Frequency Domain and Control System Introduction, with application to Launch Vehicles Part 1

March 7, 2013

Rob Hall

MSFC EV41

CRM Solutions, Inc.

Rev. 1

Summary

- **Charts provide an introduction to control system analysis/design approaches.**
- **Starts with fundamental concepts and advances to stability of a launch vehicle with unstable aerodynamics.**
- **Part 1**
  - Stability and Laplace Transforms
  - Eigenvalues
  - Spring-mass-damper system with Laplace Transforms and State Space
  - Closed-Loop vs. Open Loop
  - Root locus
  - Proportional-Integral-Derivative Control
  - Pole Placement
  - Step Response
- **Part 2**
  - Transfer function frequency response
  - Bode plots
  - Frequency response with Nyquist Plots
  - Bode vs. Nichols vs. Nyquist
  - Nyquist plots: full vs. half
  - Nyquist stability criterion
  - Stability margins
  - Stability results with an unstable plant: Launch vehicle example
  - Bending filters and their design

Simple First Order Differential Equation

- **Consider a simple differential equation representing a dynamic system:**

$$\dot{x}(t) = ax(t)$$

To solve:

$$\frac{dx}{dt} = ax$$

$$\frac{dx}{x} = a\,dt \quad \Rightarrow \quad \text{separation of variables}$$

$$\ln|x| = at + c \quad \Rightarrow \quad \text{integration}$$

$$e^{\ln|x|} = e^{(at+c)}$$

$$x = x_0 e^{at}$$

- **Example: Radiocarbon dating, where one compares the ratio of radioactive carbon to ordinary carbon to estimate age:**

$$\dot{x}(t) = kx(t)$$

x = current carbon ratio

x_0 = initial carbon ratio

t = time in years (unknown)

k = radioactive constant, ex: k = -0.00012

$$x = x_0 e^{kt} \quad \Rightarrow \quad e^{kt} = \frac{x}{x_0} \quad \text{solve for } t$$

k is negative because the radiation levels decay in time

For x to decay to zero (i.e. stable):
k<0 => xdot opposite sign of x

Example taken from Kreyszig, ref [6], section 1.3

Laplace Transforms and Stability

- **Among uses for a Laplace Transform is to determine a solution to a linear differential equation, and examine its stability properties:**

Example :

$$\dot{x}(t) = ax(t) \quad \Rightarrow \quad \dot{x}(t) - ax(t) = 0$$

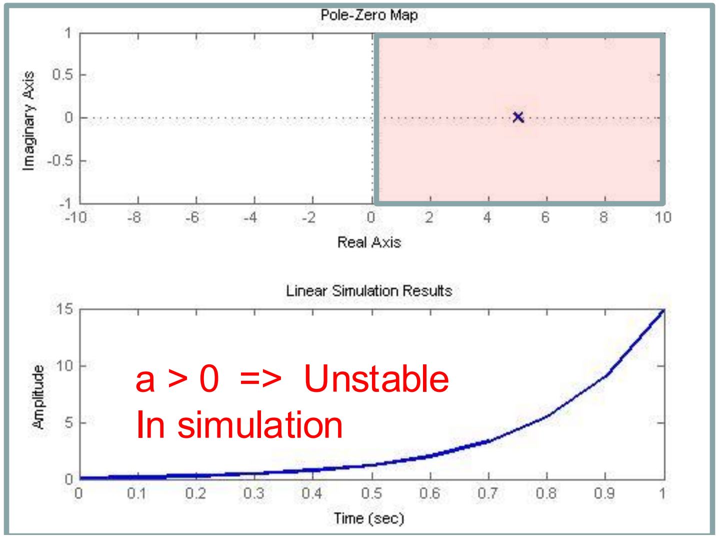
Take Laplace Transform :

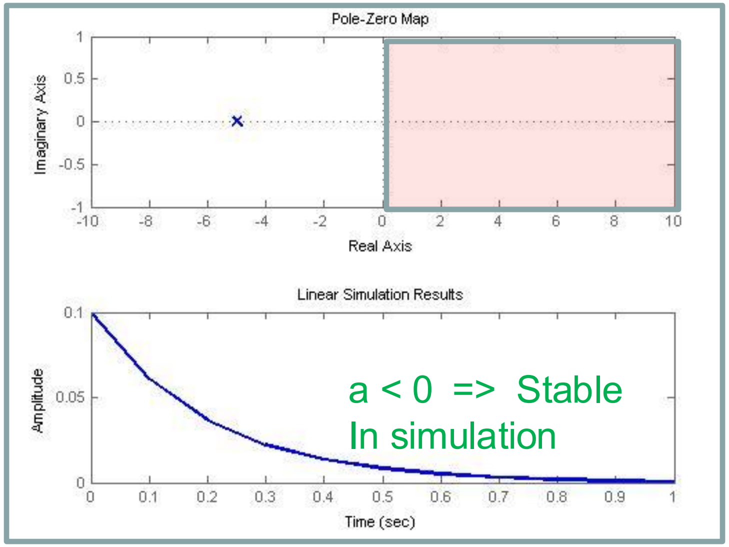
$$sX(s) - x(0) - aX(s) = 0$$

$$X(s) = \frac{x(0)}{s-a}$$

Take Inverse Laplace Transform :

$$x(t) = x(0)e^{at}$$





- **The denominator of the transfer function set = 0 is the "characteristic equation", and for stability the characteristic eqn must not have positive real roots:**

$$s - a = 0$$

  - a > 0 => Unstable (roots in right half plane)
  - a < 0 => Stable (roots in left half plane)
  - Note: These roots of the characteristic equation are also called "poles".

Recall: Principal Moments of Inertia, i.e. Eigenvalues of a Matrix

- **Recall that a vehicle mass property matrix (Inertia Tensor), I, can be transformed into a diagonal principal inertia matrix. Example:**

$$T = I\alpha$$

$$I_{body} = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{xy} & I_{yy} & I_{yz} \\ I_{xz} & I_{yz} & I_{zz} \end{bmatrix} \Rightarrow I_{principal} = \begin{bmatrix} I_1 & & \\ & I_2 & \\ & & I_3 \end{bmatrix}$$

- **Using the matlab command 'eig', one can compute the principal inertia matrix D, along with the transformation matrix to compute it:**

```
I =

     4     2    -1
     2     6     1
    -1     1     5

>> [V,D] = eig(I)

D =

    2.1049         0         0
         0    5.6027         0
         0         0    7.2924

E =

    0.7526   -0.4579    0.4732
   -0.4973    0.0759    0.8643
    0.4317    0.8857    0.1706
```

The principal moments of inertia are the 'eigenvalues' of the original 3 x 3 matrix

Matlab also computes the 'eigenvectors', which forms the transformation matrix from principal to body coordinates

- **The resulting transformation is I=E*D*E'**

```
>> I=E*D*E'

I =

    4.0000    2.0000   -1.0000
    2.0000    6.0000    1.0000
   -1.0000    1.0000    5.0000
```

The general eigenvalue problem:

$$AE = E\Delta \quad \text{or} \quad A = E\Delta E^{-1}$$

$$\Delta = \begin{bmatrix} \lambda_1 & & & \\ & \lambda_1 & & \\ & & \ldots & \\ & & & \lambda_n \end{bmatrix}$$

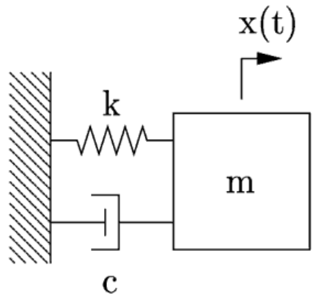
$\lambda =$ eigenvalues

$E =$ eigenvectors

Solving a More Complex Differential Equation

♦ Spring-mass-damped system

$$m\ddot{x}(t)+c\dot{x}(t)+kx(t)=0 \quad \text{or} \quad \ddot{x}=-\frac{k}{m}x-\frac{c}{m}\dot{x}$$



Put into State Space Format:

$x_1 = x$ position

$x_2 = \dot{x}$ rate

Substitute in:

$$\dot{x}_1 = \dot{x} = x_2$$

$$\dot{x}_2 = \ddot{x} = -\frac{k}{m}x_1 - \frac{c}{m}x_2$$

Put in form $\dot{x} = Ax$, where x is now a vector $x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{c}{m} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \Leftrightarrow \dot{x}(t) = Ax(t)$$

Same approach to solve:

$$\dot{x}(t) = Ax(t)$$

$$x = x_0 e^{At}$$

$$A = E\Lambda E^{-1}$$

$$x = x_0 e^{(E\Lambda E^{-1})t}$$

$$x = x_0 E e^{\Lambda t} E^{-1}$$

$$x = x_0 E e^{\begin{bmatrix} \lambda_1 t & \\ & \lambda_2 t \end{bmatrix}} E^{-1}$$

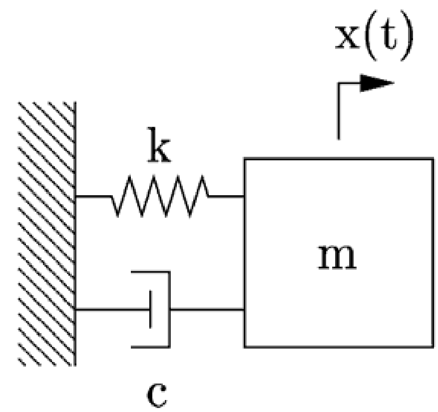
For stability: All eigenvalues (λ) must not have real values > 0. This is "necessary".

