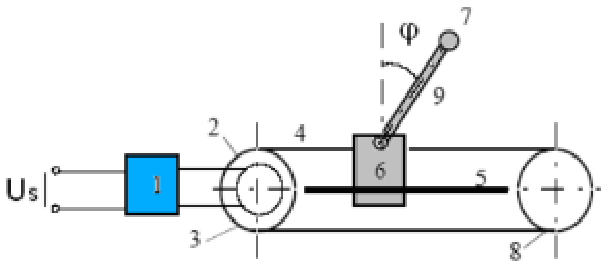
Table of Contents

1. List of Parameters of the System



1 – servo amplifier,
2 – motor,
3 – drive wheel,
4 – transmission belt,
5 – metal guiding bar,
6 – cart,
7 – pendulum weight,
8 – guide roll,
9 – pendulum rod.

Figure 1- Scheme of the inverted pendulum. [1].

List of Parameters	Values with proper units
weight of the cart (m_v)	4 kg
weight of the pendulum (m_K)	0.42 kg
total weight (m)	4.42 kg
velocity proportional friction of the cart (F_r)	6.5 N
friction of the cart (F_c)	15 N
length of the pendulum (l)	0.42 m
mass moment of inertia of pendulum (I)	0,08433 kgm^2
friction of the pendulum (C)	0.006 kg $m^2 s^{-1}$
rate constant (k_a)	7.5 N/V

Table 1. Parameters of the real system.

2. Transfer Function

The system can be described by following nonlinear differential equations:

$$m\ddot{r} + F_r\dot{r} + m_K l\,\ddot{\varphi}\,cos\varphi - m_K\,l(\dot{\varphi})^2\,sin\varphi = F \qquad \text{Eq.1}$$

$$I\ddot{\varphi} + C\dot{\varphi} + m_K lgsin\varphi + m_K l\ddot{r}cos\varphi = 0 \qquad \text{Eq.2}$$

In the drawing provided, the angle φ is defined with a reference pointing upwards, whereas the equation used to calculate φ, φ = π + θ, assumes that the reference is pointing downwards. This inconsistency in reference frames can lead to errors in calculating the angle φ. To correct this mistake, the reference for the angle ϕ should be changed to point downwards, so that it is consistent with the given equation. This will ensure that the correct value of angle φ can be obtained as φ = π + θ.

Two linear equations of the transfer function were found, where φ = π. Assume that φ = π + θ (θ represents a small angle from the vertical upward direction). Therefore, cosφ = −1, sinφ= −θ, $\frac{d^2\varphi}{dt^2} = 0$.

$$m\ddot{r} + F_r\dot{r} - m_K l\,\ddot{\theta} = F \qquad \text{Eq.3}$$

$$I\ddot{\theta} + C\dot{\theta} - m_K lg\theta - m_K l\ddot{r} = 0 \qquad \text{Eq.4}$$

Perform Laplace transform

$$(ms^2 + F_r s)R(s) - m_K l s^2 \theta(s) = F(s) \quad \text{Eq.5}$$

$$(Is^2 + Cs - m_K l g)\theta(s) = m_K l s^2 R(s) \quad \text{Eq.6}$$

Solve Eq6 for R(s)

$$R(s) = \left(\frac{m_k l s^2}{Is^2 + Cs + m_K \mathrm{lg}}\right)\theta(s) \quad \text{Eq.7}$$

Put R(s) in Eq6.

$$(ms^2 + F_r S)\left(\frac{m_k l s^2}{Is^2 + Cs + m_K \mathrm{lg}}\right) - m_K l s^2 \theta(s) = F(s) \quad \text{Eq.8}$$

Transfer Function

$$P_{Pendulum}(s) = \frac{\theta(s)}{Fs)} = \frac{l m_k s}{(-l^2 m_k{}^2 + Im)s^3 + (Cm + F_r I)s^2 + (mglm_k + CF_r)s + m_k g l} \quad \text{Eq.9}$$

3. State-Space Equations

$$m\ddot{r} + F_r\dot{r} - m_K l\,\ddot{\theta} = F \quad \text{Eq.10}$$

$$I\ddot{\theta} + C\dot{\theta} - m_K l g\theta - m_K l\ddot{r} = 0 \quad \text{Eq.11}$$

Find $\ddot{\theta}$ from Eq.11

$$\ddot{\theta} = \left(\frac{m_k l}{I}\right)\ddot{r} + \left(\frac{m_k l g}{I}\right)\theta + \left(\frac{-C}{I}\right)\dot{\theta} \quad \text{Eq.12}$$

