EECS 16B
Designing Information Devices and Systems II

Profs. Miki Lustig and JP Tennant
Department of Electrical Engineering and Computer Science

Announcements

- Midterm #1
  - Monday, February 26, 8-10pm
  - New material covered today will not be on MT #1
  - Review session is in Mulford 159, Wednesday 7-10pm
- Lab
  - New and Improved Lab #5 this week
  - No need to bring your car this week!

Today

- Resonant Circuits / Quality Factor - Correction from last time!
- Signals and Systems
- State Space Representation of a System
- Vector Differential Equations

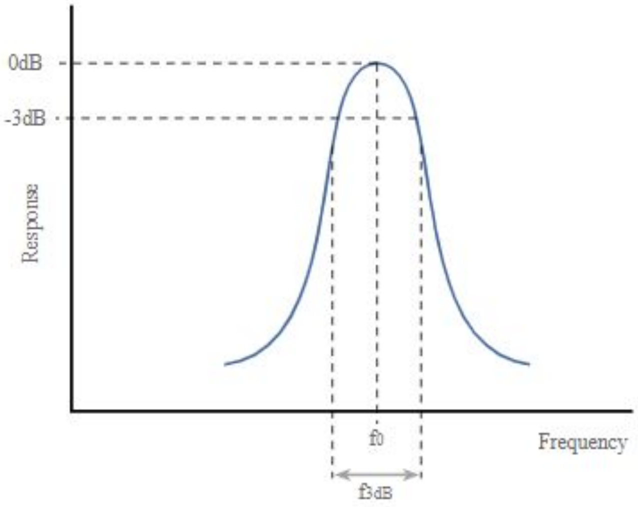
Quality Factor (Q)

- Quantifies resonance, i.e. the degree to which a system that oscillates is underdamped

Quality Factor (Q)

- Quantifies resonance, i.e. the degree to which a system that oscillates is underdamped
- Can be defined in terms of bandwidth

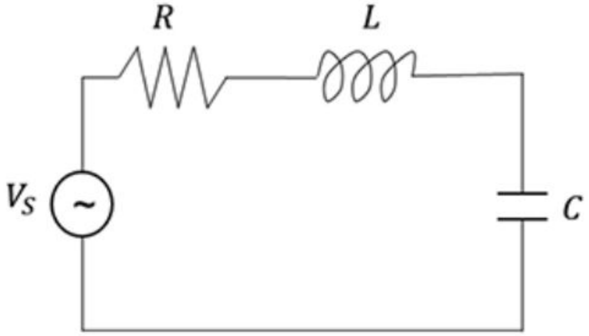
$$Q \triangleq \frac{\omega_0}{\Delta\omega_0} = \frac{f_0}{\Delta f_0}$$



Quality Factor (Q)

- Quantifies resonance, i.e. the degree to which a system that oscillates is underdamped
- Can be defined in terms of bandwidth
- Can also be defined in terms of energy or power

$$Q \triangleq \frac{E_{\text{stored}}}{E_{\text{lost per cycle}}} = \frac{P_{\text{reactive}}}{P_{\text{avg}}}$$

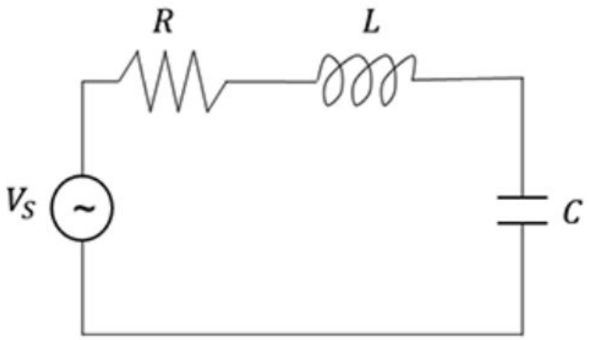


Quality Factor (Q)

- Quantifies resonance, i.e. the degree to which a system that oscillates is underdamped
- Can be defined in terms of bandwidth
- Can also be defined in terms of energy or power

$$Q \triangleq \frac{E_{\text{stored}}}{E_{\text{lost per cycle}}}$$

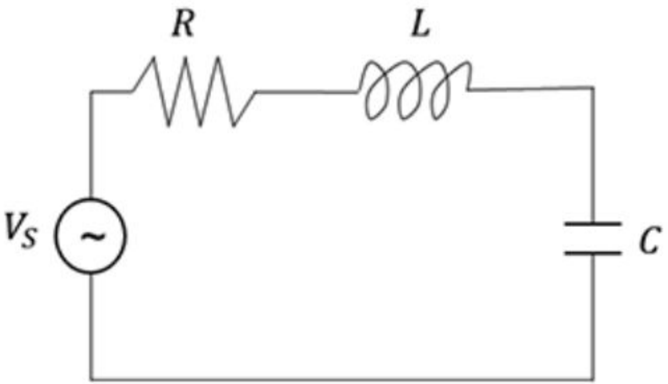
$$= \frac{P_{\text{reactive}}}{P_{\text{avg}}}$$



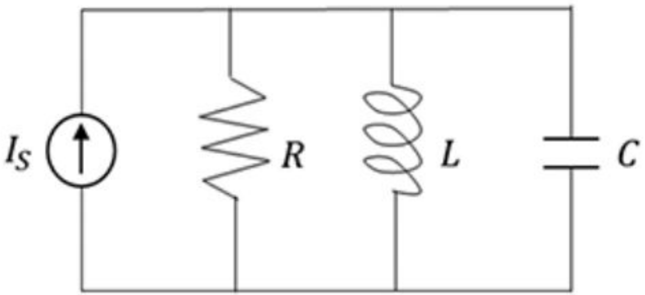
See Note 06C (optional)

