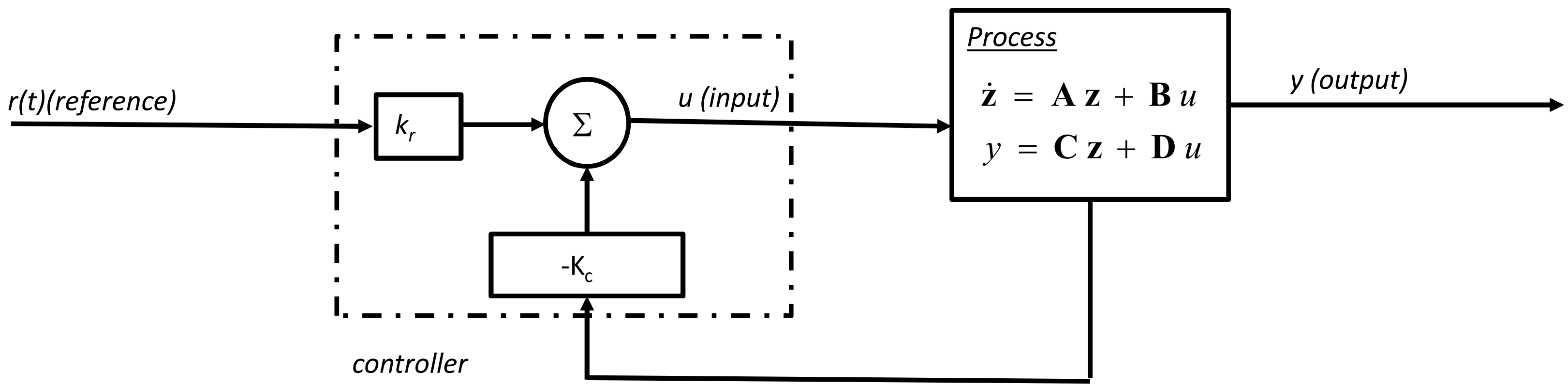
State Feedback Controller Design

Mitch Pryor

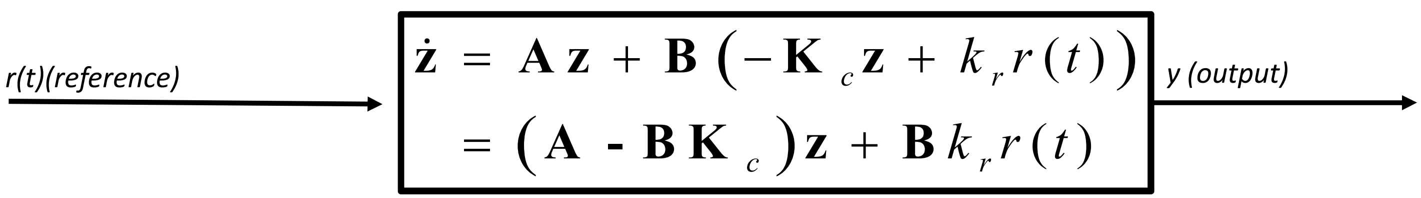
Lesson objective

- Find the gains for a state feedback controller that result in the desired system performance.
- Define a canonical form for systems useful for state feedback
- Design State feedback controllers for more complex systems.
- Complete a more complicated state feedback example
  - Quickly illustrate the impact of normalization
  - Show results for both manual and general methods
  - Quickly introduce some related MATLAB tools

State feedback control summary



The generalized state feedback control input is $u = -\mathbf{K}_c z + k_r y_r$



Goal: For systems that are reachable, find K_c such that the system is stable and behaves how we want it to.

$$p(s) = \left(\lambda\mathbf{I} - \left[\mathbf{A} - \mathbf{B}\mathbf{K}_c\right]\right) = 0 \qquad \textit{where...} \qquad \mathrm{Re}(\lambda_i) < 0$$

Note: stability is the <u>baseline</u> performance measure.

Controller Design

Objective: Find the gain matrix $\mathbf{K_c}$ such that:

$$p(s) = \left(\lambda \mathbf{I} - \left[\mathbf{A} - \mathbf{B}\mathbf{K}_{\mathbf{c}}\right]\right) = 0$$ *where...* $$\mathrm{Re}(\lambda_i) < 0$$ *Baseline performance is stability.*

A simple example: find a state feedback gain matrix that asymptotically stabilizes the system below.

$$\frac{d\mathbf{z}}{dt} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}\mathbf{z} + \begin{bmatrix} 0 \\ 1 \end{bmatrix}u \qquad y = \begin{bmatrix} 1 \\ 0 \end{bmatrix}\mathbf{z} + \left[0\right]u$$

Step 1: Determine if the system is controllable/reachable.

$$\mathbf{w}_r = \left[\mathbf{B} \quad \mathbf{A}\mathbf{B}\right] = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

$\mathbf{W}_r$ is full rank. Therefore system is reachable/controllable.

Step 2: Is the system stable without feedback?

$$\det(\lambda \mathbf{I} - \mathbf{A}) = \det\begin{bmatrix} \lambda & -1 \\ 0 & \lambda \end{bmatrix} = \lambda^2 = 0$$

The eigenvalues for this system are both 0, and the system is only neutrally stable.

Controller Design, Cont'd

$$\frac{d\mathbf{z}}{dt} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}\mathbf{z} + \begin{bmatrix} 0 \\ 1 \end{bmatrix}u \qquad y = \begin{bmatrix} 1 \\ 0 \end{bmatrix}\mathbf{z} + [0]u$$

Step 3: Insert a general state feedback controller for the system

$$u = -\mathbf{K}_c z + k_r y_r = \begin{bmatrix} -k_1 & -k_2 \end{bmatrix}\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + k_r y_r = -k_1 z_1 - k_2 z_2 + k_r y_r$$

Just a single input!

$$\frac{d\mathbf{z}}{dt} = \left[\begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} - \begin{bmatrix} 0 & 0 \\ k_1 & k_2 \end{bmatrix}\right]\mathbf{z} + \begin{bmatrix} 0 \\ k_r \end{bmatrix}u$$

$$= \begin{bmatrix} 0 & 1 \\ -k_1 & -k_2 \end{bmatrix}\mathbf{z} + \begin{bmatrix} 0 \\ k_r \end{bmatrix}u$$

Step 4: Find the characteristic equation for our controlled system.

$$\det(\lambda\mathbf{I} - (\mathbf{A} - \mathbf{B}\mathbf{K})) = \det\begin{bmatrix} \lambda & -1 \\ k_1 & \lambda + k_2 \end{bmatrix} = \lambda^2 + k_2\lambda + k_1 = 0$$

Step 5: Select gain values that result in desired behavior.

