State Space Modeling

Key Terms:

- **State:** The set of variables for which knowledge of the variables at $t=0$ and the inputs at t completely determines the behavior of the system

- **State Variables:** The variables that determine the dynamic state of a system. These are the minimum set required to completely describe the behavior of a dynamic system.

- **State Vector:** A vector of the state variables.

- **State space:** The n-dimensional space made from the axes of the state variables

- **State-space equations:** The set of equations relating the input variables, state variables, and output variables for a dynamic system. For a linear, time-invariant system, these take the form

$$\boxed{\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned}}$$

$\leftarrow$ state equation
$\leftarrow$ output equation

x: state vector
u: input/control vector
y: output vector

A: state matrix
B: input/control matrix
C: output matrix
D: direct transmission matrix

} matrix of constants (linear)

Often referred to as "modern" or "time domain" approach
Can also be used on time varying systems

Hopefully you've had a strong introduction to state space models in AE 3530, but we will review briefly

What does each term represent?

natural dynamics of system

the input added to the system

$$\boxed{\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned}}$$

the direct (feedforward) effect on the output as a result of the input, regardless of the natural dynamics

the output of the system as a result of the natural dynamics in response to the input

since we are often concerned with position & velocity of mechanical systems, this is often zero

Criteria for selecting state vector

1) The minimum set of variables needed to fully capture the state of the system

2) The components of the state vector should be linearly independant

⇒ linearly independant → knowledge of a subset of the variables does not lead to knowledge of another

⇒ ex: if $x_3 = 4x_1 + 2x_2$, x_1 and x_2 are linearly independant, but x_3 is not because its state can be determined if the states of x_1 and x_2 are known

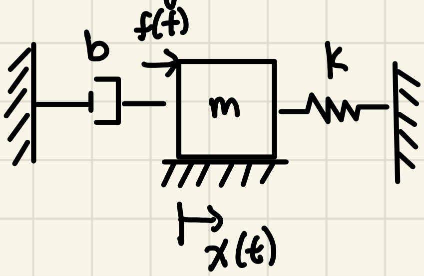
How do we determine the minimum number of variables?

⇒ in general, minimum variables = order of diff eq.

⇒ ex: if a third order differential equation describes a system, the state space model will include:

- 3 simultaneous first order diff eqns
- 3 state variables

Let's go back to our simple harmonic oscillator



in response to forcing function $f(t)$, the block is displaced $x(t)$

$$\sum F = ma \Rightarrow f(t) - kx(t) - b\dot{x}(t) = m\ddot{x}(t)$$

$$f(t) = m\ddot{x}(t) + b\dot{x}(t) + kx(t)$$

1) Define state variables

$$x_1 = x$$
$$x_2 = \dot{x}$$

2) Write differential equations for each state variable

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = -\frac{b}{m}x_2 - \frac{k}{m}x_1 + \frac{1}{m}f(t)$$

3) Put in Matrix form

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \overbrace{\begin{bmatrix} 0 & 1 \\ -k/m & -b/m \end{bmatrix}}^{A} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \overbrace{\begin{bmatrix} 0 \\ 1/m \end{bmatrix}}^{B} f(t)$$

$$y = \underbrace{\begin{bmatrix} 1 & 0 \end{bmatrix}}_{C} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \underbrace{0}_{D} f(t)$$

Let's use Matlab to look at the response, and compare to our response using the TF

```
t = 0:.01:10
k = 5
b = 4
m = 1
```