$$\lambda_{1,2} = \frac{1}{2}\left[-\frac{c}{m} \pm \sqrt{\left(\frac{c}{m}\right)^2 - 4\frac{k}{m}}\right]$$

Transfer Function for same system:

- **Unforced spring-mass with small damping:**

$$m\ddot{x}(t)+c\dot{x}(t)+kx(t)=0$$



Take Laplace Transform :

$$m\left[s^2X(s)-sx(0)-\dot{x}(0)\right]+c\left[sX(s)-x(0)\right]+kX(s)=0$$

$$X(s)=\frac{mx(0)s+m\dot{x}(0)+cx(0)}{s^2+\frac{c}{m}s+\frac{k}{m}}$$

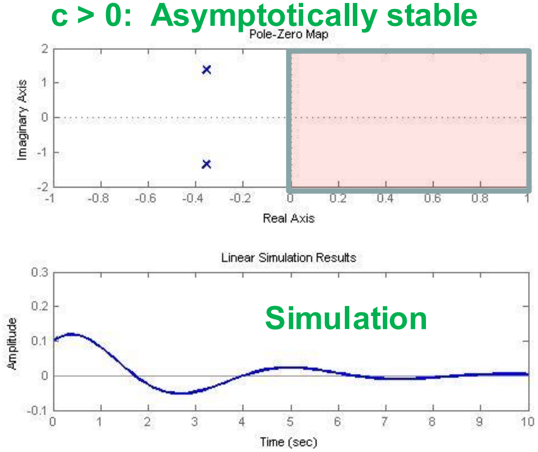
Characteristic Equation :

Real | Imaginary

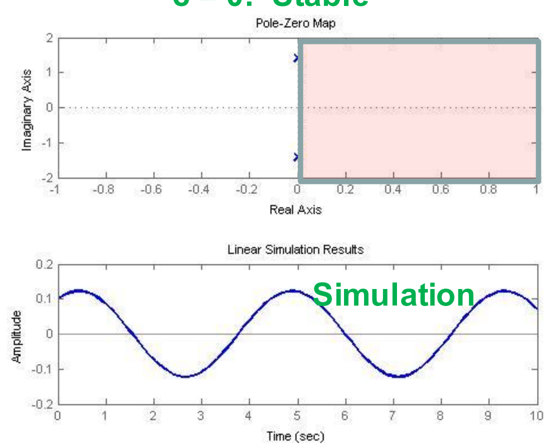
$$s^2+\frac{c}{m}s+\frac{k}{m}=0 \quad \Rightarrow \quad \text{Roots} = \quad \lambda_{1,2}=s_{1,2}=\frac{1}{2}\left[-\frac{c}{m}\pm\sqrt{\left(\frac{c}{m}\right)^2-4\frac{k}{m}}\right]$$

For stability, real part of roots (eigenvalues) cannot be > 0

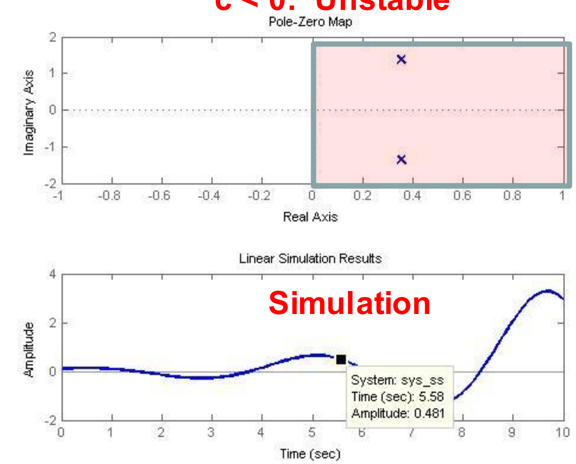
c > 0: Asymptotically stable



c = 0: Stable



c < 0: Unstable



Summary: Stability of Linear Systems

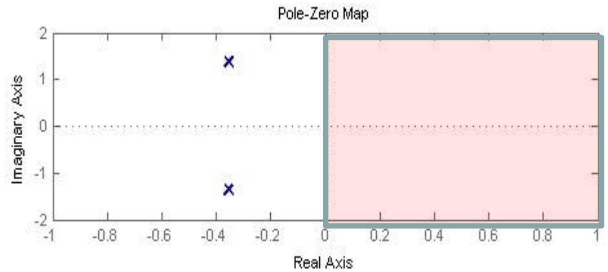
- **Given a linear homogenous (zero external force) system**

$$m\ddot{x}(t)+c\dot{x}(t)+kx(t)=0$$

- **We can define the system via either a transfer function or state space formulation:**

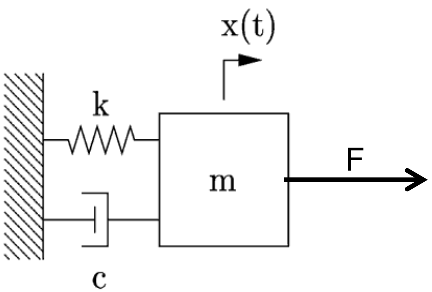
$$X(s)=\frac{mx(0)s+m\dot{x}(0)+cx(0)}{s^2+\frac{c}{m}s+\frac{k}{m}} \quad \Leftrightarrow \quad \begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix}=\begin{bmatrix}0 & 1\\ -\frac{k}{m} & -\frac{c}{m}\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix}$$

- **We can determine stability by looking at either:**
  - The eigenvalues of the state space *A* matrix => must not have positive real values.
  - The roots (poles) of the characteristic equation => must not have positive real values.
    - I.e. For asymptotic stability, it is "**necessary**" to have poles in the left hand plane.



State Space Formulation for Spring mass with forced motion (nonhomogeneous)

$$m\ddot{x}(t)+c\dot{x}(t)+kx(t)=F$$



Put into State Space Format :

$x_1 = x$ position

$x_2 = \dot{x}$ rate

Substitute in :

$$\dot{x}_1 = \dot{x} = x_2$$

$$\dot{x}_2 = \ddot{x} = -\frac{k}{m}x_1 - \frac{c}{m}x_2 + F$$

Put in state space form, where $\bar{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{c}{m} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} F$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + [0]F \quad \Rightarrow y = \text{ouput (position)}$$

$$\dot{\bar{x}} = A\bar{x} + Bu$$

$$y = C\bar{x} + Du$$

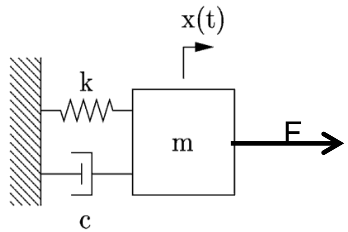
State Space Formulation

State Space Formulation for Spring mass with forced motion - Example

$x_1 = x$ position

$x_2 = \dot{x}$ rate

$$\dot{\bar{x}} = A\bar{x} + Bu$$
$$y = C\bar{x} + Du$$



$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{c}{m} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} F$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + [0]F \quad \Rightarrow y = \text{ouput (position)}$$

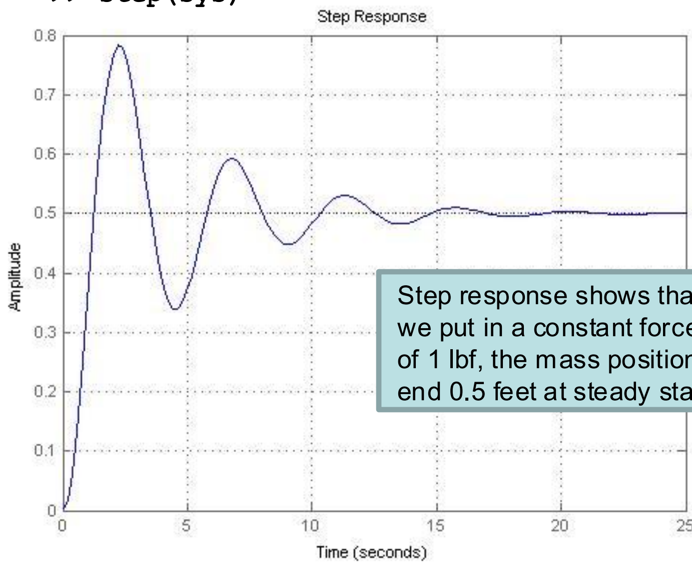
- **Example: m=1, k=2, c=0.5**

```
>> A = [0 1; -k/m -c/m]
A =
         0    1.0000
   -2.0000   -0.5000


>> eig(A)


ans =
  -0.2500 + 1.3919i
  -0.2500 - 1.3919i
```

Asymptotically Stable since eigenvalues (λ) < 0

```
>> B= [0 1]';
>> C=[1 0];
>> sys=ss(A,B,C,[0]);
```

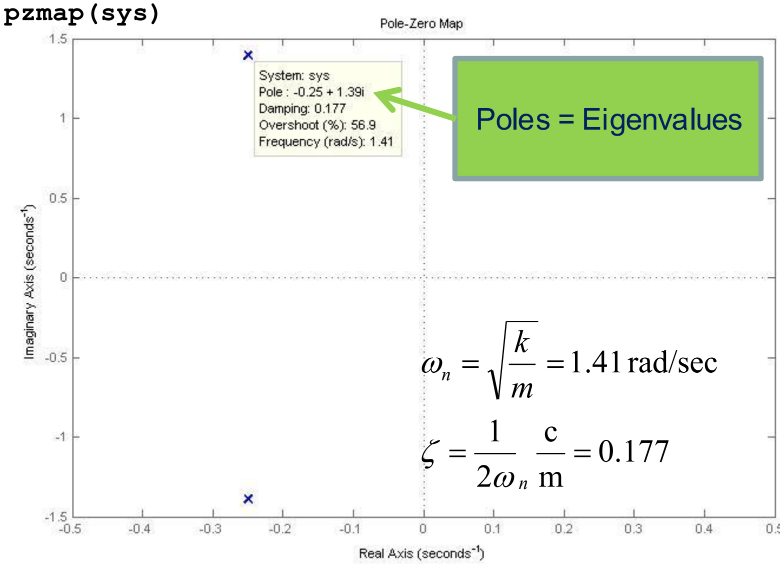
```
>> step(sys)
```



$$x_{final} = \frac{F}{k}$$

Step response shows that if we put in a constant force (F) of 1 lbf, the mass position will end 0.5 feet at steady state.

```
>> pzmap(sys)
```



