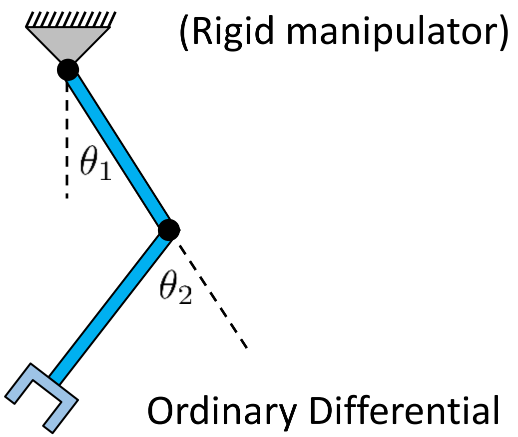
Ch4: Linear and Nonlinear Systems

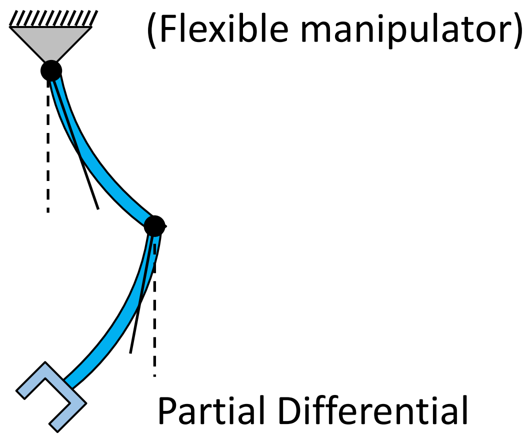
System Classifications

System Description (Analytical)

For analysis and design of control systems, the dynamic systems (i.e., mechanical, electrical, thermal, economic, biological, ...) must be **mathematically modeled** in terms of **differential equations** using fundamental **physical laws** (e.g., Newton-Euler's laws for mechanical systems and Kirchhoff's laws for electrical systems).



Ordinary Differential Equation (ODE)

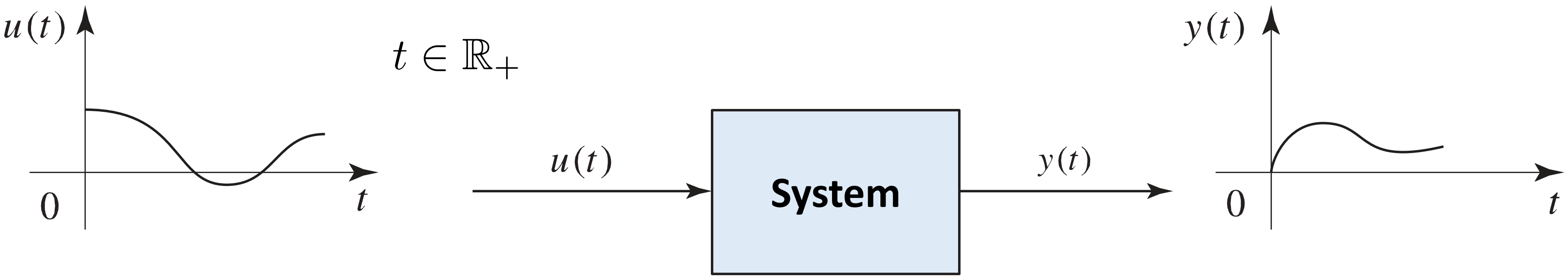


Partial Differential Equation (PDE)

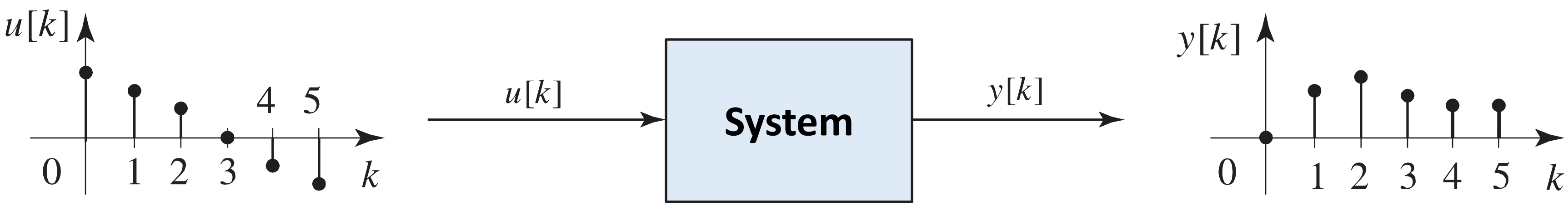
In obtaining a mathematical model, we must make a **compromise** between the **simplicity** of the model and the **accuracy** of the results of the analysis. We may simplify the system model in order to design a relatively simple controller.

Continuous-time vs. Discrete-time Systems

A system is called a **continuous-time** system if it accepts continuous-time signals as its input and generates continuous-time signals as its output.



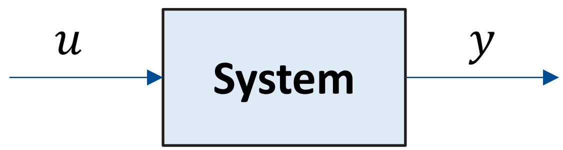
A system is called a **discrete-time** system if it accepts discrete-time signals as its input and generates discrete-time signals as its output.



$$u[k] = u(k\delta t)$$
$$k \in \mathbb{Z}_+$$

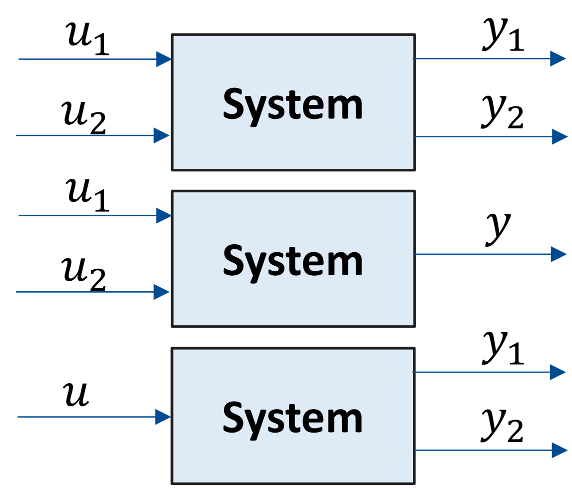
Single-variable vs. Multivariable Systems

A system with only one input and only one output is called a **single-variable** system or a **single-input single-output** (SISO) system.



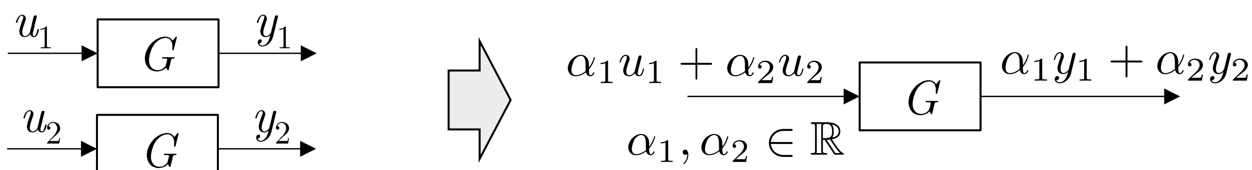
A system with two or more inputs and/or two or more outputs is called a **multivariable system**.

