MODELING & SIMULATIONS

TRANSFER FUNCTIONS AND STATE SPACE MODELING

AALBORG UNIVERSITY
DENMARK

LAPLACE TRANSFORMS RECAP

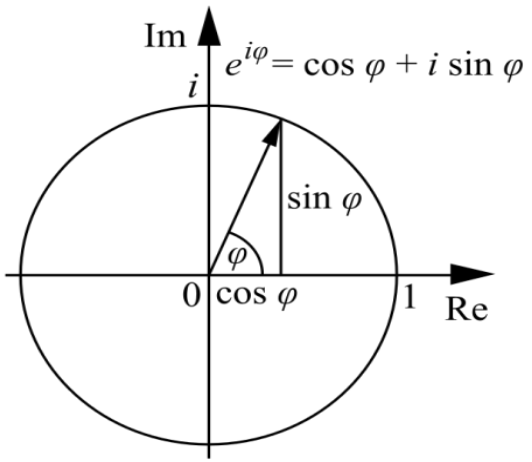
Recap of Euler's Formula

$$e^{ix} = \cos(x) + jsin(x)$$

For any real number x, where i is the imaginary unit and e is the base of the natural logarithm:

$$e = \lim_{n\to\infty}\left(1 + \frac{1}{n}\right)^n$$

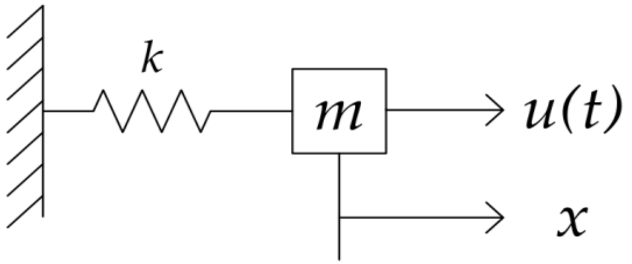
$$\ln(e) = 1, \qquad as:\ e^1 = e, \qquad and\ \ln(1) = 0, \qquad as: e^0 = 1$$



Differential Equations and Laplace

Differential equations, *example with Newton's 2nd law* $F(t) = m\dot{v}(t) = ma(t)$ *and a spring mass system*:

$$u(t) = m\ddot{x}(t) + kx(t)$$



Where $x(t) = f(\ddot{x}(t))$, and its solution is either e or sin **(show on blackboard)**

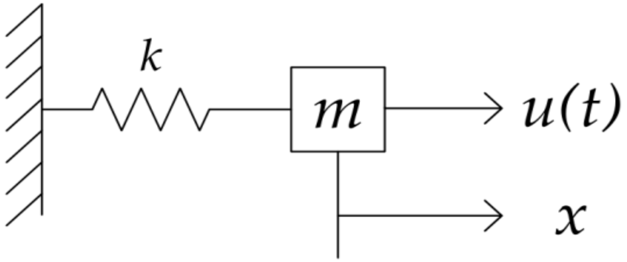
This is in time domain, but for reason which will become clear later in the course we wish to focus on frequency domain

We go to frequency domain using the Laplace transform

Laplace Transform

Differential equations, *example with Newton's 2nd law* $F(t) = m\dot{v}(t) = ma(t)$:

$$u(t) = m\ddot{x}(t) + kx(t)$$



Where $x(t) = f(\ddot{x}(t))$, and its solution is either e or sin **(show on blackboard)**

With **Laplace** we wish to map the **time domain** into the **frequency domain**,
Using **Fourier-Transform we get:**

$$Time - domain \overset{F.T.}{\Longrightarrow} Frequency - domain$$

i.e.

$$X(\omega) = \int_{-\infty}^{\infty} x(t)e^{-j\omega t}dt\,, \qquad e^{-j\omega t} = sinusoid$$

Laplace Transform

$$Time - domain \overset{F.T.}{\Longrightarrow} Frequency - domain$$

$$X(\omega) = \int_{-\infty}^{\infty} x(t) e^{-j\omega t} dt\,, \qquad e^{-j\omega t} = sinusoid$$

Example, Take F.T. of a sinusoid:

$$X(\sigma, \omega) = \int_{-\infty}^{\infty} [x(t) \cdot e^{-\sigma t}] e^{-j\omega t} dt$$

If we combine the two exponential terms into a single term

$$e^{-\sigma t} \cdot e^{-j\omega t} = e^{-\sigma - j\omega t}, \qquad \boldsymbol{s = \sigma + j\omega}$$

Where,

σ is a real portion ‘exponential responses’, and

$j\omega$ is an imaginary term ‘sinusoidal responses’

Laplace Transform

$$Time - domain \overset{F.T.}{\Longrightarrow} Frequency - domain$$

$$X(\omega) = \int_{-\infty}^{\infty} x(t) e^{-j\omega t} dt\,, \qquad e^{-j\omega \mathrm{t}} = sinusoid$$

Now Substitute ***ω*** **with s** in the F.T., we get the **Laplace transform**

$$X(s) = \int_{-\infty}^{\infty} x(t) e^{-st} dt$$

i.e.

$$Time - domain \overset{\mathcal{L}}{\Leftrightarrow} s - domain$$

Common Laplace Transforms

Table of Laplace Transforms

Number	$F(s)$	$f(t), t \geq 0$
1	1	$\delta(t)$
2	$1/s$	$1(t)$
3	$1/s^2$	t
4	$2!/s^3$	t^2
5	$3!/s^4$	t^3
6	$m!/s^{m+1}$	t^m
7	$\frac{1}{s+a}$	e^{-at}
8	$\frac{1}{(s+a)^2}$	te^{-at}
9	$\frac{1}{(s+a)^3}$	$\frac{1}{2!}t^2e^{-at}$
10	$\frac{1}{(s+a)^m}$	$\frac{1}{(m-1)!}t^{m-1}e^{-at}$
11	$\frac{a}{s(s+a)}$	$1-e^{-at}$

