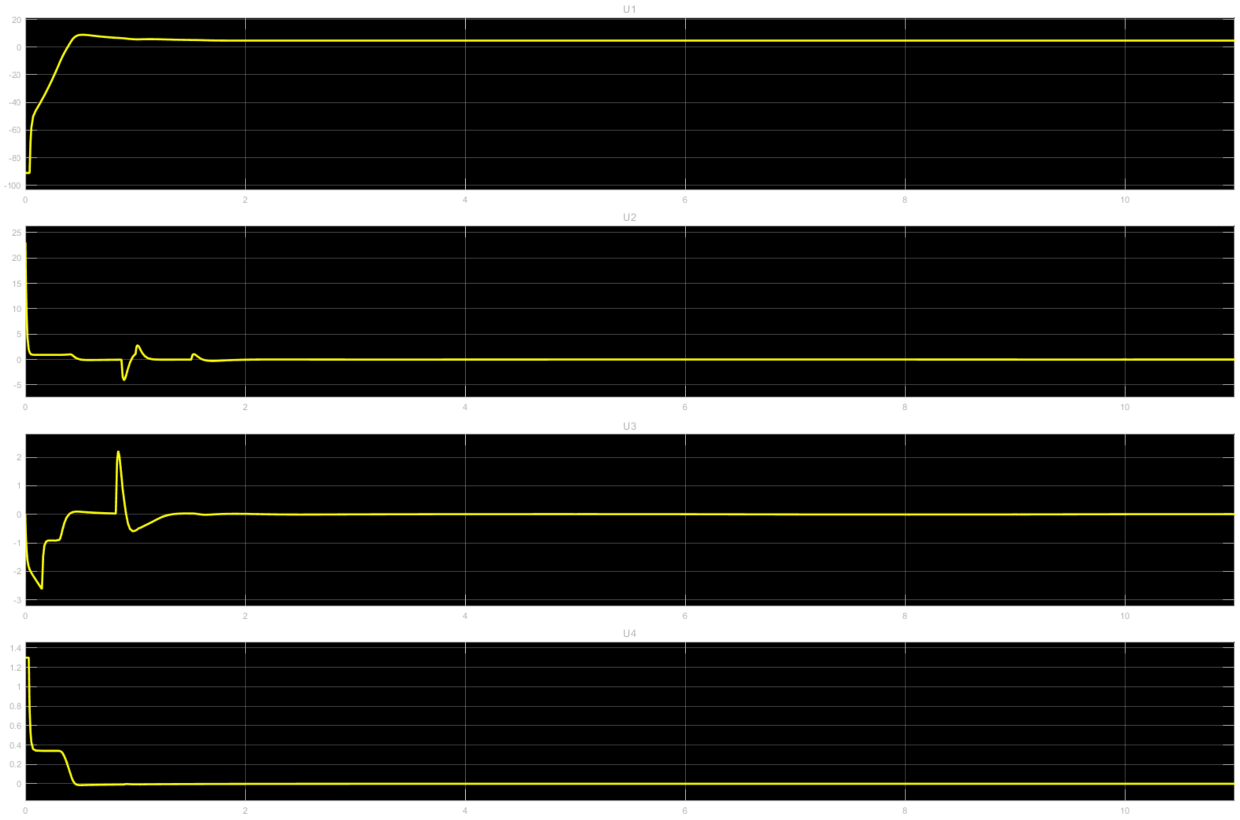
Fuzzy-PID Gain	Value
GE	1.15
GCE	0.45
GU	1.97
GCU	0

Plot of Trajectory (No Noise)

