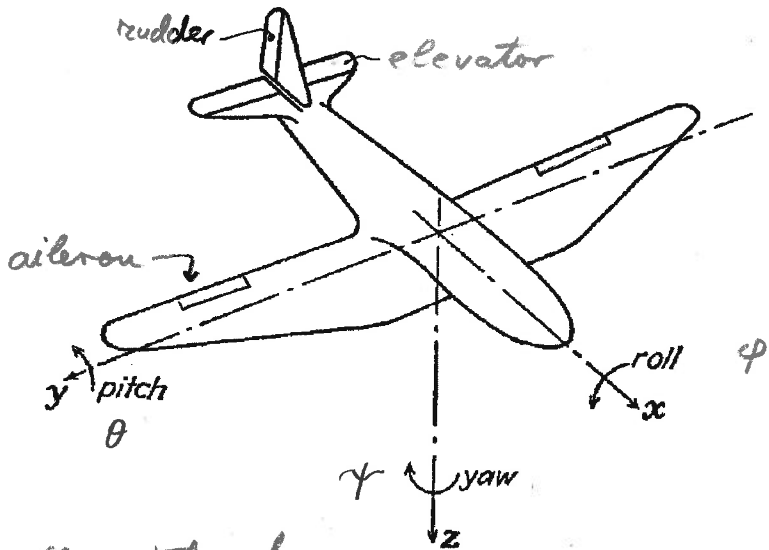
AIRCRAFT CONTROLS

2016 11 03



Aircraft control surfaces

AILERONS control the rolling motion
φ = roll angle (~ bank angle)
pilot stick moves left-right to deflect ailerons

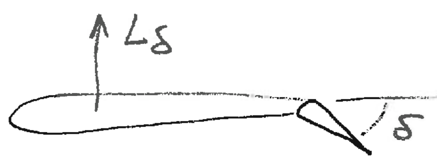
ELEVATOR controls the pitch motion
nose up / nose down
θ = pitch angle
pilot stick moves forward - backward to deflect elevator

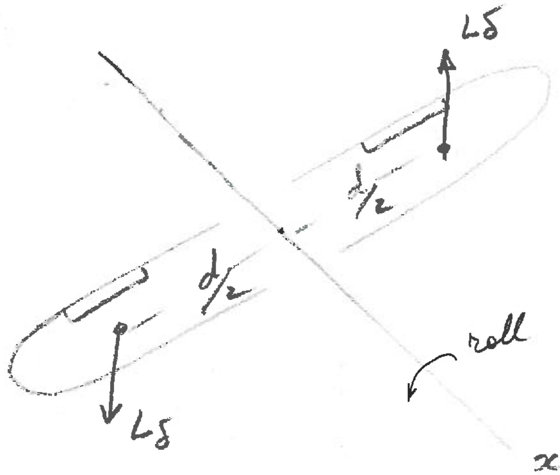
RUDDER controls yaw motion
ψ = yaw angle
pilot uses rudder pedals to deflect rudder

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,

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

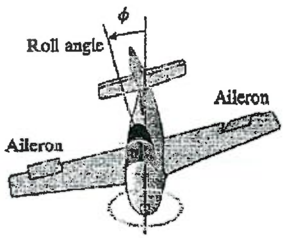
- Essentially, the rolling moment M is proportional to aileron deflection δ, i.e.

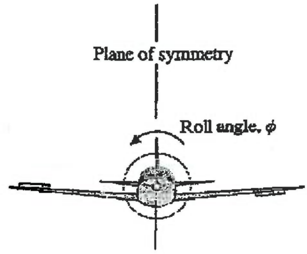
$$M = K\,\delta \qquad (K = \text{gain}) \qquad (2)$$

