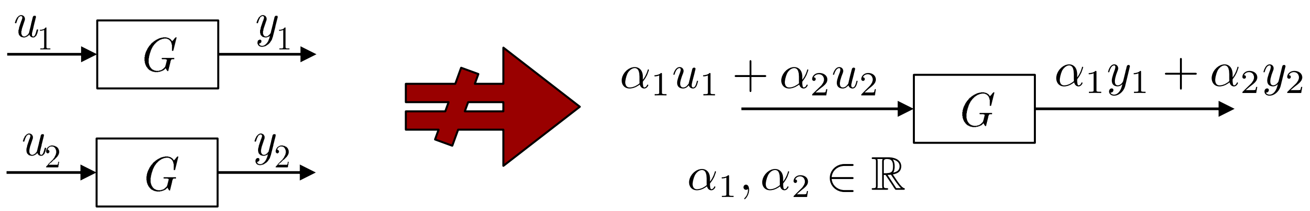
Ch3: Modeling of Dynamic Systems – Part 2

Contents:

Nonlinear Systems

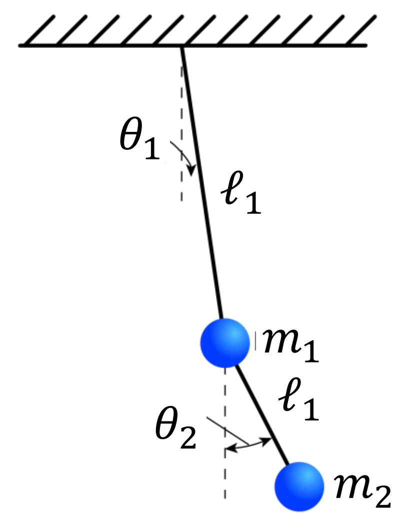
Nonlinear Systems

A system is nonlinear if the **principle of superposition** does **not** apply.



For example, in the **dynamic equations of robots** usually the nonlinear terms sin , cos , and squares of velocities appears.

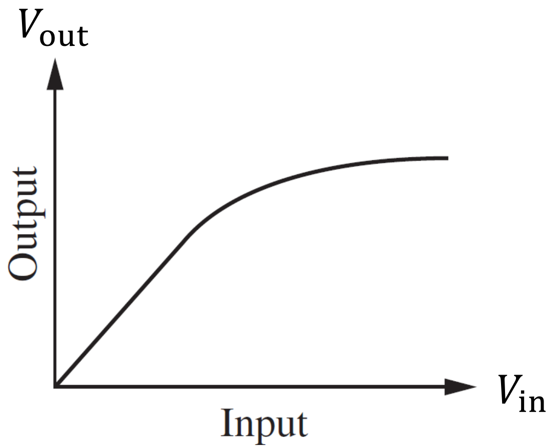
Double-Pendulum:



$$(m_1 + m_2)\ell_1\ddot{\theta}_1 + m_2\ell_2\ddot{\theta}_2\cos(\theta_1 - \theta_2) + m_2\ell_2\dot{\theta}_2^2\sin(\theta_1 - \theta_2) + g(m_1 + m_2)\sin\theta_1 = 0$$

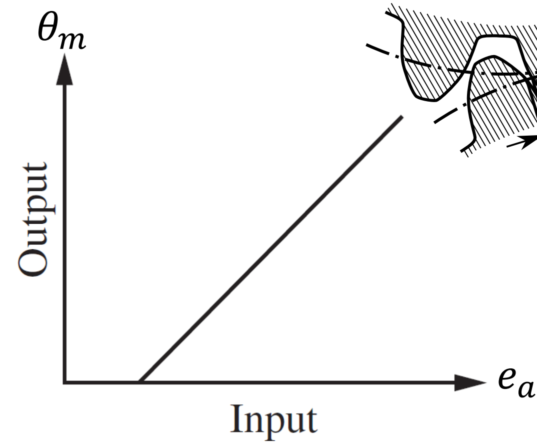
$$m_2\ell_2\ddot{\theta}_2 + m_2\ell_1\ddot{\theta}_1\cos(\theta_1 - \theta_2) - m_2\ell_1\dot{\theta}_1^2\sin(\theta_1 - \theta_2) + m_2 g\sin\theta_2 = 0$$

Examples of Physical Nonlinearities



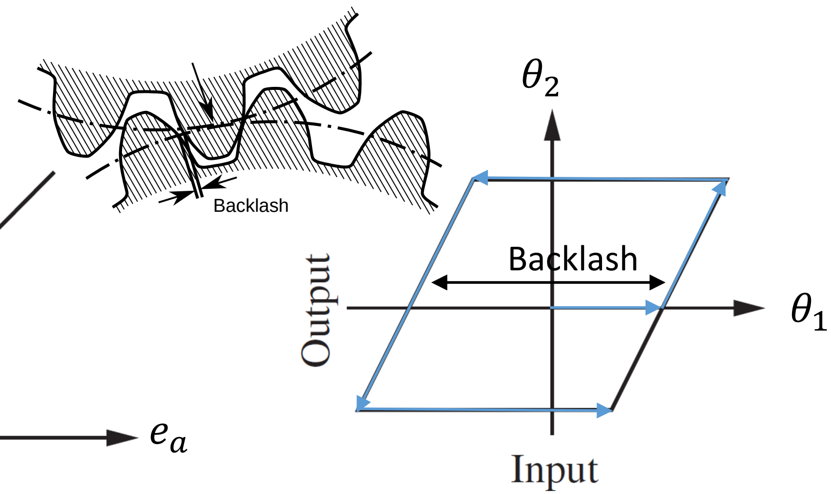
Amplifier Saturation

An electronic amplifier is linear over a specific range but exhibits the nonlinearity called saturation at high input voltages.



