7 Control Systems

7.1 On-Off (Bang-Bang) Control

On-off (bang-bang) control is the most basic type of control system, operating by simply switching the output fully on or fully off in response to the measured variable. When the process value falls below a setpoint, the controller turns the system on; when it exceeds the setpoint, it turns it off. This approach is common in applications where precise control is not required, such as hot water heaters and HVAC systems, offering a simple, low-cost, and reliable method of maintaining conditions within an acceptable range.

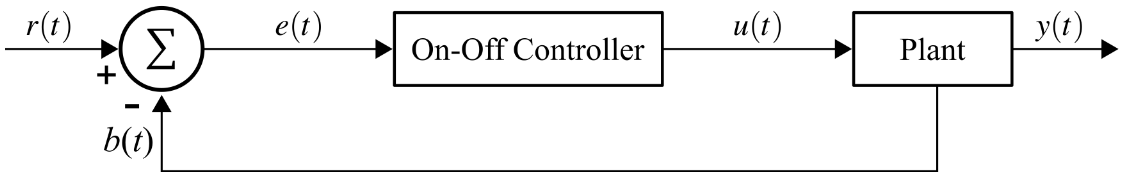


Figure 7.7: On-Off (Bang-Bang) controller for a system with feedback, where $r(t)$ is the desired setpoint (SP) and $y(t)$ is the measured process value (PV).

7.2 Proportional-Integral-Derivative (PID) Control

Proportional-Integral-Derivative (PID) Control is a three-term controller that employs feedback that is widely used in continuous control systems, including for the control of structural systems. A PID controller seeks to minimize the measured error value $e(t)$ between a desired setpoint (SP) and a measured process variable (PV) by applying corrections based on the proportional (P), integral (I), and derivative (D) terms (denoted P, I, and D respectively), from which it gets its name.

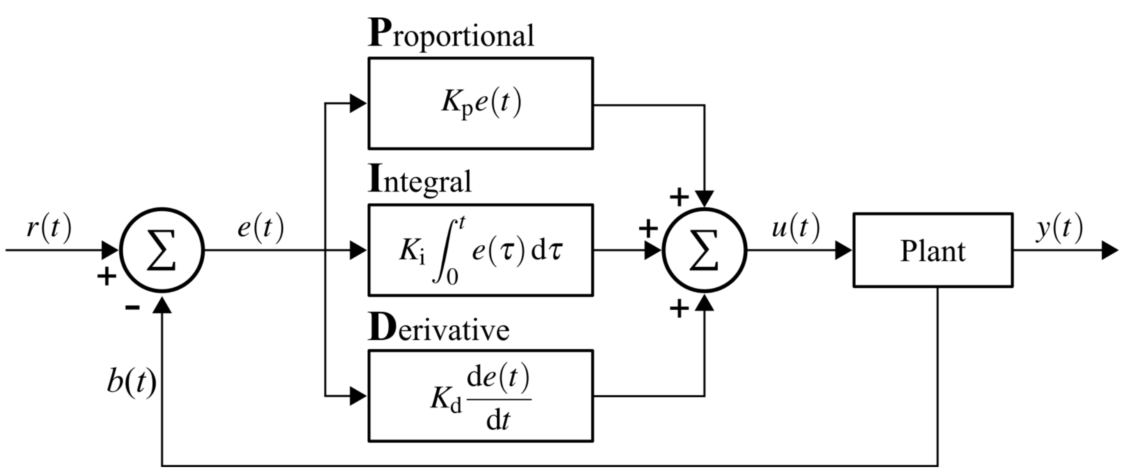


Figure 7.8: Generalized PID controller for a system with feedback, where $r(t)$ is the desired setpoint (SP) and $y(t)$ is the measured process value (PV).

The overall control equation is defined as

$$u(t) = K_\mathrm{p} e(t) + K_\mathrm{i} \int_0^t e(\tau)\,\mathrm{d}\tau + K_\mathrm{d} \frac{\mathrm{d}e(t)}{\mathrm{d}t} \tag{7.7}$$

where K_p, K_i, and K_d are non-negative coefficients for the proportional, integral, and derivative terms, respectively. The PID controller is diagrammed in figure 7.8 for a system with feedback control, such as that shown in figure **??**. Moreover, in the Laplace-derived s domain, the transfer function of the PID controller is defined as

$$\mathscr{L}[s] = K_\mathrm{p} + \frac{K_\mathrm{i}}{s} + K_\mathrm{d} s \tag{7.8}$$

where s is the complex frequency. A temporal response for the 1-DOF shown in figure **??** when controlled with a PID controller is reported in figure 7.9.

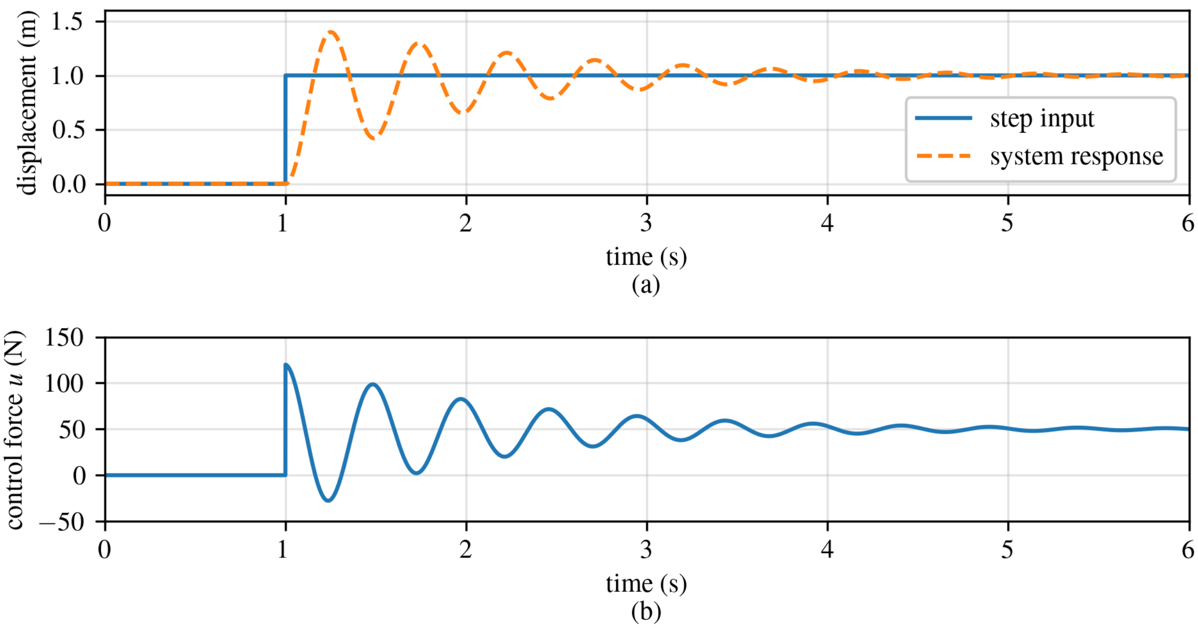


Figure 7.9: Response for a 1-DOF system controlled with a PID controller, showing: (a) system response and the step function input, and (b) the controller response imposed on the system to obtain such responses.

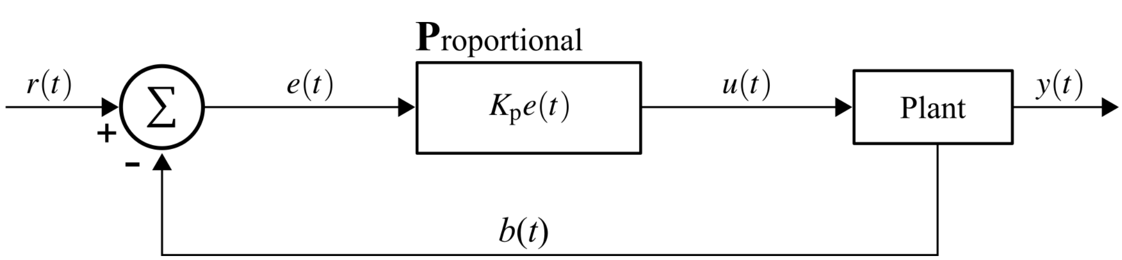


Figure 7.10: Generalized PID controller for a system with feedback, where $r(t)$ is the desired setpoint (SP) and $y(t)$ is the measured process value (PV).

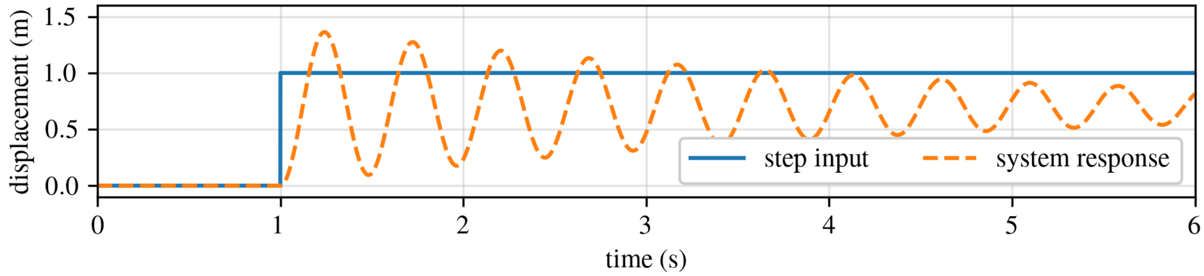


Figure 7.11: Response for a 1-DOF system controlled with a just a proportional (P) controller.

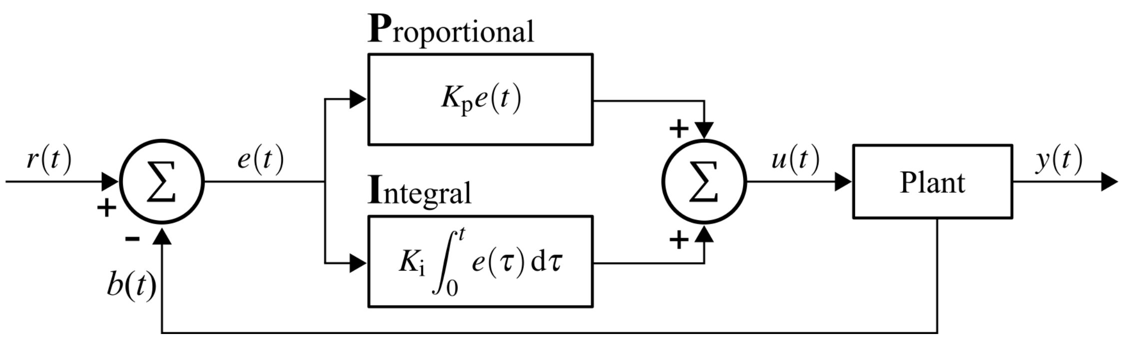


Figure 7.12: Generalized PID controller for a system with feedback, where $r(t)$ is the desired setpoint (SP) and $y(t)$ is the measured process value (PV).

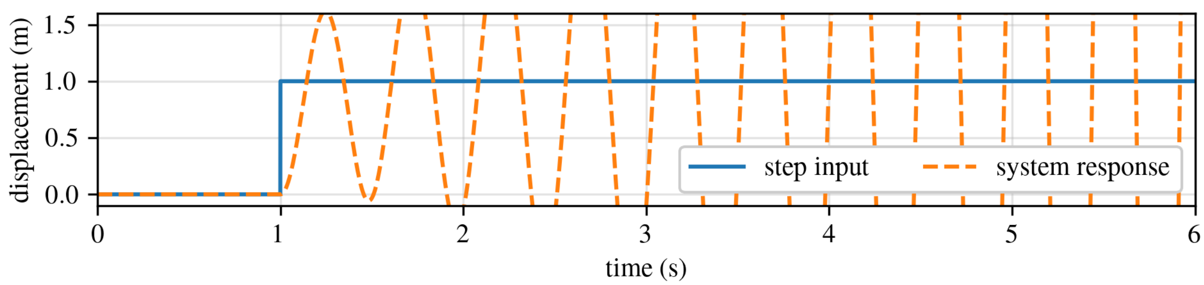


Figure 7.13: Response for a 1-DOF system controlled with a PI controller.

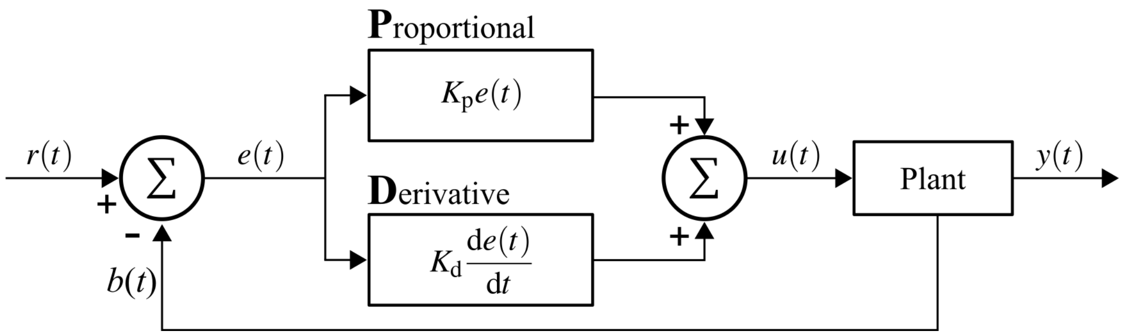


Figure 7.14: Generalized PID controller for a system with feedback, where $r(t)$ is the desired setpoint (SP) and $y(t)$ is the measured process value (PV).

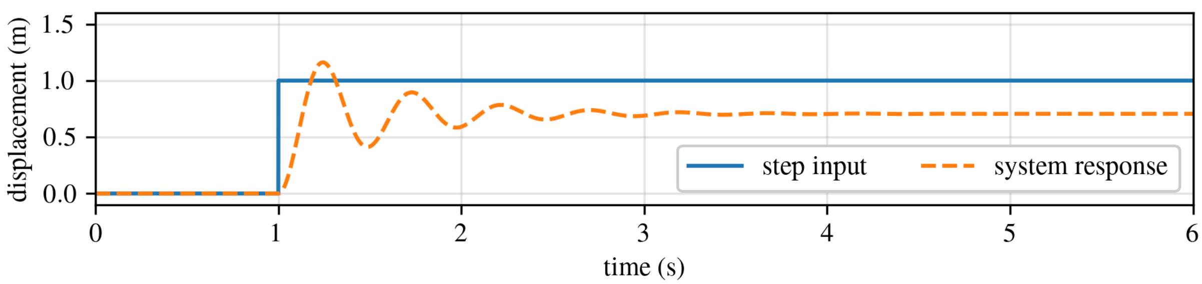
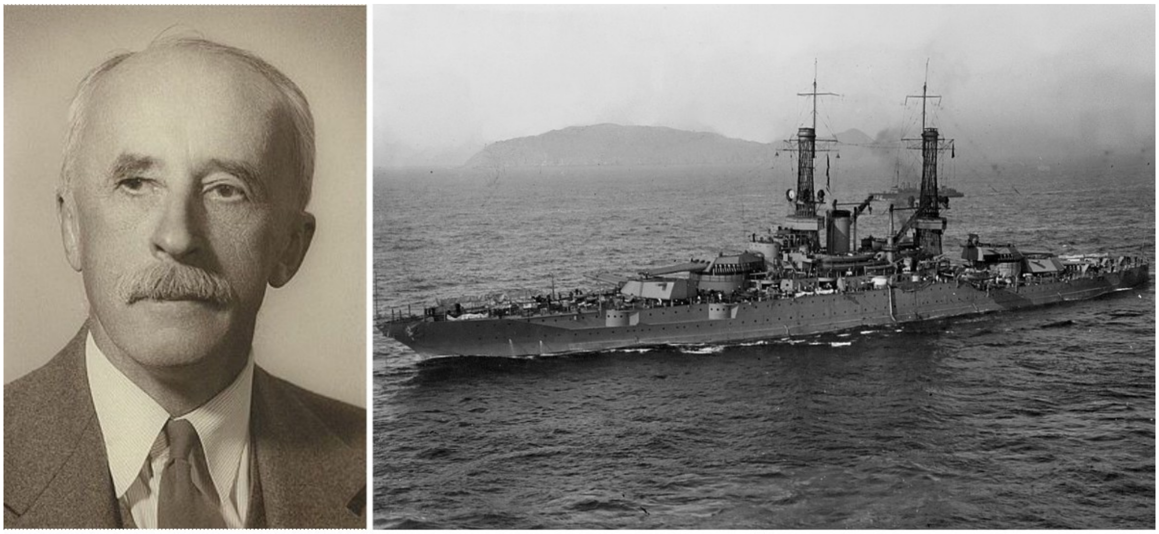


Figure 7.15: Response for a 1-DOF system controlled with a PD controller.

Review 7.1 Nicolas Minorsky and the Need for Better Control

Continuous control systems have been widely used for centuries. For example, consider that the centrifugal governor which uses spinning weights was used by Christiaan Huygens in the 1600s in the Netherlands to regulate the gap between millstones in windmills or by James Watt who famously linked a stem regulator to a centrifugal governor to control steam turbines.

Arguably, the Russian American engineer Nicolas Minorsky was the first to develop the theoretical analysis for the three-term control we now call PID. This was done in 1922 while he was researching and designing automatic ship steering for the US Navy. He based his work on watching how a ship's helmsman responds to wave loading on a ship, with a delayed input to the helm that not only considered the current ship course but also past errors and the desired rate of change for the ship. For a helmsman, the goal is stability, not absolute control, which simplifies how one thinks about the challenge of control.



(a) (b)

Figure 7.16: Historical perspective of PID control showing: (a) Portrait of Nicolas Minorsky [1] and (b) the battleship USS New Mexico (BB-40) of the United States Navy which was the first to implement PID control in its steering [2].

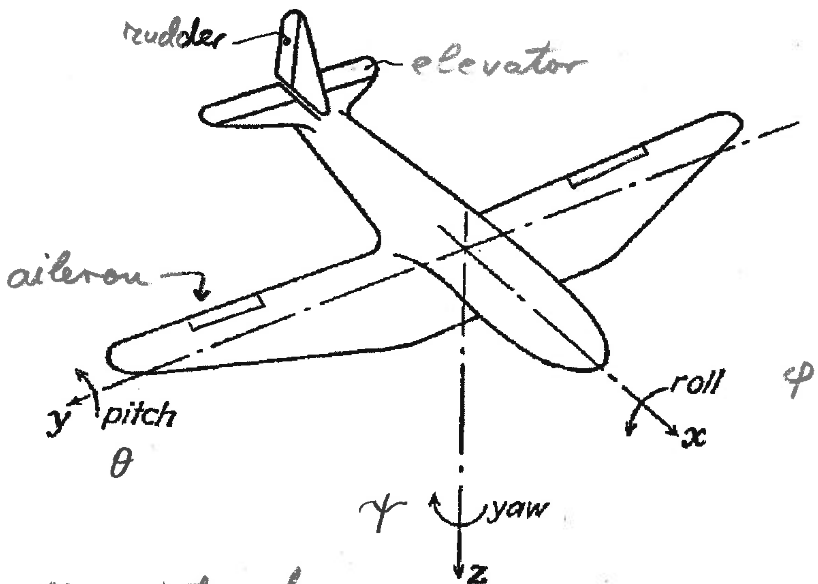
[a]Peter Minorsky, grandson of Nicolas Minorsky, CC BY-SA 1.0 <https://creativecommons.org/licenses/by-sa/1.0>, via Wikimedia Commons

[b]U.S. Navy, Public domain, via Wikimedia Commons

7.3 Control Systems

2016 11 03

AIRCRAFT CONTROLS



Aircraft control surfaces

AILERONS control the rolling motion

φ = roll angle ($\sim$ bank angle)

pilot stick moves left-right to deflect ailerons

ELEVATOR controls the pitch motion

nose up / nose down

θ = pitch angle

pilot stick moves forward-backward to deflect elevator

RUDDER controls yaw motion

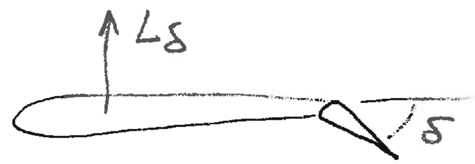
ψ = yaw angle

pilot uses rudder pedals to deflect rudder

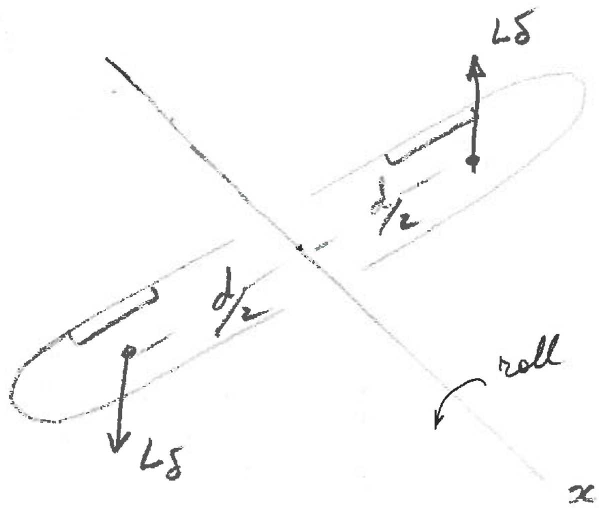
165/523

2016 11 03

AILERON DEFLECTION
AIRCRAFT ROLL MOTION



$$L_\delta = \frac{\partial L}{\partial \delta}\delta$$



- Aileron deflection δ produces additional lift L_δ
- L_δ is up/down because ailerons left/right move up/down
- Net effect is a rolling moment,

$$M = L_\delta d = d \cdot \frac{\partial L}{\partial \delta} \cdot \delta \qquad (1)$$

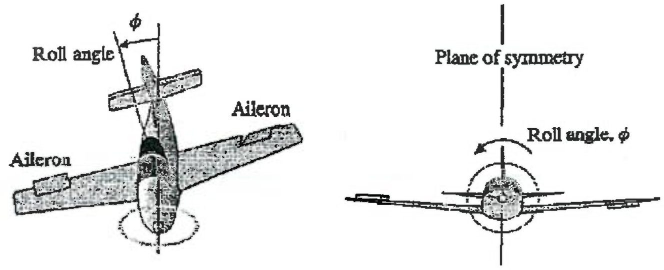
- Essentially, the rolling moment M is proportional to aileron deflection δ, i.e.

$$M = K\delta \qquad (K = \text{gain}) \qquad (2)$$

166/523

RTF

ROLL TRANSFER FUNCTION



EOM: $J\ddot{\varphi} + c\dot{\varphi} = M$ (1)

where

J = inertia : mass moment of inertia about roll axis

c = damping : air resistance to roll motion

M = rolling moment produced by aileron deflection δ

Recall:

$M = K\delta$ (2)

(2) → (1): $J\ddot{\varphi} + c\dot{\varphi} = K\delta$ (3)

Take LT of Eq.(3) to get

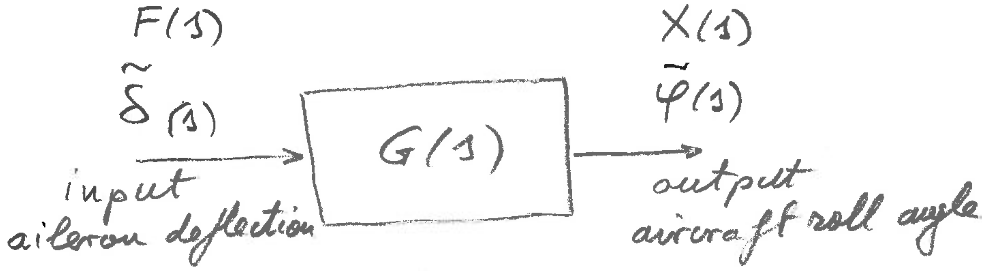
$\mathcal{L}(3)$: $(Js^2 + cs)\tilde{\varphi}(s) = K\tilde{\delta}(s)$ (4)

where $\tilde{\varphi}(s) = \mathcal{L}\,\varphi(t)$
$\tilde{\delta}(s) = \mathcal{L}\,\delta(t)$ (5)

Solution of Eq.(4) yields:

RTP

$$\tilde{\varphi}(s) = \frac{K}{Js^2 + cs}\,\tilde{\delta}(s) \qquad (6)$$



$$G(s) = \frac{K}{Js^2 + cs} \qquad (7)$$

$$X(s) = G(s)\,F(s). \qquad (8)$$

The system described by Eq.(7) is an uncontrolled 2nd order dynamic system.

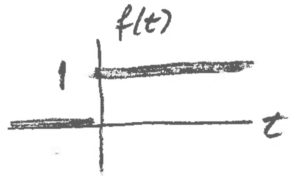
- this system is "uncontrolled" because a constant input will produce a continuously growing response (see roll response on next page)
- Another example of uncontrolled 2nd order dynamic system is the DC Motor where an applied constant voltage produces continuous rotation

168/523

RR 20161109

Roll Response to Constant Aileron Deflection

Assume the pilot moves the stick laterally such as to create a constant aileron deflection $\delta = \text{const.}$ This corresponds to a step input, i.e.,



$$\begin{cases} f(t) = 1 \\ F(s) = 1/s \end{cases} \tag{9}$$

(9) → (8):
$$X(s) = \frac{K}{Js^2 + cs} \cdot \frac{1}{s} = \frac{K}{s^2(Js + c)} \tag{10}$$

Table 2-1, # 19 has the pair

$$t - T\left(1 - e^{-t/T}\right) \underset{ILT}{\overset{LT}{\rightleftarrows}} \frac{1}{s^2(Ts+1)} \tag{11}$$

Write (10) such as to look like (11), i.e.,

$$X(s) = \frac{K}{c} \cdot \frac{1}{s^2(Ts+1)}, \quad T = \frac{J}{c} \tag{12}$$

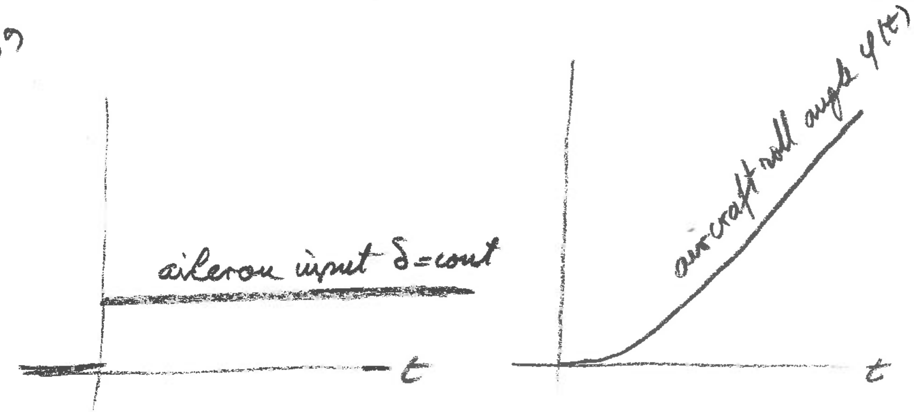
ILT of (12) gives:

$$x(t) = \frac{K}{c}\left[t - T\left(1 - e^{-t/T}\right)\right] \tag{13}$$

$$x_{ss} = \lim_{t\to\infty} x(t) = \frac{K}{c}\left(t - T\right) \tag{14}$$

169/523

2
RR
2016 11 09



Plot of eq. (13) indicates that the aircraft will roll continuously with a constant roll rate.

Thus, a constant aileron input δ=const produces continuously increasing roll angle of aircraft.

This is a general property of Type 1 system: they cannot maintain position.
The step response of a Type 1 system is unconstrained

- DC motor spins continuously under constant voltage input
- Aircraft rolls continuously under constant aileron input

170/523

3 RR 2016 11 09

Step response of Type 1 systems

$$G(s) = \frac{K}{s} \cdot \frac{(T_a s+1)(T_b s+1)\cdots}{(T_1 s+1)(T_2 s+1)\cdots} \qquad \text{Type 1}$$

$$F(s) = \frac{1}{s} \qquad \text{step excitation}$$

$$X(s) = G(s)\,F(s)$$

$$= \frac{K}{s} \, \frac{(T_a s+1)(\quad)\cdots}{(T_1 s+1)(\quad)\cdots} \cdot \frac{1}{s}$$

$$X(s) = \frac{K}{s^2} \, \frac{(\quad)(\quad)\cdots}{(\quad)(\quad)\cdots} \qquad (1)$$

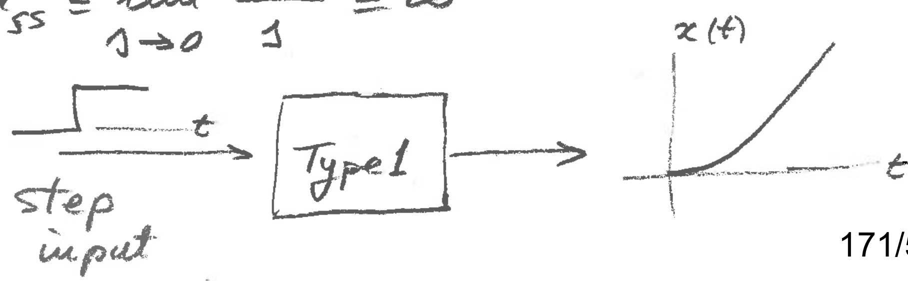
Steady state response is calculated with Final Value Theorem, i.e.,

$$x_{ss} = \lim_{t \to \infty} x(t) = \lim_{s \to 0} s\,X(s) \qquad (2)$$

(1) → (2):

$$x_{ss} = \lim_{s \to 0} \cancel{s} \frac{K}{s^{\cancel{2}}} \, \frac{\overbrace{(T_a s+1)(T_b s+1)\cdots}^{\xrightarrow{s\to 0} 1}}{\underbrace{(T_1 s+1)(T_2 s+1)\cdots}_{\to 1}}$$

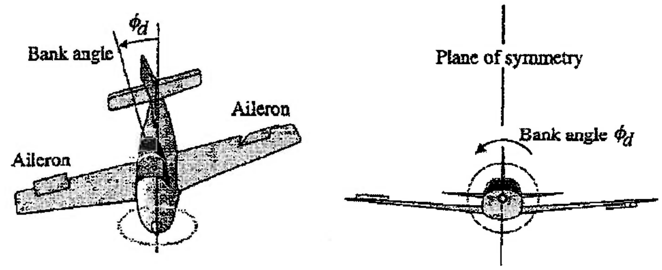
$$x_{ss} = \lim_{s \to 0} \frac{K}{s} = \infty$$



171/523

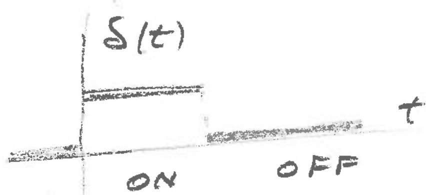
BAC

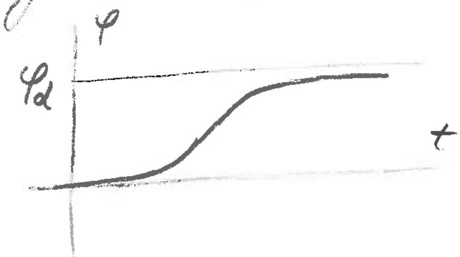
BANK ANGLE CONTROL



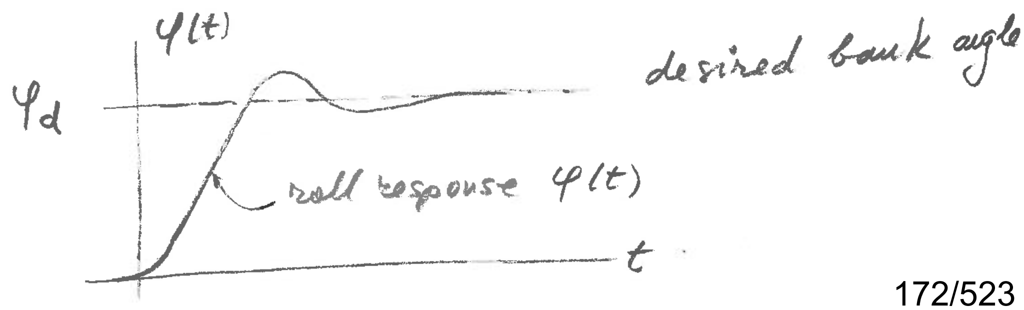
We want the aircraft to roll from flying straight & level to flying inclined with a bank angle $\varphi_d = $ const.

- Manually, the pilot creates a bank angle by an on/off deflection of aileron



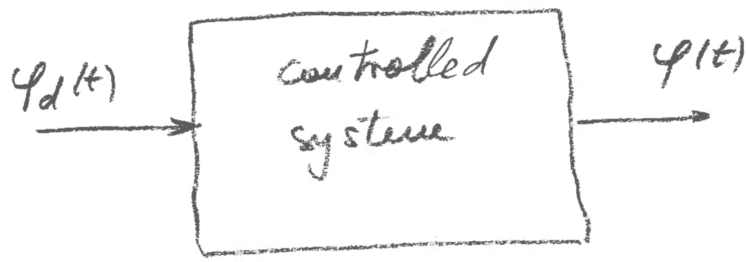


- We desire to build a FB control system to achieve this transition in a smooth way automatically.



172/523

2 BAC



The control system design process needs specifications. We choose two performance indicators, M_p and t_p and define their values as design specs.

Control design specifications

DS1 : Fast response time as measured by rise time, $t_r \leq 1.5\,sec$.