Controller Design, Cont'd

$$\frac{d\mathbf{z}}{dt} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}\mathbf{z} + \begin{bmatrix} 0 \\ 1 \end{bmatrix}u \qquad y = \begin{bmatrix} 1 \\ 0 \end{bmatrix}\mathbf{z} + [0]u$$

$$\det(\lambda \mathbf{I} - (\mathbf{A} - \mathbf{B}\mathbf{K})) = \det\begin{bmatrix} \lambda & -1 \\ k_1 & k_2 \end{bmatrix} = \lambda^2 + k_2\lambda + k_1 = 0$$

Step 5: Select gain values that result in desired behavior.

If our goal is simply stability, we pick eigenvalues where $\mathrm{Re}(\lambda_i) < 0$

For example, let the eigenvalues be -1 and -2.

$$\begin{aligned} CE &= (\lambda + 1)(\lambda + 2) \\ &= \lambda^2 + 3\lambda + 2 \\ &= \lambda^2 + k_2\lambda + k_1 \end{aligned}$$

Therefore the gains should be $\boldsymbol{k_2}$**=3 and** $\mathbf{k_1}$**=2**.

Or let the eigenvalues be -2±10j.

$$\begin{aligned} CE &= (\lambda + 2 + 10j)(\lambda + 2 - 10j) \\ &= \lambda^2 + 2\lambda - \lambda 10j + 2\lambda + 4 - 20j + \lambda 10j + 20j - 100j^2 \\ &= \lambda^2 + 4\lambda + 104 \\ &= \lambda^2 + k_2\lambda + k_1 \end{aligned}$$

Therefore the gains should be $\boldsymbol{k_2}$**=4 and** $\mathbf{k_1}$**=104**.

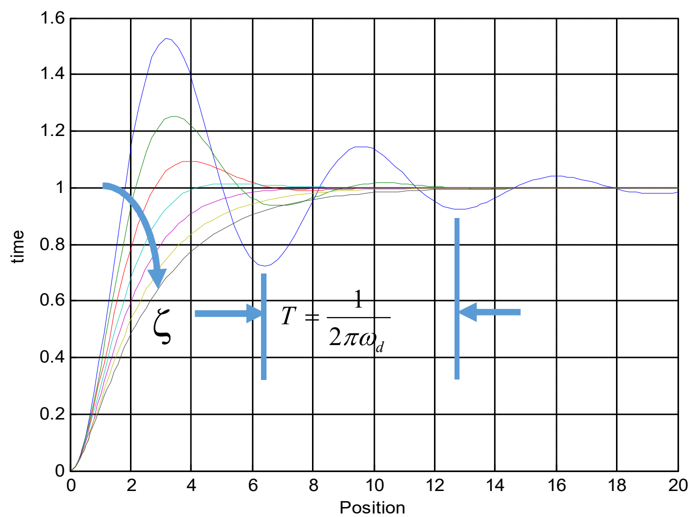
Controller Design, Cont'd

$$\frac{d\mathbf{z}}{dt} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}\mathbf{z} + \begin{bmatrix} 0 \\ 1 \end{bmatrix}u \qquad y = \begin{bmatrix} 1 \\ 0 \end{bmatrix}\mathbf{z} + [0]u$$

$$\det(\lambda\mathbf{I} - (\mathbf{A} - \mathbf{BK})) = \det\begin{bmatrix} \lambda & -1 \\ k_1 & k_2 \end{bmatrix} = \lambda^2 + k_2\lambda + k_1 = 0$$

Step 5 (cont'd): Or we can be a little more strategic about how we select our feedback gains.

Recall the 2nd order canonical system… $$\lambda^2 + 2\zeta\omega_n\lambda + \omega_n^2 = 0$$



$$\Rightarrow \quad k_1 = \omega_n^2 \qquad k_2 = 2\zeta\omega_n$$

So if we want a system to be critically damped and behave as if the natural frequency was 10, then…

$$\Rightarrow \quad k_1 = 100 \qquad k_2 = 20$$

Controller Design, Cont'd

$$\frac{d\mathbf{z}}{dt} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}\mathbf{z} + \begin{bmatrix} 0 \\ 1 \end{bmatrix}u \qquad y = \begin{bmatrix} 1 \\ 0 \end{bmatrix}\mathbf{z} + [0]u$$

Step 6: Find the feedforward gain

$$u = -\mathbf{K}_c z + k_r y_r = \begin{bmatrix} -k_1 & -k_2 \end{bmatrix}\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + k_r y_r = -k_1 z_1 - k_2 z_2 + k_r y_r$$

Note that k_r does NOT impact the stability of the system.

$$\dot{\mathbf{z}} = [\mathbf{A} - \mathbf{B}\mathbf{K}]\mathbf{z} + \mathbf{B}k_r y_r$$

Note that the <u>error</u> in the states and outputs has the same behavior as the states and outputs since $y_e = y - y_r$ and the difference is just a constant (offset).

$$\dot{\mathbf{z}}_e = \mathbf{A}\mathbf{z}_e + \mathbf{B}u \qquad y_e = \mathbf{C}\mathbf{z}_e + \mathbf{D}u$$

…and with state feedback…

$$\dot{\mathbf{z}}_e = [\mathbf{A} - \mathbf{B}\mathbf{K}]\mathbf{z}_e + \mathbf{B}k_r y_e \qquad y_e = \mathbf{C}\mathbf{z}_e + \mathbf{D}u$$

*We note that for a stable system, the rate of change in the states will go zero. If we assume (for the moment) that **D** is zero, then*

$$0 = [\mathbf{A} - \mathbf{B}\mathbf{K}]\mathbf{z}_e + \mathbf{B}k_r y_e$$

$$y_e = \mathbf{C}\mathbf{z}_e$$

Controller Design, Cont'd

$$\frac{d\mathbf{z}}{dt} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}\mathbf{z} + \begin{bmatrix} 0 \\ 1 \end{bmatrix}u \qquad y = \begin{bmatrix} 1 \\ 0 \end{bmatrix}\mathbf{z} + [0]u$$

