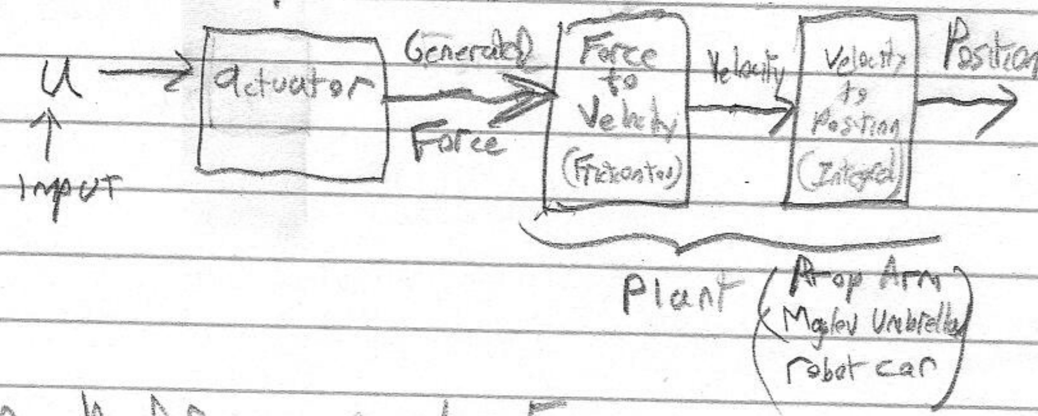
6.310 State Space & Transfer Functions 11/12/25

Typical Control Problem



System of Differential Eqns

x_1: e.g. Force → $\dot{x}_1 = \beta_1 x_1 + \gamma_1 u$ ← input (scalar)

x_2: e.g. Velocity → $\dot{x}_2 = \beta_2 x_2 + \gamma_2 x_1$

x_3: e.g. Position → $\dot{x}_3 = \beta_3 x_3 + \gamma_3 x_2$ $\qquad y = x_3$ ← output (scalar)

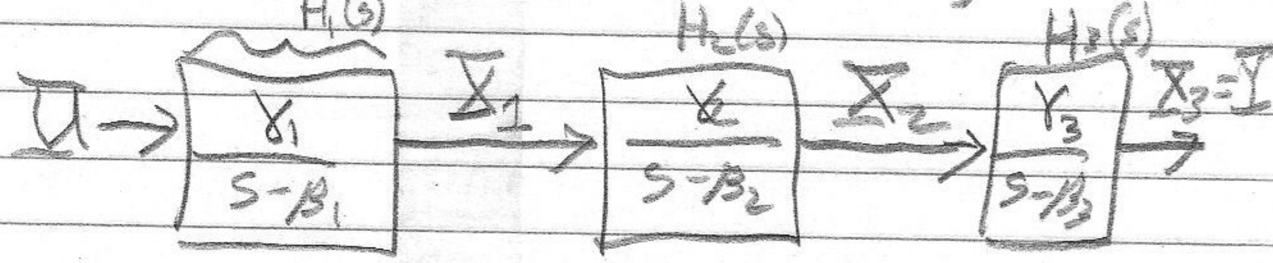
State-Space (A, B, C) representation

$$\dot{x} = Ax + Bu$$

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} \beta_1 & & \\ \gamma_2 & \beta_2 & \\ & \gamma_3 & \beta_3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} \gamma_1 \\ 0 \\ 0 \end{bmatrix} u$$ ← scalar

$$y = C\vec{x} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = x_3$$

Transfer Functions ($\frac{d}{dt} \Leftrightarrow s$)



$$\underline{Y} = \underline{X}_3 = H(s)\,\underline{U} = H_1(s)H_2(s)H_3(s)\,\underline{U}$$

$$= \frac{\gamma_1\gamma_2\gamma_3}{(s-\beta_1)(s-\beta_2)(s-\beta_3)}\,\underline{U}$$

$H_1(s)$ $\quad \dot{x}_1 = \beta_1 x_1 + \gamma_1 u \Rightarrow s\underline{X}_1 = \beta_1\underline{X}_1 + \gamma_1\underline{U}$

$$(s-\beta_1)\underline{X}_1 = \gamma_1\underline{U}$$

$$\underline{X}_1 = \underbrace{\frac{\gamma_1}{s-\beta_1}}_{H_1(s)}\underline{U}$$