DS2 : Maximum percentage overshoot for step input less than 20%

$M_p \leq 20\%$

173/523

BAC Details of:

MANUAL BANK ANGLE CONTROL

- Aircraft flies straight & level

- Pilot wants to bank 15°

- Pilot moves stick sideways.

- Aircraft starts to respond

- Pilot's eyes see the aircraft rolling and estimates actual value φ
- Pilot's mind processes the measured value φ, compares it with desired value $\varphi_d = 15°$, and sends action order to hand muscles!
  - if $\varphi < \varphi_d$, then continue to push stick
  - if $\varphi \sim \varphi_d$, move stick back to neutral position to reduce rolling moment from ailerons
  - if $\varphi > \varphi_d$, move stick the other way because the aircraft overshot the target angle and needs to roll backwards

174/523

1
FB
2016.1109

FEEDBACK CONTROL

FB concept:

- adjust the system input to obtain a desired output

FB implementation in Laplace s-domain:

- measure output
- calculate "error", i.e. difference between "desired" and "measured"
- feed "error" to the system to adjust itself until the "error" is reduced to zero.

(i.e., "measured" = "desired")

In the time domain, the FB process is a transient process of repeated adjustments until output matches input (error $\to$ 0).

The process is based on Convolution Theorem of Laplace Transform

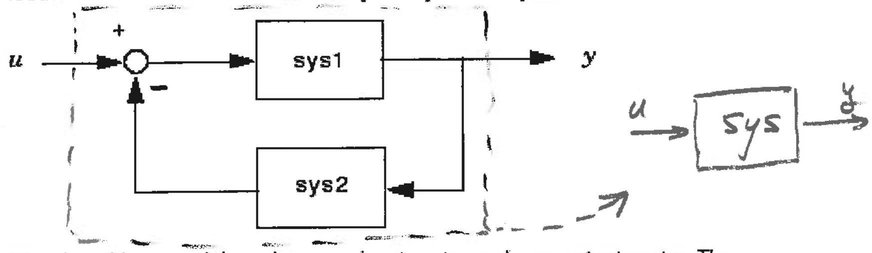
$$\mathcal{L}^{-1} G(s) F(s) = \int_0^t f(\tau) g(\tau - t) d\tau$$

175/523

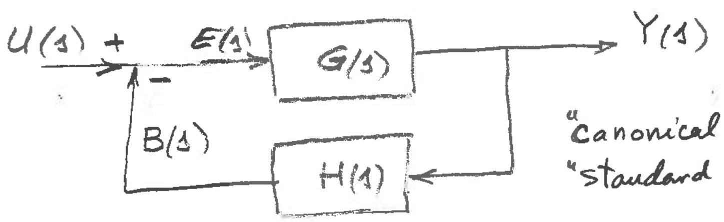
2 FB 2016/11/10

MATLAB:

`sys = feedback(sys1,sys2)` returns a model object `sys` for the negative feedback interconnection of model objects `sys1` and `sys2`.

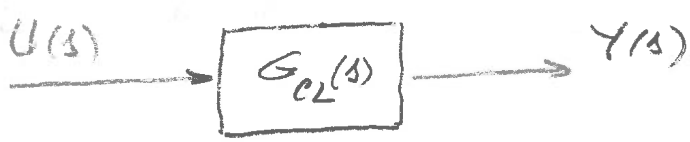


The closed-loop model `sys` has u as input vector and y as output vector. The models `sys1` and `sys2` must be both continuous or both discrete with identical sample times. Precedence rules are used to determine the resulting model type



FB sys.

"canonical form"
"standard form"



Equivalent FB sys.

$$G_{CL}(s) = \frac{G(s)}{1+G(s)H(s)}$$

closed loop transfer funct.

Proof

$$B(s) = H(s)Y(s)$$

$$E(s) = U(s) - B(s) = U(s) - H(s)Y(s)$$

$$Y(s) = G(s)E(s) = G(s)U(s) - G(s)H(s)Y(s)$$

$$Y(s) + G(s)H(s)Y(s) = G(s)U(s)$$

$$\left[1 + G(s)H(s)\right]Y(s) = G(s)U(s)$$

$$Y(s) = \frac{G(s)}{1+G(s)H(s)}U(s)$$

QED

3 FB 2016 11 10

FB Nomenclature

$U(s)$: input reference signal (desired result)

$Y(s)$: output signal

$B(s)$: feedback signal

$E(s)$: error signal

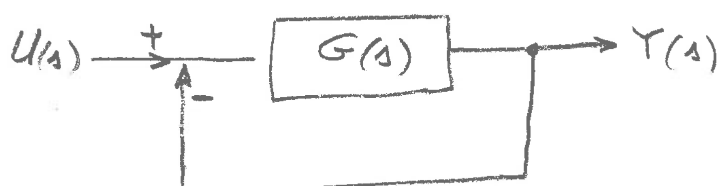
Transfer functions

feedforward T.F: $G(s) = \frac{Y(s)}{E(s)}$

open loop TF: $G(s)H(s) = \frac{B(s)}{E(s)}$

closed loop TF: $G_{CL}(s) = \frac{Y(s)}{U(s)}$.

UNIT FEEDBACK



"unit feedback" is obtained for $H(s)=1$

$$G_{CL} = \frac{G}{1+G}$$ unit feedback closed loop TF.

177/523

feedback

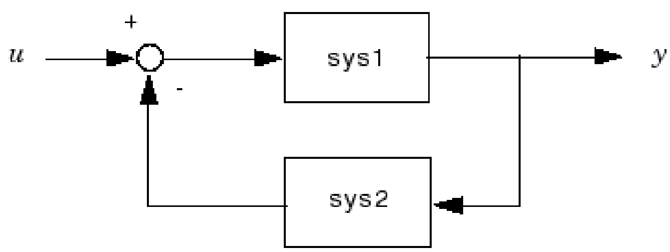
Feedback connection of two models

Syntax

```
sys = feedback(sys1,sys2)
```

Description

`sys = feedback(sys1,sys2)` returns a model object `sys` for the negative feedback interconnection of model objects `sys1` and `sys2`.



The closed-loop model `sys` has u as input vector and y as output vector. The models `sys1` and `sys2` must be both continuous or both discrete with identical sample times. Precedence rules are used to determine the resulting model type (see “Rules That Determine Model Type”).

To apply positive feedback, use the syntax

```
sys = feedback(sys1,sys2,+1)
```

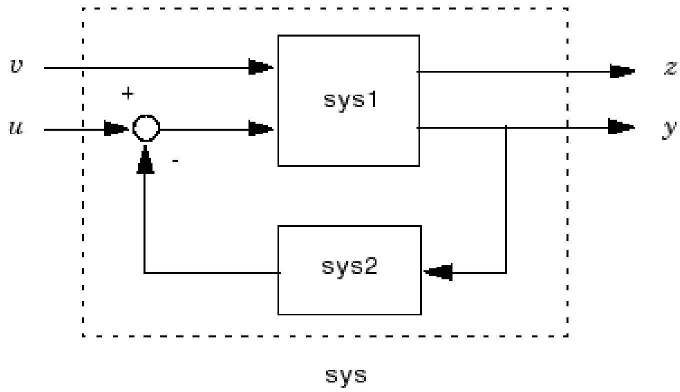
By default, `feedback(sys1,sys2)` assumes negative feedback and is equivalent to `feedback(sys1,sys2,-1)`.

Finally,

```
sys = feedback(sys1,sys2,feedin,feedout)
```

178/523

computes a closed-loop model `sys` for the more general feedback loop.



The vector `feedin` contains indices into the input vector of `sys1` and specifies which inputs u are involved in the feedback loop. Similarly, `feedout` specifies which outputs y of `sys1` are used for feedback. The resulting model `sys` has the same inputs and outputs as `sys1` (with their order preserved). As before, negative feedback is applied by default and you must use

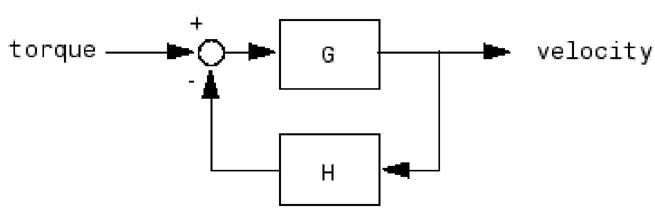
```
sys = feedback(sys1,sys2,feedin,feedout,+1)
```

to apply positive feedback.

For more complicated feedback structures, use `append` and `connect`.

Examples

Example 1



To connect the plant

$$G(s) = \frac{2s^2 + 5s + 1}{s^2 + 2s + 3}$$

with the controller

$$H(s) = \frac{5(s + 2)}{s + 10}$$

using negative feedback, type

```
G = tf([2 5 1],[1 2 3],'inputname','torque',...
                         'outputname','velocity');
H = zpk(-2,-10,5)
Cloop = feedback(G,H)
```

These commands produce the following result.

```
Zero/pole/gain from input "torque" to output "velocity":
0.18182 (s+10) (s+2.281) (s+0.2192)
-----------------------------------
 (s+3.419) (s^2 + 1.763s + 1.064)
```

The result is a zero-pole-gain model as expected from the precedence rules. Note that `Cloop` inherited the input and output names from `G`.

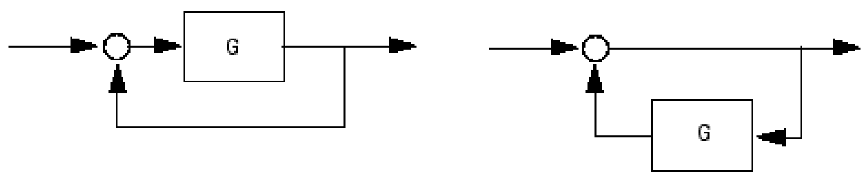
Example 2

Consider a state-space plant `P` with five inputs and four outputs and a state-space feedback controller `K` with three inputs and two outputs. To connect outputs 1, 3, and 4 of the plant to the controller inputs, and the controller outputs to inputs 4 and 2 of the plant, use

```
feedin = [4 2];
feedout = [1 3 4];
Cloop = feedback(P,K,feedin,feedout)
```

Example 3

You can form the following negative-feedback loops



by

```
Cloop = feedback(G,1)       % left diagram
Cloop = feedback(1,G)      % right diagram
```

Limitations

The feedback connection should be free of algebraic loop. If D_1 and D_2 are the feedthrough matrices of `sys1` and `sys2`, this condition is equivalent to:

- $I + D_1D_2$ nonsingular when using negative feedback
- $I - D_1D_2$ nonsingular when using positive feedback.

See Also

`series` | `parallel` | `connect`

Introduced before R2006a

1 FB

Feedback control of Type 1 systems.

- Type 1 system response is unconstrained
- Feedback can be used to control the response.

Examples

- DC motor cannot hold position; it rotates continuously under constant voltage
  With FB, a DC motor becomes a servomotor and holds position
- Aircraft rolls continuously; with FB, aircraft can maintain a constant bank angle.

Type 1 system transfer function

$$G(s) = \frac{K}{Js^2 + cs}$$

$\left\{ \begin{array}{l} K = \text{gain} \\ J = \text{inertia} \\ c = \text{damping} \end{array} \right.$

2nd order system : s^2 is highest power in denominator

Type 1 system : $G(s) = \frac{K}{s} \cdot \frac{1}{Js + c}$

"s to power 1"

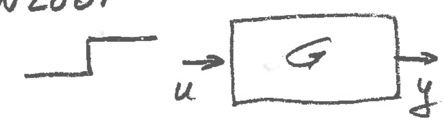
182/523

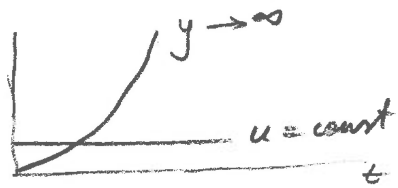
8 FB

FB Step response of Type 1 system

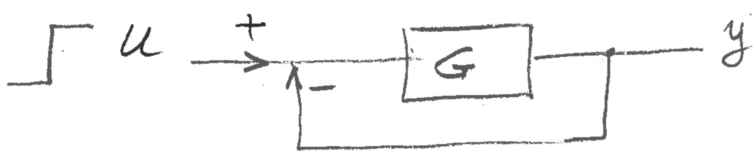
$$G(s) = \frac{K}{Js^2 + cs}$$

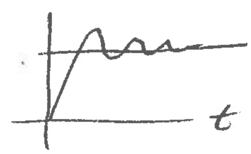
OPENLOOP





CLOSED LOOP w FB





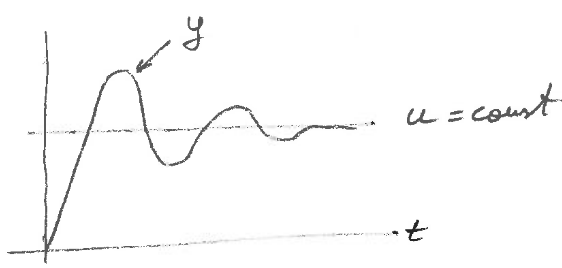
$$G_{CL} = \frac{G}{1+G}$$

$$= \frac{\frac{K}{Js^2+cs}}{1 + \frac{K}{Js^2+cs}} = \frac{K}{Js^2+cs+K} = \frac{\frac{K}{J}}{s^2 + \frac{c}{J}s + \frac{K}{J}}$$

$$G_{CL} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$ { 2^{nd} order system, Type 0 system

$$\omega_n^2 = \frac{K}{J}$$

$$\zeta = \frac{c}{2\sqrt{JK}}$$



- FB has controlled the Type 1 system
- Type 1 system w FB can hold position

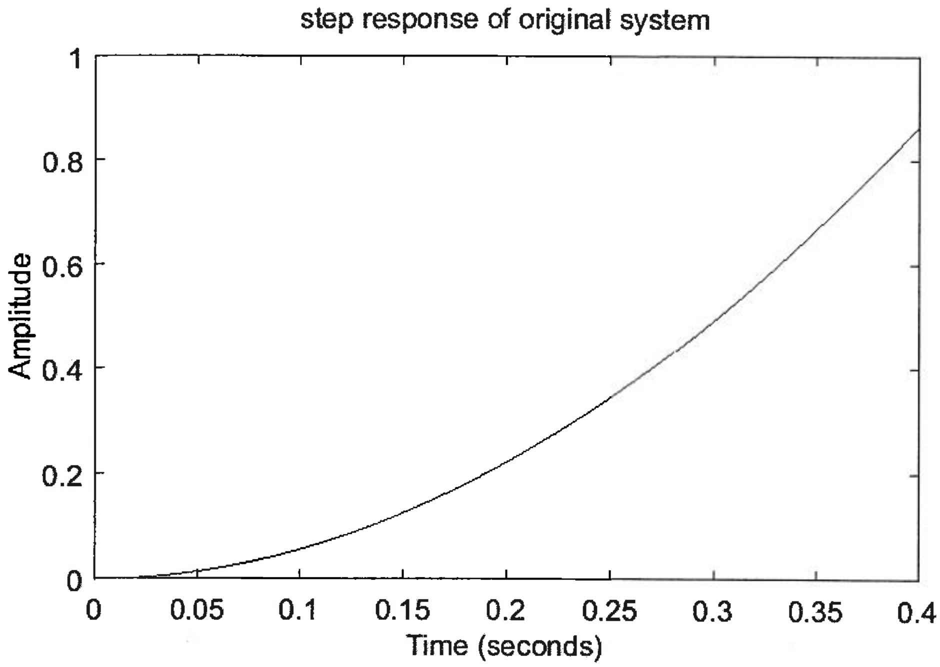
183/523

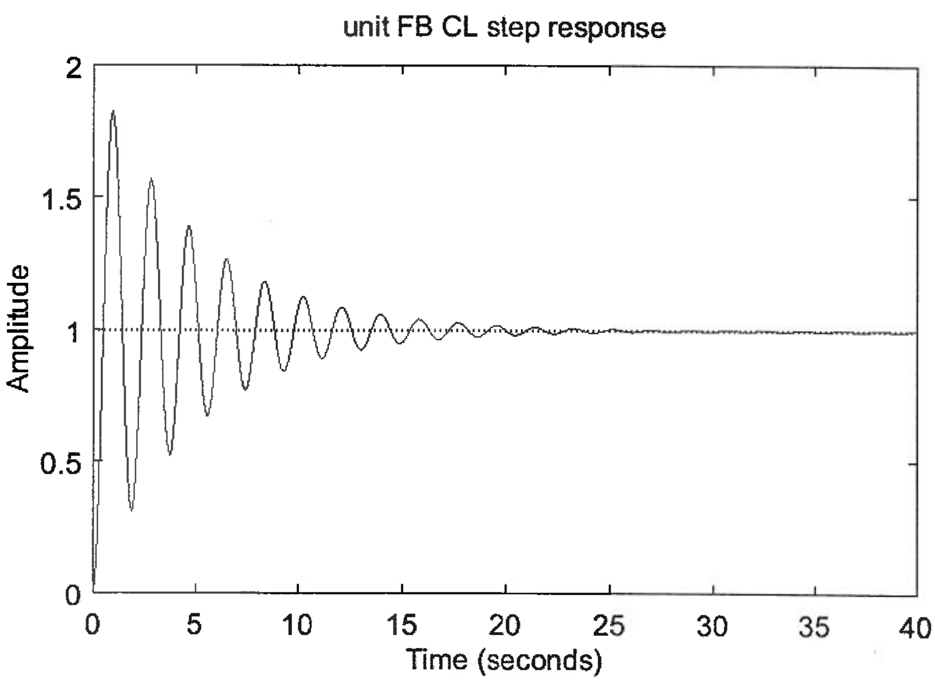
9
FB

```
unit FB example
input data
    K  |  J  |   c =
   114    10      4
======================
calculated results

G =

      114
  ------------
  10 s^2 + 4 s

Continuous-time transfer function.


G_CL =

         114
  ----------------
  10 s^2 + 4 s + 114

Continuous-time transfer function.

poles =
  -0.2000 + 3.3705i
  -0.2000 - 3.3705i

    fn,Hz   |  f,Hz   |  zeta% =
    0.5374     0.5364    5.9235
    0.5374     0.5364    5.9235
```

184/523

10 FB

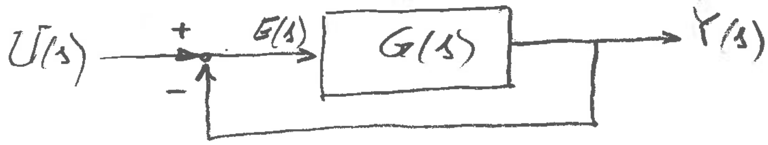

step response of original system
Amplitude
Time (seconds)
0
0.2
0.4
0.6
0.8
1
0.05
0.1
0.15
0.2
0.25
0.3
0.35
0.4

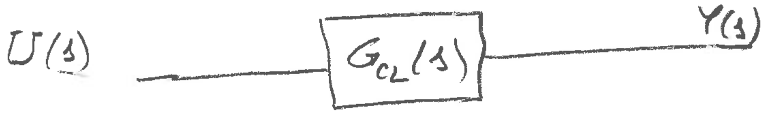

unit FB CL step response
Amplitude
Time (seconds)
0
0.5
1
1.5
2
5
10
15
20
25
30
35
40

185/523

FBE 20161202

Steady state error of feedback systems





Objective: calculate steady state error e_{ss} of CL system without calculating $G_{CL}(s)$

Method: Use FVT or $E(s)$

Details: $E(s) = U(s) - Y(s)$ (1)

$Y(s) = G(s)E(s)$ (2)

(2)→(1) $E = U - GE \longrightarrow E(s) = \dfrac{1}{1+G(s)}U(s)$ (3)

FVT: $\lim\limits_{t\to\infty} e(t) = \lim\limits_{s\to 0} sE(s)$

$$e_{ss} = \lim_{s\to 0} \frac{s}{1+G(s)}U(s)$$ steady state error of FB system (4)

SS error depends on:

- input function $\begin{cases} \text{step} & U(s) = \frac{1}{s} \\ \text{ramp} & U(s) = \frac{1}{s^2} \end{cases}$

- system type

$$G(s) = \frac{K}{s^N}\frac{(T_a s+1)(\ldots)\ldots}{(T_1 s+1)(\ldots)\ldots} \begin{cases} \text{Type } 0, & N=0 \\ \text{Type } 1, & N=1 \\ \text{Type } 2, & N=2 \end{cases}$$

186/523

2
FBE

Step error of FB systems

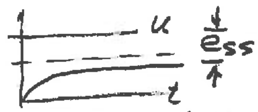
Figure of merit: "static position error constant" $= \lim_{s\to 0} G(s)$

misnomer: in fact, the larger, the better!

• **Derivation** : $u^{step}(s) = \frac{1}{s}$

$$e_{ss}^{step} = \lim_{s\to 0} \frac{\cancel{s}}{1+G(s)}\frac{1}{\cancel{s}} = \lim_{s\to 0} \frac{1}{1+G(s)} = \frac{1}{1+\lim_{s\to 0} G(s)}$$

$\boxed{\text{Type 0 system}}$ $(N=0)$: $G(s) = K\frac{(T_a s+1)(\cdots}{(T_1 s+1)(\cdots} \xrightarrow[s\to 0]{} K = \text{const}$

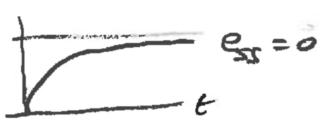


$$e_{ss}^{step} = \lim_{s\to 0} \frac{1}{1+G(s)} \xrightarrow[s\to 0]{} \frac{1}{1+K} \neq 0$$

Feedback creates non zero ss error for Type 0 sys!

$\boxed{\text{Type 1 system}}$ $(N=1)$: $G(s) = \frac{K}{s}\frac{(T_a s+1)\cdots}{(T_1 s+1)\cdots} \xrightarrow[s\to 0]{} \frac{1}{0}$

$$e_{ss}^{step} = \lim_{s\to 0} \frac{1}{1+G(s)} = \frac{1}{1+\frac{1}{0}} = \frac{0}{0+1} = 0$$



$\boxed{\text{Type 2 system}}$ $(N=2)$: $G(s) = \frac{K}{s^2}\frac{(T_a s+1)\cdots}{(T_1 s+1)\cdots} \xrightarrow[s\to 0]{} \frac{1}{0}$

$$e_{ss}^{step} = \lim_{s\to \infty} \frac{1}{1+G(s)} = \cdots = 0$$

$$e_{ss}^{step} = 0 \quad \text{for } N \geq 1$$

187/523

3
FBE
20161202

Ramp error of FB systems

Figure of merit: "static velocity error constant" $= \lim_{s\to 0} s\,G(s)$

misnomer: in fact, the larger, the better!

Derivation: $U^{ramp}(s) = \dfrac{1}{s^2}$

$$e_{ss}^{ramp} = \lim_{s\to 0} \frac{s}{1+G(s)}\frac{1}{s^2} = \lim_{s\to 0}\frac{1}{s+sG(s)} = \frac{1}{\lim_{s\to 0} sG(s)}$$

$\boxed{\text{Type 0 sys}}$ $\quad sG(s) = sK\dfrac{(T_a s+1)\cdots}{(T_1 s+1)\cdots} \xrightarrow[s\to 0]{} 0$

$$e_{ss}^{ramp} = \frac{1}{\lim_{s\to 0} sG(s)} = \frac{1}{0} = \infty$$

infinite error!
cannot follow!

$\boxed{\text{Type 1 sys}}$ $\quad sG(s) = \cancel{s}\dfrac{K}{\cancel{s}}\dfrac{(T_a s+1)\cdots}{(T_1 s+1)\cdots} \xrightarrow[s\to 0]{} K = \text{const}$

$$e_{ss}^{ramp} = \frac{1}{\lim_{s\to 0} sG(s)} = \frac{1}{K}$$

ramp offset.

$\boxed{\text{Type 2 sys}}$ $\quad sG(s) = \cancel{s}\dfrac{K}{s^2}\dfrac{(T_a s+1)\cdots}{(T_1 s+1)\cdots} \xrightarrow[s\to 0]{} \dfrac{1}{0}$

$$e_{ss}^{ramp} = \frac{1}{\lim_{s\to 0} sG(s)} = 0$$

$e_{ss}^{ramp} = 0$ for $N \geq 2$

4 FBE 20161202

System		Steady State errors	
Type	Expression	Step error e_{ss}^{step}	Ramp error e_{ss}^{ramp}
0	$K\frac{(T_a s+1)(\dots)\dots}{(T_1 s+1)(\dots)\dots}$	$\frac{1}{1+K}$	∞
1	$\frac{K}{s}\frac{(T_a s+1)(\dots)\dots}{(T_1 s+1)(\dots)\dots}$	0	$\frac{1}{K}$
2	$\frac{K}{s^2}\frac{(T_a s+1)(\dots)\dots}{(T_1 s+1)(\dots)\dots}$	0	0
$\vdots$ $N>2$	$\frac{K}{s^N}\frac{(T_a s+1)(\dots)\dots}{(T_1 s+1)(\dots)\dots}$	0	0

189/523

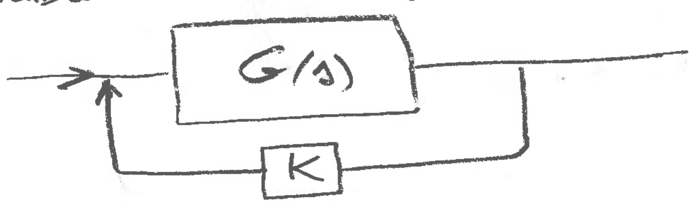
7.4 Stability of feedback control systems and root locus

1
SFB

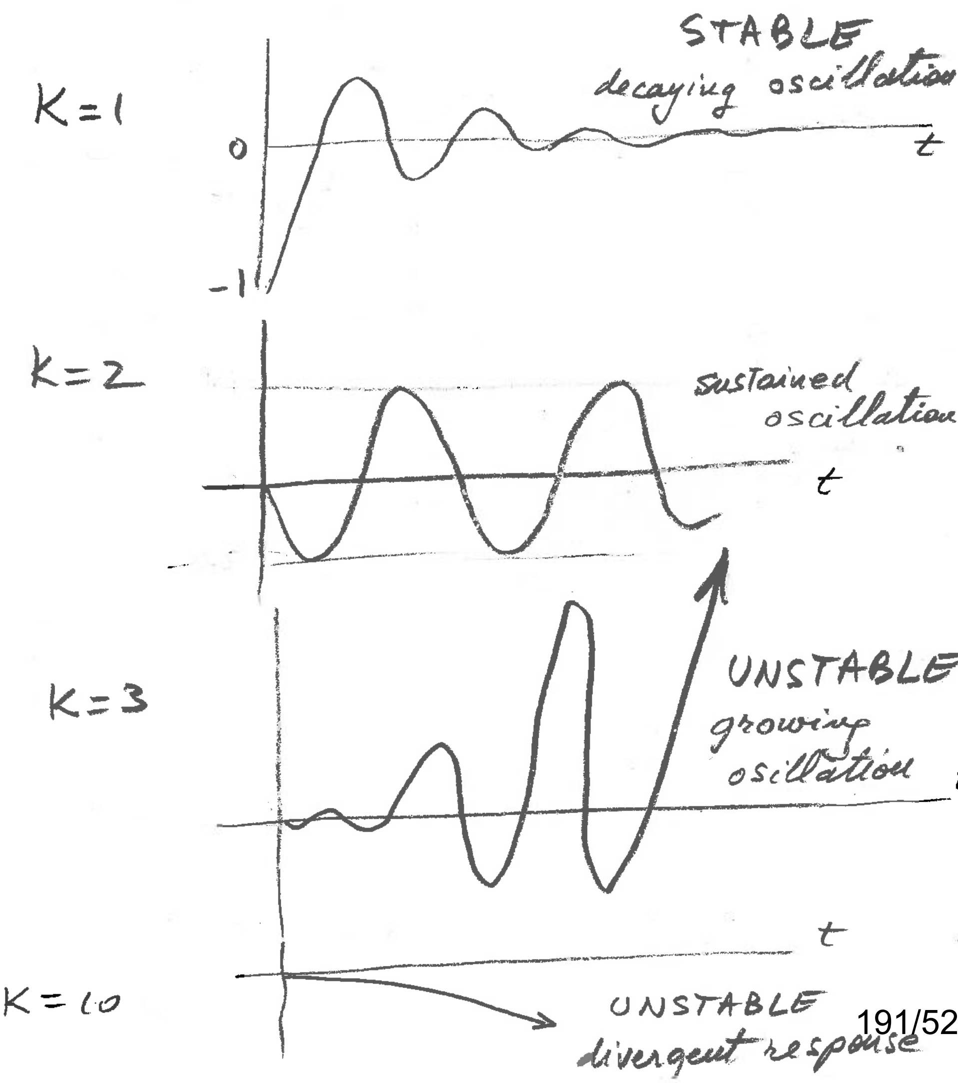
STABILITY OF FEEDBACK SYSTEMS

EXAMPLE

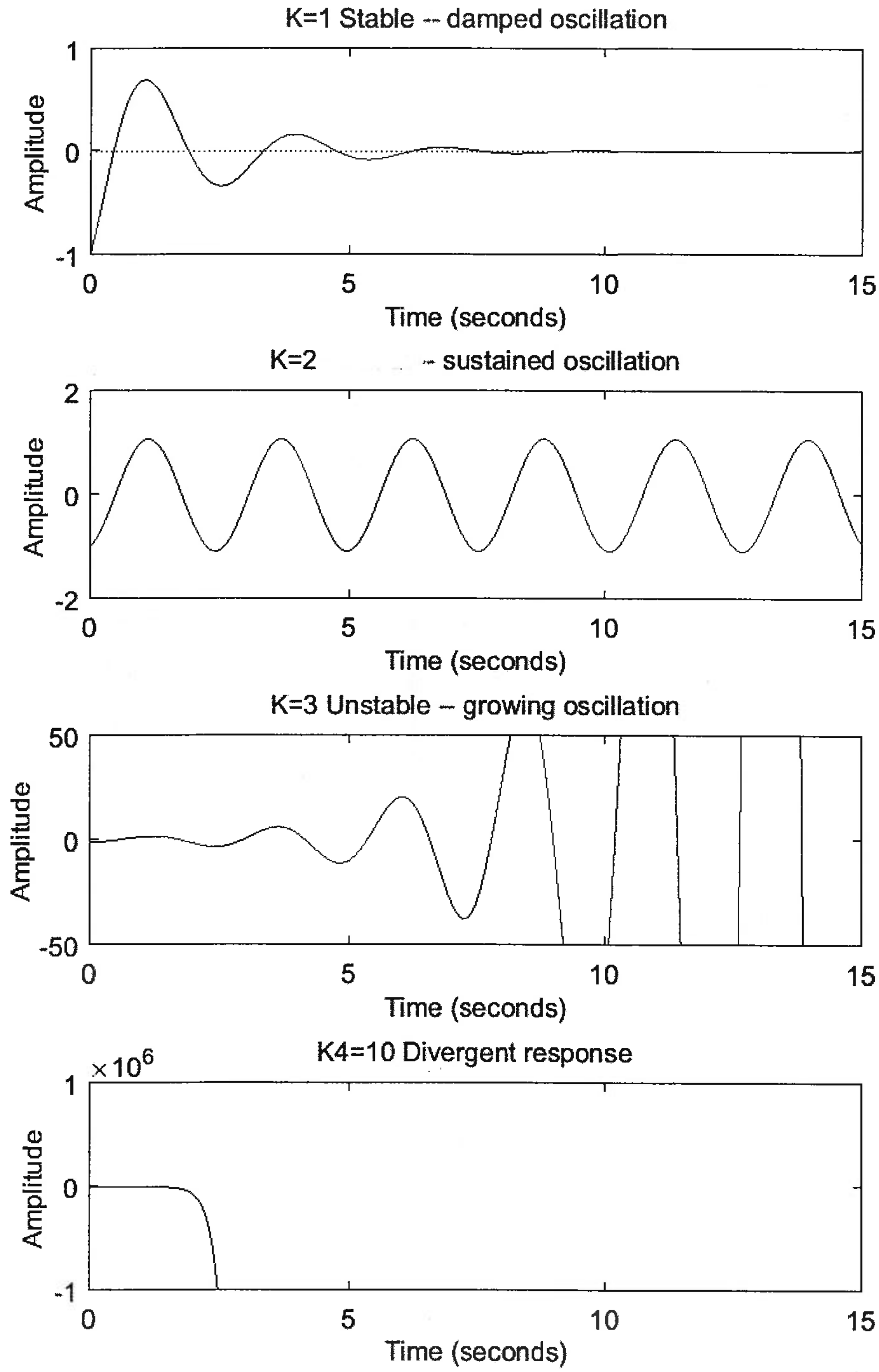
Run MATLAB code FB_stability_ex1



$$G(s) = \frac{1-s}{s^2+2s+4}$$



2
SFB


K=1 Stable -- damped oscillation
Amplitude
1
0
-1
0
5
10
15
Time (seconds)
K=2 -- sustained oscillation
Amplitude
2
0
-2
0
5
10
15
Time (seconds)
K=3 Unstable -- growing oscillation
Amplitude
50
0
-50
0
5
10
15
Time (seconds)
K4=10 Divergent response
$\times 10^6$
Amplitude
1
0
-1
0
5
10
15
Time (seconds)

192/523

3 SFB

Explanation

$$G_{CL} = \frac{G}{1+GK} = \frac{\dfrac{1-s}{s^2+2s+4}}{1+K\dfrac{1-s}{s^2+2s+4}}$$

$$= \frac{\dfrac{1-s}{s^2+2s+4}}{s^2+2s+4+K(1-s)} = \frac{1-s}{s^2+(2-K)s+(K+4)}$$

Impulse response: $X(s) = G_{CL}(s) = \dfrac{1-s}{s^2+(2-K)s+(K+4)}$

Response type depends on characteristic eq.

$$s^2+(2-K)s+(K+4) = 0$$ characteristic eqn.