1

RTF

ROLL TRANSFER FUNCTION





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

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:

$$M = K\delta \qquad (2)$$

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

Take LT of Eq. (3) to get

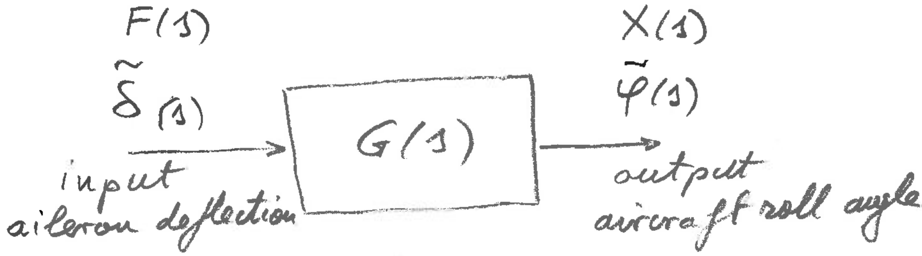
$\mathcal{L}(3)$: $\quad (Js^2 + cs)\tilde{\varphi}(s) = K\tilde{\delta}(s) \qquad (4)$

where $\quad \tilde{\varphi}(s) = \mathcal{L}\varphi(t)$

$\qquad\quad \tilde{\delta}(s) = \mathcal{L}\delta(t) \qquad (5)$

Solution of Eq. (4) yields:

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



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

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

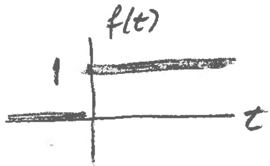
The system described by Eq. (7) is an uncontrolled 2nd order dynamic system.

- 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

1 RR 20161109

Roll Response to Constant Aileron Deflection

Assume the pilot moves the stick laterally such as to create a constant aileron deflection δ = const. This corresponds to a step input, i.e.,



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

(9) → (8):

$$X(s) = \frac{K}{Js^2 + cs} \cdot \frac{1}{s} = \frac{K}{s^2(Js+c)} \qquad (10)$$

Table 2-1, # 19 has the pair

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

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

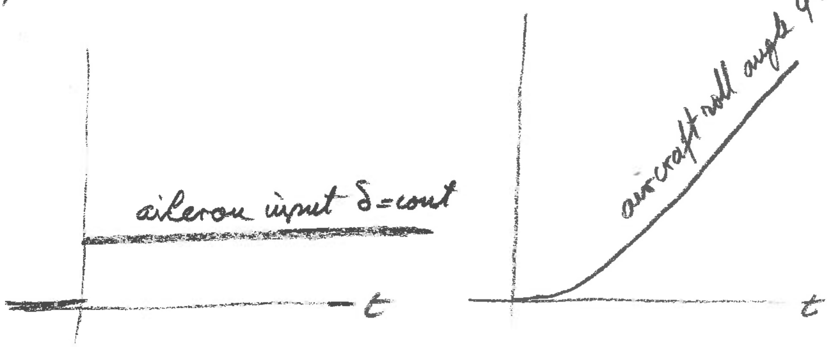
$$X(s) = \frac{K}{c} \cdot \frac{1}{s^2(Ts+1)}, \quad T = \frac{J}{c} \qquad (12)$$

ILT of (12) gives:

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

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

2 RR 2016 11 09



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

Thus, a constant aileron input $\delta = \text{const}$ produces continuosly 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 continuosly under constant voltage input
- Aircraft rolls continuously under constant aileron input

3 RR 2016 11 09

Step response of Type 1 systems

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

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

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

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

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

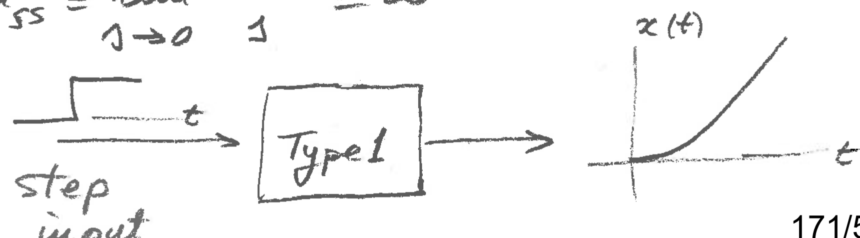
Steady state response is calculated with Final Value Theorem, i.e.,

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

(1) → (2):