$H_2(s)$, $H_3(s)$ similar derivation

How are $H(s)$ and $C(sI-A)^{-1}B$ connected

Applying Transform to A, B, C

$$\underbrace{\dot{\vec{X}}(t)}_{\frac{d}{dt}\vec{X}} = A\vec{X}(t) + BU(t) \underset{\frac{d}{dt}\to s}{\Rightarrow} s\underline{X} = A\underline{X} + B\underline{U}$$

$$\underline{Y} = C\underline{X}$$

$$s\underline{X} - A\underline{X} = (sI - A)\underline{X} = B\underline{U} \Rightarrow \underline{X} = (sI-A)^{-1}B\underline{U}$$

$$\underline{Y} = C\underline{X} = \underbrace{C(sI-A)^{-1}B}_{H(s)}\,\underline{U}$$

③

Suppose A is diagonal

$$A = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix}$$

$$(sI - A)^{-1} = \begin{bmatrix} \frac{1}{s-\lambda_1} & & \\ & \ddots & \\ & & \frac{1}{s-\lambda_n} \end{bmatrix}$$

Assuming SISO $C \in \mathbb{R}^{1 \times n}$ $B \in \mathbb{R}^{n \times 1}$

$$H(s) = [C_1, \ldots, C_n] \underbrace{\begin{bmatrix} \frac{1}{s-\lambda_1} & & \\ & \ddots & \\ & & \frac{1}{s-\lambda_n} \end{bmatrix}}_{(sI-A)^{-1}} \begin{bmatrix} B_1 \\ \vdots \\ B_n \end{bmatrix}$$

$$= \frac{C_1 B_1}{s-\lambda_1} + \frac{C_2 B_2}{s-\lambda_2} + \cdots \frac{C_n B_n}{s-\lambda_n}$$

Cross Multiplying

$$H(s) = C(sI-A)^{-1}B = \frac{\sum_{i=1}^{n} C_i B_i \frac{\prod_{j=1}^{n}(s-\lambda_j)}{(s-\lambda_i)}}{(s-\lambda_1)(s-\lambda_2) \cdots (s-\lambda_n)}$$

Order $n-1$ polynomial in s

If A is diag

1) diag of A are poles of $H(s)$

2) diag of A are eigenvalues

For Fairly General A

$$\frac{d}{dt}x = Ax + Bu$$

Assume A is diagonalizable

$$\exists \underbrace{V}_{\text{invertible}} \text{ s.t. } AV = V\begin{bmatrix}\lambda_1 & & \\ & \ddots & \\ & & \lambda_n\end{bmatrix}$$

Why Eigenvalues are poles

If $AV = V \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix}$

$A[\vec{v}_1 \ \vec{v}_2 \ \vec{v}_n] = [\lambda_1\vec{v}_1 \ \lambda_2\vec{v}_2 \ \lambda_n\vec{v}_n]$

λ_i is the eigenvalue, $\vec{v}_i$ is eigenvector

$V = \begin{bmatrix} \vec{v}_1 & \vec{v}_i & \vec{v}_n \end{bmatrix}$ $\quad \vec{v}_i \in \mathbb{R}^n$ is the eigenvector

and

Use eigenvectors

$x = \underbrace{V \ W}_{\text{weighted combo of evectors}} = \underbrace{\begin{bmatrix} \vec{v}_1 & \vec{v}_n \end{bmatrix} \begin{bmatrix} w_1 \\ \vdots \\ w_n \end{bmatrix}}$

original state

$x = \underbrace{w_1\vec{v}_1 + w_2\vec{v}_2 + \dots w_n\vec{v}_n}$

modal decomposition
or
spectral decomposition

Substituting and multiplying

$V^{-1}\left(\frac{d}{dt}\overset{x}{\widehat{Vw}} = A\overset{x}{\widehat{Vw}} + Bu\right)$

$\Rightarrow \frac{d}{dt} w = V^{-1}AVw + V^{-1}Bu$

Note

$x = V w$ Spectral Decomposition

$w = V^{-1} x$ spectral projection

$$w = \begin{bmatrix} \text{---} (V^{-1})_1 \text{---} \\ \vdots \\ \text{---} (V^{-1})_n \text{---} \end{bmatrix} \begin{bmatrix} \\ x \\ \\ \end{bmatrix}$$

mapping initial condition in x to initial cond in w.