$K=1$ $\quad s^2+s+5=0$

$$p_{1,2} = \frac{-1\pm\sqrt{1-4\times 5}}{2} = -\frac{1}{2} \pm i\frac{\sqrt{19}}{2}$$

LHS
decaying osc.
STABLE

$K=2$ $\quad s^2+6=0$

$$p_{1,2} = \pm i\sqrt{6}$$

imag. axis
sustained oscillation

$K=3$ $\quad s^2-s+7=0$

$$p_{1,2} = \frac{1\pm\sqrt{1-28}}{2} = \frac{1}{2} \pm i\frac{\sqrt{27}}{2}$$

RHS
growing osc.
UNSTABLE

$K=10$ $\quad s^2-8s+14=0$

$$p_{1,2} = 4\pm\sqrt{4^2-14} = 4\pm\sqrt{2} \begin{cases} 5.41 \\ 2.59 \end{cases}$$

RHS
UNSTABLE
non-oscillatory

4

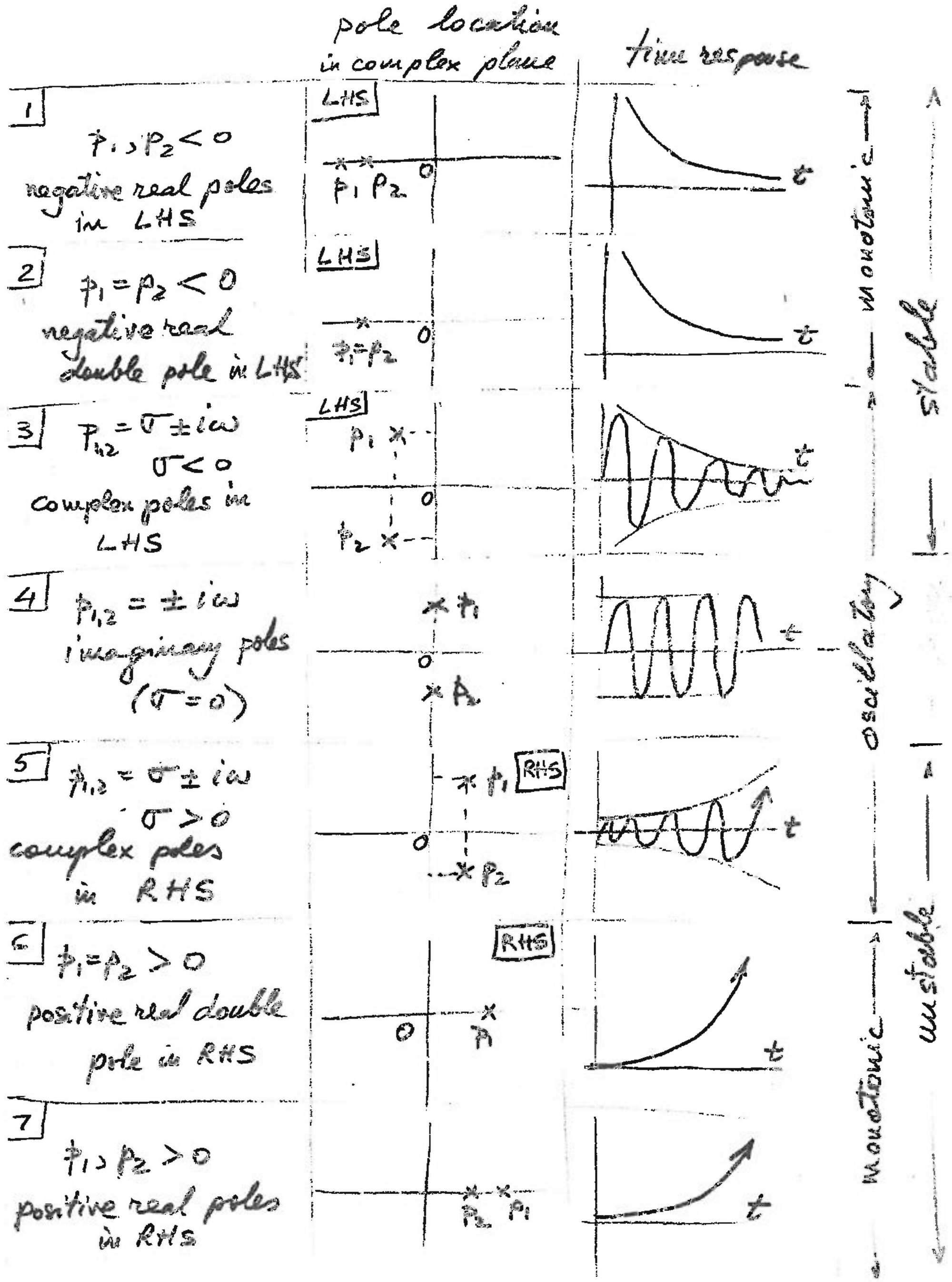
```
K1 =
     1

G1CL =

    -s + 1
  -----------
  s^2 + s + 5

Continuous-time transfer function.

p1 =
  -0.5000 + 2.1794i
  -0.5000 - 2.1794i
wn1, rad/sec | zeta1 =
    2.2361    0.2236
    2.2361    0.2236
=========================
K2 =
     2

G2CL =

  -s + 1
  -------
  s^2 + 6

Continuous-time transfer function.

p2 =
   0.0000 + 2.4495i
   0.0000 - 2.4495i
wn2, rad/sec | zeta2 =
    2.4495         0
    2.4495         0
```

194/523

```
K3 =
     3

G3CL =

     -s + 1
   -----------
   s^2 - s + 7

Continuous-time transfer function.

p3 =
   0.5000 + 2.5981i
   0.5000 - 2.5981i
wn3, rad/sec | zeta3 =
    2.6458   -0.1890
    2.6458   -0.1890
=========================
K4 =
    10

G4CL =

      -s + 1
   --------------
   s^2 - 8 s + 14

Continuous-time transfer function.

p4 =
    5.4142
    2.5858
wn4, rad/sec | zeta4 =
    5.4142   -1.0000
    2.5858   -1.0000
```

6
SFb

Recall stability analysis as function of pole location:

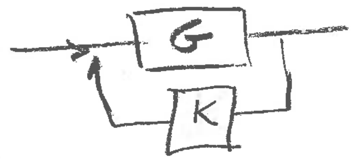


196/523

20140305

Root Locus

Given $G(s)$



Trace the poles of $G_{CL}(s)$ as K increases

$$G(s) = \frac{B(s)}{A(s)} = \frac{num(s)}{den(s)}$$

$$G_{CL} = \frac{G}{1+KG} = \frac{B(s)}{A(s)+KB(s)}$$

The root locus method looks at the root of the denominator of the CL system G_{CL}

$$A(s) + KB(s) = 0$$

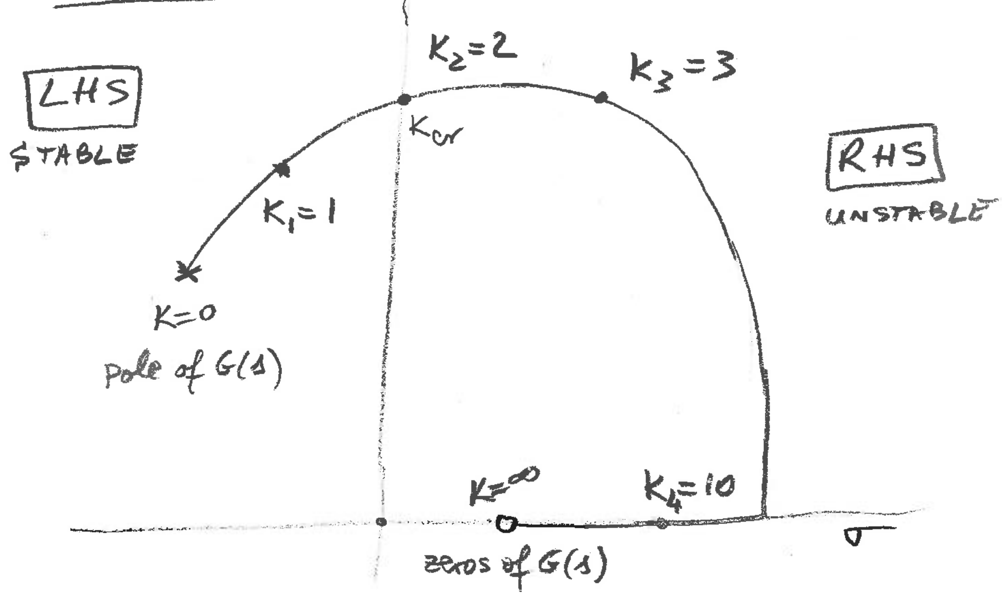
The roots are traced in the s-plane for $K = [0, \infty)$

For $K=0$, the poles of G_{CL} are the same as the poles of G.

197/523

1/5FB

Root locus



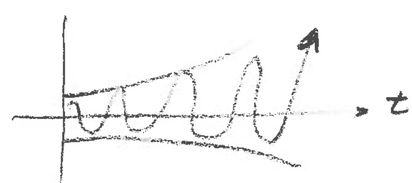
$$p_{1,2} = \sigma \pm i\omega_d = -\zeta\omega_n \pm i\,\omega_d$$

$$x(t) = C\,e^{-\zeta\omega_n t}\sin(\omega_d t + \varphi)$$

$K_1 : \ \sigma < 0 \ \longrightarrow \ \zeta > 0$

$K_2 : \ \sigma = 0 \ \longrightarrow \ \zeta = 0$
K_{cr}

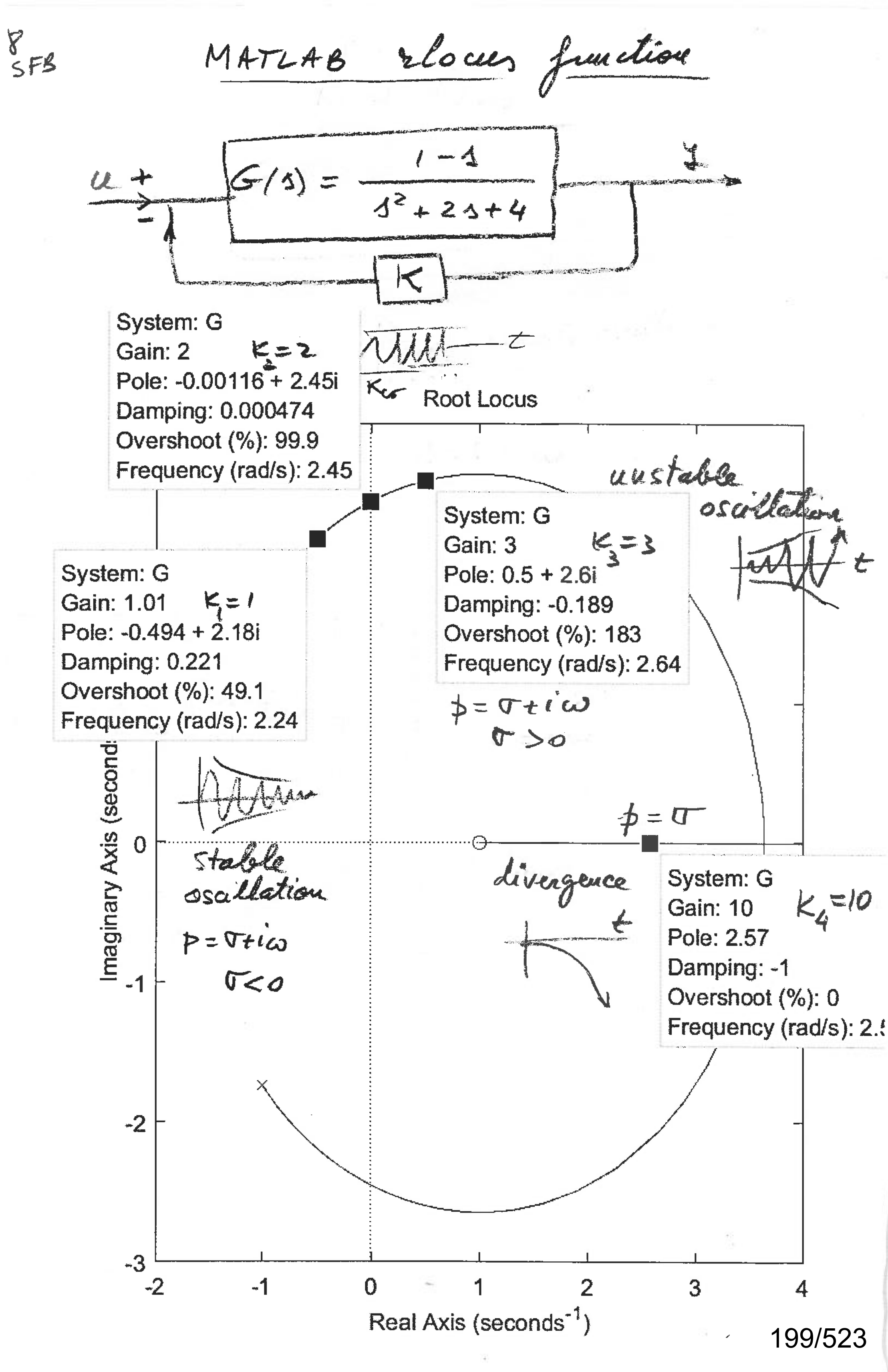
$K_3 : \ \sigma > 0 \ \longrightarrow \ \zeta < 0$



$K_4 : \ \omega = 0$

$p_1 = \sigma_1 \ , \ p_2 = \sigma_2 > 0$

$x(t) = C_1 e^{p_1 t} + C_2 e^{p_2 t}$


8
SFB
MATLAB rlocus function
u +
-
G(s) = (1 - s)/(s² + 2s + 4)
y
K
System: G
Gain: 2 K₂=2
Pole: -0.00116 + 2.45i
Damping: 0.000474
Overshoot (%): 99.9
Frequency (rad/s): 2.45
t
Kcr
Root Locus
unstable
oscillation
t
System: G
Gain: 3 K₃=3
Pole: 0.5 + 2.6i
Damping: -0.189
Overshoot (%): 183
Frequency (rad/s): 2.64
System: G
Gain: 1.01 K₁=1
Pole: -0.494 + 2.18i
Damping: 0.221
Overshoot (%): 49.1
Frequency (rad/s): 2.24
p = σ + iω
σ > 0
p = σ
stable
oscillation
divergence
t
p = σ + iω
σ < 0
System: G
Gain: 10 K₄=10
Pole: 2.57
Damping: -1
Overshoot (%): 0
Imaginary Axis (seconds
Real Axis (seconds⁻¹)
199/523

7
sFB

rlocus

Evans root locus

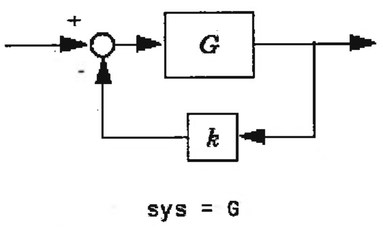
Syntax

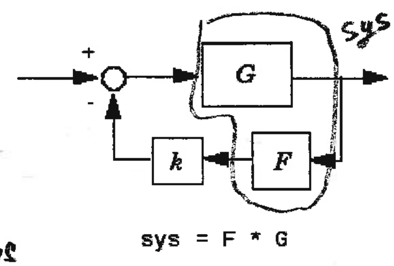
```
rlocus
rlocus(sys)
rlocus(sys1,sys2,...)
```

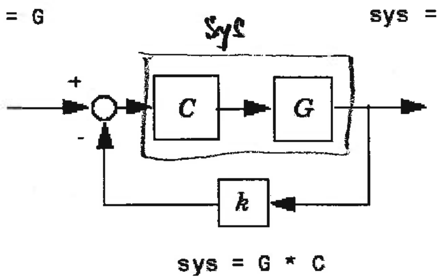
Description

`rlocus` computes the Evans root locus of a SISO open-loop model. The root locus gives the closed-loop pole trajectories as a function of the feedback gain k (assuming negative feedback). Root loci are used to study the effects of varying feedback gains on closed-loop pole locations. In turn, these locations provide indirect information on the time and frequency responses.

`rlocus(sys)` calculates and plots the root locus of the open-loop SISO model `sys`. This function can be applied to any of the following *negative* feedback loops by setting `sys` appropriately.







If `sys` has transfer function

$$h(s) = \frac{n(s)}{d(s)}$$

the closed-loop poles are the roots of

$$d(s) + k\ n(s) = 0$$

`rlocus` adaptively selects a set of positive gains k to produce a smooth plot. Alternatively,

177

200/523

10
, FB

```
rlocus(sys,k)
```

uses the user-specified vector k of gains to plot the root locus.

`rlocus(sys1,sys2,...)` draws the root loci of multiple LTI models `sys1, sys2,...` on a single plot. You can specify a color, line style, and marker for each model, as in

```
rlocus(sys1,'r',sys2,'y:',sys3,'gx').
```

When invoked with output arguments,

```
[r,k] = rlocus(sys)
r = rlocus(sys,k)
```

return the vector `k` of selected gains and the complex root locations `r` for these gains. The matrix `r` has `length(k)` columns and its `j`th column lists the closed-loop roots for the gain `k(j)`.

Remarks

You can change the properties of your plot, for example the units. For information on the ways to change properties of your plots, see Ways to Customize Plots.

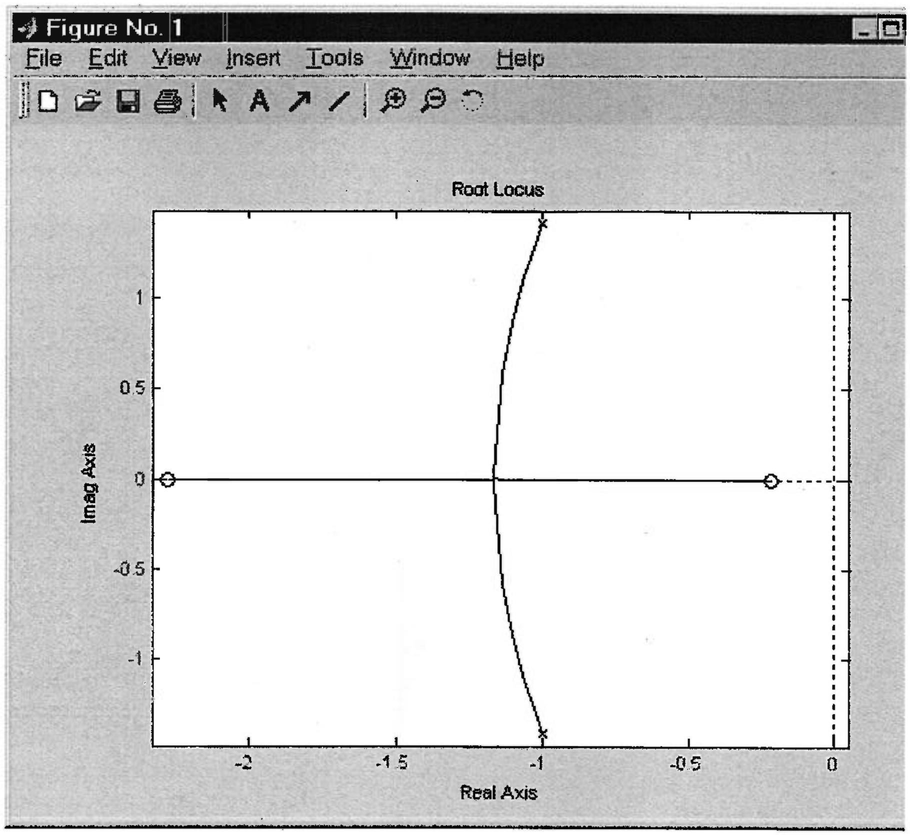
Example

Find and plot the root-locus of the following system.

$$h(s) = \frac{2s^2 + 5s + 1}{s^2 + 2s + 3}$$

```
h = tf([2 5 1],[1 2 3]);
rlocus(h)
```

11
5FB



You can use the right-click menu for rlocus to add grid lines, zoom in or out, and invoke the Property Editor to customize the plot. Also, click anywhere on the curve to activate a data marker that displays the gain value, pole, damping, overshoot, and frequency at the selected point.

See Also

pole, pzmap

Provide feedback about this page

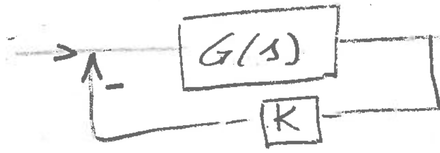
reshape rlocusplot

© 1984-2009 The MathWorks, Inc. • Terms of Use • Patents • Trademarks • Acknowledgments

12 SFB

ROOT LOCUS METHOD

$G(s) = \frac{B(s)}{A(s)} = \frac{num}{den}$

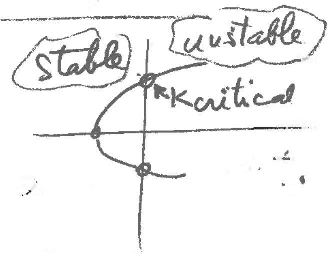


$$G_{CL} = \frac{KG}{1+KG}$$

$$G(s) = \frac{(s+z_1)(s+z_2) + \cdots (s+z_m)}{(s+p_1)(s+p_2) + \cdots (s+p_n)} = \frac{num}{den}$$

$-z_1, \ldots, -z_m$ zeros of open loop transfer function

$-p_1, \ldots, -p_n$ poles —— " ——



charact. eq^n: $1 + KG = 0$.

$A(s) + KB(s) = 0$

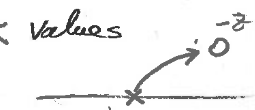
$$(s+p_1)(s+p_2)\cdots(s+p_n) + K(s+z_1)\cdots(s+z_m) = 0.$$

Solution of this equation gives the CL poles $s = \sigma + i\omega$

Root locus: trajectory of these poles as $K = 0 \longrightarrow \infty$

rlocus (num, den) ⟶ automatic K generation $K \in [0, \infty)$

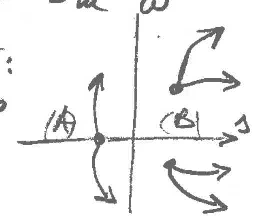
rlocus (num, den, K) ⟶ must give K values



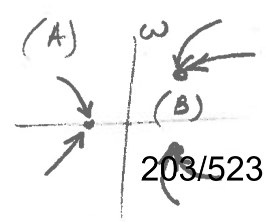
$K=0$: roots are the OL poles: $-p_1, -p_2, \ldots -p_n$

$K=\infty$: roots are the OL zeros: $-z_1, -z_2 \ldots -z_m$

Breakaway: multiple roots branch out:
(A) from real poles on real axis
(B) from conjugate poles



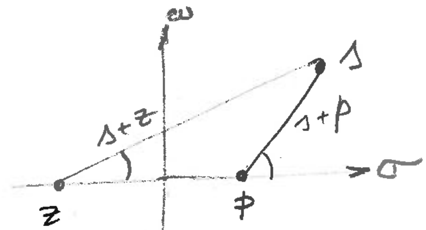
Break in: branches coalesce into multiple roots
(A) into real poles
(B) into conjugate poles



203/523

13
SFB

Angle condition



$$1 + KG = 0$$

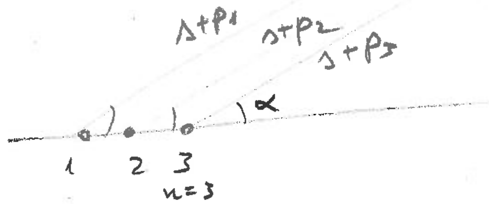
$$KG = -1$$

$$\angle KG = \angle -1 = \pm 180^\circ (2k+1), \; k = 0, 1, \ldots$$

$$\angle KG = \angle K \frac{(s+z_1)(s+z_2)\cdots}{(s+p_1)(s+p_2)\cdots}$$

$$= \angle s+z_1 + \angle s+z_2 + \ldots$$
$$- \angle s+p_1 - \angle s+p_2 - \angle s+p_3 \ldots = \pm 180^\circ(2k+1).$$

Asymptotes $\quad s \to \infty$



- angles of asymptotes

$$\angle \text{asymp} = \frac{\pm 180^\circ (2k+1)}{3}$$

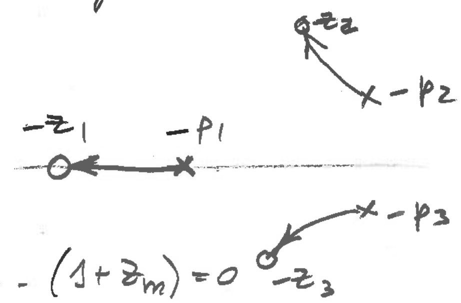
$$= 60^\circ ; 180^\circ, -60^\circ$$

- axis crossing of asymptotes

14 SFB $(s+p_1)(s+p_2)\cdots + K(s+z_1)(s+z_2)\cdots = 0$

$K=0 \longrightarrow (s+p_1)(s+p_2)\cdots(s+p_n)=0$

roots are the OL poles



$K \to \infty$: $K(s+z_1)(s+z_2)\cdots(s+z_m)=0$

roots are the OL zeros

7.5 Stability Criteria

RH 20140306

ROUTH CRITERION
FOR HIGHER ORDER POLYNOMIALS

Given: $A(s) = a_n s^n + a_{n-1} s^{n-1} + \dots + a_0$

Find if any roots of $A(s)$ are in the RHS

Solution by Routh criterion

(1) If any of the coefficients $a_0, a_1, \dots, a_n$ is negative, then at least one root is in RHS and the system is UNSTABLE. $\boxed{\text{STOP}}$

(2) If all coefficients are positive, do the Routh table:

207/523

2
RH

R Table

$$A(s) = a_n s^n + a_{n-1} s^{n-1} + a_{n-2} s^{n-2} + \dots \; a_1 s + a_0 = 0$$

$$\begin{array}{c|ccc}
s^n & a_n & a_{n-2} & a_{n-4} \; \dots \\
s^{n-1} & a_{n-1} & a_{n-3} & a_{n-5} \; \dots \\
s^{n-2} & b_1 & b_2 & \\
s^{n-3} & c_1 & c_2 & \\
\vdots & & &
\end{array}$$

$$b_1 = \frac{a_{n-1} a_{n-2} - a_n a_{n-3}}{a_{n-1}}$$

$$b_2 = \frac{a_{n-1} a_{n-4} - a_n a_{n-5}}{a_{n-1}}$$

$$c_1 = \frac{b_1 a_{n-3} - a_{n-1} b_2}{b_1}$$

$$\vdots$$

Count the number of sign changes and zeros in the first column to get the number of roots in RHS

Use MATLAB file:
Routh_Hurwitz_Stability_criterion_modified_vg1.m

3 RH

R. criterion Example VG1

$$G(s) = \frac{4s+2}{s^3+3s^2+4s+2} = \frac{B(s)}{A(s)}$$

Examine $A(s) = s^3 + 3s^2 + 4s + 2$

$a_3 = 1 \quad a_1 = 4$

$a_2 = 3 \quad a_0 = 2$

$$\begin{array}{c|cc} s^3 & 1 & 4 \\ s^2 & 3 & 2 \\ s^1 & 10/3 & \\ s^0 & 2 & \end{array}$$

$$b_1 = \frac{3\times4 - 1\times2}{3} = 10/3$$

$$b_2 = \frac{0-0}{3} = 0$$

$$c_1 = \frac{\frac{10}{3}\times2 - 3\times0}{10/3} = 2$$

Routh criterion: number of sign changes = zero (0)

i.e. NO root in RHS $\longrightarrow$ STABLE!

Verification by MATLAB

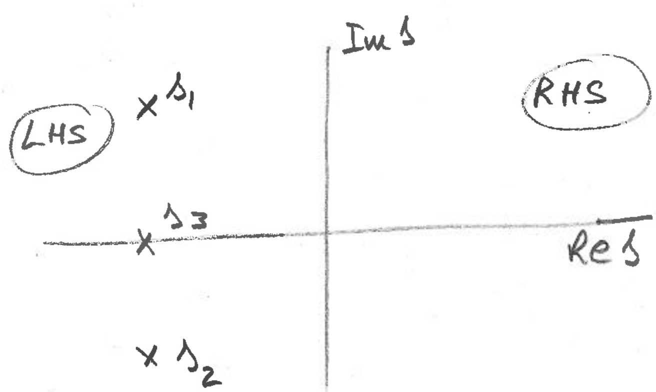
A = [1 3 4 2]

roots(A):

$s_1 = -1 + i$

$s_2 = -1 - i$

$s_3 = -1$



All roots are in LHS $\longrightarrow$ System is STABLE!

209/523

4 RH 20140303

R Criterion

Example VG2

$$G(s) = \frac{31}{s^3+5s^2+6s+31}$$

$a_3=1$ $a_2=5$ $a_1=6$ $a_0=31$.

s^3	1	6
s^2	5	31
s^1	-0.2	
s^0	31	

$$b_1 = \frac{5\times 6 - 1\times 31}{5} = -0.2$$

$$c_1 = \frac{-0.2\times 31}{-0.2} = 31$$

→ 2 sign changes.

2 roots in RHS → Unstable!

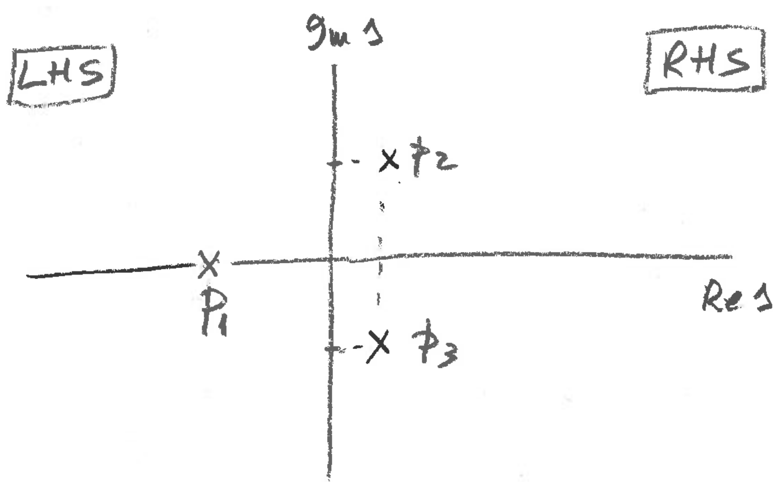
See Matlab print out on next page.

210/523

5
RH
20140303

MATLAB Command Window

```
Input coefficients of characteristic equation
[an an-1 an-2 ... a0]= [1 5 6 31]

----------------------------------------
Roots of characteristic equation is:

ans =

p1   -5.0319
p2    0.0160 + 2.4820i
p3    0.0160 - 2.4820i

--------The R array is:--------

m =

    1.0000    6.0000
    5.0000   31.0000
   -0.2000         0
   31.0000         0

         ----> System is Unstable <----
>>
```



211/523

5a
RH

R criterion Example VG3

$$G(s) = \frac{5s^2 + 8s + 3}{s^6 + 3s^5 - s^4 - 7s^3 + 10s^2 + 14s - 20}$$

A(s) has some negative coefficients

At least one root of A(s) is in RHS

system is unstable

212/523

6
RH

R. criterion Example VG4

$$G(s) = \frac{5s^2 + 8s + 3}{s^6 + 3s^5 + s^4 + 7s^3 + 10s^2 + 14s + 20}$$

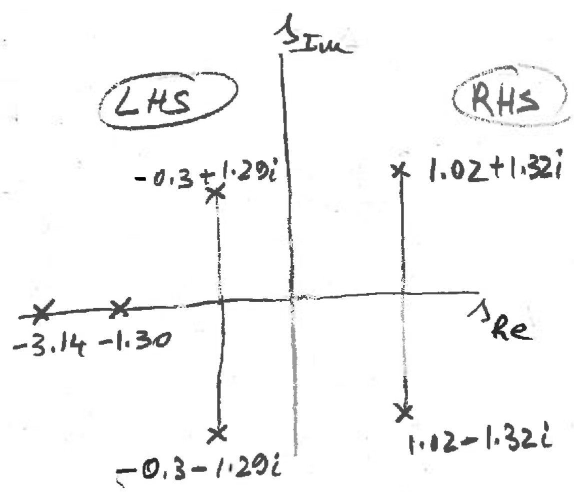
$a_6 = 1$ $a_4 = 1$ $a_2 = 10$ $a_0 = 20$
$a_5 = 3$ $a_3 = 7$ $a_1 = 14$

s^6	1	1	10	20
s^5	3	7	14	
① s^4	−1.33	5.33	20	
② s^3	19	59		
s^2	9.4737	20		
s^1	18.8889			
s^0	20			

Two (2) sign changes
2 roots in RHS
UNSTABLE!