Substitute $\ddot{\theta}$ in Eq.10

$$m\ddot{r} = F - F_r\dot{r} + m_K l\,\ddot{\theta} \quad \text{Eq.13}$$

$$\left(m - \frac{(m_K l)^2}{I}\right)\ddot{r} = F - F_r\dot{r} + \left(\frac{(m_k l)^2 g}{I}\right)\theta + \left(\frac{-m_k l C}{I}\right)\dot{\theta} \quad \text{Eq.14}$$

Consider that $\left(\frac{m_k l}{I}\right) = x$

$$(m - x^2 I)\ddot{r} = F - F_r\dot{r} + (x^2 I g)\theta + (-xC)\dot{\theta} \quad \text{Eq.15}$$

$$\ddot{r} = \left(\frac{1}{m - x^2 I}\right)F + \left(\frac{-F_r}{m - x^2 I}\right)\dot{r} + \left(\frac{x^2 I g}{m - x^2 I}\right)\theta + \left(\frac{-xC}{x^2 I g}\right)\dot{\theta} \quad \text{Eq.16}$$

Substitute $\ddot{r}$ in Eq.11

$$I\ddot{\theta} + C\dot{\theta} - m_K lg\theta = \left(\frac{m_K l}{m-x^2I}\right)F + \left(\frac{-F_r m_K l}{m-x^2I}\right)\dot{r} + \left(\frac{x^2Igm_K l}{m-x^2I}\right)\theta + \left(\frac{-xCm_K l}{x^2Ig}\right)\dot{\theta} \qquad \text{Eq.17}$$

Consider that $(m - x^2I) = y$

$$I\ddot{\theta} = \left(-\frac{IxC}{y} - C\right)\dot{\theta} + \left(\frac{I^2x^3}{y} + Ixg\right)\theta + \left(-\frac{F_rIx}{y}\right)\dot{r} + \left(\frac{Ix}{y}\right)F \qquad \text{Eq.18}$$

$$\ddot{\theta} = \left(-\frac{xC}{y} - \frac{C}{I}\right)\dot{\theta} + \left(\frac{Ix^3}{y} + xg\right)\theta + \left(-\frac{F_rIx}{y}\right)\dot{r} + \left(\frac{x}{y}\right)F \qquad \text{Eq.19}$$

$$\begin{bmatrix}\dot{r}\\ \ddot{r}\\ \dot{\theta}\\ \ddot{\theta}\end{bmatrix} = \begin{bmatrix}0 & 1 & 0 & 0\\ 0 & \frac{-F_r}{y} & \frac{x^2Ig}{y} & 0\\ 0 & 0 & 0 & 1\\ 0 & \frac{-F_rx}{y} & \frac{Ix^3}{y} + xg & \frac{-xC-C}{I}\end{bmatrix}\begin{bmatrix}r\\ \dot{r}\\ \theta\\ \dot{\theta}\end{bmatrix} + \begin{bmatrix}0\\ \frac{1}{y}\\ 0\\ \frac{x}{y}\end{bmatrix}F \qquad \text{Eq.20}$$

Substitute values from table

$$\begin{bmatrix}\dot{r}\\ \ddot{r}\\ \dot{\theta}\\ \ddot{\theta}\end{bmatrix} = \begin{bmatrix}0 & 1 & 0 & 0\\ 0 & -2.51 & 1.4 & 0\\ 0 & 0 & 0 & 1\\ 0 & -5.27 & 20.8 & -0.22\end{bmatrix}\begin{bmatrix}r\\ \dot{r}\\ \theta\\ \dot{\theta}\end{bmatrix} + \begin{bmatrix}0\\ 0.39\\ 0\\ 0.81\end{bmatrix}F \qquad \text{Eq.21}$$

From the state space we need to find the current value of both ϕ and x as outputs.

$$y = \begin{bmatrix}0 & 0 & 1 & 0\end{bmatrix}\begin{bmatrix}r\\ \dot{r}\\ \theta\\ \dot{\theta}\end{bmatrix} + [0]F \qquad \text{Eq.22}$$

However, we will simplify the analysis for only a SISO problem for $P_{Pendulum}(s) = \frac{\theta(s)}{F(s)}$

$$\frac{\theta(s)}{F(s)} = \frac{lm_ks}{(-l^2m_k{}^2+Im)s^3+(Cm+F_rI)s^2+(mglm_k+CF_r)s+m_kgl} \qquad \text{Eq.23}$$

$$[(-l^2m_k{}^2 + Im)s^3 + (Cm + F_rI)s^2 + (mglm_k + CF_r)s + m_kgl]\theta(s) = lm_ksF(s) \qquad \text{Eq.24}$$