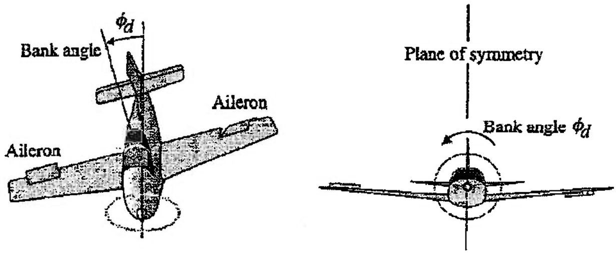
$$x_{ss} = \lim_{s\to 0} \cancel{s}\,\frac{K}{s^{\cancel{2}}}\,\frac{\overbrace{(T_a s+1)(T_b s+1)\cdots}^{\xrightarrow{s\to 0} 1}}{\underbrace{(T_1 s+1)(T_2 s+1)\cdots}_{\longrightarrow 1}}$$

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



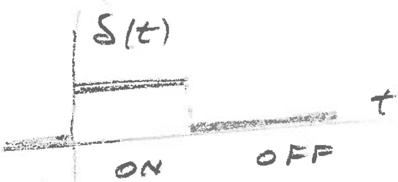
BAC

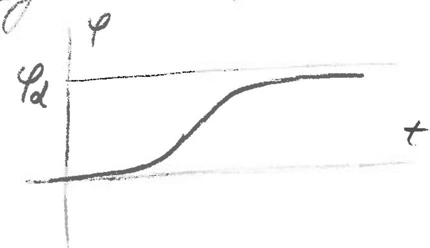
BANK ANGLE CONTROL



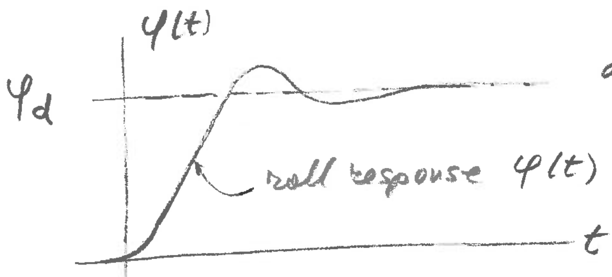
We want the aircraft to roll from flying straight & level to flying inclined with a bank angle φ_d = const.

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

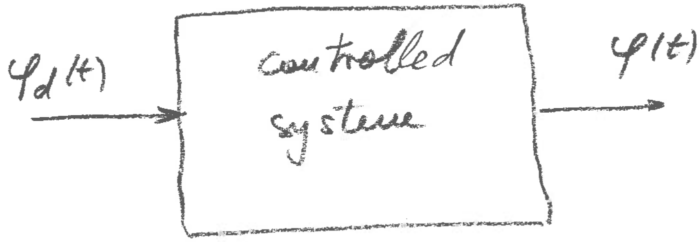




- We desire to build a FB control system to achieve this transition in a smooth way <u>automatically</u>.



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%

$$M_p \leq 20\%$$

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

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 $\rightarrow 0$).

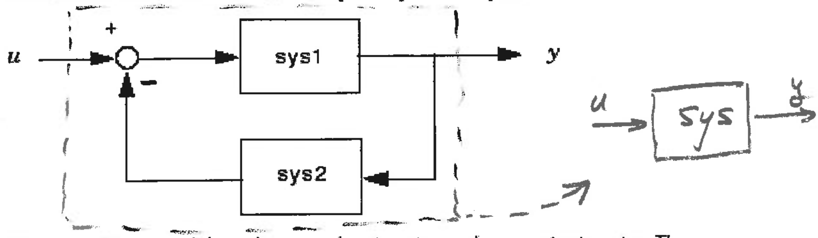
The process is based on Convolution Theorem of Laplace Transform

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

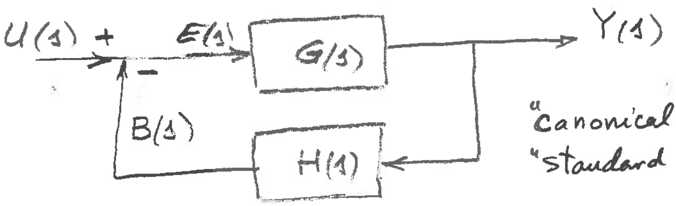
2 FB 2016 11 10

MATLAB:

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

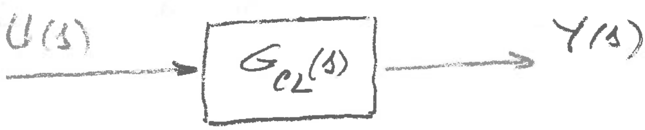


The closed-loop model `sys` has u as input vector and y as output vector. The models `sys1` and `sys2` must be both continuous or both discrete with identical sample times. Precedence rules are used to determine the resulting model type



FB sys.