MATLAB
A=[1 3 1 7 10 14 20]
roots(A):

−3.14 + 0i	LHS
−1.30 + 0i	
−0.30 + 1.29i	
−0.30 − 1.29i	
1.02 + 1.32i	RHS
1.02 − 1.32i	



See book 5. for more about Routh criterion

213/523

7
RH

Input coefficients of characteristic equation
[an an-1 an-2 ... a0]= [1 3 1 7 10 14 20]

Roots of characteristic equation are:

ans =

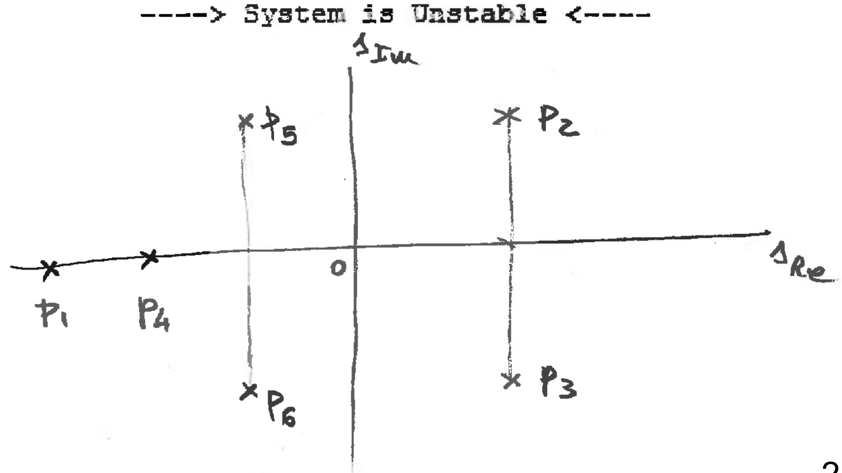
p1	-3.1462 + 0.0000i	LHS
p2	1.0202 + 1.3244i	RHS
p3	1.0202 - 1.3244i	RHS
p4	-1.2962 + 0.0000i	LHS
p5	-0.2990 + 1.2905i	LHS
p6	-0.2990 - 1.2905i	LHS

--------The Routh array is:--------

m =

1.0000	1.0000	10.0000	20.0000
3.0000	7.0000	14.0000	0
-1.3333	5.3333	20.0000	0
19.0000	59.0000	0	0
9.4737	20.0000	0	0
18.8889	0	0	0
20.0000	0	0	0

----> System is Unstable <----



8
RH

(page 116)

5.4 HURWITZ STABILITY CRITERION

The Hurwitz criterion is another method for determining whether all the roots of the characteristic equation of a continuous system have negative real parts. This criterion is applied using determinants formed from the coefficients of the characteristic equation. It is assumed that the first coefficient, a_n, is positive. The determinants Δ_i, $i = 1, 2, \ldots, n-1$, are formed as the principal minor determinants of the determinant

$$\Delta_n = \begin{vmatrix} a_{n-1} & a_{n-3} & \cdots & \begin{bmatrix} a_0 & \text{if } n \text{ odd} \\ a_1 & \text{if } n \text{ even} \end{bmatrix} & 0 & \cdots & 0 \\ a_n & a_{n-2} & \cdots & \begin{bmatrix} a_1 & \text{if } n \text{ odd} \\ a_0 & \text{if } n \text{ even} \end{bmatrix} & 0 & \cdots & 0 \\ 0 & a_{n-1} & a_{n-3} & \cdots & \cdots & \cdots & 0 \\ 0 & a_n & a_{n-2} & \cdots & \cdots & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & \cdots & \cdots & \cdots & \cdots & a_0 \end{vmatrix}$$

The determinants are thus formed as follows:

$$\Delta_1 = a_{n-1}$$

$$\Delta_2 = \begin{vmatrix} a_{n-1} & a_{n-3} \\ a_n & a_{n-2} \end{vmatrix} = a_{n-1}a_{n-2} - a_n a_{n-3}$$

$$\Delta_3 = \begin{vmatrix} a_{n-1} & a_{n-3} & a_{n-5} \\ a_n & a_{n-2} & a_{n-4} \\ 0 & a_{n-1} & a_{n-3} \end{vmatrix} = a_{n-1}a_{n-2}a_{n-3} + a_n a_{n-1} a_{n-5} - a_n a_{n-3}^2 - a_{n-4}a_{n-1}^2$$

and so on up to Δ_{n-1}.

Hurwitz Criterion: *All the roots of the characteristic equation have negative real parts if and only if* $\Delta_i > 0$, $i = 1, 2, \ldots, n$.

EXAMPLE 5.5. For $n = 3$,

$$\Delta_3 = \begin{vmatrix} a_2 & a_0 & 0 \\ a_3 & a_1 & 0 \\ 0 & a_2 & a_0 \end{vmatrix} = a_2 a_1 a_0 - a_0^2 a_3, \qquad \Delta_2 = \begin{vmatrix} a_2 & a_0 \\ a_3 & a_1 \end{vmatrix} = a_2 a_1 - a_0 a_3, \qquad \Delta_1 = a_2$$

Thus all the roots of the characteristic equation have negative real parts if

$$a_2 > 0 \qquad a_2 a_1 - a_0 a_3 > 0 \qquad a_2 a_1 a_0 - a_0^2 a_3 > 0$$

215/523

9

RH
2016 12 01

Q : What is R criterion ?

A : R criterion is a tabular method to determine if a polynomial

$A(s) = a_n s^n + a_{n-1} s^{n-1} + \dots + a_1 s + a_0$

has roots in the RHS

Q : What is R criterion good for ?

A : R criterion is used to evaluate the stability of a system $G(s) = B(s)/A(s)$

If $A(s)$ has roots in RHS, then the system is UNSTABLE

Q : How do I use R criterion ?

A : - If $A(s)$ has at least one -ve coefficient, then the system is UNSTABLE. $\boxed{\text{STOP}}$

- If all coefficients of $A(s)$ are +ve, then do R Table to find if any roots are in RHS

216/523

10
RH

Q: What is H criterion?

A: H criterion is another method of using the coefficients of a polynomial equation to determine whether all the roots have negative real parts

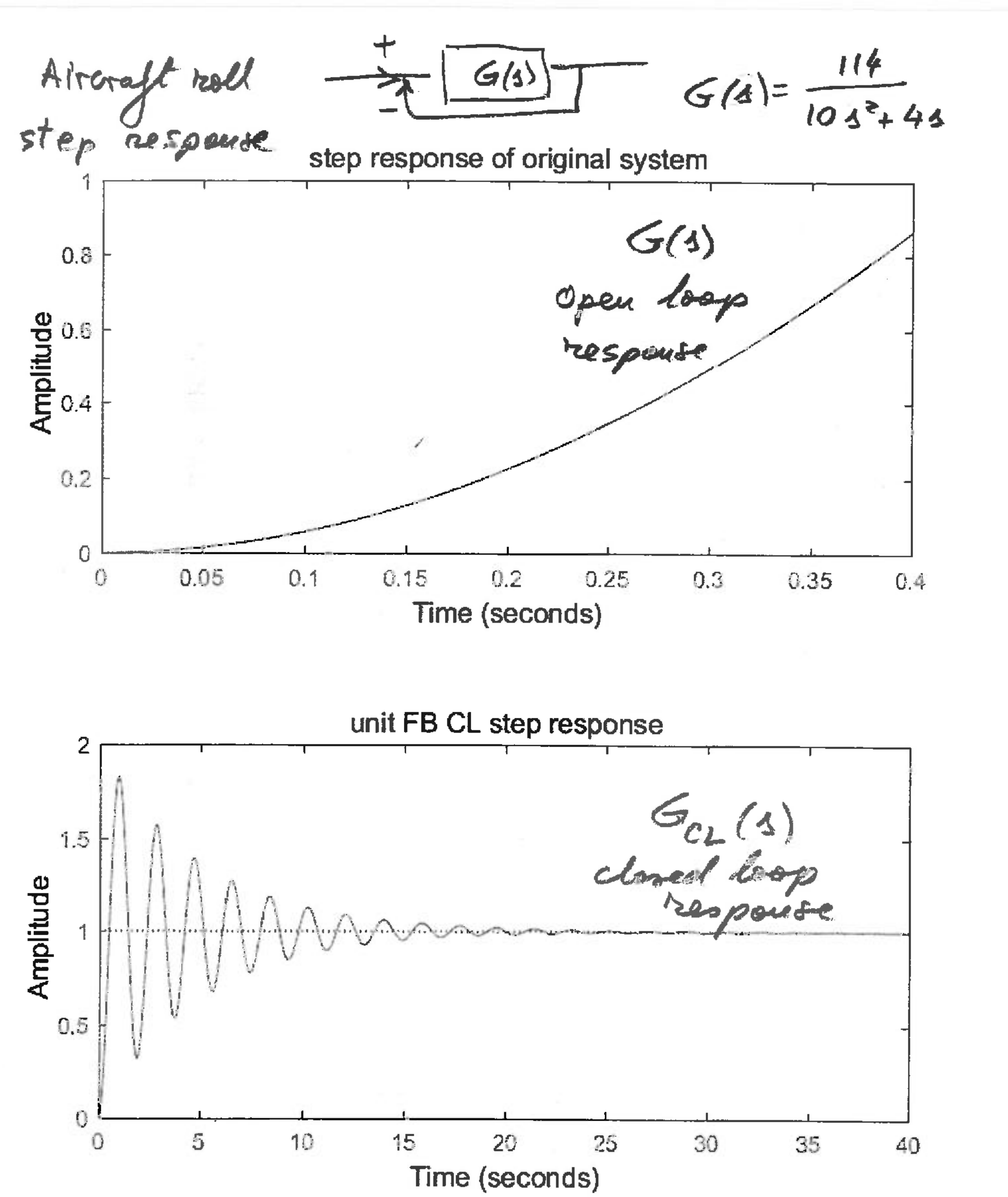
---

Q: How is H criterion different from R criterion?

A: H criterion uses determinants, whereas R criterion uses a table

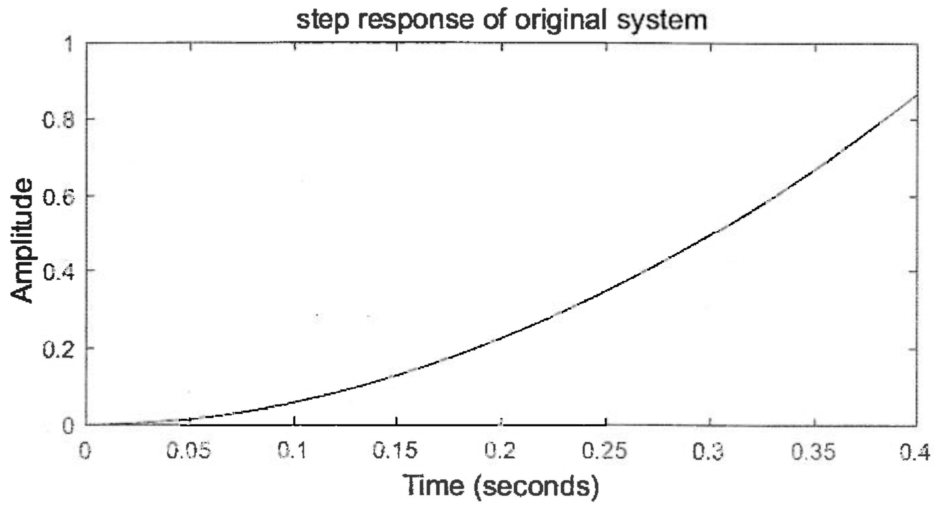
217/523

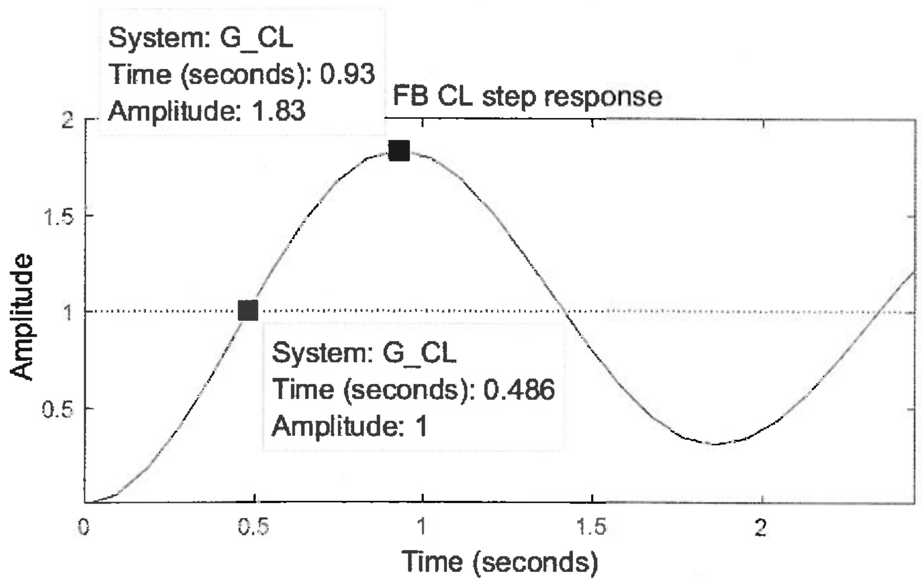
7.6 Feedback Controllers


Aircraft roll step response
+
−
G(s)
$G(s) = \frac{114}{10s^2 + 4s}$
step response of original system
G(s) Open loop response
Amplitude
Time (seconds)
unit FB CL step response
$G_{CL}(s)$ closed loop response
Amplitude
Time (seconds)

219/523

Ob c





<u>Measured</u>: $t_r = 0.486$ sec

$M_p = 83\%$

<u>Design Specs</u>

DS1: $t_r \leq 1.5$ sec

DS2: $M_p \leq 20\%$

DS1 : satisfied

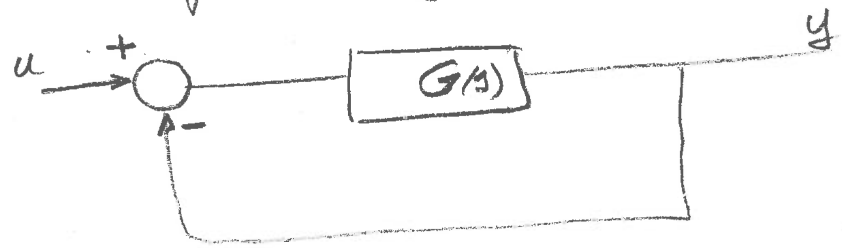
DS2 : NOT satisfied

220/523

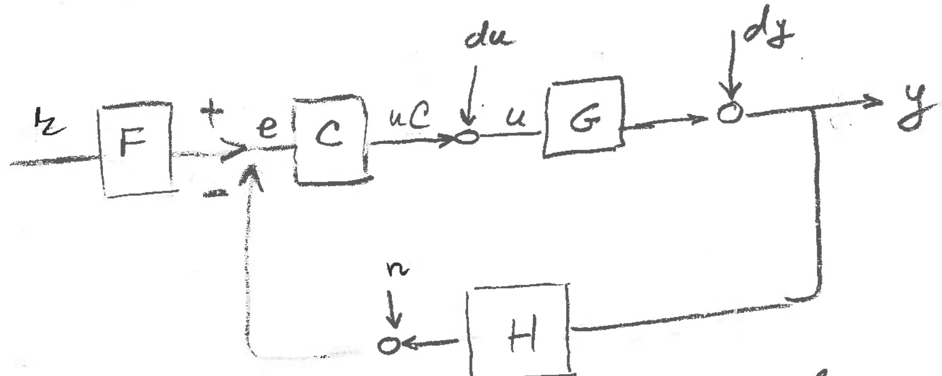
C
20140317

CONTROLLERS

Basic feedback system:



Enhanced FB control systems



Addition "boxes" with specific transfer function may be added to improve performance

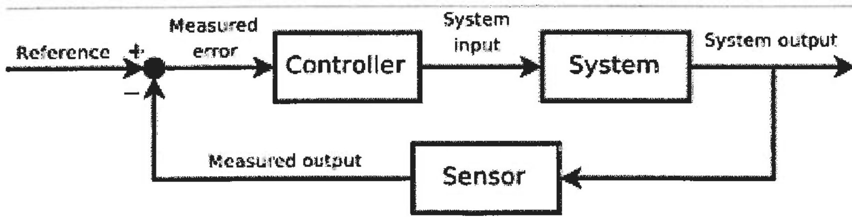
Examples:

- PID controllers
- Filters
- Compensators
- Pre-filters
- Post-filters

221/523

2
C

CONTROLLERS



Controllers modulate the feedback error to improve the performance of the feedback control system.

222/523

4c

Filters

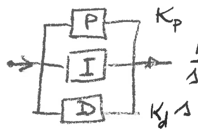
$$G_c(s) = \frac{1}{Ts+1} \quad \text{(low pass filter)}$$

$$G_c(s) = \frac{Ts}{Ts+1} \quad \text{(high pass filter)}$$

Compensators

$$G_c(s) = \frac{T_2 s + 1}{T_1 s + 1}$$

PID controllers



$$\boxed{G_c = K_p + \frac{K_i}{s} + K_d s}$$

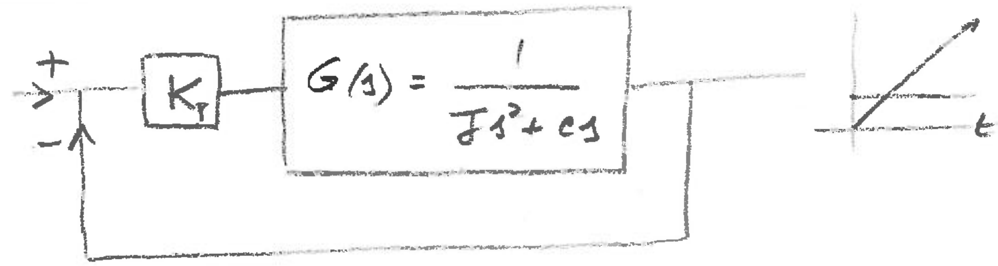
$$\boxed{G_c = K\left(1 + \frac{1}{T_i s} + T_d s\right)}$$

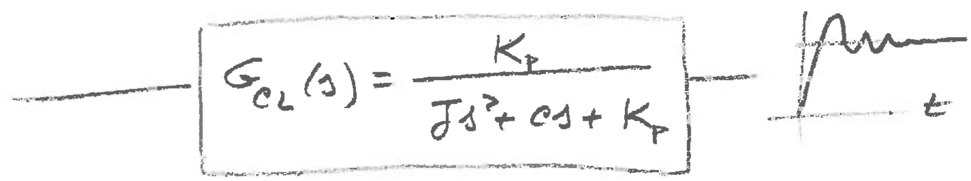
	t-domain	s-domain		comments	
P (proportional)	K	K		• simplest • adds stiffness, increases frequency	
I (integral)	$\int dt$	$\frac{K_i}{s}$	$\frac{1}{T_i s}$	• eliminates offsets • increases system order • may be unstable	"under compensator"
D (derivative)	d/dt	$K_d s$	$T_d s$	• adds damping • decreases system order • increases sensitivity • used as PD or PID	"anticipates & corrects the error"

Note: D control is not physically realisable. It is done as $\frac{Ns}{s+N} \xrightarrow{N>>1} s$ 224/523

P control of Type 1 system.

<u>Recall</u>:





$$G_{CL}(s) = \frac{K_p}{Js^2 + cs + K_p} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

$$\omega_n^2 = \frac{K_p}{J} \quad , \quad \omega_n = \sqrt{\frac{K_p}{J}}$$

$$\zeta = \frac{c}{2\sqrt{J K_p}}$$

K = factor of proportionality ; "P-control"

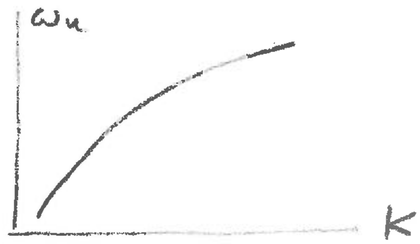
Modification of K_p can modify frequency ω_n and damping ζ

P-control of frequency and damping

K_p = factor of proportionality → P-control

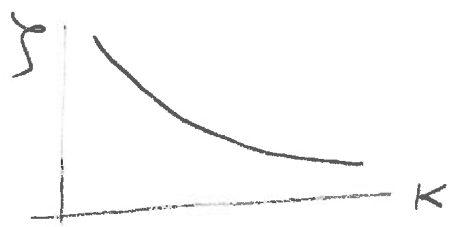
Modification of K_p can modify the frequency ω_n and the damping ζ

$$\omega_n^2 = \frac{K_p}{J}$$



Frequency increases with K

$$\zeta = \frac{c}{2\sqrt{J K_p}}$$

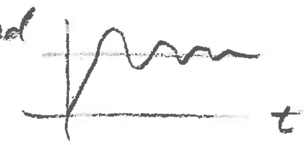


Damping decreases with K_p

Critical damping: $\zeta = 1 \rightarrow \frac{c}{2\sqrt{J K_{cr}}} = 1$: $K_{cr} = \frac{c^2}{4J}$

$0 < K_p \leq K_{cr}$. overdamped converging . exponential response .

$K_{cr} < K_p$ underdamped oscillatory response

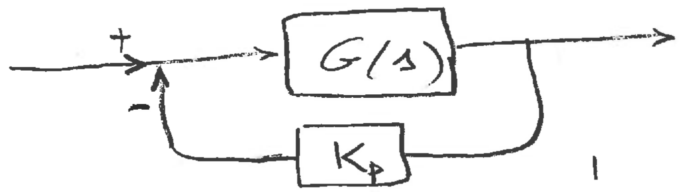


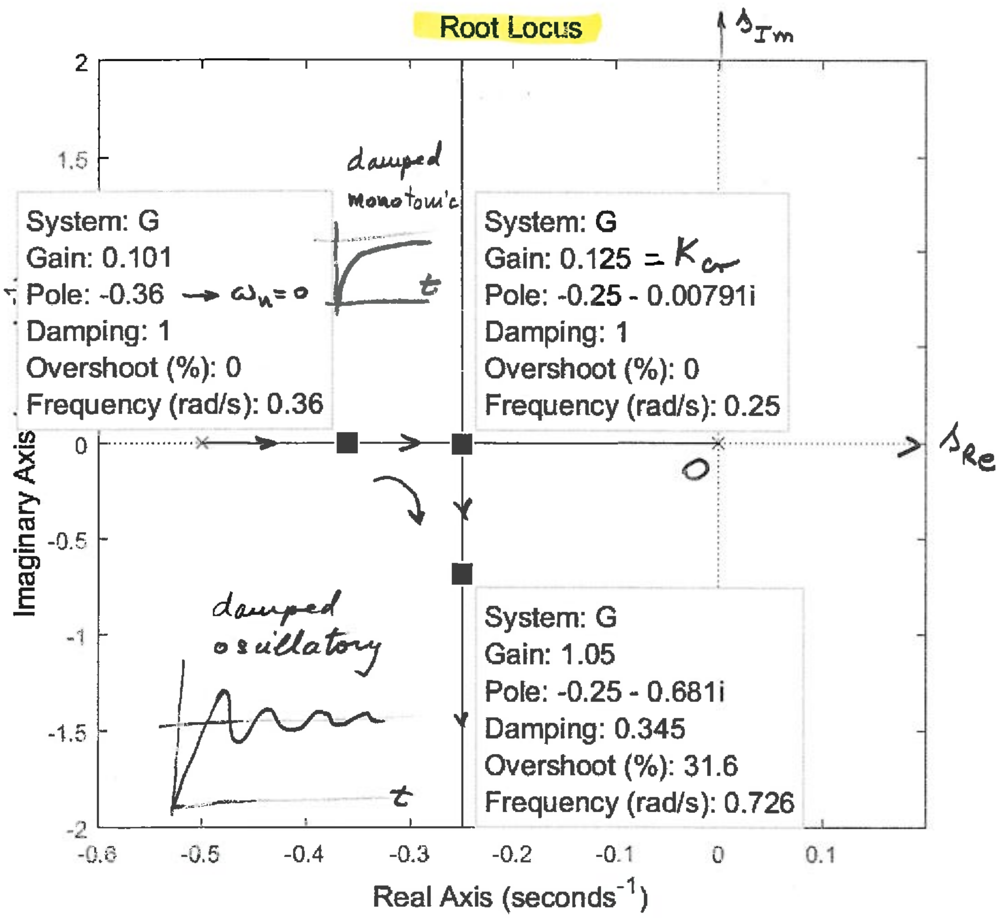
226/523

3p

$$G(s) = \frac{1}{2s^2 + s}$$

$$K_{cr} = \frac{c^2}{4J} = \frac{1}{8} = 0.125$$



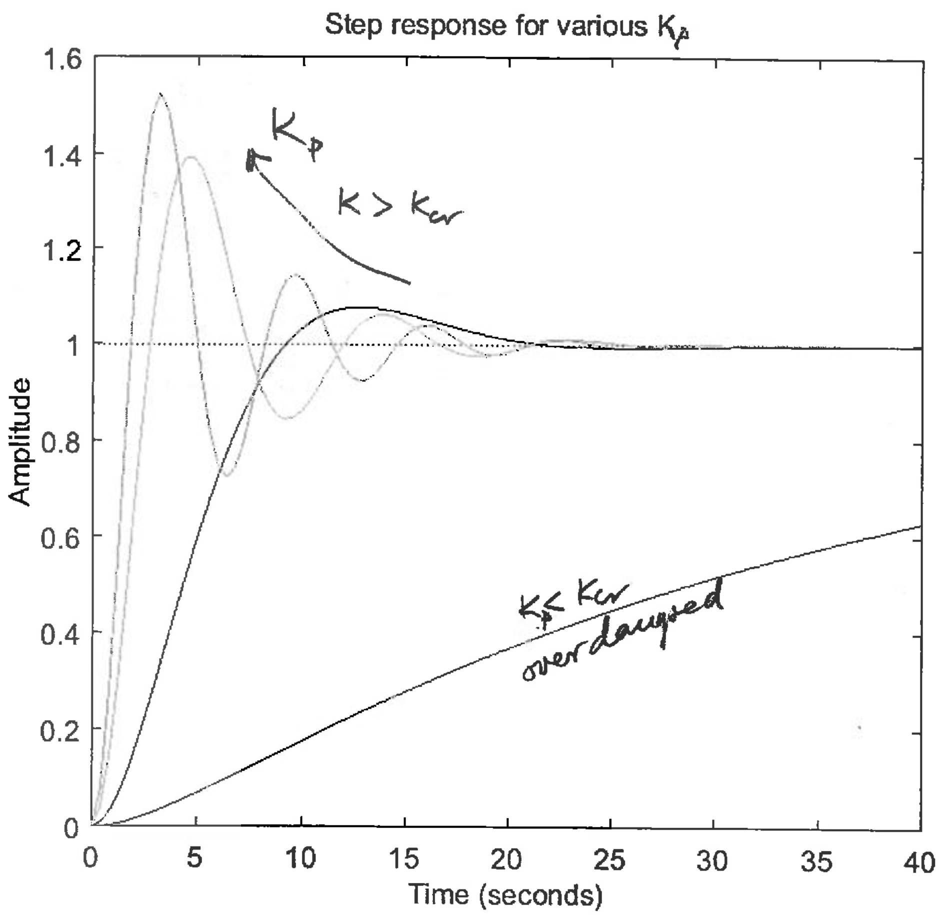


227/523

k_p

P-CONTROL

Type 1 sys



228/523

ζ_P

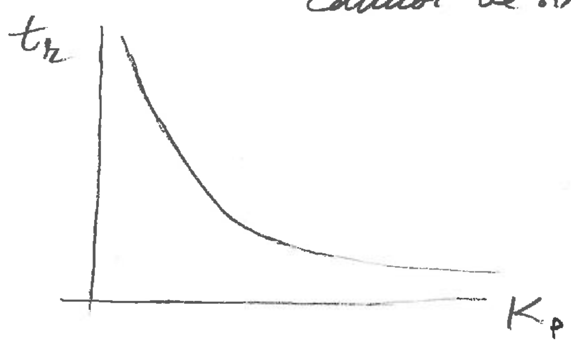
P-control of t_r and M_P

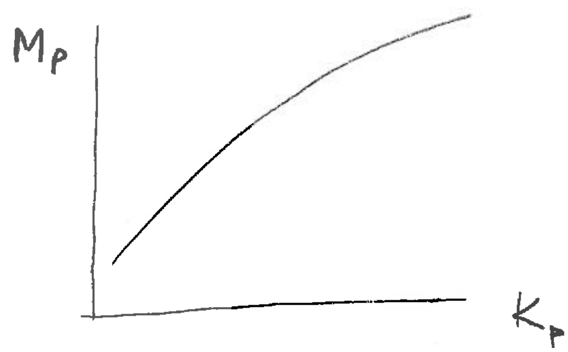
Recall:

rise time $t_r = \dfrac{\pi - \varphi}{\omega_d} = \dfrac{\pi - \varphi}{\omega_n\sqrt{1-\zeta^2}}$, $\varphi = \sin^{-1}\sqrt{1-\zeta^2}$

overshoot $M_p = e^{-\pi\frac{\zeta}{\sqrt{1-\zeta^2}}}$

Small overshoot M_p and small rise time t_r cannot be simultaneously met!



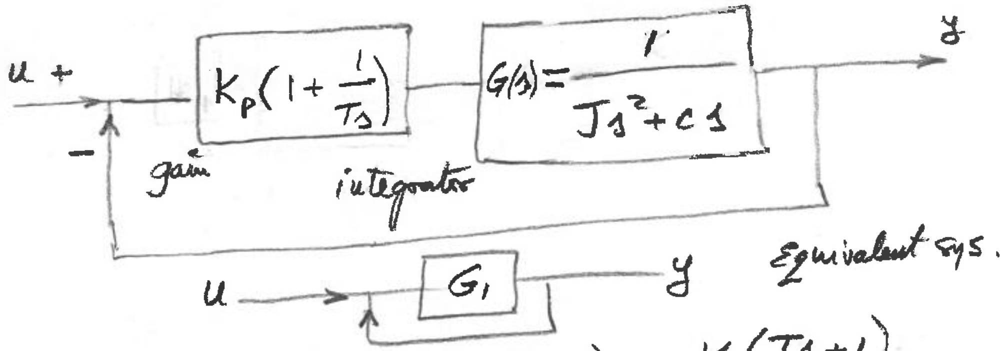


Run MATLAB program

229/523

PI

PI control, Principle



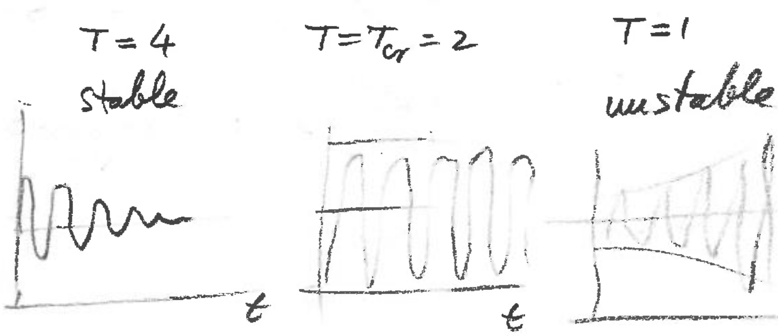
$$G_1(s)=K_p\left(1+\frac{1}{Ts}\right)\left(\frac{1}{Js^2+cs}\right)=\frac{K_p(Ts+1)}{Ts(Js^2+cs)}$$

$$G_{CL}=\frac{G_1}{1+G_1}=\frac{K_p(Ts+1)}{(TJ)s^3+(Tc)s^2+(K_pT)s+K_p}$$

- 3^{rd} order system
- May become unstable
- One could use R–H criterion to predict critical T for instability, i.e.,

$$T > T_{cr} = J/c$$ (see next page.

Example : $J=2$, $c=1$, $T_{cr}=\frac{2}{1}=2$
(see results next page)



230/523

10a P5 RH (Routh-Hurwitz) Criterion Table.

$$\overset{a_0}{(TJ)}s^3 + \overset{a_1}{(Tc)}s^2 + \overset{a_2}{(K_pT)}s + \overset{a_3}{K} = 0$$

s^3	TJ	K_pT
s^2	Tc	K_p
s^1	b_1	
s^0	K_p	

$$b_1 = \frac{(Tc)(K_pT) - (TJ)K_p}{Tc} = \frac{(cT-J)K_p}{c}$$

$$c_1 = \frac{\cancel{b_1}K_p}{\cancel{b_1}} = K_p$$

Discussion

s^3	TJ	+ve
s^2	Tc	+ve
s^1	b_1	may be +ve or -ve.
s^0	K_p	+ve

If b_1 is -ve, then sign change, i.e., INSTABILITY

For stability, $b_1 > 0$, i.e. $cT - J > 0$.