Passing to time domain:

$$(-l^2m_k{}^2 + Im)\dddot{\theta} + (Cm + F_rI)\ddot{\theta} + (mglm_k + CF_r)\dot{\theta} + m_kgl\theta = lm_kF \qquad \text{Eq.25}$$

Solving for $\dddot{\theta}$ and changing to state variables

$$\dddot{\theta} = \frac{(-Cm-F_rI)}{(-l^2m_k{}^2+Im)}\ddot{\theta} + \frac{(-mglm_k-CF_r)}{(-l^2m_k{}^2+Im)}\dot{\theta} + \frac{(-m_kgl)}{(-l^2m_k{}^2+Im)}\theta + \frac{lm_k}{(-l^2m_k{}^2+Im)}F \qquad \text{Eq.26}$$

$$\dot{x}_3 = \frac{(-Cm-F_rI)}{(-l^2m_k{}^2+Im)}x_3 + \frac{(-mglm_k-CF_r)}{(-l^2m_k{}^2+Im)}x_2 + \frac{(-m_kgl)}{(-l^2m_k{}^2+Im)}x_1 + \frac{lm_k}{(-l^2m_k{}^2+Im)}F \qquad \text{Eq.27}$$

Therefore, the state-space representation is for one single input F and one single output θ is

$$\dot{X}=\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\end{bmatrix}=\begin{bmatrix}0 & 1 & 0\\ 0 & 0 & 1\\ \frac{(-Cm-F_rI)}{(-l^2m_k{}^2+Im)} & \frac{(-mglm_k-CF_r)}{(-l^2m_k{}^2+Im)} & \frac{(-m_kgl)}{(-l^2m_k{}^2+Im)}\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix}+\begin{bmatrix}0\\ 0\\ \frac{lm_k}{(-l^2m_k{}^2+Im)}\end{bmatrix}F \qquad \text{Eq.28}$$

$$Y=\begin{bmatrix}1 & 0 & 0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix}+[0]u \qquad \text{Eq.29}$$

Which for the given numerical values of your own problem from the table in the next section:

$$\dot{X}=\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\end{bmatrix}=\begin{bmatrix}0 & 1 & 0\\ 0 & 0 & 1\\ -2.3808 & -23.5311 & -5.0655\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix}+\begin{bmatrix}0\\ 0\\ 0.5164\end{bmatrix}F \qquad \text{Eq.30}$$

$$Y=\begin{bmatrix}1 & 0 & 0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix} \qquad \text{Eq.31}$$

Which, for MATLAB purposes, we can rewrite the state-variable matrices as

$$A=[0\ 1\ 0;0\ 0\ 1;\ -2.3808\quad -23.5311\quad -5.0655],\quad B=[0;0;\ 0.5164], C=[1\ 0\ 0], D=0 \qquad \text{Eq.32}$$

4. Calculations of Numerical Transfer Function

List of Parameters	Values with proper units
weight of the pendulum (m_k)	0.42 kg
total weight (m)	4.42 kg
velocity proportional friction of the cart (F_r)	6.5 N
length of the pendulum (l)	0.42 m
Mass moment of inertia of pendulum (I)	0,08433 kgm^2
friction of the pendulum (C)	0.006 kg m^2s^{-1}

Table 2. Parameters of the real system.

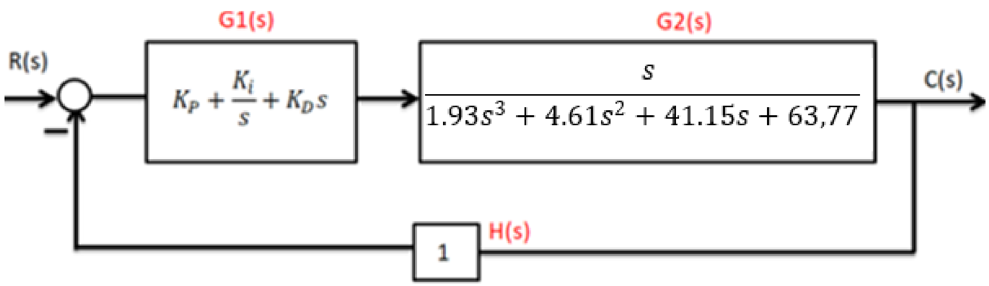
Transfer Function of pendulum angle with respect to the vertical axis

$$\frac{\theta(s)}{Fs)}=\frac{lm_ks}{(-l^2m_k{}^2+Im)s^3+(Cm+F_rI)s^2+(mglm_k+CF_r)s+m_kgl}=\frac{14.7\ s}{28.4s^3+67.8s^2+604.9s+937.35} \qquad \text{Eq.33}$$