recall these are the inputs we used before

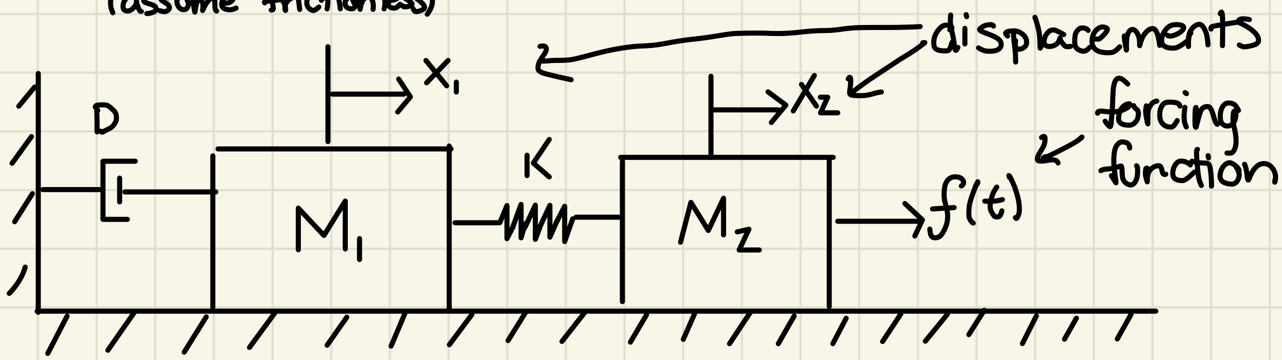
```
A = [0  1 ; -k/m  -b/m]
B = [0 ; 1/m]
C = [1  0]
D = 0

sys = ss(A, B, C, D)
opt = stepDataOptions('Step Amplitude', 5)   ← from before
y = step(sys, t, opt)
plot(t, y)
grid
```

note that this is identical to TF representation
but now we could look at velocity

like before, we now can play with our model

Example: Find the state equations for this system

(assume frictionless)



1) Write the differential equations for each mass, assume zero inital conditions

M_1: $M_1\ddot{x}_1 + D\dot{x}_1 + Kx_1 - Kx_2 = 0$

M_2: $-Kx_1 + M_2\ddot{x}_2 + Kx_2 = f(t)$

we have 2 2^{nd} order equations $\Rightarrow$ 4 state variables (2 per mass)

2) Select state variables (we'll use position & velocity for each mass)

x_1 $\quad v_1 = \dot{x}_1$

x_2 $\quad v_2 = \dot{x}_2$

3) Write the state equations $\Rightarrow$ each diff eqn $\Rightarrow$ 2 state eqns

from M_1 equations: $\dot{x}_1 = v_1$

$$\dot{v}_1 = -\frac{K}{M_1}x_1 - \frac{D}{M_1}v_1 + \frac{K}{M_1}x_2$$

from M_2 equations: $\dot{x}_2 = v_2$

$$\dot{v}_2 = \frac{K}{M_2}x_1 - \frac{K}{M_2}x_2 + \frac{1}{M_2}f(t)$$

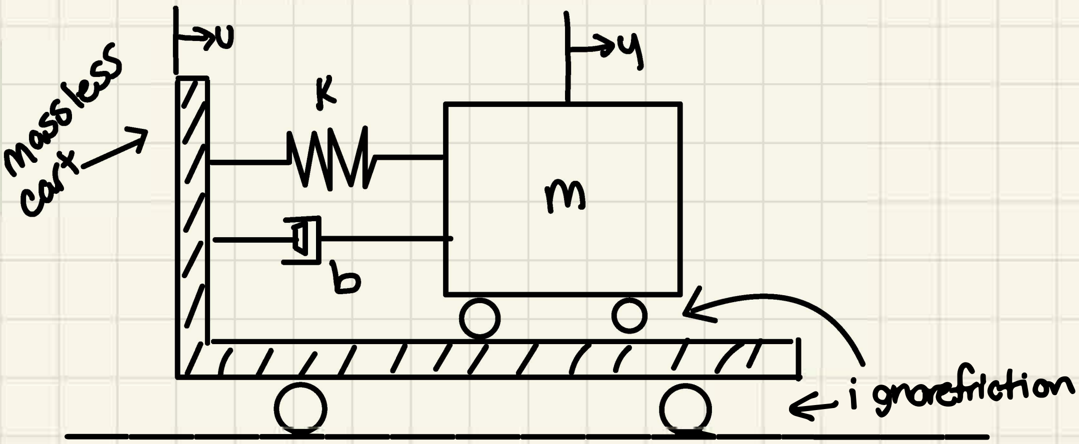
4) Put it in matrix form $\dot{x} = Ax + Bu$

$$\begin{bmatrix} \dot{x}_1 \\ \dot{v}_1 \\ \dot{x}_2 \\ \dot{v}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -K/M_1 & -D/M_1 & K/M_1 & 0 \\ 0 & 0 & 0 & 1 \\ K/M_2 & 0 & -K/M_2 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ v_1 \\ x_2 \\ v_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1/M_2 \end{bmatrix} f(t)$$

since position is our desired output

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ v_1 \\ x_2 \\ v_2 \end{bmatrix}$$