Starting with the last result

$$0 = [\mathbf{A} - \mathbf{B}\mathbf{K}]\mathbf{z}_e + \mathbf{B}k_r y_e$$

$$y_e = \mathbf{C}\mathbf{z}_e$$

We solve for the state error, $\mathbf{z}_e$

$$[\mathbf{A} - \mathbf{B}\mathbf{K}]\mathbf{z}_e = -\mathbf{B}k_r y_e \qquad y_e = \mathbf{C}\mathbf{z}_e$$

$$\mathbf{z}_e = -[\mathbf{A} - \mathbf{B}\mathbf{K}]^{-1}\mathbf{B}k_r y_r$$

$$\mathbf{C}^{-1}y_e = -[\mathbf{A} - \mathbf{B}\mathbf{K}]^{-1}\mathbf{B}k_r y_e$$

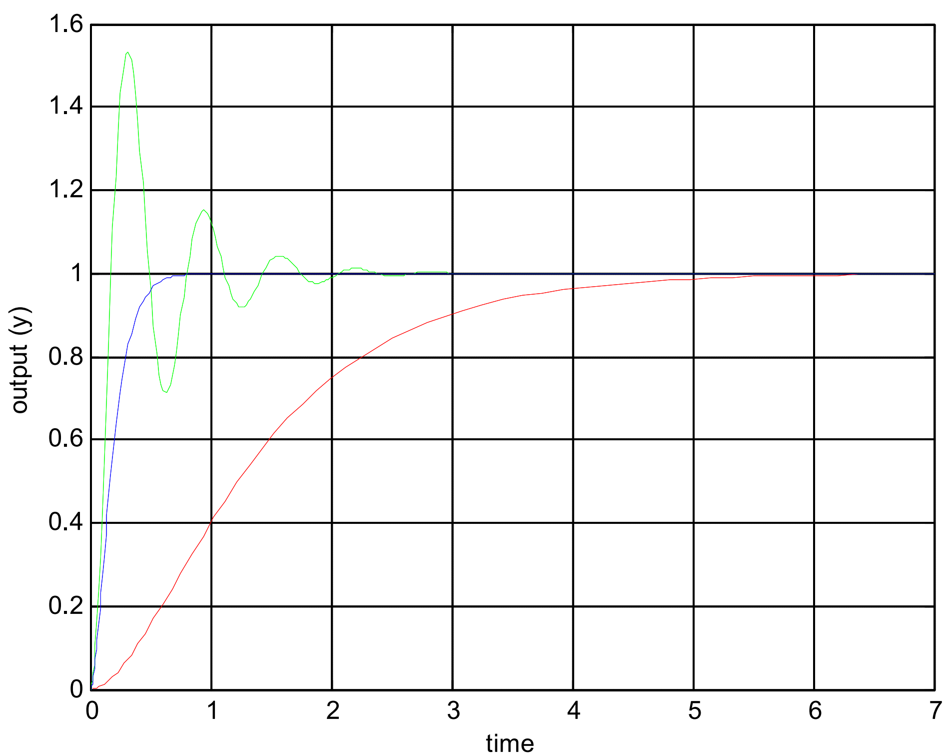
$$k_r = \frac{-1}{\mathbf{C}[\mathbf{A} - \mathbf{B}\mathbf{K}]^{-1}\mathbf{B}}$$

Which means we are picking k_r such that y_e=y_r. *This formula works if **D**=0. It is possible but a bit tougher to solve for* k_r *if **D** is nonzero.*

Controller Design results

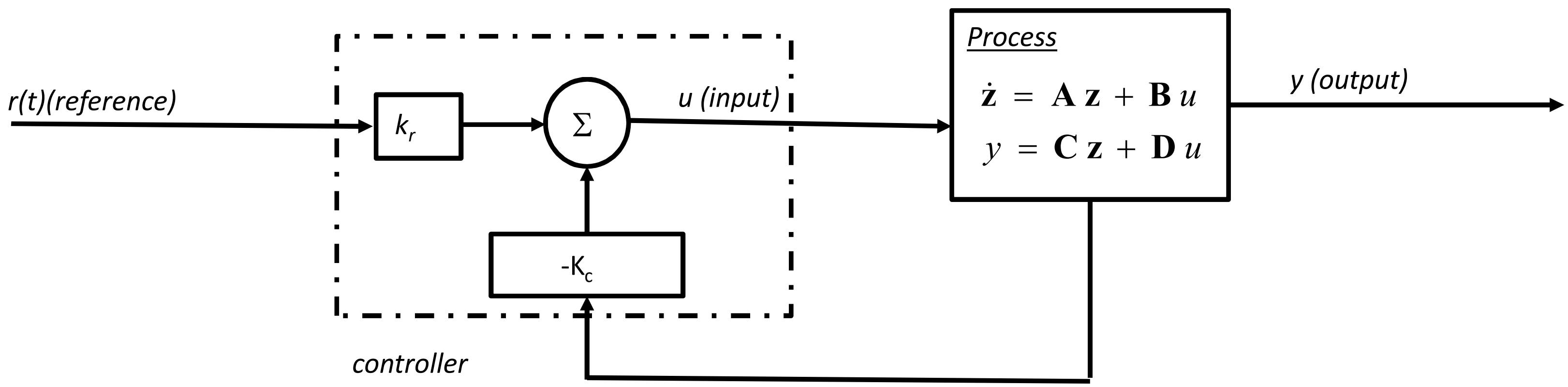
$$\frac{d\mathbf{z}}{dt} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}\mathbf{z} + \begin{bmatrix} 0 \\ 1 \end{bmatrix}u \qquad y = \begin{bmatrix} 1 \\ 0 \end{bmatrix}\mathbf{z} + [0]u$$

Let's look at the results in MATLAB




```
clear all;
global K;
tf = 7;

K = [ 2 3 ];
[t, z] = ode45( 'statefbexample', [ 0 tf ], [0 0] );
plot( t, z(:,1), 'r' );
hold on;

K = [ 104 4 ];
[t2, z2] = ode45( 'statefbexample', [ 0 tf ], [0 0] );
plot( t2, z2(:,1), 'g' );

K = [ 100 20 ];
[t3, z3] = ode45( 'statefbexample', [ 0 tf ], [0 0] );
plot( t3, z3(:,1), 'b' );
hold on;


function zprime = statefbexample( t, z )

A = [ 0 1; 0 0];
B = [ 0; 1 ];
C = [ 1 0 ];

global K;
yr = 1; %goal is to set output to 1
kr = -1/(C*inv(A-B*K)*B);

zprime = [ (A-B*K)*z + B*kr*yr; ];
```