After some arrangement to simplify:

$$\frac{\theta(s)}{Fs)}=\frac{s}{1.93s^3+4.61s^2+41.15s+63.77} \qquad \text{Eq.34}$$

5. Transfer Function of Plant with PID Controller



Transfer function for block diagram

$$T(s) = \frac{G1G2}{1+G1G2(H)} = \frac{\left(\frac{K_i}{s}+K_D s+K_P\right)\left(\frac{s}{1.93s^3+4.61s^2+41.15s+63.77}\right)}{1+\left(\frac{K_i}{s}+K_D s+K_P\right)\left(\frac{s}{1.93s^3+4.61s^2+41.15s+63.77}\right)} \qquad \text{Eq.35}$$

After some arrangements it becomes

$$T(s) = \frac{(K_D s^2 + K_P s + K_i)}{1.93s^3+(K_D+4.61)s^2+(K_p+41.15)s+(K_i+63.77)} \qquad \text{Eq.36}$$

6. Case 1: Calculations of PID Controller Parameters for Underdamped Performance

$$T_P = 0.2s \quad and \quad T_s = 0.45s \qquad \text{Eq.37}$$

$$T_s = \frac{4}{\zeta\omega_n} = 0.45 \quad so \quad Re = \zeta\omega_n = 8.88 \qquad \text{Eq.38}$$

$$T_P = \frac{\pi}{Im} = 0.2 \quad so \quad Im = 15.708 \qquad \text{Eq.39}$$

Two desired poles are

$$s_{1,2} = -\sigma_d \pm j\omega_d = -8.88 \pm j15.708 \qquad \text{Eq.40}$$

$$\omega_n = \sqrt{(8.88)^2 + (15.708)^2} = 18.05 \qquad \text{Eq.41}$$

$$\zeta\omega_n = 8.88 \quad and \quad \omega_n = 18.05 \quad so \quad \zeta = 0.492 \qquad \text{Eq.42}$$

Desired characteristic equation can be written by adding a third pole at $s = 5 * 8.88 = 44.4$

$(s + 44.4)[(s + 8.88)^2 + (15.708)^2] = s^3 + 62.16s^2 + 1114.14s + 14456.447=0$

Multiply $s^3 + 62.16s^2 + 1114.14s + 14456.447=0$ by 1.93 (see Eq. 33)

$$1.93s^3 + 119.97s^2 + 2150.29s + 27900.94 = 0 \qquad \text{Eq.43}$$

Equate the coefficients of the denominator of block diagram transfer function and desired characteristic equation

$(K_D + 4.61) = 119.97$ Eq.44

$(K_p + 41.15) = 2150.29$ Eq.45

$(K_i + 63.81) = 27900.94$ Eq.46

$K_D = 115.36 \quad K_p = 2109.14 \quad K_i = 27838.13$ Eq.47

Checking Routh-Hurwitz stability conditions

$(K_D + 4.61) > 0,\ (K_p + 41.15) > 0,\ (K_i + 63.81) > 0$. Eq.48

$(K_D + 4.61)(K_p + 41.15) > (K_i + 63.81)1.93$ Eq.49

$257970 > 53848$ Conditions are satisfied Eq.50

7. Case 2: Calculations of PID Controller Parameters for Critically Damped Performance

Two desired poles are

$s_{1,2} = -\frac{1}{T} = -\frac{1}{0.15} = -6.666$ Eq.51

We assigned additional negative real poles five times larger than the dominant pole's real part, so

$s = 5(6.666) = -33.333$ Eq.52

Desired characteristic equation is

$(s + 33.333)(s + 6.666)^2 = s^3 + 46.662s^2 + 488.791s + 1481.037=0$ Eq.53

Multiply $s^3 + 46.662s^2 + 488.791s + 1481.037 = 0$ by 1.93 (see Eq. 44)

$1.93s^3 + 90.058s^2 + 943.367s + 2858.4 = 0$ Eq.54

Equate the coefficients of the denominator of block diagram transfer function and desired characteristic equation