- Multi-input Multi-output (MIMO)
- Multi-input Single-output (MISO)
- Single-input Multi-output (SIMO)



Linear vs. Nonlinear Systems

A system is **Linear** w.r.t. its inputs and outputs iff it obeys the **Principle of Superposition**:

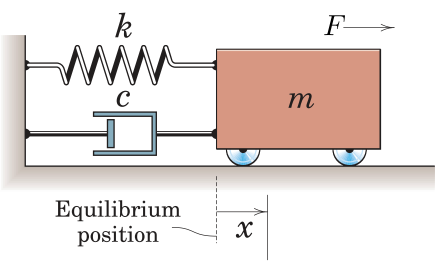


Note: A **differential equation** is linear if the **coefficients** are **constants** or **functions only of the independent variable**.

$$a_0(t)y + a_1(t)\dot{y} + a_2(t)\ddot{y} + \cdots + a_n(t)y^{(n)} = u(t)$$

Example: Mass-Spring-Damper

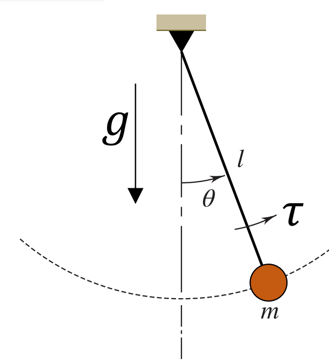
$$m\frac{d^2x}{dt^2} + c\frac{dx}{dt} + kx = F$$



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

Example: Pendulum

$$ml^2\frac{d^2\theta}{dt^2} + mgl\sin\theta = \tau$$

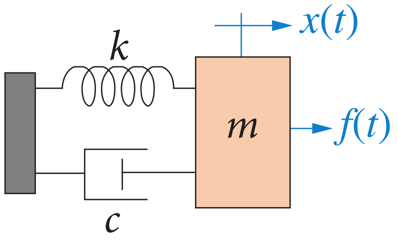


Time-Invariant vs. Time Varying Systems

A system is said to be **Time-Invariant** (or **Autonomous** for nonlinear systems) if the relationship between the input and output is **independent** of time.

- If the response to $u(t)$ is $y(t)$, then the response to $u(t-t_0)$ is $y(t-t_0)$.

Ex. A mass-spring-damper system which its physical parameters remains constant.



A system is said to be **Time-Varying/Varient** (or **Non-Autonomous** for nonlinear systems) if the relationship between the input and output is **dependent** of time.

Ex. A spacecraft system which its mass changes due to fuel consumption.

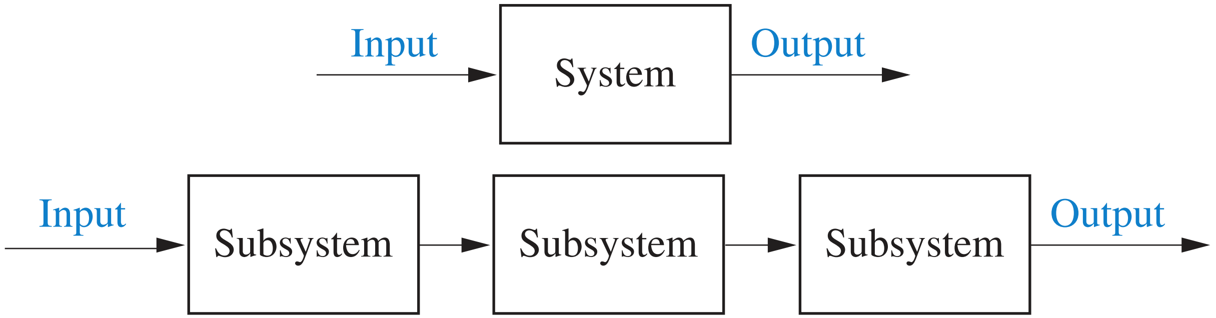
LTI: Linear Time-Invariant
LTV: Linear Time-Varying

System Representation

The ODE is not a satisfying representation because the system input $u(t)$ and output $y(t)$ appear throughout the equation. It is preferred a mathematical representation which the input, output, and system are separate parts and it can be modeled as a block diagram.

e.g., a SISO LTI system: $$\frac{d^n y(t)}{dt^n} + a_{n-1}\frac{d^{n-1} y(t)}{dt^{n-1}} + \cdots + a_0 y(t) = b_m \frac{d^m u(t)}{dt^m} + b_{m-1}\frac{d^{m-1} u(t)}{dt^{m-1}} + \cdots + b_0 u(t)$$

$$n \geq m$$

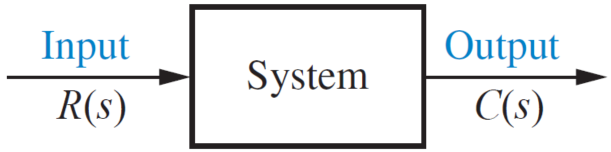


Two methods for representation of mathematical models of dynamic systems:

(1) **Transfer Function (TF)** in the **Frequency Domain**,
(2) **State-Space (SS) Representation** in the **Time Domain**.

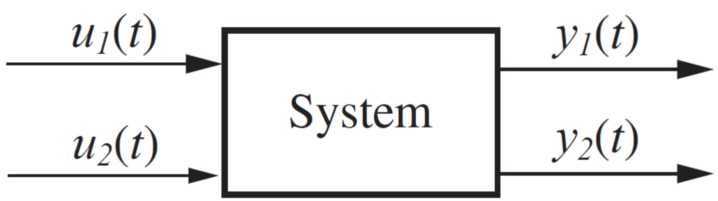
Classical vs. Modern Control Theory

Classical (or **Frequency-Domain** or **Transfer-Function**) approach [Since 1920s] can be applied only to **linear**, **time-invariant** (**LTI**), **SISO** (**Single-Input Single-Output**) systems with **zero initial conditions**, or systems that can be approximated as such. It does not use any knowledge of the **interior structure** of the system.



Transfer Function

Modern (or **Time-Domain** or **State-Space**) approach [Since 1960s] can be applied to a wide range of systems including **nonlinear**, **time variant** (**non-autonomous**), **MIMO** (**Multi-Input Multi-Output**) systems with **nonzero initial conditions** and also **LTI systems** modeled by the classical approach.



State Space

State-Space Representation

Some Definitions

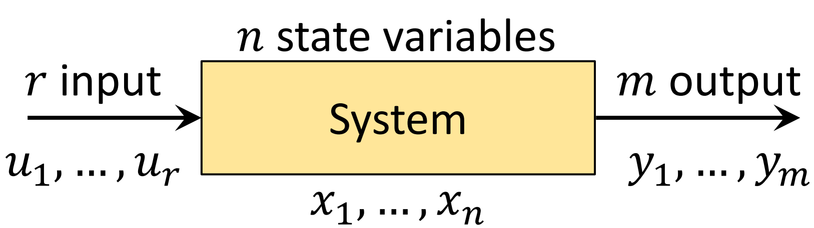
Linear Combination: A linear combination of n variables, x_i, is given by

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

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 $\boldsymbol{y}(t)$ for any time $t \geq t_0$.