"canonical form"
"standard form"



Equivalent FB sys.

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

closed loop transfer funct.

<u>Proof</u>

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

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

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

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

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

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

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

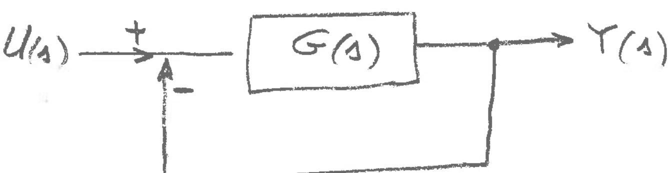
Transfer functions

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

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

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

UNIT FEEDBACK



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

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

unit feedback closed loop TF.

feedback

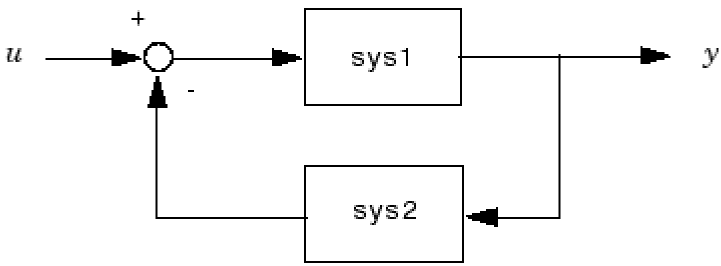
Feedback connection of two models

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

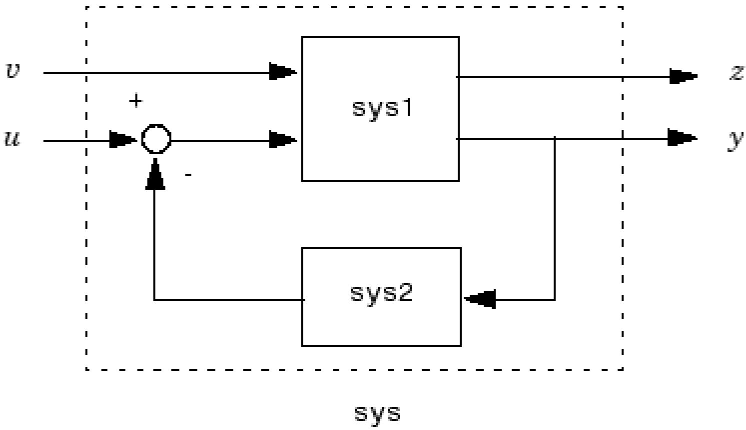
```
sys = feedback(sys1,sys2,+1)
```

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

Finally,

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

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

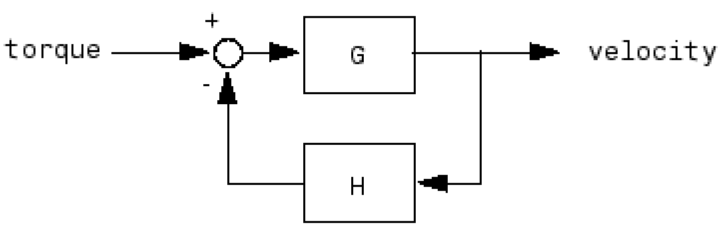
```
sys = feedback(sys1,sys2,feedin,feedout,+1)
```

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`.