Quality Factor (Q)

$$Q \triangleq \frac{E_{\text{stored}}}{E_{\text{lost per cycle}}} = \frac{P_{\text{reactive}}}{P_{\text{avg}}}$$

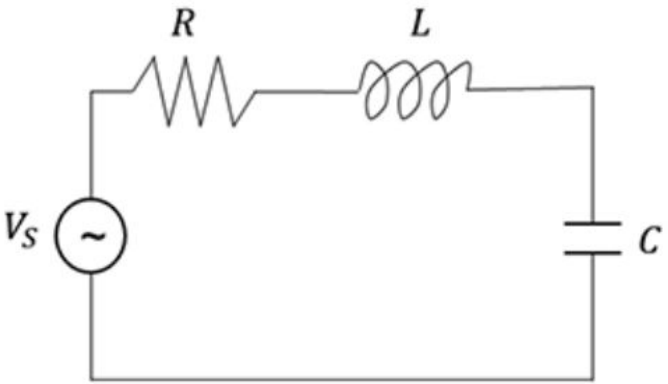


$$Q = \frac{\omega_0 L}{R} = \frac{1}{\omega_0 C R} = \frac{1}{R}\sqrt{\frac{L}{C}} \qquad \omega_0 = \frac{1}{\sqrt{LC}}$$

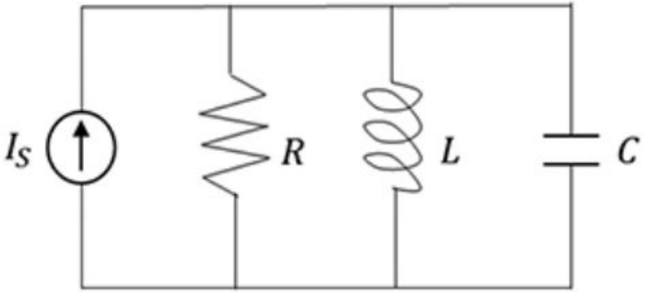


Quality Factor (Q)

$$Q \triangleq \frac{E_{\text{stored}}}{E_{\text{lost per cycle}}} = \frac{P_{\text{reactive}}}{P_{\text{avg}}}$$



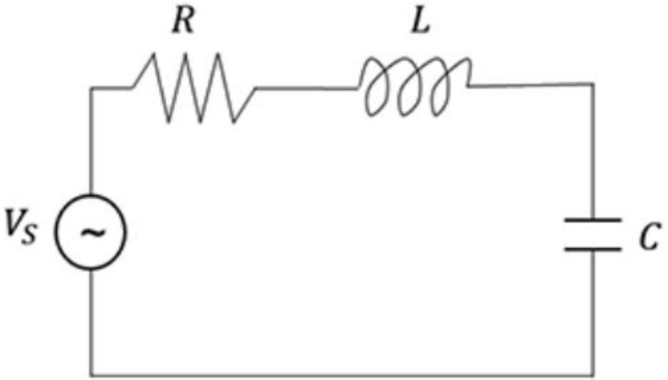
$$Q = \frac{\omega_0 L}{R} = \frac{1}{\omega_0 C R} = \frac{1}{R}\sqrt{\frac{L}{C}} \qquad \omega_0 = \frac{1}{\sqrt{LC}}$$



$$Q = \frac{R}{\omega_0 L} = \omega_0 C R = R\sqrt{\frac{C}{L}} \qquad \omega_0 = \frac{1}{\sqrt{LC}}$$

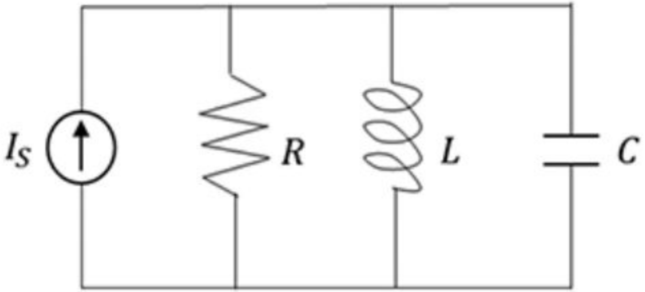
Quality Factor (Q)

$$Q \triangleq \frac{E_{\text{stored}}}{E_{\text{lost per cycle}}} = \frac{P_{\text{reactive}}}{P_{\text{avg}}}$$



$$Q = \frac{\omega_0 L}{R} = \frac{1}{\omega_0 C R} = \frac{1}{R}\sqrt{\frac{L}{C}} \qquad \omega_0 = \frac{1}{\sqrt{LC}}$$

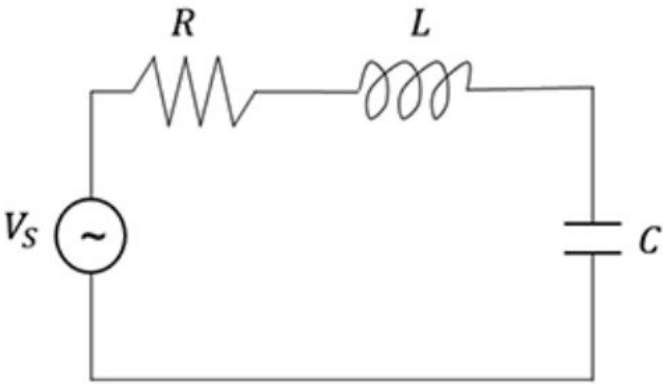
In lecture I wrote: $Q = \frac{R}{\omega_0 L} = \omega_0 C R$



$$Q = \frac{R}{\omega_0 L} = \omega_0 C R = R\sqrt{\frac{C}{L}} \qquad \omega_0 = \frac{1}{\sqrt{LC}}$$