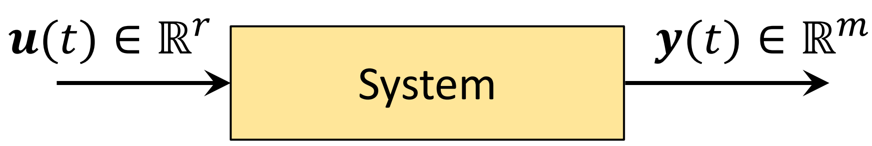


State-Space Representation

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.

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

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

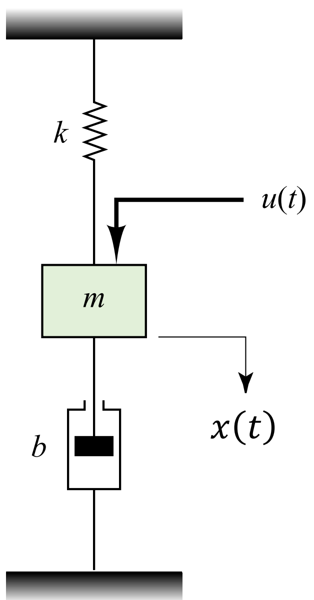


State Vector: $\boldsymbol{x}(t) \in \mathbb{R}^n$

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

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

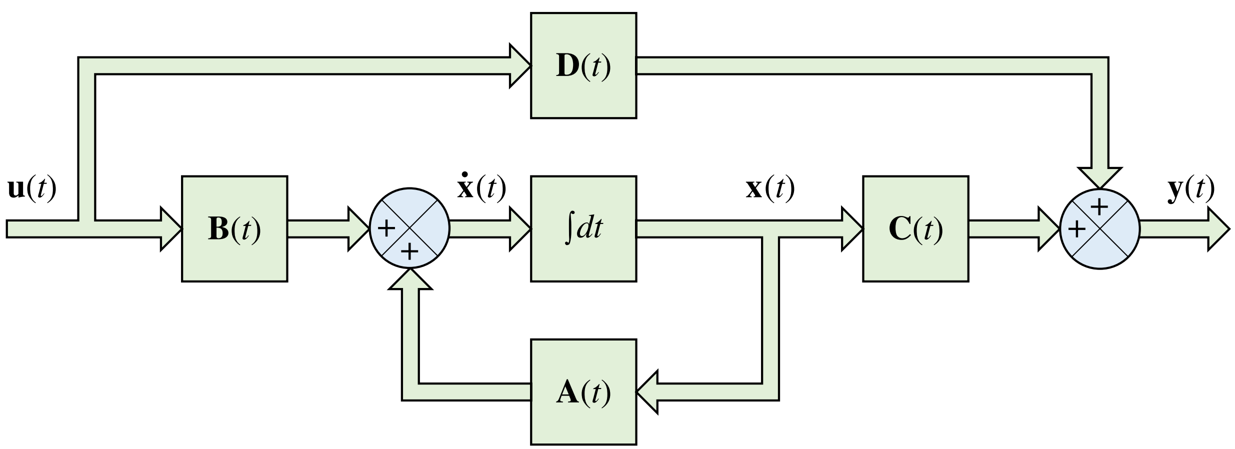
MIMO, Linear, Time Invariant

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

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

SISO, Linear, Time Invariant

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



Linear Systems

State-Space Representation of Linear 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

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

$$\begin{aligned} x_1 &= x \\ x_2 &= \frac{dx}{dt} \\ &\vdots \\ x_n &= \frac{d^{n-1}x}{dt^{n-1}} \end{aligned} \xrightarrow{\text{Differentiating}} \begin{aligned} \dot{x}_1 &= \frac{dx}{dt} \\ \dot{x}_2 &= \frac{d^2x}{dt^2} \\ &\vdots \\ \dot{x}_n &= \frac{d^n x}{dt^n} \end{aligned} \xrightarrow{\text{Substituting}} \begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= x_3 \\ &\vdots \\ \dot{x}_{n-1} &= x_n \\ \dot{x}_n &= -a_{n-1}x_n - \cdots - a_0 x_1 + b_0 u \end{aligned}$$

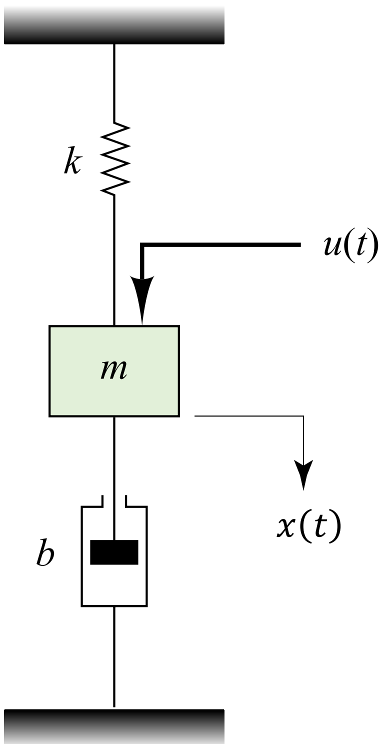
$$\xrightarrow{\text{Vector-Matrix Form}} \boxed{\begin{aligned} \dot{\boldsymbol{x}} &= \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}u \\ y &= \boldsymbol{C}\boldsymbol{x} \end{aligned}}$$

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

(Output can be the first state)

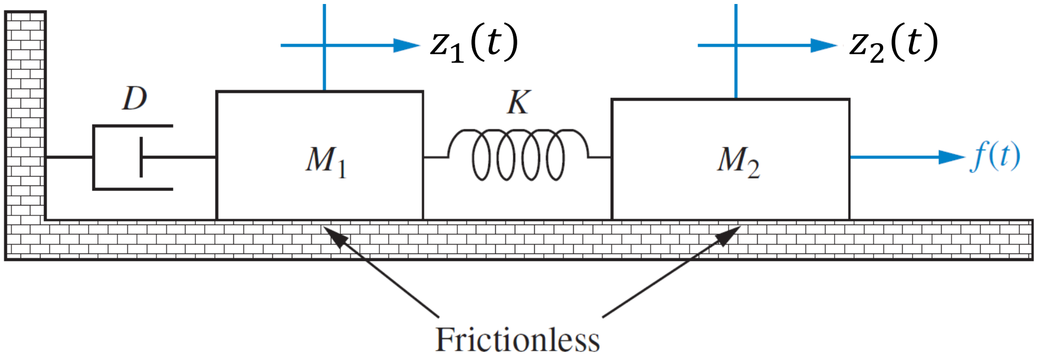
Example

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.