$(K_D + 4.61) = 90.058$ Eq.55

$(K_p + 41.15) = 943.367$ Eq.56

$(K_i + 63.81) = 2858.4$ Eq.57

$K_D = 85.448 \qquad K_p = 902.22 \qquad K_i = 2794.59$ Eq.58

Checking Routh-Hurwitz stability conditions

$(K_D + 4.61) > 0,\ (K_p + 41.15) > 0,\ (K_i + 63.81) > 0$ Eq.59

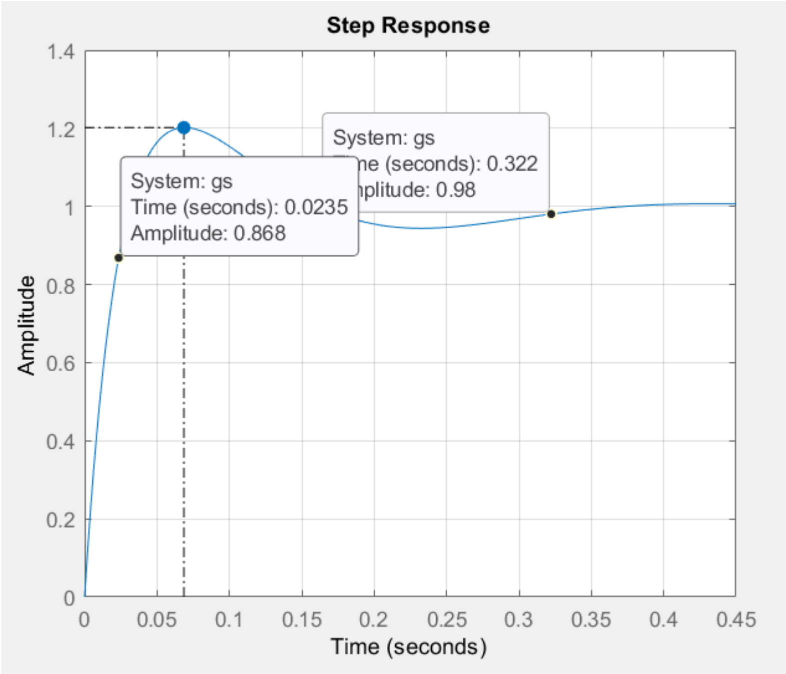
$(K_D + 4.61)(K_p + 41.15) > (K_i + 63.81)(1.93)$ Eq.60

$84958 > 5517$ Conditions are satisfied. Eq.61

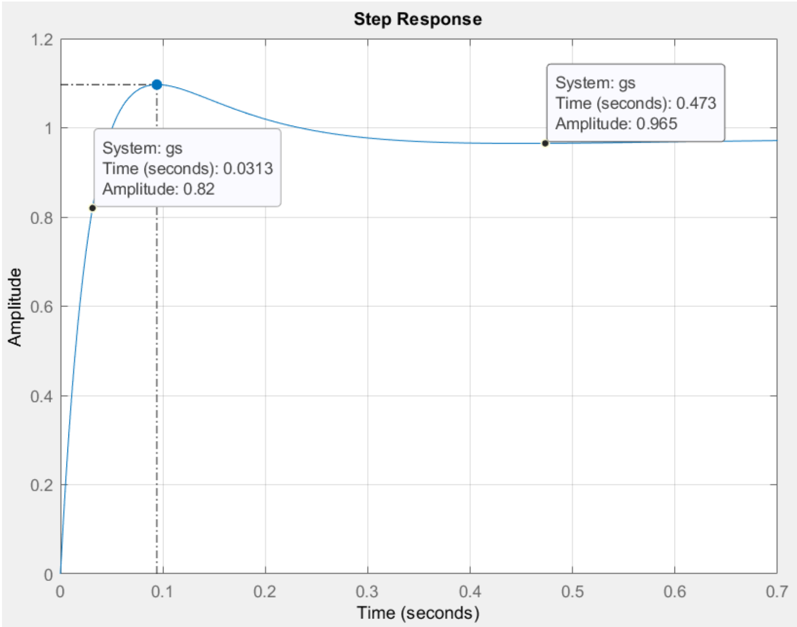
8. MATLAB Unit Step Response

```
%%
%Case1: Underdamped Performance
num= [1 0];
den= [1.93 4.61 41.15 63.77];
G1=tf(num,den);
kd=115.36; kp=2109.14; ki=27838.13;
G2=pid(kp,ki,kd);
gs = feedback(G1*G2,1);
step(gs);
%%
%Case2: Critically Damped Performance
num= [1 0];
den= [1.93 4.61 41.15 63.77];
G1=tf(num,den);
kd=85.448; kp=902.22; ki=2794.59  ;
G2=pid(kp,ki,kd);
gs = feedback(G1*G2,1);
step(gs)
```

