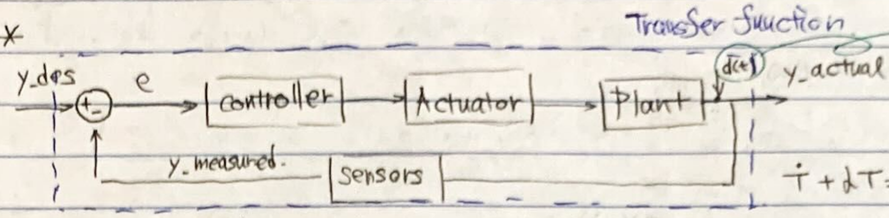
· N. Chbat
→ chbat@quadrusmedical.com
→ 518.505.6425.
· simulink

EEME E6601 9/11/23 Introduction to Control theory (continuous & discrete) ①

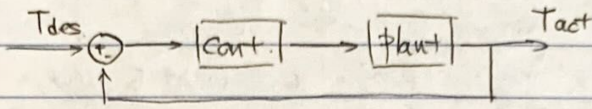
digital ↑



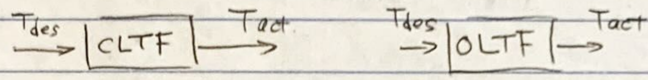
$\dot{T} + dT = u(t)$

$T(t)$.

· say actuators and sensors are perfect!
→ continuity feedback

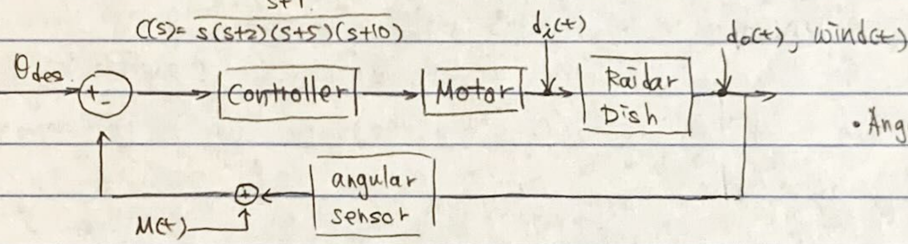


·



closed loop transfer function.

· Most plants are 1^{st} or 2^{nd} order systems → At most 2^{nd} order derivatives at most



· Angular sensor: Tachometer.

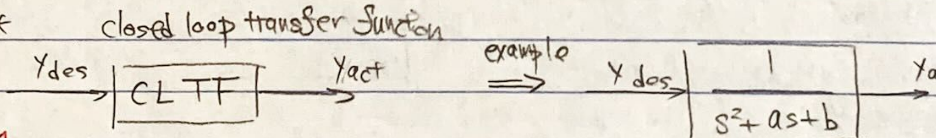
→ $d_i(t)$: input load disturbance, $d_o(t)$: output load disturbance, $M(t)$: electronic noise

$$* \quad \underbrace{a_n \frac{d^n y(t)}{dt^n} + \dots + a_1 \frac{dy(t)}{dt} + a_0 y(t)}_{\text{system (or plant)}} = \underbrace{F(t) + G(t)}_{\text{forcing functions}}$$

$$\cdot \; F(t) = b_m \cdot \frac{d^m f(t)}{dt^m} + \dots + b_1 \frac{df(t)}{dt} + b_0 f(t)$$

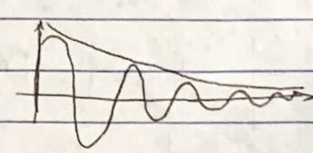
$$G(t) = c_q \cdot \frac{d^q g(t)}{dt^q} + \dots + c_1 \frac{dg(t)}{dt} + c_0 g(t)$$

* closed loop transfer function



· If LHS = 0 (no forcing function!) then we solve for the homogeneous solutions by finding characteristic polynomial

$* a_n s^m + a_{m-1} s^{m-1} + \dots + a_1 s^1 + a_0 s^0 = 0.$



· Find m roots and associated solutions according to : 1) Root s_i is real, solution is $e^{s_i t}$

2) complex conjugate roots $a \pm bi$ then two solutions : $e^{at}\cos(bt)$, $e^{at}\sin(bt)$

→ Next page

3) If any of those roots appears two times, then multiply the solution by t.
If three times multiply again by t

* The general solution is $Y = C_1Y_1 + C_2Y_2 + C_3Y_3 + \dots + C_nY_n$
  - c's are computed from initial conditions

* Next find particular solution $\xrightarrow{Y_p(t)}$ Now we consider the RHS (as not 0)
  - consider F(t) as forcing function:

Forcing Function	Guess	Forcing Function	Guess
c, a constant	A, a constant	$\sin(\omega t)$, $\cos(\omega t)$	$C\cos(\omega t) + E\sin(\omega t)$
t or At+B	Ct+D	$A\cos(\omega t) + B\sin(\omega t)$	
t^2, or $A^2t + Bt + C$	$Dt^2 + Et + F$		
e^{at}, or Ae^{at}	Be^{at}		

* Examples of finding $Y_p(t)$, the particular solution: $Y(t) = Y_H(t) + Y_p(t)$
  - superposition principle: $Y_H(t)$ has no forcing function ∴ 0
    $Y_p(t)$ has a forcing function, say F(t)

∴ solution due to both inputs (0, and F(t)) is sum of responses

* Say, $\frac{dy}{dt} + 2y = 1$

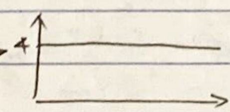
  - Guess $Y_p = A$, plug back → $0 + 2A = 1$ → ∴ $Y_p = \frac{1}{2}$
  - Guess $At + b = Y_p$ → $\left.\begin{matrix} 2A - 1 = 0 \\ A + 2B = 0 \end{matrix}\right\}$ solve...

* $\frac{dy}{dt} + 2y = \cos(t)$

  - Guess $Y_p(t) = A\cos(t) + B\sin(t)$

∴ $\left.\begin{matrix} (B + 2A - 1) = 0 \\ (2B - A) = 0 \end{matrix}\right\}$ solve

* Now say $2\ddot{y} + 3\dot{y} - y - 4 = 0$
→ $2\ddot{y} + 3\dot{y} - y = 4$



* Another way to solve ODE's:
Boy wouldn't it be nice to solve algebraic equations?

$$\mathcal{L}[y(t)] \equiv Y(s) = \int_0^\infty y(t)\, e^{-st}\, dt$$

$$\mathcal{L}^{-1}[Y(s)] \equiv y(t) = \frac{1}{2\pi i} \int_{\sigma - i = 0}^{\sigma + i = 0} Y(s)\, e^{st}\, ds \quad \left(\begin{matrix}\text{Residual} \\ \text{Theorem}\end{matrix}\right)$$

S: complex variable, $s = \sigma + i\omega$ where $i = \sqrt{-1}$ → Next page

EEME E6601 9/7/23 ②

• Apply $\mathcal{L}$ to transform $y(t)$ to obtain $Y(s)$

* so, $\mathcal{L}[y(t)] = Y(s)$, $\mathcal{L}[\dot{y}(t)] = ?$, $\mathcal{L}[\ddot{y}(t)] = ?$

• $\mathcal{L}\left[\frac{dy}{dt}\right] = \int_0^\infty \underbrace{\frac{dy}{dt}\,dt}_{dv}\; \overbrace{e^{-st}}^{u}$

→ recall: $\int_a^b u\,dv = uv\Big|_a^b - \int_a^b v\,du$

→ $y(t)e^{-st}\Big|_0^\infty - \int_0^\infty y(t)(-s)\cdot e^{-st}dt = y(\infty)\cdot \underbrace{e^{-\infty t}}_{0} - y(0)\cdot \underbrace{e^{0t}}_{1} + s\int_0^\infty y(t)\cdot e^{-st}dt$

$\therefore \mathcal{L}[\dot{y}(t)] = sY(s) - y(0)$, If 0 I.C.'s then

$\mathcal{L}[\dot{y}] = sY(s)$

• similarily, $\mathcal{L}[\ddot{y}] = s^2Y - sy(0) - \dot{y}(0)$, If 0 I.C.'s then

$\mathcal{L}[\ddot{y}(t)] = s^2Y(s)$

• conversely, $\mathcal{L}\left[\int_0^t y(t)dt\right] = \frac{Y(s)}{s}$

* what does this mean? Take Laplace of both sides of the ODE!

• say, ① $a_2 s^2 Y(s) + a_1 s Y(s) + Y(s) = b_2 s^2 X(s) + b_1 s X(s) + b_0 X(s)$

• the ODE was: $a_2\ddot{y} + a_1\dot{y} + y = \ddot{x}(t) + b_1\dot{x}(t) + b_0 x(t)$

①: $(a_2 s^2 + a_1 s + 1)Y(s) = X(s)(b_2 s^2 + b_1 s + b_0)$

$$\therefore \frac{Y(s)}{X(s)} = \frac{b_2 s^2 + b_1 s + b_0}{a_2 s^2 + a_1 s + 1}$$ → Transfer Function → $\frac{\text{output}}{\text{input}}$

* plant dynamics $G(s)$

$$\underset{\text{input}}{X(s)} \longrightarrow \boxed{\frac{b_2 s^2 + b_1 s + b_0}{a_2 s^2 + a_1 s + 1}} \longrightarrow \underset{\text{output}}{Y(s)}$$

↝ $Y(s) = G(s) \cdot X(s)$

• Laplace Transform Table :

$x(t)$	$X(s)$	$x(t)$	$X(s)$
impulse	1	$\cos(\omega t)$	$\frac{s}{s^2+\omega^2}$
step	$\frac{1}{s}$	$e^{-at}\sin(\omega t)$	$\frac{\omega}{(s+a)^2+\omega^2}$
e^{-at}	$\frac{1}{s+a}$	$e^{-at}\cos(\omega t)$	$\frac{s+a}{(s+a)^2+\omega^2}$
t	$\frac{1}{s^2}$		
$\sin(\omega t)$	$\frac{\omega}{s^2+\omega^2}$		

phone# 518.505.6425
(professor)

office hr: Dilnshita: Mon 2p-3P
Dominik: Fri 1p-2P
Haocom: Tue 1p-2p
Yuexuan: Wed 2p-3p

EEME E6601 9/14 ①

* Transfer Functions:

*say, $0.25\ddot{y}+1.25\dot{y}+y=0.5\dot{x}(t)+1.25x(t)$, x: input, y: output, 0 initial conditions

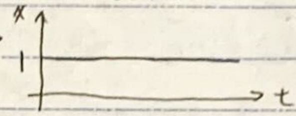
→ Take $\mathcal{L}$ transform: $0.25s^2Y(s)+1.25sY(s)+Y(s)=0.5sX(s)+1.25X(s)$

- X in laplace domain: capital, in time domain: small letter

$\therefore Y(s)[0.25s^2+1.25s+1]=[0.5s+1.25]X(s)$

→ Transfer Function $=\dfrac{Y(s)}{X(s)}=\dfrac{0.5s+1.25}{0.25s^2+1.25s+1}=\dfrac{0.5s+1.25}{(s+1)(0.25s+1)}$

*say, unit step input ⇝



$X(s)=\frac{1}{s}$, $Y(s)=\dfrac{0.5s+1.25}{s(s+1)(0.25s+1)}$

T.F.s

- $X(s) \rightarrow \boxed{G(s)} \rightarrow Y(s)$ $Y(s)=G(s)\cdot X(s)$

- In Matlab: tf, zpk, ss (from Zoom session)

```
>> G = tf([0.5 1.25], [0.25 1.25 1])
```
→ transfer function object

→ so, $Y(s)=\dfrac{0.5s+1.25}{s(s+1)(0.25s+1)}$ if you do PFE
↳ partial Fraction Expansion

- In Matlab: residue

$Y(s)=\left(\frac{1}{s}-\frac{1}{s+1}\right)+0.25\left(\frac{1}{s}-\frac{1}{s+4}\right)$

→ From Laplace Transform Table: inverse laplace goes from s domain to t domain

$\therefore y(t)=1-e^{-t}+0.25(1-e^{-4t})$

* Impulse Response and Linear Convolution.

- say, $\underbrace{Y(s)}_{output}=H(s)\cdot\underbrace{X(s)}_{input}$, $H(s)$ is unknown.

① : impulse.

when $X(s)=1$ ①, then $Y(s)=H(s)$ and $y(t)=h(t)$

- In general, $\mathcal{L}^{-1}[H(s)X(s)]\triangleq\frac{1}{2\pi i}\oint_{\sigma-i\omega}^{\sigma+i\omega}H(s)X(s)e^{st}ds\ (s=\pm\sigma\pm i\omega)$ → Residual Theorem.

$=\frac{1}{2\pi i}\int_{\sigma-i\omega}^{\sigma+i\omega}\left(\int_0^\infty h(\tau)e^{-s\tau}d\tau\right)X(s)e^{st}ds=\int_0^\infty\underbrace{\frac{1}{2\pi i}\int_{\sigma-i\omega}^{\sigma+i\omega}X(s)\cdot e^{s(t-\tau)}ds}\cdot h(\tau)d\tau$

From definition of laplace ↳ $X(t-\tau)$

How to find the dynamics of systems ★

$\therefore \underbrace{y(t)}_{\text{Response of system}}=\mathcal{L}^{-1}[H(s)\cdot X(s)]=\int_0^\infty\underbrace{h(\tau)}_{\text{impulse response}}\cdot\underbrace{x(t-\tau)}_{\text{input}}d\tau$

$X(s) \rightarrow \boxed{H(s)} \rightarrow Y(s)$

$Y(s)=H(s)X(s)$

→ Linear convolution of $h(t)$ and $x(t)$

1) $\boxed{\text{Linear convolution in time is multiplication in s-domain}}$

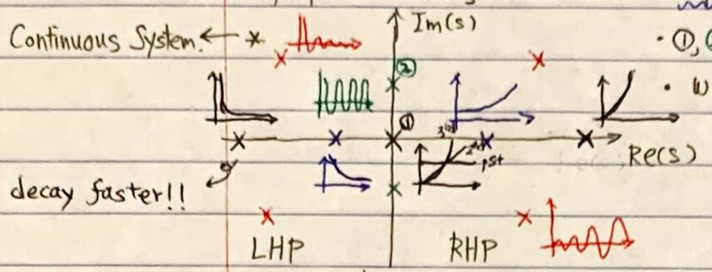
2) $\boxed{\text{impulse response provides a complete characterization of the dynamic behavior}}$

① roots of denominator of T.F.
② roots of numerator of T.F.] → Matlab: "pole", "roots"

* Say, $U(s) \rightarrow \boxed{G(s)} \rightarrow Y(s)$ $Y(s) = G(s) \cdot U(s)$, $y(t) = \mathcal{L}^{-1}[G(s) \cdot U(s)]$

depends on ①poles and ②zeroes of $G(s)$ and
depends on poles and zeroes of $U(s)$

- Impulse Response: Natural Response
- proper T.F. is one where $n_p > n_z$ ③ number of poles, ④ number of zeroes



- ①, ②: marginally stable
- we always want left half plane
- LHP: stable, Imaginary axis: marginally stable, RHP: unstable

we are still covering open loop system

- For 2^{nd} order systems, we have 2 poles
- Say, $G(s) = \dfrac{1}{4s^2+s+0.1}$ → characteristic equation: describe physics of the systems

roots: s_1, s_2

$s = \sigma \pm i\omega$

* pzmap(G)

* Frequency Response: if $U(s) = \frac{A}{s}$ (step of amplitude A) then, $Y(s) = G(s) \cdot \frac{A}{s}$

- Assuming $G(s)$ is stable, then, steady-state Response is $A\,G(0)$
- Recall F.V.T. (final value theorem)

→ $\boxed{\lim_{t\to\infty} y(t) = \lim_{s\to 0} s \cdot Y(s)}$ → HW.