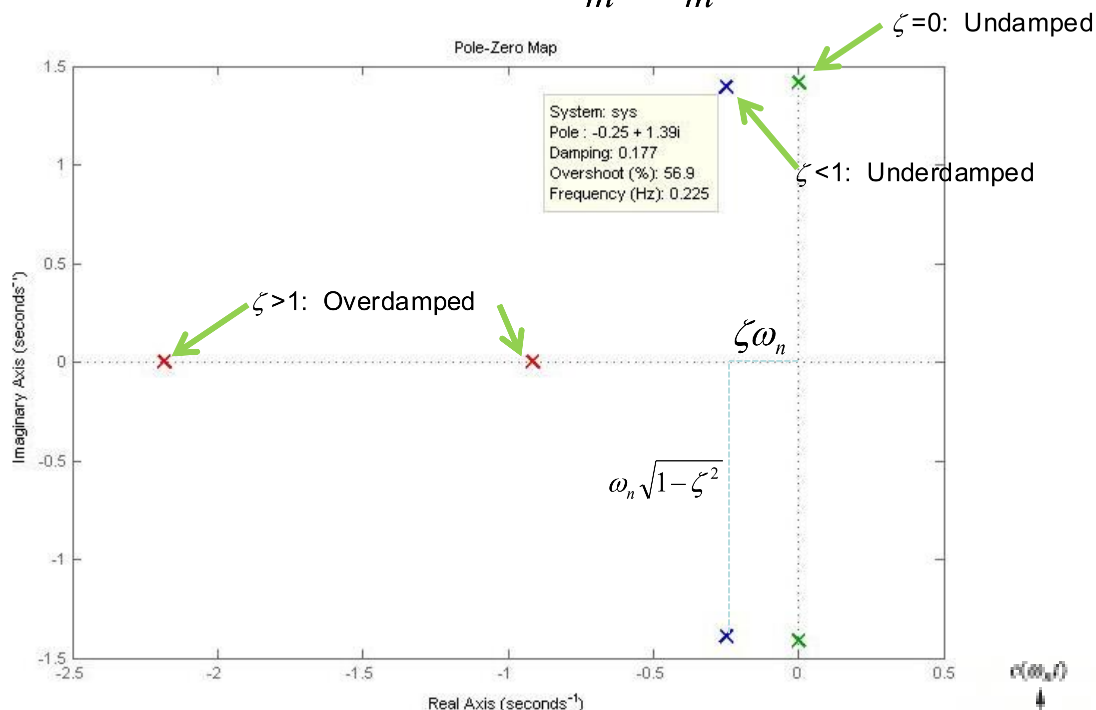
Poles = Eigenvalues

$$\omega_n = \sqrt{\frac{k}{m}} = 1.41 \text{ rad/sec}$$

$$\zeta = \frac{1}{2\omega_n}\frac{\text{c}}{\text{m}} = 0.177$$

Step Response vs. Pole Locations for a 2nd Order System

$$m\ddot{x}(t)+c\dot{x}(t)+kx(t)=0 \quad \text{or} \quad \ddot{x}=-\frac{k}{m}x-\frac{c}{m}\dot{x}$$

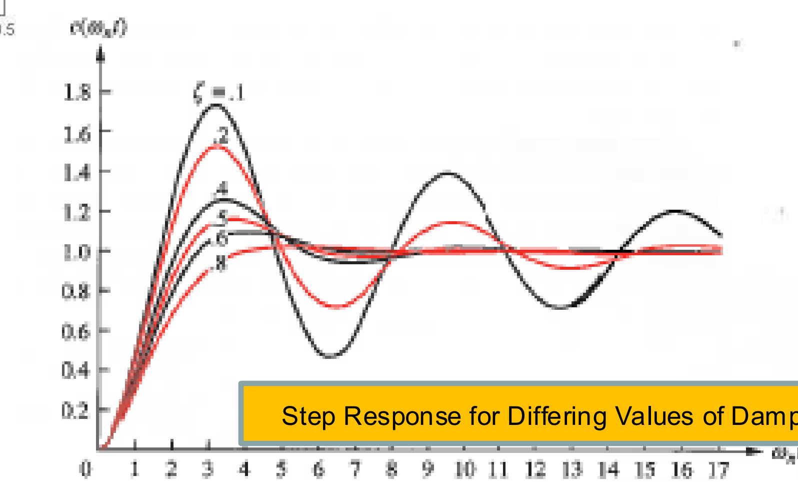


$$s_{1,2}=\frac{1}{2}\left[-\frac{c}{m}\pm\sqrt{\left(\frac{c}{m}\right)^2-4\frac{k}{m}}\right]$$

$$\omega_n=\sqrt{\frac{k}{m}}=1.41\ \text{rad/sec}$$

$$\zeta=\frac{1}{2\omega_n}\frac{\text{c}}{\text{m}}=0.177$$

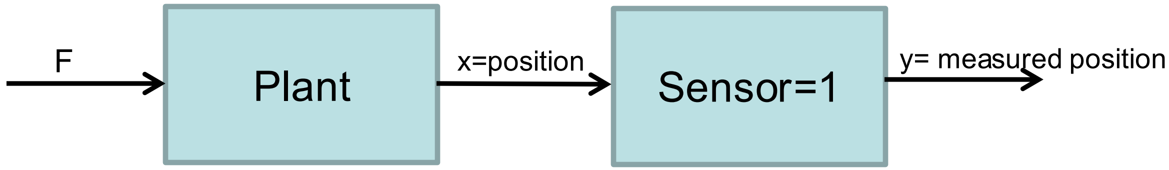
$$s_{1,2}=-\zeta\omega_n\pm\omega_n\sqrt{\zeta^2-1}$$



Step Response for Differing Values of Damping [6]

Going from State Space to Transfer function

- **Given our spring mass system with input Force F, the transfer function is (zero initial conditions):**



$$\frac{y(s)}{F(s)} = \frac{1}{s^2 + \frac{c}{m}s + \frac{k}{m}}$$

$$x(s) = \frac{1}{s^2 + \frac{c}{m}s + \frac{k}{m}} F(s) \qquad y(s) = x(s)$$

- **The relationship between state space and the transfer function is:**

$$\frac{y(s)}{F(s)} = C(sI - A)^{-1}B + D$$

$$\frac{y(s)}{F(s)} = \begin{bmatrix} 1 & 0 \end{bmatrix} \left(s \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{c}{m} \end{bmatrix} \right)^{-1} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} s & -1 \\ +\frac{k}{m} & s + \frac{c}{m} \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$= \frac{\begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} s + \frac{c}{m} & 1 \\ -\frac{k}{m} & s \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix}}{\det \begin{bmatrix} s & -1 \\ +\frac{k}{m} & s + \frac{c}{m} \end{bmatrix}} = \frac{1}{\mathrm{s}^2 + \frac{c}{m} + \frac{k}{m}}$$

Summary: Stability of Linear Systems

- **Given a linear system**

$$m\ddot{x}(t)+c\dot{x}(t)+kx(t)=F$$

- **We can derive a transfer function or state space formulation:**

$$\frac{y(s)}{F(s)}=\frac{1}{s^2+\frac{c}{m}s+\frac{k}{m}}$$

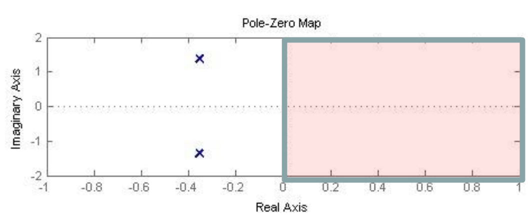
“Classical”

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix}=\begin{bmatrix}0 & 1\\ -\frac{k}{m} & -\frac{c}{m}\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix}+\begin{bmatrix}0\\ 1\end{bmatrix}F$$

$$y=\begin{bmatrix}1 & 0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix}+\begin{bmatrix}0\end{bmatrix}F \qquad \Rightarrow y=\text{ouput}$$

“Modern”

- **We can determine stability by looking at the eigenvalues (aka poles, aka roots of characteristic equation) => must not have positive real values.**
  - For asymptotic stability, it is a “necessary condition” to have poles in the left hand plane; i.e. a “necessary condition” to have eigenvalues with negative real parts…



Side Note: Linear Time Invariant Systems

- **Note with the system we have been working with, the matrix A is constant, specifically: It does not vary with time (t) or the state (x). This means it is Linear Time Invariant (LTI).**

$$\dot{x}(t) = Ax(t)$$

- **If the matrix A is linear in that it does not vary with the state (x), but does vary with time (t), then it is Linear Time Variant (LTV):**

$$\dot{x}(t) = A(t)x(t)$$

- **Another option is the system can be nonlinear and time variant:**

$$\dot{x}(x,t) = A(x,t)x(t)$$

- **For doing stability analysis in a ascent vehicle (say using ASAT or FRACTAL), we model the dynamics as linear and "freeze" the time so the resulting system is LTI.**
  - Allows linear formulations/stability approaches.