Motor Dead Zone

A motor that does not respond at very low input voltages due to frictional forces exhibits a nonlinearity called dead zone.



Backlash in Gears

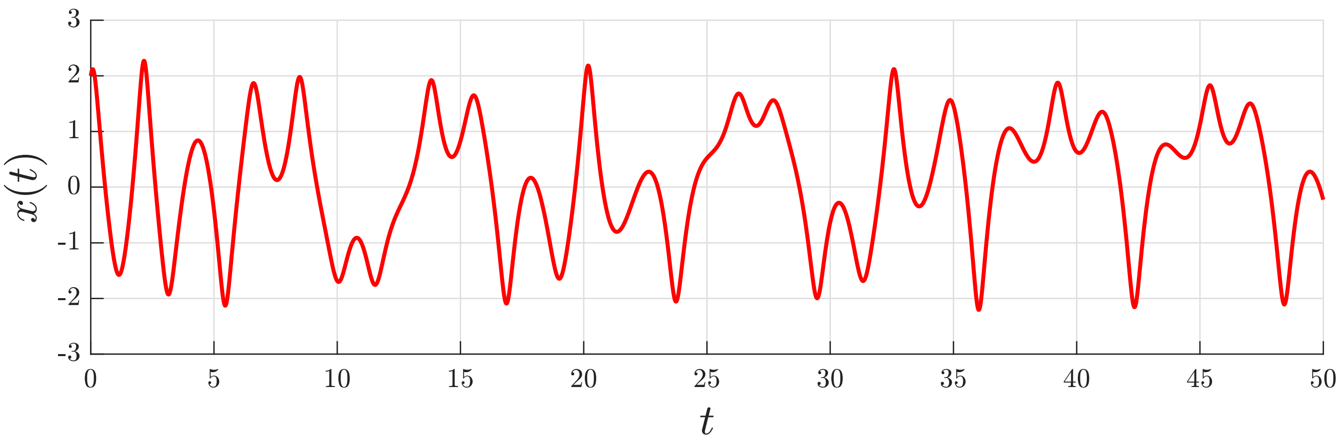
Gears that do not fit tightly exhibit a nonlinearity called backlash which the input moves over a small range without the output responding.

- Nonlinearities can be classified in terms of their mathematical properties, as **continuous** and **discontinuous**. Because discontinuous nonlinearities cannot be locally approximated by linear functions, they are also called **hard nonlinearities** (e.g., backlash, hysteresis, or stiction).

Nonlinear System Behavior: Chaos

- In the steady state, **sinusoidal inputs** to a stable LTI system generate a sinusoidal outputs of the same frequency (but different in amplitude and phase angle from the input). By contrast, the output of a nonlinear system may display sinusoidal, periodic, or chaotic behaviors.

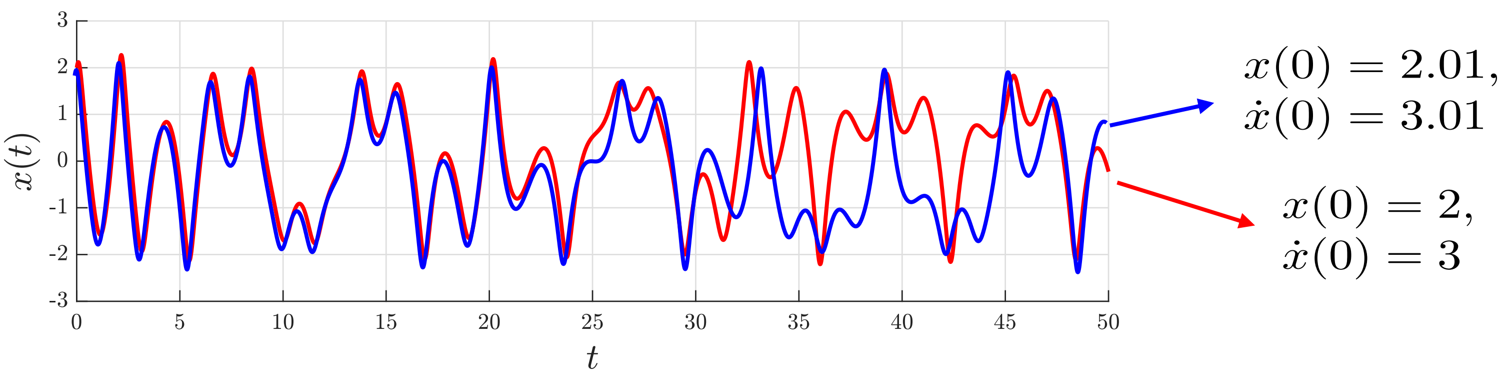
$$\ddot{x} + 0.1\dot{x} + x^5 = u(t), \quad u(t) = 6\sin t \qquad \begin{aligned} x(0) &= 2, \\ \dot{x}(0) &= 3 \end{aligned}$$



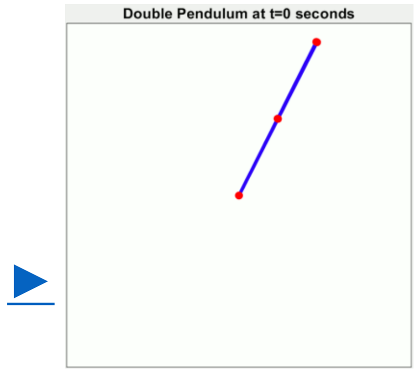
Nonlinear System Behavior: Chaos

- For stable linear systems, small differences in initial conditions can only cause small differences in output. However, output of strongly nonlinear systems is extremely sensitive to **initial conditions**.

$$\ddot{x} + 0.1\dot{x} + x^5 = u(t), \quad u(t) = 6\sin t$$



- Starting the pendulum from a slightly different initial condition would result in a vastly different trajectory.