$$w_i = \begin{bmatrix} \leftarrow (V^{-1})_i \text{---} \end{bmatrix} \begin{bmatrix} \\ x \\ \\ \end{bmatrix} \Rightarrow \underbrace{w(t)\Big|_{t=0} = V^{-1} x(t)\Big|_{t=0}}$$

Back to System

$$\frac{d}{dt} w = \underbrace{V^{-1} A V}_{= \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix}} w + V^{-1} B u$$

$$y = C V w$$

$$\frac{d}{dt} w = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix} w + \tilde{B} u$$

$$y = \tilde{C} w$$

$$\Rightarrow H(s) = \sum \frac{\tilde{C}_i \tilde{B}_i}{s - \lambda_i}$$

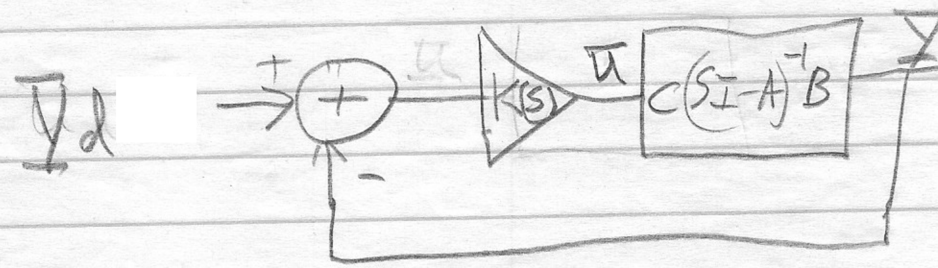
poles = evals

Control Before (SISO Case)

$$\bar{Y}(s) = \underbrace{C(sI-A)^{-1}B}_{H(s)}\,\bar{U}$$

Only Output (y) Feeds Back
$K(s)$ is a transfer function

$$\bar{U} = K(s)\,(\bar{Y}_d - \bar{Y})$$



Control Now (State Feedback)