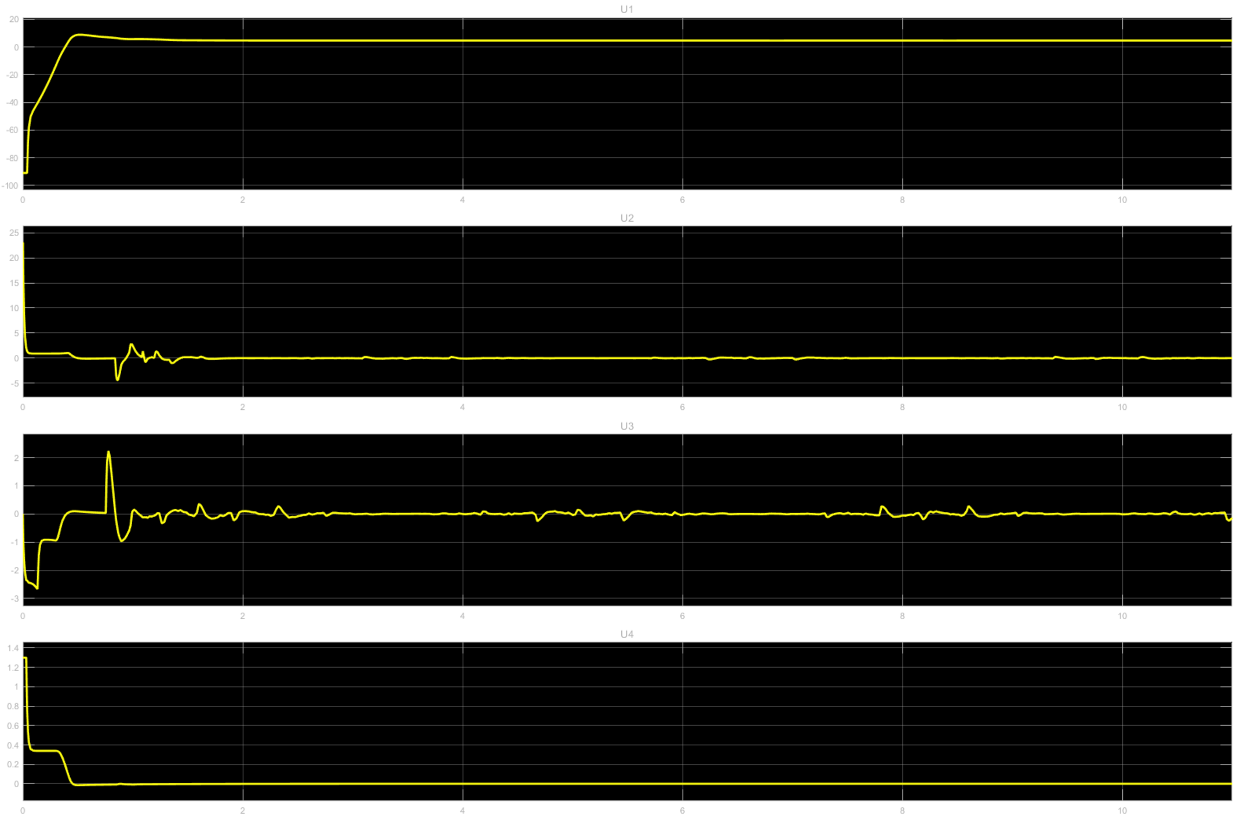




Plot of Input Signal(No Noise)