State feedback control summary



- Illustrated procedure to design a state feedback controller
- Relatively straight forward for a 2nd order system
  - Need to extend to higher order systems
  - This might be easier if we can generalize the problem by putting any system in some sort of state feedback canonical form (coming up next!)
- Not much insight into the control effort, but easy to calculate since:

$$u = -\mathbf{K}_c z + k_r y_r$$

Reachable Canonical Form

$$\dot{\mathbf{x}} = \mathbf{A}\,\mathbf{x} + \mathbf{B}\,u$$
$$y = \mathbf{C}\,\mathbf{x} + [\,\mathbf{0}\,]u$$

Given the original system…

$$\Rightarrow$$

$$\tilde{\mathbf{A}} = \mathbf{T}\,\mathbf{A}\,\mathbf{T}^{-1}$$
$$\tilde{\mathbf{B}} = \mathbf{T}\,\mathbf{B}$$
$$\tilde{\mathbf{C}} = \mathbf{T}\,\mathbf{C}$$

A transformation ***T*** *exists for controllable systems*

$$\Rightarrow$$

$$\dot{\mathbf{z}} = \tilde{\mathbf{A}}\,\mathbf{z} + \tilde{\mathbf{B}}\,u$$
$$y = \tilde{\mathbf{C}}\,\mathbf{z} + [\,\mathbf{0}\,]u$$

To place any system in a new coordinate system where the system is in ***Reachable Canonical Form***

where…

$$\tilde{\mathbf{A}} = \begin{bmatrix} -a_1 & -a_2 & \dots & -a_{n-2} & -a_{n-1} & -a_n \\ 1 & 0 & \dots & - & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 & 0 \\ \dots & \dots & \dots & 0 & \dots & \dots \\ 0 & 0 & \dots & 1 & 0 & 0 \\ 0 & 0 & \dots & 0 & 1 & 0 \end{bmatrix} \qquad \tilde{\mathbf{B}} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \qquad \tilde{\mathbf{C}} = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ \vdots \\ \vdots \\ c_n \end{bmatrix}^T$$

Note: Any system that is controllable can be put in this form!

Do the eigenvalues change given this transformation?

No.

noting…

$$CE = \lambda^n + a_1\lambda^{n-1} + a_2\lambda^{n-2} + \cdots + a_{n-1}\lambda + a_n$$

Reachable Canonical Form

$$\dot{\mathbf{z}} = \tilde{\mathbf{A}}\,\mathbf{z} + \tilde{\mathbf{B}}\,u$$

And we can find the reachability matrix for any system in this canonical form using the standard equations.

$$\tilde{\mathbf{B}} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\tilde{\mathbf{A}} = \begin{bmatrix} -a_1 & -a_2 & \dots & -a_{n-2} & -a_{n-1} & -a_n \\ 1 & 0 & \dots & - & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 & 0 \\ \dots & \dots & \dots & 0 & \dots & \dots \\ 0 & 0 & \dots & 1 & 0 & 0 \\ 0 & 0 & \dots & 0 & 1 & 0 \end{bmatrix}$$

$$\Rightarrow$$

$$\tilde{\mathbf{W}}_r = \begin{bmatrix} \tilde{\mathbf{B}} & \tilde{\mathbf{A}}\tilde{\mathbf{B}} & \tilde{\mathbf{A}}^{2}\tilde{\mathbf{B}} & \dots & \dots & \tilde{\mathbf{A}}^{\mathbf{n-1}}\tilde{\mathbf{B}} \end{bmatrix}$$

$$= \begin{bmatrix} 1 & -a_1 & (-a_1^2) - a_2 & (-a_1^3) + (-a_2^2) - a_3 & \dots & \dots \\ 0 & 1 & -a_1 & \dots & \dots & \dots \\ 0 & 0 & 1 & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 0 & 1 & 0 \\ 0 & 0 & \dots & 0 & 0 & 1 \end{bmatrix}$$

*So the trick is to find **T** that puts us in Reachable Canonical Form. If we can do this, we only have to find an algorithm that works for this system to find one that works for every controllable system.*

Reachable Canonical Form

$$\dot{\mathbf{z}} = \tilde{\mathbf{A}}\,\mathbf{z} + \tilde{\mathbf{B}}\,u$$

*So the trick is to find **T** that puts us in Reachable Canonical Form. If we can do this, we only have to find and algorithm that works for this system to find one that works for every controllable system.*

$$\tilde{\mathbf{w}}_r = \begin{bmatrix} \tilde{\mathbf{B}} & \tilde{\mathbf{A}}\tilde{\mathbf{B}} & \tilde{\mathbf{A}}^2\tilde{\mathbf{B}} & \ldots & \ldots & \tilde{\mathbf{A}}^{\mathbf{n-1}}\tilde{\mathbf{B}} \end{bmatrix}$$

where

$$\tilde{\mathbf{B}} = \mathbf{T}\,\mathbf{B}$$

$$\tilde{\mathbf{A}}\,\tilde{\mathbf{B}} = \mathbf{T}\,\mathbf{A}\,\mathbf{T}^{-1}\mathbf{T}\,\mathbf{B} = \mathbf{T}\,\mathbf{A}\,\mathbf{B}$$

$$\tilde{\mathbf{A}}^{2}\tilde{\mathbf{B}} = \mathbf{T}\,\mathbf{A}\,\mathbf{T}^{-1}\mathbf{T}\,\mathbf{A}\,\mathbf{T}^{-1}\mathbf{T}\,\mathbf{B} = \mathbf{T}\,\mathbf{A}^{2}\mathbf{B}$$

$$\tilde{\mathbf{A}}^{3}\tilde{\mathbf{B}} = \mathbf{T}\,\mathbf{A}\,\mathbf{T}^{-1}\mathbf{T}\,\mathbf{A}\,\mathbf{T}^{-1}\mathbf{T}\,\mathbf{A}\,\mathbf{T}^{-1}\mathbf{T}\,\mathbf{B} = \mathbf{T}\,\mathbf{A}^{3}\mathbf{B}$$

etc.

therefore

$$\tilde{\mathbf{w}}_{\mathbf{r}} = \begin{bmatrix} \mathbf{TB} & \mathbf{TAB} & \mathbf{TA}^2\mathbf{B} & \ldots & \ldots & \mathbf{TA}^{\mathbf{n-1}}\mathbf{B} \end{bmatrix} = \mathbf{T}\mathbf{w}_{\mathbf{r}} \Rightarrow \boxed{\mathbf{T} = \tilde{\mathbf{w}}_{\mathbf{r}}\mathbf{w}_{\mathbf{r}}^{-1}}$$

*If the system is observable/controllable, we can find **T** since we can define the reachability matrix in both the original and canonical form!*