Example

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



Converting from SS to a TF

Deriving the transfer function from the state-space equations:

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

($\boldsymbol{I}$ 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 $\boldsymbol{Y}(s) = Y(s)$:

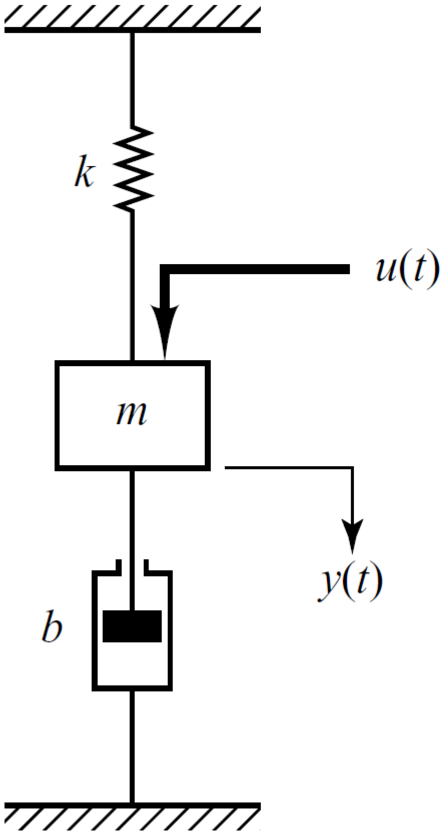
$$G(s) = \frac{Y(s)}{U(s)} = \boldsymbol{C}(s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{B} + D$$

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

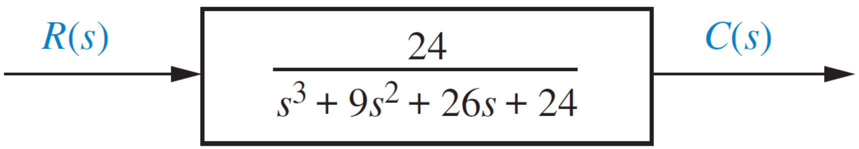
To convert a transfer function into state-space equations in phase-variable form, first convert the transfer function to a **differential equation** by cross-multiplying and taking the inverse Laplace transform, assuming zero initial conditions.

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

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.

$$U(s) \longrightarrow \boxed{\frac{b_m s^m + b_{m-1} s^{m-1} + \cdots + b_0}{s^n + a_{n-1} s^{n-1} + \cdots + a_0}} \longrightarrow Y(s)$$

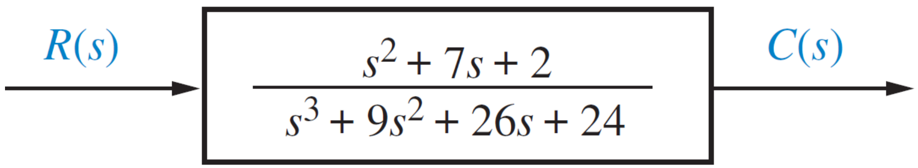
$$\Downarrow$$

$$U(s) \longrightarrow \boxed{\frac{1}{s^n + a_{n-1} s^{n-1} + \cdots + a_0}} \xrightarrow{X_1(s)} \boxed{b_m s^m + b_{m-1} s^{m-1} + \cdots + b_0} \longrightarrow Y(s)$$

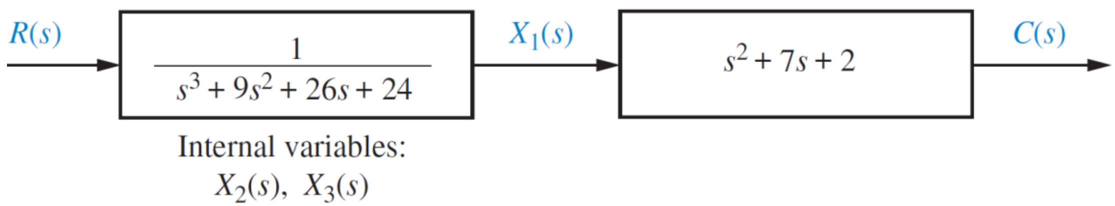
- The first transfer function with just the denominator is converted to the phase-variable representation in state space.
- The second transfer function with just the numerator yields the output equation.

Example

Find the state-space representation of the transfer function.



Solution:



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

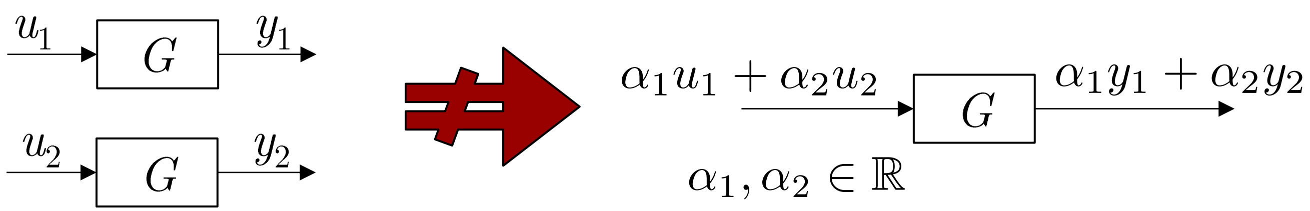
$$C(s) = (s^2 + 7s + 2)X_1(s)$$

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

Nonlinear Systems

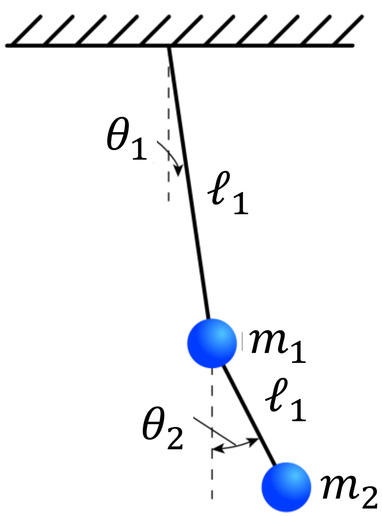
Nonlinear Systems

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.

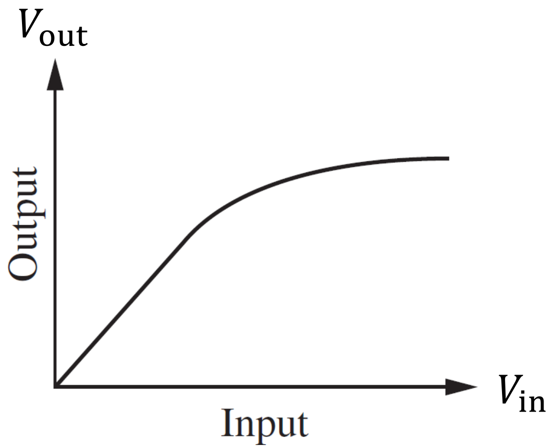
Double-Pendulum:

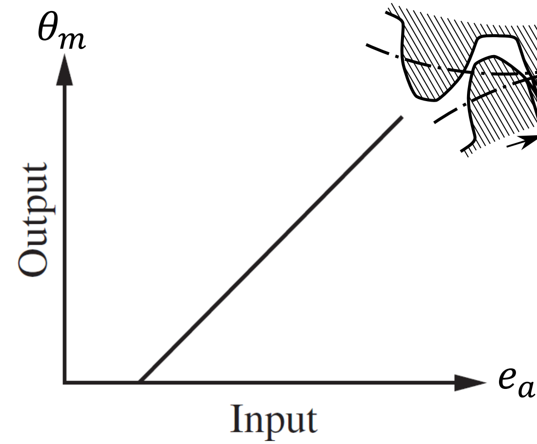


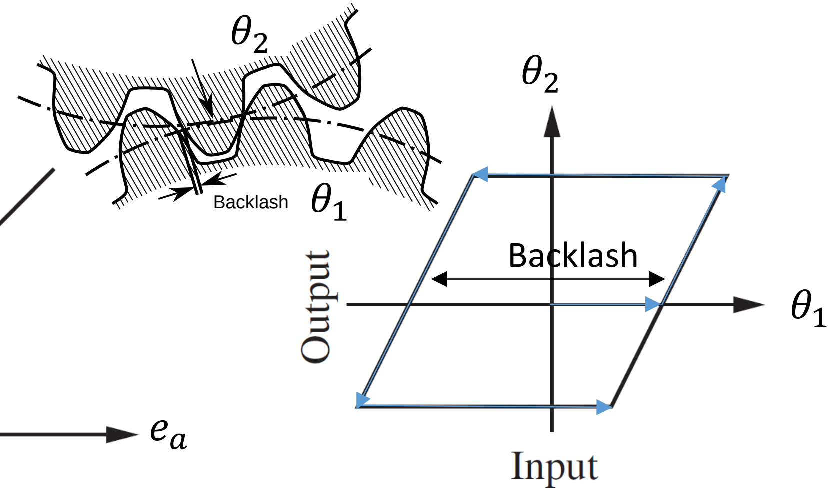
$$(m_1 + m_2)\ell_1\ddot{\theta}_1 + m_2\ell_2\ddot{\theta}_2\cos(\theta_1 - \theta_2) + m_2\ell_2\dot{\theta}_2^2\sin(\theta_1 - \theta_2) + g(m_1 + m_2)\sin\theta_1 = 0$$

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

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).

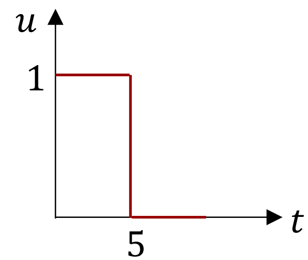
Nonlinear System Behavior: Step Response

A simplified model of the motion of an underwater vehicle (ROV):

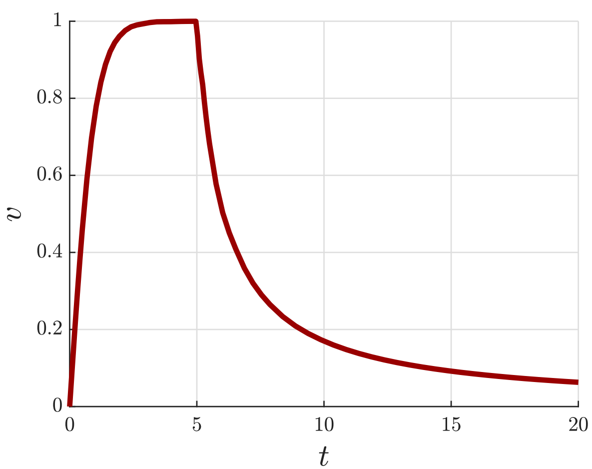
$$\dot{v} + |v|\,v = u$$

v: vehicle velocity
u: control input (thrust provided by a propeller)
$|v|\,v$: nonlinearity due to square-law drag

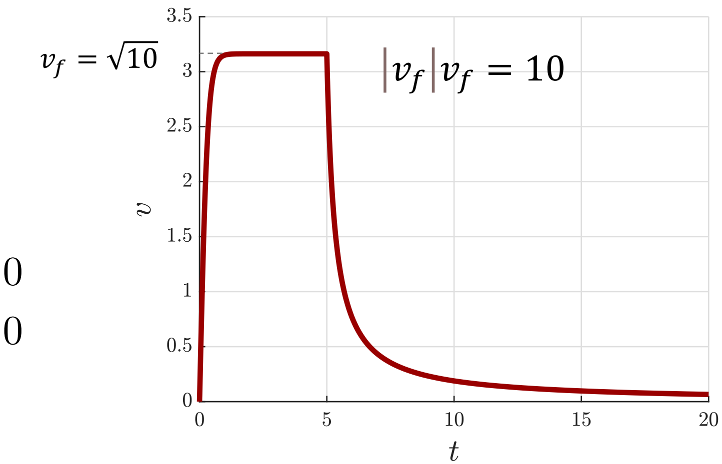
$$u = u_s(t) - u_s(t-5) \longrightarrow$$



$$u = 10\left(u_s(t) - u_s(t-5)\right)$$



$$u_s(t) = \begin{cases} 1 & t \geq 0 \\ 0 & t < 0 \end{cases}$$



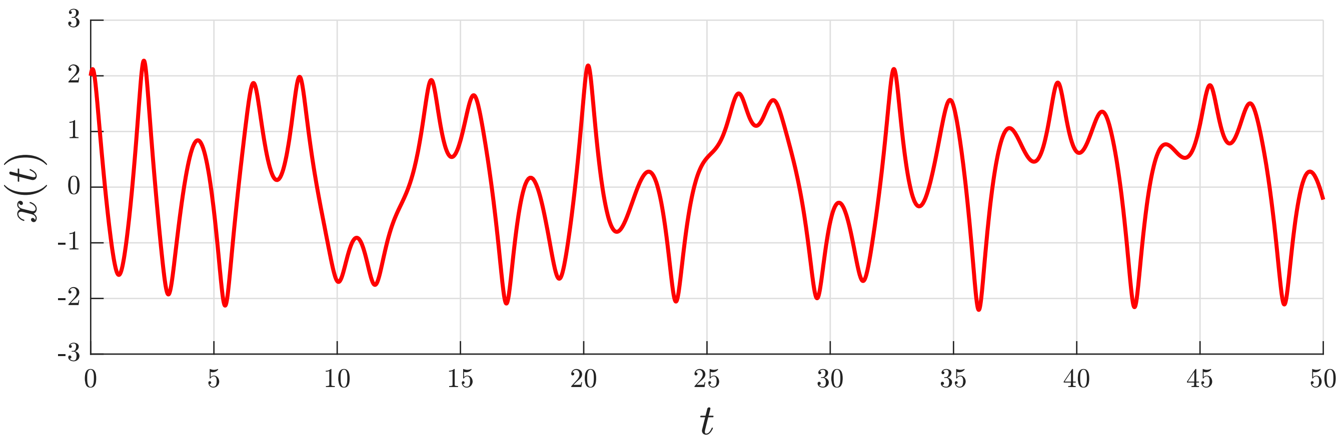
System settles much faster in response to the positive unit step than it does in response to the subsequent negative unit step.

