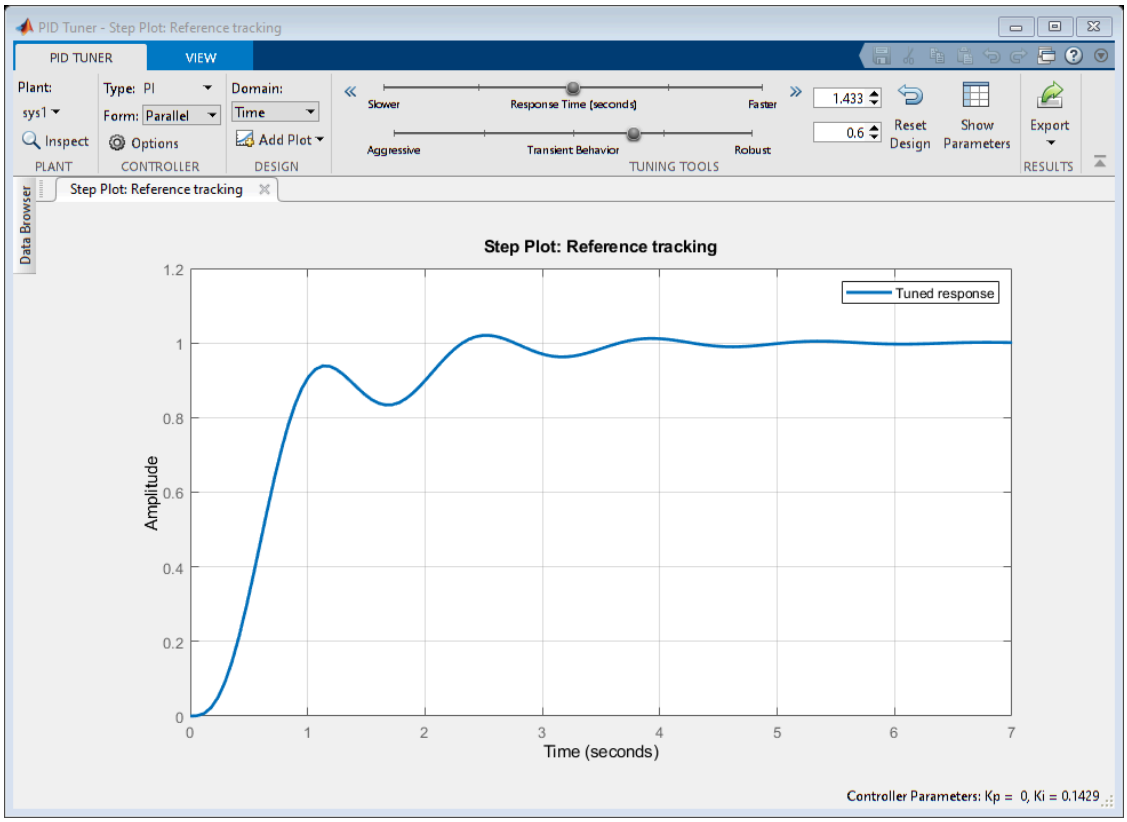
Table of Contents

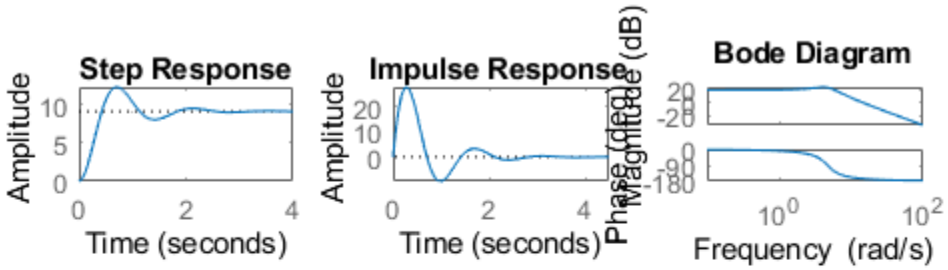
normal

```
J1 = 0.01;
b1 = 0.01;
K1 = 0.1;
R1 = 0.1;
L1 = 0.05;
sys1 = tf([K1/(J1*L1)],[1,((b1/J1)+(R1/L1)),(((K1*K1)+(R1*b1))/
(L1*J1))])
subplot(4,3,1)
step(sys1)
subplot(4,3,2)
impulse(sys1)
subplot(4,3,3)
S = stepinfo(sys1)
pzmap(sys1)
 pidTuner(sys1)
 bode(sys1)
```

```
sys1 =

       200
  --------------
  s^2 + 3 s + 22

Continuous-time transfer function.
```

```
S =

  struct with fields:

        RiseTime: 0.2882
    SettlingTime: 2.3810
     SettlingMin: 8.0006
     SettlingMax: 12.2393
       Overshoot: 34.6325
      Undershoot: 0
            Peak: 12.2393
        PeakTime: 0.7061
```