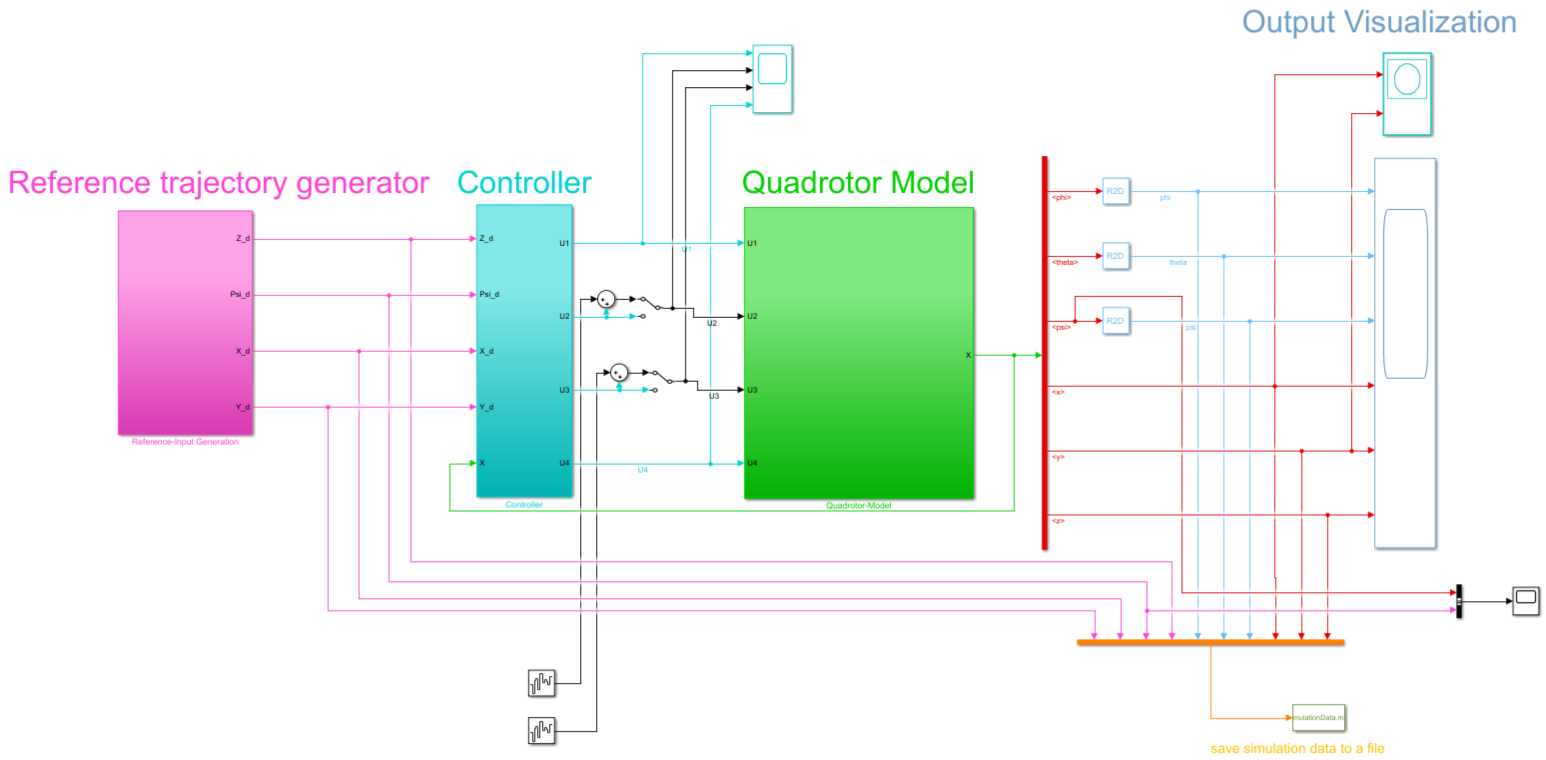
PID



Fuzzy PID

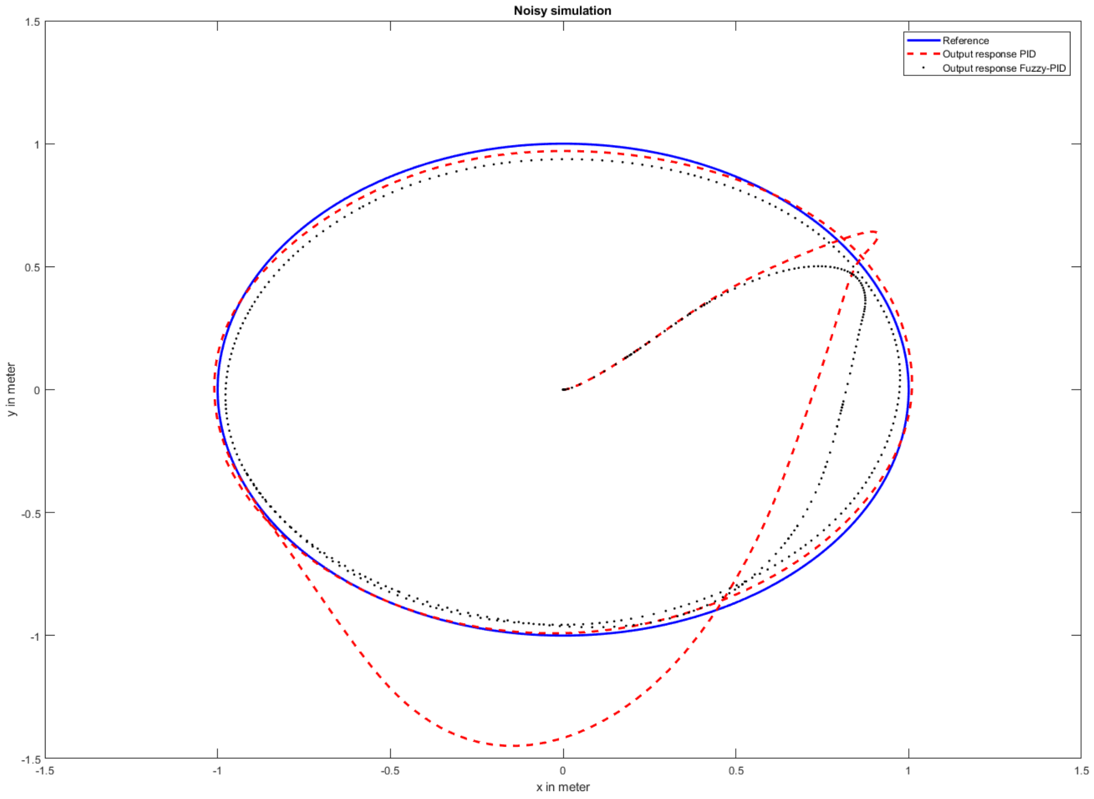
Noisy Simulation Block Diagram

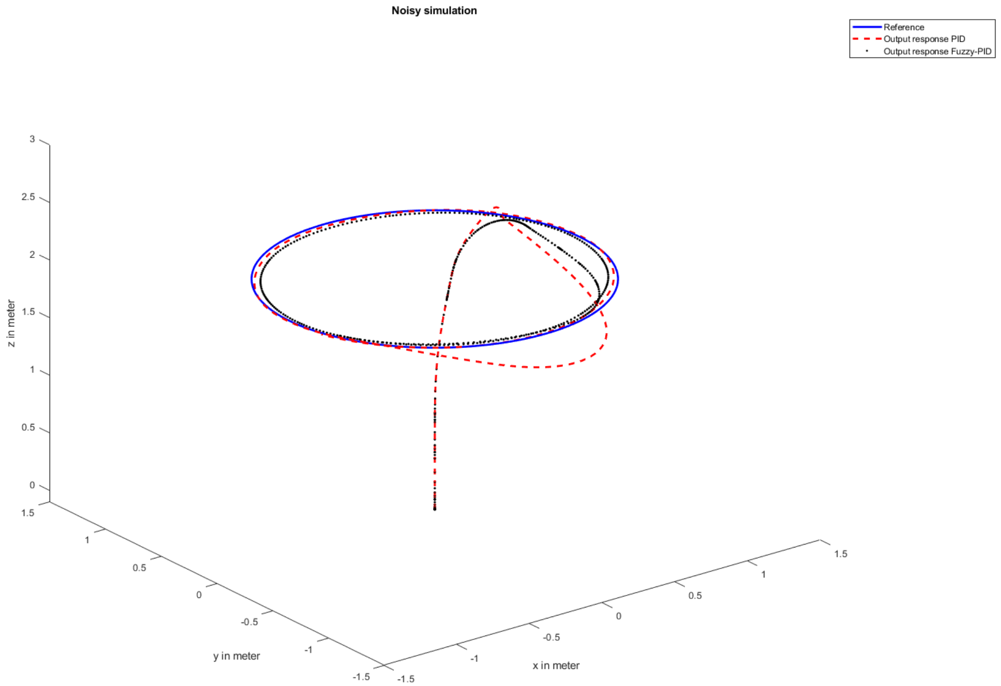
