Demo 1 Project Documentation

Table of Contents

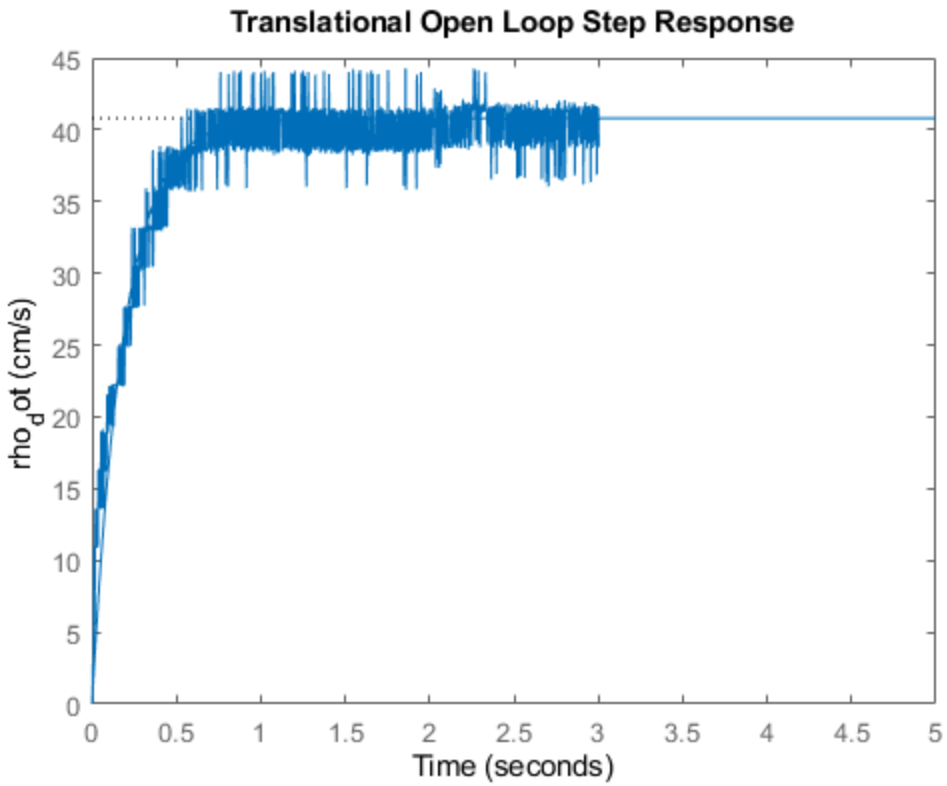
Group 10

Trevor Bachand, Andrew Burton, William Peyton, Kyra Squier

Translational Open Loop Step Response

Graphs the experimental step response and the estimated transfer function step response.