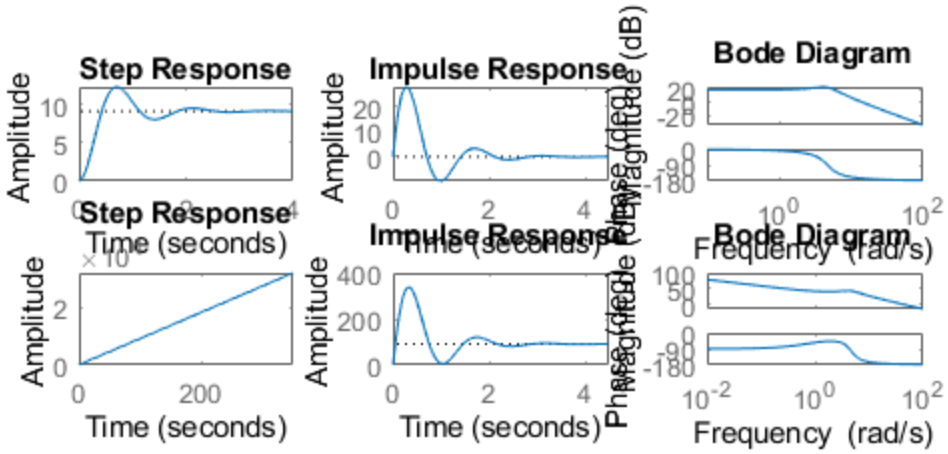

PID Tuner - Step Plot: Reference tracking
PID TUNER
VIEW
Plant:
sys1
Inspect
PLANT
Type: PI
Form: Parallel
Options
CONTROLLER
Domain:
Time
Add Plot
DESIGN
Slower
Response Time (seconds)
Faster
Aggressive
Transient Behavior
Robust
TUNING TOOLS
1.433
0.6
Reset Design
Show Parameters
Export
RESULTS
Data Browser
Step Plot: Reference tracking
Step Plot: Reference tracking
Tuned response
Amplitude
Time (seconds)
Controller Parameters: Kp = 0, Ki = 0.1429