Design of a Fuzzy-PID Controller for Controlling Quadrotor Attitude and Altitude



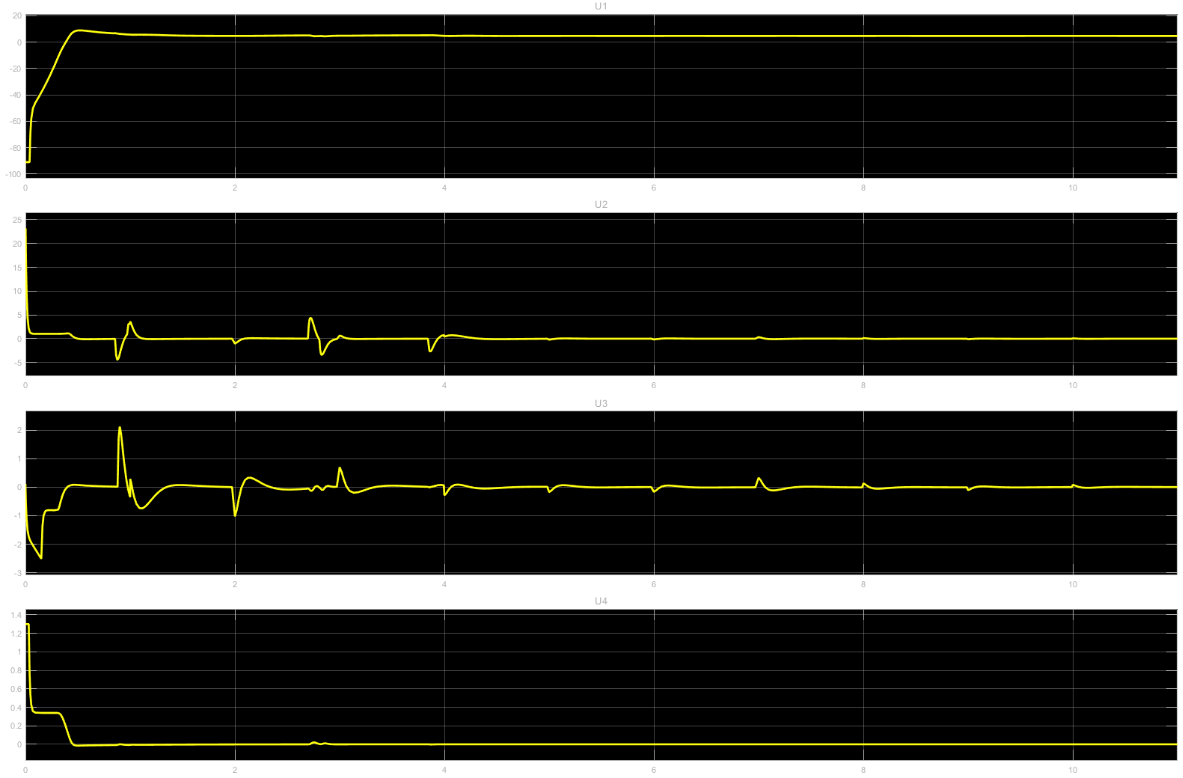
N.B. : Band Limited White Noise power 0.1