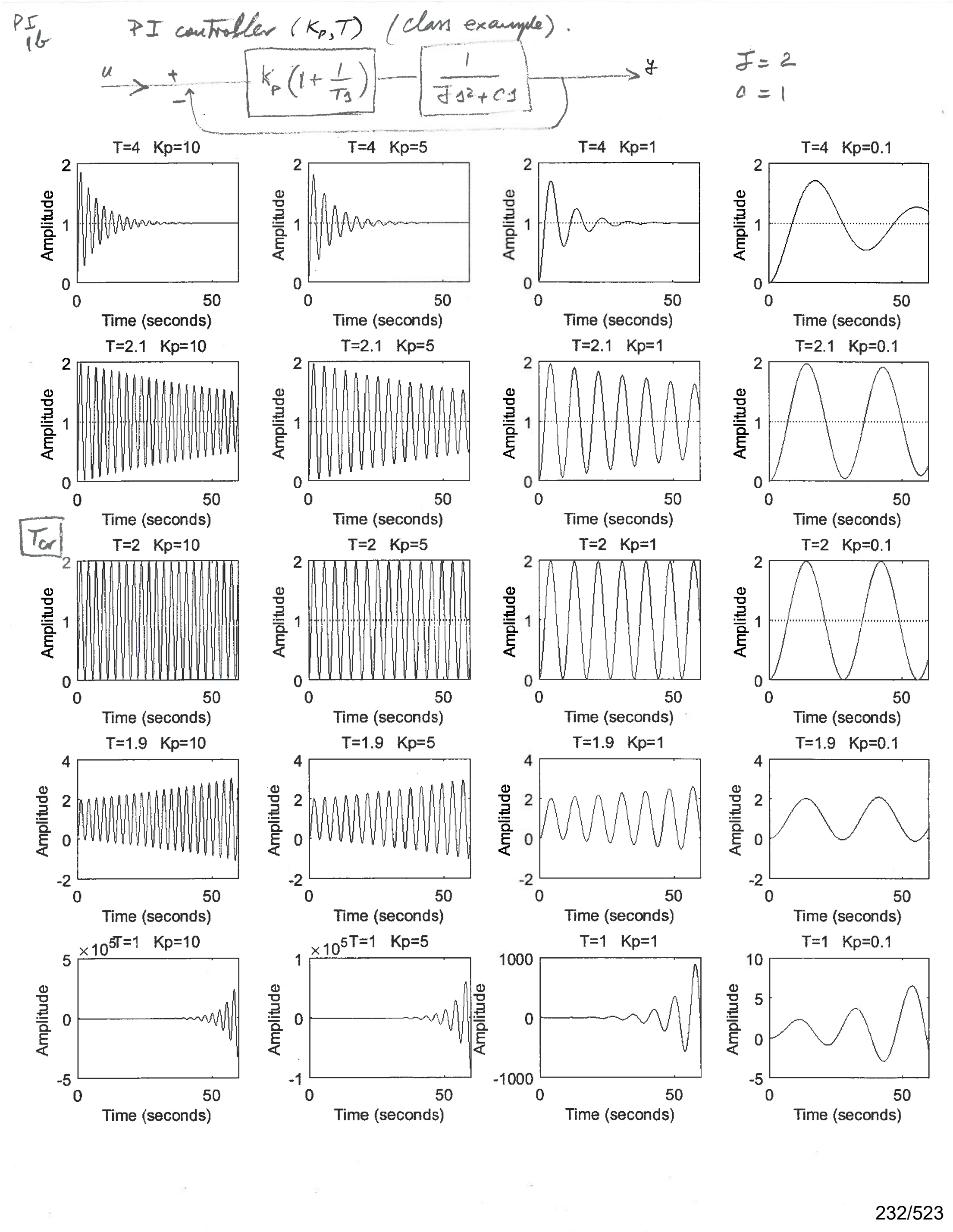
Need $cT > J$

$$T > \frac{J}{c}$$

Denote $T_{cr} = \frac{J}{c}$

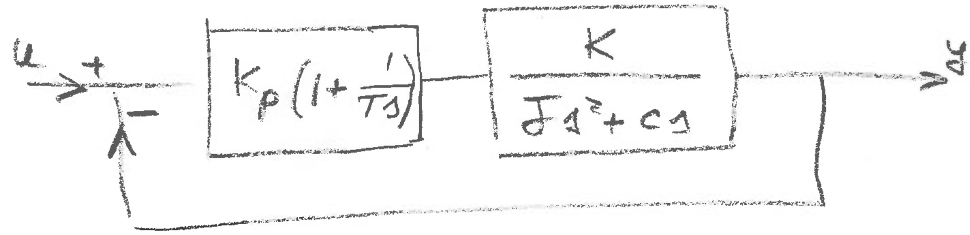
Need $T > T_{cr}$ for stability.

231/523


PI controller (Kp,T) (class example).
$K_p\left(1+\frac{1}{Ts}\right)$
$\frac{1}{Js^2+cs}$
J = 2
c = 1
T=4 Kp=10
T=4 Kp=5
T=4 Kp=1
T=4 Kp=0.1
T=2.1 Kp=10
T=2.1 Kp=5
T=2.1 Kp=1
T=2.1 Kp=0.1
Tcr
T=2 Kp=10
T=2 Kp=5
T=2 Kp=1
T=2 Kp=0.1
T=1.9 Kp=10
T=1.9 Kp=5
T=1.9 Kp=1
T=1.9 Kp=0.1
T=1 Kp=10
T=1 Kp=5
T=1 Kp=1
T=1 Kp=0.1
Amplitude
Time (seconds)
232/523

2 PI

PI Controller (K_p, T)



$$G(s) = K_p\left(1+\frac{1}{Ts}\right)\left(\frac{K}{Js^2+cs}\right) = \frac{K_pK(Ts+1)}{Ts(Js^2+cs)}$$

$$G_{CL} = \frac{G}{1+G} = \frac{K_pK(Ts+1)}{TJs^3+Tcs^2+K_pKTs+K_pK}$$

- 3^{rd} order system
- May become unstable
- R-H stability criterion requires $T > T_{cr}$,
  $T_{cr} = \frac{J}{c}$, for stability (next page)

<u>Example</u> : $K=114$, $J=10$, $c=4$, $T_{cr}=\frac{10}{4}=2.5$
(aircraft)

233/523

20
P1

R-H Table

$$(TJ)s^3+(Tc)s^2+(K_pKT)s+(K_pK)=0$$

s^3	TJ	K_pKT
s^2	Tc	K_pK
s^1	b_1	
s^0	K_pK	

$$b_1=\frac{TcK_pKT-\cancel{T}JK_pK}{\cancel{T}c}=\frac{Tc-J}{c}K_pK$$

Discussion

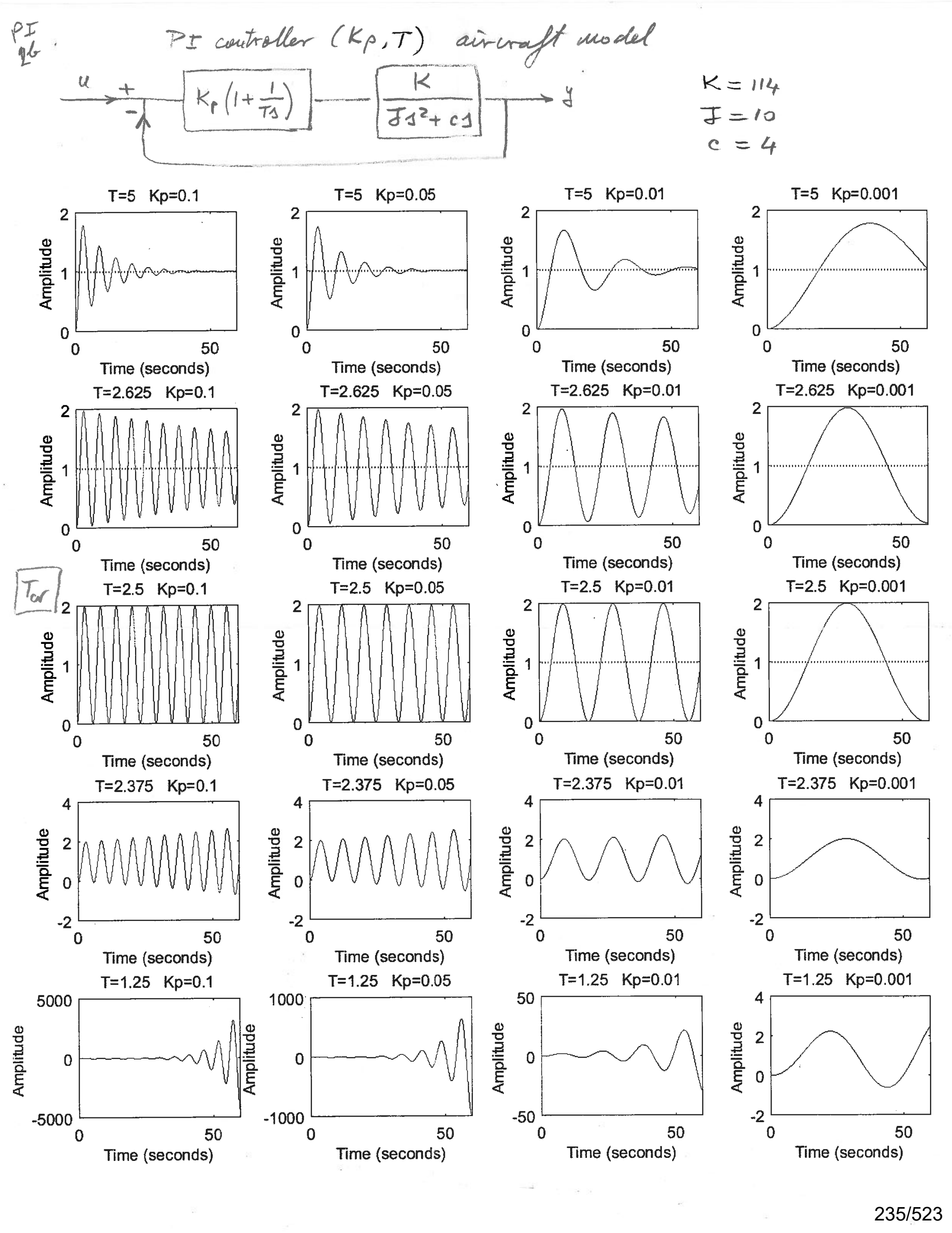
s^3	TJ	+ve
s^2	Tc	+ve
s^1	b_1	may be +ve or −ve
s^0	K_pK	+ve

INSTABILITY happens if $b_1<0$, i.e., $Tc<J$.

For stability, need $Tc>J$ or

$$T>T_{cr},\quad T_{cr}=\frac{J}{c}$$

234/523


PI 9b
PI controller (Kp, T) aircraft model
u
y
K = 114
J = 10
c = 4
T=5 Kp=0.1
T=5 Kp=0.05
T=5 Kp=0.01
T=5 Kp=0.001
T=2.625 Kp=0.1
T=2.625 Kp=0.05
T=2.625 Kp=0.01
T=2.625 Kp=0.001
Tcr
T=2.5 Kp=0.1
T=2.5 Kp=0.05
T=2.5 Kp=0.01
T=2.5 Kp=0.001
T=2.375 Kp=0.1
T=2.375 Kp=0.05
T=2.375 Kp=0.01
T=2.375 Kp=0.001
T=1.25 Kp=0.1
T=1.25 Kp=0.05
T=1.25 Kp=0.01
T=1.25 Kp=0.001
Amplitude
Time (seconds)
235/523

2c

PI

Aircraft roll response

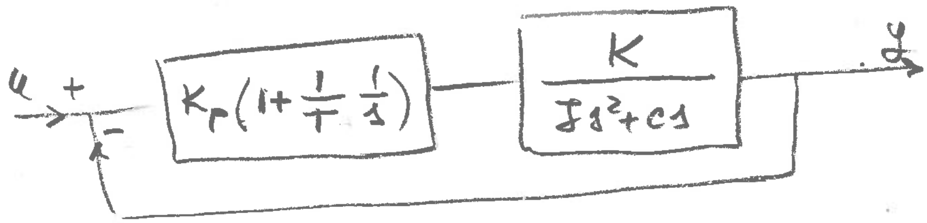
with (K_p, T) PI controller

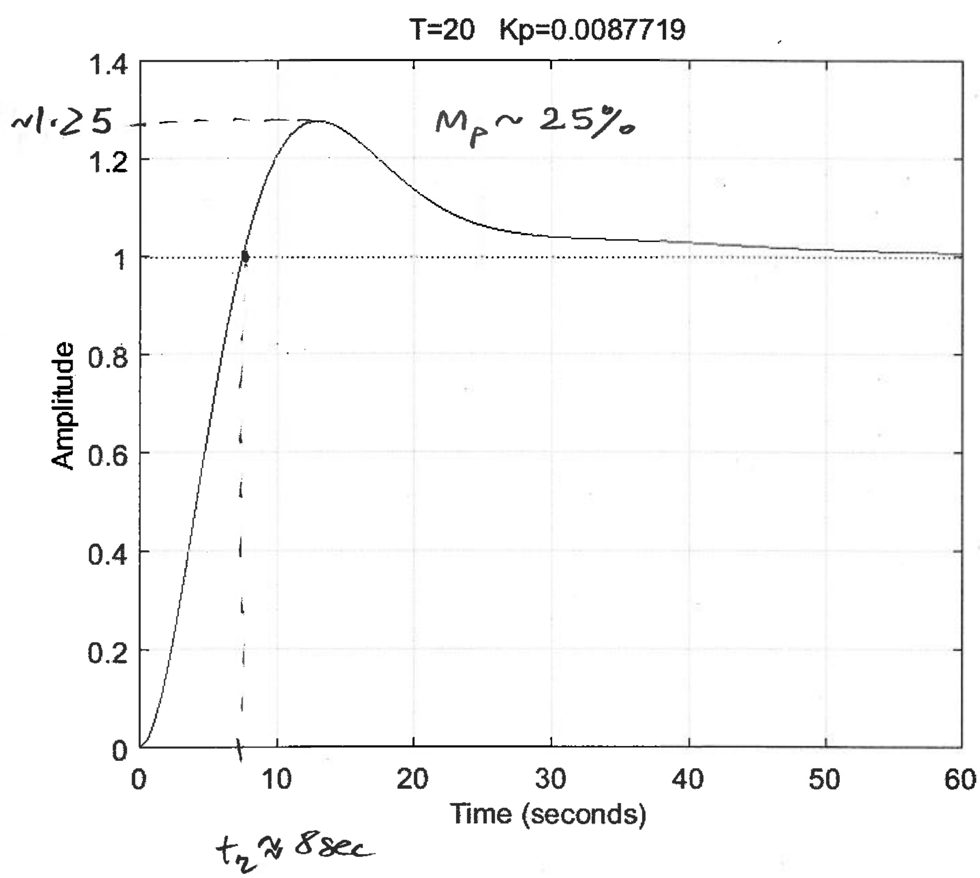
$K_p = 1/K = 1/114$

$T = 20 > T_{cr}$

$K = 114$
$J = 10$
$c = 4$

$T_{cr} = \frac{J}{c} = 2.5$

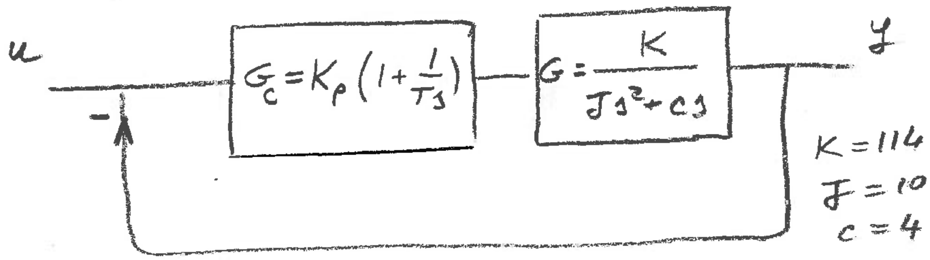




236/523

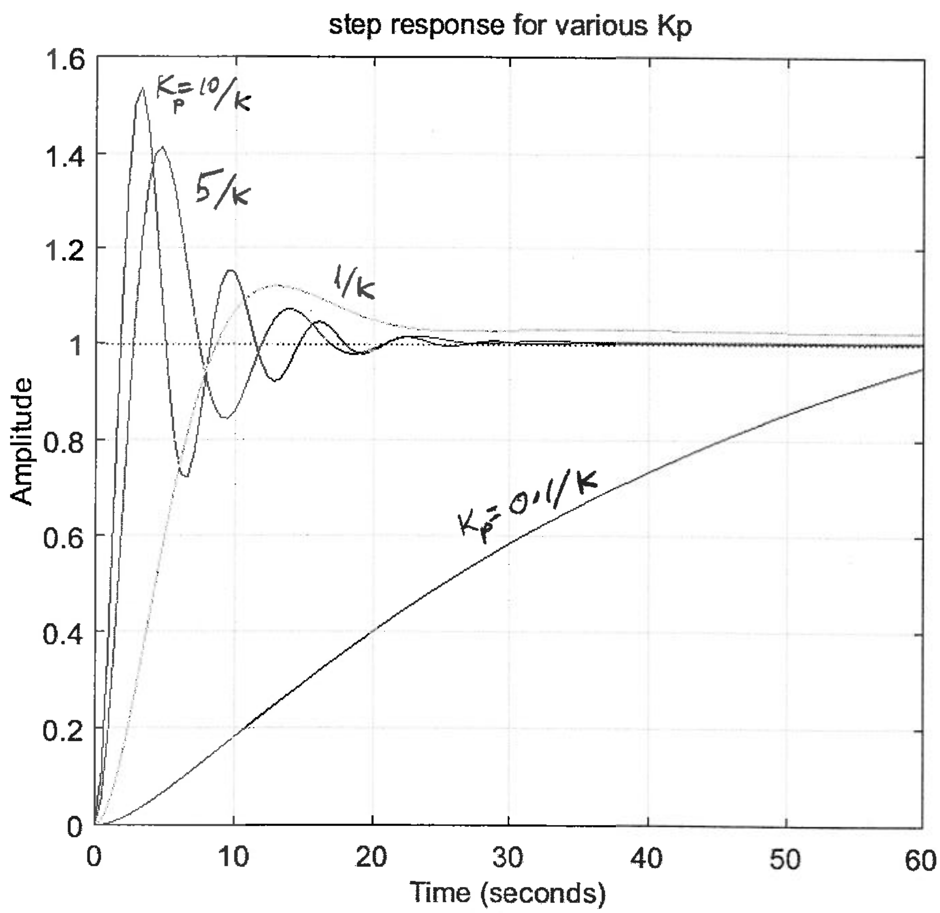
3

PI – control



$K = 114$
$J = 10$
$c = 4$
$T_{cr} = \frac{10}{4} = 2.5$

$T = 100 > T_{cr}$



237/523

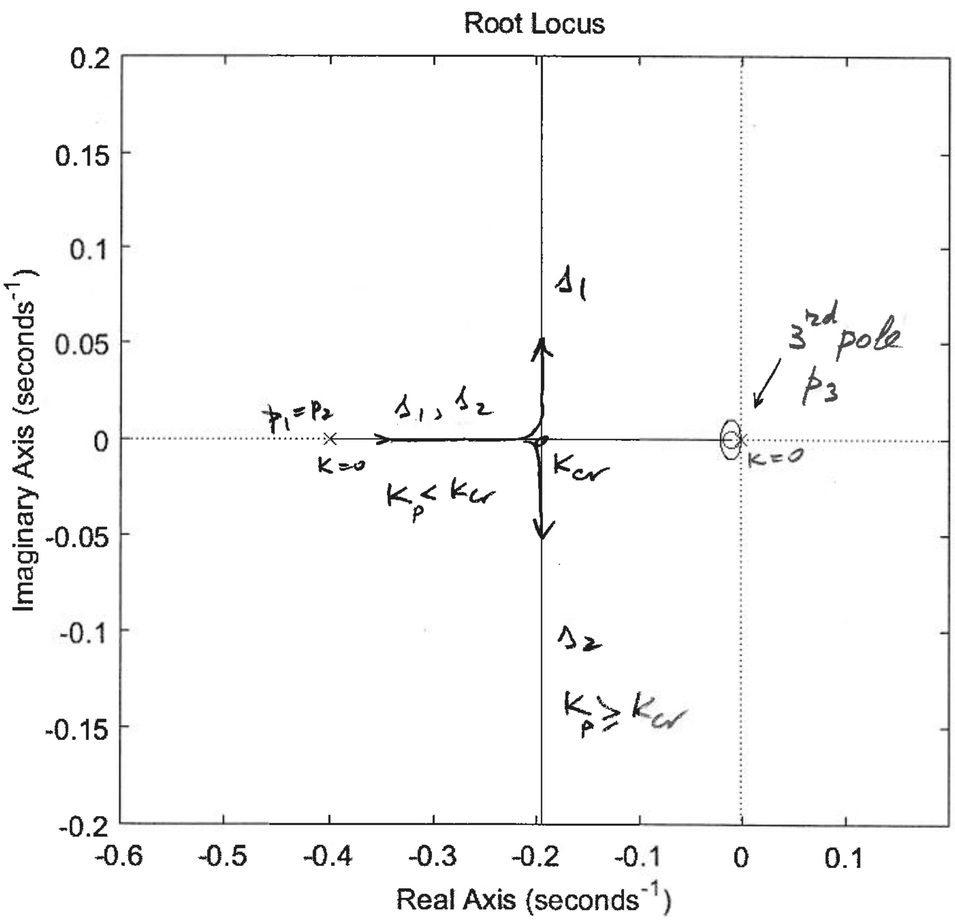
4

PI control

Root locus

aircraft $\begin{cases} K = 114 \\ J = 10 \\ c = 4 \\ T_{cr} = 2.5 \end{cases}$

$T = 100 > T_{cr}$



238/523

5

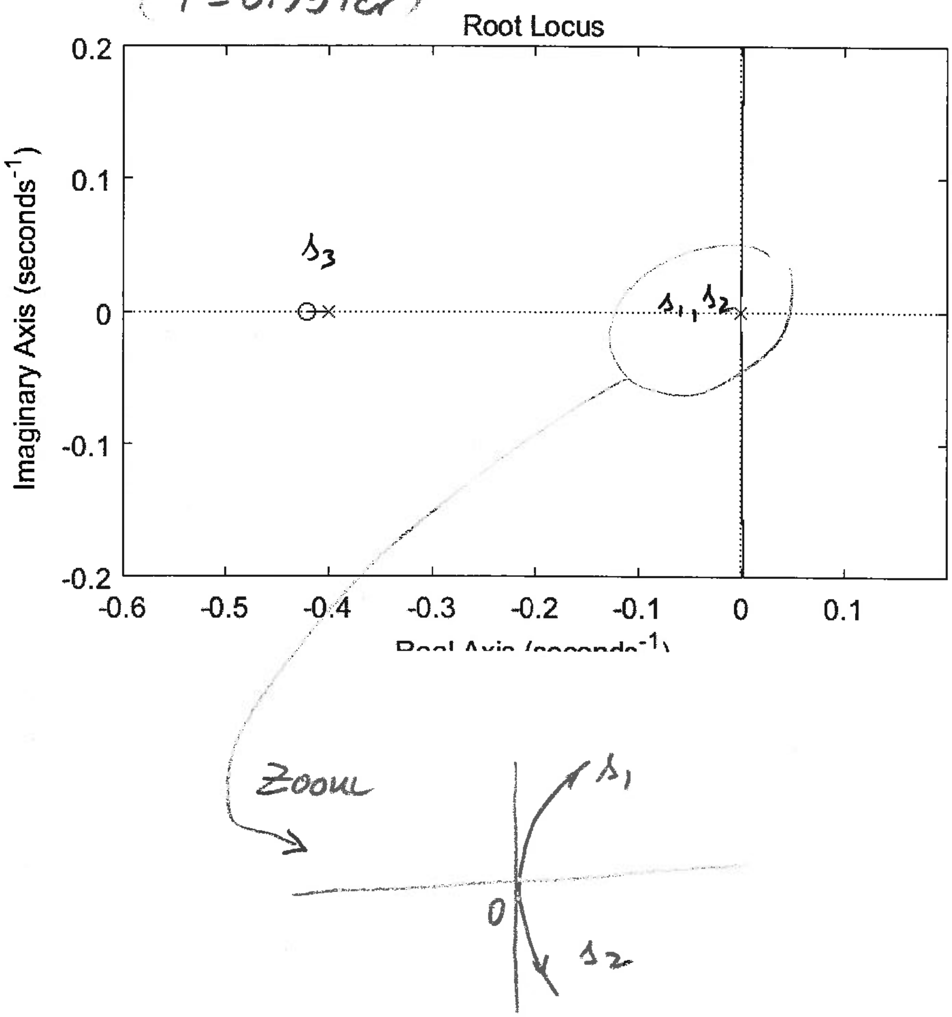
UNSTABLE PI Control
Root locus

aircraft
$K = 114$
$J = 10$
$c = 4$
$T_{cr} = 2.5$

UNSTABLE

$T < T_{cr}$

$(T = 0.95 T_{cr})$



unstable

s_1, s_2 in RHS

239/523

6

UNSTABLE PI Control

Step response

aircraft

$K = 114$

$J = 10$

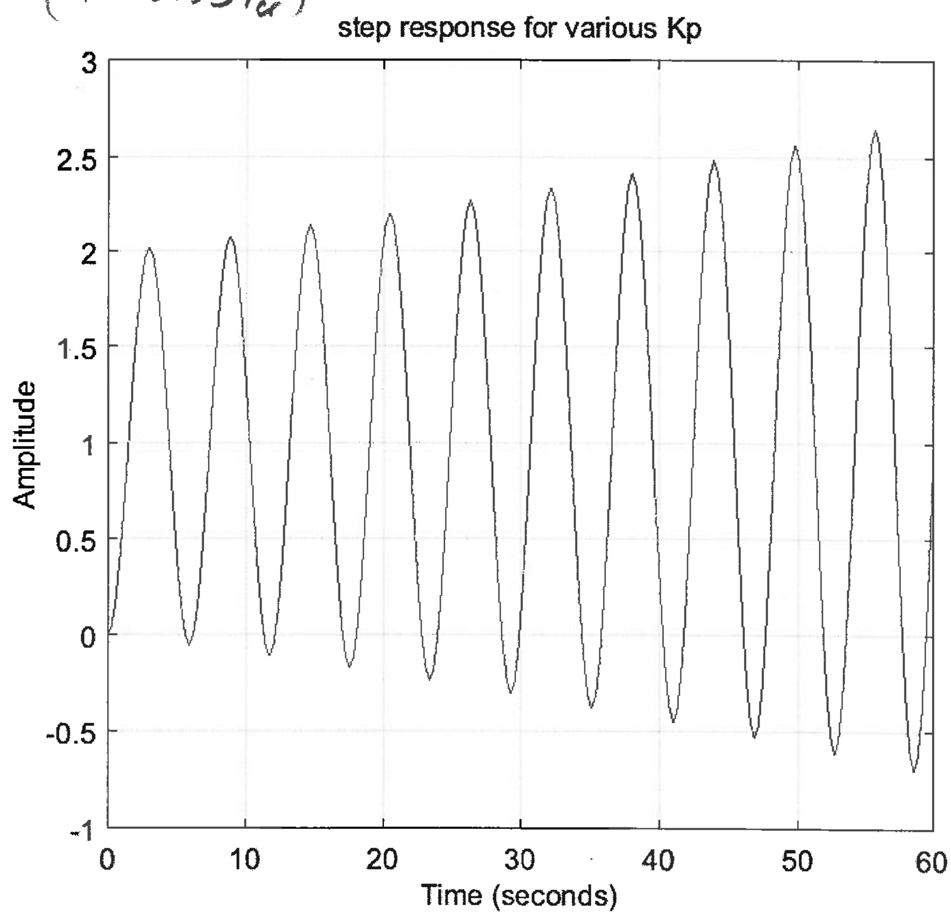
$c = 4$

$T_{cr} = 2.5$

UNSTABLE

$T < T_{cr}$

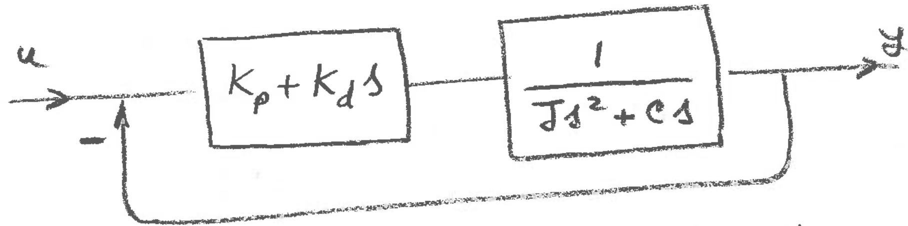
$(T = 0.95 T_{cr})$



240/523

PD

PD - Control



$$G(s) = \frac{1}{Js^2 + cs} \qquad \text{(assume } K=1 \text{ for ease)}$$

$$G_c(s) = K_p + K_d s$$

$$G_1(s) = \frac{K_p + K_d s}{Js^2 + cs}$$

$$G_{CL}(s) = \frac{K_p + K_d s}{Js^2 + cs + K_p + K_d s}$$

$$= \frac{K_p + K_d s}{Js^2 + (c + K_d)s + K_p}$$

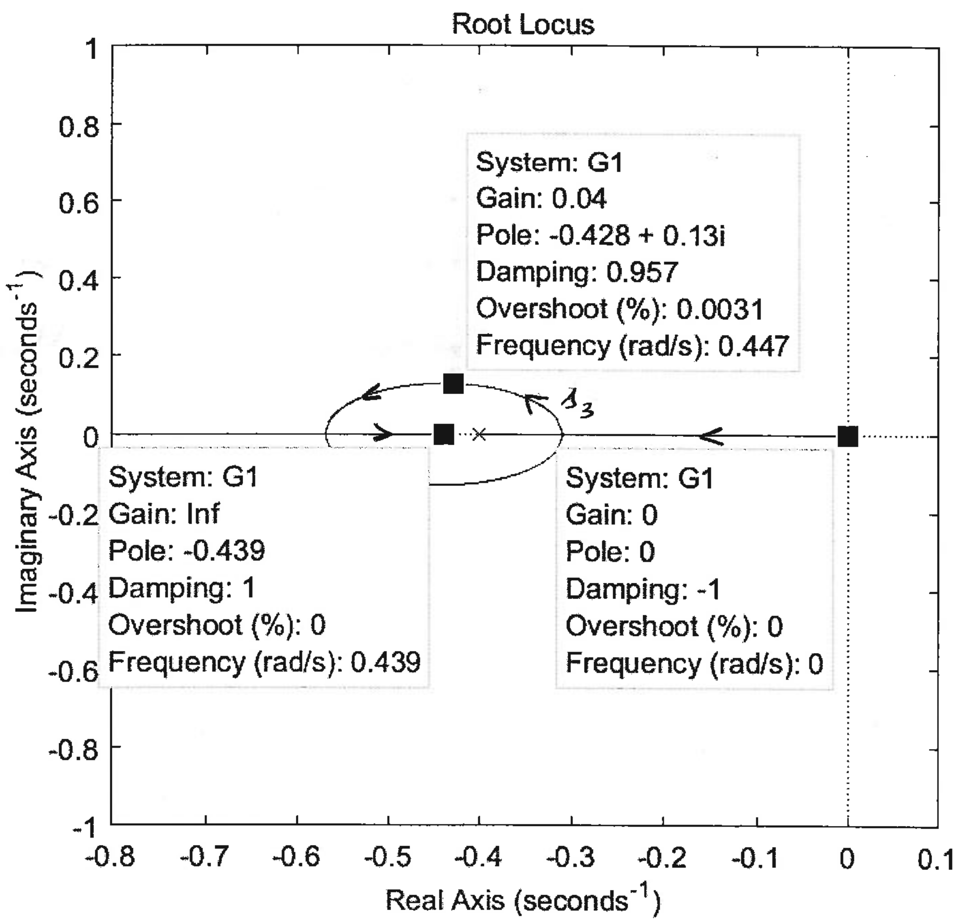
damping is augmented by K_d

Strategy

- Adjust frequency with K_p to get small t_r
- Reduce overshoot by increasing damping (with K_d)