Common Laplace Transforms

Table of Laplace Transforms

Number	$F(s)$	$f(t), t \geq 0$
12	$\frac{a}{s^2(s+a)}$	$\frac{1}{a}(at - 1 + e^{-at})$
13	$\frac{b-a}{(s+a)(s+b)}$	$e^{-at} - e^{-bt}$
14	$\frac{s}{(s+a)^2}$	$(1-at)e^{-at}$
15	$\frac{a^2}{s(s+a)^2}$	$1 - e^{-at}(1+at)$
16	$\frac{(b-a)s}{(s+a)(s+b)}$	$be^{-bt} - ae^{-at}$
17	$\frac{a}{s^2+a^2}$	$\sin at$
18	$\frac{s}{s^2+a^2}$	$\cos at$
19	$\frac{s+a}{(s+a)^2+b^2}$	$e^{-at}\cos bt$
20	$\frac{b}{(s+a)^2+b^2}$	$e^{-at}\sin bt$
21	$\frac{a^2+b^2}{s[(s+a)^2+b^2]}$	$1 - e^{-at}\left(\cos bt + \frac{a}{b}\sin bt\right)$

Computer tools for Laplace and other operations

- For more complex equations computer software can be used as an aid:
  - Matlab
  - Maple
  - Mathcad
  - TI calculators
  - …
- We'll take use of Matlab and Maple in this course for:
  - Laplace transforms
  - Linearization
  - Equation manipulations
  - And other algebraic uses

Example of Laplace transform

Find the Laplace Transform of:

$$f(t) = 1 + 2\sin(\omega t)$$

We use the tables to find the most common Laplace transforms where:

$$\mathcal{L}\{1(t)\} = \frac{1}{s}, \qquad \mathcal{L}\{k \cdot \sin(at)\} = \frac{k \cdot a}{s^2 + a^2}$$

And thus we have:

$$\mathcal{L}\{f(t)\} = \frac{1}{s} + 2 \cdot \frac{\omega}{s^2 + \omega^2}$$

Laplace Transform – Recap

Fourier Transform (FT): $X(\omega) = \int_{-\infty}^{\infty} x(t)e^{-j\omega t}dt$

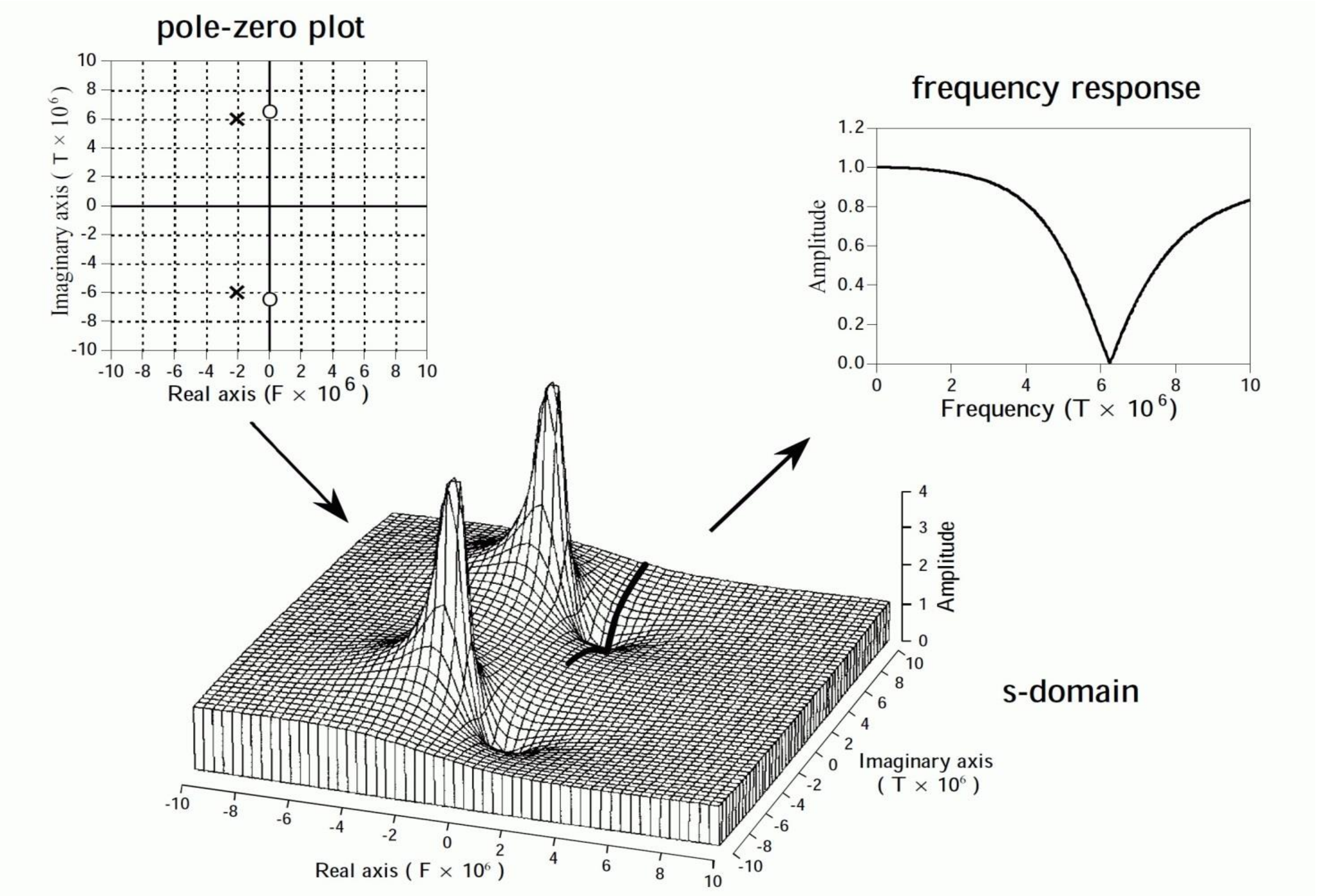
Laplace Transform (LT): $X(s) = \int_{-\infty}^{\infty} x(t)e^{-st}dt$

The difference between FT and LT is:

FT is represents only the simple frequencies, i.e. ω

LT represents also the complex frequencies, i.e. $s = \sigma + j\omega$

Graphical Visualization of P-Z, $\mathcal{L}$ and FT



Sum up of the Laplace Transform

- Laplace is a extension of the Fourier transform
- Adds the extra dimension of σ
- Differential equations in time domain $\rightarrow$ algebraic equations in s domain
- Derivative and integrals in time domain $\rightarrow$ powers in s domain

TRANSFER FUNCTIONS

Transfer Function

- The function $H(s)$ is the transfer gain from $U(s)\ to\ Y(s)$ or the (input to output), thus:

$$\frac{Y(s)}{U(s)} = H(s)$$

Assuming that all initial conditions of $H(s)$ are 0,

The transfer function $H(s)$ is the Laplace transform of the unit impulse response $h(s)$