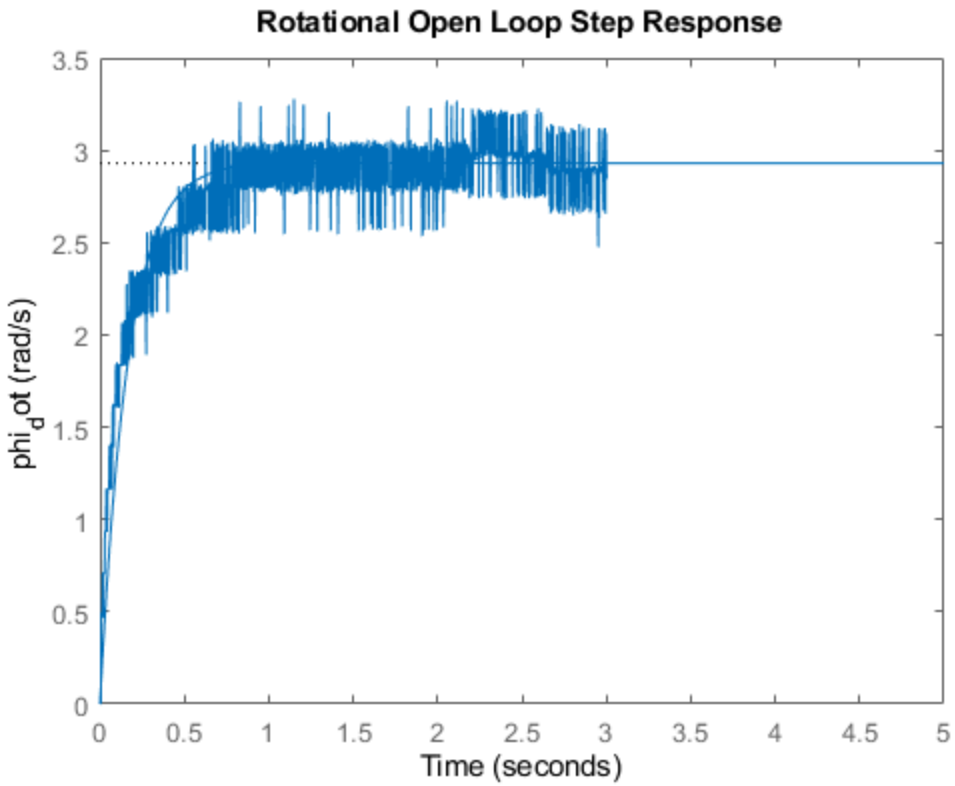
```
%experimental
[V,T,VT] = xlsread('RobotOpenLoopResponse.xlsx','Translational');
t = V(:,1);
y = V(:,8);

plot(t,y)

%transfer function
K_rho = 40.8;       %0.16
sigma_rho = 5.305;
sys = tf(K_rho*sigma_rho, [1  sigma_rho]);

hold on
step(sys)
xlim([0 5])
xlabel('Time')
ylabel('rho_dot (cm/s)')
title('Translational Open Loop Step Response')
hold off
```



Rotational Open Loop Step Response

Graphs the experimental step response and the estimated transfer function step response.

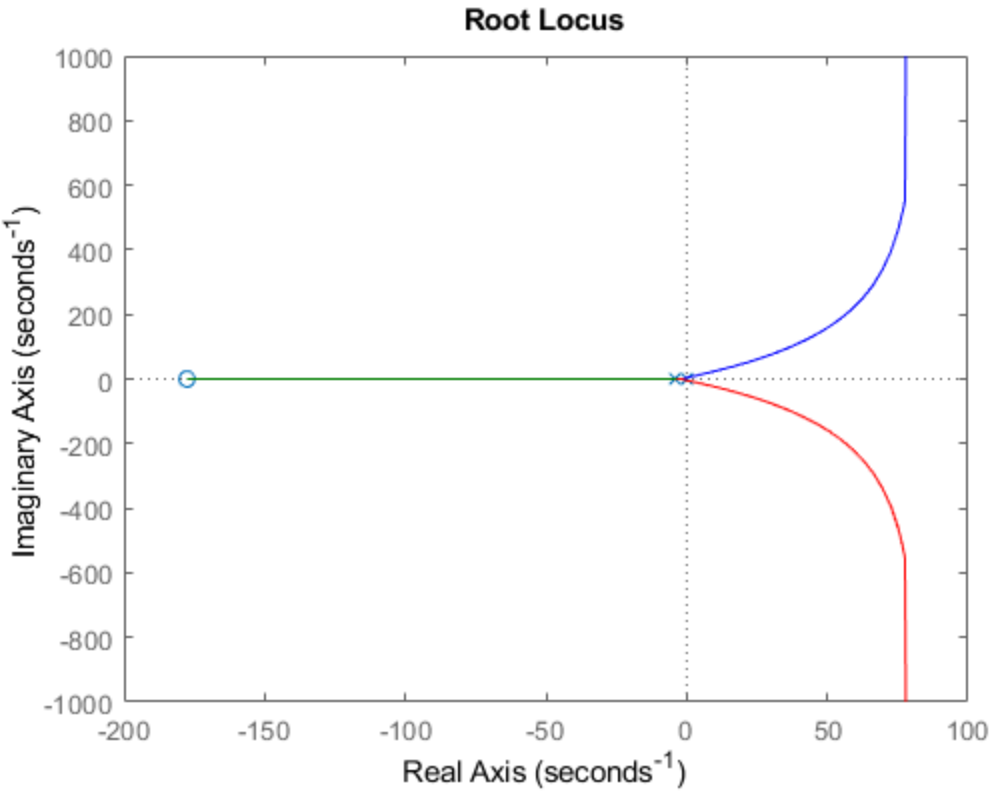
```
%experimental
[W,R,WR] = xlsread('RobotOpenLoopResponse.xlsx','Rotational');
x = W(:,1);
v = W(:,9);

plot(x,v)

%transfer function
K_phi = 2.93;     %0.0115
sigma_phi = 6.061;
sys2 = tf(K_phi*sigma_phi, [1  sigma_phi]);

hold on
step(sys2)
xlim([0 5])
xlabel('Time')
ylabel('phi_dot (rad/s)')
title('Rotational Open Loop Step Response')
hold off
```