Another Example This time, let's make a transfer function model using LaPlace and a state-space model



At $t=0$, the cart is moved at a constant speed $\dot{u}$

$u(t)$ is the cart displacement and is the system input

$y(t)$ is the output, the mass displacement

we assume the friction force from the dashpot is proportional to $(\dot{y}-\dot{u})$ and the spring is linear and proportional to $(y-u)$

Let's write the differential equation for the mass

$$m\ddot{y} = -b(\dot{y}-\dot{u}) - k(y-u)$$

$$\text{or } m\ddot{y} + b\dot{y} + ky = b\dot{u} + ku$$

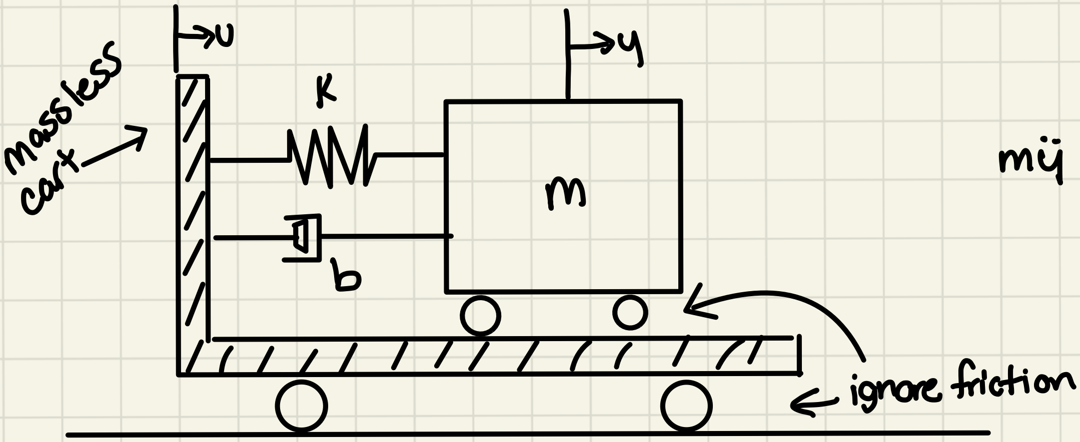
the Laplace w/ zero initial condition gives

$$(ms^2 + bs + k)\,Y(s) = (bs + k)U(s)$$

so our transfer function is:

$$\boxed{\frac{Y(s)}{U(s)} = \frac{bs + k}{ms^2 + bs + k}}$$

This is one way we can mathematically represent this system



$$m\ddot{y} + b\dot{y} + ky = b\dot{u} + ku$$

Now let's do a state space model

$$\ddot{y} + \frac{b}{m}\dot{y} + \frac{k}{m}y = \frac{b}{m}\dot{u} + \frac{k}{m}u$$

↑ There is a derivative of the input fxn!

In this case, we can use the equations presented in Ch 5.
Eq (5·21 → 5-31) to define our state variables and build our state space representation

→ These tell us that if we have a system of the form:

$$\ddot{y} + a_1\dot{y} + a_2 y = b_0\ddot{u} + b_1\dot{u} + b_2 u$$

we can choose state variables

$$x_1 = y - \beta_0 u$$
$$x_2 = \dot{x}_1 - \beta_1 u$$

where
$$\beta_0 = b_0$$
$$\beta_1 = b_1 - a_1\beta_0$$
$$\beta_2 = b_2 - a_1\beta_1 - a_2\beta_0$$

such that

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -a_2 & -a_1 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} \beta_1 \\ \beta_2 \end{bmatrix} u$$

and $y = \begin{bmatrix} 1 & 0 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \beta_0 u$

For our problem

$$\ddot{y} + \frac{b}{m}\dot{y} + \frac{k}{m}y = \frac{b}{m}\dot{u} + \frac{k}{m}u$$

$$\begin{array}{cccc} \uparrow & \uparrow & \uparrow & \uparrow \\ a_1 & a_2 & b_1 & b_2 \end{array}$$

$b_0 = 0$

so $\beta_0 = b_0 = 0$

$$\beta_1 = b_1 - a_1\beta_0 = \frac{b}{m}$$

$$\beta_2 = b_2 - a_1\beta_1 - a_2\beta_0 = \frac{k}{m} - \left(\frac{b}{m}\right)^2$$

and our state variables are

$$x_1 = y - \beta_0 u = y$$

$$x_2 = \dot{x}_1 - \beta_1 u = \dot{x}_1 - \frac{b}{m}u$$

and

$$\boxed{\begin{aligned} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ -k/m & -b/m \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} b/m \\ k/m - (b/m)^2 \end{bmatrix} u \\ y &= \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \end{aligned}}$$

This is equivalent way to model the system

This example is implemented in Matlab. Please refer to **mass_on_cart.m**.