* $G(0)$: D.C. gain → in matlab: "dcgain(G)" unit step

* If $G(s) = \frac{1}{s}$ (pole of zero!, undesired!)

$U(s) \rightarrow \boxed{G(s)} \rightarrow Y(s)$

- quick check: if unit step, then, response $Y(s) = \frac{1}{s^2}$ → ∴ unbounded!

since, $Y(s) = G(s) \cdot U(s)$

- If $u(t) = A\sin(\omega t)$ then, $y_{ss}(t) = A \cdot |G(s)| \cdot \sin(\omega t + \phi)$ → For linear system, we have same frequency

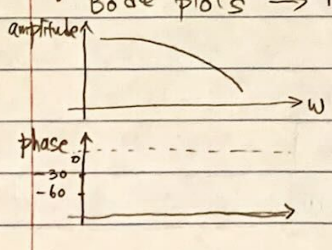
↳ $|G(s)|$: gain of system for sinusoidal inputs.

- We are interested in the frequency response → substitute $i\omega$ for s

→ ∴ $y_{ss}(t) = A \cdot |G(i\omega)| \cdot \sin(\omega t + \phi)$

→ Magnitude $G(s)$ is $|G(s)| \equiv |G(i\omega)|$

- Bode plots → matlab: bode(G)



EEME E6601 9/14 ②

* Bode Plot Construction

• Say, a C# $c = a + ib$ (for instance)

$abs(c) \equiv |c| = \sqrt{a^2+b^2}$

$phase(c) \equiv arg(c) = \tan^{-1}\left(\frac{b}{a}\right)$

Let's now say we have

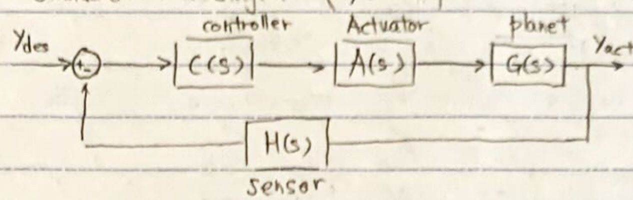
EEME E6601 Intro to Control 9/21 ① HW#3 ⇝ Bode plot by hand

*Bode (continue from last lecture)

if $d = \frac{1}{a+ib}$, can multiply top and bottom by $a-ib$

say, $G(s) = \frac{1}{s+20}$ → Transfer Function $= \frac{Y}{U} = \frac{1}{s+20}$ → $\dot{y} + 20y = u$

- Standard block diagram ($y \equiv$ temperature in this example)



- purpose: we need to design $C(s)$, to get desired room temperature
- Each box should have transfer function

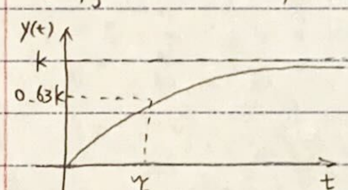
- In Bode form, $G(s) = \frac{1}{20} \cdot \frac{1}{\frac{s}{20}+1}$ → This is also called time-constant form

→ So, the Bode form is $0.05\left(\frac{1}{20}\right)$

- $\boxed{\text{Hold}}$: say $\frac{1}{(s+a)(s+b)} = G(s)$ → since highest power of s is 2, its second order system

$$G(s) = \frac{Y(s)}{U(s)} = \frac{1}{ab} \cdot \frac{1}{\left(\frac{s}{a}+1\right)\left(\frac{s}{b}+1\right)} = \frac{1}{ab} \cdot \frac{1}{(\tau_1 s+1)(\tau_2 s+1)}$$

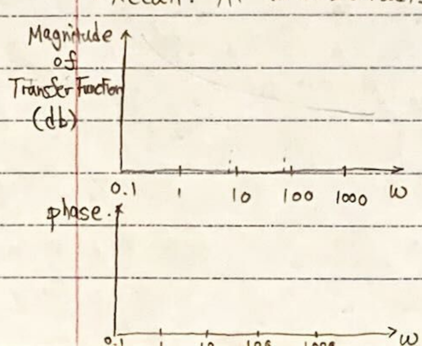
- say, 1st order system



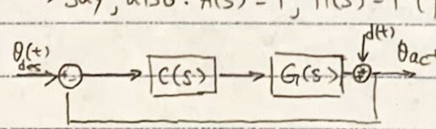
- $\boxed{\text{Back to}}$: so, $G(s) = 0.05 \cdot \frac{1}{\frac{s}{20}+1}$

→ $\lim_{t\to\infty} y(t) = \lim_{s\to 0} sY(s)$: Final Value Theorem

- DC gain in MATLAB : >> dcgain(G)
- Here, $G(s) = \frac{0.05}{\frac{s}{20}+1}$ → $G(i\omega) = \frac{0.05}{\frac{i\omega}{20}+1}$ → How this transfer function acts in different domain?
- Recall: At $\omega \ll 20$ rad/s



→ say, also: $A(s) = 1$, $H(s) = 1$ (perfect actuator & sensor)



d(t): wind, rain

then $\frac{i\omega}{20} \ll 1$ ∴ $G(i\omega) \approx 0.05$

Next page → when we draw Bode plot, draw the extreme case first

- If $\omega >> 20$ rad/s then $\frac{i\omega}{20} >> 1$. Hence, $G(i\omega) = \frac{0.05}{\frac{i\omega}{20}}$ which is $\frac{1}{i\omega}$.

→ Multipy by the complex conjugate. So, $G(i\omega) = \frac{1}{i\omega} \cdot \left(\frac{-i\omega}{-i\omega}\right) = -\frac{i}{\omega}$.

→ Recall that $i^2 = -1$.

→ Now, 1db is $20\log_{10}|\text{Magnitude}|$ by definition

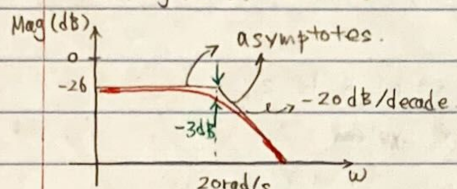
→ $20\log_{10}|G(i\omega)| = 20\log_{10}\left|\frac{-i}{\omega}\right|$

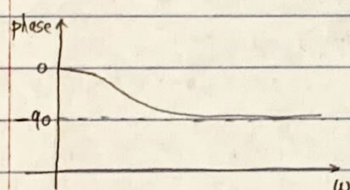
→ Thus, $|G(i\omega)| = \left|\frac{1}{\omega}\right| = 20\left[\log_{10}|1| - \log_{10}|\omega|\right] = 0 - 20\log_{10}|\omega|$

↳ On a $\log(\omega)$ axis, this looks like a straight line $y = ax+b$ where y is $G(i\omega)$, a is -20, x is $\log_{10}|\omega|$

↳ The slope has units of y-axis over x-axis (both are logarithmic)

- Slope is -20dB/decade → due to definition of the db (decibel)
- $20\log(0.05) = -26$dB





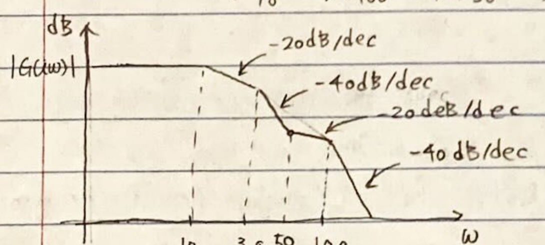
For every pole you go down by 90°

For every Zero you go up by 90°

* On the magnitude plot

- Below 0dB line. ∴ attenuation
- Above 0dB line ∴ amplification

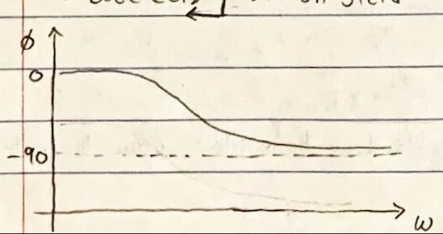
* Example: $\frac{12\left(\frac{s}{50}+1\right)}{\left(\frac{s}{10}+1\right)\left(\frac{s}{100}+1\right)\left(\frac{s}{30}+1\right)}$ ⇝ 3rd order system.



- 1dB = $20\log_{10}|\text{Mag}|$.

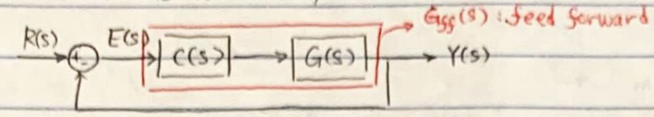
⇝ with the zeros (numerator of system)

>> bode (G) but on field

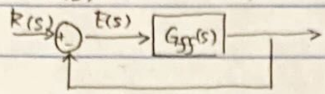


EEME E6601 Intro to Control 9/21 ②

* Bode Forms and Steady-state error



⇝ $E(s) = R(s) - Y(s)$



⇒ What is the transfer function that describes how the error is affected by the input? So, get R(s) to E(s) Transfer Function!!

→ $Y(s) = E(s)G_{ff}(s) = [R(s) - Y(s)]G_{ff}(s)$, $R(s) = E(s)(1 + G_{ff}(s))$

→ Transfer function is always $\frac{\text{output}}{\text{input}}$

$$\frac{E(s)}{R(s)} = \frac{1}{1+G_{ff}(s)}$$ → A closed loop transfer function

→ Now, let's say $G_{ff}(s) = \dfrac{k(s+z_1)(s+z_2)}{s^2(s+p_1)(s+p_2)}$, where, k is a constant gain

→ In Bode form → Bode Gain ① two free integrator.

$$\therefore G_{ff}(s) = \frac{\boxed{\left[\frac{k z_1 z_2}{p_1 p_2}\right]}\left(\frac{s}{z_1}+1\right)\left(\frac{s}{z_2}+1\right)}{s^2\left(\frac{s}{p_1}+1\right)\left(\frac{s}{p_2}+1\right)}$$

* Recall $\tau = \frac{1}{z}$ or $\frac{1}{p}$, τ is in unit 'sec', z and p are in unit rad/s.

* Zeroes : material property & geometry.

* Steady state error is

$$e_{ss} = \lim_{t\to\infty} e(t) = \lim_{s\to 0} (sE(s))$$

$$= \lim_{s\to 0} \frac{sR(s)}{1+G_{ff}(s)}$$ → CLTF → input

- Assuming all poles and zeroes are in the LHP ⇝ left hand plane.
- Nice feature of Bode form: when we take $\lim_{s\to 0}$, all monomials → 1
- except for the free integrators.
- In general, $G_{ff}(s) = \dfrac{k_B}{s^l}\cdot\dfrac{N(s)}{D(s)}$, where l is transform free integrators.

→ inputs : $\frac{A}{s}$ ⇝ step of mag. A

$\frac{A}{s^2}$ ⇝ Ramp input of mag. A ⇝ and $\frac{A}{s^q}$, q = 1 : step, q = 2 : Ramp, q = 3 : parabola.

$\frac{A}{s^3}$ ⇝ parabola input.

$$\therefore e_{ss} = \lim_{s\to 0} \frac{s\cdot A\cdot s^{-q}}{1+\frac{k_B N(s)}{s^l D(s)}} = \lim_{s\to 0} \frac{A\cdot s^{1-q}}{\frac{s^l D(s) + k_B N(s)}{s^l D(s)}}$$

$$\to e_{ss} = \lim_{s\to 0} \frac{s^l\cdot s^{1-q} A\cdot D(s)}{s^l D(s) + k_B N(s)} = \lim_{s\to 0}\left[\frac{A\cdot s^{l+1-q}}{s^l + \frac{k_B N(s)}{D(s)}}\right] = A\cdot \lim_{s\to 0}\frac{s^{l+1-q}}{s^l + k_B}$$

→ So, vary $q = \{1,2,3\}$, and $l = \{0,1,2\}$

→ Now, can fill the e_{ss} in the table of HZ

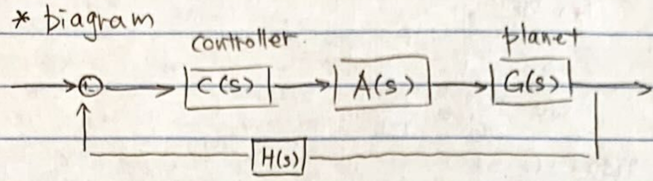
$$e_{ss} \rightarrow \begin{cases} 0 \\ \text{constant (non-zero} \\ \text{infinity} \end{cases}$$

R(s)	Type 0 System $l=0$	Type 1 $l=1$	Type 2 $l=2$
step			
Ramp			
parabola			

→ Filling chart = HW!!

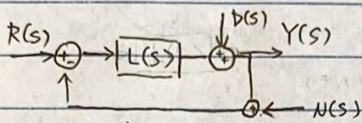
EEME E6601 9/28/23 Introduction to Control Theory.

* diagram

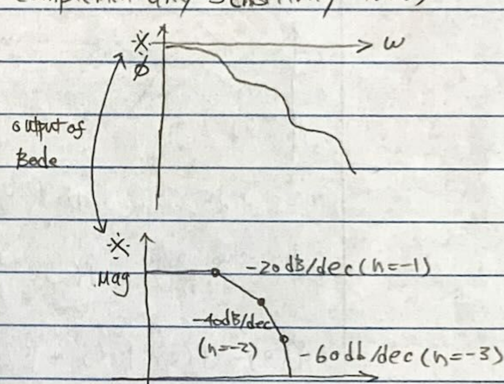


- Negative feedback loop.
- can use "feedback" in MATLAB
- CLTF in Files (canvas)

→ if $A(s)=1$ and $H(s)=1$, $\underbrace{L(s)}_{\text{Loop gain}} = G(s) * C(s)$



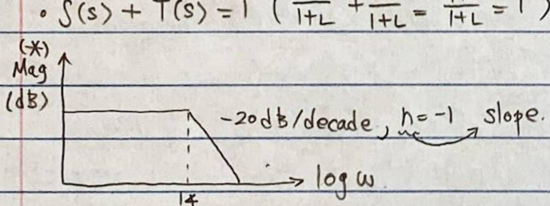
- $T.F. = \dfrac{\text{"direct"}}{1 + \text{"loop"}}$
- In CLTF attachment, if $F(s)=1$ then, left with 4 essential CLTF's "Gang of four"
- $T(s) = \dfrac{L}{1+L}$, $T(s)$ is transmission T.F. (also called complementary Sensitivity T.F.)



- Sensitivity T.F. (T.F. of the disturbance)

$S(s) = \dfrac{1}{1+L(s)}$

- $S(s) + T(s) = 1$ $\left(\dfrac{L}{1+L} + \dfrac{1}{1+L} = \dfrac{1+L}{1+L} = 1 \right)$



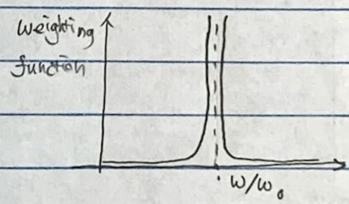
- How do we draw ϕ, ϕ: phase or phase log.

