LECTURE 24  - nonlinear systems  - linearization

So far discussed linear systems:

\[
\frac{d\mathbf{x}(t)}{dt} = A\mathbf{x}(t) + B\mathbf{u}(t) \quad \mathbf{x}[i+1] = A\mathbf{x}[i] + B\mathbf{u}[i]
\]

Today nonlinear:

\[
\frac{d\mathbf{x}(t)}{dt} = \mathbf{f}(\mathbf{x}(t),\mathbf{u}(t)) \quad \mathbf{x}[i+1] = \mathbf{f}(\mathbf{x}[i],\mathbf{u}[i])
\]

where \(\mathbf{f}: \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n\) is a vector-valued function of state \(\mathbf{x} \in \mathbb{R}^n\) and input \(\mathbf{u} \in \mathbb{R}^m\).

Linear systems are a special case: \(\mathbf{f}(\mathbf{x},\mathbf{u}) = A\mathbf{x} + B\mathbf{u}\)

Example 1:

\[
ml \frac{d^2\theta(t)}{dt^2} = -kl \frac{d\theta(t)}{dt} - mgs\sin\theta(t) - (1)
\]

Let \(\mathbf{x}(t) = \begin{bmatrix} \theta(t) \\ w(t) \end{bmatrix}\)

\(w(t) = \frac{d\theta(t)}{dt}\) (angular velocity)
Equilibrium (Operating) Points:

For a continuous-time system with no input

\[
\frac{d}{dt} \mathbf{x}(t) = \mathbf{f}_c(\mathbf{x}(t))
\]

the solutions of the static eq'n \( \mathbf{f}_c(\mathbf{x}) = 0 \) are called equilibrium points. If \( \mathbf{x}^* \) is an equilibrium, i.e., if \( \mathbf{f}_c(\mathbf{x}^*) = 0 \), then \( \dot{\mathbf{x}}(t) = \mathbf{x}^* \) is a solution of diff. eq'n above with \( \mathbf{x}(0) = \mathbf{x}^* \): substitute \( \mathbf{x}(t) = \mathbf{x}^* \) in (2):

\[
\frac{d}{dt} \mathbf{x}^* = \mathbf{f}_c(\mathbf{x}^*) = 0 \quad \text{be} / \mathbf{x}^* = \text{cst.}
\]

Pendulum example: \( \mathbf{f}_c(\mathbf{x}) = \begin{bmatrix} \frac{x_2}{2} \\ -\frac{k}{m} x_2 - \frac{g}{l} \sin(x_1) \end{bmatrix} = 0 \)
\[
\begin{align*}
\Rightarrow & \quad \left\{ \begin{array}{l}
x_2 = 0 \\
- \frac{k}{m} x_2 - \frac{g}{L} \sin x_1 = 0 
\end{array} \right. \quad \text{--- (3)} \\
\sin x_1 = 0 & \Rightarrow x_1 = 0, \pi \quad \text{--- (4)}
\end{align*}
\]

Substitute (3) in (4): \( \sin x_1 = 0 \Rightarrow x_1 = 0, \pi \)

Two equilibrium points: \( (x_1, x_2) = (0, 0) \) \( \downarrow \) \text{downward pointing} \( \uparrow \) \text{upward pointing} \( (x_1, x_2) = (\pi, 0) \)

What about discrete time equilibria?

\[
\tilde{x}[i+1] = \tilde{f}_d(\tilde{x}[i]) \text{ --- (5)}
\]

\( \tilde{x}^* \text{ is an equilibrium if: } \tilde{x}^* = \tilde{f}_d(\tilde{x}^*) \).

\( \tilde{x}[i] = \tilde{x}^* \) for all \( i \) is a solution of (5) because

\[
\tilde{x}[i] = \tilde{x}^* \Rightarrow \tilde{x}[i+1] = \tilde{f}_d(\tilde{x}[i]) = \tilde{f}_d(\tilde{x}^*) = \tilde{x}^*.
\]

Systems with inputs:

\( (\tilde{x}^*, \tilde{u}^*) \) is an “operating point” of

\[
\frac{d}{dt} \tilde{x}(t) = \tilde{f}_c(\tilde{x}(t), \tilde{u}(t)) \text{ --- (6)}
\]

if \( \tilde{f}_c(\tilde{x}^*, \tilde{u}^*) = 0 \text{ --- (7)} \)

\( (\tilde{x}^*, \tilde{u}^*) \) is an “operating point” of

\[
\tilde{x}[i+1] = \tilde{f}_d(\tilde{x}[i], \tilde{u}[i])
\]

if \( \tilde{x}^* = \tilde{f}_d(\tilde{x}^*, \tilde{u}^*) \)
If we apply the constant input \( \dot{u}(t) = \dot{u}^* \) then \( \dot{x}(t) = \dot{x}^* \) is a solution of (6) with \( x(0) = \dot{x}^* \).