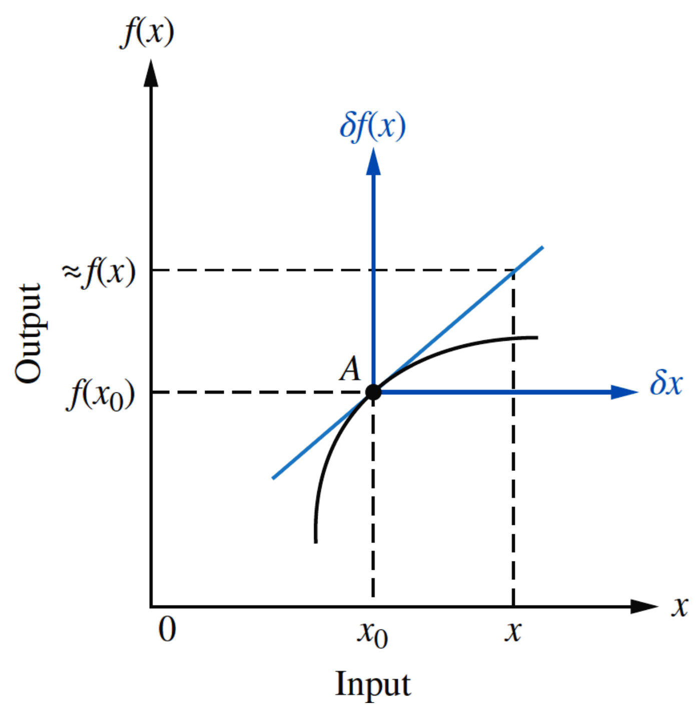


Linearization of Nonlinear Systems

Linearization of Nonlinear Systems

In control engineering, a normal operation of the system may be around an **equilibrium point** or **a limited operating range**. Therefor, it is possible to approximate the nonlinear system by an equivalent linear system within the limited operating range.

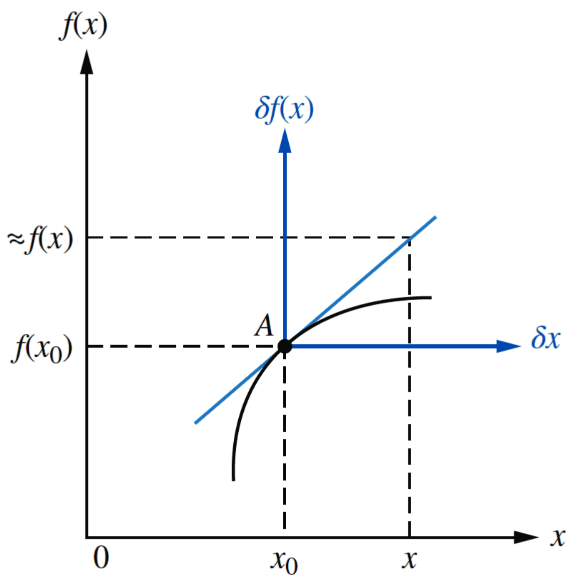
- Linear approximations simplify the analysis and design of a system.



Linear Approximation of Nonlinear Mathematical Models

The linearization procedure is based on (1) the expansion of nonlinear function $f(x)$ into a **Taylor Series** about the operating point $A\left(x_0, y_0 = f(x_0)\right)$ and (2) the retention of only the linear term.

Note: Since the variables deviate only slightly from the operating condition $(x - x_0)$, higher-order terms of the Taylor series expansion can be neglected.



$$y = f(x)$$

$$f(x) = f(x_0) + \frac{f'(x_0)}{1!}(x - x_0) + \frac{f''(x_0)}{2!}(x - x_0)^2 + \cdots$$

$$\boxed{f(x) \approx f(x_0) + f'(x_0)(x - x_0)}$$

(A straight-line relationship)

Expressing this straight line in frame $\delta x - \delta f(x)$: $\begin{cases} \delta x = x - x_0 \\ \delta f(x) = f(x) - f(x_0) \end{cases} \Rightarrow \quad \delta f(x) = f'(x_0)\delta x$

Example

Linearize $f(x) = 5\cos x$ about $x = \pi/2$.