→ $y = A|G(i\omega)| \cos(\omega t + \phi)$

$\angle G(i\omega)$ = argument or angle

- It would be nice to quickly draw phase log diagram

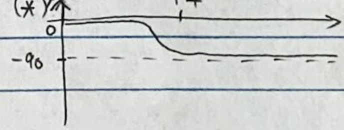
※ weighting function



- $\angle G(i\omega) = \dfrac{1}{\pi} \displaystyle\int_{-\infty}^{\infty} \underbrace{\dfrac{d \log |G(i\omega)|}{d \log \omega}}_{\text{slope of the magnitude plot } (N(\omega))} \cdot \underbrace{\log \left(\dfrac{\omega + \omega_0}{\omega - \omega_0} \right)}_{\text{weighting function}} \cdot d(\log(\omega))$

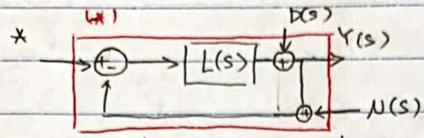
- Slope: n (-1, -2, -3)

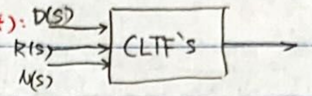
Weight function: $\displaystyle\int_{-\infty}^{\infty} \log \left| \dfrac{\omega+\omega_0}{\omega-\omega_0} \right| \rightarrow \dfrac{\pi^2}{2}$



- $\angle G(i\omega) = \dfrac{1}{\pi} \cdot n \cdot \dfrac{\pi^2}{2} \cdot (-1) = -N \cdot \dfrac{\pi}{2}$ → slope

perfect solution if $G(s) = \dfrac{1}{s^n}$

* 

(∗): 

• Say, $\frac{Y(s)}{N(s)} = \frac{-L}{1+L}$

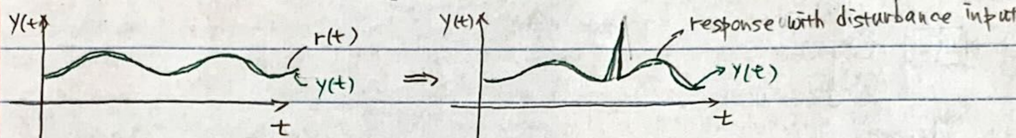
• $T(s) = \frac{L(s)}{1+L(s)}$, $S(s) = \frac{1}{1+L(s)}$ → $T(s)+S(s)=1$

→ $Y(s) = T(s)\cdot R(s) + S(s)\cdot D(s) - N(s)\cdot\frac{L(s)}{1+L(s)}$

• If perfect tracking following then $T(s)\to 1$, $S(s)\to 0$

• Since, $T(s) \equiv \frac{Y(s)}{R(s)} = 1$

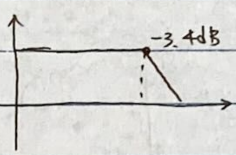
* For perfect disturbance rejection



• If $|L(s)| \ll 1$, then $T(s) \to L(s)$, $S(s) \to 1$

If $|L(s)| \gg 1$, then $T(s) \to 1$, $S(s) \to \frac{1}{L(s)}$

* Simple pole



• $\omega = 2\pi f = \frac{2\pi}{T}$

time constant

• -3.4 dB means: $10^{\left(\frac{-3.4}{20}\right)} \approx 0.64 \approx 0.63\% \approx 1\tau$ → This is CLτ

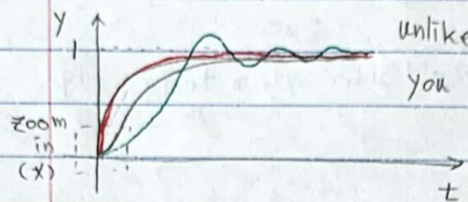
• By the way: $T(s) = \frac{Y(s)}{R(s)}$

• $Y(s) = \underset{1}{\cancel{T(s)}}\cdot R(s) + \underset{0}{\cancel{S(s)}}\cdot D(s) - \underset{0}{\cancel{N(s)}}\cdot\frac{L(s)}{1+L(s)}$

*proper T.F.: degree (denominator) > degree (numerator)

EEME E6601 Intro to Control Theory 10/5

* Closed Loop Transfer Function's :



unlike the open loop step response that you have seen

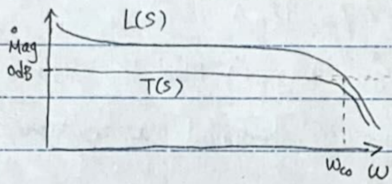
⇝ ideal : To reach to '1' (steady-state) without any oscillation, and fast!

– : due to the controller

HW#4 : Design $C(s)$

* $T(s) = \frac{Y(s)}{R(s)} = \frac{L(s)}{1+L(s)} \rightarrow Y(s) = \frac{L(s)}{1+L(s)} \cdot R(s) \rightarrow y(t) = \checkmark,\ L(s) = G(s) \cdot C(s)$

• Designing Controller : Gain, pole, zero



→ speed of motion = $\frac{1}{\tau}$, (τ: time constant)

(*) : slope of red line. : $\frac{2\times10^{-3}}{0.2} = 0.01$ → From the figure "CL Step Response" ⇝ MATLAB.

if you want faster response → add another poles and zeroes in the middle and high frequency of the $L(s)$.

MATLAB "loopshape2send"

• when you design a Controller, draw Bode plot for Plant function!

→ And then shape (**) the plant T.F. to get desired output using controller.

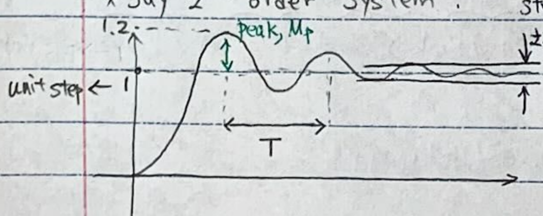
(**) : shape $L(s)$!!

• when $t \rightarrow \infty$, the response reaches steady state and frequency is zero

• Below the yellow line ($S(s)$, sensitivity T.F.) will be attenuated.

* Lowest frequency zero, speed of response before closed loop settles.

* Say 2^{nd} order System .



⇝ General 2^{nd} order System

$$\boxed{TF(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}}$$

• rise time : τ_r, settling time : τ_s → To view 'stepinfo()', 'ltiview()' → MATLAB.

Ex. say, $G(s) = \frac{9}{s^2+8s+9} \rightarrow$ $\omega_n^2 = 9 \rightarrow \omega_n = 3\,rps$ (ω_n: frequency when it will oscillate)

$8 = 2\zeta\omega_n \rightarrow \zeta = \frac{8}{6}$ (ζ : damping ratio)

→ $s_{1,2} = \sigma \pm i\omega_d$ (ω_d : damped natural frequency, $\omega_d = \omega_n\sqrt{1-\zeta^2}$

$s \triangleq \sigma \pm i\omega$.

∴ $y(t) = c_1 e^{s_1 t} + c_2 e^{s_2 t}$ → general solution (exponential sinusoid)

Time Domain Requirement

→ * $\tau_r \simeq \frac{1.8}{\omega_n}$, if we need $\tau_r \le \alpha$: $\frac{1.8}{\omega_n} \le \alpha$ (Time Domain Requirement)

→ By adding poles and zeroes in the transfer function, get desired rise time

↳ 2^{nd} order sys. in the first page.

* peak M_p: $M_p = \frac{e^{-\pi\zeta}}{\sqrt{1-\zeta^2}} \le \beta$

* Also, $\tau_s \simeq \frac{3}{\sigma} \le 8$

* Bode to CLTF's ...

All CLTF's have in common $1+L(s)$ on denominator!

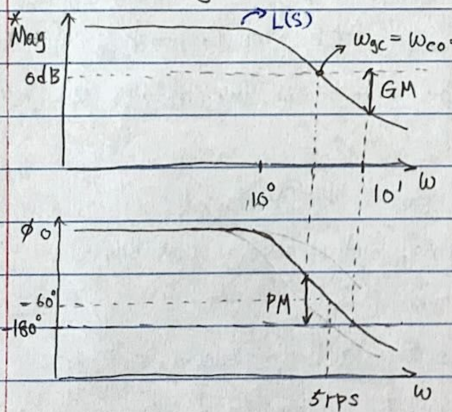
↳ This means that we don't want it $(1+L(s))$ to be zero, since all CLTF's → ∞ for any input. Let's see when it goes to → zero, (when Denominator(s) → 0, that is)

• $1+L(s)=0$ → $L(s)=-1$ → Magnitude of 1 and $\phi = -180°$

→ So we need to avoid this condition ($\phi = -180°$, Mag = 1) like the plague!

• rps: radian per second.

→ So much so that how far you are from either is deemed a relative stability distance (margin)



• Say ω = 5 rps, delay = $\frac{0.2}{6}$ = 0.033 seconds

• we want GM to be bigger

• $\boxed{\text{phase margin (PM)}}$: safe guard against unknown (unmodeled) delays in the system

→ PM > 30° ① (if you want to have larger PM, add zeroes in the transfer function).

↳ it will make response slower. (pay back).

>> margin(L) "MATLAB"

① 30° away from −180° from phase plot.

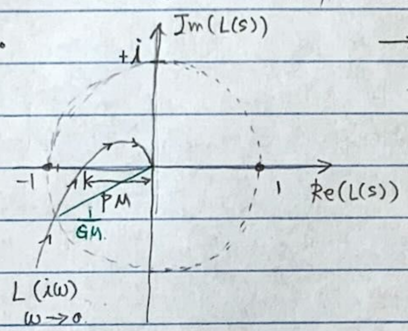
• $\boxed{\text{Gain margin (GM)}}$: safe guard againt unknown (unmodeled) gains

→ GM > 2.

EEME E6601 Intro to Control Theory 10/5

* Bode Condition for stability : $|L(i\omega_{-180})| < 1$
- BIBO stable : bounded input, bounded output.
- $1+L(s) \neq 0$ is a function of $1+G(s)C(s) \neq 0$, $C(s)$: poles, zeroes, gain what you put
  ↳ we should design $C(s)$ such that $1+L(s) \neq 0$
- So poles of $T(s)$
- if one pole such that it $L(s)=0$ → ∴ unstable closed loop behavior (1 pole in RHP)

* Conformal Mapping
- Cauchy : principle of the argument, $1+L(s)$.
  ↳ go around a simple contour
  ↳ say $F(s)=0$ $\left(F(s) = \frac{Num(s)}{Den(s)}\right)$, $\overset{cw}{N} = \overset{cw}{Z} - \overset{cw}{P}$ (zeroes - poles), cw : clock-wise.
  N tells us how many roots are unstable in $F(s)$
-



→ $L(s) = -1$, NYquist diagram

→ NYquist is to check stability of the T.F

→ In MATLAB, >> "nyquist(G)"

EEME E6601 Intro to Control Theory 10/12

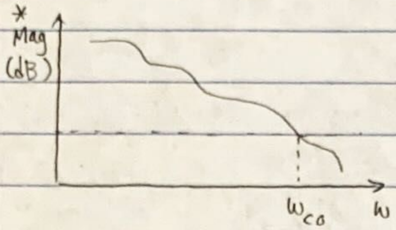
* Routh-Hurwitz Criterion.

Lecture5 Notes ←
- Root Locus : rlocus, rlocfind
- Routh
- NYquist

↓

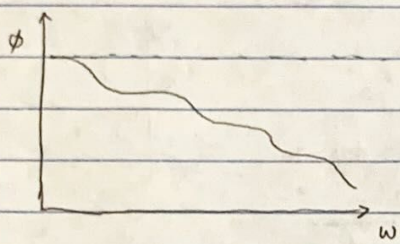
For stability

*

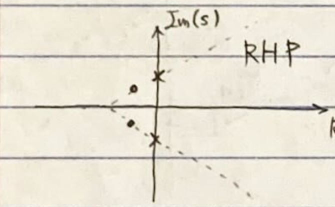


→ $L(s) = G(s) \cdot C(s)$, $C(s)$: gain, zeroes, poles.

→ $\frac{1}{\omega_{co}} = \tau_{co}$



- Root locus



→ If root goes on RHP, the system will be unstable.

- Routh Array : say denominator $s^3 + 4s^2 + 2s + k = 0$ $(1 + L(s) = 0)$

→ If the element * is 0, then can not divide by zero!
so replace 0 with ε(a very small number)

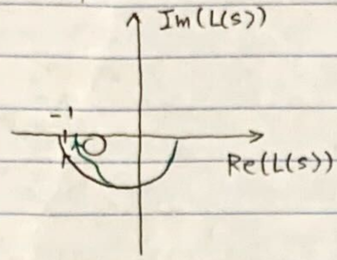
①

s^3	1	2
s^2	4	k
s^1	$\frac{1 \cdot 2 - k \cdot 1}{4}$	✓
s^0	✓	

②

s^3	1	2
s^2	$0 \to \varepsilon$	K
s^1	$\frac{1 \cdot 2 - k \cdot 1}{4}$	✓
s^0	✓	

• NYquest

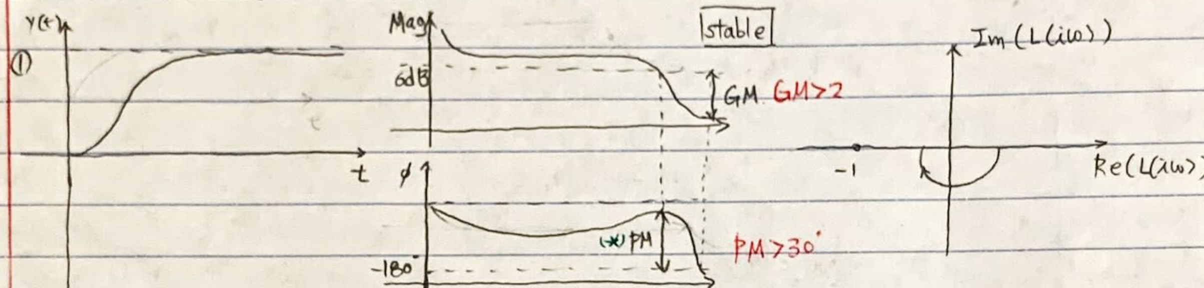


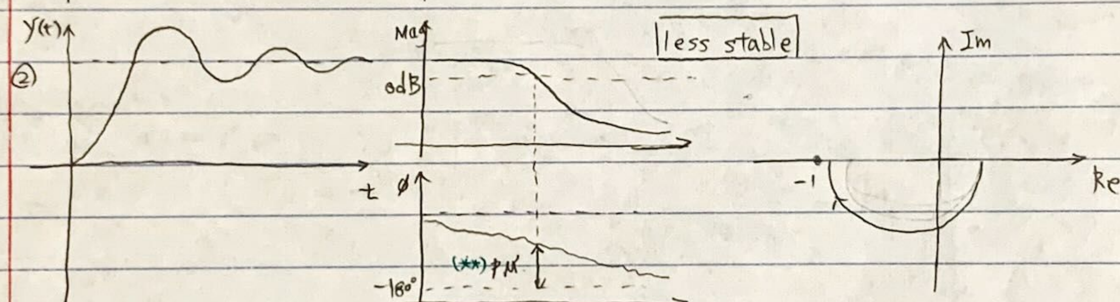
$\rightarrow 1 + L(s) = 0$

$1 + L(i\omega) = 0$

$L(i\omega) = -1$

* Relations among Bode, NYquest, and step response, $L(s) = C(s) \cdot G(s)$ → gain, zeroes, poles

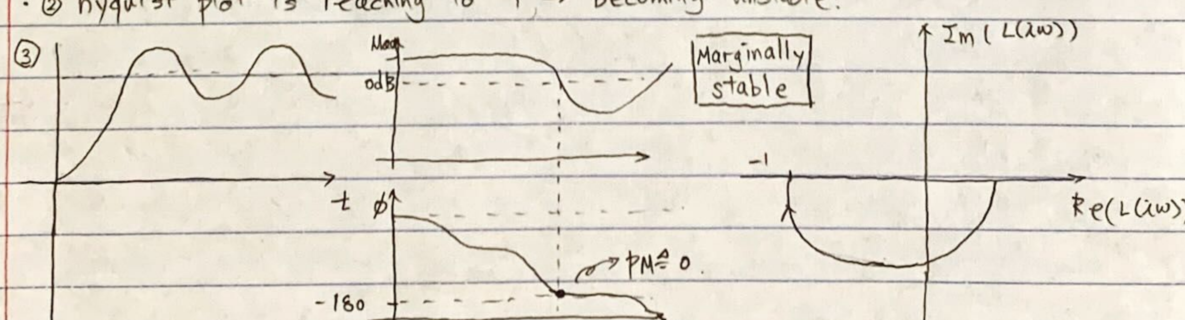


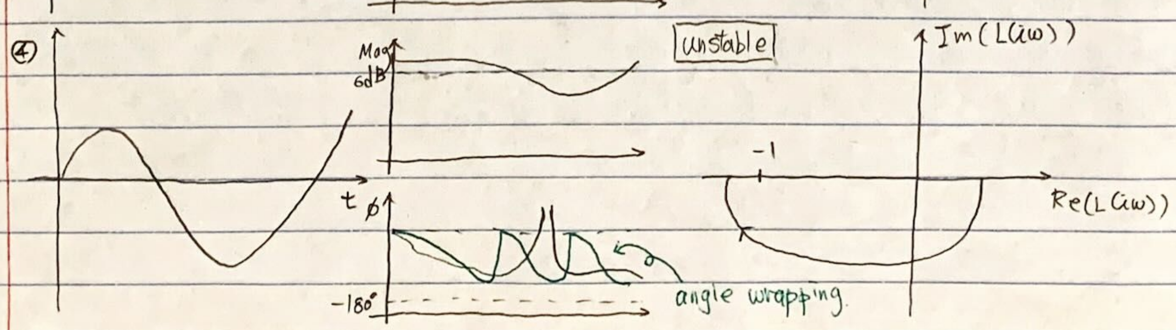


• ① is better result for response

• (*) is larger than (**), we want PM large!!