Entire State Feeds Back

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

$$u(t) = -K\,x(t) + K_r\,y_d(t)$$

$K \in \mathbb{R}^{1\times n}$ for Single Input

$$y(t) = C\,x(t)$$

$C \in \mathbb{R}^{n\times 1}$ for single output

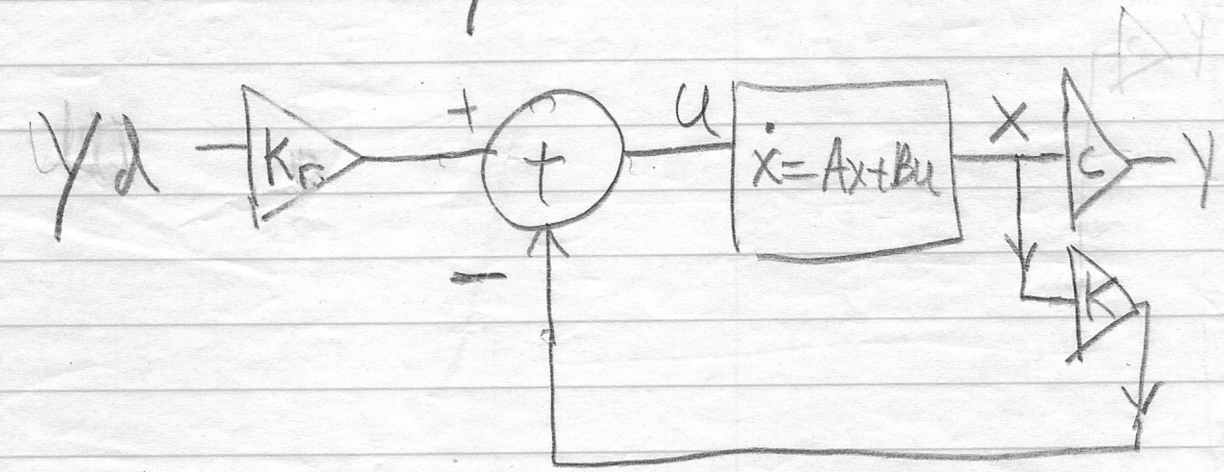
Closed Loop

$$\dot{x} = Ax + B(-Kx + K_r y_d)$$
$$y = Cx$$

Closed Loop

$$\dot{x} = Ax + Bu$$

$$u = K_r y_d - Kx$$



Closed loop system

$$\dot{x} = (A - BK)x + BK_r y_d$$

poles unaffected by K_r

$$y = Cx$$

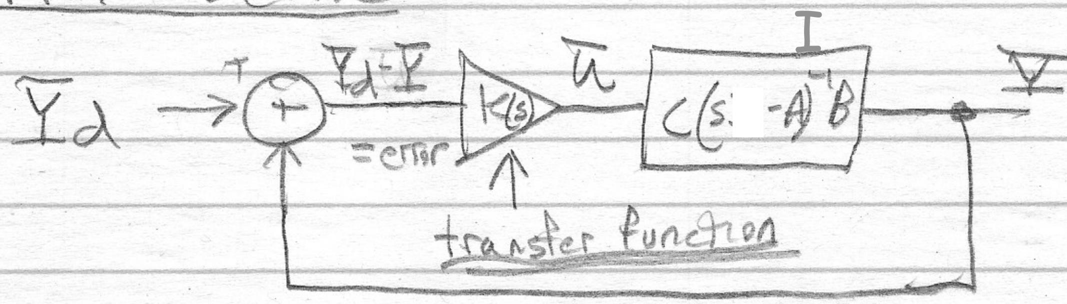
* **Pole Placement Goal** Pick K s.t.:

Make least stable pole have as negative a real part as possible:

$$\underbrace{\min_{\text{all } K\text{'s}}}_{\text{Pick best gains}} \Big(\underbrace{\max_{\text{all } \lambda(A-BK)}}_{\text{pick out least stable pole}} \text{Real} \Big)$$

What is K_r?

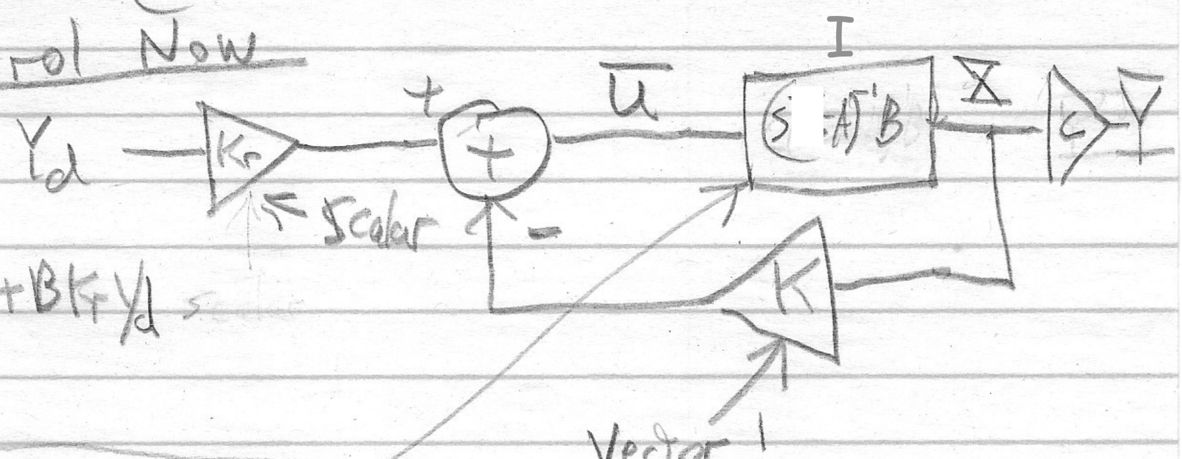
Control Before



$$\bar{Y} = G(s)\,\bar{Y}_d = \frac{K(s)\,\overbrace{\left(C(sI-A)^{-1}B\right)}^{H(s)}}{1 + K(s)\left(C(sI-A)^{-1}B\right)}\bar{Y}_d$$

<u>Want</u> $\mathrm{Re}(\text{poles}(G(s))) < 0$ (Stability)