State Feedback in RCF systems

So back to our system in Reachable Canonical Form (RCF)

$$\dot{\mathbf{z}} = \begin{bmatrix} -a_1 & -a_2 & \dots & -a_{n-2} & -a_{n-1} & -a_n \\ 1 & 0 & \dots & - & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 & 0 \\ \dots & \dots & \dots & 0 & \dots & \dots \\ 0 & 0 & \dots & 1 & 0 & 0 \\ 0 & 0 & \dots & 0 & 1 & 0 \end{bmatrix} \mathbf{z} + \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} u$$

$$y = \tilde{\mathbf{C}}\mathbf{z} = \begin{bmatrix} c_1 & c_2 & \dots & \dots & \dots & c_n \end{bmatrix} \mathbf{z} + \begin{bmatrix} 0 \end{bmatrix} u$$

We know our characteristic equation (CE) for this system is…

$$CE = \det(\lambda \mathbf{I} - \mathbf{A}) = \lambda^n + a_1\lambda^{n-1} + a_2\lambda^{n-2} + \cdots + a_{n-1}\lambda + a_n$$

And our general control input would be…

$$u = -\tilde{\mathbf{K}}_c \mathbf{z} + k_r y_r = -\tilde{k}_1 z_1 - \tilde{k}_2 z_2 - \cdots - \tilde{k}_n z_n - k_r y_r$$

State Feedback in RCF systems

And our general control input would be…

$$u = -\tilde{\mathbf{K}}_c \mathbf{z} + k_r y_r = -\tilde{k}_1 z_1 - \tilde{k}_2 z_2 - \cdots - \tilde{k}_n z_n - k_r y_r$$

*Which means the RCF system with state feedback (**A-BK**) becomes…*

$$\dot{\mathbf{z}} = \begin{bmatrix} -a_1 - \tilde{k}_1 & -a_2 - \tilde{k}_2 & \dots & -a_{n-2} - \tilde{k}_{n-2} & -a_{n-1} - \tilde{k}_{n-1} & -a_n - \tilde{k}_n \\ 1 & 0 & \dots & - & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & 0 & \vdots & \vdots \\ 0 & 0 & \dots & 1 & 0 & 0 \\ 0 & 0 & \dots & 0 & 1 & 0 \end{bmatrix} \mathbf{z} + \begin{bmatrix} k_r \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} u$$

$$y = \tilde{\mathbf{C}} \mathbf{z} = \begin{bmatrix} c_1 & c_2 & \dots & \dots & \dots & c_n \end{bmatrix} \mathbf{z} + \begin{bmatrix} 0 \end{bmatrix} u$$

We know our characteristic equation (CE) for this system becomes…

$$CE = \det(\lambda \mathbf{I} - \mathbf{A}) = \lambda^n + \left(a_1 - \tilde{k}_1\right)\lambda^{n-1} + \left(a_2 - \tilde{k}_2\right)\lambda^{n-2} + \cdots + \left(a_{n-1} - \tilde{k}_{n-1}\right)\lambda + \left(a_n - \tilde{k}_n\right)$$

State Feedback in RCF systems

We know our characteristic equation (CE) for this system becomes…

$$CE = \det(\lambda \mathbf{I} - \mathbf{A}) = \lambda^n + \left(a_1 - \tilde{k}_1\right)\lambda^{n-1} + \left(a_2 - \tilde{k}_2\right)\lambda^{n-2} + \cdots + \left(a_{n-1} - \tilde{k}_{n-1}\right)\lambda + \left(a_n - \tilde{k}_n\right)$$

So if we have desired closed loop performance characterized by a particular CE (using s instead of λ)

$$CE = \det(\lambda \mathbf{I} - \mathbf{A}) = s^n + p_1 s^{n-1} + p_2 s^{n-2} + \cdots + p_{n-1}s + p_n$$

Then we can find our new gains (in the RCF coordinate system) simply by subtracting…

$\tilde{k}_1 = p_1 - a_1$ *All the way through* $\tilde{k}_n = p_n - a_n$

Or simply

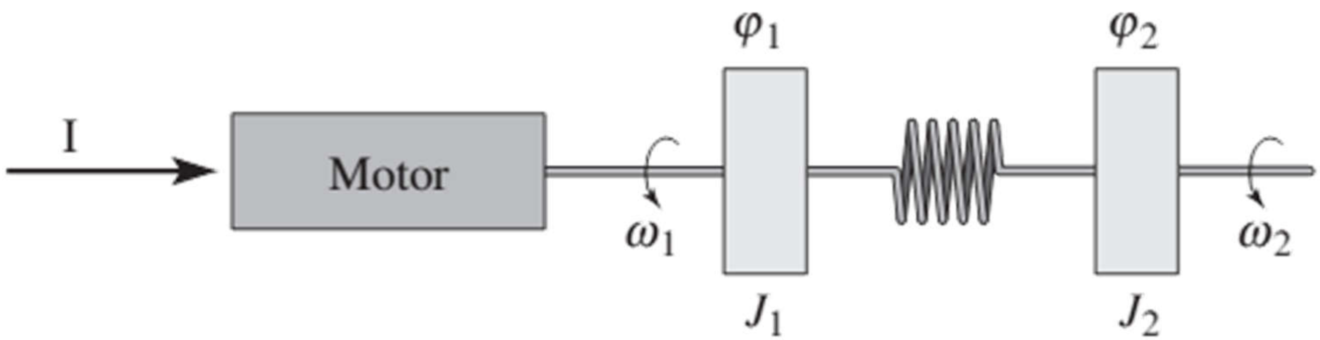
$$\tilde{\mathbf{K}} = \begin{bmatrix} p_1 - a_1 & p_2 - a_2 & \cdots & p_n - a_n \end{bmatrix}$$

Which maps to gains in the original coordinate system, by

$$\mathbf{K} = \tilde{\mathbf{K}}\,\mathbf{T} = \begin{bmatrix} p_1 - a_1 & p_2 - a_2 & \cdots & p_n - a_n \end{bmatrix} \tilde{\mathbf{w}}_r \mathbf{w}_r^{-1}$$

Let's do an example….