Rotational Outer Loop Response

Determines the gain of the outer loop controller by plotting the root locus.

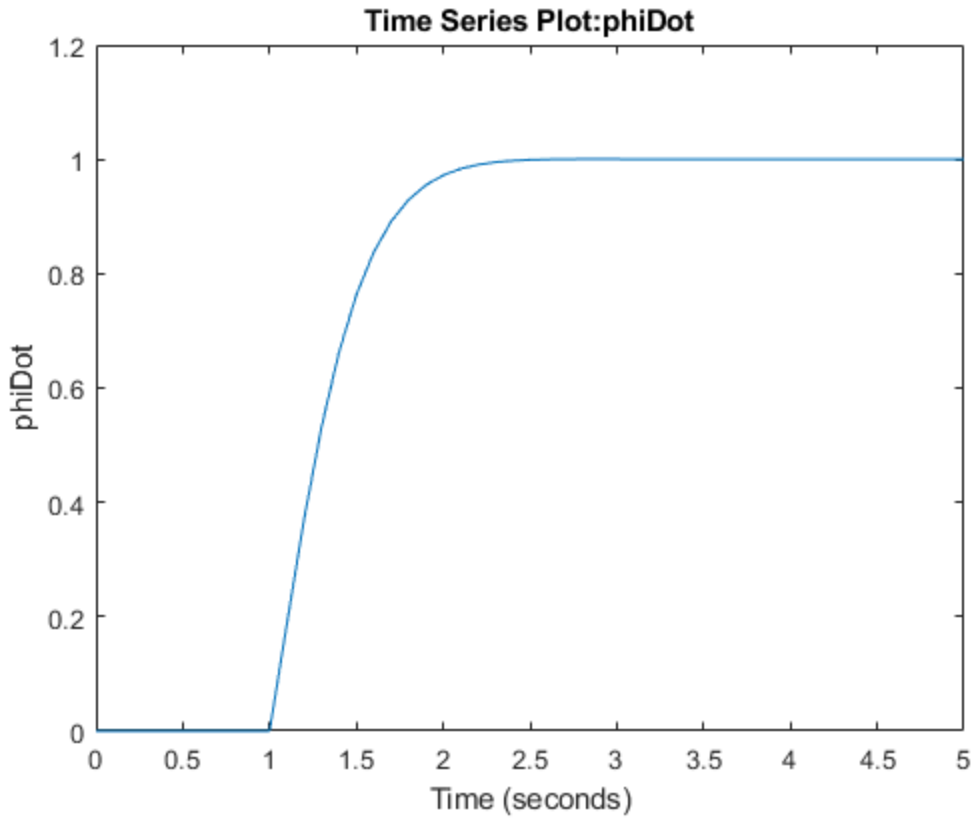
```
sys3 = tf([0.1 17.759],[1 7.837 17.759 0]);
rlocus(sys3)
hold on
hold off
```