Settling speed is not 10 times, as it would be in a linear system!

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.

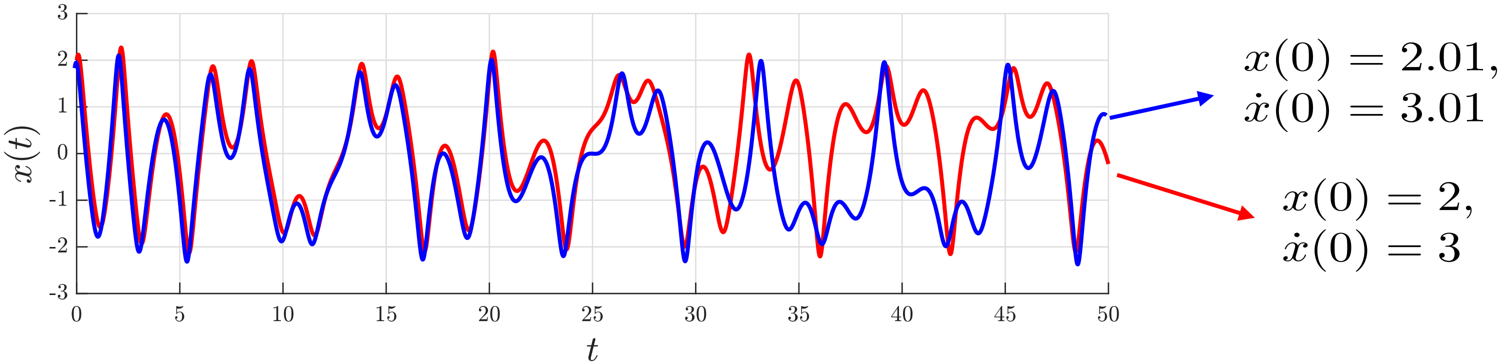
$$\ddot{x} + 0.1\dot{x} + x^5 = u(t), \quad u(t) = 6\sin t \qquad \begin{aligned} x(0) &= 2, \\ \dot{x}(0) &= 3 \end{aligned}$$



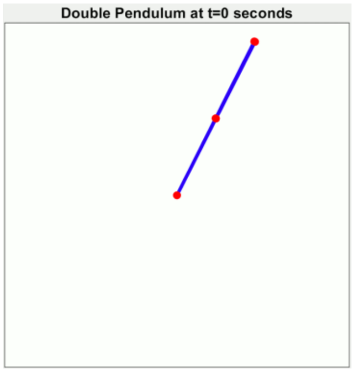
Nonlinear System Behavior: Chaos

- For stable linear systems, small differences in initial conditions can only cause small differences in output. Strongly nonlinear systems, however, can display a phenomenon called **chaos**, i.e., the system output is extremely sensitive to **initial conditions**.

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



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

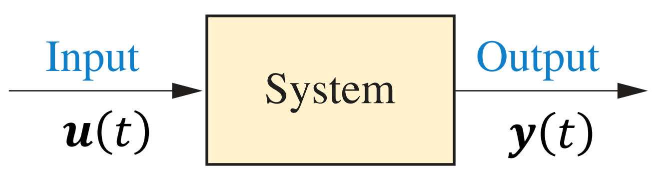


State-Space Representation of Nonlinear Systems

In nonlinear systems, e.g., robotic manipulators, the underlying physical behavior is described by **nonlinear differential equations**. Although the **state-space representation** is capable of handling these systems, the **transfer function methods fail**.

A general nth-order nonlinear, continuous-time, TIV, SISO system is described by a nonlinear, scalar, constant-coefficient ODE:

$$\frac{d^n x(t)}{dt^n} = f\left(x(t), x^{(1)}(t), \cdots, x^{(n-1)}(t), u(t), u^{(1)}(t), \cdots, u^{(m)}(t)\right) \qquad n \geq m$$



State-Space Representation of Nonlinear Systems

$$\frac{d^n x(t)}{dt^n} = f\left(x(t), x^{(1)}(t), \cdots, x^{(n-1)}(t), u(t), u^{(1)}(t), \cdots, u^{(m)}(t)\right) \qquad n \geq m$$

$$\begin{aligned} x_1(t) &= x(t) \\ x_2(t) &= \dot{x}(t) \\ &\vdots \\ x_n(t) &= x^{(n-1)}(t) \end{aligned} \quad \Rightarrow \quad \begin{aligned} \dot{x}_1(t) &= x_2(t) \\ \dot{x}_2(t) &= x_3(t) \\ &\vdots \\ \dot{x}_n(t) &= f\left(x(t), x^{(1)}(t), \cdots, x^{(n-1)}(t), u(t), u^{(1)}(t), \cdots, u^{(m)}(t)\right) \end{aligned}$$

$$\dot{\boldsymbol{x}}(t) = \mathbf{f}(\boldsymbol{x}(t), \boldsymbol{u}(t)) \qquad \boldsymbol{u}(t) = \left[u(t), u^{(1)}(t), \cdots, u^{(m)}(t)\right]^T \in \mathbb{R}^m$$

If we choose $y(t) = x(t)$, then $y(t) = \mathbf{C}\boldsymbol{x}(t)$ $\quad \mathbf{C} = [1, 0, \cdots, 0] \in \mathbb{R}^{1\times n}$

General Form:

$$\boxed{\begin{aligned} \dot{\boldsymbol{x}}(t) &= \mathbf{f}(\boldsymbol{x}(t), \boldsymbol{u}(t), t) \\ \boldsymbol{y}(t) &= \mathbf{g}(\boldsymbol{x}(t), \boldsymbol{u}(t), t) \end{aligned}}$$

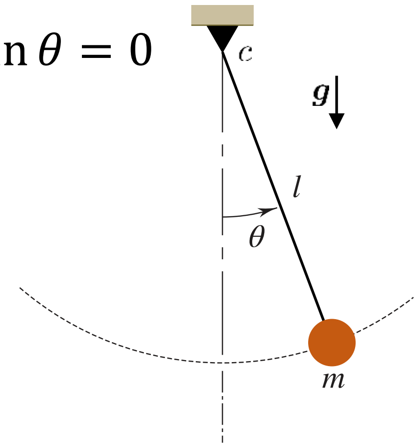
State Equation

Output Equation

f and **g** are nonlinear vector functions.