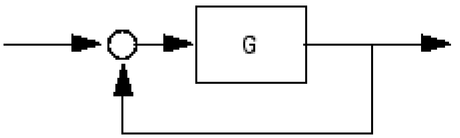
Example 2

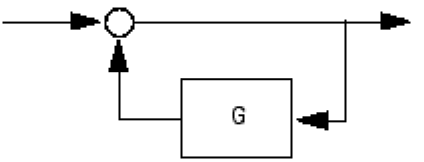
Consider a state-space plant `P` with five inputs and four outputs and a state-space feedback controller `K` with three inputs and two outputs. To connect outputs 1, 3, and 4 of the plant to the controller inputs, and the controller outputs to inputs 4 and 2 of the plant, use

```
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

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

K = gain
J = inertia
c = damping

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

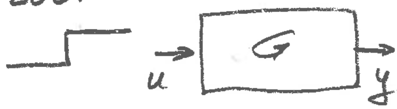
Type 1 system : $G(s) = \frac{K}{s} \cdot \frac{1}{Js + c}$

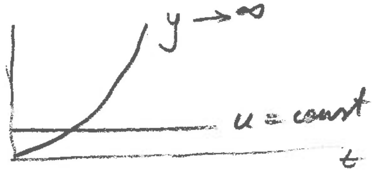
"s to power 1"

FB Step response of Type 1 system

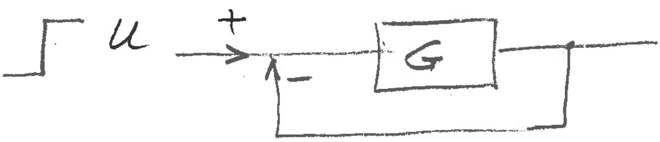
$$G(s) = \frac{K}{Js^2 + cs}$$

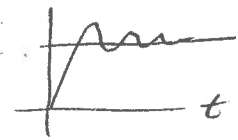
OPEN LOOP





CLOSED LOOP w FB





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

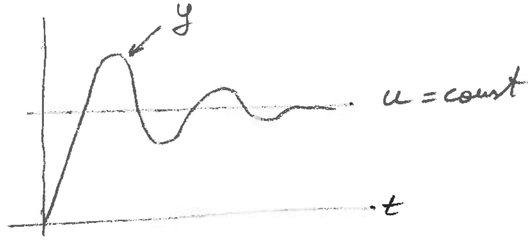
$$= \frac{\frac{K}{Js^2+cs}}{1+\frac{K}{Js^2+cs}} = \frac{K}{Js^2+cs+K} = \frac{\frac{K}{J}}{s^2+\frac{c}{J}s+\frac{K}{J}}$$

$$G_{CL} = \frac{\omega_n^2}{s^2+2\zeta\omega_n s+\omega_n^2}$$

{ 2nd order system
Type 0 system

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

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

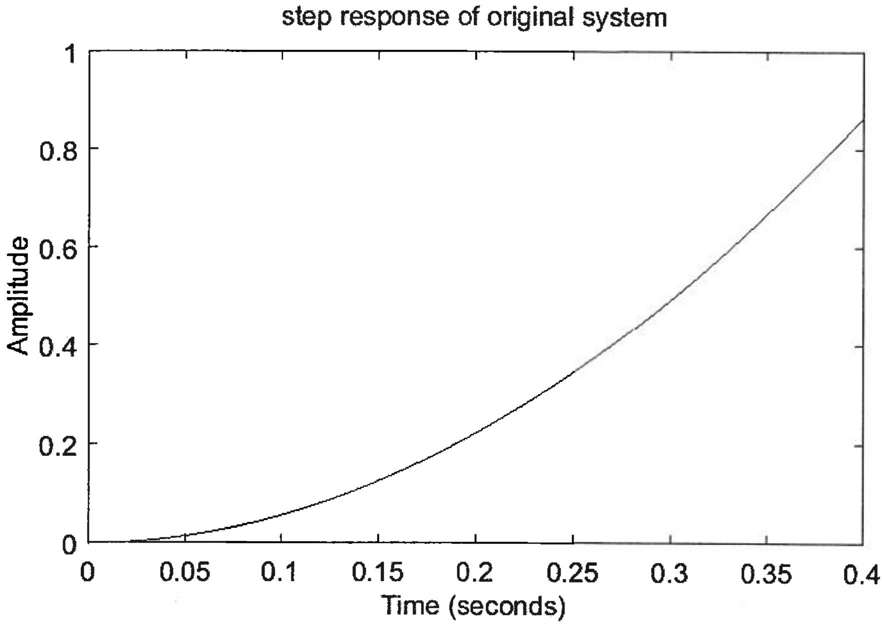


- FB has controlled the Type 1 system
- Type 1 system w FB can hold position

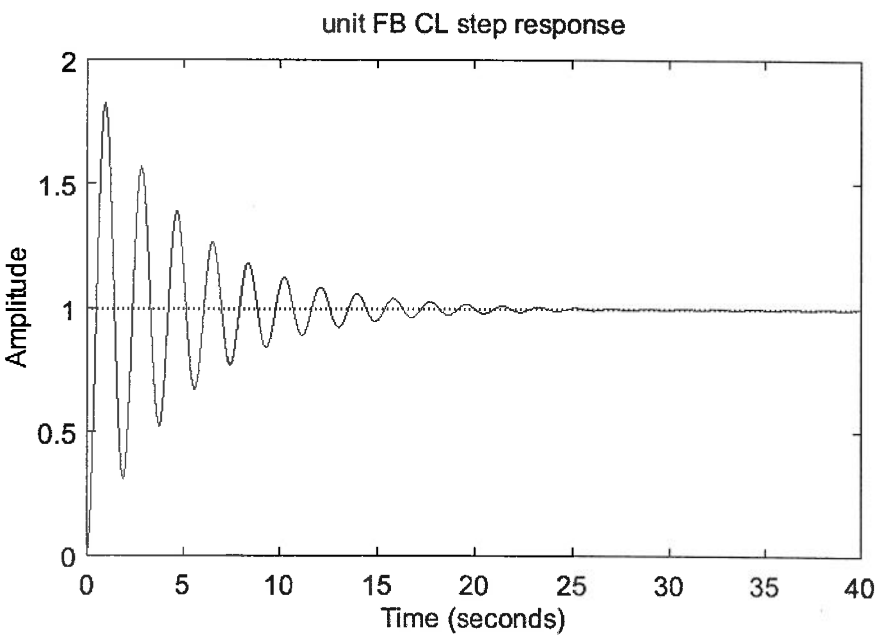
9
FB