Consider a motor driving a system consisting of two inertial disks connected by a compliant shaft:

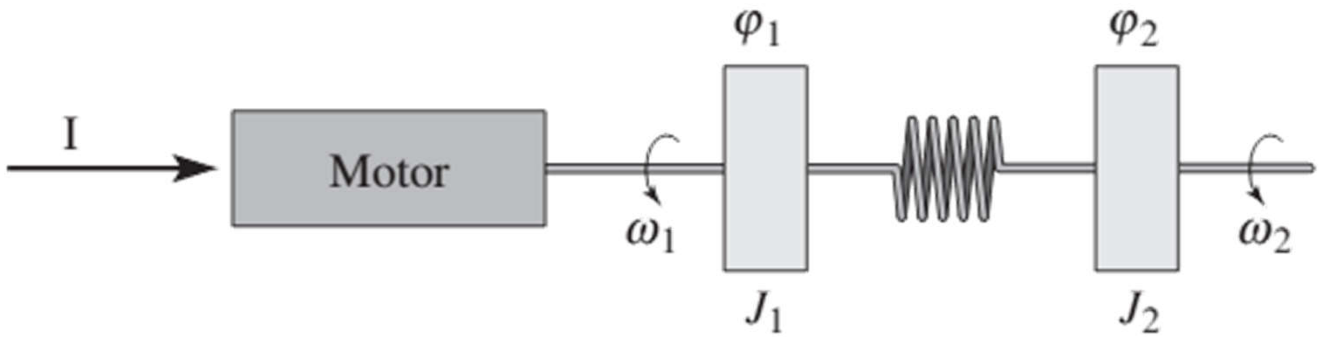


Where the inertial of each disk is given as J_1 and J_2. The spring constant and friction in the shaft are c and k. And the motor constant is k_I.

$$J_1 = 10/9, \quad J_2 = 10, \quad c = 0.1, \quad k = 1, \quad k_I = 1,$$

- *Verify the eigenvalues of the open loop system are 0, 0, and -0.05±i*
- *Design a state feedback controller that produce the system eigenvalues -2, -1, and -1±i.*
- *Simulate the system for a commanded step change in position of the second (outer) inertial disk.*

The equations of motion



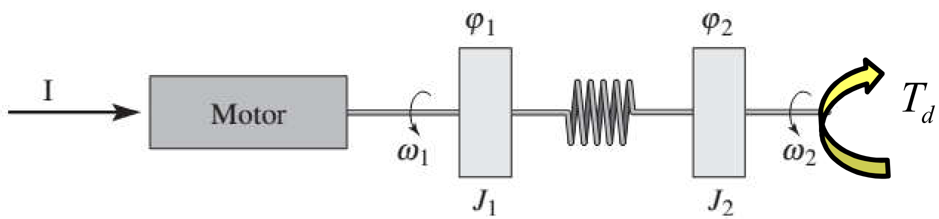
Motor constant

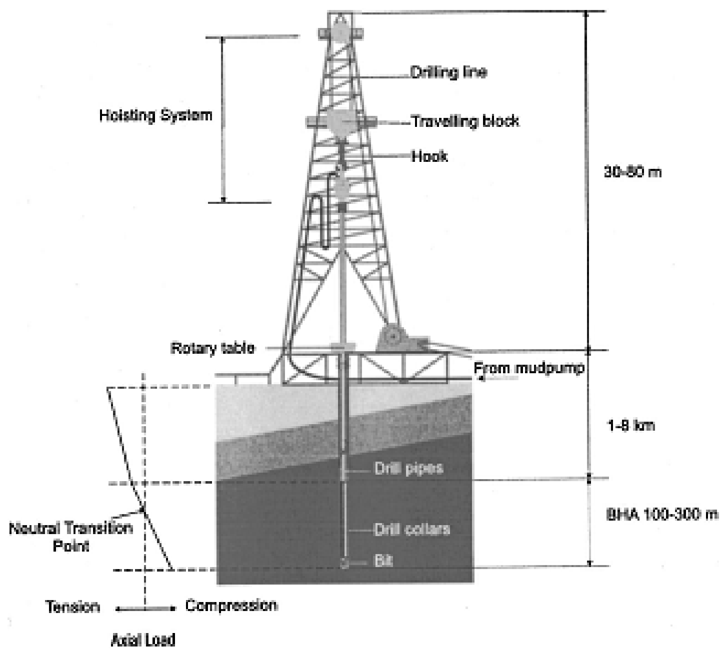
$$J_1 \frac{d^2\varphi_1}{dt^2} + c\left(\frac{d\varphi_1}{dt} - \frac{d\varphi_2}{dt}\right) + k(\varphi_1 - \varphi_2) = k_I I,$$ ←Input Current

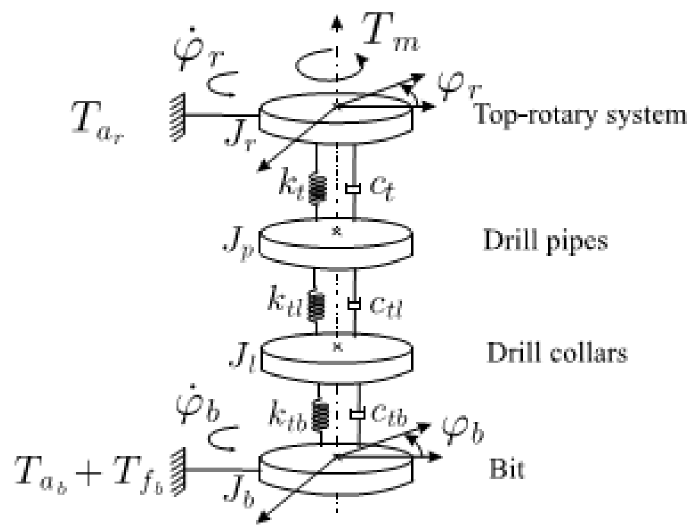
$$J_2 \frac{d^2\varphi_2}{dt^2} + c\left(\frac{d\varphi_2}{dt} - \frac{d\varphi_1}{dt}\right) + k(\varphi_2 - \varphi_1) = T_d.$$ ←Disturbance torque

Derive a state space model for the system by introducing the (normalized) state variables $x_1 = \varphi_1$, $x_2 = \varphi_2$, $x_3 = \omega_1/\omega_0$, and $x_4 = \omega_2/\omega_0$, where $\omega_0 = \sqrt{k(J_1 + J_2)/(J_1 J_2)}$ is the undamped natural frequency of the system when the control signal is zero.

So is this a real thing? Yes.