Transfer Function

$$\frac{Y(s)}{U(s)} = H(s)$$

If the *unit impulse* $\delta(t)$ is the input $u(t)$, than $y(t)$ is the *unit impulse response:*

Then we have that $\mathcal{L}\big(u(t)\big) = 1$, and $\mathcal{L}\big(y(t)\big) = H(s)$, as:

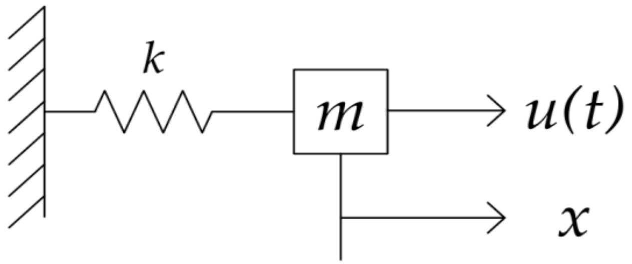
$$Y(s) = H(s)$$

i.e. The transfer function $H(s)$ is the Laplace transform of the unit impulse response $h(s)$

Example: Differential equation to Transfer function

Differential equation of a harmonic oscillator from slide 22, i.e. equation of motion:

$$u(t) = m\ddot{x}(t) + kx(t)$$



The Laplace transform of the differential equation:

$$ms^2X(s) + kX(s) = U(s)$$

Set up in transfer fraction construction from previous slide, output over input, $\frac{Y(s)}{U(s)} = H(s)$:

$$\frac{X(s)}{U(s)} = \frac{1}{ms^2 + k}$$

Example Finish on Black Board

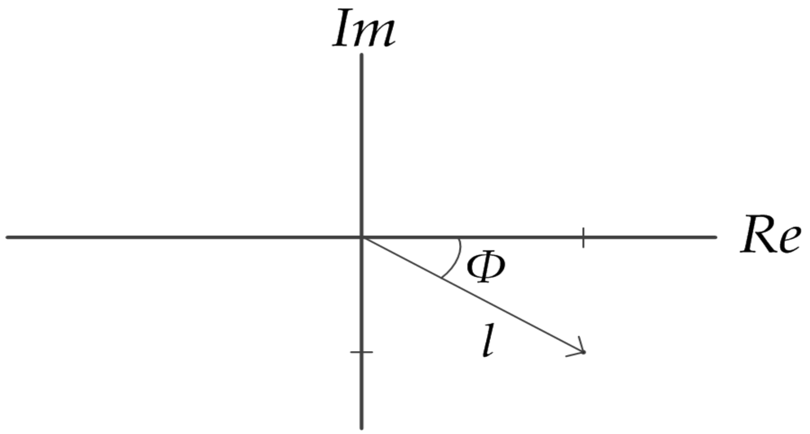
- Find poles and zeros
  - No poles
  - 1 zero
- Check stability
  - System is stable

Example: Frequency response

- Frequency analysis (of the previous system):
  - Calculate the steady state frequency response, i.e. $s \ = j\omega$
  - **Such that $\boldsymbol{H(j\omega) = real + imag \cdot j}$**
  - *Example with mass spring system:*

$$\frac{1}{mj\omega^2 + k}$$

- Visualize the whole frequency spectrum, wrt:
  - **Magnitude**: $M(\omega) = |H(j\omega)| = \sqrt{real^2 + imag^2}$
  - **Phase**: $\phi(\omega) = \arg(H(j\omega) = atan2(imag, real)$



Poles and zeros

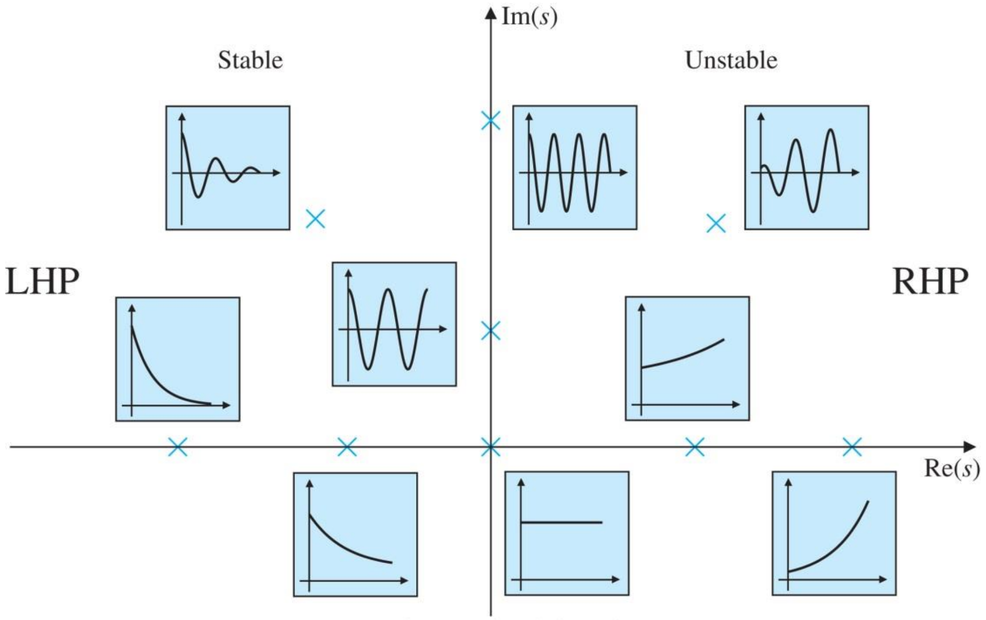
- When a transfer function is obtained, the zeros and poles can easily be identified
- The poles describe the **stability** of the system $\quad p_{i+1} = \sigma_i \pm j\omega_i$
  - When **all** the poles are located on the left half-plane of the real axis the system is stable:

$$H(s) = \frac{b_m s^m + b_{m-1} s^{m-1} + \ldots + b_1 s + b_0}{a_n s^n + a_{n-1} s^{n-1} + \ldots + a_1 s + a_0}$$

$$H(s) = \frac{N(s)}{D(s)} = K \frac{(s - z_1)(s - z_2)\ldots(s - z_{m-1})(s - z_m)}{(s - p_1)(s - p_2)\ldots(s - p_{n-1})(s - p_n)}$$

$$y_h(t) = \sum_{i=1}^{n} C_i e^{p_i t}$$

Location of poles



Franklin, Gene F., et al. *Feedback control of dynamic systems*. Vol. 3. Reading, MA: Addison-Wesley, 1994.