```
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
```

10 FB



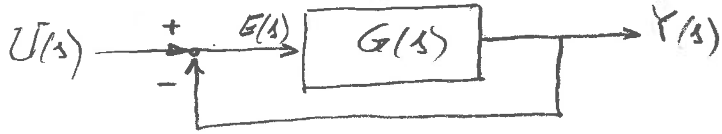
step response of original system
Amplitude
Time (seconds)
0
0.2
0.4
0.6
0.8
1
0
0.05
0.1
0.15
0.2
0.25
0.3
0.35
0.4

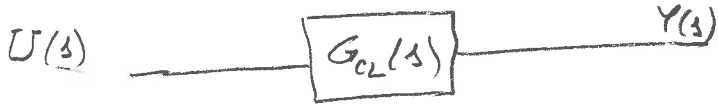


unit FB CL step response
Amplitude
Time (seconds)
0
0.5
1
1.5
2
0
5
10
15
20
25
30
35
40

1
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:

$$E(s) = U(s) - Y(s) \qquad (1)$$

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

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

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

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

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

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

Step error of FB systems

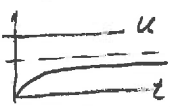
Figure of merit: "static position error constant" $= \lim_{s\to 0} G(s)$

misnomer: in fact, the larger, the better!

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

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

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



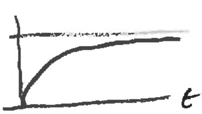
e_{ss}

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

Feedback creates non zero ss error for Type 0 sys!

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

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



$e_{ss} = 0$

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

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

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

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

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

$\boxed{\text{Type 0 sys}}$

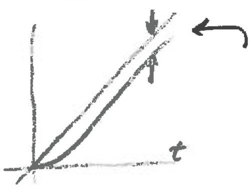
$$s\,G(s) = sK\frac{(T_a s+1)\cdots}{(T_1 s+1)\cdots} \xrightarrow[s\to 0]{} 0$$

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

infinite error! cannot follow!

$\boxed{\text{Type 1 sys}}$

$$s\,G(s) = \cancel{s}\frac{K}{\cancel{s}}\frac{(T_a s+1)\cdots}{(T_1 s+1)\cdots} \xrightarrow[s\to 0]{} K = \text{const}$$



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

ramp offset.

$\boxed{\text{Type 2 sys}}$

$$s\,G(s) = \cancel{s}\frac{K}{s^{\cancel{2}}}\frac{(T_a s+1)\cdots}{(T_1 s+1)\cdots} \xrightarrow[s\to 0]{} \frac{1}{0}$$

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

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

4
FBE
20161202

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