Control Now



$$\dot{X} = (A - BK)X + BK_r Y_d$$
$$Y = CX$$

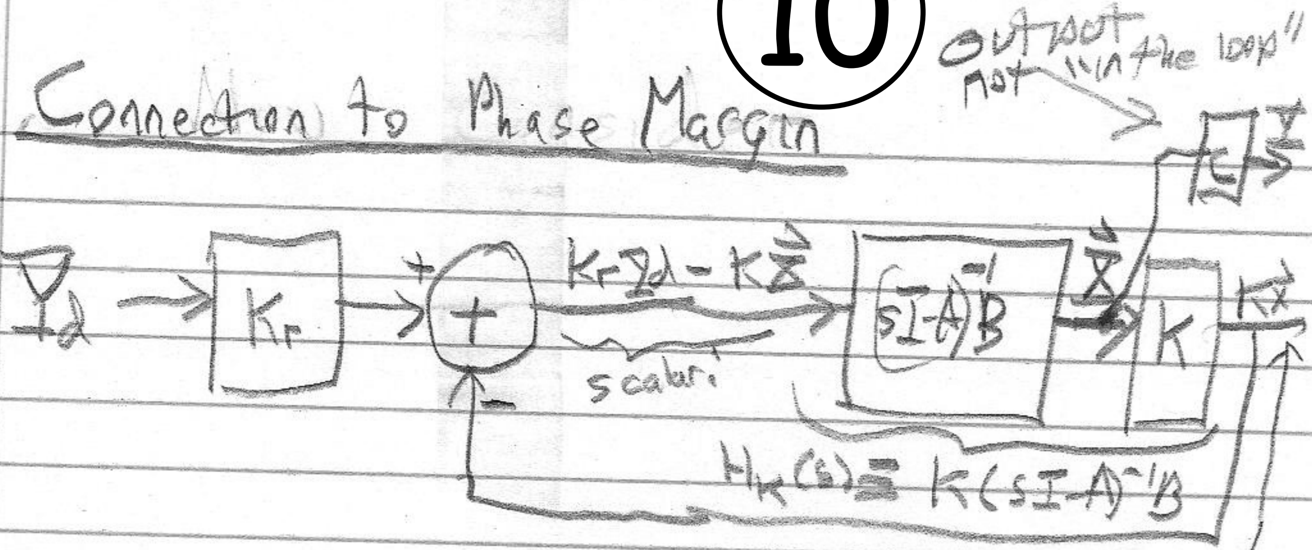
1 input, n output transfer function

$$\bar{Y} = C\left(sI - (A - BK)\right)^{-1} BK_r \bar{Y}_d$$

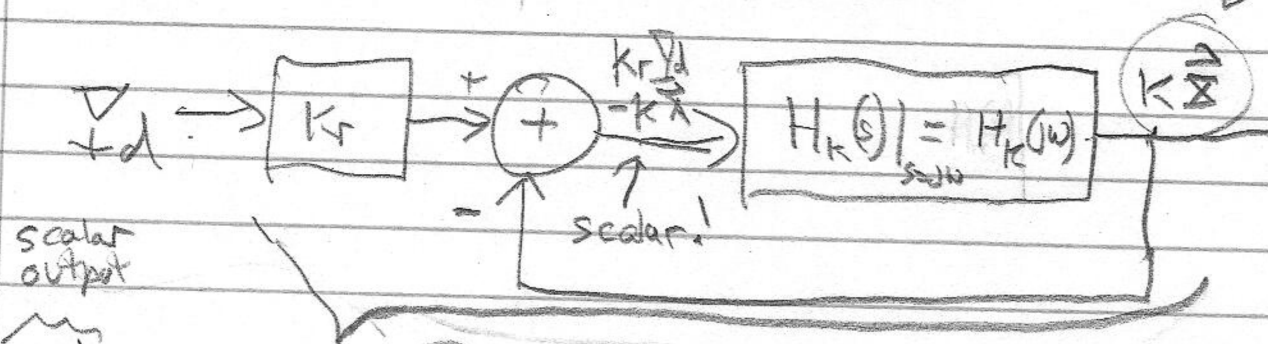
<u>In steady state ($s = 0$)</u> $\quad \bar{Y}_{ss} = C(A - BK)^{-1}BK_r\bar{Y}_d$