Linear vs Nonlinear Example: Slosh Damping

LTI Model:

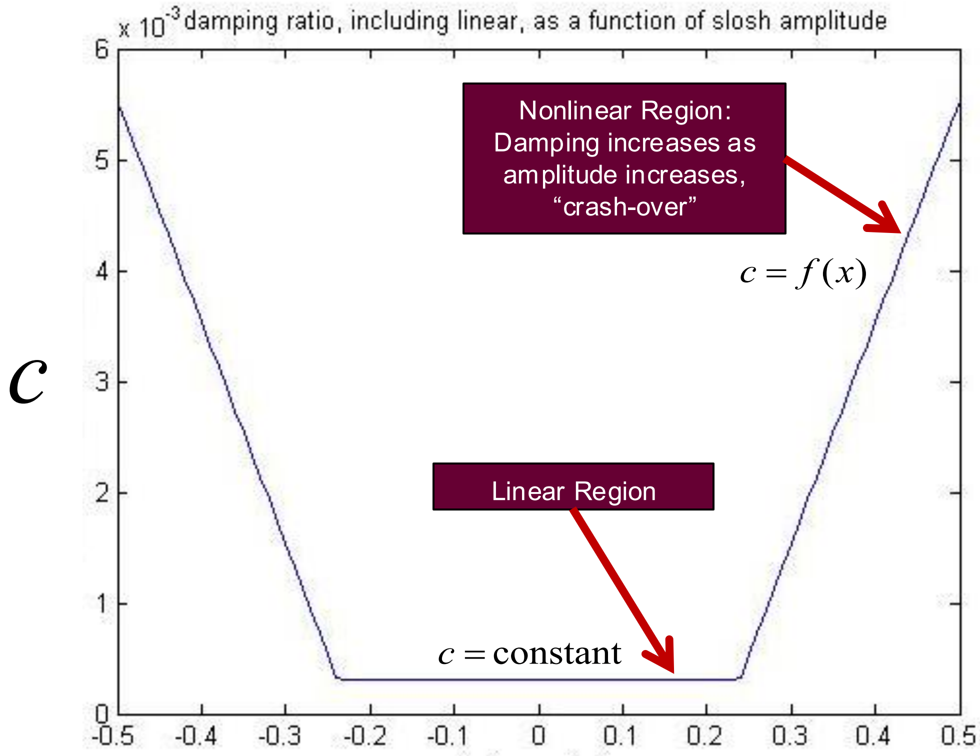
$$m\ddot{x}(t) + c\dot{x}(t) + kx(t) = F$$

x slosh mass position

$\dot{x}$ slosh mass rate

m, c, k are constant

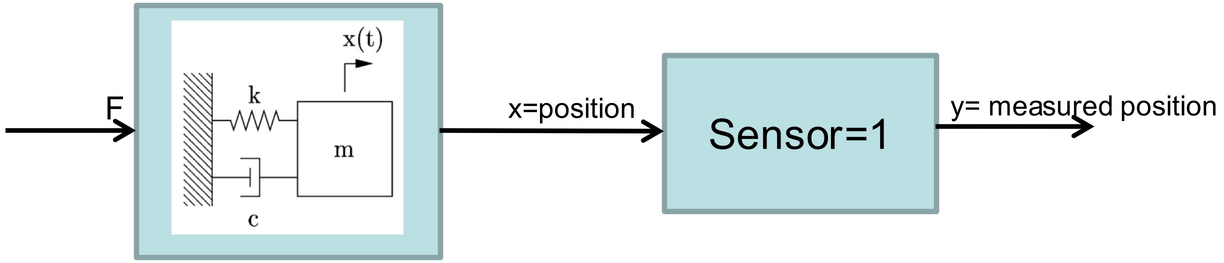
When we analyze slosh stability, our LTI analysis is only valid when damping (c) is not a function of slosh position, i.e. where slosh amplitude is small.



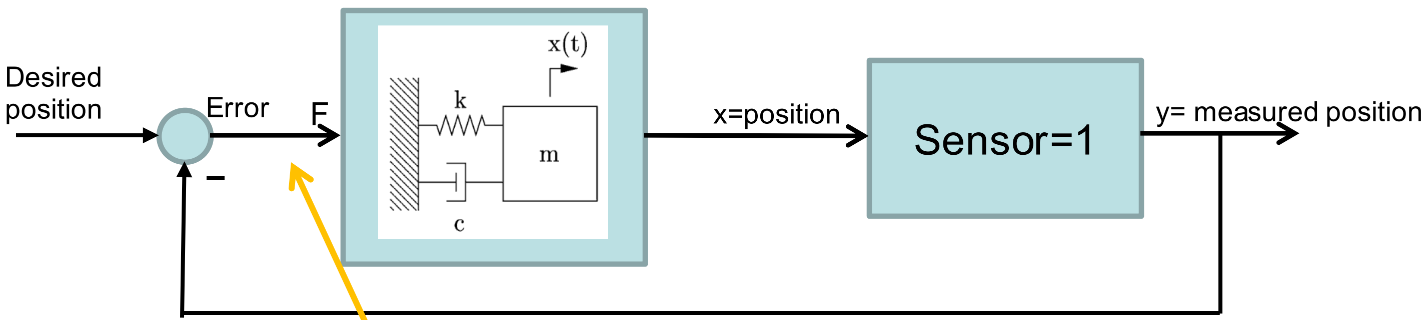
x = Slosh Amplitude (scaled to tank radius)

Open Loop vs Closed Loop

- Our example is "open loop" because the input force has no information on how the plant responds



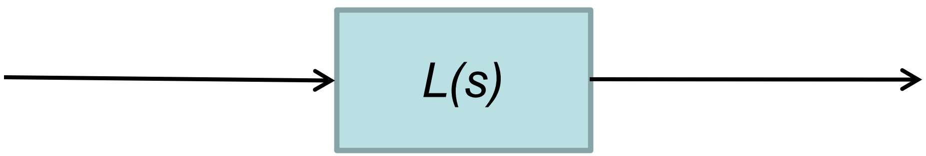
- We can "close the loop" by feeding back the actual position and use that to calculate the force:



This is "proportional" control, because the force F is now proportional to the input Error: (Force=K_p*Error, K_p=1)

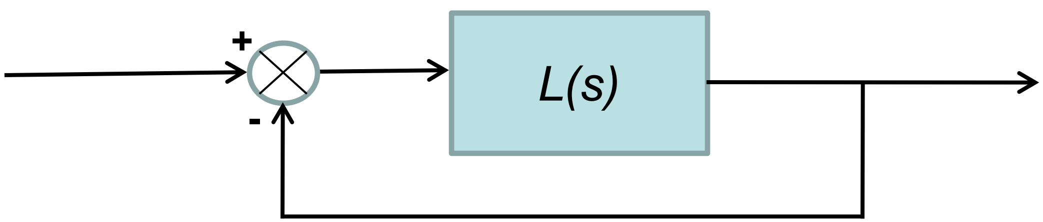
Open Loop vs. Closed Loop Transfer Functions

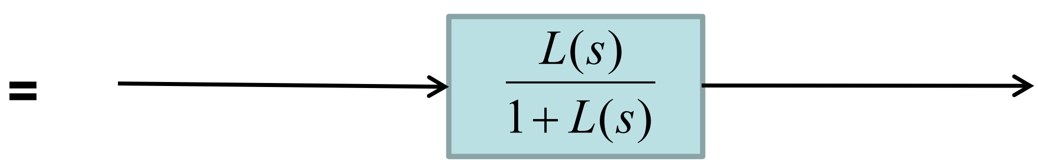
- **Open loop**



Open Loop System may be Stable or Unstable

- **Closed Loop**

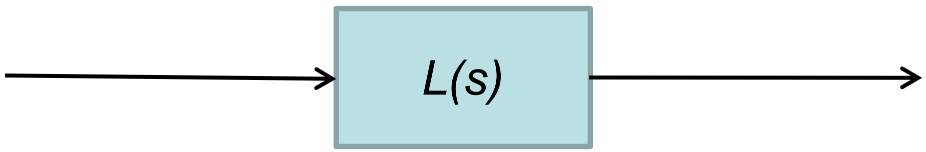




Closed Loop Characteristic Equation: *1+L(s)=0*
For stability, roots of characteristic equation should not be in the right half plane

