6.302 Fall 2021
Lecture 9

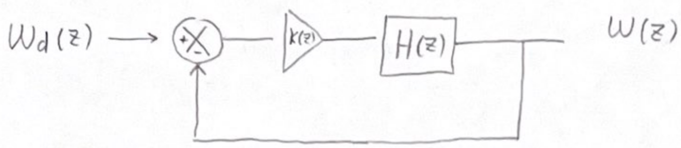
Today: 1. Z-transform example
- useful matlab commands
- comparison with analytical methods

2. Intuition of PID controller

1 Z-transform example:

from lab 1: $w[n] = w[n-1] + \Delta T\,(\beta\, w[n-1] + \gamma\, c[n-1])$

$c[n] = K_p\,(w_d[n] - w[n])$



from last class

$$H(z) = \frac{\gamma \Delta T z^{-1}}{1 - z^{-1} - \Delta T z^{-1}\beta} \quad \Leftarrow \text{open loop}$$

$$K(z) = K_p$$

step input: $x[n] = 1$ for $n \geq 0$
0 for $n < 0$

$$X(z) = \frac{z}{z-1} \quad \text{or} \quad \frac{1}{1-z^{-1}}$$

$$W(z) = H_{close}(z)\, W_d(z)$$

$$= \frac{\Delta T \gamma K_p z^{-1}}{1 - z^{-1}(1 + \Delta T \beta - \Delta T \gamma K_p)}\, W_d(z)$$

Solving this problem with matlab:

```
% define variables
  dt = 1e-2;
  z = tf([1 0], 1, -1);
```

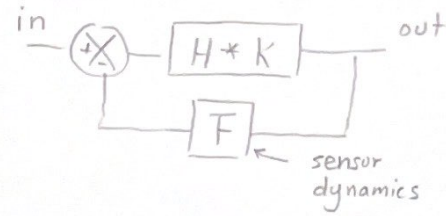
$$\underbrace{[1\ \ 0]}_{num},\ \underbrace{1}_{den},\ \underbrace{-1}_{discrete} \quad \Rightarrow \quad \frac{1 \cdot z + 0 \cdot 1}{1 \cdot 1}$$

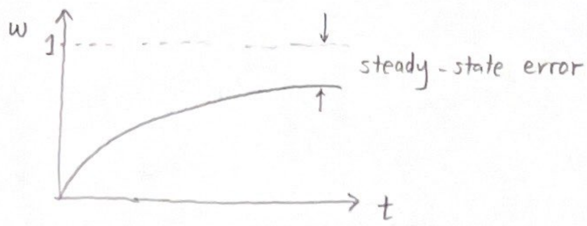
```
  zi = z^(-1);
  beta = -10;
  gamma = 100;
% define open loop transfer function and controller
  Kp = 0.5;    K = Kp;
  Hopen = (dt * gamma) / (z - 1 - beta * dt);
```

% solve the problem:

Hclose = feedback (Hopen * K , 1);

step (Hclose)





Solving the problem by going through detailed steps:

```
% Hstep = z / (z-1) ;
  Hsolution = Hstep * Hclose

  num = cell2mat (Hsolution . numerator);
  den = cell2mat (Hsolution . denominator);

  [r, p] = residuez (num, den);
                % partial fraction

  ycalc = r(1)*p(1).^t + ...
          r(2)*p(2).^t ;
```

$$W_d(z) = \frac{z}{z-1} = \frac{1}{1-z^{-1}}$$

$$W(z) = \frac{\Delta T g k_p z^{-1}}{1-z^{-1}(1+\Delta T\beta - \Delta T g k_p)} W_d(z)$$

$$= \frac{0.5\, z}{z^2 - 1.4z + 0.4}$$

num = [0 0.5 0];

den = [1 -1.4 0.4];

r = [0.8333 ; -0.8333]

p = [1 ; 0.4]

$$\frac{0.5\,z}{z^2 - 1.4z + 0.4} = \frac{0.8333}{1-z^{-1}} + \frac{-0.8333}{1-0.4z^{-1}}$$

$$w[n] = 0.8333(1)^n + -0.8333(0.4)^n$$

note: ycalc is identical to step (Hclose)