Method 1:

$$f(x) \approx f(x_0) + f'(x_0)(x - x_0)$$

$$f(x) \approx 5\cos\left(\frac{\pi}{2}\right) - 5\sin\left(\frac{\pi}{2}\right)\left(x - \frac{\pi}{2}\right)$$

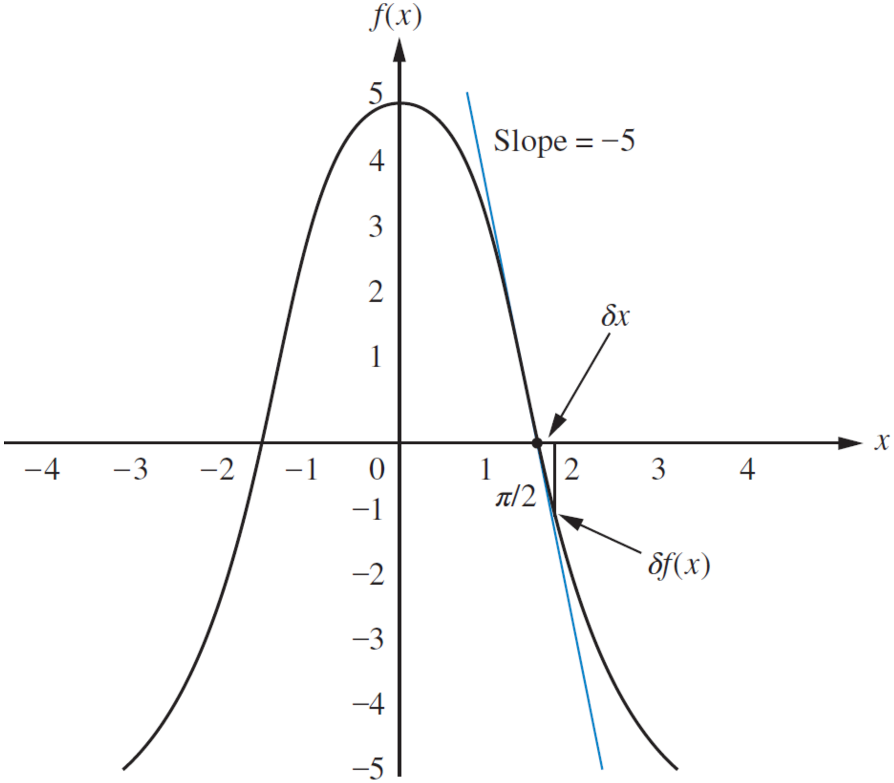
$$f(x) \approx -5\left(x - \frac{\pi}{2}\right)$$

Method 2:

For small deviation about $\frac{\pi}{2}$: $\quad x = \frac{\pi}{2} + \delta x$

$$f(x) = 5\cos\left(\delta x + \frac{\pi}{2}\right) = -5\sin\delta x$$

$$f(x) \approx -5\delta x$$



Example

Linearize $\ddot{x} + 2\dot{x} + \cos x = 0$ for small deviations about $x = \pi/4$.

Answer:

$$\ddot{x} + 2\dot{x} - \frac{\sqrt{2}}{2}x = -\frac{\sqrt{2}}{2} - \frac{\sqrt{2}}{2}\frac{\pi}{4}$$

State-Space Representation

Some Definitions

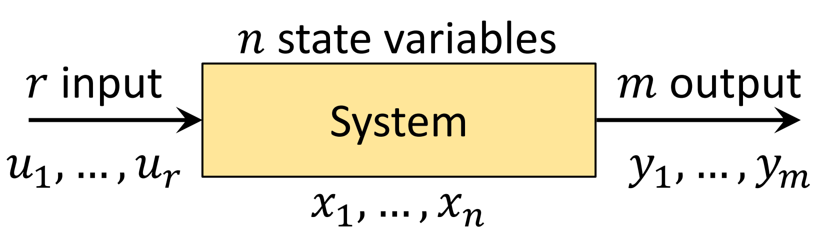
Linear Combination: A linear combination of n variables, x_i, is given by

$$S = k_1 x_1 + k_2 x_2 + \cdots + k_n x_n \quad , \quad k_i = \text{constant}, \quad i = 1, \dots, n$$

Linear Independence: A set of variables is said to be linearly independent if none of the variables can be written as a linear combination of the others.

System Variable: Any variable that responds to an input or initial conditions in a system.

State Variables: The **smallest set of linearly independent** system variables $(x_1, \dots, x_n)$ such that knowledge of these variables at $t = t_0$, together with knowledge of the input $(\boldsymbol{u}(t))$ for $t \geq t_0$, completely determines the behavior of the system for any time $t \geq t_0$.

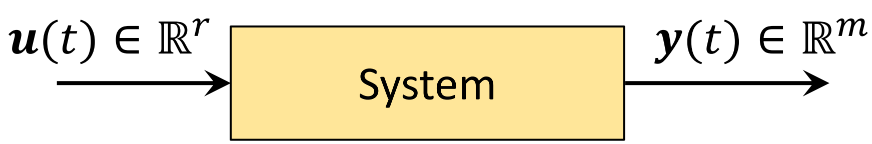


State-Space Representation

State-space Representation is a mathematical model of a physical system as a set of input $\boldsymbol{u}(t) \in \mathbb{R}^r$, output $\boldsymbol{y}(t) \in \mathbb{R}^m$, and state variables $\boldsymbol{x}(t) \in \mathbb{R}^n$ related by n simultaneous first-order differential equations.

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t), t) \quad \text{State Equation}$$
$$\boldsymbol{y}(t) = \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{u}(t), t) \quad \text{Output Equation}$$

$\boldsymbol{f}$ and $\boldsymbol{g}$ are vector functions.

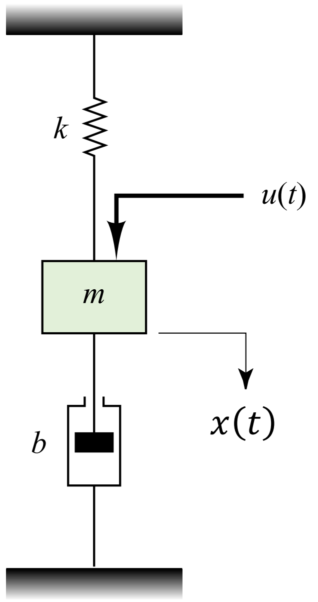


State Vector: $\boldsymbol{x}(t) \in \mathbb{R}^n$

The number of states (n) is the **order** of the system.