Examples

Find the state equations of a damped pendulum. $ml^2\ddot{\theta} + c\dot{\theta} + mgl\sin\theta = 0$



Find the state equations of a rigid manipulator. $\boldsymbol{M}(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + \boldsymbol{c}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) + \boldsymbol{g}(\boldsymbol{\theta}) = \boldsymbol{\tau}$

$\boldsymbol{\theta} \in \mathbb{R}^{n\times1}$
$\boldsymbol{M} \in \mathbb{R}^{n\times n}$
$\boldsymbol{c} \in \mathbb{R}^{n\times1}$
$\boldsymbol{g} \in \mathbb{R}^{n\times1}$
$\boldsymbol{\tau} \in \mathbb{R}^{n\times1}$

Autonomous & Non-Autonomous Systems

- Dynamic of a nonlinear system $\dot{\boldsymbol{x}}(t) = \mathbf{f}(\boldsymbol{x}(t), \boldsymbol{u}(t), t)$ when $\boldsymbol{u}(t) = \mathbf{0}$ can be represented as

$$\dot{\boldsymbol{x}}(t) = \mathbf{f}(\boldsymbol{x}(t), t)$$

- Moreover, the closed-loop dynamics of a feedback control system when $\boldsymbol{u}(t) = \mathbf{k}(\boldsymbol{x}, t)$ can be also represented as

$$\dot{\boldsymbol{x}}(t) = \mathbf{f}(\boldsymbol{x}(t), \boldsymbol{u}(t), t) \longrightarrow \dot{\boldsymbol{x}}(t) = \mathbf{f}(\boldsymbol{x}(t), \mathbf{k}(\boldsymbol{x}, t), t) \longrightarrow \dot{\boldsymbol{x}}(t) = \mathbf{f}(\boldsymbol{x}(t), t)$$

A nonlinear system of the form $\dot{\boldsymbol{x}}(t) = \mathbf{f}(\boldsymbol{x}(t), t)$ is said to be **Autonomous** (or **Time-Invariant**) if the function $\mathbf{f}$ does not depend explicitly on time, i.e.,

$$\dot{\boldsymbol{x}}(t) = \mathbf{f}(\boldsymbol{x}(t))$$

otherwise, the system is called **Non-autonomous** (or **Time-Varying/Varient**).

Autonomous & Non-Autonomous Systems

- A special class of nonlinear systems are linear systems. **LTI** systems are **autonomous** and **LTV** systems are **non-autonomous**.

$$\dot{\boldsymbol{x}}(t) = \mathbf{A}(t)\boldsymbol{x}(t)$$

$$\dot{\boldsymbol{x}}(t) = \mathbf{A}\boldsymbol{x}(t)$$

- The non-autonomous nature of a control system may be due to a time-variation either in the plant or in the control law, e.g., trajectory trackers or adaptive controllers (adaptive controllers for linear time-invariant plants usually make the closed-loop control systems nonlinear and non-autonomous).

$$\dot{\boldsymbol{x}}(t) = \mathbf{f}(\boldsymbol{x}(t), t)$$

Example: the closed-loop system of the simple plant

$$\dot{x} = -x + u$$

by choosing u

$$u(t) = -x^2 \sin t$$

as is nonlinear and non-autonomous.

$$\dot{x} = -x - x^2 \sin t$$

Equilibrium Points

Equilibrium Points

A state $\boldsymbol{x}_e$ is an **Equilibrium Point** (or **Equilibrium State**) of the system if once $\boldsymbol{x} = \boldsymbol{x}_e$, it remains equal to $\boldsymbol{x}_e$ for all future time.

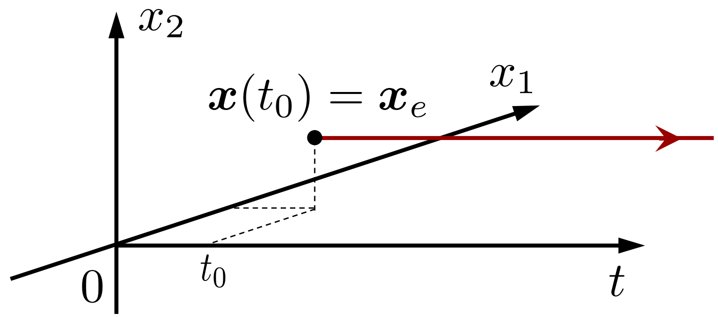
$$\dot{\boldsymbol{x}} = \mathbf{f}(\boldsymbol{x}_e) = \mathbf{0} \quad \forall t \geq 0$$

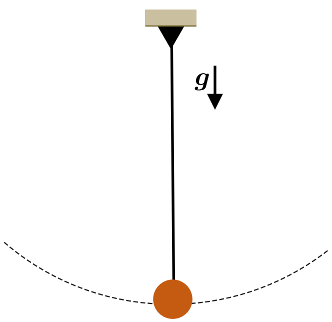
(For Autonomous Systems)

$$\dot{\boldsymbol{x}} = \mathbf{f}(\boldsymbol{x}_e, t) = \mathbf{0} \quad \forall t \geq t_0$$

(For Non-autonomous Systems)

i.e., a point for which if the system starts there (initial state $\boldsymbol{x}(t_0) = \boldsymbol{x}_e$) it will remain there for all future time.





- Stability of a system ≡ stability of systems at equilibrium points. Thus, many stability problems are naturally formulated with respect to equilibrium points.

Example

Consider the system $\dot{x} = x^2 - 1$

This autonomous system has two equilibrium points ± 1.

Consider the system $\dot{x} = tx^2 - 1$

This non-autonomous system has no equilibrium points. Although it might seem that it has 2 equilibrium points $x_e = \pm 1$ at time $t_0 = 1$. However, these are not equilibrium points for all $t \geq 1$.

Isolated Equilibrium Points

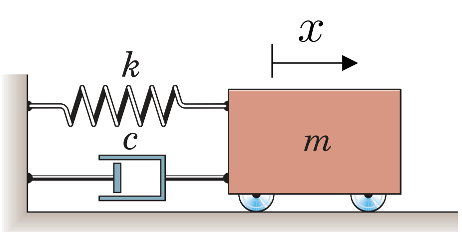
An equilibrium point $\boldsymbol{x}_e$ (at t_0) of $\dot{\boldsymbol{x}} = \mathbf{f}(\boldsymbol{x}, t)$ is said to be **Isolated** if there exists a real positive number r such that there may not be any equilibrium point other than $\boldsymbol{x}_e$ in Ω, where

$$\Omega = \{\boldsymbol{x} \in \mathbb{R}^n : \|\boldsymbol{x} - \boldsymbol{x}_e\| < r\}$$

In the case that there does not exist any r that satisfies the above then the equilibrium point $\boldsymbol{x}_e$ is **not isolated**.

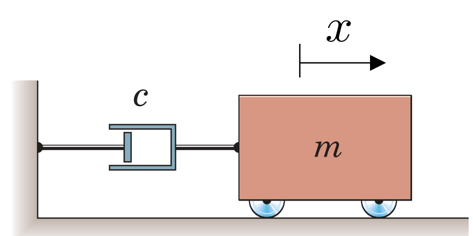
- A **linear** system (LTI or LTV) has a **single** isolated equilibrium point at the origin $\mathbf{0}$ **<u>if</u>** $\mathbf{A}$ is nonsingular. However, a **nonlinear** system often has **more than one** isolated equilibrium point.

$$\dot{\boldsymbol{x}}(t) = \mathbf{A}(t)\boldsymbol{x}(t)$$



$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -\dfrac{k}{m} & -\dfrac{c}{m} \end{bmatrix}$$

$$\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} = \mathbf{0}$$



$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ 0 & -\dfrac{c}{m} \end{bmatrix}$$

All x are equilibrium points!