• ② nyquist plot is reaching to -1 → becoming unstable.





EEME E6601 Intro to Control Theory 10/12

- If delay: $e^{-T_d s}$, say $e^{-0.5s}$

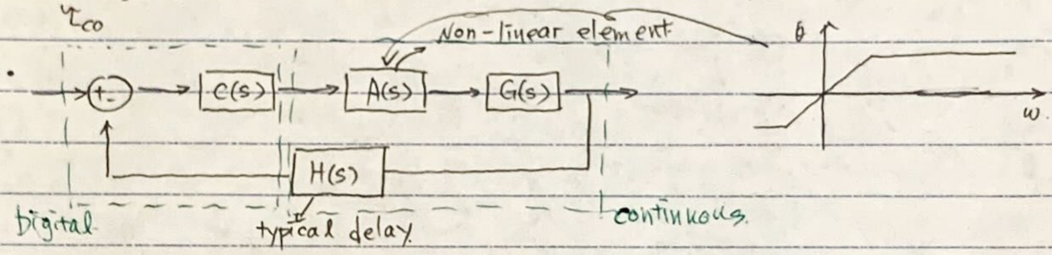
pade $\approx \dfrac{1+\frac{T_d}{2}s}{1-\frac{T_d}{2}s}$

* Actuator Limitations :

- ok, $\tau_{co} = \dfrac{1}{w_{co}}$ for a unit step
  ↳ time constant

- For a non-unit step, the initial system rate would be

→ $\dfrac{r_m}{\tau_{co}}$, r_m: maximum allowable magnitude of the step input.



- For the actuator limit not to be exceeded then

- $\dfrac{r_m}{\tau_{co}}$ is ~~set = actuator rate limit (say, a_{rl})~~

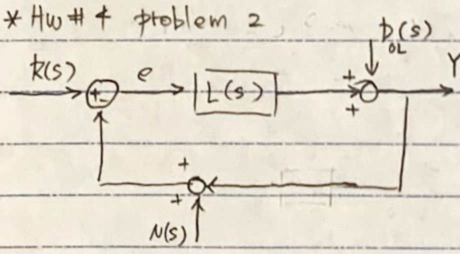
- a_{rl} has magnitude limits per sec :

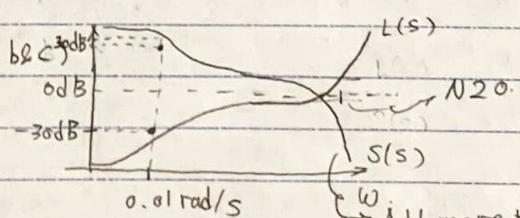
$$\frac{r_m}{\tau_{co}} = a_{re} \therefore \frac{a_{re}}{r_m} = \frac{1}{\tau_{co}} = w_{co}$$

↳ so, w_{co} (cross-over frequency) can be computed

↳ Based on the actuator rate limit, $w_{co} = \dfrac{a_{rl}}{r_m}$

* Hw # 4 problem 2





↳ Add more poles to have more than -20dB/dec

↳ Noise suppression

b) $\dfrac{Y(s)}{D_{OL}(s)} = \dfrac{1}{1+L(s)} = S(s)$

c) $\dfrac{Y(s)}{N(s)} = \dfrac{-L(s)}{1+L(s)} = -T(s)$

↳ Noise to output.

EEME E6601 Intro to Control Theory 10/19 ①

★ State Space Representation

- physical plant ⟶ O.D.E → in MATLAB: tf, zero, pole, gain, steady state
- Say, $\ddot{y} + a_1\dot{y} + a_2 y = b\cdot u(t)$ (*) → System whose output is y and u is input → open loop.

→ Convert this 2^{nd} order ODE into two 1^{st} order ODE

↳ Because 1^{st} order ODE is fundamental, many things are developed with 1^{st} order ODE.

$$\xrightarrow[U(s)]{\overset{\text{input}}{u(t)}} \boxed{T.F.} \xrightarrow[Y(s)]{\overset{\text{output}}{y(t)}}$$

- if $a_1 = 600$ for (*) : over damped system.
- if there is a sign change in the coefficient in (*), there is a root in the RHP → unstable.
- Introduce z's

$\dot{z}_1 = z_2$ ··· (1)

$\dot{z}_2 + a_1 z_2 + a_2 z_1 = b u(t)$

$\dot{z}_2 = -a_1 z_2 - a_2 z_1 + bu(t)$ ··· (2)

From equations (1) & (2), we get $\begin{cases} \dot{z}_1 = 0\cdot z_1 + 1\, z_2 + 0\cdot u(t) \\ \dot{z}_2 = -a_2 z_1 - a_1\cdot z_2 + bu(t) \end{cases}$ ⟶ one system

$$\therefore \begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -a_2 & -a_1 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \begin{bmatrix} 0 \\ b \end{bmatrix} \cdot u(t)$$

→ ∴ $\dot{z} = Az + Bu$ (state equation, where u is input, z is output)

- If you have sensors then: (sensor should measure the output z's.)

$$y = \underbrace{[1\ 0]}_{(**)} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \cancel{D}\cdot u$$

0, mostly (**): output matrix

- Both equations form the state-space representation

$\dot{z} = Az + Bu$ ⟶ Hence, in MATLAB

$Y = Cz + Du$ ≫ G = ss(A, B, C, D) → system

※ if it were $\dot{z} = Az$ → Homogeneous solution

if we have $\dot{z} = Az + Bu$ → particular solution

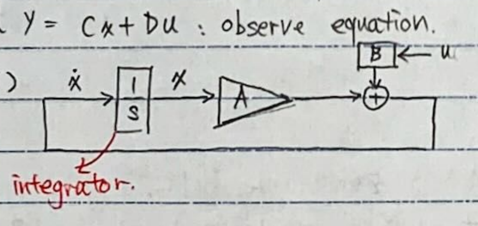
- What about the T.F. of a system represented by state-space? → Use MATLAB 'lsim'

- Use sensor for measuring position

$$* \xrightarrow[U(s)]{u(t)} \boxed{T.F.} \xrightarrow[Y(s)]{y(t)}$$

- Get the T.F of a state-space system

$\begin{cases} \dot{x} = Ax + Bu : \text{state equation} \\ y = Cx + Du : \text{observe equation.} \end{cases}$ ※ A, B, C, D : parameters

i) 

$\dot{x} = Ax + Bu$

Laplace → $sX(s) = AX(s) + BU(s)$

Next page →

$\therefore X(s)[s\underset{\sim}{I} - A] = BU(s)$ → identity matrix $\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

input to the state output.

$$\therefore \boxed{\frac{X(s)}{U(s)} = (sI - A)^{-1}B} \rightarrow \text{without sensor}$$

• What about TF from $U(s)$ to $Y(s)$?, what is $\frac{Y(s)}{U(s)}$?

Then, $Y(s) = CX(s) + DU(s)$

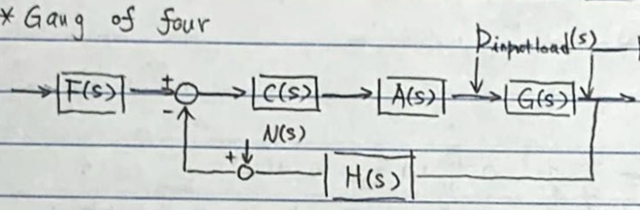
→ plug in: $Y(s) = C[sI - A]^{-1} \cdot BU(s) + DU(s)$

$$\therefore Y(s) = [C(sI - A)^{-1} \cdot B + D]\, U(s)$$

$$\therefore \boxed{\frac{Y(s)}{U(s)} = C(sI - A)^{-1}B + D} \rightarrow \text{with sensor}$$

• Stability : Recall Routh : roots of $1 + L(s) = 0$ in RHP?, $L(s) = C(s) \cdot G(s)$

hold * Gang of four



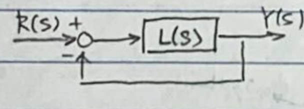
→ is any input comes in, $Y(s)$ gets affected.

→ Any closed loop transfer function here has denominator of $1 + L(s)$

Back • State-space : Another way of expressing transfer function

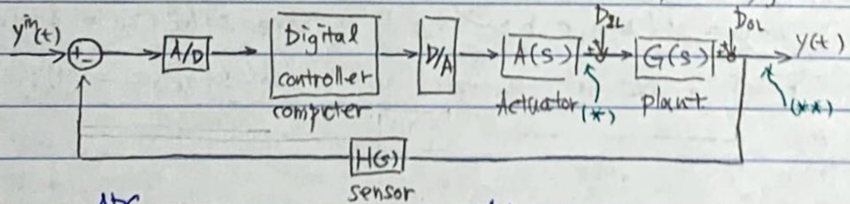
: This is for open-loop system

• say, $D = 0$: $\xrightarrow{\text{step or impulse}} \boxed{C(sI - A)^{-1}B} \rightarrow$ ⇝ $\xrightarrow{\text{input}} \boxed{G(s)} \xrightarrow{\text{output}}$

• 

$$T = \frac{L(s)}{1 + L(s)}$$

will not be in Exam 1. → * Digital control systems (Discrete-time control systems), we need to discretize the time domain

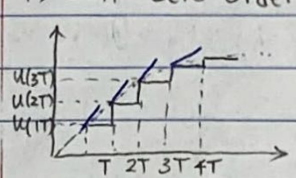


• $\boxed{A/D}$ (ADC) : Analog to Digital, $\boxed{D/A}$ (DAC) : Digital to Analog converter

(*): manipulated variable, (**): Controlled Variable

• DAC (ZOH, FOH, polygonal Hold)

i) ZOH : Zero order Hold



• T: sample time

• Output of a controller is Zero-order Hold:

$u(t) = u(kT),\ kT < t < (k+1)T,\ k = 0, 1, 2, 3, \ldots, \infty.$

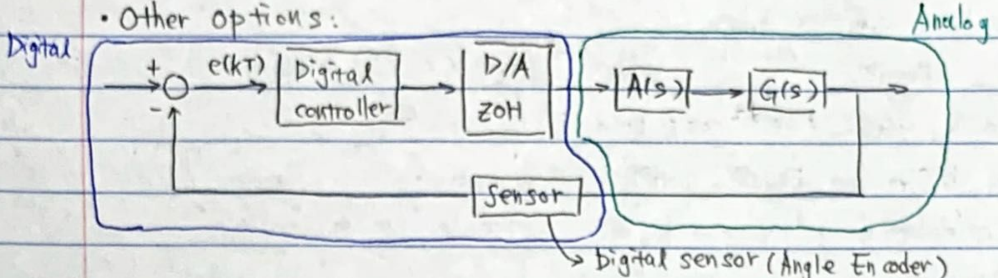
EEME E6601 Intro to Control Theory 10/19 ②

- In FOH, we extrapolate the information from the previous time step to estimate the slope at the given time step. , blue line in the graph.

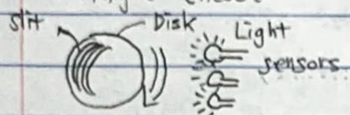
$$u(t) = u(kT) + \frac{u(kT) - u((k-1)T)}{T}(t-kT)$$

for $kT < t < (k+1)T$, Also for $0 \le t < T$: $u(t) = u(0)$

- Other options:



* Angle Encoder



* In discrete-time systems, we deal with difference equations, not differential equations:

For example: $y((k+2)T) + 3y((k+1)T) + 2y(kT) = 0 \;\; \forall k$ → Homogeneous.

- As opposed to: $\frac{d^2y(t)}{dt^2} + 3\frac{dy(t)}{dt} + 2y(t) = 0$

→ Initial Conditions: $y(0) = y_0$, $\frac{dy}{dt}\Big|_{t=0} = \dot{y}_0$, both known

- To go from O.D.E to Difference Equations, we "discretize".

→ Use backward, forward or Central Difference to approximate derivatives.

- For the O.D.E, let's try $y(t) = ce^{st}$

→ plug back in original O.D.E $\therefore (s^2+3s+2)ce^{st} = 0$.

→ $(s+1)(s+2) = 0$ → $s_1 = -1$, $s_2 = -2$.

→ $\therefore y(t) = c_1 \cdot e^{s_1 t} + c_2 e^{s_2 t} = c_1 e^{-t} + c_2 e^{-2t}$ (Apply Initial Conditions, Solve for c_1, c_2)

- Consider $y(k+2) + 3y(k+1) + 2y(k) = 0$ (dropped T)

→ try $y(kT) = y(k) = c \cdot e^{skT} = c \cdot (e^{sT})^k = c \cdot z^k$, where $z = e^{sT}$

→ substitute: $cz^{k+2} + 3cz^{k+1} + 2cz^k = 0$ (homogeneous still)

$\therefore c(z^2+3z+2)z^k = 0$ → $z^2+3z+2=0$ → $(z+2)(z+1)=0$ → $z_1 = -1$, $z_2 = -2$

→ so, there are two solutions $y_1(k) = c_1 z_1^k = c_1(-1)^k$ and $y_2(k) = c_2(-2)^k$

- In fact: $y_1(k+2) + 3y_1(k+1) + 2y_1(k) = c_1(-1)^{k+2} + 3c_1(-1)^{k+1} + 2c_1(-1)^k$, $\forall k$

$= c_1[(-1)^2 + 3(-1)^1 + 2](-1)^k = c_1(1-3+2)(-1)^k = 0$, $\forall k$

→ ~~Next page~~

- $Y_2(k+2) + 3Y_2(k+1) + 2Y_2(k) = \dots = C_2(4-6+2)(-2)^k = 0, \forall k$
- Since the difference equation is linear then, superposition holds: $y(k) = Y_1(k) + Y_2(k)$

$= C_1 z_1^k + C_2 z_2^k = C_1(-1)^k + C_2(-2)^k$

$\therefore [Y_1(k+2) + Y_2(k+2)] + 3[Y_1(k+1) + Y_2(k+1)] + 2[Y_1(k) + Y_2(k)] = 0, \forall k$

$\rightarrow y(k) = C_1(-1)^k + C_2(-2)^k \rightarrow$ General solution of Homogeneous Difference Equation

- what if you have complex roots?