First Order Systems

- Linear time-invariant (LTI) systems can be expressed as a combination of first order transfer functions (as shown with partial fraction decomposition):

$$H(s) = \frac{K}{\tau s + 1} = \frac{K/\tau}{s + 1/\tau} = \frac{K/\tau}{s - \left(-\frac{1}{\tau}\right)}$$

$$h(t) = \frac{K}{\tau} e^{-t/\tau}$$

- DC gain: $H(0)$ - Equal **K**
- Time constant: $\boldsymbol{\tau}$ - Convergence speed of the system

Second Order Systems

- Damping factor
- ω_n is the natural frequency
- ω_R is the resonance frequency
- $\omega_R = \omega_n \sqrt{1-\zeta^2}$

$$G(s) = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2}$$

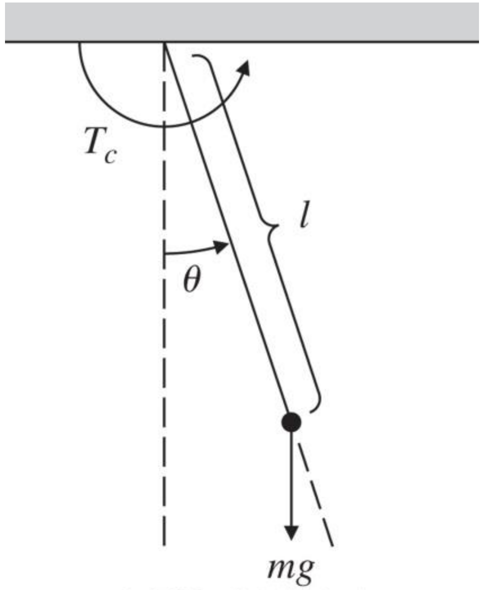
Case	Description	Roots
$\xi = 0$	Undampted	$s_{1,2} = \pm j\omega$
$\xi < 1$	Underdampted	$s_{1,2} = -\sigma \pm j\omega$
$\xi = 1$	Critically damped	$s_1 = s_2 = -\sigma$
$\xi > 1$	Overdamped	$s_1 = -\sigma_1,\ s_2 = -\sigma_2$

Transfer Function Example

- Model of a pendulum is

$$\ddot{\theta}(t) + \frac{g}{l} sin\theta(t) = \frac{T_c(t)}{ml^2}$$

Convert this into a transfer function of the form $\frac{out(s)}{In(s)} = G(s)$



Summary of

SYSTEM STABILITY

AALBORG UNIVERSITY
DENMARK

System stability

- What characterizes a stable system?

Stability theory addresses the stability of solutions of differential equations and of trajectories of dynamical systems under small perturbations of initial conditions.

  - If a system drives towards (approaches) a finite value for $t \rightarrow \infty$, the system can be considered stable.

- Example of a stable system:

$$\mathcal{L}^{-1}\{H(s)\} = \mathcal{L}^{-1}\left\{\frac{Y(s)}{U(s)}\right\} = \mathcal{L}^{-1}\left\{\frac{A}{s+a}\right\} = A\mathcal{L}^{-1}\left\{\frac{1}{s+a}\right\} = Ae^{-at}$$

Asymptotic stable

- Example of an unstable system:

$$\mathcal{L}^{-1}\{H(s)\} = \mathcal{L}^{-1}\left\{\frac{Y(s)}{U(s)}\right\} = \mathcal{L}^{-1}\left\{\frac{A}{s-a}\right\} A\mathcal{L}^{-1}\left\{\frac{1}{s-a}\right\} = Ae^{at}$$

Will increase as t increases

System stability

- Can be found directly on the transfer function
  - Independent of the zeros
  - Only **depends** on the **poles**
  - For **any** unstable pole, the entire system is unstable

STATE SPACE MODELS

State-space Models

- In the state-space method the a dynamic system is described by a set of first-order differential equations in the vector-valued state, and the solution is visualized as a trajectory of this state vector in space.
- General class of ODE models

- Advantages:
  - To study more general models
  - To deal with multiple input and multiple output systems
    - Without having extra TF's.
  - To connect internal and external descriptions

State-space definition (non-linear, time invariant)

State-space equations

State equation:

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

Output equation:

$$\boldsymbol{y}(t) = \boldsymbol{h}(\boldsymbol{x}(t), \boldsymbol{u}(t))$$

Explanation

- $\boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t))$ is a nonlinear vector function of state and input
- $\boldsymbol{h}(\boldsymbol{x}(t), \boldsymbol{u}(t))$ is a nonlinear vector function of state and input

- $\boldsymbol{x}(t)$ is the state
- $\boldsymbol{u}(t)$ is the input

State-space definition (linear, time invariant)

State-space equations

State equation:

$$\frac{d\boldsymbol{x}(t)}{dt} = A \cdot \boldsymbol{x}(t) + B \cdot \boldsymbol{u}(t)$$

Output equation:

$$\boldsymbol{y}(t) = C \cdot \boldsymbol{x}(t) + D \cdot \boldsymbol{u}(t)$$

Explanation

- **x**(t) is an $n \times 1$ vector representing the state (e.g., position and velocity variables in mechanical systems)
- **u**(t) is a $m \times 1$ vector representing the input
- **y**(t) is a $p \times 1$ vector representing the output

- The matrices $A(n \times n)$, $B(n \times m)$, and $\mathrm{C}(p \times n)$, determine the relationships between the state and input and output variables. These are constant matrices.