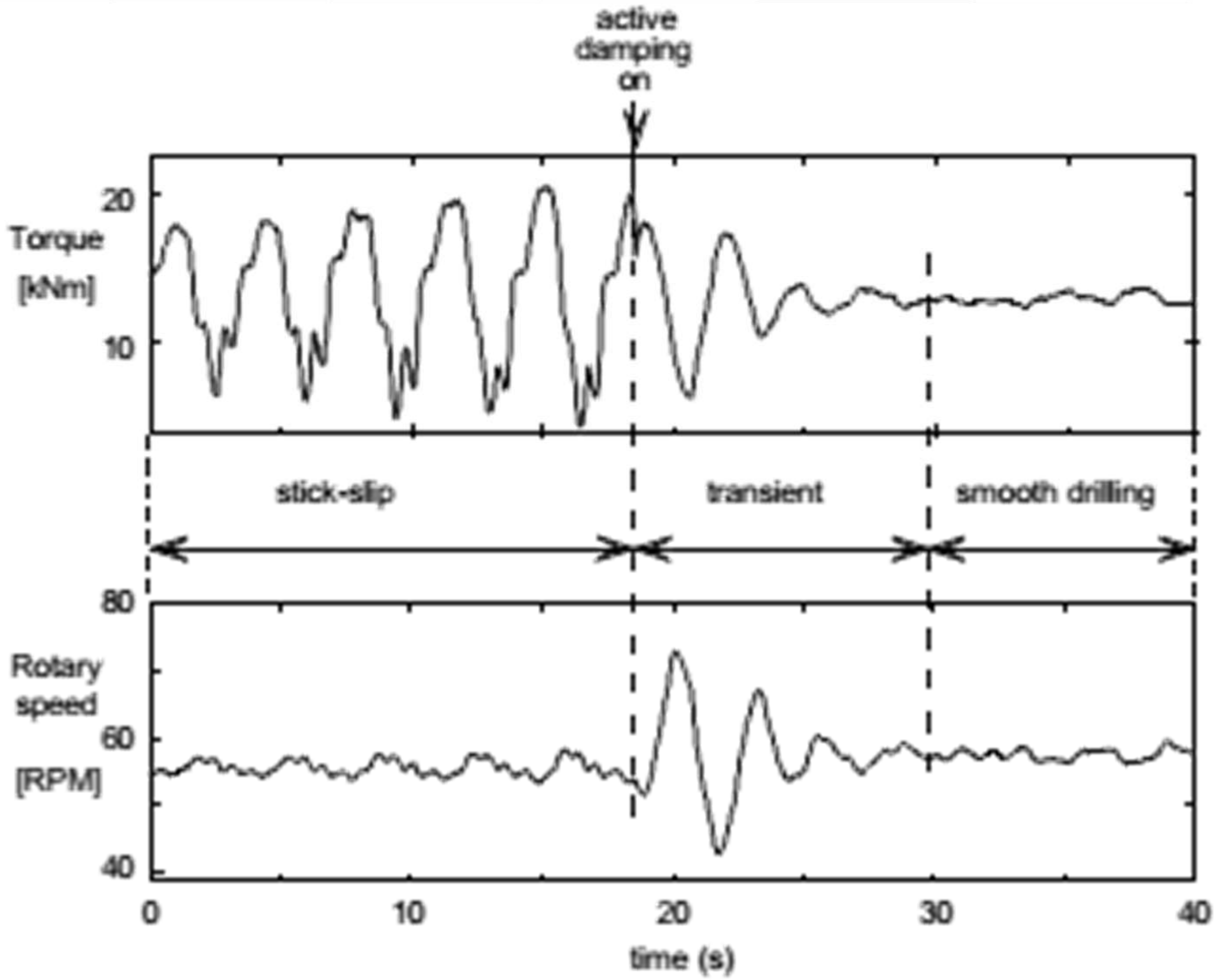


ASME DSC2014-6147 Shor, Pryor, & Van Oort, "Drill string vibration observation, modeling and prevention in the oil and gas industry"

SLB Drillstring Vibration Video

In the film skipped ahead to 0:30 and show for about 2 minutes to show whirl reinitiating.

Soft Torque™ controller system



Or this...

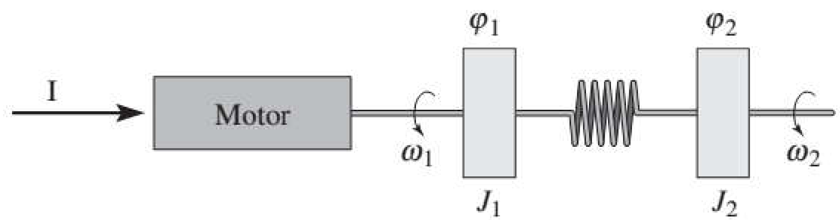


Fig. 3: Robonaut 2 lifting 20 lbs at full extension

Fig. 2: Custom torsion springs from the R2 series elastic actuators

Find the s-s model

We define our state variables…

$$\mathbf{z} = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} = \begin{bmatrix} \varphi_1 \\ \omega_1 \\ \varphi_2 \\ \omega_2 \end{bmatrix}$$

$$J_1 \frac{d^2\varphi_1}{dt^2} = -c(\omega_1 - \omega_2) - k\left(\varphi_1 - \varphi_2\right) + k_1 I$$

$$J_2 \frac{d^2\varphi_2}{dt^2} = -c(\omega_2 - \omega_1) - k\left(\varphi_2 - \varphi_1\right) + T_d$$

$$\Rightarrow$$

or normalized if you like…

$$\mathbf{z} = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} = \begin{bmatrix} \varphi_1 \\ \varphi_2 \\ \omega_1 / \omega_0 \\ \omega_2 / \omega_0 \end{bmatrix}$$

But it doesn't really matter….

Find the s-s model…

$$J_1 \frac{d^2\varphi_1}{dt^2} = -c(\omega_1 - \omega_2) - k\left(\varphi_1 - \varphi_2\right) + k_1 I$$

$$J_2 \frac{d^2\varphi_2}{dt^2} = -c(\omega_2 - \omega_1) - k\left(\varphi_2 - \varphi_1\right) + T_d$$

If we use….

$$\mathbf{z} = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} = \begin{bmatrix} \varphi_1 \\ \omega_1 \\ \varphi_2 \\ \omega_2 \end{bmatrix}$$

$$\frac{dz}{dt} = \mathbf{Az} + \mathbf{Bu}$$

$$= \begin{bmatrix} 0 & 1 & 0 & 0 \\ {}^{-k}/_{J_1} & {}^{-c}/_{J_1} & {}^{k}/_{J_1} & {}^{c}/_{J_1} \\ 0 & 0 & 0 & 1 \\ {}^{k}/_{J_2} & {}^{c}/_{J_2} & {}^{-k}/_{J_2} & {}^{-c}/_{J_2} \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ {}^{k_I}/_{J_1} & 0 \\ 0 & 0 \\ 0 & {}^{1}/_{J_2} \end{bmatrix} \begin{bmatrix} I \\ T_d \end{bmatrix}$$