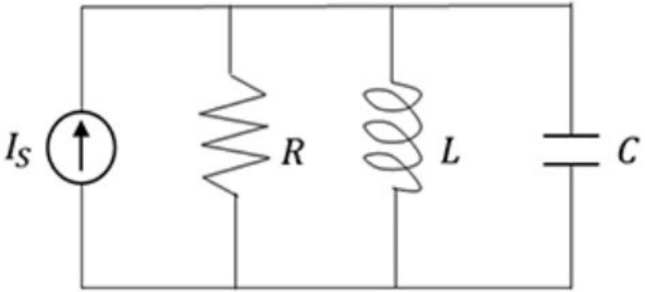
Quality Factor (Q)

$$Q \triangleq \frac{E_{\text{stored}}}{E_{\text{lost per cycle}}} = \frac{P_{\text{reactive}}}{P_{\text{avg}}}$$



$$Q = \frac{\omega_0 L}{R} = \frac{1}{\omega_0 CR} = \frac{1}{R}\sqrt{\frac{L}{C}} \qquad \omega_0 = \frac{1}{\sqrt{LC}}$$

In lecture I wrote: ~~$Q = \frac{R}{\omega_0 L} = \omega_0 CR$~~



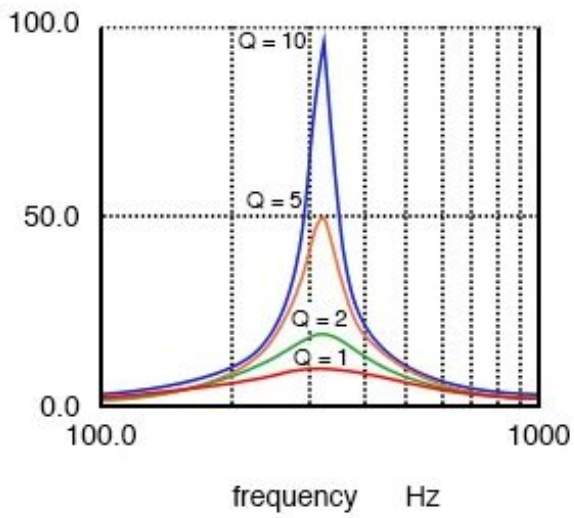
$$Q = \frac{R}{\omega_0 L} = \omega_0 CR = R\sqrt{\frac{C}{L}} \qquad \omega_0 = \frac{1}{\sqrt{LC}}$$

Quality Factor (Q)

Is Resonance (high Q) good?

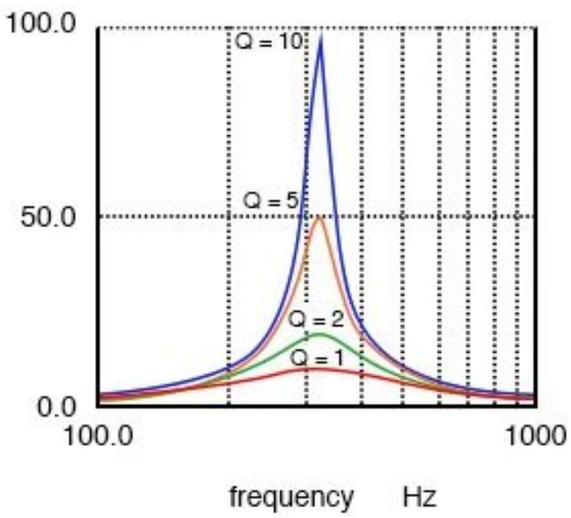
Quality Factor (Q)

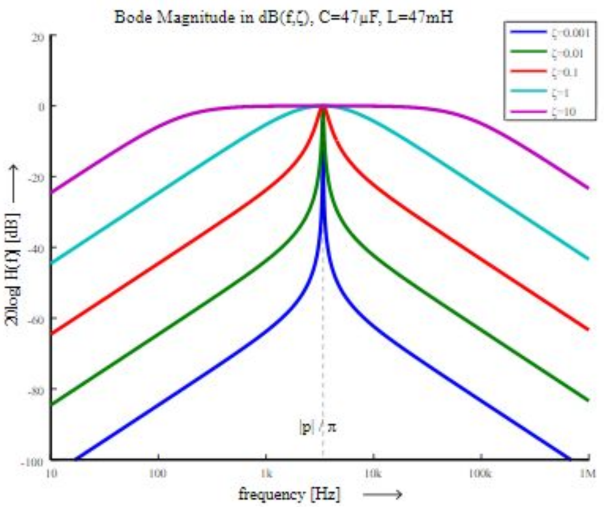
Is Resonance (high Q) good?