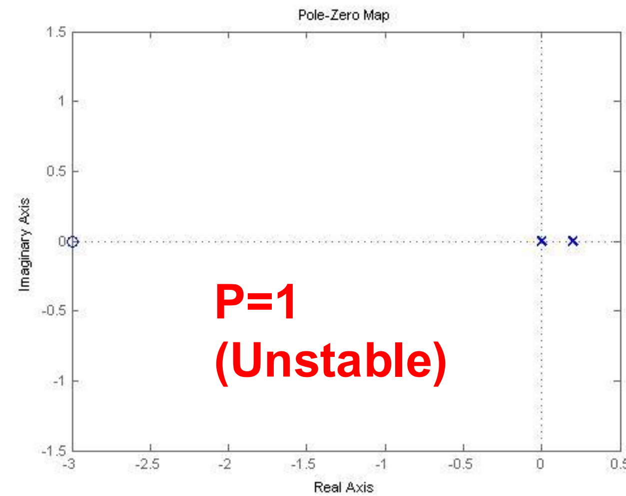
Open Loop vs. Closed Loop Example

Open Loop, can be unstable



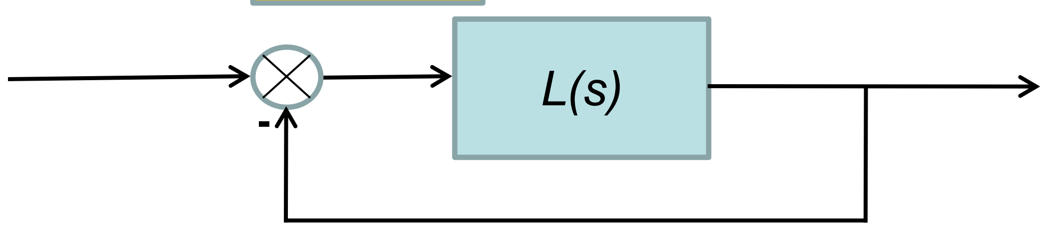
Open Loop Transfer Function :

$$L(s)=\frac{2(s+3)}{s(5s-1)}$$



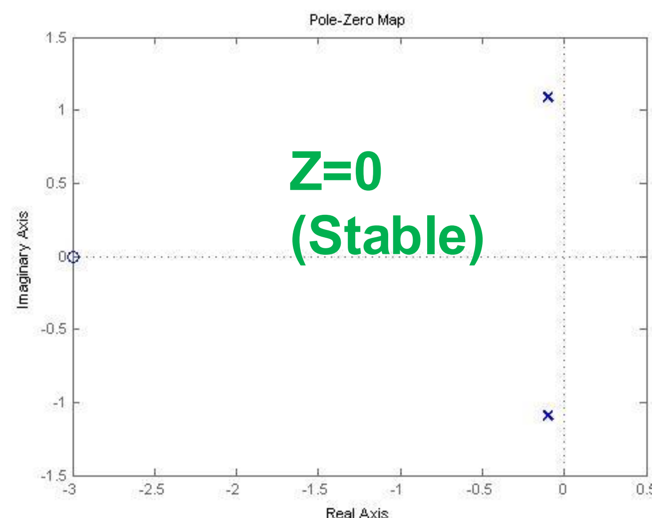
P=Number of poles in open loop transfer function in right hand plane

Closed Loop



Closed Loop Transfer Function :

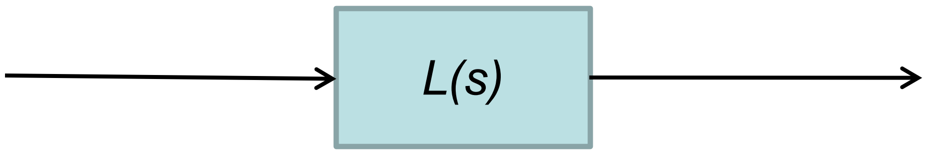
$$\frac{L(s)}{1+L(s)}=\frac{\frac{2(s+3)}{s(5s-1)}}{1+\frac{2(s+3)}{s(5s-1)}}=\frac{2(s+3)}{s(5s-1)+2(s+3)}=\frac{2(s+3)}{5s^2+s+6}$$



Z=Number of poles in closed loop transfer function in right hand plane

Open Loop vs. Closed Loop Example

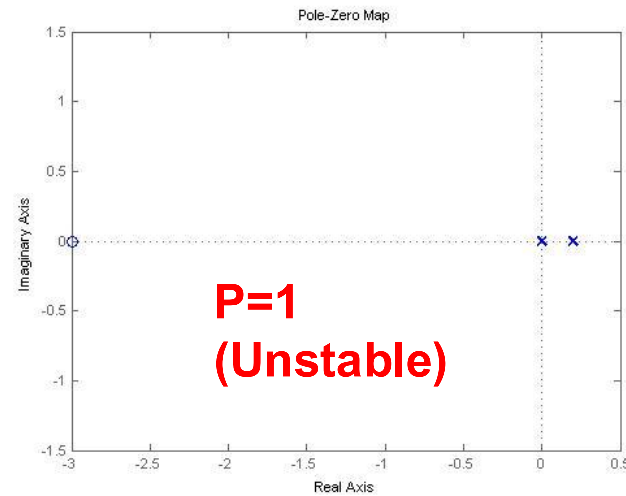
Open Loop



Open Loop Transfer Function :

$$L(s) = \frac{2(s+3)}{s(5s-1)}$$

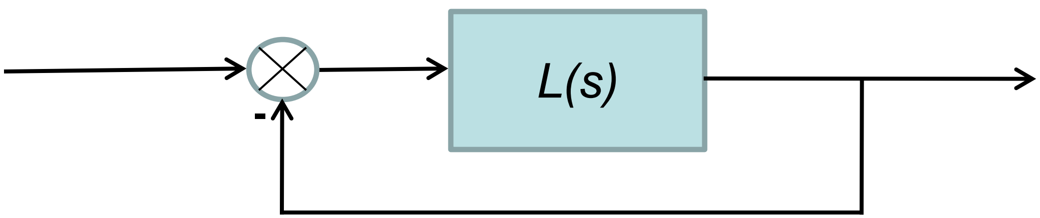
Ls=tf([2 6], [5 -1 0])



P=Number of poles in open loop transfer function in right hand plane

---

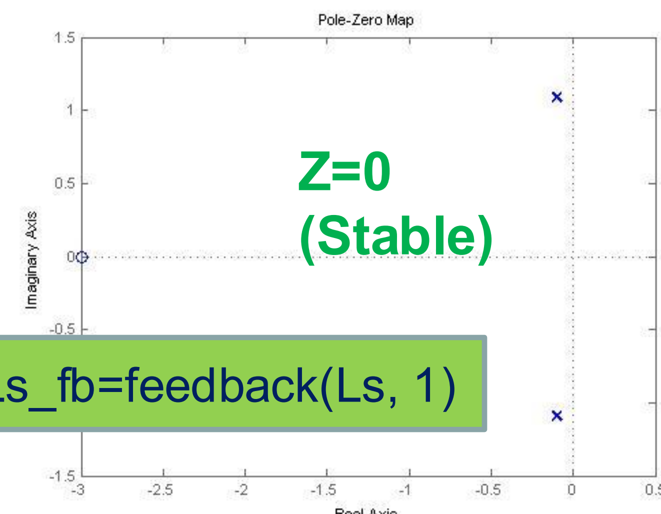
Closed Loop



Closed Loop Transfer Function :

$$\frac{L(s)}{1+L(s)} = \frac{\frac{2(s+3)}{s(5s-1)}}{1+\frac{2(s+3)}{s(5s-1)}} = \frac{2(s+3)}{s(5s-1)+2(s+3)} = \frac{2(s+3)}{5s^2+s+6}$$

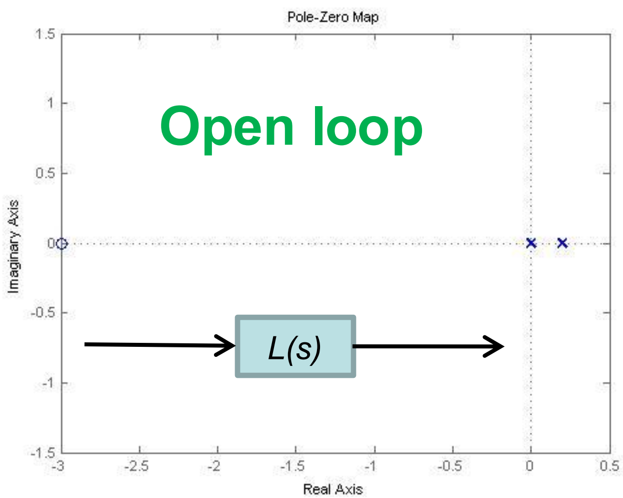
Ls_fb=feedback(Ls, 1)



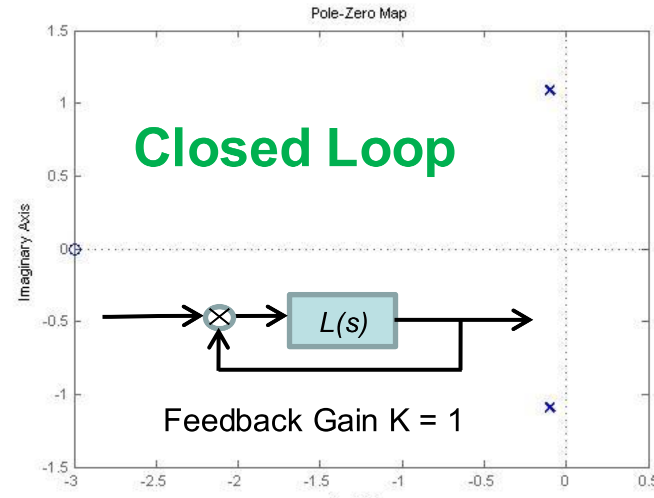
Z=Number of poles in closed loop transfer function in right hand plane

Same Example, But use Root Locus

- **Root locus is a tool that allows one to understand how the poles move when the loop is closed and the feedback gain varied**

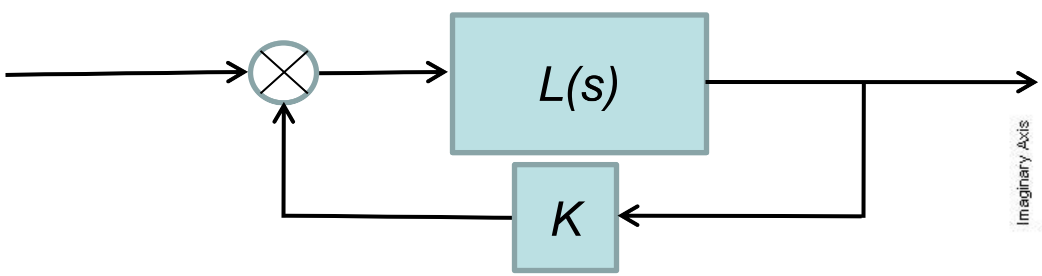


P=1
(Unstable)