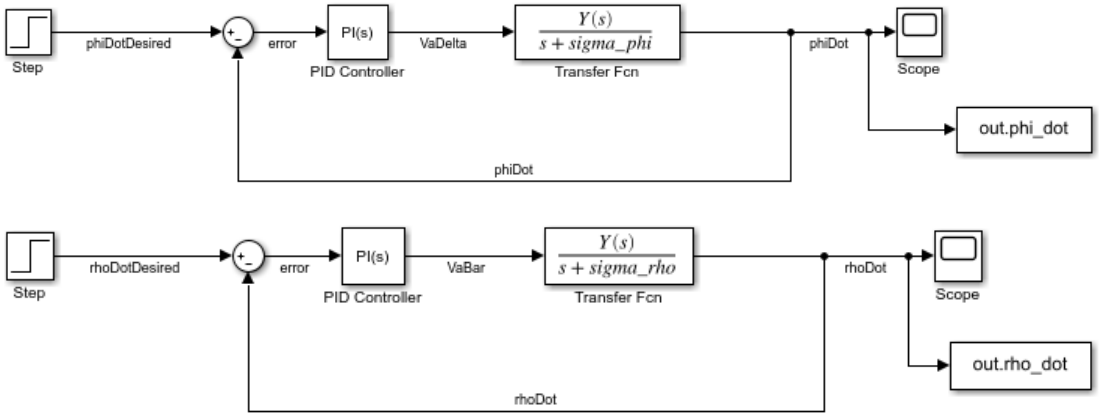
Controller Design

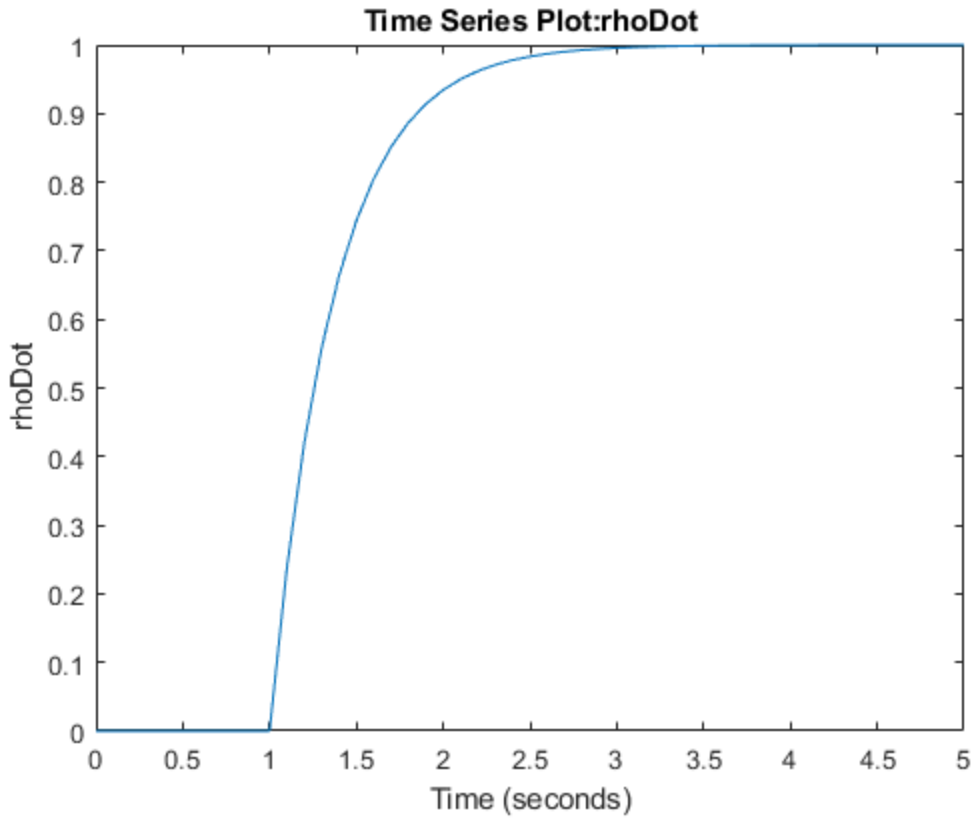
Once the transfer functions were estimated using the open loop step response, the closed loop systems were created in Simulink. The proportional and intergral gains were tuned using the built-in tuning function. The designs in Simulink have an overshoot of ~0% and a rise time of less than 1 second.

```
out = sim('innerLoopPhi');
plot(out.phi_dot)
hold on
hold off
```




```
open_system('innerLoopPhi')

out = sim('innerLoopRho');
plot(out.rho_dot)


open_system('innerLoopRho')
```





Published with MATLAB® R2020b