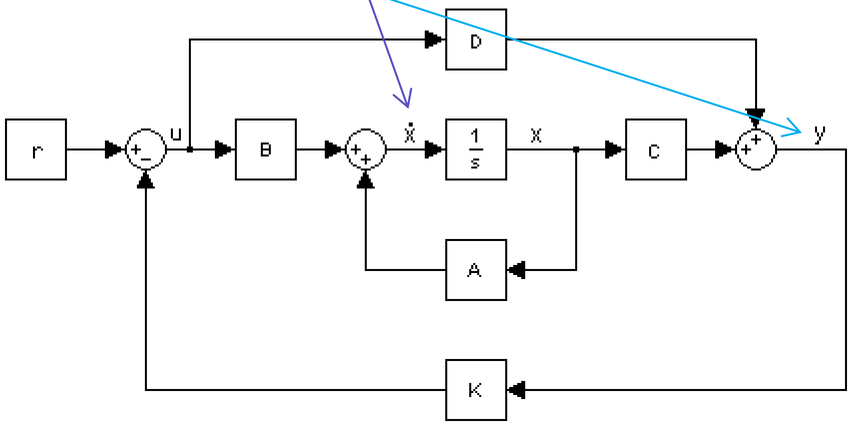
AALBORG UNIVERSITY
DENMARK

State Space Block Diagram

The state space representation has the following block diagram.

$$\frac{d\boldsymbol{x}(t)}{dt} = A \cdot \boldsymbol{x}(t) + B \cdot \mathbf{u}(t)$$

$$\boldsymbol{y}(t) = C \cdot \boldsymbol{x}(t) + D \cdot \mathbf{u}(t)$$



Non-linear models

- All systems are nonlinear (Large signals)
  - I.e. its dynamics depends on the process variables in a non linear fashion
- Prefer linear models for control design
- Use small signals to linearize non-linear systems

Linearization of Non-linear models

- All systems are nonlinear (Large signals)
  - I.e. its dynamics depends on the process variables in a non linear fashion
- Prefer linear models for control design
  - Control + analysis is easier with linear models
- Use small signals to linearize non-linear systems
- For control linear systems OK, control keeps system close to equilibrium

- Three ways of linearization:
  - **Linearization of the model (focus of this course)**
  - Linearization by feedback
  - Inverse linearization

Stability of nonlinear systems

- If the linearized system around a specific equilibrium point is strictly stable, (roots in LHP)
- The non-linear system is stable around the equilibrium point of the linearization
- If it has at least 1 root in RHP, then the nonlinear system is unstable around the equilibrium point

Linearization: Small Signal Analysis

Linear:

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

Nonlinear:

$$\dot{x} = f(x, u)$$

1st Define equilibrium values: u_0, x_0, such that:

$$\dot{x}_0 = 0 = f(x_o, u_0)$$

And the state and input vectors are:

$$x = x_0 + \delta x, u = u_0 + \delta u$$

Linearization: Small Signal Analysis

The **nonlinear function** $\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u})$, can now be represented with perturbations from the equilibrium points:

$$\dot{x}_0 + \delta\dot{x} \cong f(x_0, u_0) + A\delta x + B\delta u$$

A and B can be computed as following:

$$A = \left[\frac{\delta f}{\delta x}\right]_{x_0,u_0}, \qquad B = \left[\frac{\delta f}{\delta u}\right]_{x_0,u_0}$$

Where A and B are the best linear fits to the nonlinear function,

The linear diff. eq. approximating the dynamics of the state around the equilibrium points (x_0, u_0):

$$\delta\dot{x} = A\delta x + B\delta u$$

Linearization method (state and input)

- Jacobian = $\frac{d\boldsymbol{f}}{d\boldsymbol{x}}$
- Matlab:

`jacobian(f,x)`

$$A = \frac{d\boldsymbol{f}}{d\boldsymbol{x}} = \begin{bmatrix} \frac{df_1}{dx_1} & \cdots & \frac{df_1}{dx_n} \\ \vdots & \ddots & \vdots \\ \frac{df_n}{dx_1} & \cdots & \frac{df_n}{dx_n} \end{bmatrix}_{\begin{subarray}{l} \boldsymbol{x}=\boldsymbol{x}_e \\ \boldsymbol{u}=\boldsymbol{u}_e \end{subarray}}$$

$$\text{B}= \frac{d\boldsymbol{f}}{d\boldsymbol{u}} = \begin{bmatrix} \frac{df_1}{du_1} & \cdots & \frac{df_1}{du_m} \\ \vdots & \ddots & \vdots \\ \frac{df_n}{du_1} & \cdots & \frac{df_n}{du_m} \end{bmatrix}_{\begin{subarray}{l} \boldsymbol{x}=\boldsymbol{x}_e \\ \boldsymbol{u}=\boldsymbol{u}_e \end{subarray}}$$

$$C = \frac{d\boldsymbol{h}}{d\boldsymbol{x}} = \begin{bmatrix} \frac{dh_1}{dx_1} & \cdots & \frac{dh_1}{dx_n} \\ \vdots & \ddots & \vdots \\ \frac{dh_p}{dx_1} & \cdots & \frac{dh_p}{dx_n} \end{bmatrix}_{\begin{subarray}{l} \boldsymbol{x}=\boldsymbol{x}_e \\ \boldsymbol{u}=\boldsymbol{u}_e \end{subarray}}$$

$$\text{D} = \frac{d\boldsymbol{h}}{d\boldsymbol{u}} = \begin{bmatrix} \frac{dh_1}{du_1} & \cdots & \frac{dh_1}{du_p} \\ \vdots & \ddots & \vdots \\ \frac{dh_p}{du_1} & \cdots & \frac{dh_p}{du_m} \end{bmatrix}_{\begin{subarray}{l} \boldsymbol{x}=\boldsymbol{x}_e \\ \boldsymbol{u}=\boldsymbol{u}_e \end{subarray}}$$

Linearization with different variables

Taylor series expansion of a single variable function, presented with $\boldsymbol{x_s}$ as the steady state operation point:

$$f(x) \approx f(x_s) + \frac{f'(x_s)}{1!} \cdot (x - x_s) + \frac{f''(x_s)}{2!} \cdot (x - x_s)^2 + \cdots$$

We truncate after the first order term (linear term) and set the deviation variables (from steady state) equal $x' = x - x_s$

$$f'(x_s) = \left.\frac{\delta\, f(x)}{dx}\right|_{x=x_s}$$

Taylor series expansion with the output and input variable:

$$f(y,u) \cong f(y_s,u_s) + \left.\frac{\delta f}{\delta y}\right|_{y_s,u_s} (y, y_s) + \left.\frac{\delta f}{\delta u}\right|_{y_s,u_s} (u, u_s)$$

Where in steady state we have that $\boldsymbol{f(y_s, u_s)}$**=0,** where, the deviation variables (from steady state) equal $y' = y - y_s \;\; and \;\; u' = u - u_s$. Then we have:

$$\frac{dy'}{dt} = \left.\frac{\delta f}{\delta y}\right|_{y_s,u_s} y' + \left.\frac{\delta f}{\delta u}\right|_{y_s,u_s} u'$$

AALBORG UNIVERSITY
DENMARK

Taylor´s Theorem

Power series

$$\sum_{\{n=0\}}^{\infty} a_n(x - c)^n = a_0 + a_1(x - c)^1 + a_2(x - c)^2 + \cdots$$

Example of a power series, represent the exponential function with a power series

$$e^x = \sum_{\{n=0\}}^{\infty} \frac{x^n}{n!} = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots,$$

Taylor series is a power series with $a_n = \frac{1}{n!}$, where c is the center of the series.