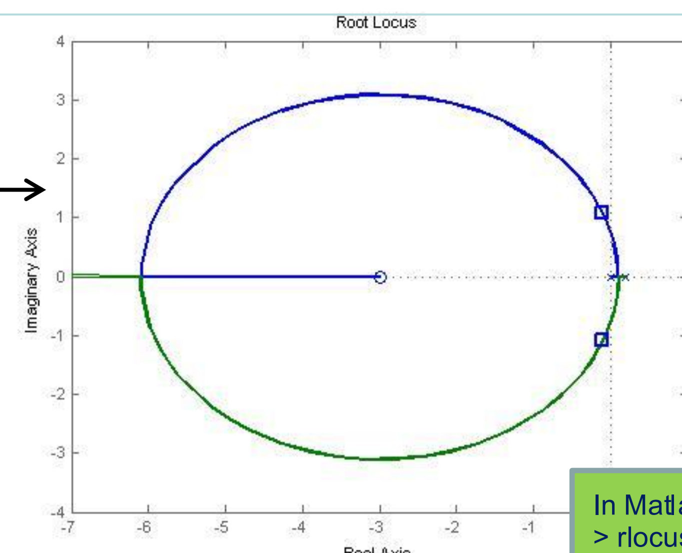


Z=0
(Stable)

Root Locus Analysis

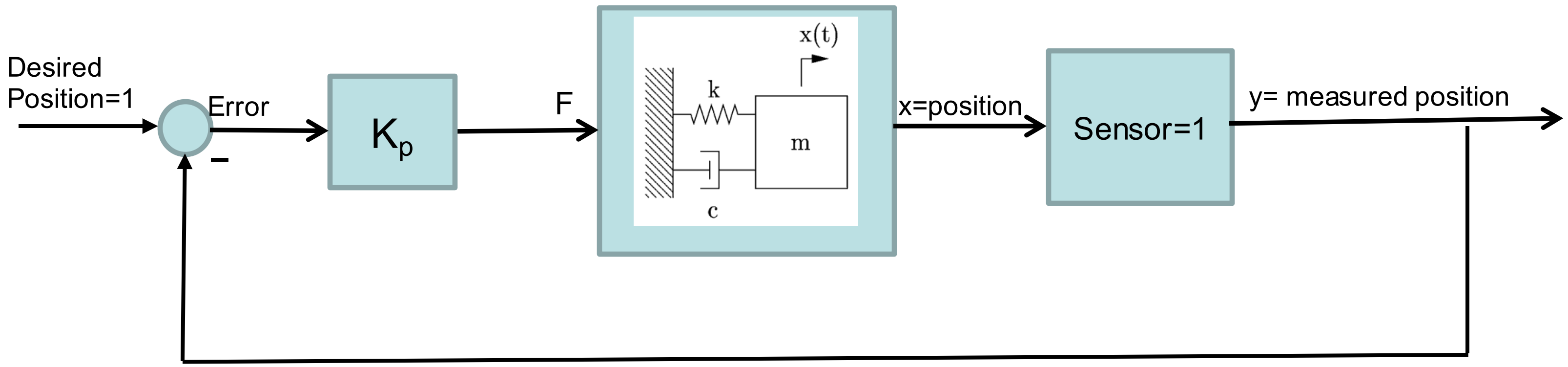


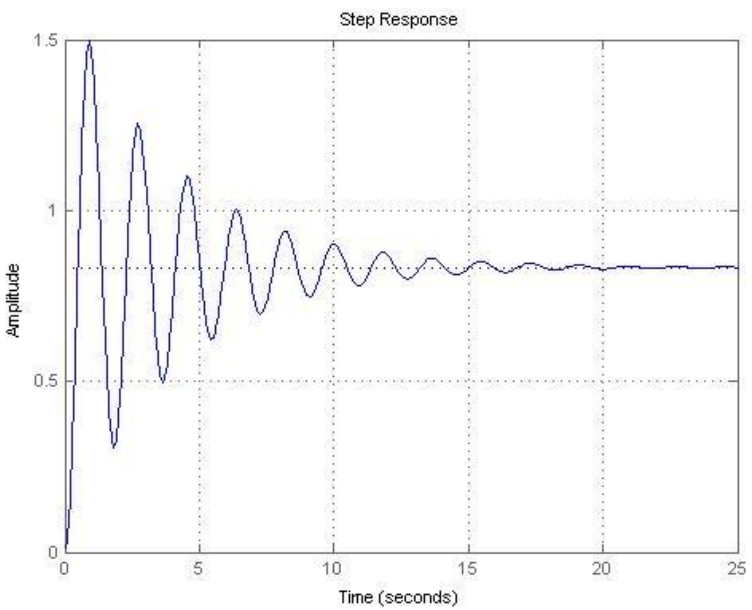
Vary feedback Gain K from (0, ∞) and observe pole movement



In Matlab:
> rlocus(Ls)

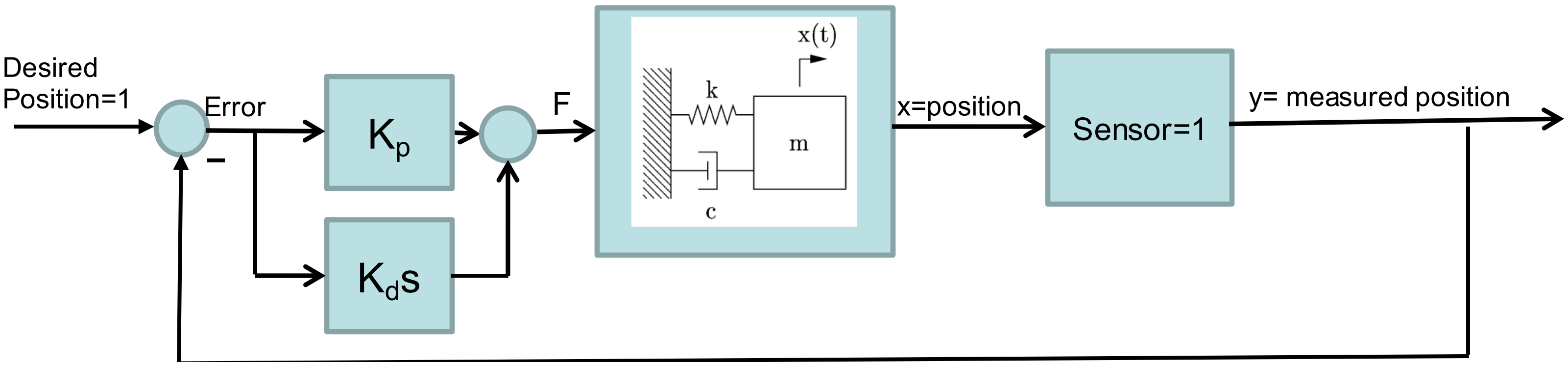
Proportional (P) Control Only

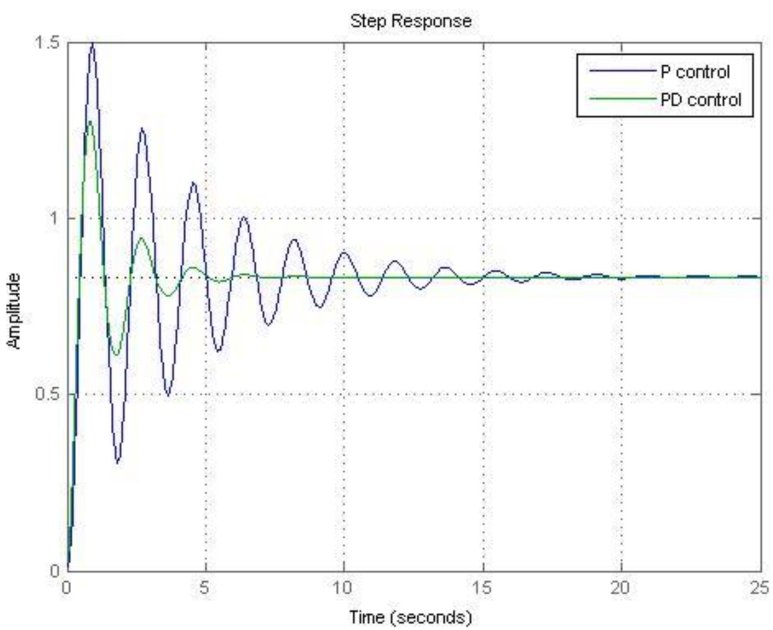





```
sys=ss(A,B,C,D)

kp=10

cont=pid(kp);   % p control only

Tp=feedback(cont*sys,1)

step(Tp)
```