PD controller

Interesting behavior of the poles s_1, s_2 in the root locus



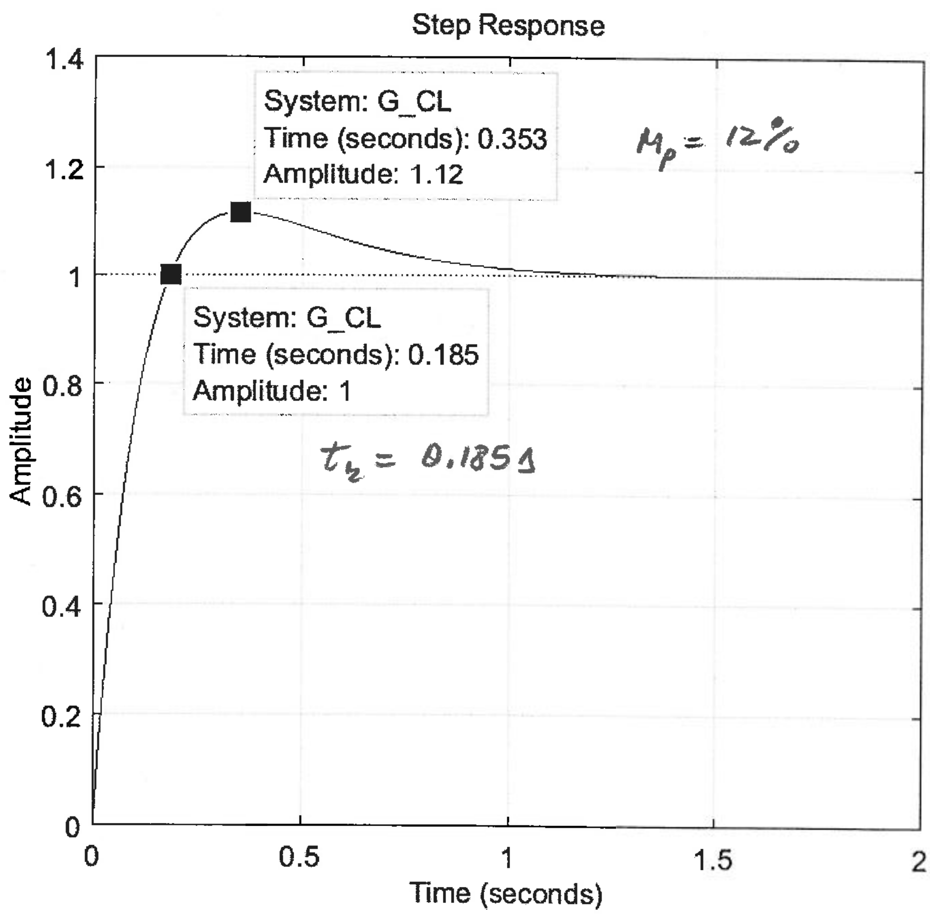
242/523

3
PD

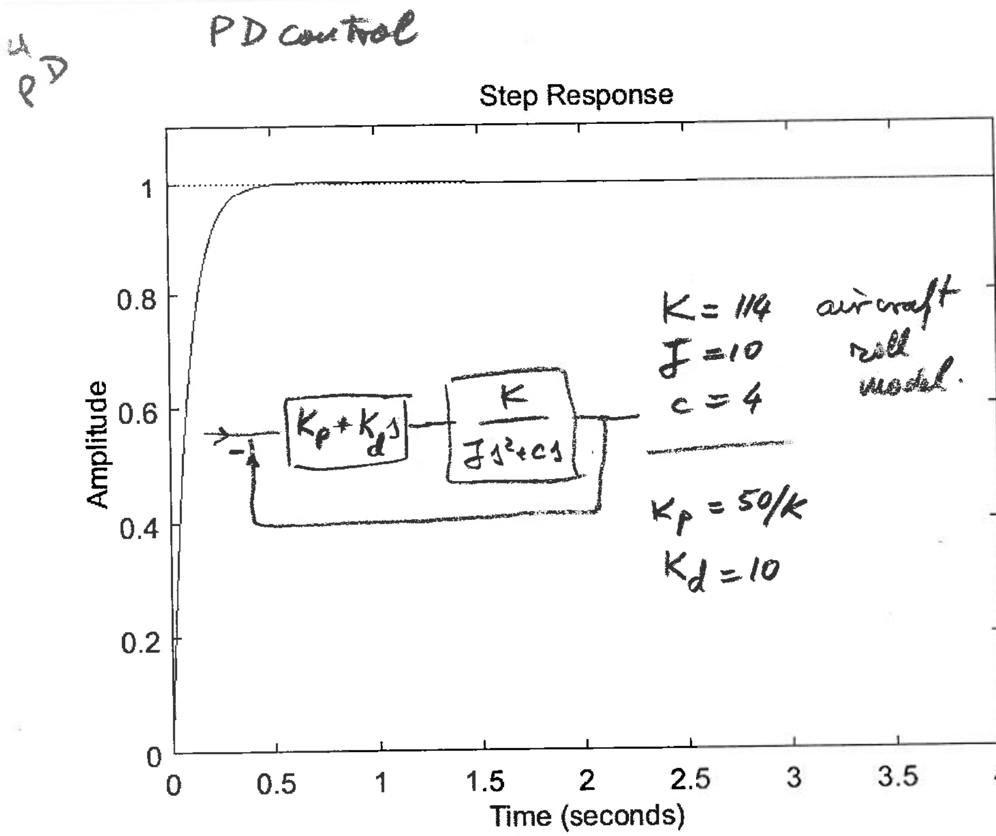
PD control

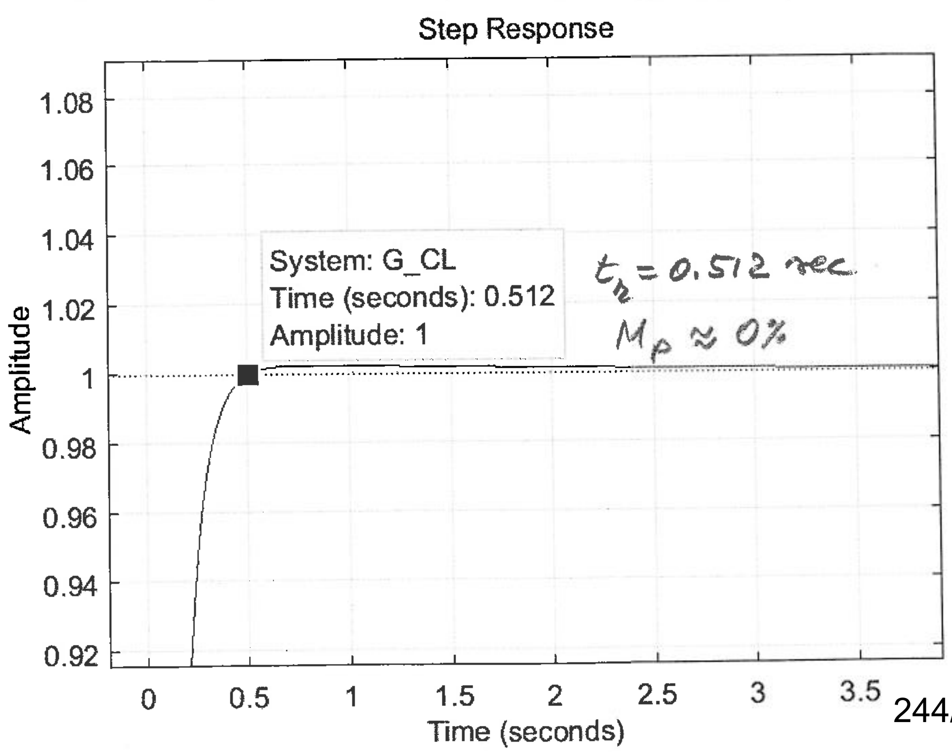
$K = 114$
$J = 10$
$c = 4$

$K_p = 3$
$K_d = 1$



243/523


PD control
Step Response
Amplitude
Time (seconds)
K = 114
J = 10
c = 4
aircraft roll model.
Kp = 50/K
Kd = 10
Kp + Kd s
K
J s² + c s


Step Response
Amplitude
Time (seconds)
System: G_CL
Time (seconds): 0.512
Amplitude: 1
tr = 0.512 sec
Mp ≈ 0%

244/523

7.7 SIMULINK Aircraft Roll Motion

SIMULINK

Aircraft Roll Motion Autopilot Development

Table of Contents:

1 CONTROL SYSTEM DESIGN OBJECTIVES

Consider the aircraft roll transfer function $G(s) = \dfrac{K}{Js^2 + cs} = \dfrac{114}{10s^2 + 4s}$

Design a feedback control system to control the aircraft roll motion with the following control design objectives:

Design Objective 1: Control the unconstraint aircraft motion resulting from an aileron input. Have a autopilot system that can maintain the aircraft at a constant bank angle

We wish to achieve this objective through feedback (FB)

Design Objective 2: Achieve a reasonable aircraft roll response.

We define 'reasonable response' using two control design specifications:

- DS1: Fast response time as measured by rise time
  $t_r \leq 1.5$ sec
- DS2: maximum percentage overshoot for step input less than 20%
  $M_p \leq 20\%$

2 UNCONSTRAINED AIRCRAFT ROLL MOTION

2.1 MODEL

Open a new model canvas and save it as 'SIMULINK_airplane_roll_unitFB'

From Simulink Library Browser drag onto the new canvas the following blocks:

- select 'Step' from 'Sources'
- select 'Transfer Fcn' from 'Continuous'
- select 'Scope' from 'Sinks', enter 'Number of inputs' = 2

Create annotation text boxes above each box as follows:

- 'reference signal' above the 'Step' box
- 'aircraft roll dynamics' above 'Transfer Fcn' box
- 'display' above 'Scope'

Change properties of 'Transfer Fcn' block to represent the aircraft roll transfer function $G(s) = \frac{K}{Js^2 + cs} = \frac{114}{10s^2 + 4s}$

Connect the blocks:

- 'reference signal' → 'aircraft dynamics' → 'display' (1st port)
- 'reference signal' → 'display' (2nd port)

Use the pull down menu: 'Edit → Copy Current View to Clipboard → Metafile' to capture only the model as shown in Figure 1.

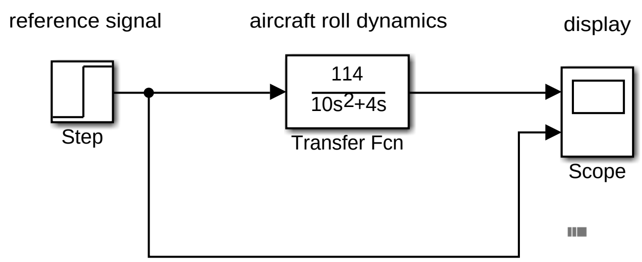


Figure 1

247/523

2.2 SIMULATION PARAMETERS

Use the pull down menu to open Configuration Parameters window, i.e.,
Simulation → Model Configuration Parameters
Make:

- Max step size: 1e-4
- Stop time: 2

The rest should remain unchanged (verify that they are the same as inFigure 2)

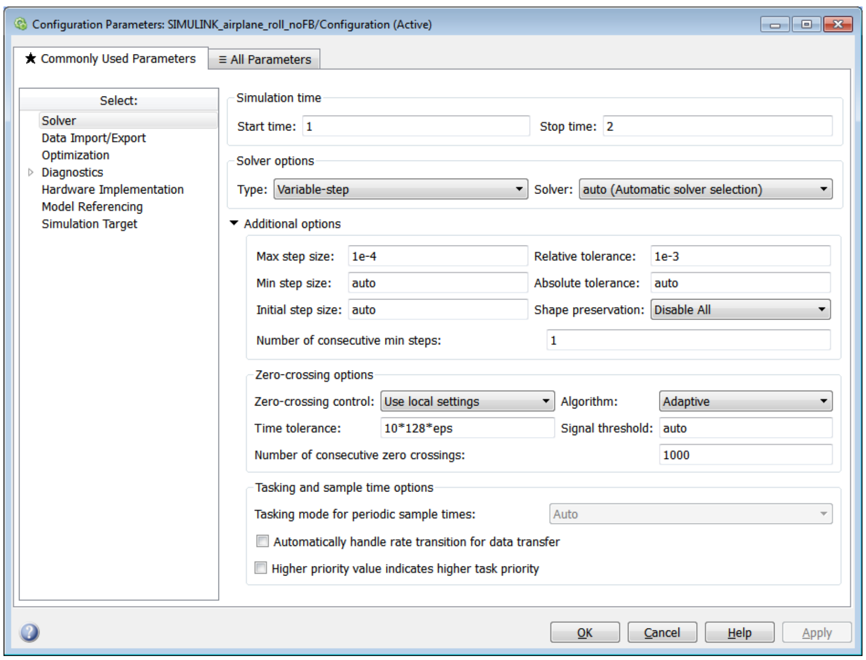


Figure 2

2.3 Display Setup

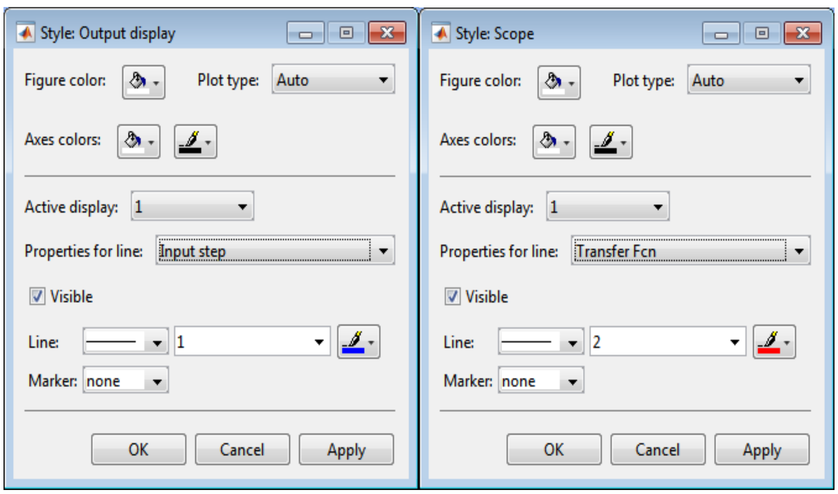
Double click on the ‘Scope’ block in the SIMULINK model. The ‘Scope’ display should open in a new window.

Select ‘View →Style’ to open ‘Style Scope’ dialog box (Figure 3a).
Choose color white for:

- Figure color: ‘bucket’
- Axes colors: ‘bucket’

Choose color black for Axes colors: ‘brush’

Choose color blue for ‘Properties for line: Input step’ (Figure 3a)
Choose color red for ‘Properties for line: Transfer Fcn’ (Figure 3b)



(a) (b)

Figure 3

The ‘Scope’ display should look as shown in Figure 4:

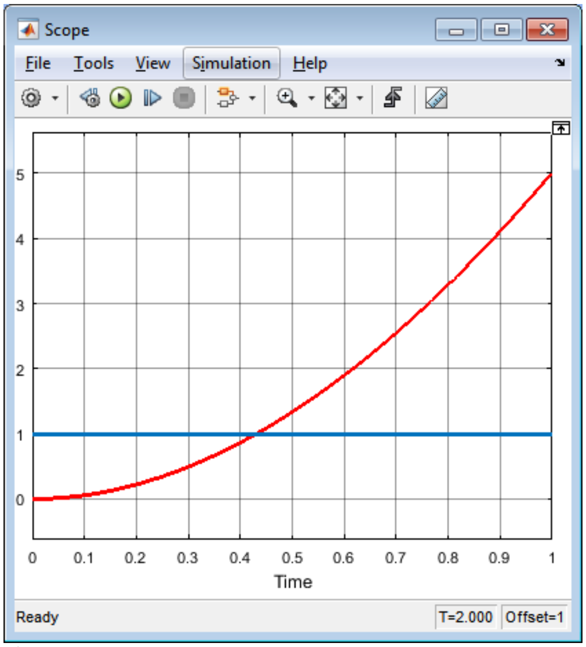


Figure 4

Note that the blue line represents the step input whereas the red line represent the aircraft roll response, which growth continuously. This situation is unacceptable. To counteract it, use feedback control as shown in next section.

3 UNIT FEEDBACK CONTROL OF AIRCRAFT ROLL MOTION

Open a new model canvas and save it as 'SIMULINK_airplane_roll_unitFB'
From Simulink Library Browser drag onto the new canvas the following blocks:

- 'Step' from 'Sources'
- 'Sum' from 'Commonly Used Blocks'
- 'Scope' from 'Sinks'
- 'Transfer Fcn' from 'Continuous'

Create annotation text boxes above each box as follows:

- 'reference signal' above the 'Step' box
- 'aircraft roll dynamics' above 'Transfer Fcn' box
- 'output signal' above 'Scope'

Change properties of 'Transfer Fcn' block to represent the aircraft roll transfer function $G(s) = \dfrac{114}{10s^2 + 4s}$

Change the second port of the 'Sum' block to negative (-). Do this by using 'right-click' → 'Block Parameters: Sum' and then putting '+-' in 'List of signs'

Connect the blocks:

- 'reference signal' → 'Sum' → 'aircraft dynamics' → 'display'
- output from 'aircraft dynamics' to negative (-) port of the 'Sum block' (this is the unit feedback closed loop)

Use the pull down menu: 'Edit → Copy Current View to Clipboard → Metafile' to capture only the model (Figure 5).

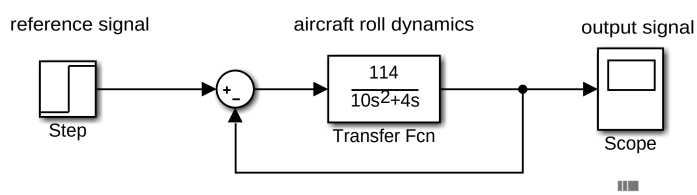


Figure 5

3.1 SIMULATION PARAMETERS

Use the pull down menu to open Configuration Parameters window, i.e.,
Simulation → Model Configuration Parameters
Make:

- Max step size: 1e-4
- Stop time: 40

The rest should remain unchanged (verify that they are the same as in the figure)

251/523

3.2 Display Setup

Double click on the ‘Scope’ block in the SIMULINK model. The ‘Scope’ display should open in a new window.

Select ‘View →Style’ to open ‘Style Scope’ dialog box.
Choose color white for:

- Figure color
- Axes colors: bucket

Choose color black for Axes colors: brush
‘Properties for line: Transfer Fcn’: 1.5 weight, red (Figure 6).

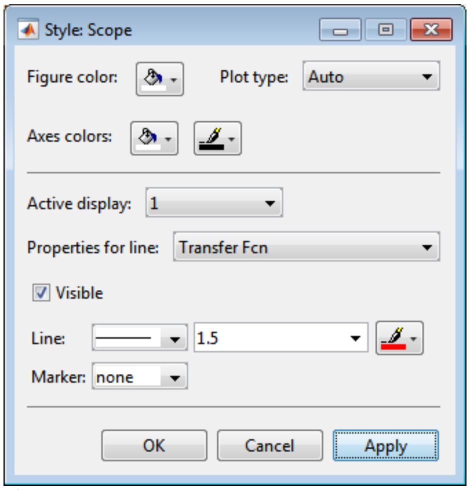


Figure 6

252/523

3.3 Aircraft Roll Response with Unit FB

Press the 'Run' button. The aircraft roll response with unit FB is displayed (Figure 7).

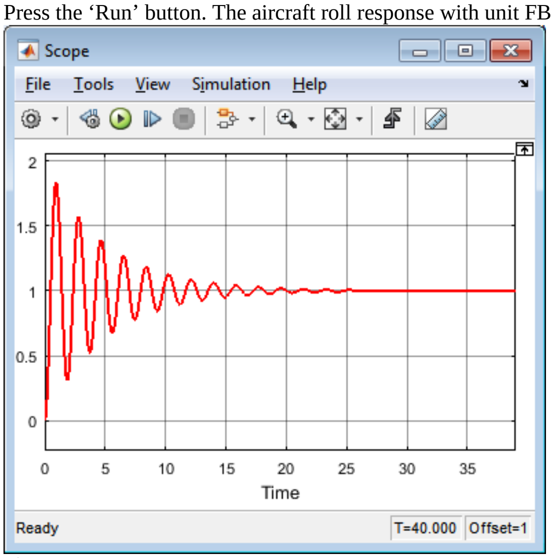


Figure 7

It is apparent that the response is unsatisfactory because:

- many oscillations until it settles down to $x_{ss} = 1$
- large overshoot ($x_p \approx 1.8$, $M_p \approx 80\%$)

4 VARIABLE GAIN FB CONTROL OF AIRCRAFT ROLL MOTION (P CONTROL)

To improve the aircraft FB response, we can try to use variable gain control. This is known as proportional control or ‘P control’.

4.1 P-controlled Aircraft Roll Model

Add a gain box to the model to obtain the P-controller (Figure 8).

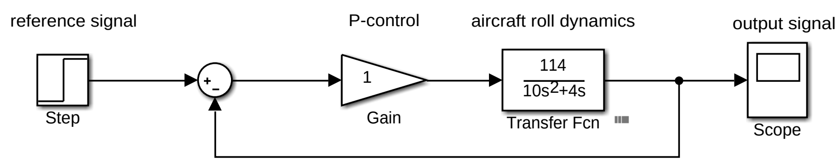


Figure 8

Save this model as ‘SIMULINK_aircraft_roll_P_control’

4.2 Display Setup

Have the display setup as before.
In addition, stop automatic axes scaling by doing the following:

- Tools → Axes Scaling → Axis Scaling Properties

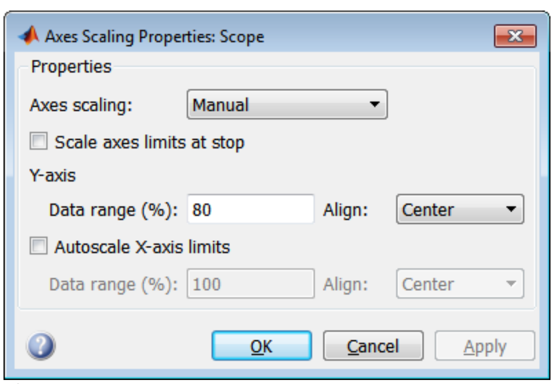


Figure 9

4.3 P-Control Gain Changes

The P-control gain Kp can be modified as follows:

Right click 'Gain' box, select 'Block Parameters'. The 'Block Parameters: Gain' dialog box open (Figure 10).

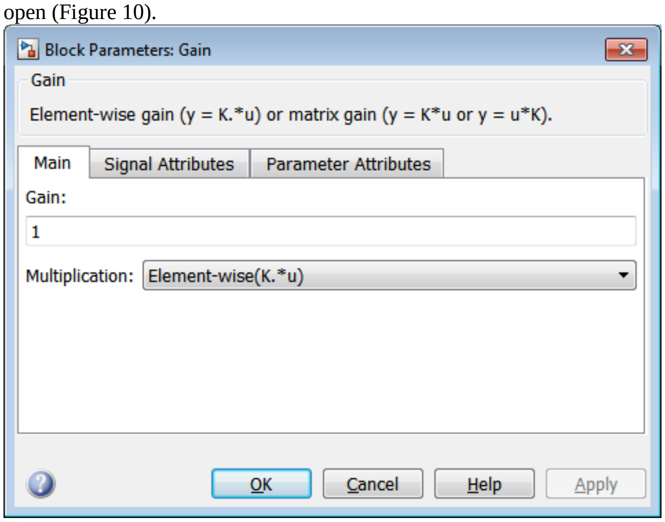


Figure 10

The gain default value is 'Gain: 1'. , i.e., Kp=1. After entering a new gain value, press 'Apply'. Keep the dialog box open for the next gain value. (simulation runs OK with the dialog box open.)

Try a number of different gains to see their effect on the response. Gains to be tried are:

- Kp=1
- Kp=10
- Kp=0.1
- Kp=0.01

4.3.1 Kp=1 Response

Put ‘Gain = 1’ in ‘Block Parameters: Gain’ dialog box and press ‘Apply’. The response is shown in Figure 11.

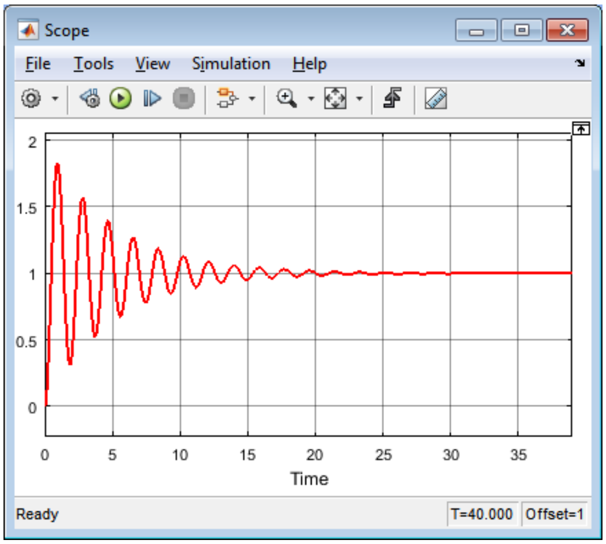


Figure 11

This Kp=1 case corresponds to Unit FB which we have already studied. It is unsatisfactory because

- many oscillations until it settles down to $x_{ss} = 1$
- large overshoot ($x_p \approx 1.8$, $M_p \approx 80\%$)

4.3.2 Kp=10 Response

Put ‘Gain = 10’ in ‘Block Parameters: Gain’ dialog box and press ‘Apply’. The response is shown in Figure 12.

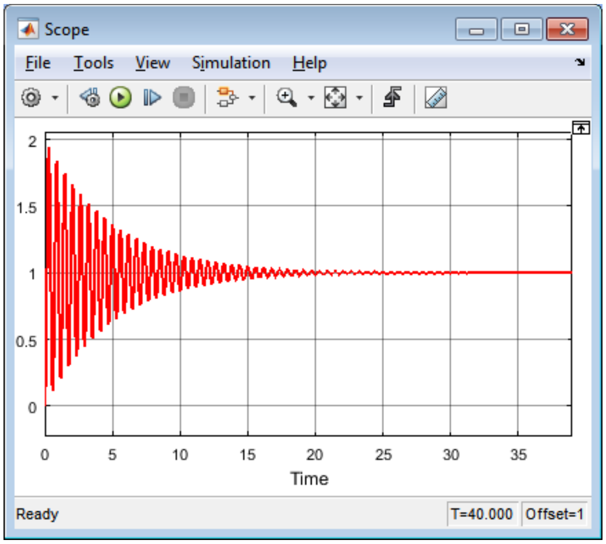


Figure 12

This Kp=10 response is unsatisfactory because

- very many oscillations until it settles down to $x_{ss} = 1$
- large overshoot ($x_p \approx 2$, $M_p \approx 100\%$)

4.3.3 Kp=0.1 Response

Put ‘Gain = 0.1’ in ‘Block Parameters: Gain’ dialog box and press ‘Apply’. The response is shown in Figure 13.

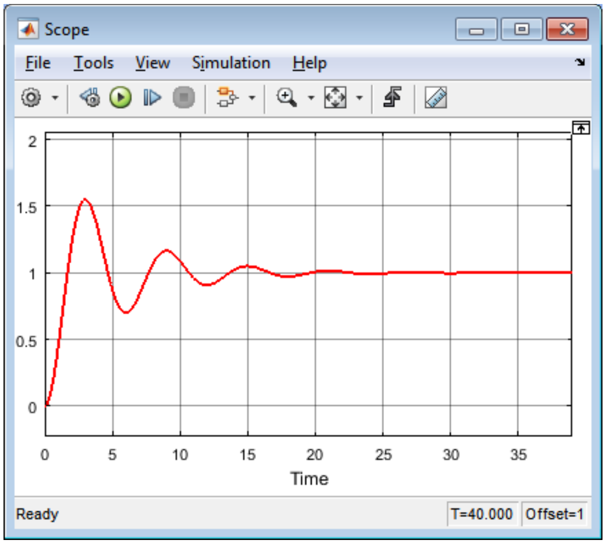


Figure 13

The response for Kp=0.1 seems better because:

- fewer oscillations until it settles down to $x_{ss} = 1$
- smaller overshoot ($x_p \approx 1.55$, $M_p \approx 55\%$)

4.3.4 Kp=0.01 Response

Put 'Gain = 0.01' in 'Block Parameters: Gain' dialog box and press 'Apply'. The response is shown in Figure 14.

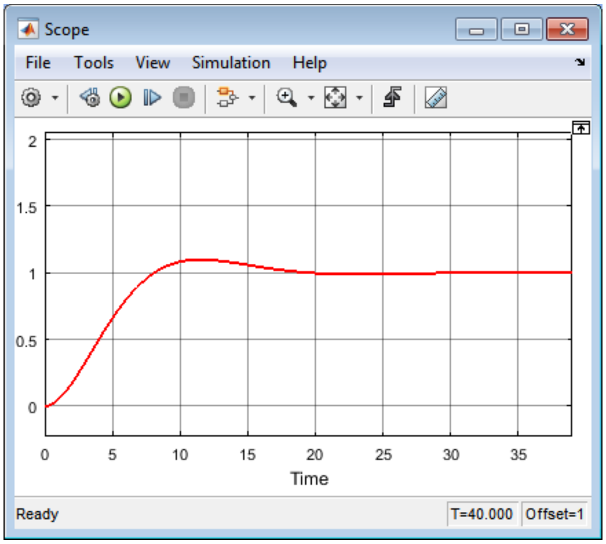


Figure 14

The response for Kp=0.01 seems even better because:

- almost no oscillation until it settles down to $x_{ss} = 1$
- much smaller overshoot ($x_p \approx 1.1$, $M_p \approx 10\%$)

However, the rise time is much longer than before ($t_r \approx 8$ sec). This is unacceptable because it makes the aircraft very sluggish.

We need to examine the mathematics of P-control. Recall:

$\omega_n = \sqrt{\frac{K}{J}}$, $\zeta = \frac{c}{2\sqrt{JK}}$

where K is the overall forward gain (K=Kp*K1, with K1=114 for our aircraft roll model). Hence we can calculate:

- rise time: $t_r = \frac{\pi - \varphi}{\omega_n \sqrt{1-\zeta^2}}$ where $\varphi = \sin^{-1}\sqrt{1-\zeta^2}$
- overshoot: $M_p = e^{-\pi \frac{\zeta}{\sqrt{1-\zeta^2}}}$

A plot of these values is given in Figure 15.

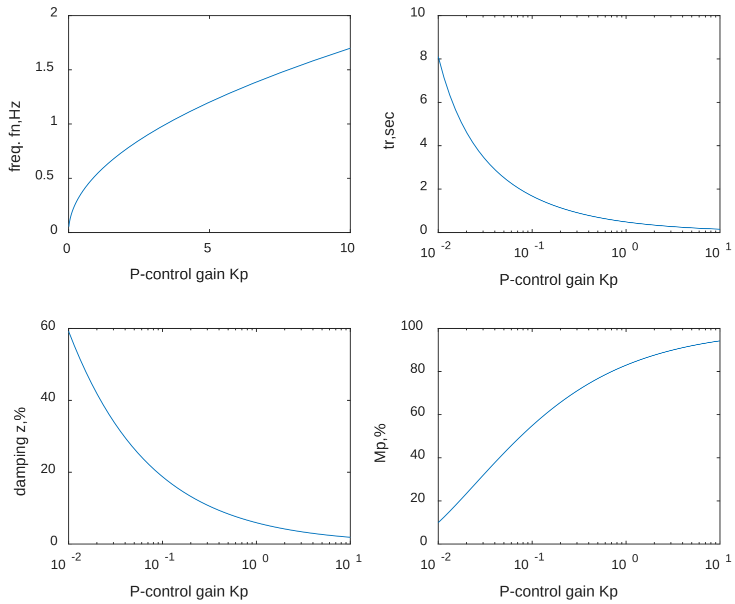


Figure 15

The above plot shows that small overshoot M_p happens at low gain values. However, at these low gain values, the rise time t_r becomes large. Hence, we conclude that **P-control gain has an opposite effect on the two performance indicators, M_p and t_r considered in this controller design.** We need to explore other control options.

5 PI CONTROL

PI control stands for ‘proportional + integrative control’.

5.1 PI CONTROL SETUP

The aircraft model with PI control has two branches, one for the P block (which is just a simple gain) and the other for the I block which consists of a gain followed by an integrator. The PI controlled aircraft model looks as shown in Figure 16.

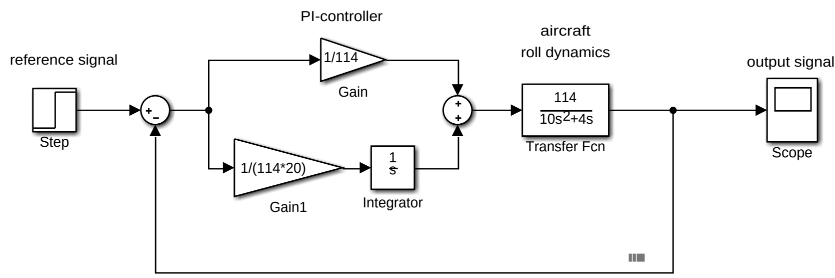


Figure 16

Construct this model and save it as ‘SIMULINK_aircraft_roll_PI_control_Kp_Ki’

Observe that the model has two gains:

- Proportional gain, Kp=1/114
- Integrative gain Ki=1/(114*20)

The resulting PI controller has the expression:

$$G_{PI}(s) = K_p + K_i \frac{1}{s}$$

5.2 Aircraft Roll Response with PI Control

Run the PI control model. The resulting response will show up in the 'Scope' window (Figure 17):

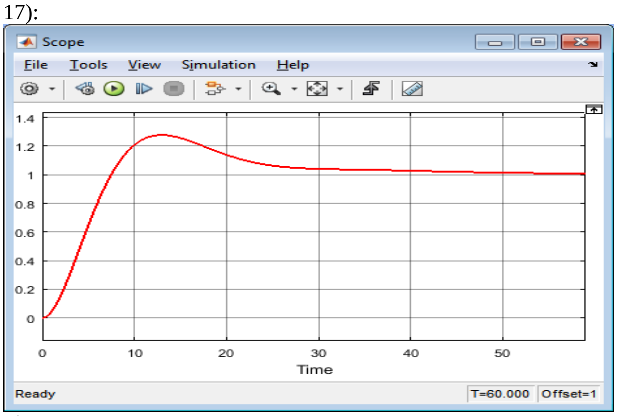


Figure 17

Press the 'Cursor Measurements' button and set the cursors to measure rise time and overshoot (Figure 18).