$x_1 = x$

$x_2 = \dot{x}$



Note: State variables need not be physically measurable or observable quantities.

Note: The choice of state variables of a system is not unique, but the number of states is unique. For all invertible $\boldsymbol{T} \in \mathbb{R}^{n\times n}$, $\overline{\boldsymbol{x}}(t) = \mathbf{T}\boldsymbol{x}(t)$ can be also the system state variables.

State-Space Representation

General Form:
MIMO, Nonlinear, Time Variant (General Form)

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}, t)$$
$$\boldsymbol{y}(t) = \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{u}, t)$$

MIMO, Linear, Time Variant

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

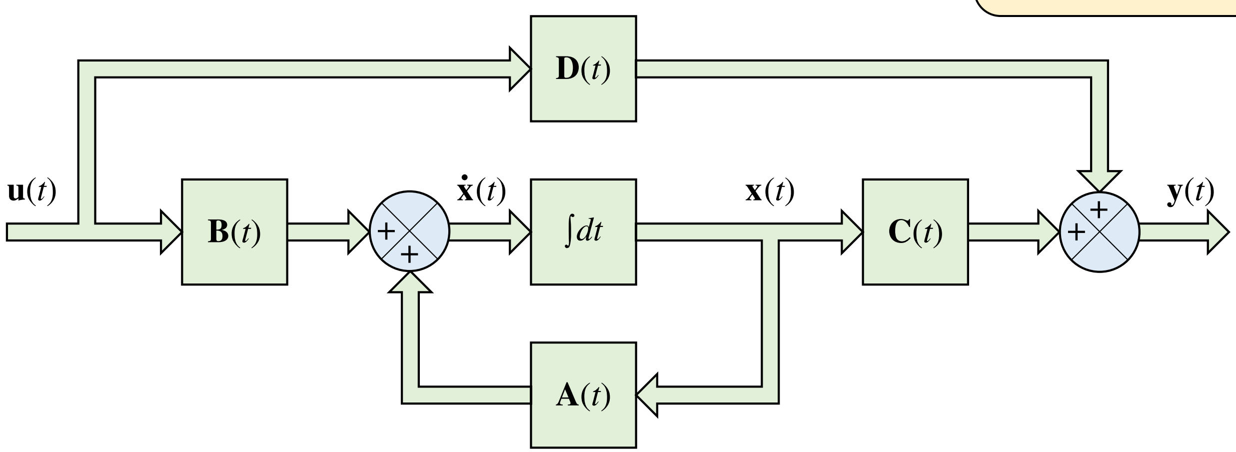
MIMO, Linear, Time Invariant

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

$\boldsymbol{A}$: State matrix, $\boldsymbol{B}$: Input matrix
$\boldsymbol{C}$: Output matrix, $\boldsymbol{D}$: Feedforward matrix

SISO, Linear, Time Invariant

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



State-Space Representation of LTI Systems

Consider a general, nth-order, linear differential equation with constant coefficients:

$$\frac{d^n x}{dt^n} + a_{n-1}\frac{d^{n-1}x}{dt^{n-1}} + \cdots + a_0 x = b_0 u$$

- An nth-order differential equation can be converted to n simultaneous **first-order differential equations**.
- There are many ways to do this conversion and obtain state-space representations of systems, such as **phase-variable** form, **controllable** canonical form, **observable** canonical form, **diagonal** canonical form, and **Jordan** canonical form.

A convenient way to choose state variables is to choose $x(t)$ and its $(n-1)$ derivatives as the state variables, which are called **phase variables**.

$\Downarrow$

State-Space Representation of LTI Systems

$$\frac{d^n x}{dt^n} + a_{n-1}\frac{d^{n-1}x}{dt^{n-1}} + \cdots + a_0 x = b_0 u$$

$$x_1 = x,\quad x_2 = \frac{dx}{dt},\quad \vdots,\quad x_n = \frac{d^{n-1}x}{dt^{n-1}}$$

Differentiating

$$\dot{x}_1 = \frac{dx}{dt},\quad \dot{x}_2 = \frac{d^2x}{dt^2},\quad \vdots,\quad \dot{x}_n = \frac{d^n x}{dt^n}$$

Substituting

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= x_3 \\ &\vdots \\ \dot{x}_{n-1} &= x_n \\ \dot{x}_n &= -a_{n-1}x_n - \cdots - a_0 x_1 + b_0 u \end{aligned}$$

Vector-Matrix Form

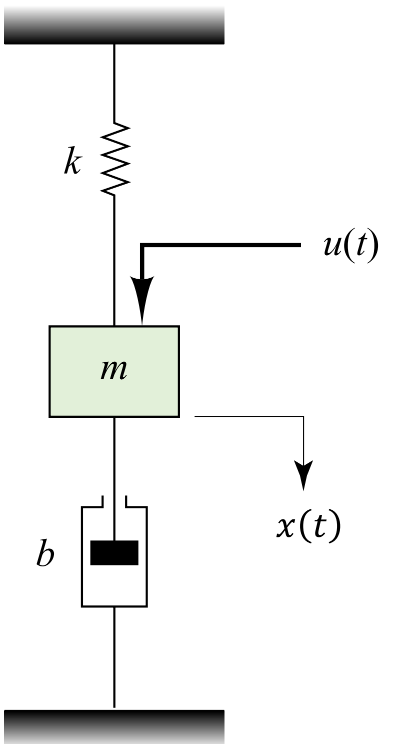
$$\boxed{\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}u \\ y &= \boldsymbol{C}\boldsymbol{x} \end{aligned}}$$