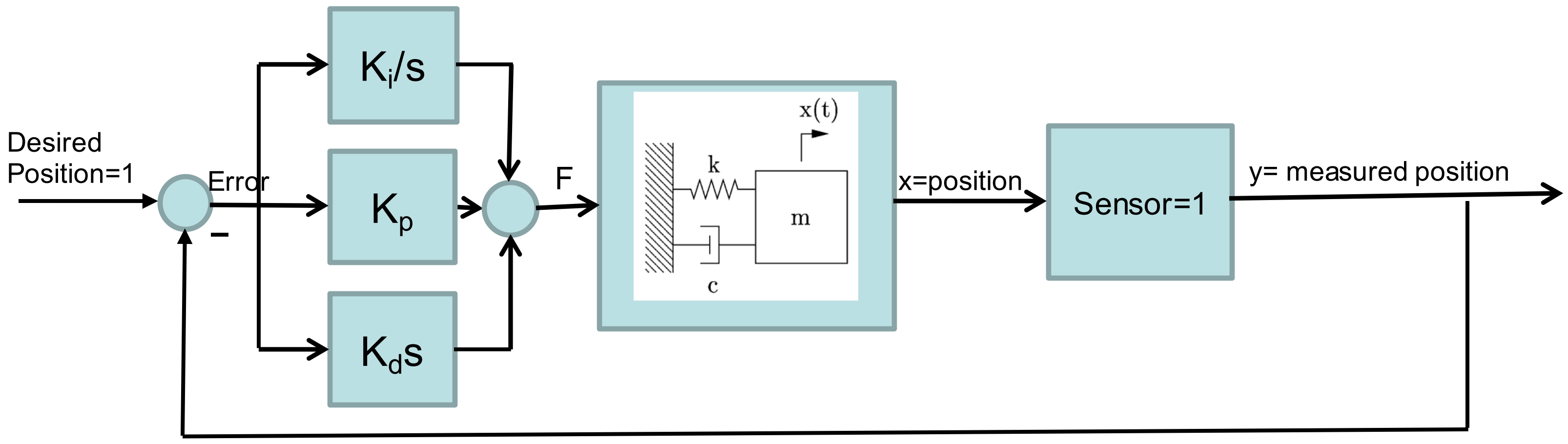
Proportional –Derivative (PD) Control

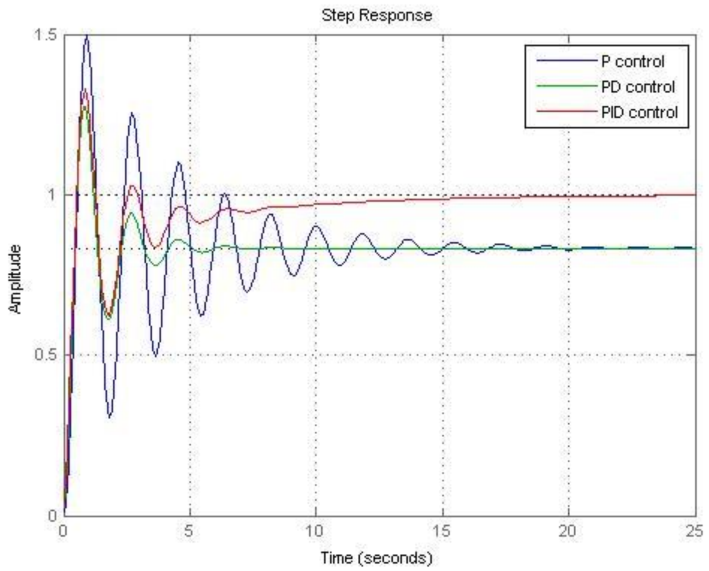





```
sys=ss(A,B,C,D)
kp=10
kd=1

cont=pid(kp, 0, kd);  % pd control

Tpd=feedback(cont*sys,1)
```

Proportional –Integral-Derivative (PID) Control

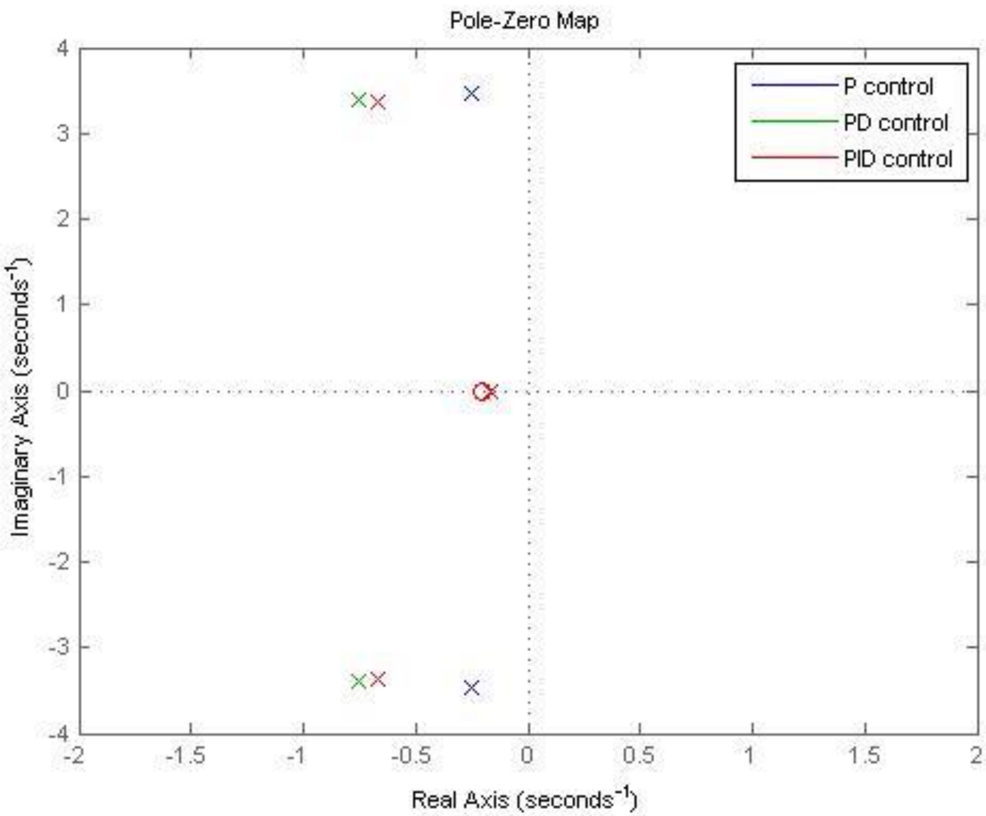





```
sys=ss(A,B,C,D)
kp=10
kd=1
ki=2

cont=pid(kp, ki, kd);  % pid control

Tpid=feedback(cont*sys,1)
```

PID control is a very common control approach for linear systems, used by Saturn, Shuttle, Ares I-X, SLS, etc,

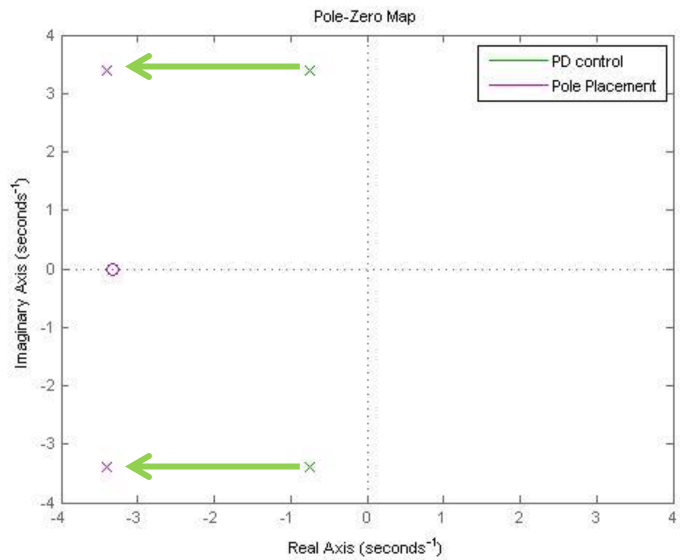
Compare Poles of Three Designs

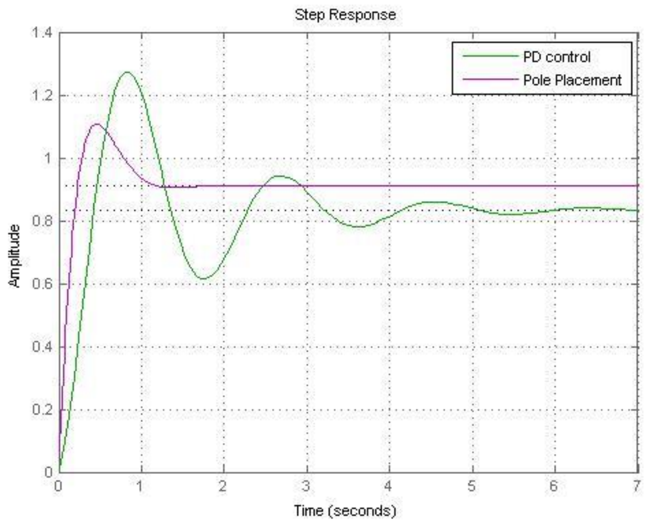


In Matlab:
>> pzmap(Tp,'b', Tpd, 'g', Tpid, 'r')

- **Observations:**
  - All controllers (P, PD, PID) are stable because the poles are in the left hand plane.
  - The P controller (in blue) is perhaps the least stable because its pole is closest to the imaginary axis line.
  - What if we pick our gains (say K_p and K_d) to move to closed-loop poles to a particular location?
  - This design approach is called 'pole placement'.

