Method#02 - PID control: Tuning PID controller of the LTI, SISO system

Let's consider the following UAV stabilization system

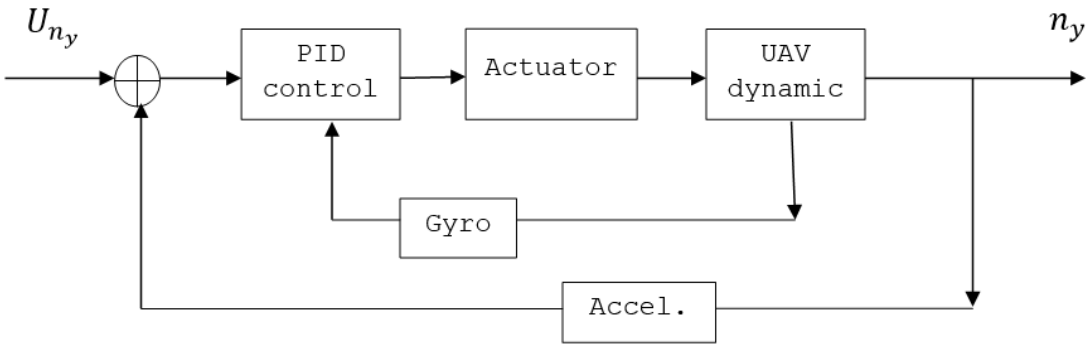


Fig.2.01 - Functional block-diagram of the UAV stabilization system

Assumptions

Measurement noise & errors of the Gyro and Accelerometer aren't taking into account in the model: $W_{gyro}(s)=1,\ W_{accel}(s)=1$.

…

PID controller

$$\delta(t) = K_P e(t) + K_D \omega_z(t) + K_I \int_0^T e(t)dt\ , \qquad (2.01)$$

Actuator

$$W_{act} = \frac{1}{T_{act}s+1}, \qquad (2.02)$$

where $T_{act} = \frac{1}{K_{act}}$ is actuator time constant, $K_{act} = 20$.

UAV dynamics

$$W_{\delta}^{\omega_z} = \frac{K(T_1 s+1)}{T_2^2 s^2 + 2\xi T_2 + 1},\ W_{\omega_z}^{\dot{\theta}} = \frac{1}{T_1 s+1},\ W_{\dot{\theta}}^{n_y} = \frac{V}{g}, \qquad (2.03)$$

