DARBHANGA COLLEGE OF ENGINEERING,DARBHANGA

MODERN CONTROL THEORY
(SEM-VIII:EEE)

AKHIL MOHAMMED KK
Asst.Prof. EEE,DCE Darbhanga

Module 1

Lecture 1

Introduction to State Space Systems

The transfer function modeling approach

The good stuff !

- Transfer functions are defined in the Laplace domain and are a generalization of Fourier transforms
- System: input-output view.
  - Input can be signals of various frequencies
  - Bandwidth of the control system
- Algebraic properties of transfer functions
  - Combine different systems
- Very dominant in the industry !

The drawbacks

- Can be used only for linear systems
  - Can be extended for (some) nonlinear systems, but tricky !
- Limited to single input single output systems
  - Can handle MIMO systems, with some jugglery !
- The internal behaviour of the system is not known – black box !

What is a state space system?

- A state space representation of a system is a mathematical model where the inputs, outputs and states (internal variables) are related by first order differential equations
  - Chemical process or Audio Video signal amplifiers (multiple inputs, states and outputs)
- Can isolate individual system “components” (states) into controllable/uncontrollable or observable/unobservable types
- Really useful techniques from linear algebra for analysis and design

Standard differential equation analysis is now supplemented with (really easy) linear algebraic properties !

General procedure to obtain a state space system

- State space model: a representation of the dynamics of a system in the form of a set of N first order differential equations
  - Convert the Nth order differential equation that governs the dynamics into N first-order differential equations

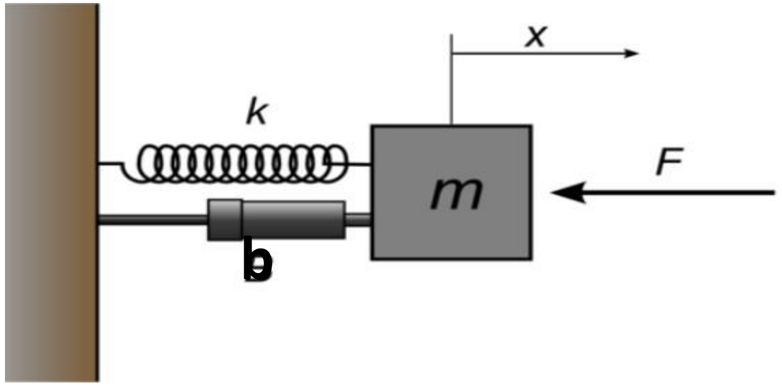
- The number of states is (usually) equal to the order of the differential equation which models your system
  - example: for an electrical system it is usually equal to the number of energy storing elements

- Sounds easy? It is !!

Standard differential equation analysis is now supplemented with (really easy) linear algebraic properties !

A simple example

A mass-spring damper system



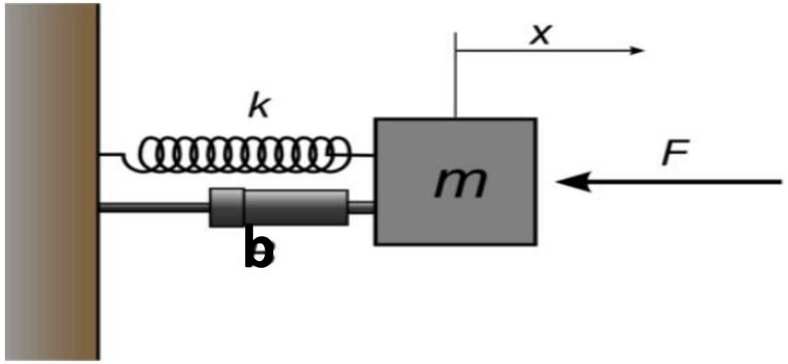
A mass-spring system would Ideally behave like a harmonic oscillator. With the damper in place we would (hope to) see damped oscillatory behaviour.

Let p denote the position of the mass in the above mass spring damper (MSD) system. From the force balance equation we know that the dynamics of the MSD is given by a second order differential equation

$$m\frac{d^2x}{dt^2}+b\frac{dx}{dt}\;kx=F$$

where m is the mass (kg), b is the damping coefficient (Ns/m) and k is the spring constant(N/m).
F is the applied force(N).

A simple example



The final state space model will be as below

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{b}{m} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix} F$$