$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_n \end{bmatrix},\quad \boldsymbol{A} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \cdot & \cdot & \cdot & & \cdot \\ \cdot & \cdot & \cdot & & \cdot \\ \cdot & \cdot & \cdot & & \cdot \\ 0 & 0 & 0 & \cdots & 1 \\ -a_0 & -a_1 & -a_2 & \cdots & -a_{n-1} \end{bmatrix},\quad \boldsymbol{B} = \begin{bmatrix} 0 \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ 0 \\ b_0 \end{bmatrix},\quad \boldsymbol{C} = [1 \quad 0 \quad \cdots \quad 0]$$

(Output can be the first state)

Example

The external force $u(t)$ is the input to the system, and the displacement $x(t)$ of the mass, measured from the equilibrium position in the absence of the external force, is the output. Find the state equations.



Solution:

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

Let's define: $\begin{aligned} x_1 &= x \\ x_2 &= \dot{x} \end{aligned}$

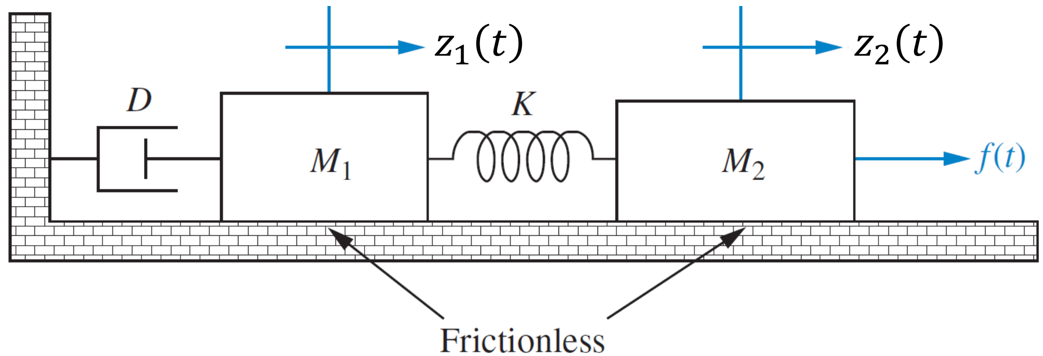
$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= \frac{1}{m}(-kx - b\dot{x}) + \frac{1}{m}u \end{aligned} \quad \Rightarrow \quad \begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -\frac{k}{m}x_1 - \frac{b}{m}x_2 + \frac{1}{m}u \\ y &= x = x_1 \end{aligned}$$

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

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}u \\ y &= \boldsymbol{C}\boldsymbol{x} \end{aligned}$$

Example

Find the state equations. What is the output equation if the output is $z_1(t)$?



Solution:

$$M_1\ddot{z}_1 + D\dot{z}_1 + Kz_1 - Kz_2 = 0$$
$$M_2\ddot{z}_2 - Kz_1 + Kz_2 = f(t)$$

$$x_1 = z_1 \qquad x_2 = \dot{z}_1 \qquad x_3 = z_2 \qquad x_4 = \dot{z}_2$$

$$\Rightarrow$$

$$\dot{x}_1 = x_2$$
$$\dot{x}_2 = -\frac{K}{M_1}x_1 - \frac{D}{M_1}x_2 + \frac{K}{M_1}x_3$$
$$\dot{x}_3 = x_4$$
$$\dot{x}_4 = +\frac{K}{M_2}x_1 - \frac{K}{M_2}x_3 + \frac{1}{M_2}f(t)$$

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\\ \dot{x}_4\end{bmatrix} = \begin{bmatrix}0 & 1 & 0 & 0\\ -K/M_1 & -D/M_1 & K/M_1 & 0\\ 0 & 0 & 0 & 1\\ K/M_2 & 0 & -K/M_2 & 0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\\ x_4\end{bmatrix} + \begin{bmatrix}0\\ 0\\ 0\\ 1/M_2\end{bmatrix}f(t)$$

$$y = \begin{bmatrix}1 & 0 & 0 & 0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\\ x_4\end{bmatrix}$$

Converting from SS to a TF

Deriving the transfer function from the state-space equations:

$$\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u} \\ \boldsymbol{y} &= \boldsymbol{C}\boldsymbol{x} + \boldsymbol{D}\boldsymbol{u} \end{aligned} \xrightarrow[\text{initial conditions}]{\text{Laplace transform assuming zero}} \begin{aligned} s\boldsymbol{X}(s) &= \boldsymbol{A}\boldsymbol{X}(s) + \boldsymbol{B}\boldsymbol{U}(s) \rightarrow \boldsymbol{X}(s) = (s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{B}\boldsymbol{U}(s) \\ \boldsymbol{Y}(s) &= \boldsymbol{C}\boldsymbol{X}(s) + \boldsymbol{D}\boldsymbol{U}(s) \end{aligned}$$

($\boldsymbol{I}$ is the identity matrix)

$$\boldsymbol{Y}(s) = \underbrace{[\boldsymbol{C}(s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{B} + \boldsymbol{D}]}_{\text{Transfer Function Matrix}}\boldsymbol{U}(s)$$