**Example 2:**

\[
M \frac{d\dot{u}(t)}{dt} = -\beta \dot{u}(t)^2 + \frac{1}{R} u(t)
\]

\( x = \dot{v} \) is the (single) state, \( f(x, u) = -\frac{\beta}{M} x^2 + \frac{1}{RM} u \).

\((x^*, u^*)\) is an operating point if (from Eq. 7):

\[
f(x^*, u^*) = 0 \Rightarrow u^* = \beta R x^*.
\]

If we want speed \( x^* \) we must apply torque \( u^* \) to overcome drag \( \beta R x^* \).

**Linearization:** linear approximation of nonlinear model around an operating point

Easy when \( x \in R \):

1) no input: \( \frac{d}{dt} x(t) = f(x(t)) \), \( f(x^*) = 0 \) \[ (8) \]
   
   Taylor approximation: \( f(x) \approx f(x^*) + f'(x^*) (x - x^*) \)

   \[ f(x) = 0 \]
   
   \( f(x) \)
Define $\delta x(t) = x(t) - x^*$. Then:

$$\frac{d}{dt} \delta x(t) = \frac{d}{dt} \left( x(t) - x^* \right) = \frac{d}{dt} x(t) - \frac{d}{dt} x^* = f(x(t))$$

(8)

$\propto f'(x^*) \delta x(t)$

Linearized model:

$$\frac{d}{dt} \delta x(t) = f'(x^*) \delta x(t)$$

Equation (8)

$$\Rightarrow \lambda$$

2) with input $u \in \mathbb{R}$:

$$\frac{d}{dt} x(t) = f(x(t), u(t))$$

Suppose $(x^*, u^*)$ operating point: $f(x^*, u^*) = 0$

$$f(x, u) \approx f(x^*, u^*) + \frac{\partial f}{\partial x}(x^*, u^*)(x - x^*) + \frac{\partial f}{\partial u}(x^*, u^*)(u - u^*)$$

$$= 0$$

$$\Rightarrow \lambda$$

$$\begin{align*}
\frac{d}{dt} \delta x(t) &= \frac{d}{dt} x(t) - \frac{d}{dt} x^* \\
&= \lambda \delta x(t) + b \delta u(t)
\end{align*}$$

Example 2: $f(x, u) = -\frac{B}{M} x^2 + \frac{1}{RM} u$

$$\begin{align*}
\frac{\partial f}{\partial x}(x, u) &= -\frac{2B}{M} x \\
\frac{\partial f}{\partial u}(x, u) &= \frac{1}{RM}
\end{align*}$$

$$\begin{align*}
\lambda &= \frac{\partial f}{\partial x}(x^*, u^*) = -\frac{2B}{M} x^* \\
b &= \frac{\partial f}{\partial u}(x^*, u^*) = \frac{1}{RM}
\end{align*}$$
\[ \frac{d}{dt} \Delta x(t) = 2 \Delta x(t) + b \Delta u(t) \]

where \( \Delta x(t) = x(t) - x^* \), \( \Delta u(t) = u(t) - u^* \).

\( u^* = \beta R x^*^2 \). Assume we apply \( \Delta u = 0 \)

\((u(t) = u^*) : \quad \frac{d}{dt} \Delta x(t) = 2 \Delta x(t) \)

\[ \Rightarrow \Delta x(t) = e^{2t} \Delta x(0) \]

\[ \lambda = -\frac{2\beta x^*}{M} < 0 \quad \text{so} \quad \Delta x(t) \to 0, \ i.e., \ x(t) \to x^*. \]

If \( \lambda \) not negative enough (slow convergence to \( x^* \)) we can apply feedback:

\[ \Delta u(t) = k \Delta x(t) \]

Closed-loop system: \( \frac{d}{dt} \Delta x(t) = (\lambda + bk) \Delta x(t) \)

We can choose \( k \) to make \( \lambda + bk \) as negative as we wish:

\[ \Delta x(t) = e^{(\lambda + bk)t} \Delta x(0) \]

\[ \Rightarrow 0 \quad \text{faster} \]

\[ u(t) = u^* + \Delta u(t) \]

\[ = \beta R x^*^2 + k \Delta x(t) \]

\[ u(t) = \beta R x^*^2 + k (x(t) - x^*) \quad \text{Cruise Control} \]
Next assume $\tilde{\mathbf{x}} \in \mathbb{R}^2$, $u \in \mathbb{R}$.