Plot for Case1



Plot for Case2



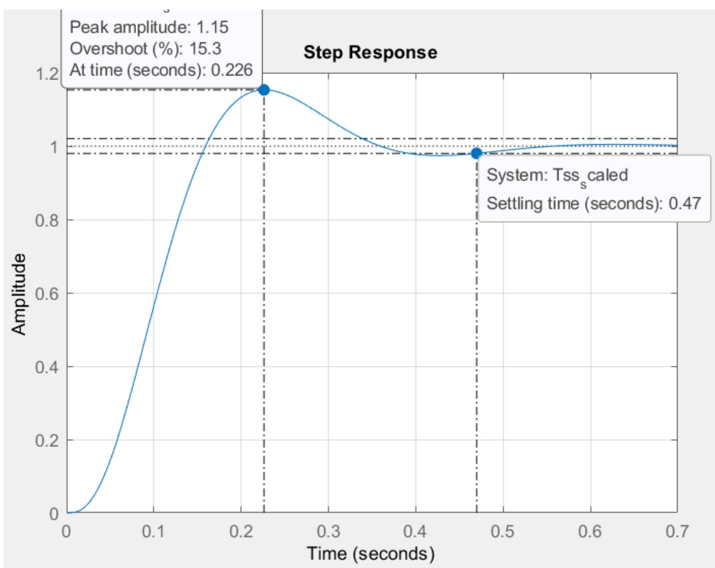
In case-2 the system behaving like overdamped instead of behaving like critically damped. Because it is not considering the zeros at T(s) in calculating the PID parameters.

9. State-Variable Feedback Design Method (Pole Placement)

Case 1: Underdamped performance

```
%Case 1: Underdamped performance
A=[0 1 0;0 0 1;-2.3808 -23.5311 -5.0655];
B=[0;0; 0.5164], C=[1 0 0], D=0;
Cm = ctrb(A,B);  % Calculate the controllability matrix.
Rank = rank(Cm)          % Check if the system is controllable.
z =0.492;                % Damping ratio as obtained in chapter 6.
wn =18.05;               % Natural frequency as obtained in chapter 6.
[num, den] = ord2(wn, z); % Produce a 2nd-order system that meets the response requirements.
r = roots(den);          % Assign the dominant poles to a vector "r".
poles = [r(1) r(2) -44.4]; % Place dominant & remaining poles.
K = acker(A, B, poles);  % Compute the state-variable controller gains.
Tss = ss(A-B*K, B, C, D); % The new system.
Kr = 1/dcgain(Tss);      % The scaling factor to eliminate steady state error.
Tss_scaled = ss(A-B*K, Kr.*B, C, D); % Scaled system to eliminate steady-state error.
poles = eig(A-B*K);      % Display the poles of the new system.
step(Tss_scaled)
```

Plot for Case1



The State-Variable Feedback controller design gives a better response than the PID controller for underdamped performance. The PID controller response shows NaN overshoot, no settling time and infinite steady-state, while the State-Variable Feedback controller response shows %15.3 overshoot, settling time of 0.47 seconds and a steady-state final value of 1. The peak amplitude of the State-Variable Feedback controller response is 1.15, which is slightly lower than the PID controller's peak amplitude of

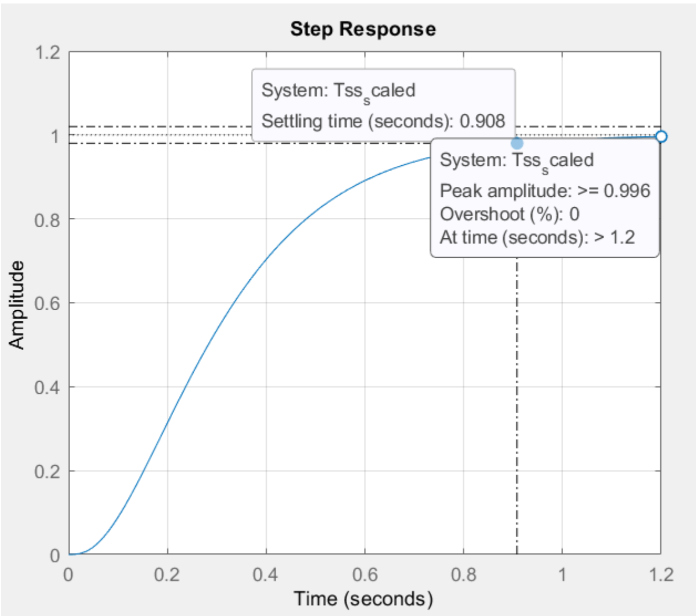
1.2. The peak time of the State-Variable Feedback controller response is 0.226 seconds, which is higher than the PID controller's peak time of 0.0685 seconds. Overall, the State-Variable Feedback controller design gives a more stable and predictable response than the PID controller for underdamped performance.