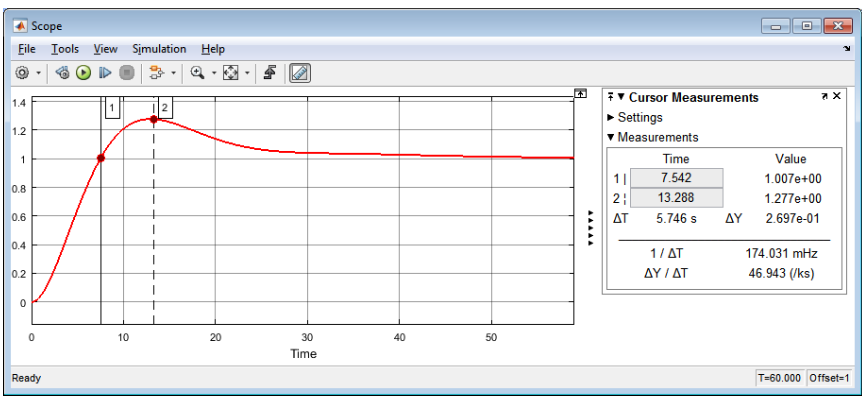


Figure 18

The results are:

- rise time $t_r = 7.542$ sec
- overshoot $M_p = 27.7\%$

These values are better than the values obtained with P-control. However, they are still below the target values. Need to try something else.

6 PD CONTROL

PD control stands for ‘proportional + derivative control’.

6.1 PD Control Setup

The aircraft model with PI control has two branches, one for the proportional P-block (which is just a simple gain) and the other for the derivative D-block. The PD controlled aircraft model looks as shown in Figure 19.

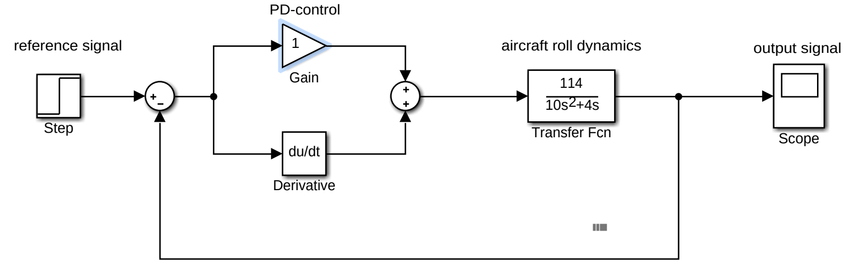


Figure 19

Construct this model and save it as ‘SIMULINK_aircraft_roll_PD_control’

6.2 Aircraft Roll Response with PD Control

Set the model run time to 20 sec and run it. The response is as shown in Figure 20.

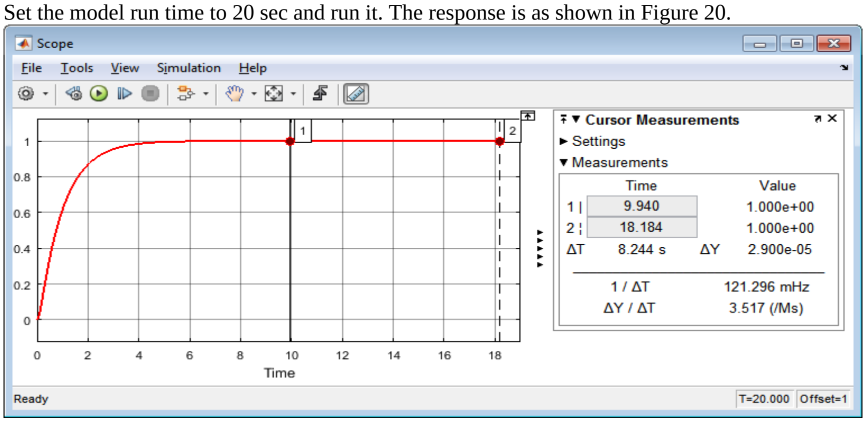


Figure 20

The response convergence continuously towards the target value 1, but the rise time is around 10 sec.

This is unacceptable. But we can now try different value of the P gain Kp.

263/523

Increase P control gain to Kp=10. The model looks as shown in Figure 21.

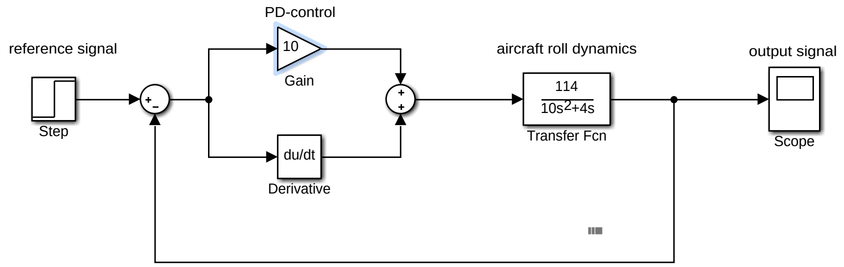


Figure 21

The response of this model with Kp=10 is as shown in Figure 22.

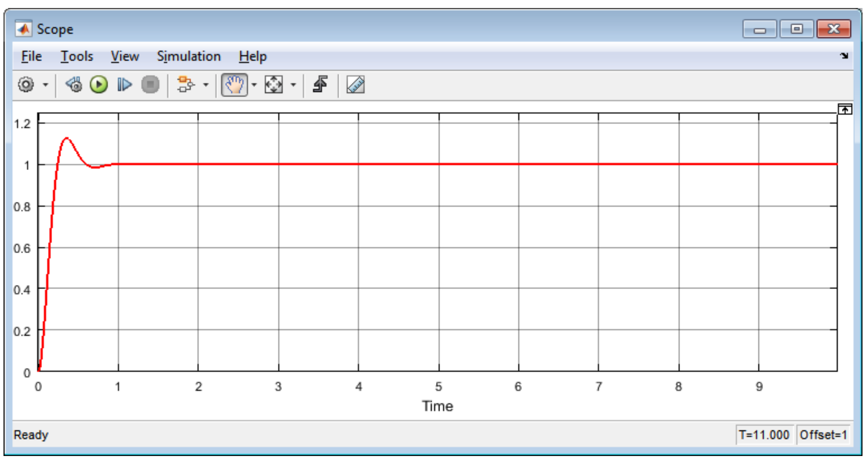


Figure 22

Note that the response has become much crispier but overshoot is now present.

Press the ‘Cursor Measurements’ button to get measurements (Figure 23).

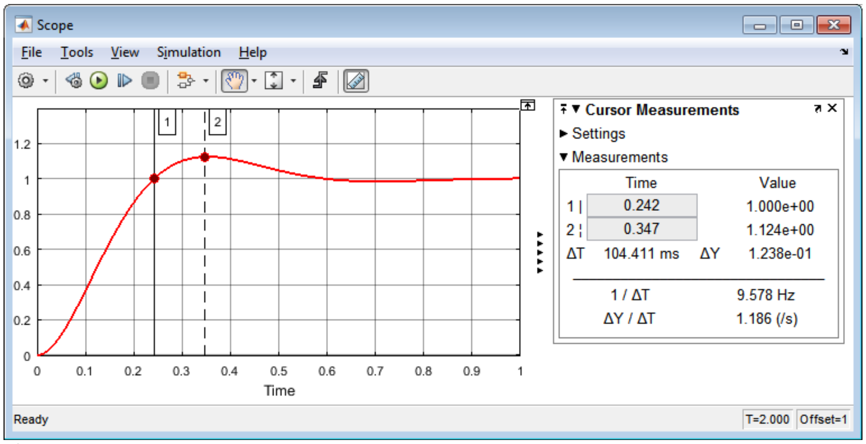


Figure 23

The cursor measurements allow us to determine $t_r = 0.242$ sec and $M_p = 12.4\%$.

<u>Conclusion</u>:
PD control with Kp=10 seems to give a reasonable response:

- short rise time $t_r = 0.242$ sec
- small overshoot $M_p = 12.4\%$

Note of caution: I cannot reproduce these results exactly in MATLAB control systems toolbox. It is perhaps because PD control is not a stable control. In practice, the D part of a PD controller is difficult to implement because differentiation is not easy to obtain with electrical system; therefore, the D part of the controller is implemented as a compensator $s/(T_1 s+1)$.

7 PID CONTROL

PID control stands for 'proportional + integrative + derivative control'.

7.1 PID Control Setup

The aircraft model with PID control has two branches, uses the PID block which can be found in 'Continuous' part of the library. The PI controlled aircraft model looks as shown in Figure 24.

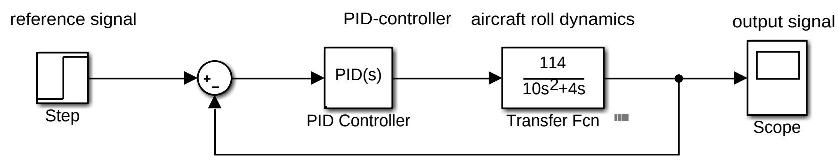


Figure 24

Construct this model and save it as 'SIMULINK_aircraft_roll_PID_control'.
The PID block represents the following controller transfer function:

$$G_c(s) = K_P + \frac{K_I}{s} + K_D s$$

The default values at startup are $K_p = 1,\ K_I = 1,\ K_D = 0$. The response with these default values is as shown in Figure 25.

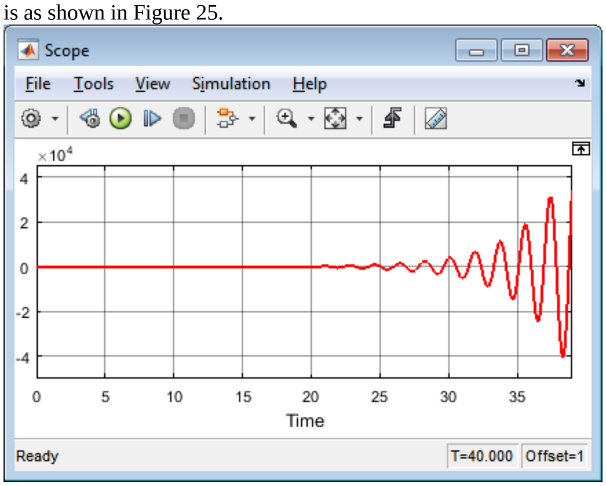


Figure 25

This is an unstable response and needs to be corrected! We will adjust the PID controller parameters to obtain a stable response which meets the design specifications. This process of adjustment is called 'tuning'.

7.2 AUTOMATIC TUNING OF THE PID CONTROLLER

Double click the PID box and open it (Figure 26).

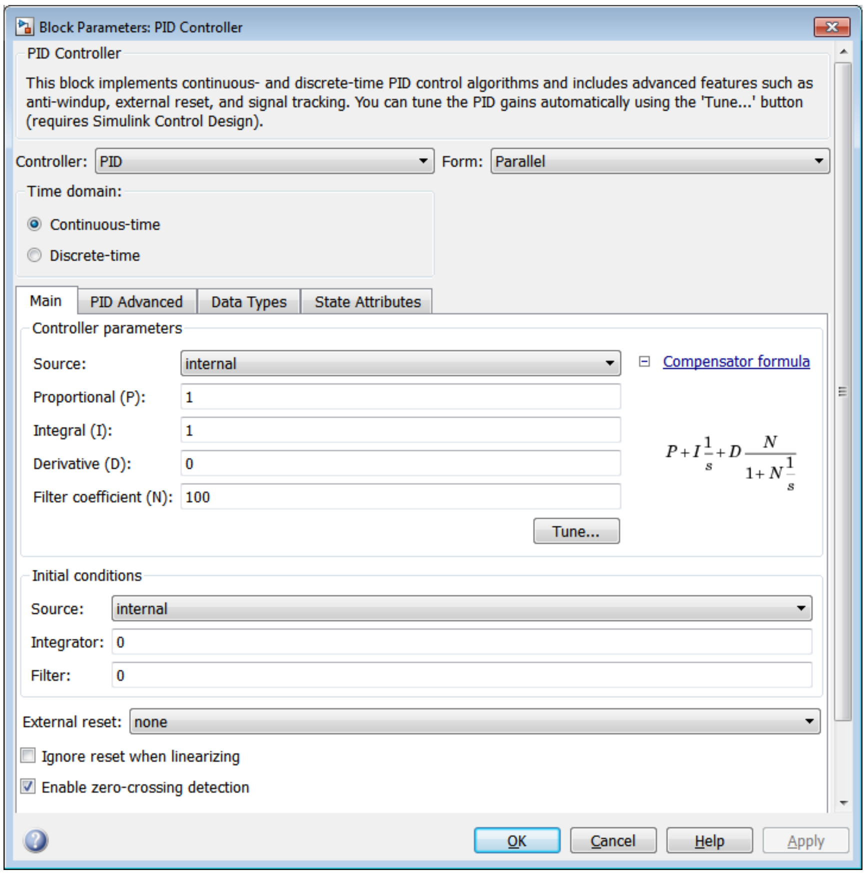


Figure 26

Next, press the ‘Tune…’ button. This will open a separate window for tuning, i.e., the ‘PID Tuner’ window shown in Figure 27.

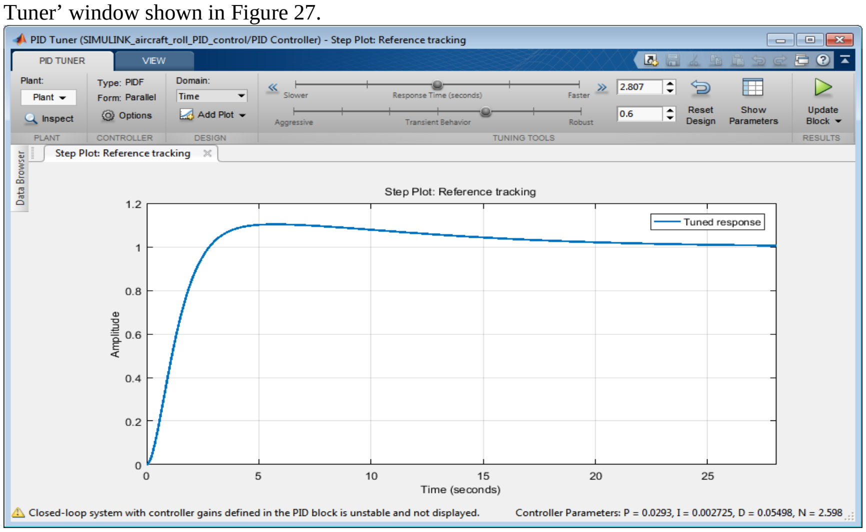


Figure 27

Note the message at the bottom of the window in the left corner: ‘Closed-loop system with controller gains defined in the PID block is unstable and not displayed’.

In the right bottom corner, we read: ‘Controller Parameters: P=0.0293, I=0.002725, D=0.05498, N=2.598’. These controller parameters have been obtained by the tuner through an internal algorithm.

Press the ‘Show parameters’ button in the upper right corner to open Figure 28.

Controller Parameters

	Tuned
P	0.029305
I	0.0027248
D	0.054976
N	2.598

Performance and Robustness

	Tuned
Rise time	1.85 seconds
Settling time	20.3 seconds
Overshoot	10.3 %
Peak	1.1
Gain margin	-Inf dB @ 0 rad/s
Phase margin	69 deg @ 0.712 rad/s
Closed-loop stability	Stable

Figure 28

In this window, we read Rise time = 1.85 sec and Overshoot = 10.3%. Note that MATLAB defines the rise time in the 10%--90% range To evaluate the 0--100% rise time t_r,. read the upper-right side window next the ‘Response Time (seconds) Faster’. Here, we read $t_r = 2.807$ sec.

It is apparent that the overshoot is within specifications but the rise time is to long. Need to do some tuning adjustments manually.

7.3 MANUAL ADJUSTMENT OF THE PID TUNER

To tune the system manually, use the ‘Response time (seconds)’ slider. When the slider is moved to the right, the response time becomes faster.

Under ‘Options’, check the box ‘Show Block Response’ to display the original response (‘Block response’) besides the ‘Tuned response’.

Move the slider until the time t_r shown in the small window on the right is less than 1.5 sec (Figure 29).

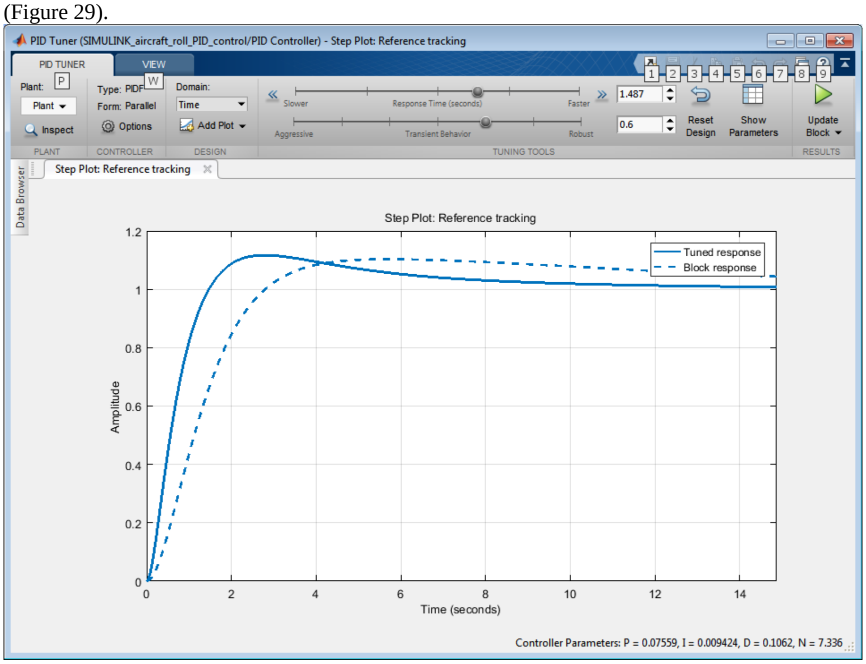
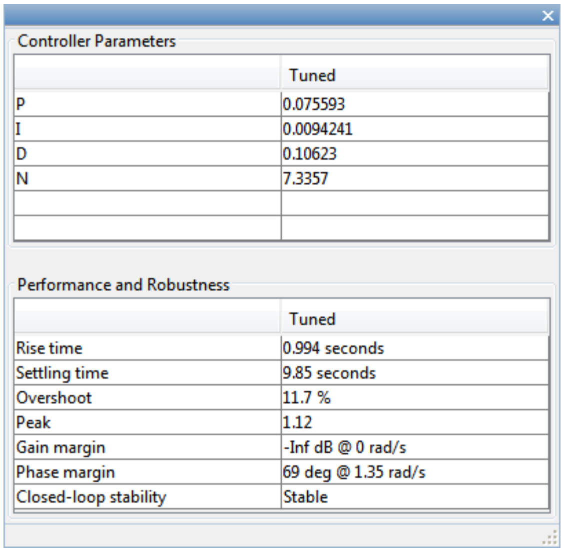


Figure 29

We read $t_r = 1.487$ sec

At this moment, the ‘Block Parameters PID’ window looks as shown in Figure 30.



Controller Parameters

	Tuned
P	0.075593
I	0.0094241
D	0.10623
N	7.3357

Performance and Robustness

	Tuned
Rise time	0.994 seconds
Settling time	9.85 seconds
Overshoot	11.7 %
Peak	1.12
Gain margin	-Inf dB @ 0 rad/s
Phase margin	69 deg @ 1.35 rad/s
Closed-loop stability	Stable

Figure 30

In this window, we read ‘Overshoot = 11.7%’ which means $M_p = 11.7\%$ which is less than the required 20%.

It is now apparent that the design specification have been met. The final PID settings are read from the ‘Block Parameters PID’ window as:

- ‘P= 0.075593’, i.e., Kp = 0.075593
- ‘I = 0.0094241’, i.e., Ki = 0.0094241
- ‘D = 0.10623’, Kd = 0.10623
- N = 7.3357

The aircraft roll response now meets the design specifications, i.e.,

- DS1: Fast response time as measured by rise time
  $t_r = 1.487 < 1.5$ sec
- DS2: maximum percentage overshoot for step input less than 20%
  $M_p = 11.7 < 20\%$

7.4 VERIFICATION OF THE PID TUNING PROCESS

In the ‘PID Tuner’ window, press the ‘Update Block’ green arrow. Look in the ‘Block Parameters: PID Controller’ window to verify that the PID parameters have been updated. The window should look like Figure 31.

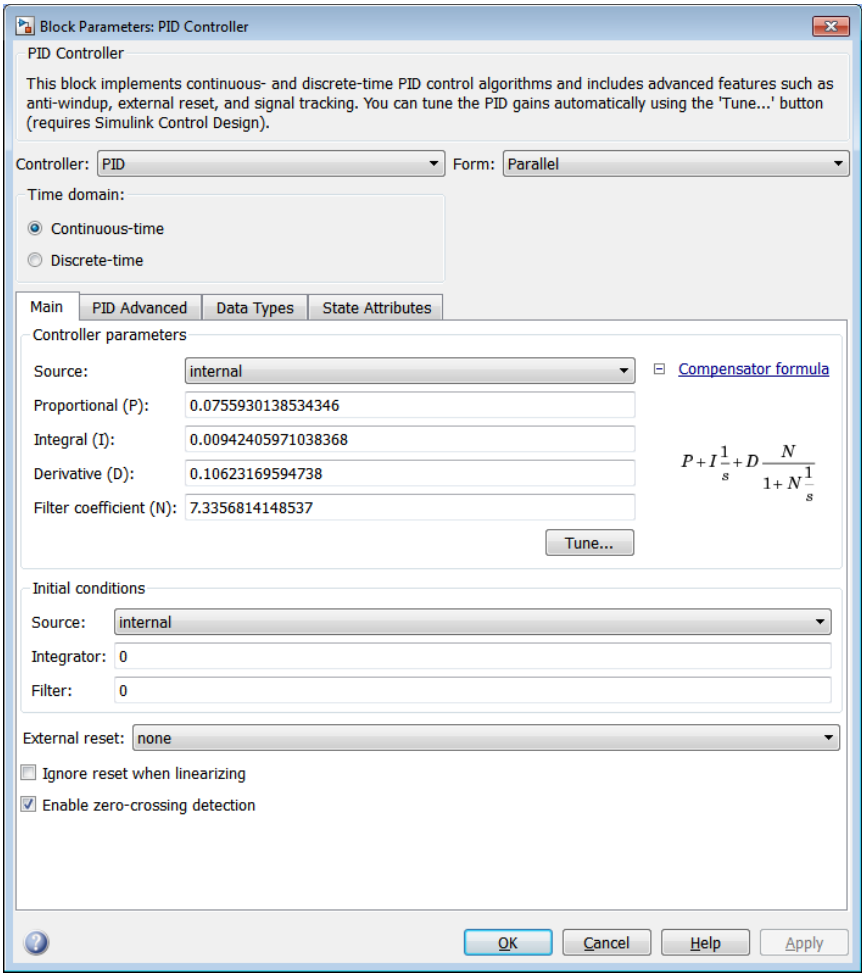


Figure 31

271/523

Now, reduce the SIMULINK run time to 20 sec and run the model (Figure 32).

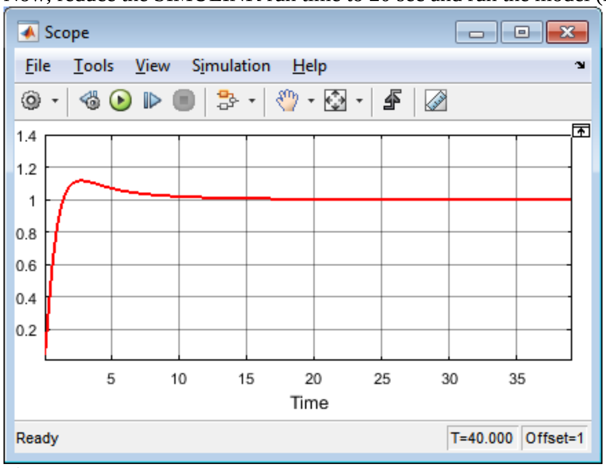


Figure 32

Press the 'Cursor Measurements' button. The window should look like Figure 33.

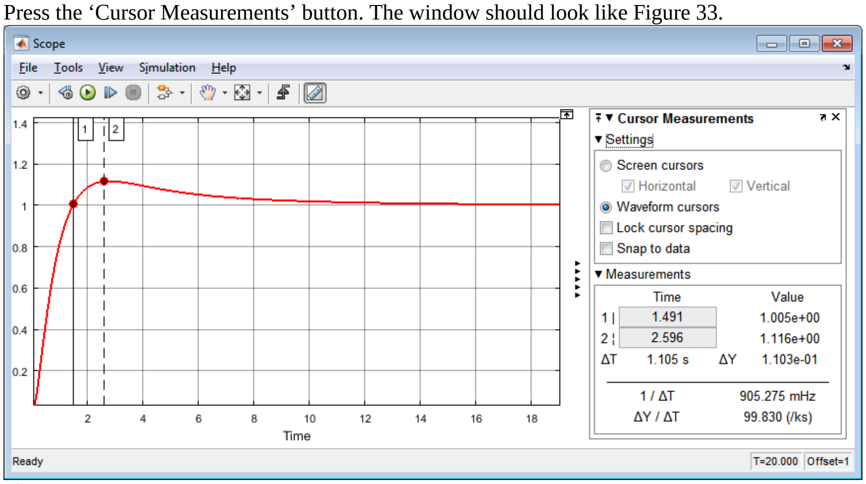


Figure 33

Measure the rise time and overshoot.

- $t_r = 1.483 \le 1.5$ sec
- $M_p = 11.6 \le 20\%$

It apparent that the two design specifications are satisfied.

To verify the MATLAB 10%--90% rise time of 0.994 seconds indicated in the previously shown ‘Controller Parameters’ window, try to measure this rise time in the ‘Scope’ window.

To do this measurement, we need higher resolution; hence, reduce the run time to 5 sec and run again. The Scope window should look like Figure 34.

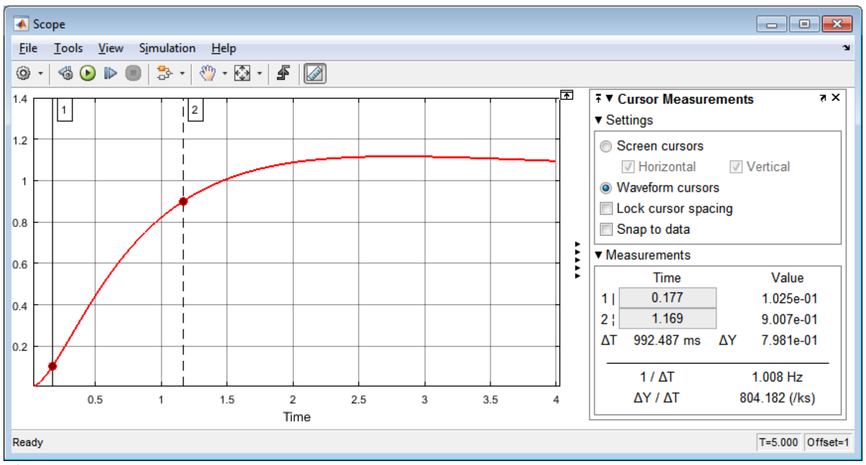


Figure 34

The cursors 1 and 2 are used to identify the 10% and 90% points on the curve. To determine the rise time from this image, read the time difference between the two cursors, i.e., $\Delta T = 992.487$ ms. This is the rise time between 10% and 90%. It agrees with the 0.994 seconds indicated in the previously shown ‘Controller Parameters’ window.

7.8 Feedback Controllers (continued)

PID

(See SIMULINK first)

PID Control

$K_p = 0.094$

$K_i = 0.0094241$

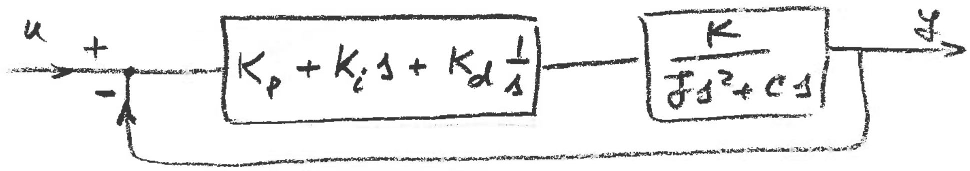
$K_d = 0.10623$

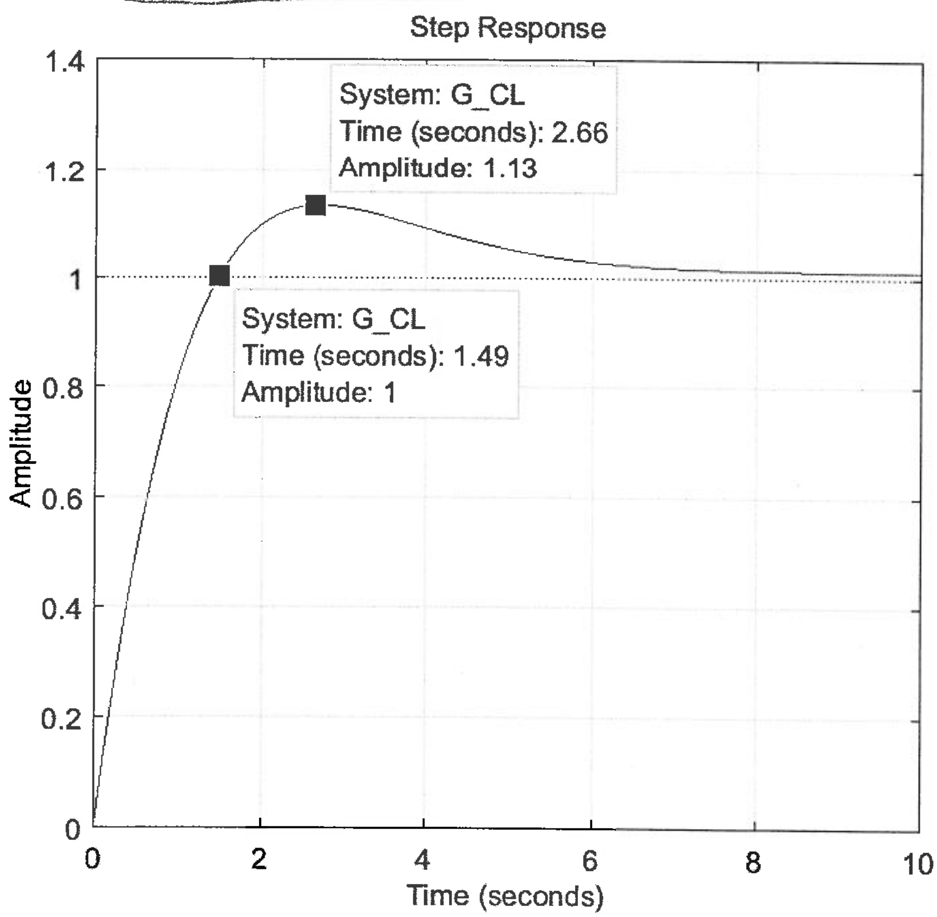
Aircraft model

$K = 114$

$J = 10$

$c = 4$



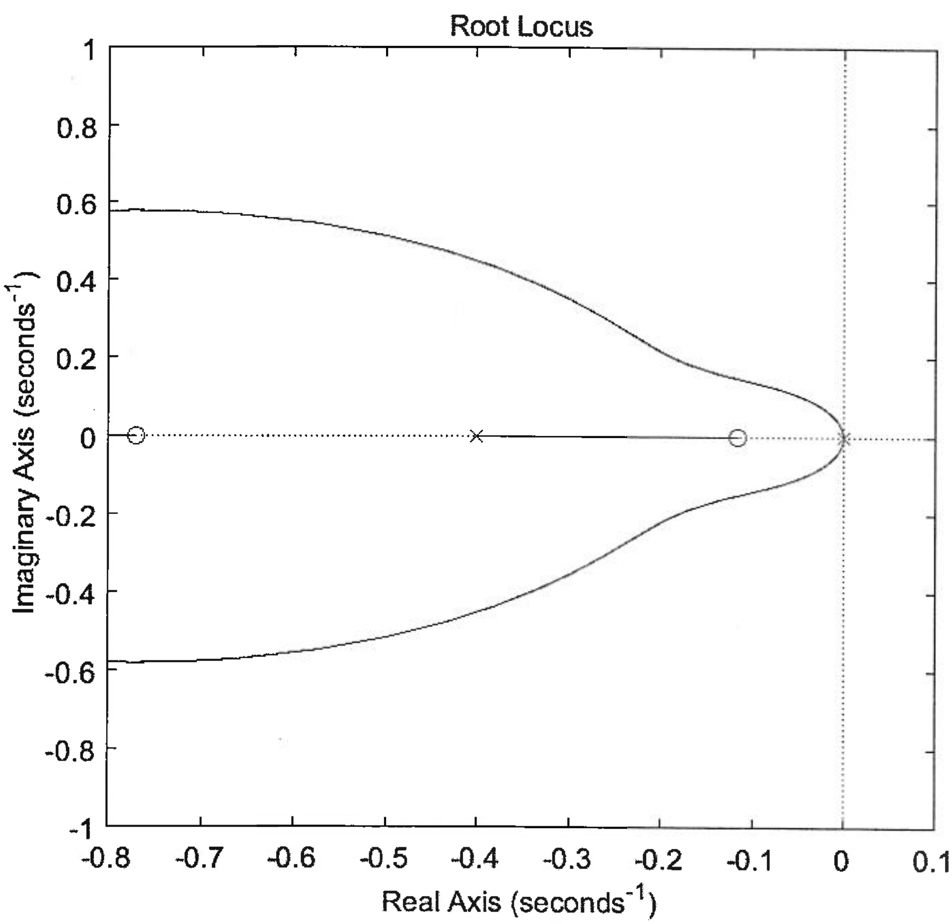


274/523

2
PID

PID control

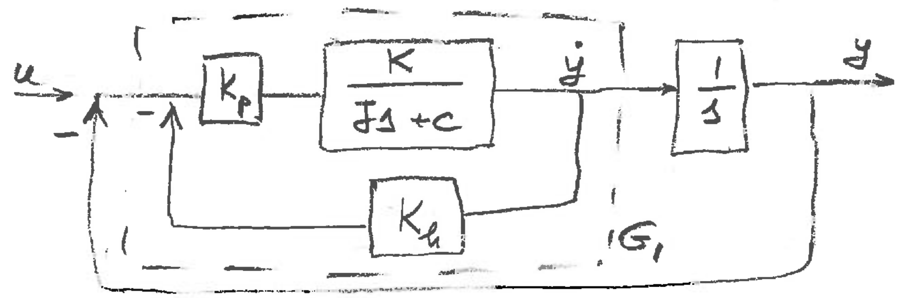
Root locus



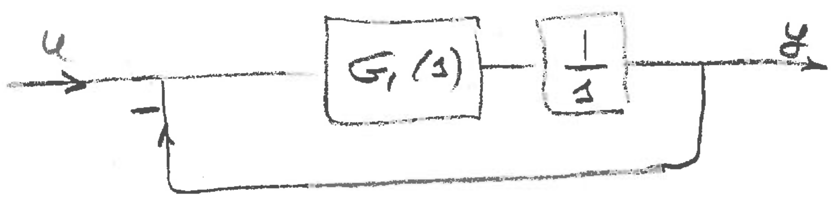
275/523

Velocity Feedback P-Controller

$$G(s) = \frac{K}{Js^2+cs} = \frac{K}{Js+c}\cdot\frac{1}{s}$$



$$G_1(s) = \frac{\frac{K_pK}{Js+c}}{1+\frac{K_pK}{Js+c}K_h} = \frac{K_pK}{Js+c+K_pKK_h} \qquad (1)$$



$$G_2 = G_1(s)\frac{1}{s} = \frac{K_pK}{Js^2+(c+K_pKK_h)s} \qquad (2)$$

$$G_{CL} = \frac{G_2}{1+G_2} = \frac{K_pK}{Js^2+\underbrace{(c+K_pKK_h)}_{c^*}s+K_pK}$$

$$G_{CL} = \frac{K_pK}{Js^2+c^*s+K_pK} \qquad (3)$$

2

$$c^* = c + K_p K K_h \qquad (4)$$

$$G_{CL} = \frac{\frac{K_p K}{J}}{s^2 + \frac{c^*}{J}s + \frac{K_p K}{J}} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

$$\omega_n^2 = \frac{K_p K}{J} \longrightarrow \omega_n = \sqrt{\frac{K_p K}{J}} \qquad (5)$$

$$2\zeta\omega_n = \frac{c^*}{J} \rightarrow \zeta = \frac{c^*}{2\omega_n J} = \frac{c^*}{2\sqrt{K_p K J}}$$