$$K_r = \frac{1}{C(A-BK)^{-1}B}$$

Connection to Phase Margin



For Frequency Response



$$\widehat{K\vec{X}} = G(j\omega)\, Y_d \leftarrow \text{scalar input}$$

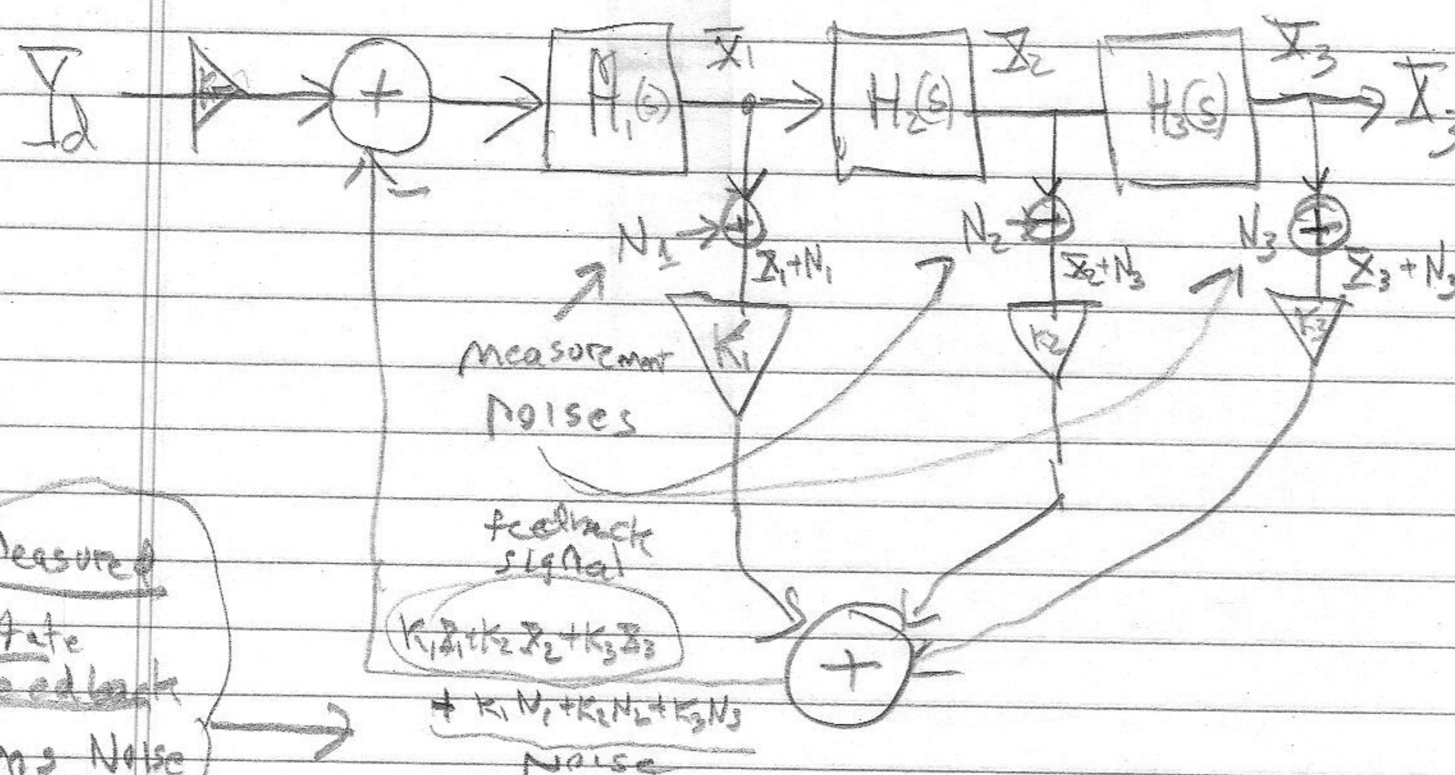
$$\underbrace{G(j\omega) = \frac{H_K(j\omega)}{1 + H_K(j\omega)}}_{\text{Black's Formula}}$$

So $H_K(j\omega)$ better not be near -1
(otherwise $G(j\omega) \to \infty$)