Case 2: Critically damped performance

```
A=[0 1 0;0 0 1;-2.3808 -23.5311 -5.0655];
B=[0;0; 0.5164], C=[1 0 0], D=0;
poles = [-6.666 -6.666 -33.33];    % Place the desired poles as obtained in chapter 7.
K = acker(A, B, poles);    % Compute the state-variable controller gains.
Tss = ss(A-B*K, B, C, D); % The new system.
Kr = 1/dcgain(Tss);        % The scaling factor to eliminate steady state error.
Tss_scaled = ss(A-B*K, Kr.*B, C, D); % Scaled system to eliminate steady-state error.
poles = eig(A-B*K);        % Display the poles of the new system.
step(Tss_scaled)
```

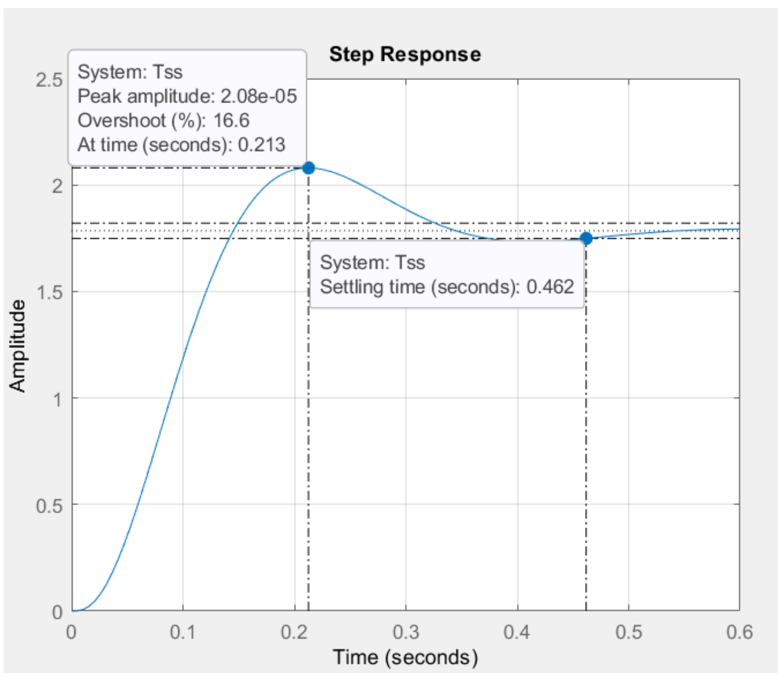
In the critically damped performance case, we designed a state-variable feedback controller using the desired poles and calculated the scaling factor to eliminate steady-state error. The step response of the system showed a settling time of 0.908, a steady state final value of 1, and no overshoot with a peak amplitude of 0.996 and peak time of 1.2. When compared to the PID controller for underdamped performance, the critically damped system showed a faster settling time and no overshoot, but a slightly later peak time. The PID controller showed a settling time of 2.95, an overshoot of NaN, and a peak time of 0.0972. Therefore, the state-variable feedback controller design performed better than the PID controller for this critically damped system.