Pole Placement Control Design, PD Control





- **Previous PD design used gains of: k_p=10, k_d=1**
  - Stable design with decent performance
- **Lets move the closed-loop poles to a new, more stable position:**
  - Lets choose: p=[-3.4+3.4i -3.4-3.4i]
  - Have Matlab compute the feedback gains to move the poles to this new location:

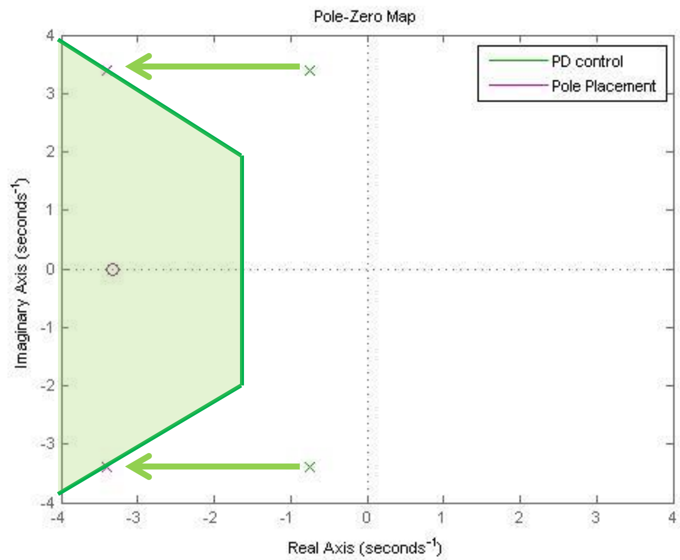
```
>> Knew = place(A,B,p)
>> Knew =
        21.1200    6.3000
```

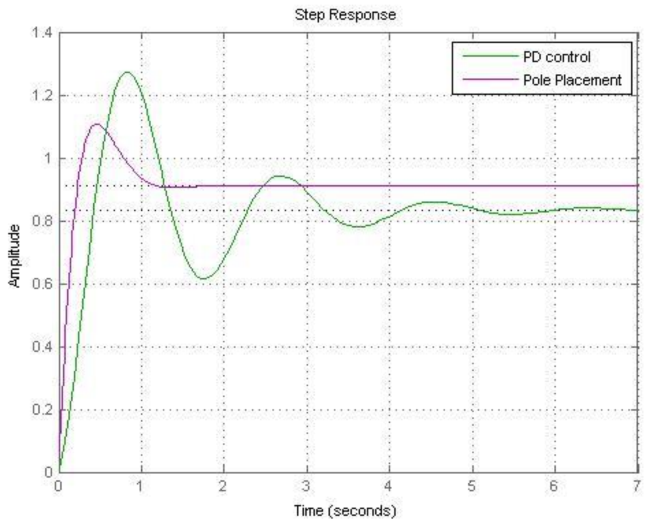
  - Recompute closed loop response with new gains:

```
>> cont=pid(Knew(1), 0, Knew(2));

>> ss_cl=feedback(cont*sys,1)
```

- **New control design improves performance:**
  - Smaller overshoot
  - Smaller steady state error

Pole Placement Control Design, PD Control





- **Previous PD design used gains of: k_p=10, k_d=1**
  - Stable design with decent performance
- **Lets move the closed-loop poles to a new, more stable position:**
  - Lets choose: p=[-3.4+3.4i -3.4-3.4i]
  - Have Matlab compute the feedback gains to move the poles to this new location:

```
>> Knew = place(A,B,p)
>> Knew =
          21.1200    6.3000
```

  - Recompute closed loop response with new gains:

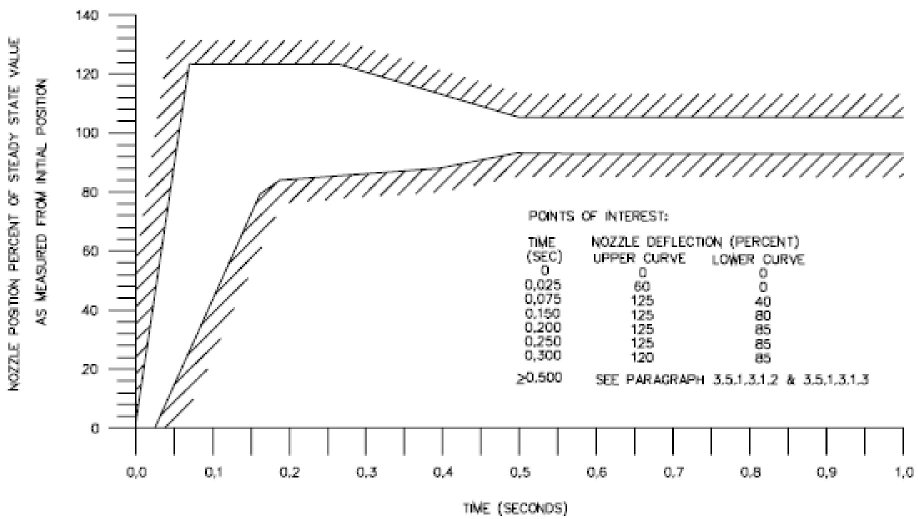
```
>>  cont=pid(Knew(1), 0, Knew(2));

>>  ss_cl=feedback(cont*sys,1)
```

- **New control design improves performance:**
  - Smaller overshoot
  - Smaller steady state error

Note on Step Response

- A unit step input is a very common way to define an envelop of allowable closed-loop system time response.
- At time=0, a "step" command is given to the system, and one defines an allowable response envelope which controls:
  - Overshoot
  - Rise time
  - Allowable steady state error
- For example: Shuttle Step Response Requirement on closed-loop TVC system:

IRN No. BI-0195



TIME (SEC)	NOZZLE DEFLECTION (PERCENT) UPPER CURVE	LOWER CURVE
0	0	0
0,025	60	0
0,075	125	40
0,150	125	80
0,200	125	85
0,250	125	85
0,300	120	85
≥0.500	SEE PARAGRAPH 3.5.1.3.1.2 & 3.5.1.3.1.3	

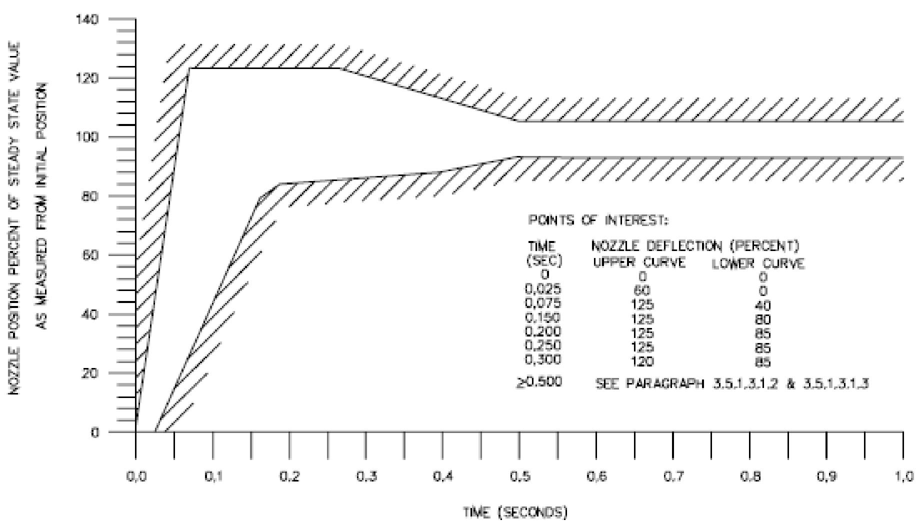
ICD-2-14001

ORBITER VEHICLE/SOLID ROCKET BOOSTER

Note on Step Response

- A unit step input is a very common way to define an envelop of allowable closed-loop system time response.
- At time=0, a “step” command is given to the system, and one defines an allowable response envelope which controls:
  - Overshoot
  - Rise time
  - Allowable steady state error
- For example: Shuttle Step Response Requirement on closed-loop TVC system:

IRN No. BI-0195



TIME (SEC)	NOZZLE DEFLECTION (PERCENT) UPPER CURVE	LOWER CURVE
0	0	0
0,025	60	0
0,075	125	40
0,150	125	80
0,200	125	85
0,250	125	85
0,300	120	85
≥0.500	SEE PARAGRAPH 3.5.1.3.1.2 & 3.5.1.3.1.3	

Control system design is a trade between margins from instability vs. time domain performance.

Overly-large stability margins can result in poorer control performance.

ICD-2-14001

ORBITER VEHICLE/SOLID ROCKET BOOSTER

Summary

- Linear dynamic systems, represented by differential equations, can be modeled both with Laplace Transforms and Space State Formulations.
- For asymptotic stability, a necessary condition is these systems must have negative eigenvalues, which is the same thing as having transfer function poles in the left hand plane.
- Feedback is added to increase performance, including stabilizing an unstable system.
- PID control is a commonly used approach to utilize feedback and shape the closed-loop response.
- Pole placement is one approach to design the control system gains (including PID) to achieve the desired performance and margins.
- A step response is a common approach to quantify system time response, including for design requirements.
- Control system design is a trade between margins from instability vs. time domain performance.

References

1. Ogata, K., *Modern Control Engineering*, 2nd Edition, Prentice Hall, New Jersey, 1990.
2. Lurie, B., Enright, P., *Classical Feedback Control with Matlab*, Marcel Dekker, New York, 2000.
3. Garcia-Sanz, M., “Stability Criteria in Non-Polar Diagrams”, Int. J. Elect. Engin. Educ., Vol 36, pp 65-72, Manchester U.P, 1999.
4. Frosch, J., Vallely, D., “Saturn AS-501/S-IC Flight Control System Design”, J. Spacecraft, Vol. 4, No. 8, August 1967.
5. Kreyszig, E., *Advanced Engineering Mathematics*, John Wiley & Sons, Inc., 2006.
6. Nise, N., *Control Systems Engineering*, Benjamin/Cummings Publishing Company, Inc., 1992.