Quality Factor (Q)

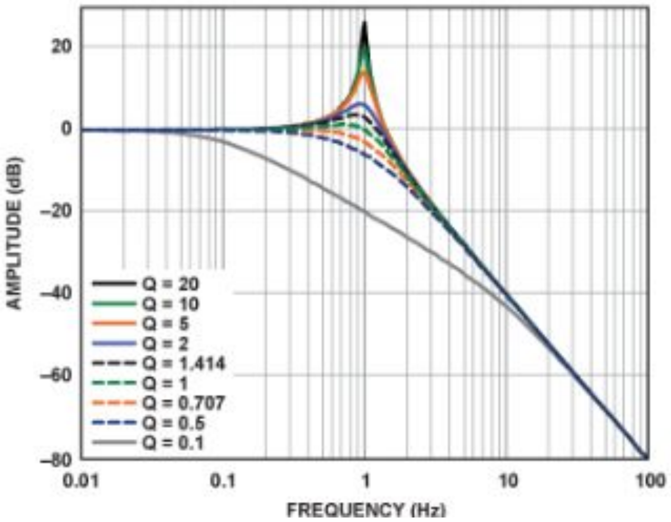
Is Resonance (high Q) good?

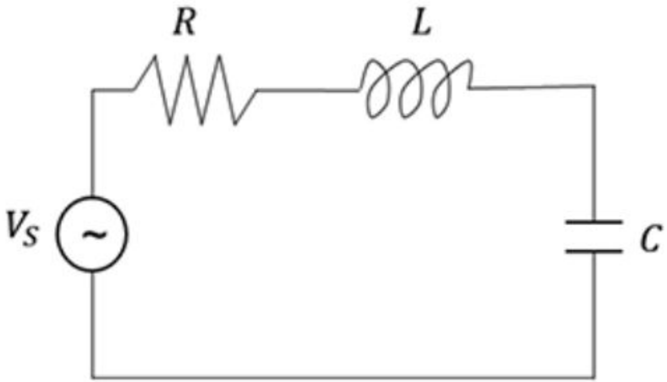




Quality Factor (Q)

Is Resonance (high Q) good?





Quality Factor (Q)

Is Resonance (high Q) good?

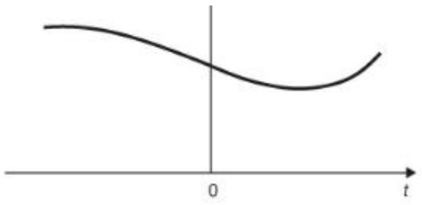
Quality Factor (Q)

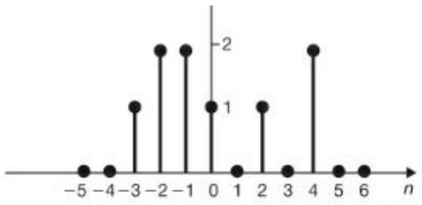
Is Resonance (high Q) good? Not always…

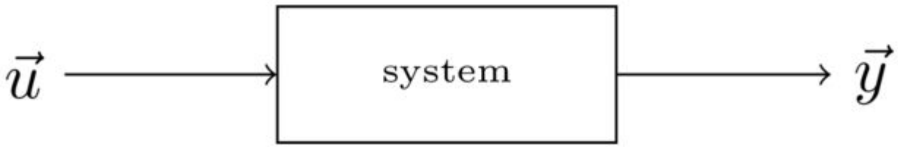
Quality Factor (Q)

Is Resonance (high Q) good? Not always…

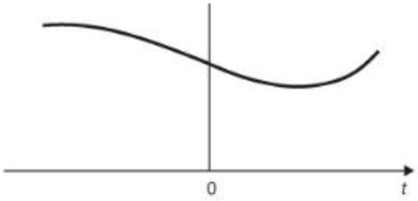
Signals and Systems

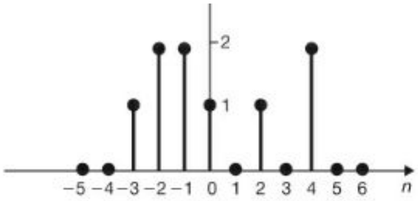


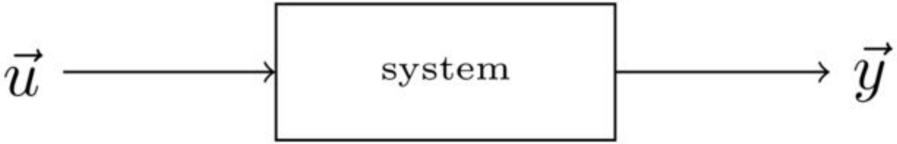




Signals and Systems



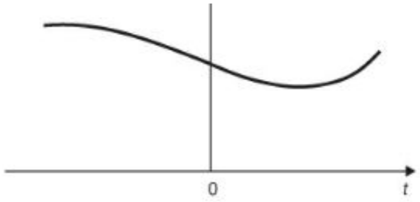


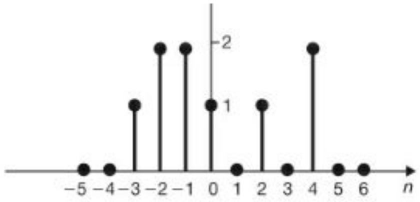


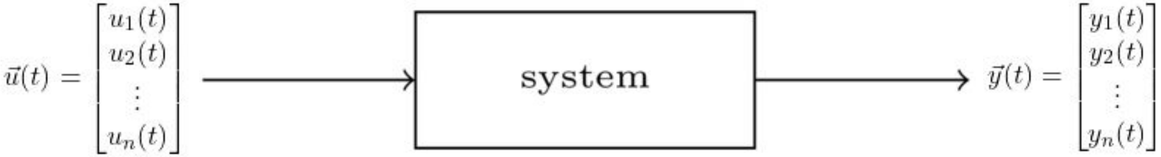
The systems we will study are generally:

- Multiple Input / Multiple Output

Signals and Systems



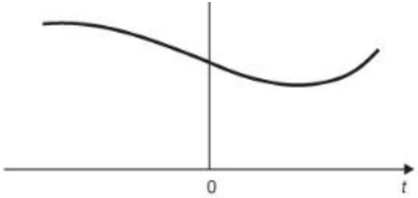


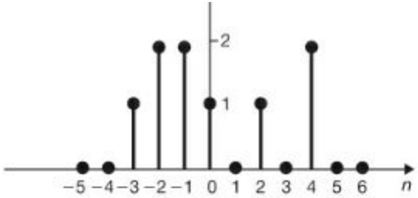


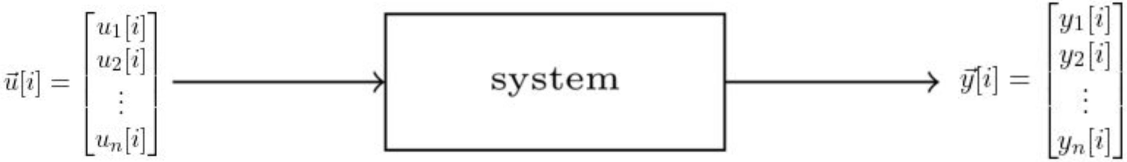
The systems we will study are generally:

- Multiple Input / Multiple Output

Signals and Systems



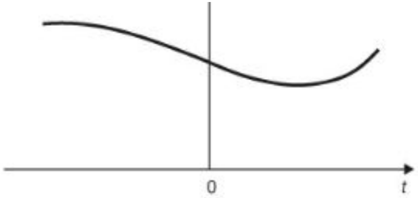


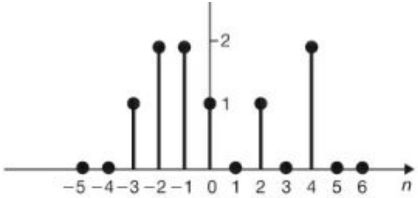


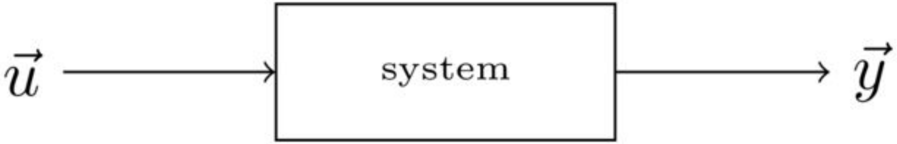
The systems we will study are generally:

- Multiple Input / Multiple Output

Signals and Systems



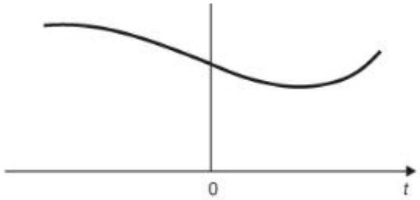


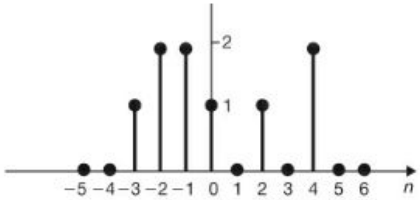


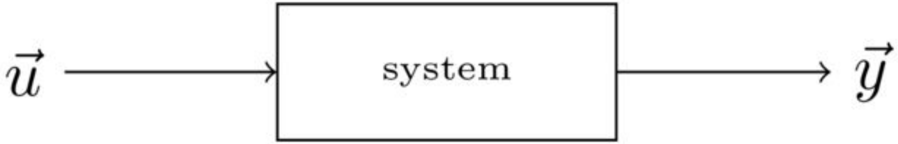
The systems we will study are generally:

- Multiple Input / Multiple Output
- Linear

Signals and Systems





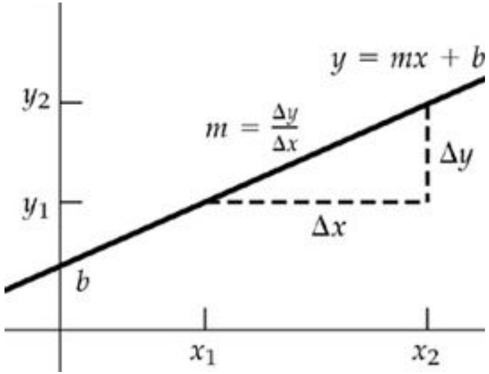


The systems we will study are generally:

- Multiple Input / Multiple Output
- Linear
- Time-Invariant

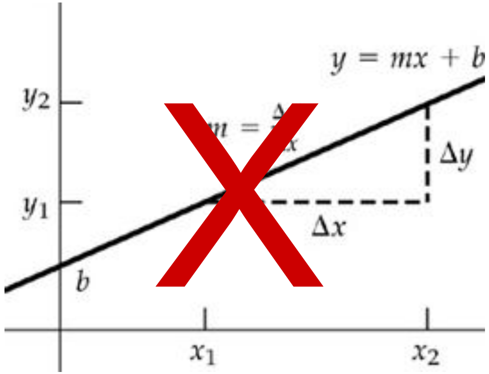
What does “linear” mean?