Thus the Taylor series of a function $f(x)$ is:

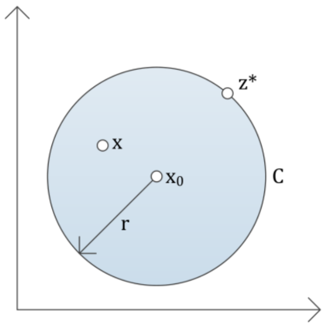
$$f(x) = \sum_{n=1}^{\infty} a_n(x - x_0)^n, \qquad where \qquad a_n = \frac{1}{n!} f^{(n)}(x_0)$$

Taylor´s Theorem

$$f(x) = \sum_{n=1}^{\infty} a_n (x - x_0)^n, \qquad where \qquad a_n = \frac{1}{n!} f^{(n)}(x_0)$$

Can be represented by the derivative of an analytic function, $f^{(n)}(x_0) = \frac{n!}{2\pi i} \oint_C \frac{f(x)}{(x - x_0)^{n+1}} dz$ (Cauchy's formula):

$$a_n = \frac{1}{2\pi i} \oint_C \frac{f(x^*)}{(x^* - x_0)^{n+1}} dz^*$$



Linearization example with Taylor expansion: $\boldsymbol{sin}(\boldsymbol{x})$

We have the Taylor series

$$f(a)+\frac{f'(a)}{1!}\cdot(x-a)+\frac{f''(a)}{2!}\cdot(x-a)^2+\cdots$$

Inserting the function $\mathrm{f(x)=sin(x)}$

$$sin(a)+\frac{cos(a)}{1!}\cdot(x-a)+\frac{-sin(a)}{2!}\cdot(x-a)^2+\cdots$$

Set $x=0=a$, i.e. the angle is 0, and let $f(x)=sin(x)$,

$$sin(0)+\frac{cos(0)}{1!}\cdot(x-0)+\frac{-sin(0)}{2!}\cdot(x-0)^2+\cdots$$

and truncate after the first order term (linear term) we get:

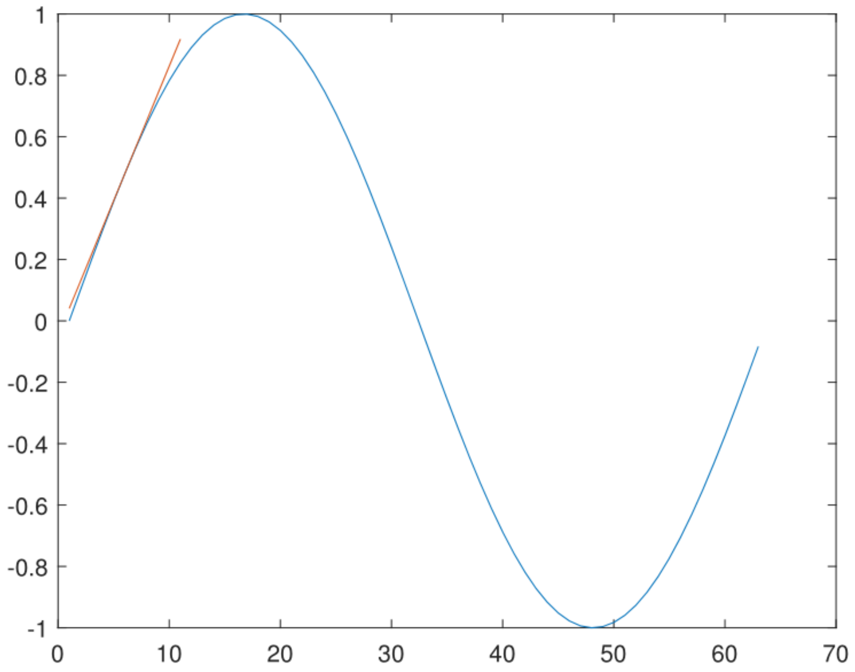
$$\sin(x)\cong 0+1+x=x$$

AALBORG UNIVERSITY
DENMARK

Linearization drawbacks

- Linearization of sin(x) around a operating point of 0.5.

Matlab & Maple example



AALBORG UNIVERSITY
DENMARK

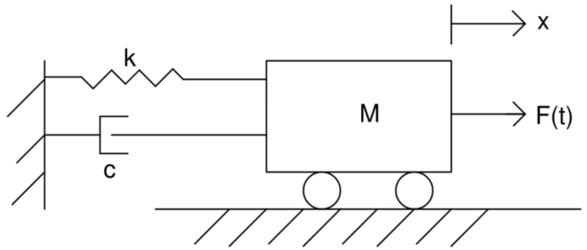
Linearization in Matlab

- Linearization of functions in Matlab:
  - `[A,B,C,D]=linmod('SYS',X,U)`
  - `[A,B,C,D]=linmod2('SYS',X,U)`

Spring mass damper system example

Example: State Space from Diff. Eq.

Convert the N^th order differential equation that governs the dynamics into N first-order differential equations



We start with a spring mass damper system

$$m\ddot{x} + c\dot{x} + kx = F$$

We then redefine the equation of motion into the state variable form where the states can be defined as:

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} x \\ \dot{x} \end{bmatrix}, \text{ and } \dot{x_2} = \ddot{x} = \frac{F - c\dot{x} + kx}{m} = \frac{F - cx_2 + kx_1}{m}$$

We construct the state space model, where u is the force F

$$\dot{x} = Ax + Bu \rightarrow \begin{bmatrix} \dot{x} \\ \ddot{x} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -k/m & -c/m \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \end{bmatrix} + \begin{bmatrix} 0 \\ 1/m \end{bmatrix} u$$