Plot for Case2;



10. Full-Order State Observer Design

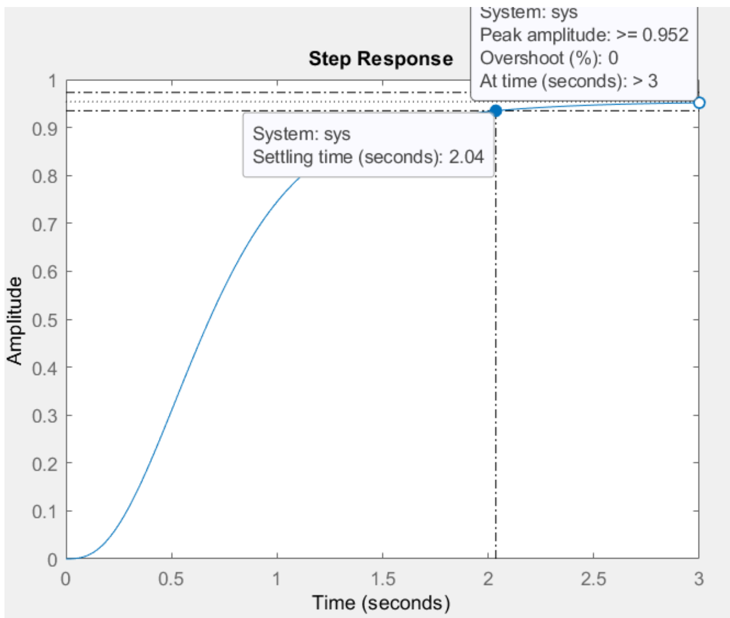
```
%Full-Order State Observer Design
A=[0 1 0;0 0 1;-2.3808 -23.5311 -5.0655];
B=[0;0; 0.5164], C=[1 0 0], D=0;
Om = obsv(A,C) % Calculate the observability matrix.
Rank = rank(Om)          % Check if the system is observable.
z =0.492;                % Damping ratio as obtained in chapter 6.
wn =18.05;               % Natural frequency as obtained in chapter 6.
[num, den] = ord2(wn, z); % Produce a 2nd-order system that meets the response requirements.
r = roots(den);          % Assign dominant poles to a vector "r".
poles = [r' 10*real(r(1))]; % Place the desired poles.
ob_poles = 10*poles;     % Increase 10 times the observer poles.
L = place(A',C', ob_poles).'     % Calculate observer gains for the original SS.
K = acker(A, B, poles);  % Compute & display controller gains.
Tss = ss(A-B*K, B, C, D); % Form the new SS system LTI object.
step(Tss)                % Produce compensated system step response
```



The response of the system obtained by Full-Order State Observer design is significantly better than that obtained with the PID controller for underdamped performance. The new system has a lower percent overshoot (16.6% vs NaN), shorter settling time (0.462 vs not shown), and a slightly longer peak time (0.213 vs 0.0685). Additionally, the new system has a finite steady state value, whereas the PID controller had a steady state infinite. Therefore, Full-Order State Observer design is a better approach for controlling underdamped systems than PID controller.

11. LQR Design

```
A=[0 1 0;0 0 1;-2.3808 -23.5311 -5.0655];
B=[0;0; 0.5164], C=[1 0 0], D=0;
Q = [100 0 0; 0 1 0; 0 0 1];
R = [0.01];
K = lqr(A, B, Q, R);
sys=ss(A-B*K, B*K(1), C, D);
step(sys)
```



The system obtained by LQR design shows zero overshoot, while the system obtained with PID controller shows NaN overshoot. The settling time of the system obtained by LQR design is 2.04, while the system obtained with PID controller does not show a settling time. The peak time of the system obtained by LQR design is 3, while the system obtained with PID controller has a peak time of 0.0685. Overall, the LQR controller design provides a more stable and better-performing system compared to the PID controller design for underdamped performance.

12. Conclusions

PID controllers have advantages over state-variable feedback controller, full-order state observer, and LQR controllers in terms of their simplicity, ease of implementation, and low computational requirements. PID controllers are also robust and effective for linear systems with stable dynamics.

State-variable feedback controllers have advantages over PID controllers, including the ability to handle more complex systems with non-linear dynamics, better disturbance rejection, and improved tracking performance.

Full-order state observer controllers have advantages over PID controllers, including the ability to estimate the system's state variables, even if they are not directly measurable. Full-order state observers can also enhance the system's robustness and disturbance rejection.

LQR controllers have advantages over PID controllers, state-variable feedback controllers, and full-order state observer controllers, including optimal control performance, robustness to modeling uncertainties, and the ability to handle both linear and non-linear systems.

In terms of practical implementation in MATLAB, PID controllers are the easiest to implement due to their simplicity and low computational requirements. The LQR controller resulted in better system performance in terms of steady-state error, disturbance rejection, and control effort. However, the choice of the best controller depends on the specific application and the desired performance criteria.

References

1. Marholt, Jiří & Gazdoš, František (2010). “Modelling, identification and simulation of the inverted pendulum PS600”, Acta Montanistica Slovaca, issue 1, pages 14-18.