$\rightarrow$ In ODE's : Recall: $e^{(\alpha+i\beta)t} = e^{\alpha t} e^{i\beta t} = e^{\alpha t}(\cos\beta t + i\sin\beta t)$

General solution: $y(t) = A \cdot e^{\alpha t}\cos\beta t + B \cdot e^{\alpha t}\sin\beta t$

* Homogeneous Difference Equation with Complex roots:

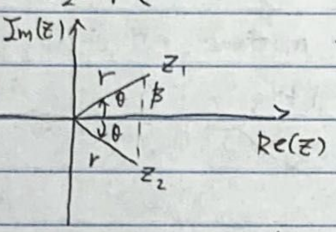
- characteristic equation: $z^2 + \beta_1 z + \beta_2 = 0 \rightarrow (z-z_1)(z-z_2) = 0$

if $z_1 = \alpha + i\beta$, $z_2 = \alpha - i\beta$

$\therefore y(k) = C_1 z_1^k + C_2 z_2^k = C_1(\alpha+i\beta)^k + C_2(\alpha - i\beta)^k$

$\rightarrow z_1 = \alpha + i\beta = re^{i\theta}$ (because $z_1 = r\cos\theta + ir\sin\theta$)

$z_2 = re^{-i\theta}$

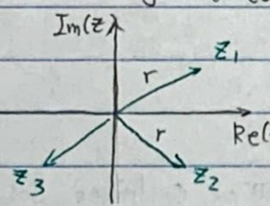


$\rightarrow$ polar representation's

- $y(k) = C_1 z_1^k + C_2 z_2^k = C_1(re^{i\theta})^k + C_2(re^{-i\theta})^k = (C_1+C_2)r^k\cos k\theta + (iC_1 - iC_2)r^k\sin k\theta$

$\therefore \boxed{y(k) = Ar^k\cos k\theta + Br^k\sin k\theta} \rightarrow$ Important to get θ correct. $\rightarrow$ General solutions for Homogeneous Difference Equations with complex roots

- How to get θ correct:



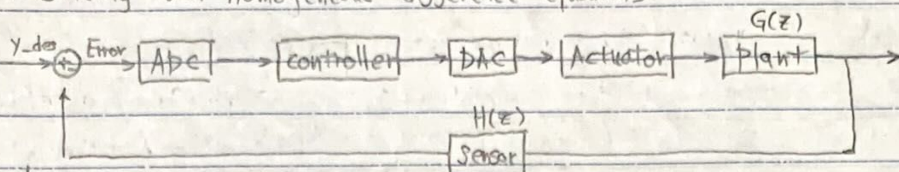
- $z_1 = \alpha + i\beta = rc\theta + irs\theta$
- Find magnitude r & angle θ

$r^2 = \alpha^2 + \beta^2$, $\theta = \tan^{-1}\left(\frac{\beta}{\alpha}\right)$

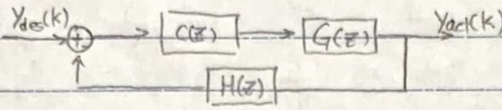
EEME E6601 Intro to Control Theory 10/26 ①

* Solving Non-homogeneous difference equations

we are in digital control world. →



convert



- Say, $y((k+1)T) + 2y(kT) = 1$, T: sampling period.

↳ Homogeneous solution associated with above problem:

$y((k+1)T) + 2y(kT) = 0$

$cz^{k+1} + 2cz^k = (z+2)cz^k = 0$.

→ characteristic equation: $z+2=0 \rightarrow z=-2 \rightarrow \therefore \boxed{y_{homogeneous} = c(-2)^k}$

- To find particular solution: ① first shift origin by 1, so $k \rightarrow k+1$

→ $y((k+2)T) + 2y((k+1)T) = 1$

② second, subtract original equation from the shifted equation.

$$
\begin{array}{rl}
 & y_{k+2} + 2y_{k+1} = 1 \\
- & y_{k+1} + 2y_k = 1 \\
\hline
 & y_{k+2} + y_{k+1} - 2y_k = 0.
\end{array}
$$

→ Now, we have a homogeneous difference equation (different from original homogeneous solution):

- Say, for this new homogeneous equation: $y_k = cz^k$ to find:

→ $y_{k+2} + y_{k+1} - 2y_k = (z^2+z-2)cz^k = 0$

→ characteristic equation: $z^2+z-2=0 \rightarrow (z-1)(z+2)=0$

→ roots are $z_1 = 1$, $z_2 = -2$ → already found to be a solution to the homogeneous equation of the original problem.

- Hence, root $z_1 = 1$ is the one that produces the new solution

→ $\therefore y_1(kT) = c_1 z_1^k = c_1(1)^k = c_1$

- To find c_1, substitute in original equation ($y_{k+1} + 2y_k = 1$)

→ $c_1 + 2c_1 = 1 \rightarrow \therefore c_1 = \frac{1}{3} \rightarrow \boxed{y(kT) = c(-2)^k + \frac{1}{3}}$

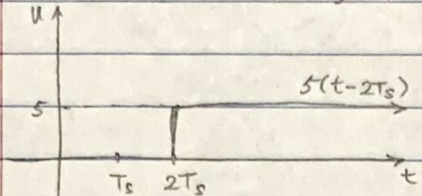
* To find particular solution of

- O.D.E. (done in H1 and Lecture 1)
- Difference equation: first shift forward one step

  second, subtract the original equation.

* z - transform: just as Laplace transform is a generalization of the Fourier transform for continuous time signals, the z-transform is its parallel for discrete_time (digital) signals.

- Fourier transform: Specturum content → combination of sin() functions to define any signals.
- $\boxed{U(z) = z[u(kT)] = \sum_{k=0}^{\infty} u_k \cdot z^{-k} \text{ (1 sided)}}$, $\sum_{k=-\infty}^{\infty} u_k z^{-k}$ (2 sided) → Not here x.
- $u_k \cdot z^{-k} = u((k-1)T)$, $u_k z^{+k} = u((k+1)T)$



- z^{-1}: time shift of $-1 T_s$ → backward time shift by 1
- z^{+1}: time shift of $+1 T_s$ → forward time shift by 1

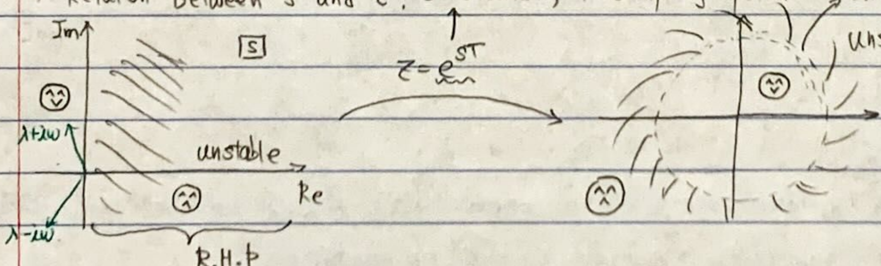
- $\underbrace{\sum_{k=0}^{\infty} u(k) z^{-k}}_{U(z)} = u(0) + u(1) z^{-1} + u(2) z^{-2} + \cdots$ (*). forward time shift by nT steps.

- Some properties of this transform:

① Convolution: $z[x(k) * h(k)] = X(z) H(z)$

② Time shift: $z[\underbrace{x(k+n)}_{(*)}] = z^{n} \cdot X(z)$, $z[x(k-n)] = z^{-n} \cdot X(z)$

* Relation between s and z: s: s domain, T: sampling period



- # in z-domain are generally small.

$H(z) = \frac{1}{(z-0.05)(z-0.8995)}$

- 2 poles in s-domain, transformed to z-domain give:

$z = e^{sT} = e^{(\lambda + i\omega)T} = e^{\lambda}[\cos(\omega T t) + i \sin(\omega T t)]$

→ As you change gain k, the unit circle shrink and expand

* If mag of z is unity → ∴ $z = 1(\cos(\omega T t) + i \sin(\omega T t))$

Here's a z-transform table:

(**). $z[1(kT)] = \sum_{k=0}^{\infty} 1(k) z^{-k} = 1 + z^{-1} + z^{-2} + \cdots$

	$y(t)$	$Y(s) = \mathcal{L}[y(t)]$	$Y(z) = z[y(kT)]$
step	1	$\frac{1}{s}$	$\frac{z}{z-1}$ $\left(= \frac{1}{1-z^{-1}}\right.$ in negative points$\left.\right)$ (**)
ramp	t	$\frac{1}{s^2}$	$\frac{zT}{(z-1)^2}$
exponential	e^{-at}	$\frac{1}{s+a}$	$\frac{z}{z-e^{-at}}$
sinusoid	$\sin(\omega t)$	$\frac{\omega}{s^2+\omega^2}$	$\frac{z(\sin(\omega t))}{z^2 - 2z\cos(\omega T) + 1}$

EEME E6601 Into to Control Theory 10/26. ②

* Z-transforms of exponential function

- Say, $x_k = e^{-akT}$ (in continuous time $x(t) = e^{-at}$

$\rightarrow Z\{x_k\} = \sum_{k=0}^{\infty} x(kT) \cdot z^{-k} = 1 + e^{-aT} \cdot z^{-1} + e^{-2aT} \cdot z^{-2} + \cdots = \frac{1}{1-e^{-aT}z^{-1}} = \frac{z}{z-e^{-aT}}$

- Say, $4\dot{y} + 5y = 0$, discretize it:

Recall : Definitions of derivatives

$\frac{dy}{dt} = \lim_{\Delta t \to 0} \frac{y(t+\Delta t) - y(t)}{\Delta t} \rightarrow$ Forward Difference

Also : $= \lim_{\Delta t \to 0} \frac{y(t) - y(t-\Delta t)}{\Delta t} \rightarrow$ Backward Difference

Also : $= \lim_{\Delta t \to 0} \frac{y(t+\Delta t) - y(t-\Delta t)}{2\Delta t} \rightarrow$ Central Difference.

- As $\Delta t \to 0$, all 3 should give similar results
- Here Δt is T (or T_s, the sampling period). So, our ODE becomes

$\rightarrow \frac{4(y(k+1) - y(k))}{T} + 5y(k) = 0 \rightarrow \therefore 4(y(k+1) - y(k)) + 5y(k) = 0$ (say unit sampling $T = 1$ sec)

$\rightarrow$ if not unit sampling, $\frac{4y(k+1)}{T} - \frac{4y(k)}{T} + 5y(k) = 0$

$\frac{4}{T} y(k+1) + y(k)\left(5 - \frac{4}{T}\right) = 0 \rightarrow$ If $T = 1 \rightarrow \therefore 4y_{k+1} + y_k = 0$

* Examples : $y(k+2) + 0.25y(k+1) - 0.375y(k) = 2u(k+1) + u(k)$

- See that : $G(z) = \frac{Y(z)}{U(z)} = \frac{2z+1}{z^2 + 0.25z - 0.375} \rightarrow$ MATLAB in tf form

- $G(z) = \frac{2(z+0.5)}{(z+0.75)(z-0.5)} \rightarrow$ MATLAB in zpk form.

- Use a 3rd argument for T for discrete systems :

```
>> G = tf([2 1], [1 0.25 -0.375], 0.25)
```

* The Final Value Theorem in continuous time.

- $y(t\to\infty) = \lim_{s\to 0} s \cdot Y(s)$, where $\frac{Y(s)}{U(s)} = G(s)$

steady-state $\leftarrow$ - $y_{ss}(\infty) = \lim_{z\to 1} \left(\frac{z-1}{z}\right) Y(z) \rightarrow$ proof will be posted in canvas

* In continuous time), $\dot{x} = Ax + Bu$

- $sX - Ax = Bu \rightarrow x(SI - A) = Bu \rightarrow \frac{X}{U} = (SI - A)^{-1}B$

and also if $y = cx + Du$ then what is $\frac{Y(s)}{U(s)} = ?$

$\rightarrow Y(s) = C(SI - A)^{-1}BU + Du$

$\therefore \boxed{\frac{Y(s)}{U(s)} = C(SI - A)^{-1}B + D}$

* Let's define the state-transition matrix $\Phi(t)$ on an $n \times n$ matrix that satisfies the homogeneous equation: $\dot{x} - Ax = 0$

- It satisfies the equation: $\frac{d\Phi(t)}{dt} = A\Phi(t)$
- If $x(0)$ is initial condition, $x(t) = \Phi(t) \cdot x(0)$

From $\dot{x} = Ax \rightarrow sX(s) = AX(s) \rightarrow$ If 0 initial condition

- If initial condition $\neq 0$ then $sX(s) - x(0) = Ax(s)$

$X(s) = (SI - A)^{-1}x(0)$

- but $x(t) = \mathcal{L}^{-1}[x(s)] \rightarrow \therefore x(t) = \mathcal{L}^{-1}(SI - A)^{-1} \cdot x(0)$
- In terms of $\Phi(t)$: $x(t) = \Phi(t) \cdot x(0)$

$\rightarrow \Phi = \mathcal{L}^{-1}[(SI - A)^{-1}]$, $x(t) = e^{At} \cdot x(0)$ → exponential matrix in MATLAB expression

- $e^{At} = I + At + \frac{A^2t^2}{2!} + \frac{A^3t^3}{3!} + \cdots$

$\rightarrow$ definition of e^t: $e^t = 1 - t + \frac{t^2}{2!} + \frac{t^3}{3!} + \cdots$

- Again: $\Phi = \mathcal{L}^{-1}[(SI - A)^{-1}]$, $\Phi = e^{At} = I + At + \frac{A^2t^2}{2!} + \frac{A^3t^3}{3!} + \cdots = x(t - t_0)\, x(0)$
- Recall that the full equation (non-homogeneous) is $\dot{x} = Ax + Bu$ (1st order O.D.E.)

$$\boxed{x(t) = \underbrace{[\mathcal{L}^{-1}(SI - A)^{-1}]\, x(0)}_{\text{homogeneous}} + \underbrace{\mathcal{L}^{-1}[(SI - A)^{-1}BU(s)]}_{\text{particular}}} \quad (sX(s) - x(0) = Ax + Bu)$$

$\Rightarrow$ Full solution is

$$x(t) = \Phi(t)\, x(0) + \int_0^t \Phi(t - \tau)\, Bu(\tau)\, d\tau$$

EEME E6601 Intro to Control Theory 11/2 ①, state space form (state equation)

* continuous time (CT) : $\dot{x} + Ax = Bu$ → first order system (the basic or atomic expression of system)

Solution for the state equation as a function of Φ

· $\boxed{x(t) = \Phi(t) \cdot x(0) + \int_0^t \Phi(t-\tau) B \cdot u(\tau) d\tau}$ → solution to above equation

· If homogeneous ($u=0$), then solution to homogeneous state equation is :

$\boxed{x(t) = \Phi(t) x(0)}$ → Φ : state transition matrix, $x(0)$: intial condition.