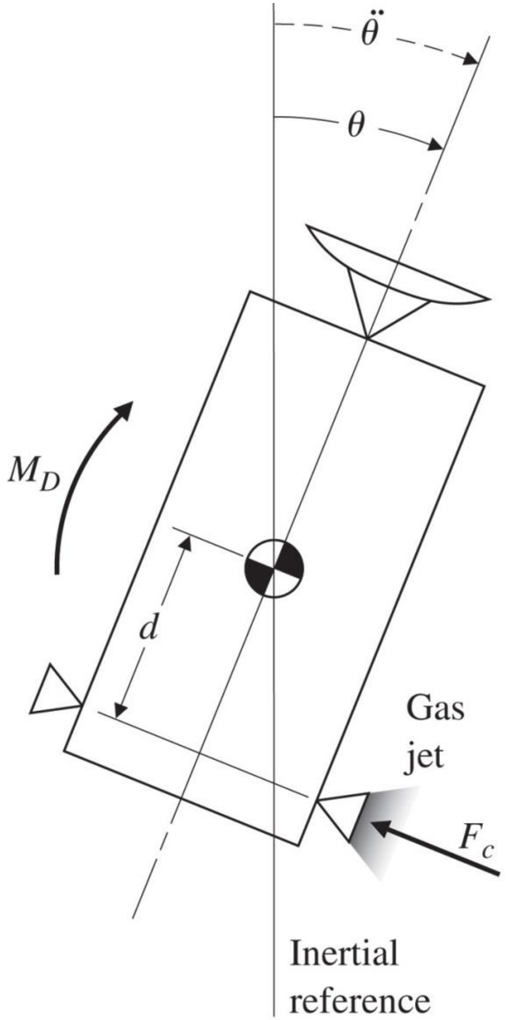
If the output of the system y is the position x, then we have:

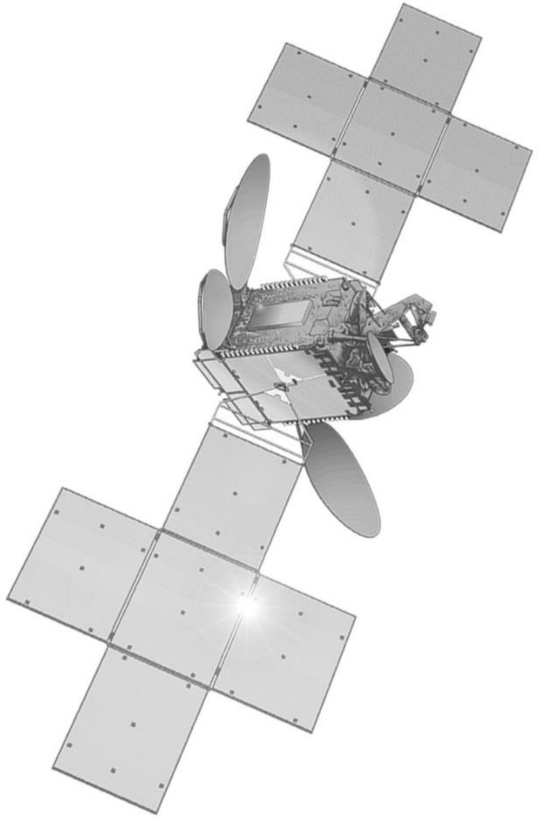
$$y = Cx \rightarrow y = [1\ 0] \begin{bmatrix} x \\ \dot{x} \end{bmatrix}$$

Satellite Model Example

AALBORG UNIVERSITY
DENMARK

Satellite model



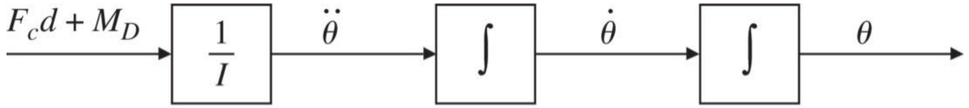


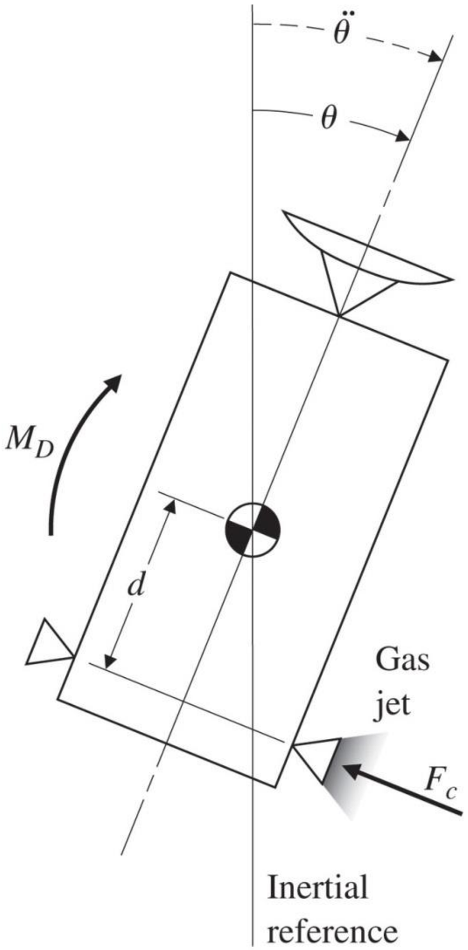
- We wish to model the angular motion of the satellite around its **center of mass** with the **angular position $\boldsymbol{\Theta}$**
- Then the motion is described by the **angular acceleration $\boldsymbol{\ddot{\Theta}}$**

The **angular acceleration** can be modeled by the following diff. equation:

$$I\ddot{\Theta} = F_c d + M_D$$

And can be represented by the following block diagram:





AALBORG UNIVERSITY DENMARK

Conversion into State Space

We have the 2nd order diff. equation

$$I\ddot{\Theta} = F_c d + M_D$$

We regard $M_D = 0$ *for simplicity*

We can define the State Variables $x = \left[\ddot{\Theta}\ \dot{\Theta}\right]^T$

Rocket thrust as the input: $u = F_c$

Satellite attitude as the output: $y = \Theta$

Where:

$$\ddot{\Theta} = \frac{d}{I} F_c$$

We can arrange the states following the state-variable form:

$$\dot{x} = Ax + Bu$$
$$y = Cx + Du$$

Conversion into State Space

$$I\ddot{\Theta} = F_c d + M_D$$
$$x = [\ddot{\Theta}\ \dot{\Theta}]^T$$
$$u = F_c$$
$$y = \Theta$$

$$\dot{x} = Ax + Bu$$
$$y = Cx + Du$$

Which becomes:

$$\begin{bmatrix} \dot{\Theta} \\ \ddot{\Theta} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \Theta \\ \dot{\Theta} \end{bmatrix} + \begin{bmatrix} 0 \\ d/I \end{bmatrix} F_c$$

$$y = [1\ 0] \begin{bmatrix} \Theta \\ \dot{\Theta} \end{bmatrix}$$

Giving us the state variables: $A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix},\ B = \begin{bmatrix} 0 \\ d/I \end{bmatrix},\ C = [1\ 0],\ D = 0$