$\tilde{f}(\mathbf{x}, u) \in \mathbb{R}^2$, so we can write $\tilde{f}(\mathbf{x}, u)$ as

\[
\begin{bmatrix}
  f_1(x_1, x_2, u) \\
  f_2(x_1, x_2, u)
\end{bmatrix}
\]

where $f_1, f_2$ are scalar valued functions.

As before, 

\[ f_1(x_1, x_2, u) \approx f_1(x_1^*, x_2^*, u^*) + \frac{\partial f_1}{\partial x_1} (x_1 - x_1^*) 
+ \frac{\partial f_1}{\partial x_2} (x_2 - x_2^*) 
+ \frac{\partial f_1}{\partial u} (u - u^*) \tag{9} \]

Similarly, 

\[ f_2(x_1, x_2, u) \approx f_2(x_1^*, x_2^*, u^*) + \frac{\partial f_2}{\partial x_1} (x_1 - x_1^*) 
+ \frac{\partial f_2}{\partial x_2} (x_2 - x_2^*) 
+ \frac{\partial f_2}{\partial u} (u - u^*) \tag{10} \]

Combining (9)-(10) in matrix/vector form:

\[
\begin{bmatrix}
  f_1(x_1, x_2, u) \\
  f_2(x_1, x_2, u)
\end{bmatrix} \approx \begin{bmatrix}
  \frac{\partial f_1}{\partial x_1} (x_1^*, x_2^*, u^*) & \frac{\partial f_1}{\partial x_2} (x_1^*, x_2^*, u^*) \\
  \frac{\partial f_2}{\partial x_1} (x_1^*, x_2^*, u^*) & \frac{\partial f_2}{\partial x_2} (x_1^*, x_2^*, u^*)
\end{bmatrix} \begin{bmatrix}
  \delta x_1 \\
  \delta x_2
\end{bmatrix} + \begin{bmatrix}
  \frac{\partial f_1}{\partial u} (x_1^*, x_2^*, u^*) \\
  \frac{\partial f_2}{\partial u} (x_1^*, x_2^*, u^*)
\end{bmatrix} \delta u 
\]

(11)
Then, 
\[
\frac{d}{dt} \left[ \frac{\delta x(t)}{\delta x(t)} \right] = \frac{d}{dt} [x(t)] - \frac{d}{dt} [x^*] = \mathbf{f}(x(t), u(t))
\]

and substitution of (11) gives linearized model:

\[
\frac{d}{dt} \left[ \delta x(t) \right] = A \left[ \delta x(t) \right] + B \delta u(t),
\]

where \(A, B\) are as defined in (11).

Easily generalizable to \(x \in \mathbb{R}^n, u \in \mathbb{R}^m\).

\[
\mathbf{f}(x, u) = \begin{bmatrix}
    f_1(x_1, \ldots, x_n, u_1, \ldots, u_m) \\
    \vdots \\
    f_n(x_1, \ldots, x_n, u_1, \ldots, u_m)
\end{bmatrix}
\]

Linearized model at a given operating point \((x^*, u^*)\) is:

\[
\frac{d}{dt} \delta x(t) = A \delta x(t) + B \delta u(t)
\]

where, \(\delta x(t) = x(t) - x^*, \delta u(t) = u(t) - u^*\),

\[
A = \begin{bmatrix}
    \frac{\partial f_1}{\partial x_1}(x^*, \ldots, x^*, u^*, \ldots, u_m^*) & \cdots & \frac{\partial f_1}{\partial x_n}(x^*, \ldots, x^*, u^*, \ldots, u_m^*) \\
    \vdots & & \vdots \\
    \frac{\partial f_n}{\partial x_1}(x^*, \ldots, x^*, u^*, \ldots, u_m^*) & \cdots & \frac{\partial f_n}{\partial x_n}(x^*, \ldots, x^*, u^*, \ldots, u_m^*)
\end{bmatrix}
\]

that is, \(A(i, j) = \frac{\partial f_i}{\partial x_j}(x^*, \ldots, x^*, u^*, \ldots, u_m^*)\), and
\[ B = \begin{bmatrix}
\frac{\partial f_1}{\partial u_1}(x_1^*, \ldots, x_n^*, u_1^*, \ldots, u_m^*) & \cdots & \frac{\partial f_1}{\partial u_m}(x_1^*, \ldots, x_n^*, u_1^*, \ldots, u_m^*) \\
\vdots & & \vdots \\
\frac{\partial f_n}{\partial u_1}(x_1^*, \ldots, x_n^*, u_1^*, \ldots, u_m^*) & \cdots & \frac{\partial f_n}{\partial u_m}(x_1^*, \ldots, x_n^*, u_1^*, \ldots, u_m^*)
\end{bmatrix} \]

\[ B_{(i, j)} = \frac{\partial f_i}{\partial u_j}(x_1^*, \ldots, x_n^*, u_1^*, \ldots, u_m^*). \]