Transfer Function for a SISO system which $\boldsymbol{U}(s) = U(s)$ and $\boldsymbol{Y}(s) = Y(s)$:

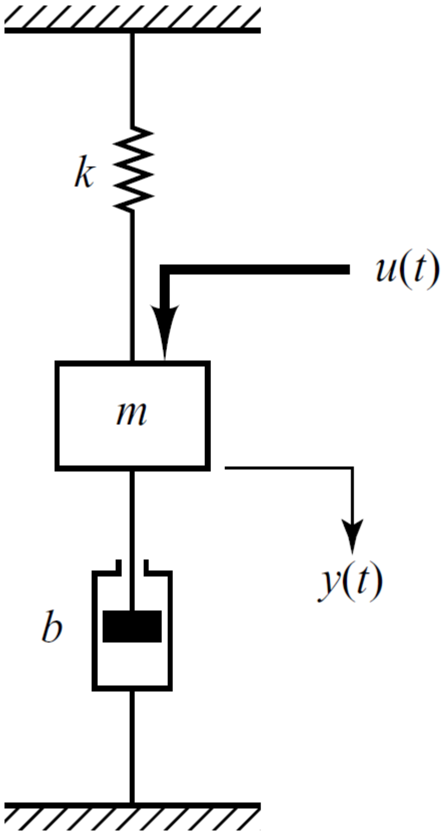
$$G(s) = \frac{Y(s)}{U(s)} = \boldsymbol{C}(s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{B} + D$$

Example

Obtain the transfer function $Y(s)/U(s)$ from the state-space equations of the system shown in the previous example.

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

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



Solution:

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

$$G(s) = \frac{1}{ms^2 + bs + k}$$

Converting a TF to SS

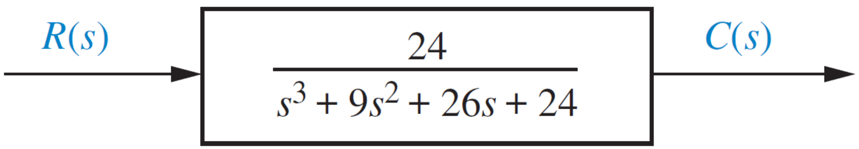
To convert a transfer function into state-space equations in phase-variable form, first convert the transfer function to a **differential equation** by cross-multiplying and taking the inverse Laplace transform, assuming zero initial conditions.

$$\frac{Y(s)}{U(s)} = \frac{b_0}{s^n + a_{n-1}s^{n-1} + \cdots + a_0} \longrightarrow \frac{d^n y}{dt^n} + a_{n-1}\frac{d^{n-1}y}{dt^{n-1}} + \cdots + a_0 y = b_0 u$$

Then, convert this nth-order differential equation to n simultaneous first-order differential equations.

Example

Find the state-space representation in phase-variable form.



Solution:

$$(s^3 + 9s^2 + 26s + 24)C(s) = 24R(s)$$

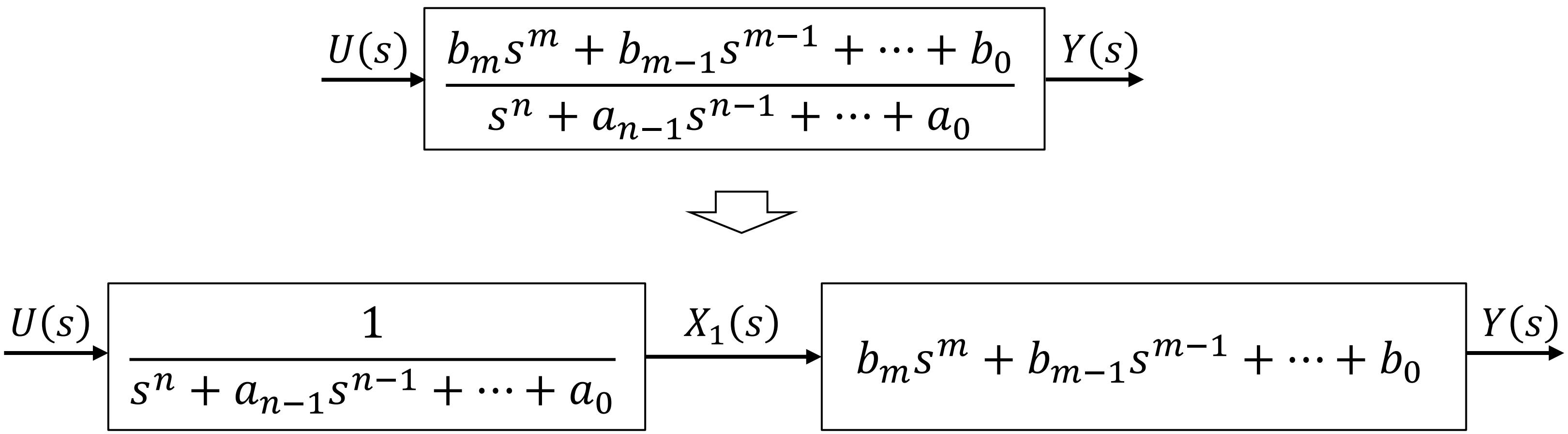
$$\dddot{c} + 9\ddot{c} + 26\dot{c} + 24c = 24r$$

$$\begin{aligned} x_1 &= c \\ x_2 &= \dot{c} \\ x_3 &= \ddot{c} \end{aligned} \longrightarrow \begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= x_3 \\ \dot{x}_3 &= -24x_1 - 26x_2 - 9x_3 + 24r \\ y &= c = x_1 \end{aligned}$$