Simulation in Matlab

$$\begin{bmatrix}\dot{\Theta}\\ \ddot{\Theta}\end{bmatrix} = \begin{bmatrix}0 & 1\\ 0 & 0\end{bmatrix}\begin{bmatrix}\Theta\\ \dot{\Theta}\end{bmatrix} + \begin{bmatrix}0\\ d/I\end{bmatrix}F_c$$
$$y = [1\ 0]\begin{bmatrix}\Theta\\ \dot{\Theta}\end{bmatrix}$$

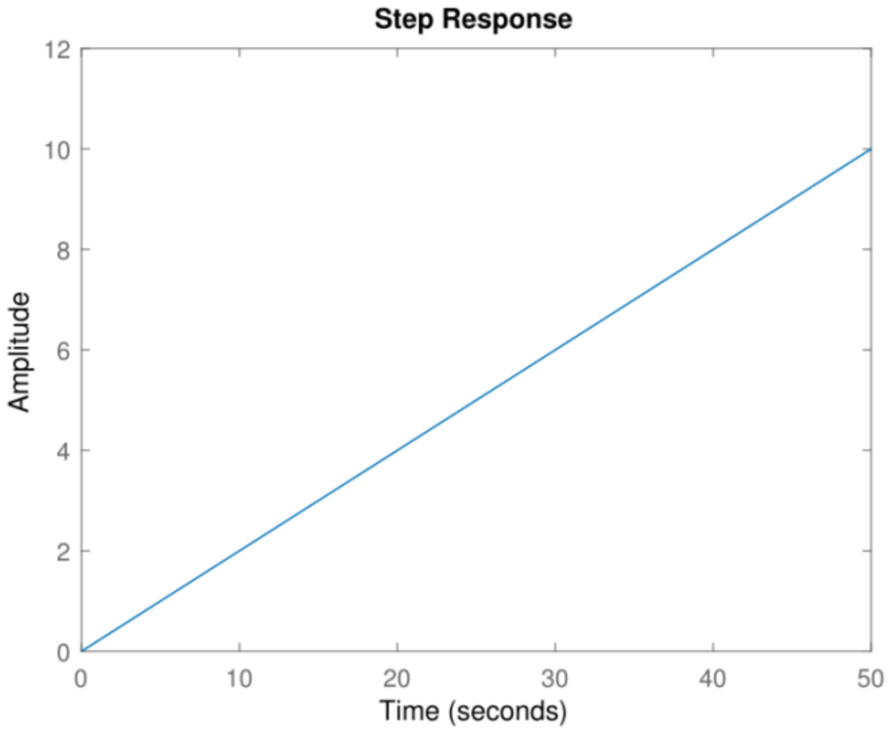
We find that the parameters of the model are: $d = 0.5m, I = m \cdot r^2 = 2.5$

And insert into our State Space model : $A = \begin{bmatrix}0 & 1\\ 0 & 0\end{bmatrix},\ B = \begin{bmatrix}0\\ 0.5/2.5\end{bmatrix},\ C = [1\ 0],\ D = 0$

Simulation in Matlab

Step response of the linear model:

```
step(sys);
```

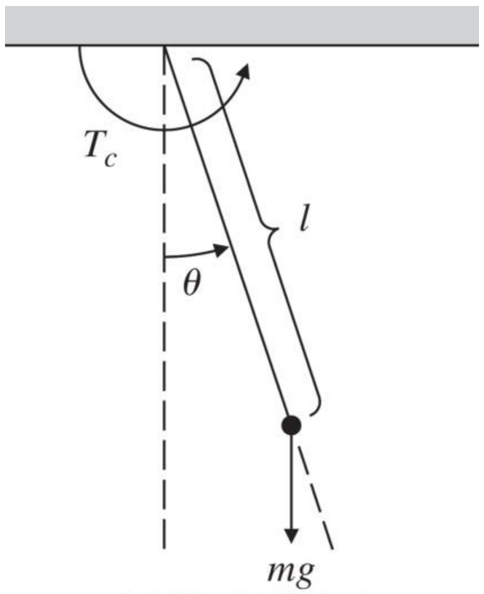


BLACK BOARD EXAMPLE

Example on the black board with linearization and State space construction

- Using the Pendulum as an example with the following model:

$$\ddot{\theta} + \frac{g}{l} sin\theta = \frac{T_c}{ml^2}$$



How to get SS representations

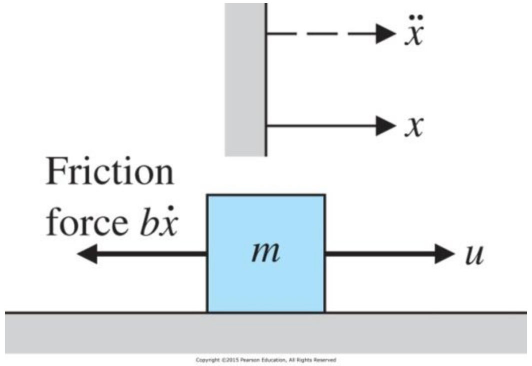
- Via modelling techniques
  - Differential equations (+ potential linearization)
  - `sys = ss(A,B,C,D)`
- Via transfer functions
  - MATLAB: `[A,B,C,D] = tf2ss(NUM,DEN)`
- Via other SS descriptions
  - MATLAB: `sysT = ss2ss(sys,T)`
- Via system identification
  - MATLAB: `ident`

Next Lecture

STATE SPACE MODEL ANALYSIS

AALBORG UNIVERSITY
DENMARK

Exercise



1. Given the following Linear equation

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

   a) Create a transfer function model
   b) Simulate the model
      a) Let m = 1000 and b =100, and set u as 100
   c) Analyze the poles of the system
      a) Try with different parameters
2. Create a state space model
   1. Simulate the model
      1. Let m = 1000 and b =100, and set u as 100
   2. Analyze the poles of the system
3. Linearize sin(x), x^2, cos(x)
   1. Set $x_0 = 0$
   2. Change $x_0 = 0.1, 0.5, 10$
   3. Plot the different functions alongside the non-liner functions