Solution to state equation

· $y(t) = Cx(t) + Du(t)$ → observation equation → there is no dynamics in the equation because it's just algebraic equation ($\dot{x} + Ax = Bu$ is differential equation and contains information of dynamics)

$$\therefore y(t) = C\Phi(t)x(0) + \int_0^t C\phi(t-\tau)B \cdot u(\tau) + Du(\tau) d\tau$$

* Digital time (DT) :

$x(k+1) = Ax(k) + Bu(k)$: state equation

$y(k) = Cx(k) + Du(k)$: observe equation

※ $x_{k+1} = x((k+1)T_s)$

* DT $\begin{cases} x(k+1) = A_d x(k) + B_d u(k) \\ y(k) = C_d x(k) + D_d u(k) \end{cases}$

CT $\begin{cases} \dot{x} = A_c x(t) + B_c u(t) \\ y = C_c x(t) + D_c u(t) \end{cases}$

$\boxed{\text{Hold}}$:

· z-transform of $x(k+1)$, $x(k+2)$, $x(k+n)$, $x(k-n)$

· $x(k+n)$: sequence shifted to the left by n sampling periods

· $Z\{x(k+1)\} = \sum_{k=0}^{\infty} x(k+1) \cdot z^{-k}$

$= x(1) \cdot z^0 + x(2) \cdot z^{-1} + x(3) \cdot z^{-2} + \dots$ → same as $\sum_{k=1}^{\infty} x(k) \cdot z^{-k+1}$

$\therefore Z\{x(k+1)\} = \sum_{k=1}^{\infty} x(k) \cdot z^{-k} \cdot z = z[x(z) - z x(0)] = z[\sum_{k=0}^{\infty} x(k) \cdot z^{-k} - x(0)]$

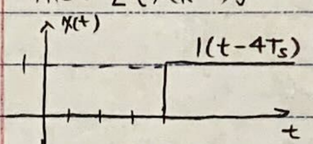
· Again, if $x(0) = 0$, $Z\{x(k+1)\} = zX(z)$ → forward shift in time

· similary, $Z\{x(k+2)\} = z^2 X(z) - z^2 x(0) - z x(1)$

→ $Z\{x(k+n)\} = z^n x(z) - z^n x(0) - z^{n-1} x(1) - z^{n-2} x(2) - \dots - z x(n-1)$

↳ forward shift by n in time

· Also $Z\{x(k-n)\} = z^{-n} x(z)$



→ Initial conditions are automatically included

→ backward Shift.

· E.g. $Z\{1(t-T_s)\} = z^{-1} Z\{1(t)\} = z^{-1} \cdot \frac{1}{1-z^{-1}} = \frac{z^{-1}}{1-z^{-1}}$

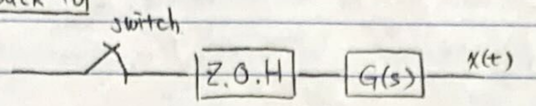
* For example:

$Z\{1(t-4T_s)\} = \frac{z^{-4}}{1-z^{-1}}$

$\rightarrow Z\{a^k\} = \frac{1}{1-az^{-1}}, \quad k=1,2,3,\ldots$

$\boxed{\text{Back to}}$

open loop system. ←



→ switch works at certain frequency.

* Discretization: (S|H) sample & hold of an open loop system.

- Say a system is described in state-space as:

$$\underbrace{\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix}}_{2\times1} = \underbrace{\begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix}}_{A_c \; 2\times2} \underbrace{\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}}_{2\times1} + \underbrace{\begin{bmatrix} 0 \\ 1 \end{bmatrix}}_{B_c \; 2\times1} \underbrace{u(k)}_{1\times1}$$

* Discretize the system

→ By using Φ

$\Phi = \mathcal{L}^{-1}[(SI-A)^{-1}]$ because: $\dot{x} = Ax + Bu$ (CT)

$SX(s) - x(0) = AX(s) + BU(s) \rightarrow \therefore X(s)[SI-A] = x(0) + BU(s)$

$\rightarrow X(s) = (SI-A)^{-1}x(0) + (SI-A)^{-1}BU(s)$

* The inverse Laplace gives me $x(t)$ → time.

$\therefore x(t) = \mathcal{L}^{-1}(SI-A)^{-1}x(0) + \mathcal{L}^{-1}(SI-A)BU(s)$

- If homogeneous, then $u(t) = 0 = U(s)$, then $\Phi(t) = \mathcal{L}^{-1}(SI-A)^{-1}x(0)$

→ you need to know how to take inverse of 1x1, 2x2, 3x3 matrices.

* 2x2 matrix example: $\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} = A \rightarrow A^{-1} = \frac{1}{\det(A)} \begin{bmatrix} a_{22} & -a_{12} \\ -a_{21} & a_{11} \end{bmatrix}$

* 3x3 matrix example: follow Cramer's Rule.

* Back to the example: need to take $\mathcal{L}^{-1}[(SI-A)^{-1}]$

- Step 1: Form SI-A → $SI-A = \begin{bmatrix} s & -1 \\ 2 & s+3 \end{bmatrix}$

- Find $(SI-A)^{-1} = \frac{1}{s^2+3s+2} \begin{bmatrix} s+3 & 1 \\ -2 & s \end{bmatrix}$

- $\Phi(t) = \mathcal{L}^{-1}[(SI-A)^{-1}] = \begin{bmatrix} \frac{s+3}{s^2+3s+2} & \frac{1}{s^2+3s+2} \\ \frac{-2}{s^2+3s+2} & \frac{s}{s^2+3s+2} \end{bmatrix}$ → partial fraction then use Laplace Table

- $\Phi(t) = \mathcal{L}^{-1}[(SI-A)^{-1}] = \begin{bmatrix} 2e^{-t} - e^{-2t} & e^{-t} - e^{-2t} \\ -2e^{-t} + 2e^{-2t} & -e^{-t} + 2e^{-2t} \end{bmatrix}$

EEME E6601 Intro to Control Theory 11/2 ②

$\boxed{\text{Hold}}$

$\Phi(t) = e^{At}$ (different methods to compute it)

$\Phi = \mathcal{L}^{-1}[(SI-A)^{-1}]$

Also, Φ can be written in terms of the eigenvalues of the system

- If A is diagonal then $e^{At} = \begin{bmatrix} e^{\lambda_1 t} & & & 0 \\ & e^{\lambda_2 t} & & \\ & & \ddots & \\ 0 & & & e^{\lambda_n t} \end{bmatrix}$

- If λ_1 has multiplicity of 3, say then

$$e^{At} = \begin{bmatrix} e^{\lambda_1 t} & & & \\ & te^{\lambda_1 t} & & \\ & & t^2 e^{\lambda_1 t} & \\ & & & \ddots \\ & & & & e^{\lambda_n t} \end{bmatrix}, \quad e^{At} = I + At + \frac{A^2t^2}{2} + \frac{A^3t^3}{3} + \dots$$

- $\dot{x} = Ax + Bu$, eig(A)
- $A_d = \Phi_c(T_s) = e^{AT_s} = I + \frac{A_d T_s}{1} + \frac{A_d^2 T_s^2}{2!} + \frac{A_d^3 T_s^3}{3!} + \dots$
- $B_d = \left[\int_0^{T_s} \Phi(t-\tau)\,d\tau\right] B_c$

$$= \left[I \cdot T_s + \frac{A_c T_s^2}{2!} + \frac{A_c^2 T_s^3}{3!} + \frac{A_c^3 T_s^4}{4!} + \dots \right] B_c$$

- $C_d = C_c$, $D_d = D_c$

$\boxed{\text{Back to}}$

- Need to find:

$$\begin{cases} x_1((k+1)T_s) = \dots \\ x_2((k+1)T_s) = \end{cases}$$

- Hence : $B_d = \int_0^{T_s} \Phi_c(T_s - \tau)\,d\tau$

$$= \int_0^{T_s} \begin{bmatrix} e^{-(T_s-\tau)} - e^{-2(T_s-\tau)} \\ -e^{-(T_s-\tau)} + 2e^{-2(T_s-\tau)} \end{bmatrix} d\tau = \underbrace{\begin{bmatrix} 0.5 - e^{-T_s} + 0.5e^{-2T_s} \\ e^{-T_s} - e^{-2T_s} \end{bmatrix}}_{B_d \text{ or can use } B_d \text{ formula}}$$

- $B_d = I \cdot T_s + \frac{A_c T_s^2}{2} + \dots$

$$\therefore \begin{cases} x_1((k+1)T_s) \\ x_2((k+1)T_s) \end{cases} = \underbrace{\begin{bmatrix} 2e^{-T_s} - e^{-2T_s} & -e^{-T_s} - e^{-2T_s} \\ -2e^{-T_s} + 2e^{-2T_s} & -e^{-T_s} + 2e^{-2T_s} \end{bmatrix}_{2\times2}}_{A_d} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0.5 - e^{-T_s} + 0.5e^{-2T_s} \\ e^{-T_s} - e^{-2T_s} \end{bmatrix}_{2\times1} u(k)_{1\times1}$$

→ In MATLAB

```
[numc, denc] = ss2tf(A,B,C,D)
(Md, dd) = c2d (numc, denc) % continuous to discrete!
(Ad, Bd, Cd, Dd) = tf2ss(Md, dd)
```

→ Next page

* Exercise 2.1C
Read chp 5
Example script
5.7 discretization

· Also, tf with a T_s assumes a z-domain TF.
" without a T_s assumes an s-domain TF.

* $A_c = \begin{bmatrix} -4 & +3 \\ 1 & 0 \end{bmatrix}$, $B_c = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$, $C_c = \begin{bmatrix} 1 \\ 5 \end{bmatrix}$, $D_c = 0$

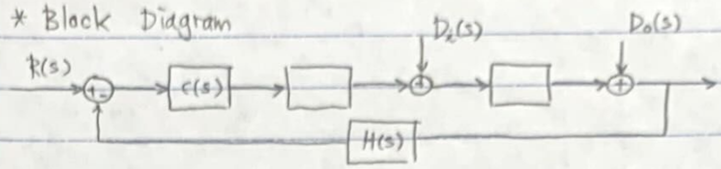
· $G_p = ss(A_c, B_c, C_c, D_c)$
$T_s = 0.1$
$G_{z\_ZOH} = c2d(G_p, T_s, \text{'ZOH'})$

* $e(t) \xrightarrow[T_s]{} e(kT_s) \rightarrow \boxed{ZOH} \rightarrow \boxed{G_p(s)} \rightarrow$

* while discretizing, we can face some issues like aliasing

EEME E6601 Intro to Control Theory 11/9 ①

* Block Diagram



- Fullstate feedback

Control Law → • $u = -Kx$, x: states

- $u = -(k_1 x + k_2 \dot{x} + k_3 \theta + k_4 \dot{\theta})$

* $Z\{C(z)\}$ → Difference equation

- $u(k) = -(x_1(k+1) + x_2(k) + x_3(k+1) + x_4(k))$

* C & D matrices of SS models → C & D matrices are related to sensor because 'y' output depends on sensor datas

- $y = CX + DU$
- Say, $\underbrace{M\ddot{w} + \xi\dot{w} + kw = f(w,t)}_{eq(1)}$ → general 2^{nd} order system. (For Meche: M: mass, ξ: damping ratio, k: stiffness)

↳ Get SS model (many ways)

- $A_c = \begin{bmatrix} 0 & I \\ -M^{-1}K & -M^{-1}\xi \end{bmatrix}$: state matrix, $x = \begin{Bmatrix} w \\ \dot{w} \end{Bmatrix}$, $n = 2n_2$ → # 2^{nd} order ODE, $f(w,t) = B_2 u(t)$

  # 1^{st} order ODE.

- $\dot{x} = A_c x + B_c u$
- Say, using m sensors (accelermeter, tachometer, straingage) then,

  $y = C_a\ddot{w} + C_v\dot{w} + C_d w$ → eq(2), y: matrix output equation

  → C_a, C_v, C_d : output influence matrices

Solve for $\ddot{w}$ in eq(1) and substitute in eq(2) gives:

D: feed forward matrix. → $y = C_a M^{-1}[B_2 u - \xi\dot{w} - kw] + C_v\dot{w} + C_d w$ or

$y = Cx + Du$ → format then: ↳ direct transmission matrix

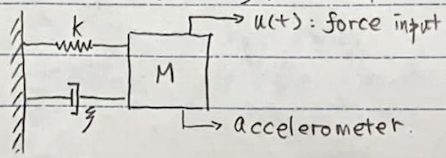
C_a: accelerometer → $C = [C_d - C_a M^{-1}K \quad C_v - C_a M^{-1}\xi]$ and $D = C_a M^{-1} B_2$.

- Example: Say x translation mechanical system with mass, stiffness, and damping characteristic. If the actuator is a force type and the sensor is an accelerometer, what is the SS model of this system?

→ Recall: $A_c \neq A_d$, $B_c \neq B_d$, $C_c = C_d$, $D_c = D_d$

→ E.O.M.: $M\ddot{w} + \xi\dot{w} + kw = B_2 u(t)$, $B_2 = 1$.

Take $M = 1$, $k = 1$, $\xi = 0.01$



• $z_1 = \dot{\omega}$ $\qquad\qquad$ $\therefore$ $z_1 = \dot{\omega} + 0u$

$z_2 = -\zeta\dot{\omega} - \omega + u$ $\qquad$ $z_2 = -1\omega - 0.01\dot{\omega} + u$

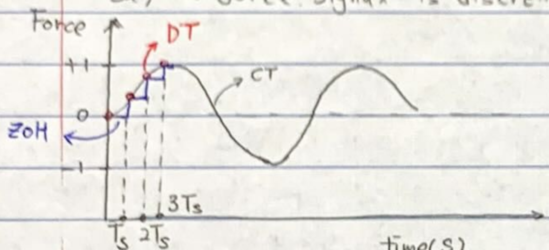
$$\dot{x} = \begin{bmatrix} \dot{\omega} \\ \ddot{\omega} \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 \\ -1 & -0.01 \end{bmatrix}}_{A} \begin{bmatrix} \omega \\ \dot{\omega} \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ 1 \end{bmatrix}}_{B} u$$

$$y = \underbrace{\begin{bmatrix} -1 & -0.01 \end{bmatrix}}_{C} \begin{bmatrix} \omega \\ \dot{\omega} \end{bmatrix} + \underbrace{1}_{D} \cdot u \quad (\text{take } C_a = 1)$$

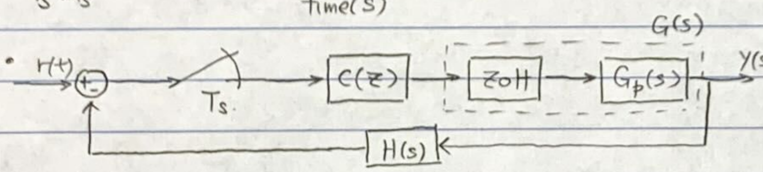
→ So, SS model has such A, B, C, D.

* Let's go back to DT systems.

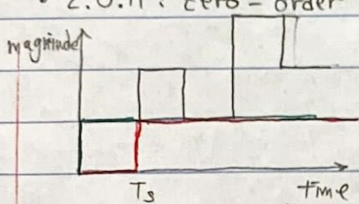
• Say a force signal is discretized.



• ZOH allows to form continuous line (stairs) from the DT.