Where

$K = 1,$

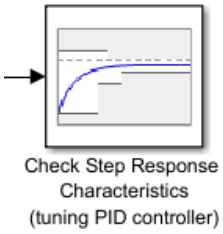
$T_1 = 0.7\ (s),\ T_2 = 0.5\ (s),$

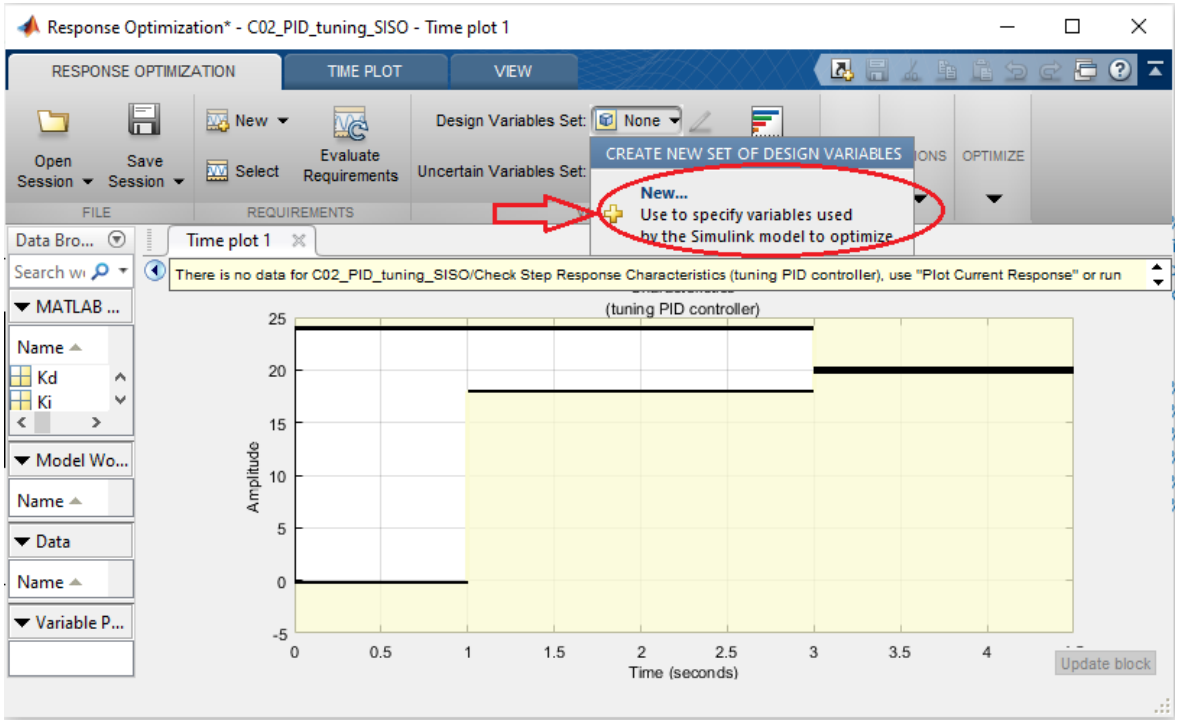
$\xi = 0.3.$

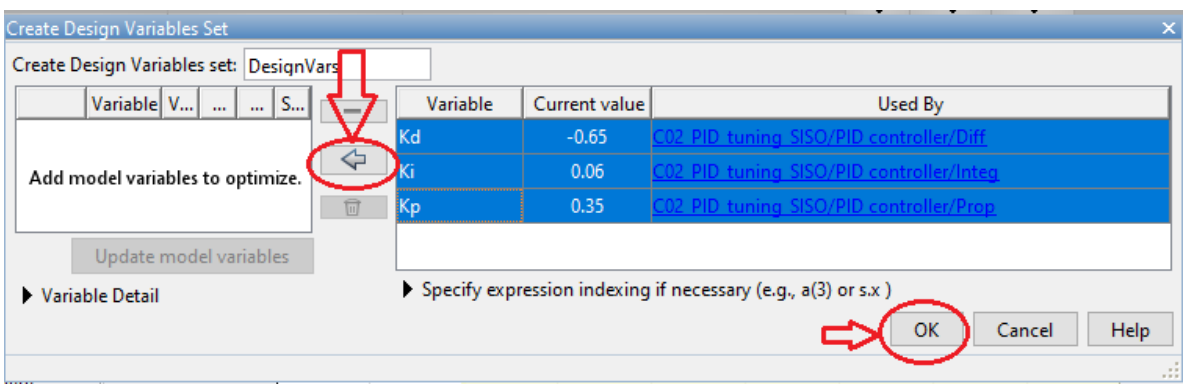
1st step - Initial PID coefficients load into Workspace

```
>> clear all, close all
>> uiopen('D:\! MATLAB\!GitHub\Control\!done\2_PID\C02_PID_tuning_SISO.slx',1)
>> Kp = 0.35; Kd = -0.65; Ki = 0.06;
>>
```

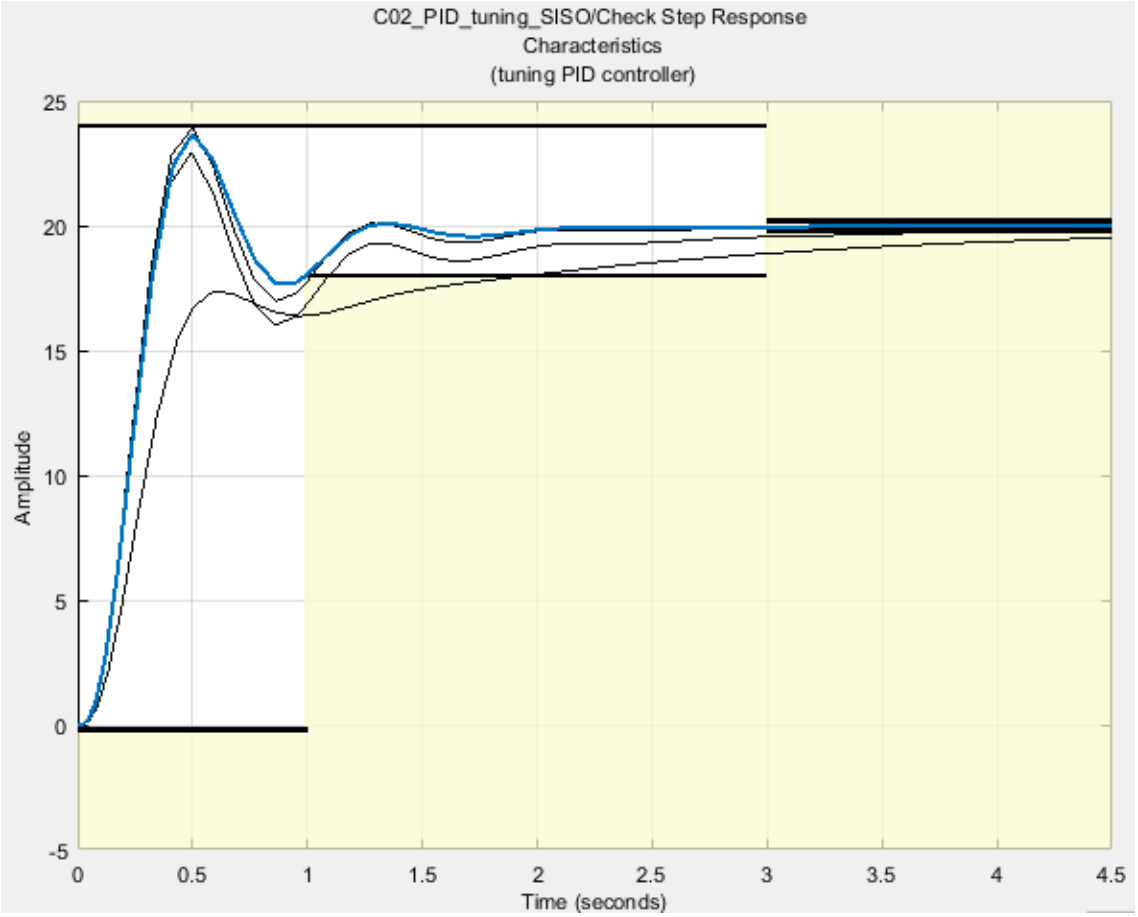
2nd step – Response Optimization setting







3rd step – Optimize



4th step – Analysis of the optimization results