Note that

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \text{ where}$$

$x_1 = y$ and $x_2 = \dot{x}_1 - \frac{b}{m} u$

While $x_1 = y(t)$ corresponds to the displacement of the mass 'm', x_2 does not have an obvious meaningful ~~rep~~ physical quantity representation.

Hence, to obtain $\dot{y}$, i.e. the velocity of the mass 'm', we must do the following:

Differentiate the x_1 equation & substitute into the x_2 equation

$$\therefore \dot{x}_1 = \dot{y}$$

$$x_2 = \dot{x}_1 - \frac{b}{m} u = \dot{y} - \frac{b}{m} u$$

Now, on rearranging

$$\dot{y} = x_2 + \frac{b}{m} u$$

$$\therefore \boxed{\begin{aligned} y(t) &= x_1 \\ \dot{y}(t) &= x_2 + \frac{b}{m} u \end{aligned}}$$

Let's compare our models:

$$\boxed{\frac{Y(s)}{U(s)} = \frac{bs + k}{ms^2 + bs + k}}$$

$$\boxed{m\ddot{y} + b\dot{y} + ky = b\dot{u} + ku}$$

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -k/m & -b/m \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} b/m \\ k/m - (b/m)^2 \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

where $x_1 = y$ and $x_2 = \dot{x}_1 - \frac{b}{m} u$

Let's specifically consider the natural dynamics

TF: poles: $-\frac{b}{2m} \pm \frac{\sqrt{(b/m)^2 - 4k/m}}{2}$

Eigens of Matrix A:

$$\left| \begin{bmatrix} 0 - \lambda & 1 \\ -k/m & -b/m - \lambda \end{bmatrix} \right| = (-\lambda)(-b/m - \lambda) + k/m$$

$$= \lambda^2 + b/m\,\lambda + k/m$$

$$\lambda = -\frac{b}{2m} \pm \frac{\sqrt{(b/m)^2 - 4k/m}}{2}$$

What do you notice?

These are equivalent ways to represent our system, but we need to keep in mind some differences in the approaches

Transfer Functions	State Space
"Classical" Approach	"Modern" Approach
Applies to SISO LTI systems	Can apply linear & non-linear, time-variant or invariant, and SISO & MIMO system with varying degrees of complexity
I.C. are zero	I.C. can be considered
enabler for Bode, Root Locus generally easy to work with	enabler for more precise control methods, more computaionally expensive
no insight to internal variables	gives insight to internal state

Converting SS to TF

Remember that I said a system could be represented equally in state-space or by a transfer function?

We can convert between these representations!

We do this by taking the LaPlace transforms of our state space equations with zero initial conditions

$\dot{x} = Ax + Bu \overset{\mathcal{L}}{\Rightarrow} sX(s) = AX(s) + BU(s)$

$y = Cx + Du \overset{\mathcal{L}}{\Rightarrow} Y(s) = CX(s) + DU(s)$

the transfer function is $Y(s)/U(s)$, so we solve for $X(s)$ from our first equation and plug it into the second

identity matrix

$$(sI - A)X(s) = BU(s)$$

$$X(s) = (sI - A)^{-1} BU(s)$$

plug it in!

$$Y(s) = C(sI - A)^{-1} BU(s) + DU(s)$$

and

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

note: We can also do this in matlab with ss2tf functions
tf2ss

Up until this point, we have focused on modeling fairly simple systems.

In general, we are interested in deriving models for more complex systems.

Block diagrams are a technique to help us derive models for more complex systems from simple models of their components and their relationships

After we learn how to use block diagrams to represent and create transfer functions for more complex systems, we are going to revisit those poles/zeros/gains to start learning about controlling things!