• Z.O.H : Zero - order hold



• ZOH = green − red

$$\frac{1}{s} - \frac{e^{-sT_s}}{s}$$

$$\therefore G_{ZOH}(s) = \frac{1 - e^{-sT_s}}{s}$$

• If no $\underset{\sim}{S/H}$ device between $G_p(s)$ (plant) and $H(s)$ (sensor) then can express them together $\therefore \overline{GH}(z) = H(z) \cdot G(z)$

(S/H → sample/hold → For discreted analog system like above diagram)

• Also, define $G(z) \triangleq \mathcal{Z}\{G_p(s) \cdot G_{ZOH}(s)\} = \mathcal{Z}\left\{\frac{1 - e^{-sT_s}}{s} \cdot G_p(s)\right\}$

EEME E6601 Intro to Control Theory 11/9 ②

$\boxed{\text{Hold}}$

- ADC - Analog to Digital Converter

Analog signal → $\boxed{\text{Sample}}$ → $\boxed{\text{Hold}}$ → $\boxed{\text{Quantize}}$ → $\boxed{\text{Encoder}}$ → Digital signal

$\boxed{\text{Back to}}$

so, $G(z) = \mathcal{Z}\{(1-e^{-sT_s}) \cdot \frac{G_p(s)}{s}\} = (1-z^{-1})\,\mathcal{Z}\{\frac{G_p(s)}{s}\}$ always true

- Again, $\mathcal{Z}\{a^k\} = \sum_{k=0}^{\infty} a^k \cdot z^{-k} = 1 + az^{-1} + a^2 z^{-2} + \dots = \frac{1}{1-az^{-1}}$ (also $= \frac{z}{z-a}$)
- $\mathcal{Z}\left[\frac{1}{s+a}\right] = \frac{1}{1-e^{-aT_s} z^{-1}}$ (also $= \frac{z}{z-e^{aT_s}}$)
- Most CL methods still work in the z-domain

* Stability: Routh-Hurwits → Jury stability, TEST

- Jury: Method to determine the location of roots of a z-polynomial.
- Say, $F(z) = a_m z^m + a_{m-1} z^{m-1} + \dots + a_1 z + a_0 = 0$.

→ a's : real coeffs

→ Assume $a_m > 0$.

- Jury table.

Row	z^0	z^1	z^2
1	a_0	a_1	a_2
2	a_m	a_{m-1}	a_{m-2}
3	b_0	b_1	b_2
4	b_{n-1}	b_{n-2}	b_{n-3}
5	c_0	c_1	c_2

→ where, $b_k = \begin{vmatrix} a_0 & a_{n-k} \\ a_n & a_k \end{vmatrix}$, $c_k = \begin{vmatrix} b_0 & b_{n-1-k} \\ b_{n-1} & b_k \end{vmatrix}$, $d_k = \begin{vmatrix} c_0 & c_{n-2-k} \\ c_{n-2} & c_k \end{vmatrix}$

→ Necessary sufficient conditions for $F(z)$ to have no roots outside the unit circle are:

$F(1) > 0$, $F(-1) \begin{cases} > 0 & \text{if } n \text{ even} \\ < 0 & \text{if } n \text{ odd} \end{cases}$, $|a_0| < a_n$, $|b_0| > |b_{n-1}|$, $|c_0| > |c_{n-2}|$, $|d_0| > |d_{n-3}|$, $|q_0| > |q_2|$

EEME E6601 Intro to Control Theory 11/16 ①

* More on the transition state matrix (using the $\mathcal{Z}^{-1}$)

- $x(k+1) = Ax(k) + Bu(k)$: state equation (in S.S. form)

→ solution in terms of $\Phi(k)$: state transition matrix is:

$x(k) = \Phi(k)x(0) + \sum_{i=0}^{k-1} \Phi(k-1-i)Bu(i)$ ··· Eq (1)

→ A: state matrix, B: input matrix.

- $\Phi(k) = A^k$

- Z-transform of state equation with non-zero initial condition

→ $zX(z) - zx(0) = Ax(z)$: Assume homogeneous

- Transfer function is $\frac{\text{output}}{\text{input}}$ → $\therefore \underbrace{X(z)}_{\text{output}} = z[zI-A]^{-1} \underbrace{x(0)}_{\text{input}}$

$\therefore x(k) = \mathcal{Z}^{-1}[X(z)] = \mathcal{Z}^{-1}[z[zI-A]^{-1}]x(0)$ ··· Eq(2)

- Comparing eq(2) with eq(1) gives:

$\Phi(k) = \mathcal{Z}^{-1}[z[zI-A]^{-1}]$

* Example: say, $A = \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix}$ → $\therefore zI - A = \begin{bmatrix} z & 0 \\ 0 & z \end{bmatrix} + \begin{bmatrix} 0 & -1 \\ 2 & 3 \end{bmatrix} = \begin{bmatrix} z & -1 \\ 2 & z+3 \end{bmatrix}$

- determinant : $|zI - A| = z^2+3z+2 = (z+1)(z+2)$

- $[zI-A]^{-1} = \begin{bmatrix} \frac{z+3}{(z+1)(z+2)} & \frac{1}{(z+1)(z+2)} \\ \frac{-2}{(z+1)(z+2)} & \frac{z}{(z+1)(z+2)} \end{bmatrix}$, $z[zI-A]^{-1} = \begin{bmatrix} \frac{z(z+3)}{(z+1)(z+2)} & \frac{z}{(z+1)(z+2)} \\ \frac{-2z}{(z+1)(z+2)} & \frac{z^2}{(z+1)(z+2)} \end{bmatrix}$

- $\Phi(k) = \mathcal{Z}^{-1}[z(zI-A)^{-1}]$ → can do $\mathcal{Z}^{-1}$ by going through the P.F.E.

→ $z(zI-A)^{-1} = \begin{bmatrix} \frac{2z}{z+1} + \frac{-z}{z+2} & \frac{z}{z+1} + \frac{-z}{z+2} \\ \frac{-2z}{z+1} + \frac{2z}{z+2} & \frac{-z}{z+1} + \frac{2z}{z+2} \end{bmatrix}$

$\therefore \mathcal{Z}\left[z[zI-A]^{-1}\right] = \begin{bmatrix} 2(-1)^k - (-2)^k & (-1)^k - (-2)^k \\ -2(-1)^k + 2(-2)^k & -(-1)^k + 2(-2)^k \end{bmatrix} = \Phi(k)$

- Then given $x(0) = \begin{bmatrix} x_1(0) \\ x_2(0) \end{bmatrix}$, can calculate solution at any time k by

$\begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix}_{2\times1} = \Phi(k)_{2\times2} \cdot \begin{bmatrix} x_1(0) \\ x_2(0) \end{bmatrix}_{2\times1}$

- Further notes on Z-domain in closed loop:

L(z) : OLTF , $1+L(z)=0$ (denominator of "gang of four")

- Again, DT. SS system:

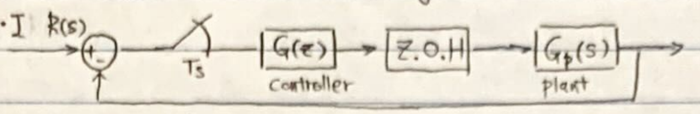
→ $x(k+1) = Ax(k) + Bu(k)$

$y(k) = Cx(k) + Du(k)$

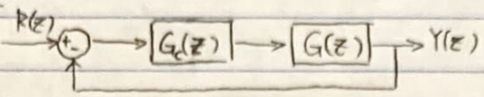
linear

* Draw the feedback block diagram (say, unity feedback)

• I



• II



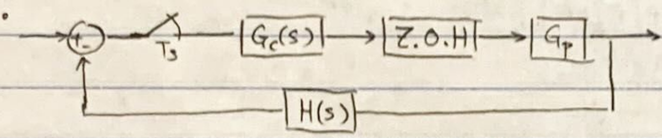
→ where, $\frac{Y(z)}{R(z)} = C[zI - A]^{-1}B + D$

→ still unity feedback

→ where $G(z) = (1 - z^{-1})\, Z\left[\frac{G_p(s)}{s}\right]$ (Combined Z.O.H and $G_p(s)$)

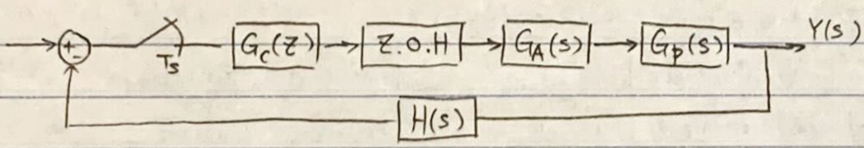
↳ There are no sample/hold between plant and sensor (can combine them).

•



→ then $G(z) = (1 - z^{-1})\, Z\left[\frac{H(s)G_p(s)}{s}\right]$

• If further we have $G_A(s)$: Actuator T.F.



→ $G(z) = (1 - z^{-1}) \cdot Z\left[\frac{H(s) \cdot G_p(s) \cdot G_A(s)}{s}\right]$

• Since $z = e^{sT_s}$ then, straight line approximations of the Bode don't apply anymore.

• can still design $G_c(z)$ using root locus and Nyquist and even Bode but without the straight line approximations.

• Some of the frequency domain, characteristics & stability property can be lost in the z-domain.

• Then wouldn't it be nice if somehow we can design in the continuous-time domain and easily convert it to discrete?

→ 4 ways to convert between c & d

1) Matched z-transform

2) Impulse-invariant

3) Tustin (bilinear)

4) Tustin with frequency, pre-warping

→ only # 3 preserves frequency domain and stability properties

EEME E6601 Intro to Control Theory 11/16 ②

* between discrete and continuous

→ Also, nowadays, controller repeats.

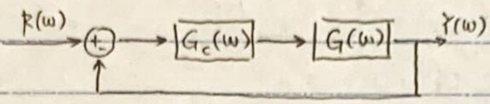
→ may not specify step response

→ rise time, overshoot, settling time, etc..

→ They may well include robustness and frequency domain requirement such as GM, PM, etc...

• For all those reasons, lets convert from $\boxed{z}$ to $\boxed{w}$ (w-domain) since it preserves frequency / stability properties etc.

$$z = \frac{1+\left(\frac{T_s}{2}\right)w}{1-\left(\frac{T_s}{2}\right)w} \quad \rightarrow \text{ solving for } w, \text{ we get } w = \frac{2}{T_s}\cdot\frac{z-1}{z+1}$$

III. 

→ say, with unity actuator + feedback

• In MATLAB: << d2c or c2d(TF', tustin')

→ So can design via LSD (loop shape design)

• According to the rules articulated in class and summarized in HW 4

* Example (z to w):

Say, $$G(z) = \frac{z-1}{z}\left[\frac{(e^{-T_s}+T_s-1)z^2+(1-e^{-T_s}-T_s e^{-T_s})z}{(z-1)^2(z-e^{-T_s})}\right] = \frac{0.00464z+0.00468}{(z-1)(z-0.905)}, \quad T_s = 0.1\,\text{sec}$$

• what is $G(w)$?

$$G(w) = G(z)\Big|_{z=\left[\frac{1+\left(\frac{T_s}{2}\right)w}{1-\left(\frac{T_s}{2}\right)w}\right]}$$

$$\therefore G(w) = \frac{-0.000420w^2-0.0491w+1}{w^2+0.997w}$$

• So in MATLAB:

```
Ts = 0.1
numz = [0.00464  0.00468]
denz = [1  -1.905  0.905]
Gz = tf(numz, denz, Ts)
Gw = d2c(Gz, 'tustin')
```

→ gives it to you in 's'!!

• Treat it exactly like it is in the time (CT) domain

• At the end, convert $G_c(w)$ to $G_c(z)$.

• Code its difference equations on a computer.

* pole placement: place CL poles in desired locations

>> acker] → MATLAB
>> place.

- Full. state feedback controller, i.e. the control law: $u = -kx$ (k: gain matrix, x: state vector)
- we assume that all the states are observable, i.e. measured
- So, plant has to be expressed in state-space format

* Example

$$x(k+1) = \begin{bmatrix} 1 & 0.0951 \\ 1 & -0.0905 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} + \begin{bmatrix} 0.00483 \\ 0.09516 \end{bmatrix} u(k), \quad y(k) = [1 \ 0] x(k)$$

- Since $u(k) = -k x(k)$ and $x(k+1) = A x(k) + B u(k)$

then $x(k+1) = A x(k) - Bk x(k) = (A - BK) x(k)$, where $k = [k_1, k_2, \dots, k_n]$, n: # states

- we know that $zI - A$ is characteristic equation.
- Now, we have $A - Bk$ as the state matrix but since it's in CL then CL characteristic equation is $\alpha_c(k)$: CL characteristic equation.

→ $\alpha_c(k) = |zI - (A - Bk)| = zI - A + Bk = (z - \lambda_1)(z - \lambda_2) \cdots (z - \lambda_n)$, ($\lambda_1, \lambda_2 \dots \lambda_n$: poles)

- controllability is a pre-condition!
  ↳ do you have enough actuators to reach from A to B.

EEME E6601 Intro to Control Theory 11/30 ①

* H4 : $A(s) = \frac{10}{s}$ (as per code)

• s, z, w

$$C(z) \equiv \frac{U(z)}{E(z)} = \frac{(z-0.1)(z+0.5)}{(z+0.9)(z-0.8)} = \frac{z^2+0.4z-0.5}{z^2+0.1z-0.72} = \frac{1+0.4z^{-1}-0.5z^{-2}}{1+0.1z^{-1}-0.72z^{-2}}$$

• $-0.72\,u(k-2) + \cdots = -0.5\,e(k-2) + \cdots \rightarrow u(k) = \cdots$

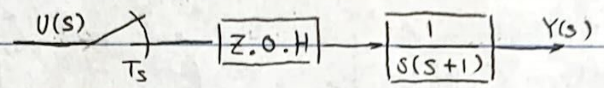
* Culprits : Sensor delay

* pole placement

Modern Control → world

→ here, we do not need to use loop shape method to design a controller. But can use state space → representation (Another way of designing a controller)

• say,



• 2^{nd} order model

$$\underset{2\times1}{x(k+1)} = \underset{2\times2}{\begin{bmatrix} 1 & 0.095163 \\ 0 & 0.904837 \end{bmatrix}} \underset{2\times1}{x(k)} + \underset{2\times1}{\begin{bmatrix} 0.0048374 \\ 0.095163 \end{bmatrix}} \underset{1\times1}{u(k)}$$

$y(k) = [1 \; 0]\, x(k)$, say position angle is measured. Control Law is a linear combination of the states

• $u(k) = -k\,x(k)$, k: gain matrix, $x(k)$: state.

• fullstate feedback

$u(k) = -k_1x_1(k) - k_2x_2(k) = -k\,x(k)$, where $k = [k_1 \; k_2]$

• Substitute into state equation

$$\therefore x(k+1) = \begin{bmatrix} 1 & 0.095163 \\ 0 & 0.904837 \end{bmatrix} x(k) - \begin{bmatrix} 0.0048374 \\ 0.095163 \end{bmatrix} [k_1x_1(k) + k_2x_2(k)]$$

$$= \underbrace{\begin{bmatrix} 1-0.0048374k_1 & 0.095163-0.0048374k_2 \\ -0.095163k_1 & 0.904837-0.095163k_2 \end{bmatrix}}_{\text{closed loop system matrix} \equiv A_{CL}} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix}$$

* closed loop system equation is $x(k+1) = A_{CL} \cdot x(k)$

* The characteristic equation of a square matrix: $\det(A - \lambda I) = 0$.