High School:

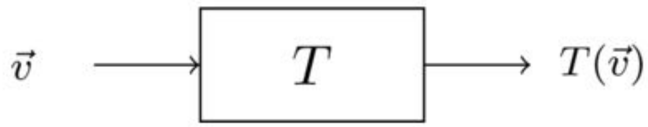


What does “linear” mean?

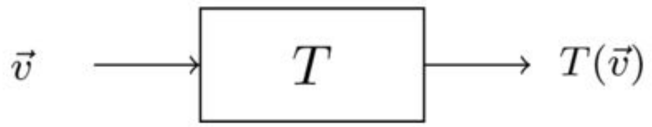
High School:



What does “linear” mean?

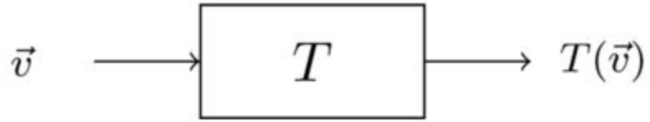


What does “linear” mean?



This is a linear transformation (linear system) iff:

What does “linear” mean?

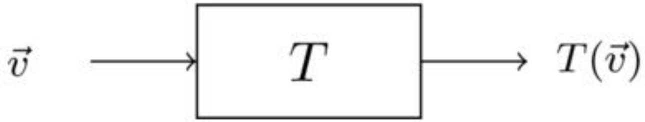


This is a linear transformation (linear system) iff:

$$T(C\vec{v}) = C\,T(\vec{v})$$

preserves scaling

What does “linear” mean?



This is a linear transformation (linear system) iff:

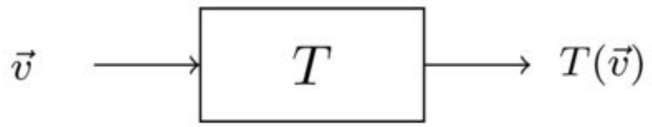
$$T\left(C\vec{v}\right) = C\,T\left(\vec{v}\right)$$

preserves scaling

$$T\left(\vec{v} + \vec{w}\right) = T\left(\vec{v}\right) + T\left(\vec{w}\right)$$

preserves addition

What does “linear” mean?



This is a linear transformation (linear system) iff:

$$T\left(C\vec{v}\right) = C\,T\left(\vec{v}\right)$$

preserves scaling

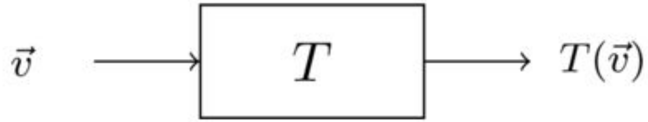
$$T\left(\vec{v} + \vec{w}\right) = T\left(\vec{v}\right) + T\left(\vec{w}\right)$$

preserves addition

$$T\left(C_1\vec{v} + C_2\vec{w}\right) = C_1T\left(\vec{v}\right) + C_2T\left(\vec{w}\right)$$

=> preserves linear combinations

What does “linear” mean?



This is a linear transformation (linear system) iff:

$$T\left(C\vec{v}\right) = C\,T\left(\vec{v}\right)$$ preserves scaling

$$T\left(\vec{v} + \vec{w}\right) = T\left(\vec{v}\right) + T\left(\vec{w}\right)$$ preserves addition

$$T\left(C_1\vec{v} + C_2\vec{w}\right) = C_1T\left(\vec{v}\right) + C_2T\left(\vec{w}\right)$$ => preserves linear combinations

=> the “superposition” principle applies

Suppose: $f(x) = 3x + 5$

True or False: The function f represents a linear transformation on x.

1. True
2. False

Check yourself:

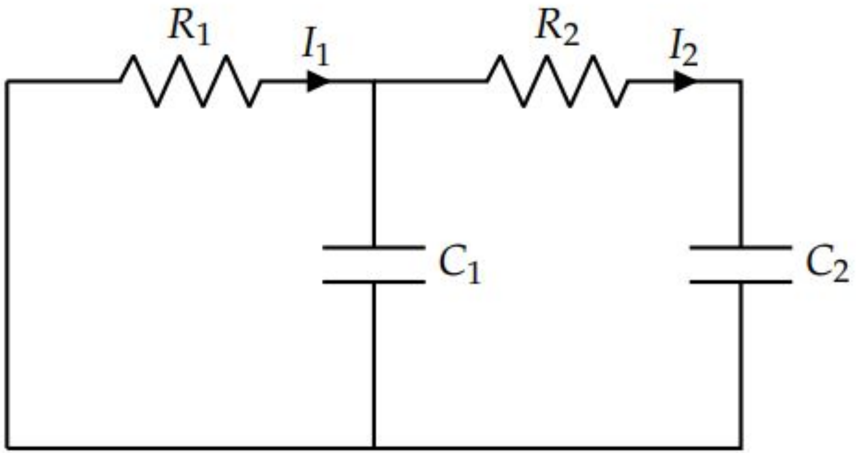
Suppose: $f(x) = 3x + 5$

True or False: The function f represents a linear transformation on x.

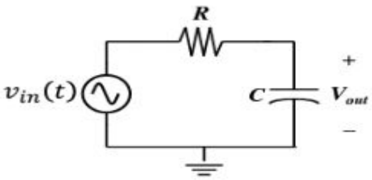
1. True
2. False

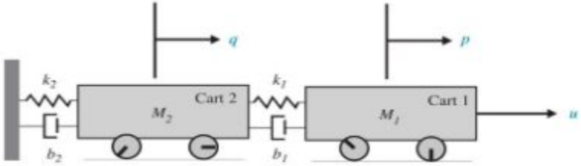
False. This represents a linear transformation plus a translation.
Mathematicians call this an “affine” transformation.