With…

$$J_1 = 10/9$$
$$J_2 = 10$$
$$c = 0.1$$
$$k = 1$$
$$k_I = 1$$

$$= \begin{bmatrix} 0 & 1 & 0 & 0 \\ -.9 & -.09 & .9 & .09 \\ 0 & 0 & 0 & 1 \\ .1 & .01 & -.1 & -.01 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ .9 & 0 \\ 0 & 0 \\ 0 & .1 \end{bmatrix} \begin{bmatrix} I \\ T_d \end{bmatrix}$$

Find the s-s model…

$$J_1 \frac{d^2\varphi_1}{dt^2} = -c(\omega_1 - \omega_2) - k\left(\varphi_1 - \varphi_2\right) + k_1 I$$

$$J_2 \frac{d^2\varphi_2}{dt^2} = -c(\omega_2 - \omega_1) - k\left(\varphi_2 - \varphi_1\right) + T_d$$

If we use….

$$\mathbf{z} = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} = \begin{bmatrix} \varphi_1 \\ \varphi_2 \\ \omega_1/\omega_0 \\ \omega_2/\omega_0 \end{bmatrix} \qquad \frac{dz}{dt} = \mathbf{Az} + \mathbf{Bu}$$

$$= \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -\omega_o k/J_1 & \omega_o k/J_1 & -\omega_o c/J_1 & \omega_o c/J_1 \\ \omega_o k/J_2 & -\omega_o k/J_2 & \omega_o c/J_2 & -\omega_o c/J_2 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ k_I/J_1 & 0 \\ 0 & 1/J_2 \end{bmatrix} \begin{bmatrix} I \\ T_d \end{bmatrix}$$

With…

$$J_1 = 10/9$$
$$J_2 = 10$$
$$c = 0.1$$
$$k = 1$$
$$k_I = 1$$

$$= \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -.9 & .09 & -.9 & 0.09 \\ .1 & -.1 & .01 & -.01 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ .9 & 0 \\ 0 & .1 \end{bmatrix} \begin{bmatrix} I \\ T_d \end{bmatrix}$$

What is interesting about these values? *Advice: test with simple parameters!*

Find the s-s model…

$$J_1 \frac{d^2\varphi_1}{dt^2} = -c(\omega_1 - \omega_2) - k\left(\varphi_1 - \varphi_2\right) + k_1 I$$

$$J_2 \frac{d^2\varphi_2}{dt^2} = -c(\omega_2 - \omega_1) - k\left(\varphi_2 - \varphi_1\right) + T_d$$

Or if we use….

$$\mathbf{z} = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} = \begin{bmatrix} \varphi_1 \\ \omega_1 \\ \varphi_2 \\ \omega_2 \end{bmatrix} \qquad \frac{dz}{dt} = \mathbf{Az} + \mathbf{Bu} + d$$

$$= \begin{bmatrix} 0 & 1 & 0 & 0 \\ {}^{-k}\!/_{J_1} & {}^{-c}\!/_{J_1} & {}^{k}\!/_{J_1} & {}^{c}\!/_{J_1} \\ 0 & 0 & 0 & 1 \\ {}^{k}\!/_{J_2} & {}^{c}\!/_{J_2} & {}^{-k}\!/_{J_2} & {}^{-c}\!/_{J_2} \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} + \begin{bmatrix} 0 \\ {}^{k_I}\!/_{J_1} \\ 0 \\ 0 \end{bmatrix} I + \cancel{\begin{bmatrix} 0 \\ 0 \\ 0 \\ {}^{1}\!/_{J_2} \end{bmatrix}} T_d$$

With…

$$J_1 = 10/9$$
$$J_2 = 10$$
$$c = 0.1$$
$$k = 1$$
$$k_I = 1$$

$$= \begin{bmatrix} 0 & 1 & 0 & 0 \\ -.9 & -.09 & .9 & .09 \\ 0 & 0 & 0 & 1 \\ .1 & .01 & -.1 & -.01 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} + \begin{bmatrix} 0 \\ .9 \\ 0 \\ 0 \end{bmatrix} I + \cancel{\begin{bmatrix} 0 \\ 0 \\ 0 \\ .1 \end{bmatrix}} T_d$$

So does normalization matter?

```
%main function
clear all;

%system parameters
J1 = 10/9; J2 = 10;
c = 0.1; k = 1; ki = 1;

%state space model
A = [ 0 1 0 0;
    -k/J1 -c/J1 k/J1 c/J1;
    0 0 0 1;
    k/J2 c/J2 -k/J2 -c/J2 ];

%state space model normalized
w0 = sqrt( k*(J1+J2)/(J1*J2) );

An = [ 0 0 1 0;
    0 0 0 1;
    -w0*k/J1 w0*k/J1 -w0*c/J1 w0*c/J1;
    w0*k/J2 -w0*k/J2 w0*c/J2  -w0*c/J2 ];

%verify open loop eigenvalues
eig(A)
eig(An)
cond(A)
cond(An)
```

```
>> Prob6_11

ans =

  -0.0500 + 0.9987i
  -0.0500 - 0.9987i
   0.0000 + 0.0000i
   0.0000 - 0.0000i


ans =

  -0.0500 + 0.9987i
  -0.0500 - 0.9987i
   0.0000 + 0.0000i
   0.0000 - 0.0000i

ans =   2.8921e+17

ans =   2.1357e+18
```

No…. (and we got the right answer)

But, whoa….

State Feedback for RCF Systems Summary

For controllable systems we can find state feedback gains to give us a desired performance CE.

$$\mathbf{K} = \tilde{\mathbf{K}}\mathbf{T} = \begin{bmatrix} p_1 - a_1 & p_2 - a_2 & \cdots & p_n - a_n \end{bmatrix} \tilde{\mathbf{w}}_r \mathbf{w}_r^{-1}$$

We find the feedforward terms using the same equations as before.

$$k_r = \frac{-1}{\mathbf{C}\left[\mathbf{A} - \mathbf{B}\mathbf{K}\right]^{-1}\mathbf{B}}$$

We already have a basic understanding of the relationship between eigenvalues and selected performance indices, and we will learn more later in the course.

For now, for second order systems we can map our equations for rise time, overshoot, etc. to the desired natural frequency and damping coefficients, which can then be mapped to eigenvalues using the 2nd order canonical form.

$$\lambda^2 + 2\zeta\omega_n\lambda + \omega_n^2$$

On option for higher order systems is to select coefficients that neglect higher order terms so the system dynamics can be approximated as a second order system.

Started an example. Reviewed normalization. Will continue example in next lesson!