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -24 & -26 & -9 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 24 \end{bmatrix} r \qquad y = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

Converting a TF to SS

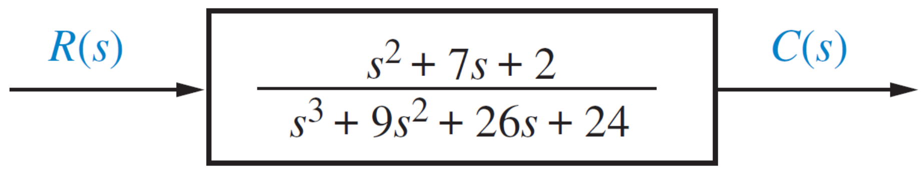
If a transfer function has a polynomial in s in the numerator, separate the transfer function into two cascaded transfer functions; the first is the denominator and the second is just the numerator.



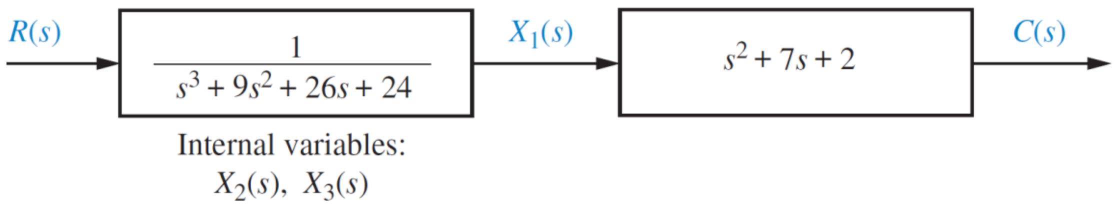
- The first transfer function with just the denominator is converted to the phase-variable representation in state space.
- The second transfer function with just the numerator yields the output equation.

Example

Find the state-space representation of the transfer function.



Solution:



From previous example:
$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\end{bmatrix} = \begin{bmatrix}0 & 1 & 0\\ 0 & 0 & 1\\ -24 & -26 & -9\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix} + \begin{bmatrix}0\\ 0\\ 1\end{bmatrix} r$$

$$C(s) = (s^2 + 7s + 2)X_1(s)$$

$$c = \ddot{x}_1 + 7\dot{x}_1 + 2x_1 \xrightarrow[\ddot{x}_1 = x_3]{x_1 = x_1,\ \dot{x}_1 = x_2} y = c(t) = x_3 + x_2 + 2x_1 \longrightarrow y = \begin{bmatrix}2 & 7 & 1\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix}$$

Using MATLAB and Control System Toolbox

Transfer-Function Representation Using tf, zpk

```
%% Transfer-Function Representation, Method 1
F1 = tf([3] , [1 2 5 0]); % or
F1 = tf([3] , conv([1 0],[1 2 5]));
F2 = tf([1] , [1 0]);
F3 = tf([1 0] , [1]);
F4 = tf(16, poly([0 -1 -1]));

%% Transfer-Function Representation, Method 2
s = tf('s');
F1 = 3/(s^3 + 2*s^2 + 5*s);
F2 = 1/s;
F3 = s;
F4 = 16/(s*(s+1)^2);

%% Transfer-Function Representation, Method 3
F1 = zpk([],[0 -1+2i -1-2i],[3]);
F2 = zpk([],[0],[1]);
F3 = zpk([0],[],[1]);
F4 = zpk([],[0 -1 -1],[16]);
```

Method 1

Method 2

Method 3

sys = tf(numerator,denominator)

numerator and denominator are row vectors of polynomial coefficients in order of descending power.

$$F_1(s) = \frac{3}{s(s^2 + 2s + 5)}$$

$$F_2(s) = \frac{1}{s}$$

$$F_3(s) = s$$

$$F_4(s) = \frac{16}{s(s+1)^2}$$

sys = zpk(zeros,poles,gain)

zero-pole-gain model with zeros and poles specified as row vectors of roots of numerator and denominator, and the scalar value of gain.

State-Space Representation Using ss and Conversions Using tf2ss, ss2tf

```
%% State-Space Representation
A = [-4 -1.5;4 0];
B = [2 0]';
C = [1.5 0.625];
D = 0;
T_ss = ss(A,B,C,D);
```

$$\dot{\boldsymbol{x}} = \begin{bmatrix} -4 & -1.5 \\ 4 & 0 \end{bmatrix} \boldsymbol{x} + \begin{bmatrix} 2 \\ 0 \end{bmatrix} u(t)$$
$$y = \begin{bmatrix} 1.5 & 0.625 \end{bmatrix} \boldsymbol{x}$$

```
% converting SS to TF
T_tf = tf(T_ss);
```

$$T(s) = \frac{3s + 5}{s^2 + 4s + 6}$$

```
%% TF to SS, SS to TF
num = [1 7 2];
den = [1 9 26 24];
[A, B, C, D] = tf2ss(num, den);
T1 = ss(A,B,C,D);
T1 = tf(T1);

% For SISO systems
[num,den] = ss2tf(A,B,C,D);
T2 = tf(num,den);
```

$$T_1(s) = T_2(s) = \frac{s^2 + 7s + 2}{s^3 + 9s^2 + 26s + 24}$$