Plot of Trajectory (Noisy)

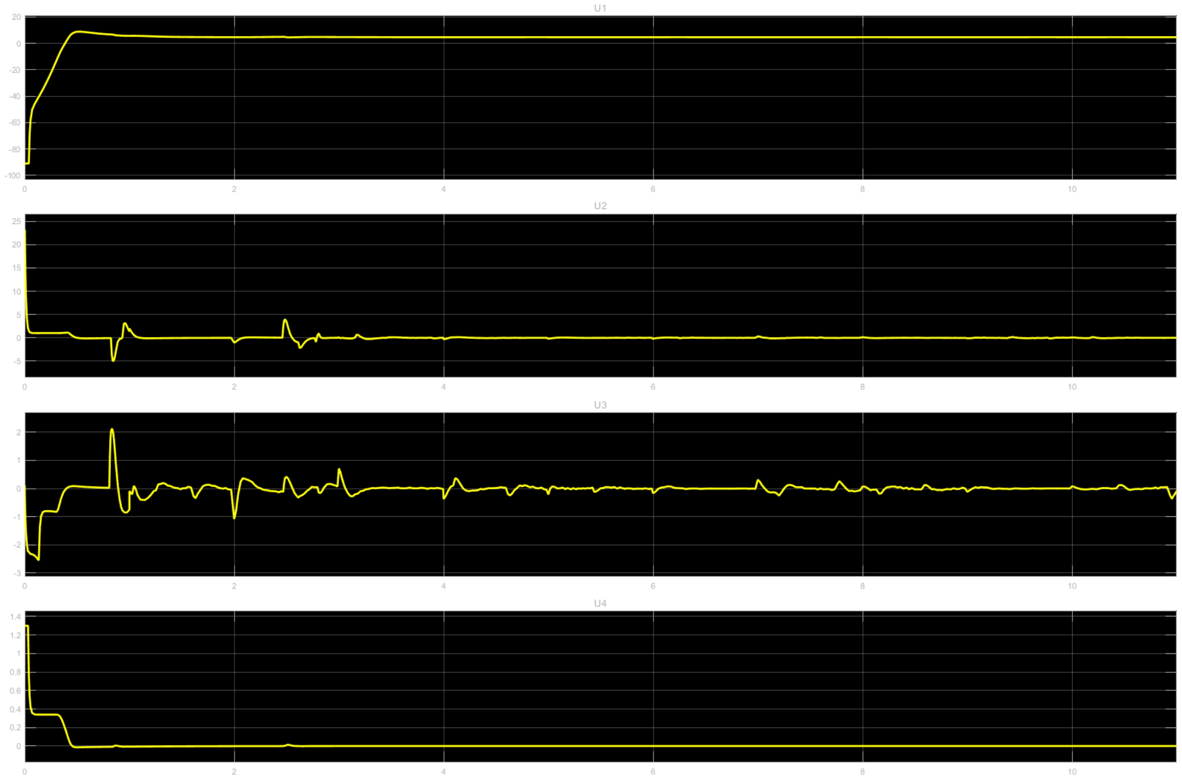




Plot of Input Signal(Noisy)



PID



Fuzzy PID

Conclusion

- Studied the mathematical modelling of quadcopter
- Developed conventional PID controller for controlling attitude and altitude of a quadcopter
- Simulated the system with conventional PID controller in Simulink
- Developed Fuzzy-PID controller to control the position of the quadcopter
- Simulated the system with Fuzzy-PID controller
- After adding noise, the system is simulated using both PID and fuzzy-PID controller

Remarks:

- With proper tuning of gains in noise free system the performance of conventional PID and Fuzzy PID is same
- But in noisy system Fuzzy PID is better than conventional PID

Thank You