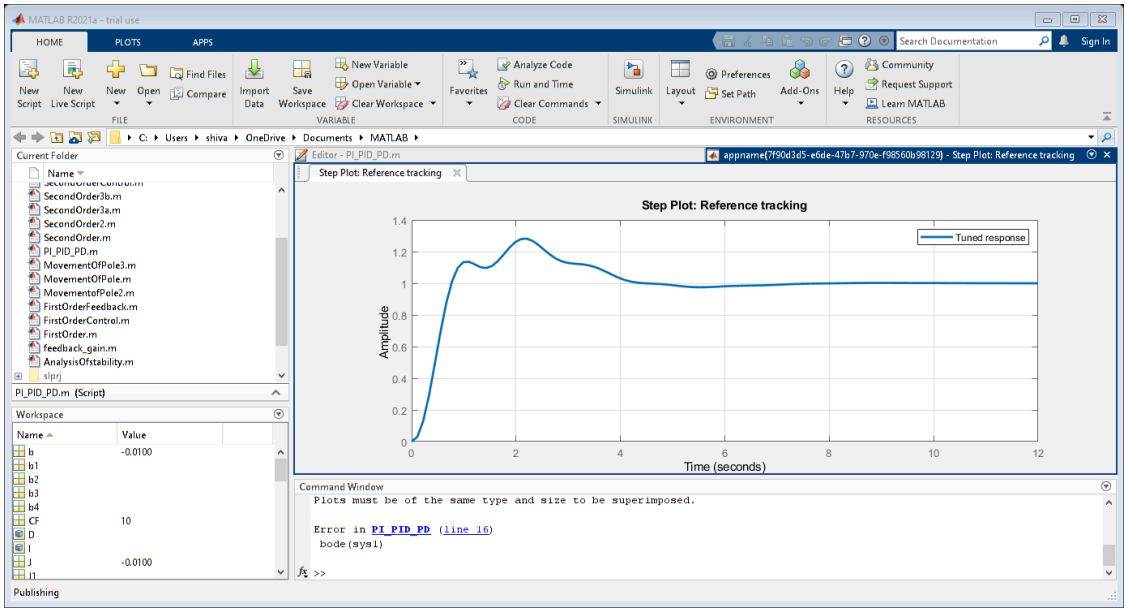

Step Response
Amplitude
Time (seconds)
Impulse Response
Amplitude
Time (seconds)
Bode Diagram
Magnitude (dB)
Phase (deg)
Frequency (rad/s)

pi

```
J2 = 0.01;
b2 = 0.01;
K2 = 0.1;
R2 = 0.1;
L2 = 0.05;
Kp=10;
I=tf([10],[1,0]);  %Ki
PI=Kp+I;
sys2 = tf([K2/(J2*L2)],[1,((b2/J2)+(R2/L2)),(((K2*K2)+(R2*b2))/
(L2*J2))])*(PI)
subplot(4,3,4)
step(sys2)
subplot(4,3,5)
impulse(sys2)
subplot(4,3,6)
S = stepinfo(sys2)
pzmap(sys2)
 pidTuner(sys2)
bode(sys2)
```

```
sys2 =

    2000 s + 2000
  ------------------
  s^3 + 3 s^2 + 22 s

Continuous-time transfer function.


S =

  struct with fields:

        RiseTime: NaN
    SettlingTime: NaN
     SettlingMin: NaN
     SettlingMax: NaN
       Overshoot: NaN
      Undershoot: NaN
            Peak: Inf
        PeakTime: Inf
```





PD

```
J3 = 0.01;
b3 = 0.01;
K3 = 0.1;
R3 = 0.1;
```

```
L3 = 0.05;
Kp=10;
D=tf([10,1],[0,1]);  %Kd
PD=Kp+D;
sys3 = tf([K3/(J3*L3)],[1,((b3/J3)+(R3/L3)),(((K3*K3)+(R3*b3))/
(L3*J3))])*(PD)
subplot(4,3,7)
step(sys3)
subplot(4,3,8)
impulse(sys3)
subplot(4,3,9)
S = stepinfo(sys3)
pzmap(sys3)
 pidTuner(sys3);
 bode(sys3)
```

```
sys3 =

  2000 s + 2200
  --------------
  s^2 + 3 s + 22

Continuous-time transfer function.


S =

  struct with fields:

        RiseTime: 0.0426
    SettlingTime: 2.7143
     SettlingMin: 14.7945
     SettlingMax: 346.0086
       Overshoot: 246.0086
      Undershoot: 0
            Peak: 346.0086
        PeakTime: 0.3377
```

```
Error using javaMethodEDT
The second argument to javaMethod must be a Java object, or a
 character vector or string naming the class for a static method.

Error in ctrluis.toolstrip.StatusMessage (line 40)
            this.StatusBar = javaMethodEDT('getStatusBar', Frame);

Error in pidtool.PIDToolDesktop/open (line 112)
         this.StatusBar = ctrluis.toolstrip.StatusMessage(Frame);

Error in pidtool.PIDToolDesktop (line 75)
         this.open();

Error in pidtool (line 112)
```

```
      eval(cmd);

Error in pidTuner (line 91)
        eval(cmd);

Error in PI_PID_PD (line 59)
 pidTuner(sys3);
```

PID

```
J4 = 0.01;
b4 = 0.01;
K4 = 0.1;
R4 = 0.1;
L4 = 0.05;
Kp=10;
D=tf([10,1],[0,1]);  %Kd
I=tf([10],[1,0]);  %Ki
PID=Kp+D+I;
sys4 = tf([K4/(J4*L4)],[1,((b4/J4)+(R4/L4)),(((K4*K4)+(R4*b4))/
(L4*J4))])*(PID)
subplot(4,3,10)
step(sys4)
subplot(4,3,11)
impulse(sys4)
subplot(4,3,12)
S = stepinfo(sys4)
pzmap(sys4)
 pidTuner(sys4)
 bode(sys4)
```

Published with MATLAB® R2021a