If the measured output of the system is the position, then we have that

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

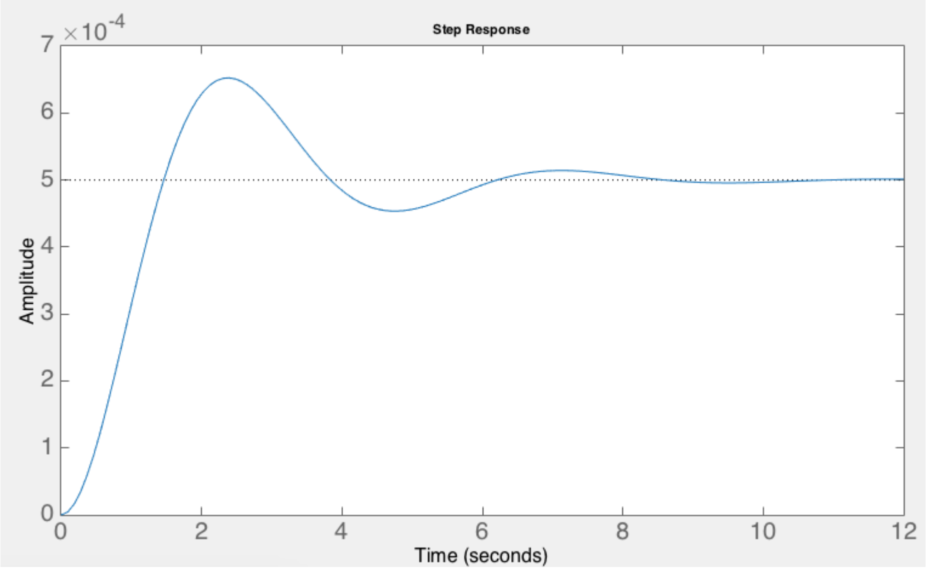
Simulation of the MSD

m = 1000 kg
k = 2000 Nm^{-1}
b = 1000 N / ms^{-1}

$$A = \begin{bmatrix} 0 & 1 \\ -2 & -1 \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ 0.001 \end{bmatrix}$$

MATLAB code

```
M=1000; k=2000; B=1000;
A = [0 1;-k/M -B/M]
B=[0 1/M]'
sys=ss(A,B,[1 0],0);
step(sys)
```



We have a stable damped step response, with a small overshoot.

Can we guess the pole locations from this?

Simulation of the MSD

m = 1000 kg
k = 2000 Nm^{-1}
b = 1000 N / ms^{-1}

$$A = \begin{bmatrix} 0 & 1 \\ -2 & -1 \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ 0.001 \end{bmatrix}$$

We observed a damped stable response in the previous slide. Can we guess the pole locations from this? Lets see…

A is a symmetric matrix and a natural question is "what are the eigen values of A"? We will obtain the following eigenvalues

```
-0.5000 + 1.3229i
-0.5000 - 1.3229i
```

Interestingly the poles computed from the transfer function are also the same !

We shall see in a couple of slides that the eigenvalues of the A matrix are exactly equal to the poles of the corresponding transfer function

What is a state?

x(t) is called the state of the system of time t because

- The future output of the system depends only on the current state and input

- The future output depends on the past input only through the state

- The state summarizes the effects of all past inputs on future outputs…
  - In other words the state acts like the *memory of the system*

Example: Rechargeable mobile phone – the state is the current state of charge of the battery. If you know that state, then you do not need to know how that level of charge was achieved (assuming a ideal battery) to predict the future working of the phone

- But to consider all nonlinear effects, you might also need to know how many cycles the battery has gone through

General Form of a linear time invariant state space model

$$\dot{\mathbf{x}}(t) = A\mathbf{x}(t) + B\mathbf{u}(t)$$
$$\mathbf{y}(t) = C\mathbf{x}(t) + D\mathbf{u}(t)$$

- $t \in \mathbb{R}$ denotes time
- $\mathbf{x}(t) \in \mathbb{R}^n$ is the state (vector)
- $\mathbf{u}(t) \in \mathbb{R}^m$ is the input or control
- $\mathbf{y}(t) \in \mathbb{R}^p$ is the output

- $A(t) \in \mathbb{R}^{n\times n}$ is the dynamics matrix
- $B(t) \in \mathbb{R}^{n\times m}$ is the input matrix
- $C(t) \in \mathbb{R}^{p\times n}$ is the output or sensor matrix
- $D(t) \in \mathbb{R}^{p\times m}$ is the feedthrough matrix

Obtaining state space equations from differential equations

consider an n[th] order linear plant model described by the differential equation

$$\frac{d^n y}{dt^n} + a_{n-1}\frac{d^{n-1}y}{dt^{n-1}} + \cdots + a_1\frac{dy}{dt} + a_0 y = u(t)$$

We can define a useful set of state variables as

$$x_1 = y, \quad x_2 = \dot{y}, \quad x_3 = \ddot{y}, \ldots, x_n = \frac{d^{n-1}y}{dt^{n-1}}$$

Taking derivatives of the first $n-1$ state variables, we have

$$\dot{x}_1 = x_2, \quad \dot{x}_2 = x_3, \quad \ldots, \quad \dot{x}_{n-1} = x_n$$

Finally:

$$\dot{x}_n = -a_0 x_1 - a_1 x_2 - \cdots - a_{n-1}x_n + u(t)$$

How does this look in matrix form?

Obtaining state space equations from differential equations

In matrix form, this looks like

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \vdots \\ \dot{x}_{n-1} \\ \dot{x}_n \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -a_0 & -a_1 & -a_2 & \cdots & -a_{n-1} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{n-1} \\ x_n \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} u(t)$$

If we are measuring only one state, then we have

$$y = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \end{bmatrix} \vec{x}$$

General Form of a linear time invariant state space model

$$
\begin{aligned}
\dot{\mathbf{x}}(t) &= A\mathbf{x}(t) + B\mathbf{u}(t) \\
\mathbf{y}(t) &= C\mathbf{x}(t) + D\mathbf{u}(t)
\end{aligned}
$$