State Space Representation of Systems

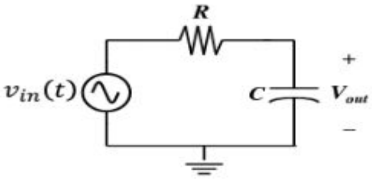


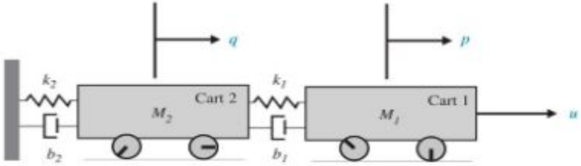
State Space Representation of Systems





State Space Representation of Systems

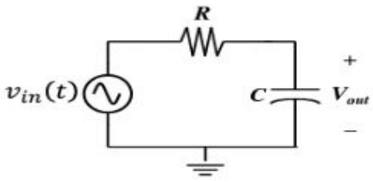


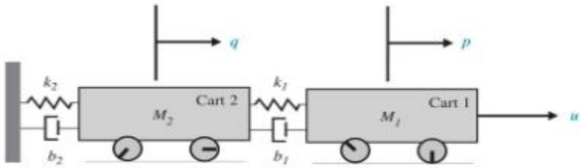


Process starts with developing a model:

1. Define inputs, outputs, state variables.

State Space Representation of Systems

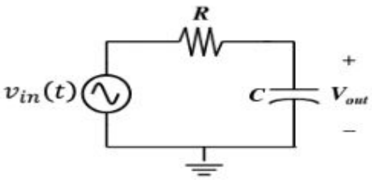


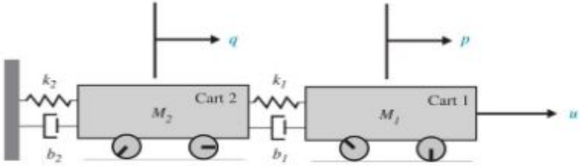


Process starts with developing a model:

1. Define inputs, outputs, state variables.
2. Write differential equations to capture system behavior.

State Space Representation of Systems





Process starts with developing a model:

1. Define inputs, outputs, state variables.
2. Write differential equations to capture system behavior.
3. Test, refine model as necessary.

State Space Representation of Systems

State Space Representation of Systems

State Variables:

- position (x,y,z)

State Space Representation of Systems

State Variables:

- position (x,y,z)
- orientation (roll, pitch, yaw)

State Space Representation of Systems

State Variables:

- position (x,y,z)
- orientation (roll, pitch, yaw)
- velocities

State Space Representation of Systems

State Variables:

- position (x,y,z)
- orientation (roll, pitch, yaw)
- velocities
- control surface positions

State Space Representation of Systems

State Variables:

- position (x,y,z)
- orientation (roll, pitch, yaw)
- velocities
- control surface positions
- etc

State Space Representation of Systems

State Variables:

- position (x,y,z)
- orientation (roll, pitch, yaw)
- velocities
- control surface positions
- etc

Inputs:

- thrust
- control surface positions

State Space Representation of Systems

State Variables:

- position (x,y,z)
- orientation (roll, pitch, yaw)
- velocities
- control surface positions
- etc

Inputs:

- thrust
- control surface positions

Outputs:

- position
- orientation
- velocities

State Space Representation / VDEs

$$\frac{d\vec{x}}{dt} = A\vec{x} + B\vec{u}$$

$$\vec{y} = C\vec{x}$$

This is what we call a vector differential equation.

State Space Representation / VDEs

$$\frac{d\vec{x}}{dt} = A\vec{x} + B\vec{u}$$

$$\vec{y} = C\vec{x}$$

This is what we call a vector differential equation.

Compare to: $\frac{dx}{dt} = \lambda x + bu(t)$

State Space Representation / VDEs

$\vec{x} \in \mathbb{R}^n$

$$\begin{aligned} \frac{d\vec{x}}{dt} &= A\vec{x} + B\vec{u} \\ \vec{y} &= C\vec{x} \end{aligned}$$

This is what we call a vector differential equation.

Compare to: $\frac{dx}{dt} = \lambda x + bu(t)$

State Space Representation / VDEs

$\vec{x} \in \mathbb{R}^n$

$A \in \mathbb{R}^{n \times n}$

$$\frac{d\vec{x}}{dt} = A\vec{x} + B\vec{u}$$

$$\vec{y} = C\vec{x}$$

This is what we call a vector differential equation.

Compare to: $\frac{dx}{dt} = \lambda x + bu(t)$

State Space Representation / VDEs

$\vec{x} \in \mathbb{R}^n$

$A \in \mathbb{R}^{n \times n}$

$\vec{u} \in \mathbb{R}^m$

$$\frac{d\vec{x}}{dt} = A\vec{x} + B\vec{u}$$

$$\vec{y} = C\vec{x}$$

This is what we call a vector differential equation.

Compare to: $\frac{dx}{dt} = \lambda x + bu(t)$

State Space Representation / VDEs

$$\vec{x} \in \mathbb{R}^n$$
$$A \in \mathbb{R}^{n \times n}$$
$$\vec{u} \in \mathbb{R}^m$$
$$B \in \mathbb{R}^{n \times m}$$

$$\begin{aligned} \frac{d\vec{x}}{dt} &= A\vec{x} + B\vec{u} \\ \vec{y} &= C\vec{x} \end{aligned}$$

This is what we call a vector differential equation.