Recap of partial fraction:

e.g. $W(z) = \dfrac{z}{(z-a)(z-b)} = \dfrac{zA}{z-a} + \dfrac{zB}{z-b}$

<u>want</u>: A and B

We introduce a math trick: 1 multiply, 2 evaluate

1 multiply: $\left[\dfrac{z}{(z-a)(z-b)} = \dfrac{zA}{z-a} + \dfrac{zB}{z-b} \right] (z-a)$

$$\Rightarrow \frac{z}{z-b} = zA + \frac{zB(z-a)}{z-b}$$

2 evaluate: (at $z=a$) $\left[\dfrac{z}{z-b} = zA + \dfrac{zB(z-a)}{z-b} \right] \Big|_{z=a}$

$$\Rightarrow \frac{\cancel{a}^{1}}{a-b} = \cancel{a}^{1}A + 0$$

Summary of partial fraction :

given $\quad W(z) = \dfrac{z\, n(z)}{(z-a_1)(z-a_2)\cdots(z-a_n)} = \dfrac{z A_1}{(z-a_1)} + \cdots + \dfrac{z A_n}{(z-a_n)}$

for each A_i, perform 2 steps

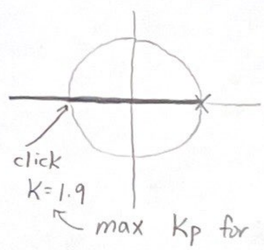
1. multiply by $(z-a_i)$
2. evaluate at $z = a_i$

$$A_i = \Big[\underbrace{W(z)\ (z-a_i)}_{\text{① multiply}} \Big] \Big|_{\underbrace{z=a_i}_{\text{② evaluate}}}$$

Stability condition :

% rlocus (Hopen * k)

set this to 1



Steady state error :

FVT: $w[\infty] = \lim_{z\to 1} (z-1)\, W(z) \quad \Rightarrow$ equivalent to $\lim_{z\to 1} H_{close} = 1$

% evalfr (Hclose , 1) = 0.8333 $\Rightarrow$ there is a steady-state error!

% substitute $z=1$ into Hclose

2 Intuition about PID controller

$\Rightarrow$ how do we remove the steady-state error ?

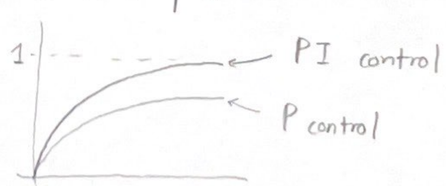
$\Rightarrow$ move to a PID controller ! set kd = 0; ki = 5;

here $\quad K(z) = k_p + \dfrac{k_d}{\Delta T}(1-z_i) + k_i \Delta T \dfrac{1}{1-z_i}$

% K-pid = Kp + kd / dt * (1-zi) + ki * dt / (1-zi)

Hclose_pid = feedback (Hopen * K_pid , 1)

step (Hclose-pid)



Based on matlab, the K_i term removes the steady-state error.

Why?

$$\lim_{z\to 1} H_{close} = \frac{\overbrace{\Delta T \gamma \; K(z)}}{z - (1 + \Delta T \beta - \underbrace{\Delta T \gamma \, K(z)})}$$

how to design $K(z)$

let this term goes to ∞

↳ this is equivalent to adding a "pole": $\frac{1}{1 - z_i}$

$$\text{let } K(z) = K_p + K_i \, \Delta T \, \frac{1}{1 - z_i}$$

intuition:

K_p: small: stable

large: smaller s.s error, but less stable

K_d: help to stablize, but can't be too big

K_i: help to remove s.s error, but can't be too big.