We can ask what is phase margin of $H_K(j\omega)$

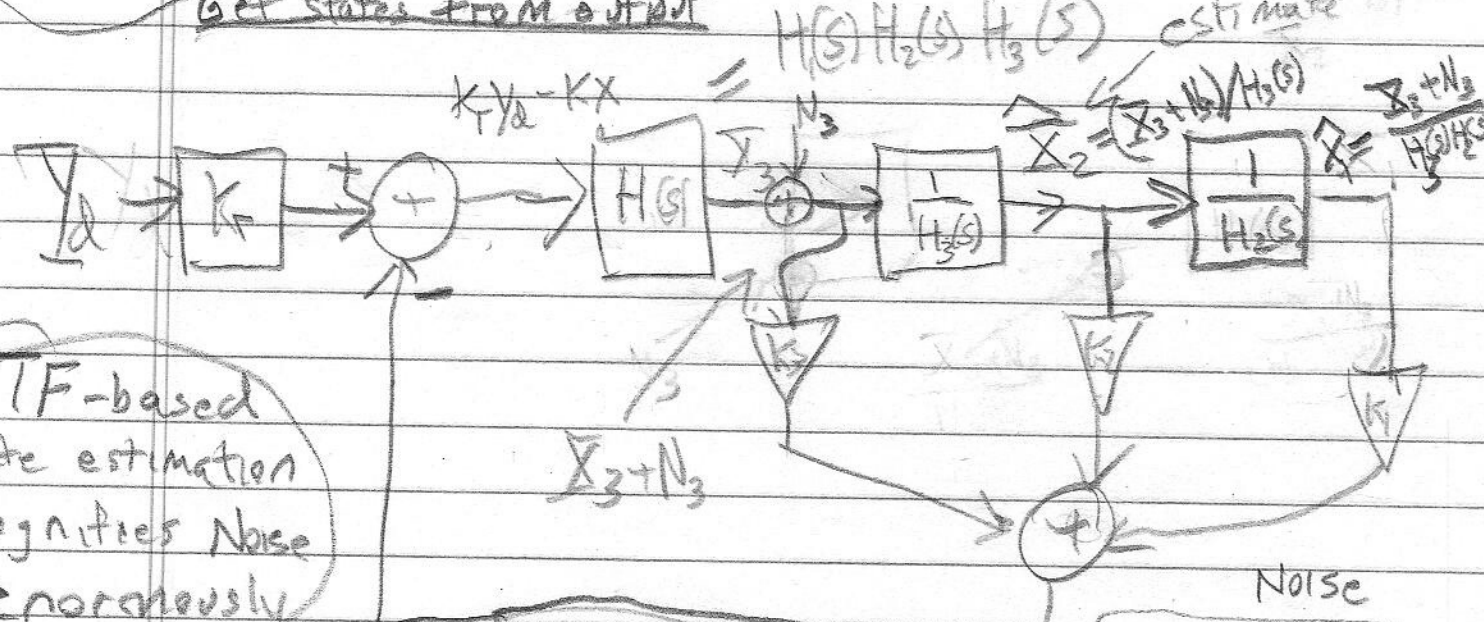
What else is true

T.F. Representation of S. S. controller



Measured State feedback Sums Noise

Get states from output



TF-based state estimation magnifies Noise enormously

$$K_1 X_3 + K_2 \underbrace{X_2}_{X_3/H_3(s)} + K_1 \underbrace{X_1}_{X_3/(H_3(s)H_2(s))} + K_1 N_3 + K_2 \frac{N_3}{H_3(s)} + K_3 \frac{N_3}{H_3(s)H_2(s)}$$

T.F.-based State Estimation Noise Magnification

Force to velocity, no friction, mass = 1

Example $H_3(s) = 1/s$ $\quad H_2(s) = \frac{1}{s}$

velocity position

$$K_1\left(X_3 + N_3\right) + K_2\left(\frac{X_3 + N_3}{H_3(s)}\right) + K_1 \frac{(X_3 + N_3)}{H_3(s)H_2(s)}$$

$$= \underbrace{K_1 X_3 + K_2 s X_3 + K_1 s^2 X_3}_{\text{PD \& D controller}}$$

$$+ \underbrace{K_1 N_3 + K_2 s N_3 + K_3 s^2 N_3}$$

$$\text{Noise} = \underbrace{\left(K_1 + K_2 s + K_3 s^2\right)}_{K(s)} N_3$$

PD & D controller Magnifies noise enormously