Compare to: $\frac{dx}{dt} = \lambda x + bu(t)$

State Space Representation / VDEs

$$\begin{aligned} \frac{d\vec{x}}{dt} &= A\vec{x} + B\vec{u} \\ \vec{y} &= C\vec{x} \end{aligned}$$

$\vec{x} \in \mathbb{R}^n$

$A \in \mathbb{R}^{n \times n}$

$\vec{u} \in \mathbb{R}^m$

$B \in \mathbb{R}^{n \times m}$

$\vec{y} \in \mathbb{R}^k$

This is what we call a vector differential equation.

Compare to: $\frac{dx}{dt} = \lambda x + bu(t)$

State Space Representation / VDEs

$$\begin{aligned} \frac{d\vec{x}}{dt} &= A\vec{x} + B\vec{u} \\ \vec{y} &= C\vec{x} \end{aligned}$$

$\vec{x} \in \mathbb{R}^n$

$A \in \mathbb{R}^{n \times n}$

$\vec{u} \in \mathbb{R}^m$

$B \in \mathbb{R}^{n \times m}$

$\vec{y} \in \mathbb{R}^k$

$C \in \mathbb{R}^{k \times n}$

This is what we call a vector differential equation.

Compare to: $\frac{dx}{dt} = \lambda x + bu(t)$

State Space Representation / VDEs

$$\begin{aligned}\frac{d\vec{x}}{dt} &= A\vec{x} + B\vec{u} \\ \vec{y} &= C\vec{x}\end{aligned}$$

- Input, output, and state vectors are signals. (These are *time-varying*.)

State Space Representation / VDEs

$$\frac{d\vec{x}}{dt} = A\vec{x} + B\vec{u}$$

$$\vec{y} = C\vec{x}$$

- Input, output, and state vectors are signals. (These are *time-varying*.)
- A, B, and C matrices capture system behavior. (These are *time-invariant*.)

State Space Representation / VDEs

$$
\begin{aligned}
\frac{d\vec{x}}{dt} &= A\vec{x} + B\vec{u} \\
\vec{y} &= C\vec{x}
\end{aligned}
$$

- Input, output, and state vectors are signals. (These are *time-varying*.)
- A, B, and C matrices capture system behavior. (These are *time-invariant*.)
- We often ignore the output equation and treat the states as the outputs.

State Space Representation / VDEs

Our state-space model is Linear Time Invariant (LTI) => A, B, and C are matrices of *constants*:

$$\begin{aligned}\frac{d\vec{x}}{dt} &= A\vec{x} + B\vec{u} \\ \vec{y} &= C\vec{x}\end{aligned}$$

The time-invariance assumption is often a good one. However, almost everything is <u>non-linear</u>.

Why are we focusing on linear systems?

State Space Representation / VDEs

Our state-space model is Linear Time Invariant (LTI) => A, B, and C are matrices of *constants*:

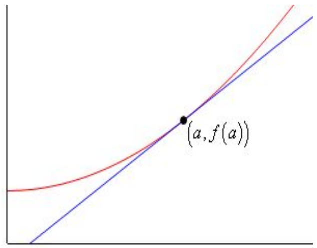
$$\frac{d\vec{x}}{dt} = A\vec{x} + B\vec{u}$$

$$\vec{y} = C\vec{x}$$

The time-invariance assumption is often a good one. However, almost everything is <u>non-linear</u>.

Why are we focusing on linear systems?

- Nonlinear systems can often be approximated using a linear model, especially locally.



State Space Representation / VDEs

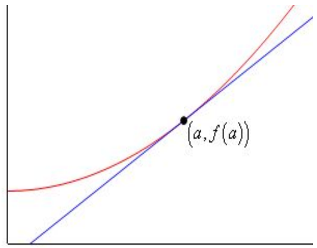
Our state-space model is Linear Time Invariant (LTI) => A, B, and C are matrices of *constants*:

$$\begin{aligned} \frac{d\vec{x}}{dt} &= A\vec{x} + B\vec{u} \\ \vec{y} &= C\vec{x} \end{aligned}$$

The time-invariance assumption is often a good one. However, almost everything is <u>non-linear</u>.

Why are we focusing on linear systems?

- Nonlinear systems can often be approximated using a linear model, especially locally.
- Linear systems are well-understood, and much easier to work with.
  - superposition
  - tools (Fourier, Laplace, Phasors, etc)



State Space Representation / VDEs

$$\frac{d\vec{x}}{dt} = A\vec{x} + B\vec{u}$$

$$\vec{y} = C\vec{x}$$

1. Given initial conditions and an input signal, can we determine the state trajectory?

State Space Representation / VDEs

$$\begin{aligned} \frac{d\vec{x}}{dt} &= A\vec{x} + B\vec{u} \\ \vec{y} &= C\vec{x} \end{aligned}$$

1. Given initial conditions and an input signal, can we determine the state trajectory?
2. **Path Planning:** Given an initial state and a desired final state, find feasible (or optimal) state trajectory.

State Space Representation / VDEs

$$\frac{d\vec{x}}{dt} = A\vec{x} + B\vec{u}$$

$$\vec{y} = C\vec{x}$$

1. Given initial conditions and an input signal, can we determine the state trajectory?
2. **Path Planning:** Given an initial state and a desired final state, find feasible (or optimal) state trajectory.
3. **Control:** Given a desired state trajectory, find inputs to achieve that trajectory.