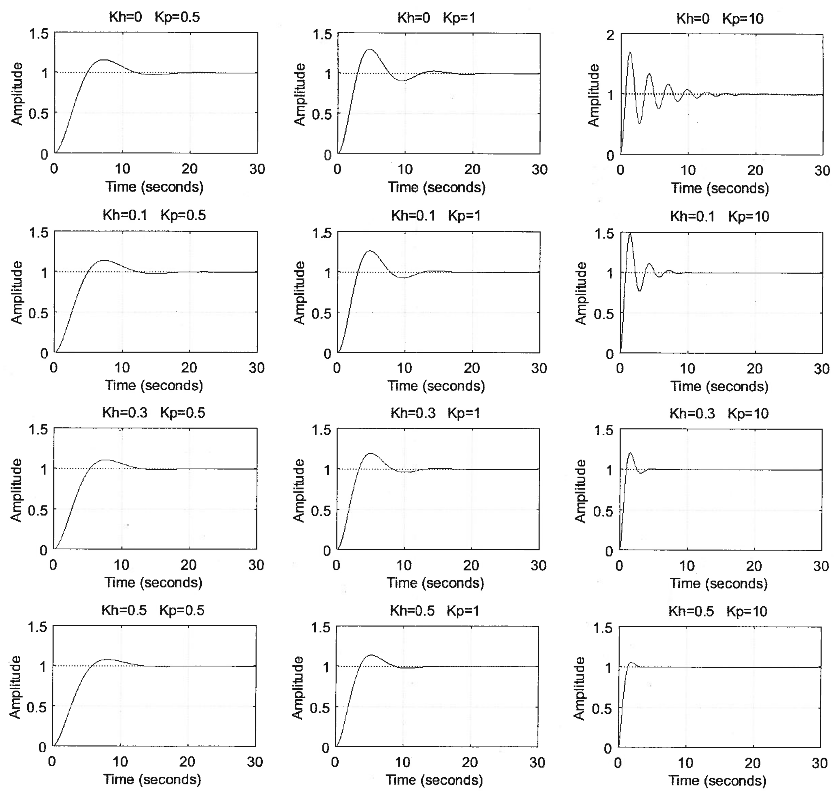
$$\zeta = \frac{c + K_p K K_h}{2\sqrt{K_p K J}}$$

- Velocity feedback gain K_h increases ζ (damping)
- P-control gain K_p modifies both freq. ω_n and damping ζ

Strategy

- modify frequency with K_p
- add more damping with K_h

277/523



class example.

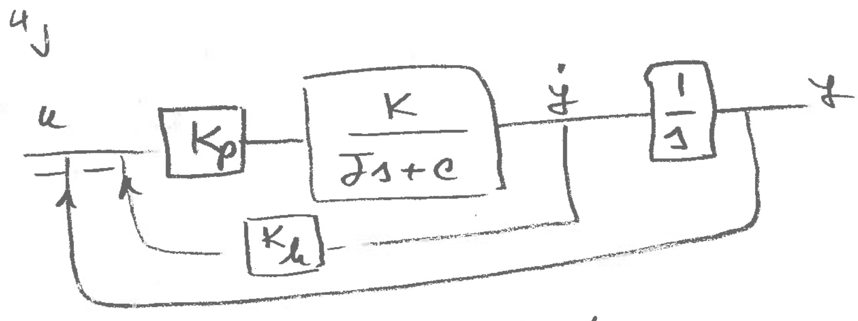
$K=1$

$J=2$

$c=1$

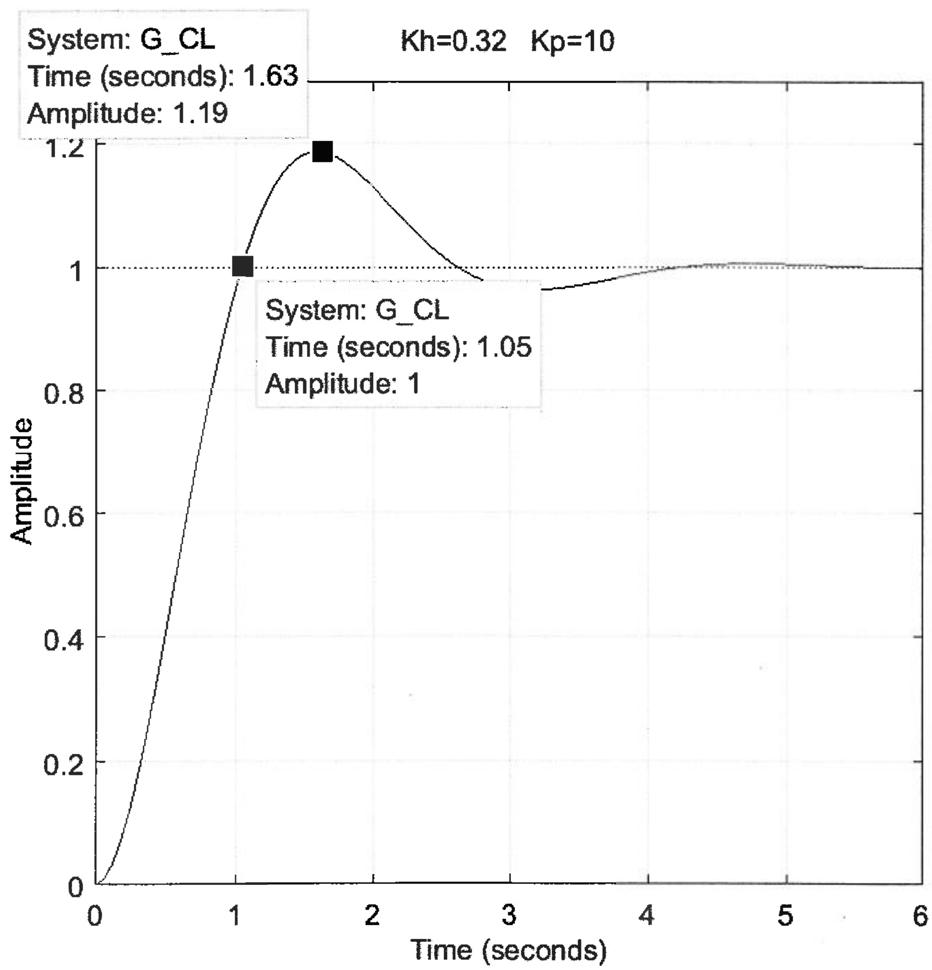
$\frac{K}{Js+c}\cdot\frac{1}{s}$

278/523

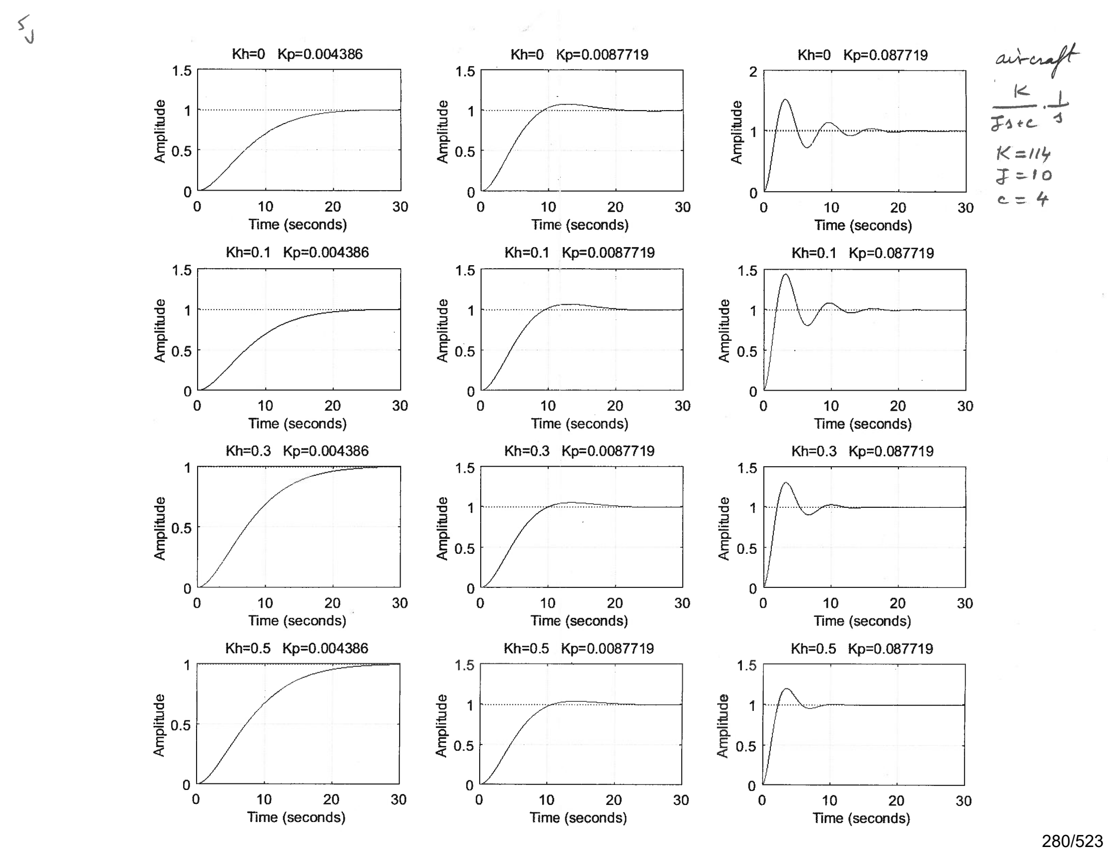


$K = 1$
$J = 2$
$c = 1$

class example



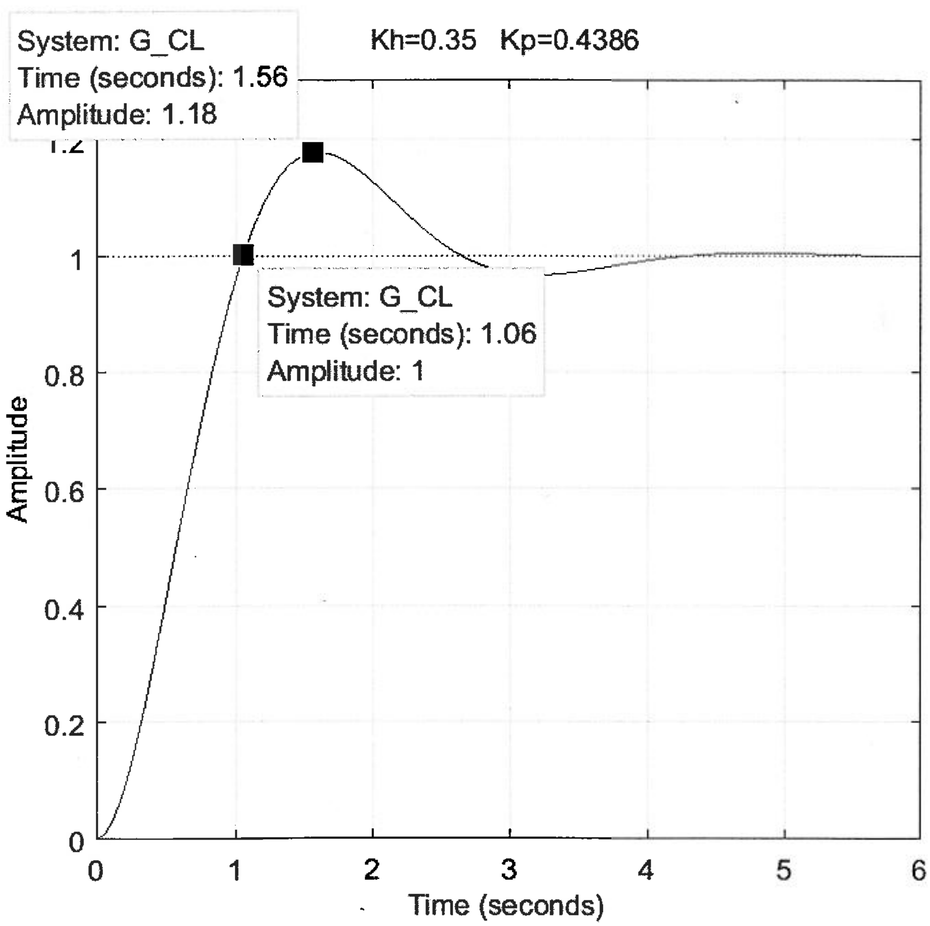
279/523


Kh=0 Kp=0.004386
Kh=0 Kp=0.0087719
Kh=0 Kp=0.087719
Kh=0.1 Kp=0.004386
Kh=0.1 Kp=0.0087719
Kh=0.1 Kp=0.087719
Kh=0.3 Kp=0.004386
Kh=0.3 Kp=0.0087719
Kh=0.3 Kp=0.087719
Kh=0.5 Kp=0.004386
Kh=0.5 Kp=0.0087719
Kh=0.5 Kp=0.087719
Amplitude
Time (seconds)
aircraft
$\frac{K}{Js+c} \cdot \frac{1}{s}$
K = 114
J = 10
c = 4
280/523

6)

velocity feedback $\frac{K}{Js+C} \cdot \frac{1}{s}$

$K = 114$ aircraft
$J = 10$ roll
$C = 4$ model



281/523

7.9 SIMULINK Aircraft Roll Motion (continued)

8 VELOCITY FEEDBACK CONTROL

Velocity feedback control concept consists of two controls:

- P-control with gain Kp
- velocity control with gain Kh

The advantage of this control combination is that one can use the P-control gain Kp to improve the response time by increasing the natural frequency and then use the velocity feedback gain Kh to reduce the overshoot by increasing damping.

The initial values for the model are Kp=1/114, Kh=0. With these values, the SIMULINK model for velocity feedback looks like Figure 35.

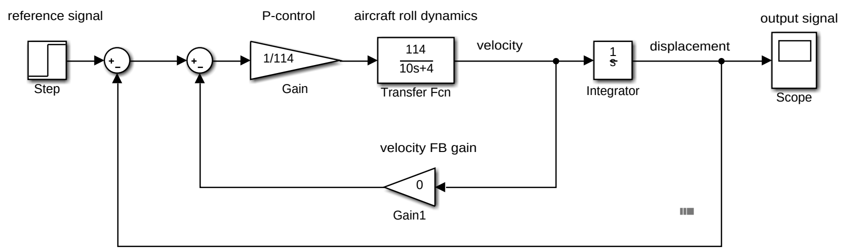


Figure 35

Note: when inserting ‘Gain1’ for velocity FB, use right click to flip the box to point backward.

The aircraft response is sluggish with $t_p \approx 9$ sec, as shown in Figure 36.

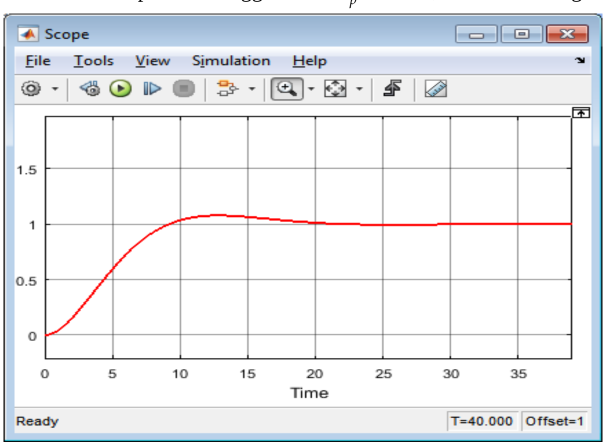


Figure 36

To accelerate the aircraft response, increase the P-control gain to Kp=50/114 (Figure 37).

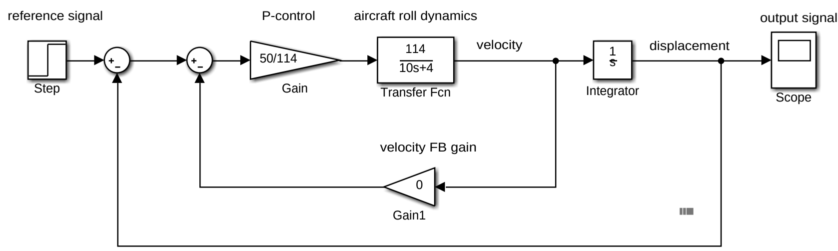


Figure 37

The aircraft response now looks like Figure 38.

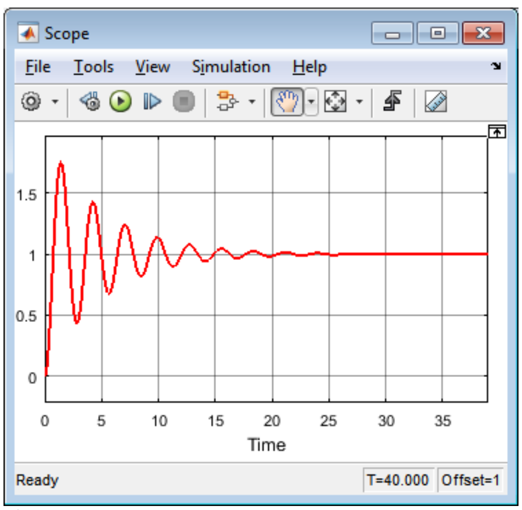


Figure 38

Notice that the aircraft response is much more rapid now. However, it has a large overshoot which must be reduced.

To reduce the overshoot, increase the velocity feedback gain to Kh=0.2 (Figure 39).

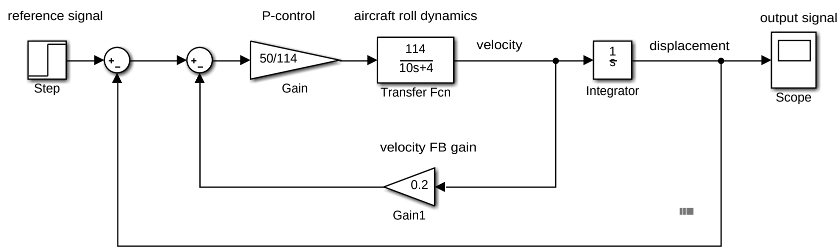


Figure 39

The response has become less oscillatory while remaining relatively fast (Figure 40).

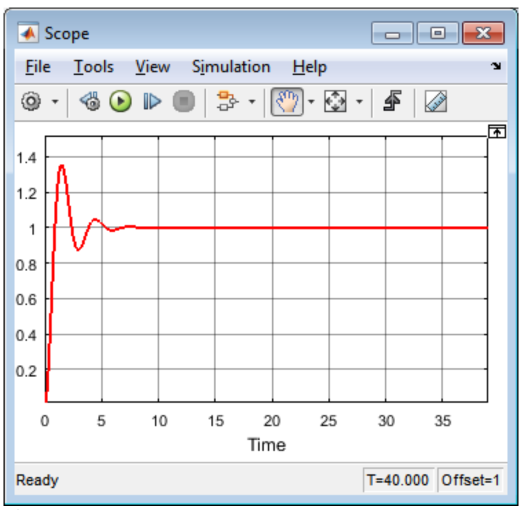


Figure 40

Further adjustments of Kh can ensure that the design specifications are met.

Reduce the computation time to 6 sec in order to expand the initial response zone and get a better reading of rise time and overshoot.

Increase Kh to various values until a satisfactory reduction of overshoot is obtained. For Kh=0.35, we get Figure 41:

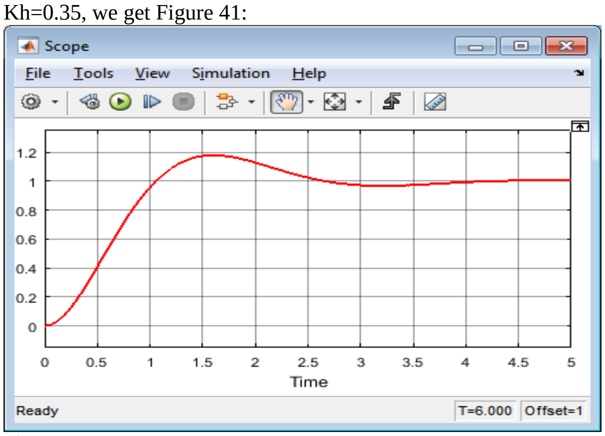


Figure 41

Open the cursors to read the rise time and overshoot(Figure 42).

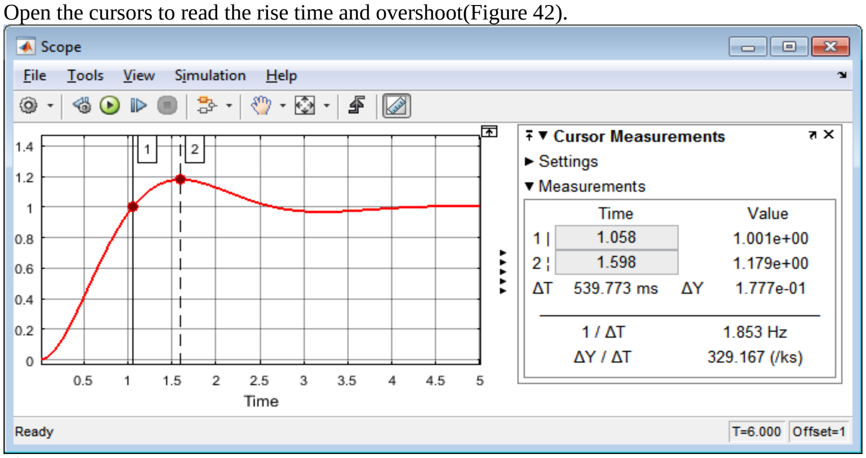


Figure 42

The readings indicate satisfactory results, i.e.,

- $t_r = 1.058 < 1.5$ sec
- $M_p = 17.9 < 20\%$

7.10 Instability Suppression with velocity feedback

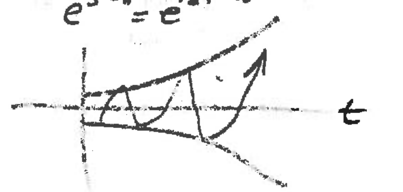
VFB

20160613

INSTABILITY SUPPRESSION WITH VELOCITY FEEDBACK

Assume a 2nd order system with negative damping, $\zeta < 0$. Hence, the response will be unstable:

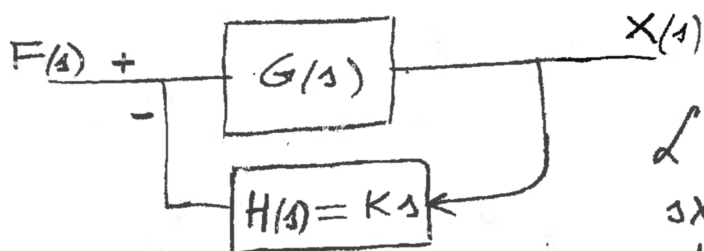
$e^{-\zeta\omega_n t} = e^{|\zeta|\omega_n t}$



$\zeta < 0$

$-\zeta\omega_n t = |\zeta|\omega_n$

We can use FB to suppress this instability.



$\mathcal{L}\,\dot{x} = sX$

sX is the LT of velocity!

$$G_{CL} = \frac{G}{1+GH} = \frac{\dfrac{\omega_n^2}{s^2+2\zeta\omega_n s+\omega_n^2}}{1+\dfrac{\omega_n^2}{s^2+2\zeta\omega_n s+\omega_n^2}Ks}$$

$$= \frac{\omega_n^2}{s^2+(2\zeta\omega_n+K\omega_n^2)s+\omega_n^2} \qquad (1)$$

$$G_{CL} = \frac{\omega_n^2}{s^2+2\zeta_{CL}\omega_n s+\omega_n^2} \qquad (2)$$

286/523

FB4
20181022 (1)&(2):

$$2\zeta_{CL} = 2\zeta + K\omega_n$$

$$\zeta_{CL} = \zeta + \frac{K\omega_n}{2} \qquad (3)$$

1) Velocity feedback increases damping.

2) If the initial system is unstable because it has −ve damping ($\zeta<0$), then velocity feedback can give the CL system a positive damping, $\zeta_{CL}>0$

Solve Eq. (3) to get

$$K = \frac{2}{\omega_n}(\zeta_{CL} - \zeta) \qquad (4)$$

Critical gain K_{cr} is the gain that makes $\zeta_{CL}=0$.

Recall Eq (3) and set it to zero to get

$$\zeta + \frac{K\omega_n}{2} = 0 \longrightarrow K_{cr} = -\frac{2\zeta}{\omega_n} \qquad (5)$$

Example:

$\zeta = -5\% = -0.05$ $\qquad f_n = 4$ Hz

$\omega_n = 2\pi \times 4 = 8\pi$ rad/s.

$$K_{cr} = -\frac{2\times(-0.05)}{8\pi} \cong 0.004$$

For $K > K_{cr}$ the system is stable

287/523 47

FB4a
20181022

Calculation of FB gain K

Recall (4) $$K = \frac{2}{\omega_n}(\zeta_{CL} - \zeta)$$

Factor out $-\zeta$ to get $$K = -\frac{2\zeta}{\omega_n}\left(1 - \frac{\zeta_{CL}}{\zeta}\right)$$

Recall (5) and write

$$K = K_{cr}\left(1 - \frac{\zeta_{CL}}{\zeta}\right) \qquad (6)$$

Define ζ_{ratio} as

$$\zeta_{ratio} = \frac{\zeta_{CL}}{\zeta} \qquad (7)$$

(7) → (6): $$K = K_{cr}(1 - \zeta_{ratio}) \qquad (8)$$

Eq(8) gives the FB gain K as a function of ζ_{ratio}

Gain ratio K_{ratio}

Divide Eq(8) by K_{cr} to get

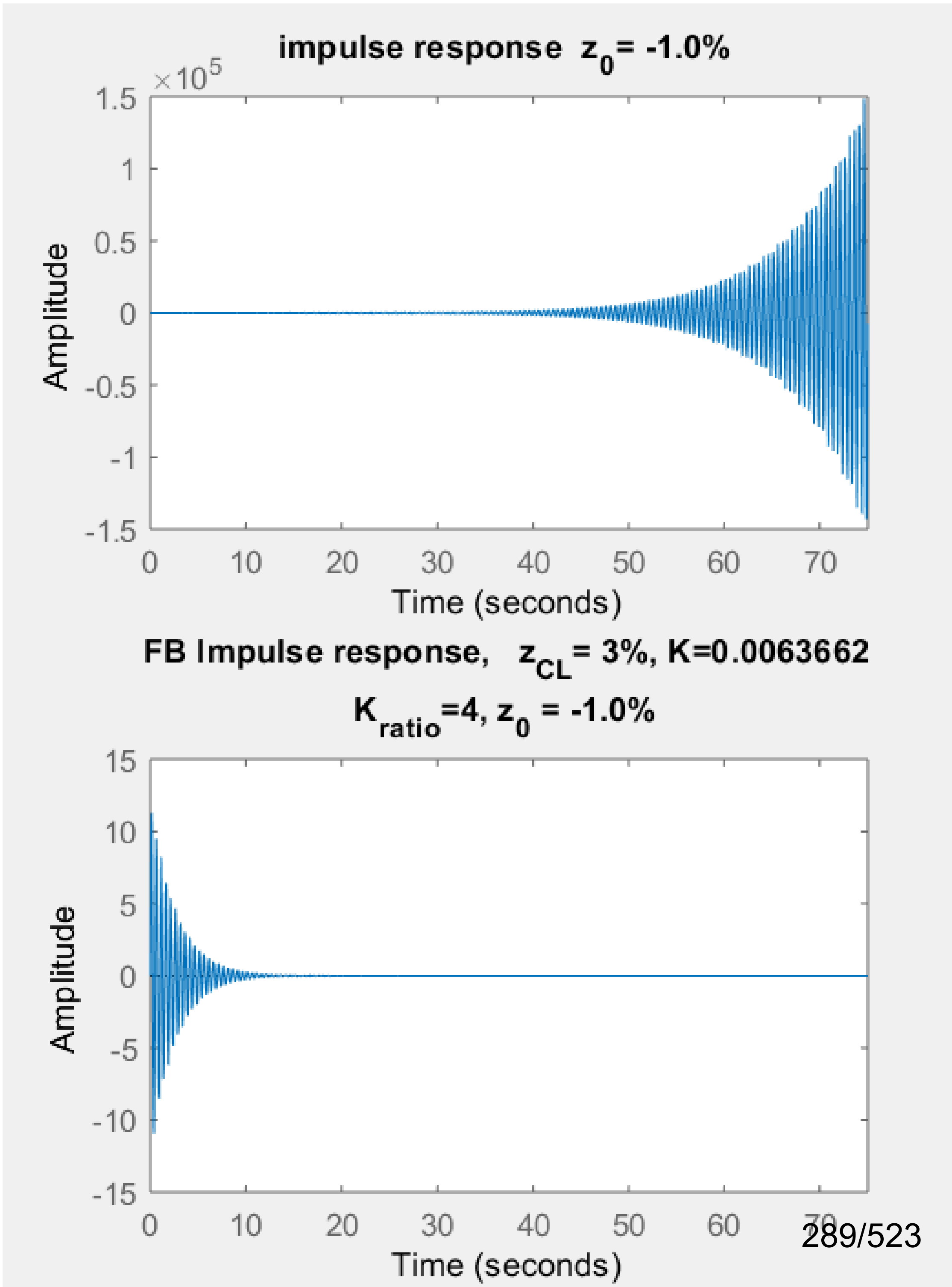
$$K_{ratio} = 1 - \zeta_{ratio} \qquad (9)$$

Discussion of Eq.(8)

(a) if $\zeta < 0$, then $\zeta_{ratio} < 0$, $K_{cr} > 0$ → $\boxed{K > 0}$

(b) if $0 < \zeta < \zeta_{CL}$, then $K_{cr} < 0$, $\zeta_{ratio} > 1$, $(1 - \zeta_{ratio}) < 0$, $\boxed{K > 0}$

(c) $\zeta_{CL} < \zeta$, then $K_{cr} < 0$, $\zeta_{ratio} < 1$, $1 - \zeta_{ratio} < 0$, $\boxed{K < 0}$ 288/523
reduce ζ_{CL}


impulse response z_0= -1.0%
×10^5
1.5
1
0.5
0
-0.5
-1
-1.5
Amplitude
0 10 20 30 40 50 60 70
Time (seconds)
FB Impulse response, z_{CL}= 3%, K=0.0063662
K_{ratio}=4, z_0 = -1.0%
15
10
5
0
-5
-10
-15
Amplitude
0 10 20 30 40 50 60 70
Time (seconds)
289/523