$\therefore \alpha_c(z) = |zI - A_{CL}| = 0$

$\boxed{\text{Hold}}$

* Recall : TF of $\begin{cases} x(k+1) = Ax(k) + Bu(k) \\ y(k) = Cx(k) + Du(k) \end{cases}$

• Take z-transforms similar to Laplace for CT systems. So $\dot{x} = Ax + Bu$, $y = Cx + Du$

A_c, B_c : Continuous

• Taking Laplace of both as: $sX(s) = A_cX(s) + B_cU(s) \rightarrow \therefore X(s)[sI - A_c] = B_cU(s)$

$\therefore \frac{X(s)}{U(s)} = (sI - A_c)^{-1}B_c$, $\frac{Y(s)}{U(s)} = C(sI-A)^{-1}B + D$

* Similarly, $\dfrac{X(z)}{U(z)} = (zI-A)^{-1}B$ and $\dfrac{Y(z)}{U(z)} = C(zI-A)^{-1}B + D$

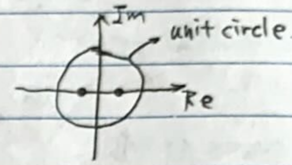
Back to

- $z^2 + [(0.0048374k_1) + 0.095163k_2 - 1.904837]z - 0.095163k_2 + 0.904837 = 0.$

  ↳ characteristic equation

- We know: λ_1 and λ_2 are the roots of $\alpha_c(z) = 0$ $\qquad (z-\lambda_1)(z-\lambda_2) = 0$
- Also, $\alpha_c(z) = (z-\lambda_1)(z-\lambda_2) = z^2 - (\lambda_1+\lambda_2)z + \lambda_1\lambda_2 = 0$
- Equate coefficients:

$$\begin{cases} 0.0048374k_1 + 0.095163k_2 = -(\lambda_1+\lambda_2) + 1.904837 \\ 0.0046788k_1 - 0.095163k_2 = \lambda_1\lambda_2 - 0.904837 \end{cases}$$



→ 2 algebraic equations with 2 unknowns: k_1 and k_2.

→ You place λ_1 and λ_2 (your choice), any place on the PZ map

- great but tedious to compute k matrix. Is there a simpler way?

→ Yes! Ackermann formula

- So, choose root locations to satisfy design criteria.

* In general, say we have an n^{th} order plant.
- $x(k+1) = Ax(k) + Bu(k)$, $u(k) = -Kx(k)$, where $K = \underbrace{[k_1\ k_2\ k_3 \cdots k_n]}_{\text{constants}}$
- substitute: $\underbrace{x(k+1)}_{n\times 1} = \underbrace{(A-BK)}_{\substack{n\times n \\ A_{CL}}}\underbrace{x(k)}_{n\times 1}$

- $A_{CL} = A - BK$
- pick desired poles: $z = \lambda_1, \lambda_2, \lambda_3, \ldots, \lambda_n$
- Closed loop characteristic equation is $\alpha_c(z) = |zI - A + BK| = (z-\lambda_1)(z-\lambda_2)\cdots(z-\lambda_n)$
- $k_1, k_2, \ldots, k_n$: unknowns
- Solve by equating coefs
- So, if, say

$$x(k+1) = \underbrace{\begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & & & & \\ -a_0 & -a_1 & -a_2 & \cdots & -a_{n-1} \end{bmatrix}}_{A} x(k) + \underbrace{\begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}}_{B} u(k)$$

- The plant characteristic equation:

  (open loop): $\alpha(z) = |zI - A| = z^n + a_{n-1}z^{n-1} + \cdots + a_1 z + a_0 = 0$

- $BK = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}_{n\times 1} [k_1\ k_2 \cdots k_n]_{1\times n} = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & & & \\ k_1 & k_2 & \cdots & k_n \end{bmatrix}$, $\quad A_{CL} = \underbrace{A}_{n\times n} - \underbrace{BK}_{n\times n}$

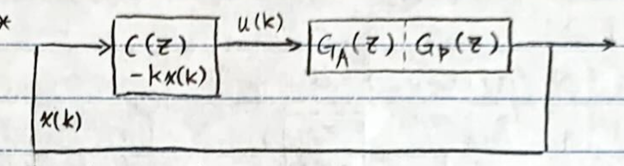
→ next page.

EEME E6601 Intro to Control Theory 11/30 ②

$$\therefore A_{CL} = \begin{bmatrix} 0 & 1 & \cdots & 0 \\ 0 & 0 & & 0 \\ \vdots & & & \\ (-a_0+k_1) & (-a_1+k_2) & \cdots & -(a_{n-1}+k_n) \end{bmatrix}$$

- characteristic equation: $|zI - A + Bk| = z^n + (a_{n-1}+k_n)z^{n-1} + \cdots + (a_1+k_2)z + (a_0+k_1) = 0$
  closed loop

*



* The desired characteristic equation is $\alpha_c(z) = z^n + \alpha_{n-1}z^{n-1} + \alpha_{n-2}z^{n-2} + \cdots + \alpha_1 z + \alpha_0 = 0.$

- Equate coef's $\Rightarrow$ yields $k_{i+1} = \alpha_i - a_i$, $i = 0, 1, 2, \ldots, n-1$

$\boxed{\text{Hold}}$ * Cayley-Hamilton Theorem

- a square matrix satisfies its own characteristic equation

$\boxed{\text{Back to}}$ $\therefore \alpha_c(\overset{(**)}{A}) = A^n + \alpha_{n-1}A^{n-1} + \cdots + \alpha_1 A + \alpha_0 I$

- Recall: $x(k+1) = \overset{(*)}{A}x(k) + Bu(k)$

(**) $\alpha_c(A)$ $\downarrow$ state matrix

- Ackermann's formula to solve for matrix is:

(*): same state matrix.

$$\boxed{k = [0\ 0 \cdots 0\ 1][B\ \ AB\ \ A^2B\ \cdots\ A^{n-2}B\ \ A^{n-1}B]^{-1}\alpha_c(A)}$$

* Example: $x(k+1) = \begin{bmatrix} 1 & 0.095163 \\ 0 & 0.904837 \end{bmatrix} x(k) + \begin{bmatrix} 0.0048374 \\ 0.095163 \end{bmatrix} u(k)$

- The desired characteristic equation: $\alpha_c(z) = z^2 - 1.778644z + 0.818731$
- Now, apply Cayley-Hamilton:

$$\alpha_c(A) = \begin{bmatrix} 1 & 0.095163 \\ 0 & 0.904837 \end{bmatrix}^2 - 1.778644 \begin{bmatrix} 1 & 0.095163 \\ 0 & 0.904837 \end{bmatrix} + 0.818731 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\therefore \alpha_c(A) = \begin{bmatrix} 0.040087 & 0.012009 \\ 0 & 0.026078 \end{bmatrix}, \quad AB?$$

- $AB = [A]_{2\times2}[B]_{2\times1} = \begin{bmatrix} 0.01369 \\ 0.0861 \end{bmatrix}_{2\times1}$

- So, $[B\ AB]^{-1} = \begin{bmatrix} -95.082 & 15.342 \\ 105.082 & -5.342 \end{bmatrix}$

- From Ackermann Formula:

$$k = [0\ 1]\begin{bmatrix} -95.082 & 15.342 \\ 105.082 & -5.342 \end{bmatrix}\begin{bmatrix} 0.040087 & 0.012009 \\ 0 & 0.026078 \end{bmatrix} \rightarrow \therefore k = [4.212\ \ 1.112]$$

Next $\rightarrow$ page

· Introduce : MATLAB

>> acker (A, B, P) characteristic equation or directly assign desired poles to p if

where >> p = roots(alphac) % desired poles p is given in the problem.

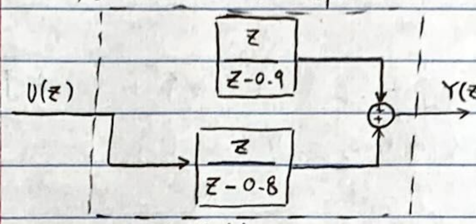
· works well for low order system (numerically)

· For high order system use "place"

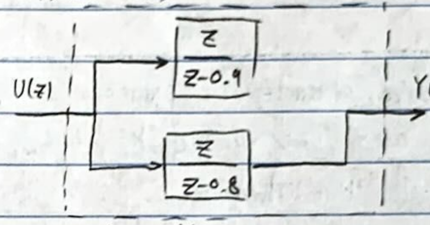
· Judging by the Ackermann formula, k's hinge on existence of $\underbrace{[B \ AB \ \cdots]}_{\text{matrix}}$

* Controllability

· Say, characteristic equation is $(z-0.9)(z-0.8)=0$.



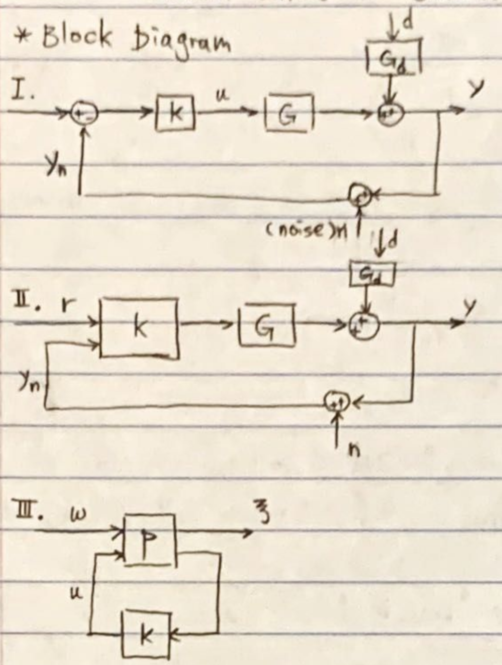
uncontrollable.



unobservable.

EEME E6601 Intro to Control Theory 12/7 ①

* Block Diagram



- Three different control configurations.

  k: controller (in all 3 systems)
- I, II : Conventional control configuration
- I : single d.o.f control configuration
- II: two d.o.f control configuration.

* Controllability

- Recall: Ackerimann's formula

$$k = [0\ 0\ 0 \cdots 1]\underbrace{[B\ AB\ A^2B \cdots A^{N-1}B]}_{e^{-1}}{}^{-1}\,\alpha_c(A)$$

- A system (A, B, C, D) is completely controllable if all of its states are controllable. A state $x(k)$ is controllable if it (state) can be reached from an initial state $x(0)$ in finite time.

- $x(k+1) = Ax(k) + Bu(k)$

i) $k=0$: $x(1) = Ax(0) + Bu(0)$] substitute.

ii) $k=1$: $x(2) = Ax(1) + Bu(1)$

$$= A^2x(0) + ABu(0) + Bu(1)$$

$\vdots$

N is finite # → $k = N-1$: $x(N) = A^N x(0) + A^{N-1}Bu(0) + \cdots + ABu(N-2) + Bu(N-1)$

$\therefore x(N) = A^N x(0) + [B\ AB\ A^2B \cdots A^{N-1}B][u(N-1)\ u(N-2) \cdots u(0)]^T$

- With $x(0)$ and $x(N)$ are known, we can write:

$$[B\ AB\ A^2B \cdots A^{N-1}B][u(N-1)\ u(N-2)\ u(N-3) \cdots u(0)]^T = x(N) - A^N x(0)$$

$\therefore$ N linear simultaneous equations for a solution to exist, the rank of the coefficient (parameter) matrix needs to = n also

C matrix (funky c) $\therefore$ rank(c) $\overset{\text{must}}{=}$ n so there is a solution

↳ to check $\therefore$ the system hence has to be controllable

controllability of the system $\therefore$ inverse of c has to exist! (equivalent statements)

- In MATLAB : >> $C_o = \text{ctrb}(A, B)$ returns C matrix
- $k = [0\ 0 \cdots 1]\, C^{-1} \alpha_c(A)$

control Law

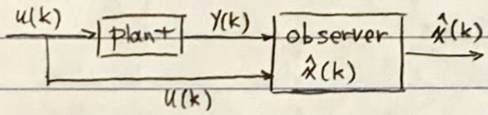
- $u = -kx$ (full state feedback)

↳ This is other way of designing controller from traditional way (using poles and zeroes)

* Discrete time LTI system

$$\begin{cases} x(k+1) = Ax(k) + Bu(k) & \rightarrow \text{state equation} \\ y(k) = Cx(k) & \rightarrow \text{output equation.} \end{cases}$$

- $y(k)$ are the plant signals.
- Also, fullstate feedback control law requires all states to be measured. This is impractical sometimes. It would be nice to measure what we can measure and estimate the rest!
- we know : $y(k)$, $u(k)$



⇝ $\hat{x}(k)$: prediction.

- observer dynamics : $\hat{x}(k+1) = F\hat{x}(k) + Gy(k) + Hu(k)$

(skipping details, see uploaded notes)

- observer dynamics become : $\hat{x}(k+1) = \underbrace{(A - GC)}_{}\hat{x}(k) + Gy(k) + Hu(k)$

↳ looks like $A - Bk$ in pole placement

→ select $H \equiv B$

→

$$G = \alpha_c(A) = \underbrace{\begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{bmatrix}}_{} \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}$$

⟹ looks like Ackermann's formula.

↳ observability matrix (funcky $\mathcal{O}$)

* optimal control theory.

- LQR (linear quadratic regulator) → quadratic : cost function.

↳ $u = -kx$

↳ $x(k+1) = Ax(k) + Bu(k)$

↳ pick $u(k)$ so that a cost function J,

$J = \frac{1}{2}\sum_{k=0}^{N} \left[x^T(k) Q_1 x(k) + u^T(k) Q_2 u(k) \right]$ is minimized.

↳ Q_1 and Q_2 : nonnegative definite.

Next → page

$\frac{\partial J'}{\partial u(k)}$

EEME E6601 Intro to Control Theory 12/17. ②

- Minimize J subject to the constraint.

$\rightarrow x(k+1) + Ax(k) + Bu(k) = 0$: constraints

- $J' = J + \lambda \cdot$ constraints, λ : Lagrangian multiplier.
- So, solve problem as minimizing on unconstrained problem with J' as its objective (cost) function
- $J' = \sum_{k=0}^{N} \left\{ \left[\frac{1}{2} x_k^T Q_1 x_k + \frac{1}{2} u_k^T Q_2 u_k \right] + \lambda_{k+1} (-x_{k+1} + Ax_k + Bu_k) \right\}$, ~~$x_k$: k sampling time of x.~~

$\frac{\partial J'}{\partial u_k} = 0 \rightarrow \therefore u_k^T Q_2 + \lambda_{k+1}^T B = 0 \rightarrow$ Controls equations

$\frac{\partial J'}{\partial \lambda_{k+1}} = 0 \rightarrow \therefore -x_{k+1} + Ax(k) + Bu(k) = 0 \rightarrow$ state equations

$\frac{\partial J'}{\partial x_k} = 0 \rightarrow \therefore x_k^T Q_1 - \lambda_k^T + \lambda_{k+1} A^T = 0 \rightarrow$ Adjoint $\rightarrow$ Costate equations

① : state equation
② : From controls equation
③ : From adjoint equation.

- Adjoint equation can be written as $\lambda_k = A^T \lambda_{k+1} + Q_1 x(k)$

$\rightarrow$ recall : ① $\boxed{x(k+1) = Ax(k) + Bu(k)}$ also : ② $\boxed{u(k) = -Q_2^{-1} B^T \lambda(k+1)}$

- In fwd difference equation form : ③ $\boxed{\lambda(k+1) = A^T \lambda(k) - A^T Q_1 x(k)}$

$\rightarrow$ 3 equation with 3 unknowns : $u(k)$, $x(k)$, $\lambda(k)$

- A bit difficult to solve so, say, $\lambda(N) = Q x(N)$

$\lambda(k) = S(k) x(k)$

- By this transformation of λ, we have gone from two-point boundary value problem to one point boundary value.