Shifting an Equilibrium Point to Origin

If the equilibrium point of interest is not at the origin, by defining the difference between the original state $\boldsymbol{x}$ and the specific equilibrium point $\boldsymbol{x}_e$ as a new set of state variables, one can always shift the equilibrium point to the origin $\mathbf{0}$ (for analytical simplicity).

$$\boldsymbol{y} = \boldsymbol{x} - \boldsymbol{x}_e$$

$$\dot{\boldsymbol{x}} = \mathbf{f}(\boldsymbol{x}) \xrightarrow{\boldsymbol{x} = \boldsymbol{y} + \boldsymbol{x}_e} \dot{\boldsymbol{y}} = \mathbf{f}(\boldsymbol{y} + \boldsymbol{x}_e) \qquad (\boldsymbol{y} = \mathbf{0} \;\leftrightarrow\; \boldsymbol{x} = \boldsymbol{x}_e)$$

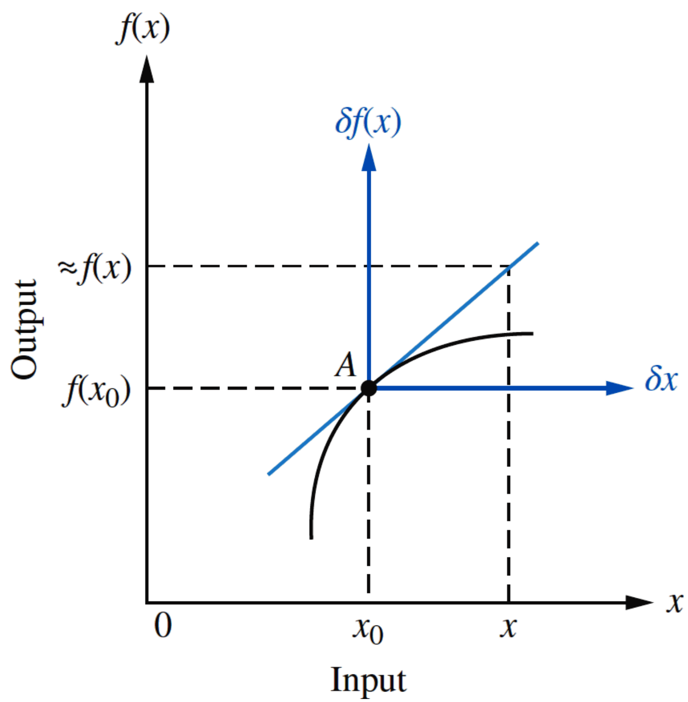
Therefore, instead of studying the behavior of $\dot{\boldsymbol{x}} = \mathbf{f}(\boldsymbol{x})$ in the neighborhood of $\boldsymbol{x}_e$, one can equivalently study the behavior of $\dot{\boldsymbol{y}} = \mathbf{f}(\boldsymbol{y} + \boldsymbol{x}_e)$ in the neighborhood of the origin $\mathbf{0}$.

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.

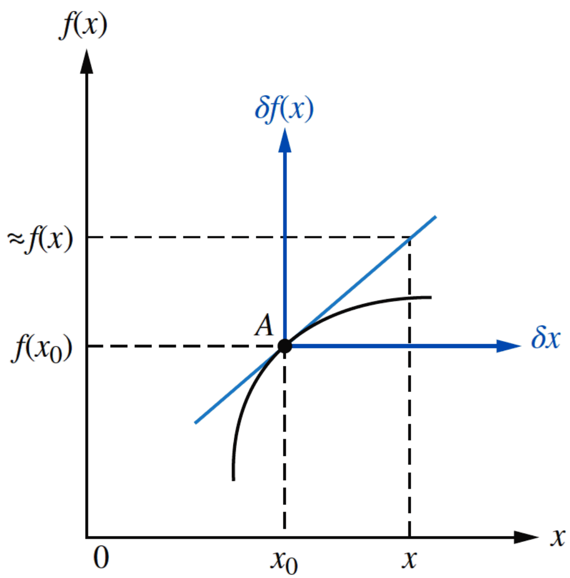
- Linear approximations simplify the analysis and design of a system.



Linear Approximation of Nonlinear Systems

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.



$$y = f(x)$$

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

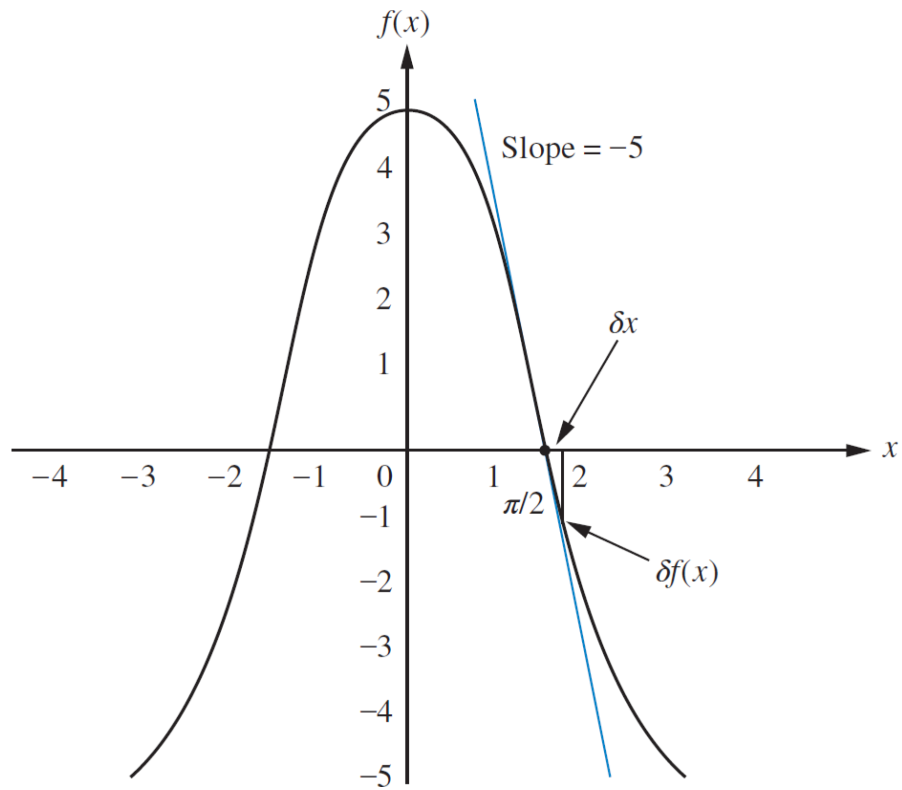
$$\boxed{f(x) \approx f(x_0) + f'(x_0)(x - x_0)}$$

(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 $x = \pi/2$.



Example

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

* **Note**: If δx is a small variable: $\sin \delta x \approx \delta x, \quad \cos \delta x \approx 1, \quad \delta x^2 \approx 0$