http://elec3004.org

Signals as Vectors
Systems as Maps

ELEC 3004: **Digital Linear Dynamical Systems**: Signals & Controls

Dr. Surya Singh

Lecture 2

(Makes reference to material from EE263 and ELEC6.003)

elec3004@itee.uq.edu.au

http://robotics.itee.uq.edu.au/~elec3004/

March 9, 2015

© 2014 School of Information Technology and Electrical Engineering at The University of Queensland

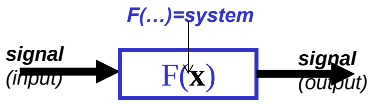
BY-NC-SA

Lecture Schedule:

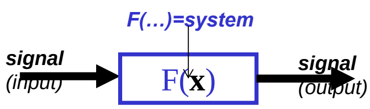
Week	Date	Lecture Title
1	2-Mar	Introduction
	3-Mar	Systems Overview
2	9-Mar	Signals as Vectors & Systems as Maps
	10-Mar	[Signals]
3	16-Mar	Sampling & Data Acquisition & Antialiasing Filters
	17-Mar	[Sampling]
4	23-Mar	System Analysis & Convolution
	24-Mar	[Convolution & FT]
5	30-Mar	Frequency Response & Filter Analysis
	31-Mar	[Filters]
6	13-Apr	Discrete Systems & Z-Transforms
	14-Apr	[Z-Transforms]
7	20-Apr	Introduction to Digital Control
	21-Apr	[Feedback]
8	27-Apr	Digital Filters
	28-Apr	[Digital Filters]
9	4-May	Digital Control Design
	5-May	[Digitial Control]
10	11-May	Stability of Digital Systems
	12-May	[Stability]
11	18-May	State-Space
	19-May	Controllability & Observability
12	25-May	PID Control & System Identification
	26-May	Digitial Control System Hardware
13	31-May	Applications in Industry & Information Theory & Communications
	2-Jun	Summary and Course Review

Signals as Vectors

- Back to the beginning!



Signals as Vectors



- There is a perfect analogy between signals and vectors …

Signals are vectors!

- A vector can be represented as a sum of its components in a variety of ways, depending upon the choice of coordinate system. A signal can also be represented as a sum of its components in a variety of ways.

Types of Linear Systems

From Last Week:

- LDS:

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

- LTI – LDS:

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

Types of Linear Systems

From Last Week:

- LDS:

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

- LTI – LDS:

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

Types of Linear Systems

From Last Week:

- LDS:

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

To Review:

- Continuous-time linear dynamical system (CT LDS):

$$\frac{dx}{dt} = A(t)x(t) + B(t)u(t), \quad y(t) = C(t)x(t) + D(t)u(t)$$

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

Types of Linear Systems

- LDS:

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

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

➔ state equations, or “*m*-input, *n*-state, *p*-output’ LDS

Types of Linear Systems

- LDS:

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

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

➔ state equations, or "*m*-input, *n*-state, *p*-output' LDS

Types of Linear Systems

- LDS:

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

- **Time-invariant:** where A(t), B(t), C(t) and D(t) are **constant**
- **Autonomous:** there is no input *u* (B,D are irrelevant)
- **No Feedthrough**: D = 0

- SISO: *u(t)* and *y(t)* are scalars
- MIMO: *u(t)* and *y(t)*: They're vectors: Big Deal ?

Discrete-time Linear Dynamical System

- Discrete-time Linear Dynamical System (DT LDS) has the form:

$$x(t+1) = A(t)x(t) + B(t)u(t), \qquad y(t) = C(t)x(t) + D(t)u(t)$$

- ***t*** ∈ ℤ denotes time index : ℤ={0, ±1, …, ± **n**}
- *x(t), u(t), y(t)* ∈ are sequences
- Differentiation handled as difference equation:
  ➔ first-order vector recursion

Signals as Vectors

- Represent them as Column Vectors

$$x = \begin{bmatrix} x[1] \\ x[2] \\ x[3] \\ \vdots \\ x[N] \end{bmatrix}.$$

Signals as Vectors

- Can represent phenomena of interest in terms of signals
- Natural vector space structure (addition/substraction/norms)
- Can use norms to describe and quantify properties of signals

Signals as vectors

Signals can take real or complex values.

In both cases, a natural vector space structure:

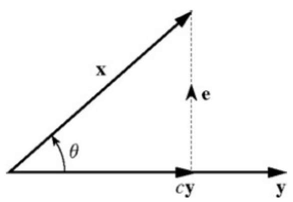
- Can add two signals: $x_1[n] + x_2[n]$
- Can multiply a signal by a scalar number: $C \cdot x[n]$
- Form linear combinations: $C_1 \cdot x_1[n] + C_2 \cdot x_2[n]$

Various Types

- Audio signal (sound pressure on microphone)
- B/W video signal (light intensity on
- photosensor)
- Voltage/current in a circuit (measure with
- multimeter)
- Car speed (from tachometer)
- Robot arm position (from rotary encoder)
- Daily prices of books / air tickets / stocks
- Hourly glucose level in blood (from glucose monitor)
- Heart rate (from heart rate sensor)

Vector Refresher

$$\mathbf{x} \cdot \mathbf{y} = |\mathbf{x}||\mathbf{y}| \cos \theta \quad (6.46)$$



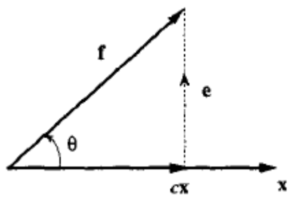
- Length: $|\mathbf{x}|^2 = \mathbf{x} \cdot \mathbf{x}$
- Decomposition: $\mathbf{x} = c_1\mathbf{y} + \mathbf{e}_1 = c_2\mathbf{y} + \mathbf{e}_2$
- Dot Product of ⊥ is 0: $\mathbf{x} \cdot \mathbf{y} = 0$

Vectors [2]

- Magnitude and Direction

$$f \cdot x = |f||x| \cos(\theta)$$

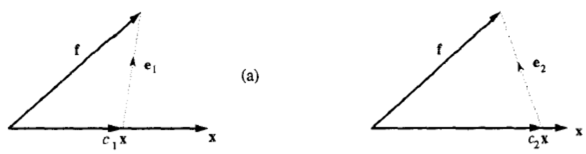
- Component (projection) of a vector along another vector



$\mathbf{f} = c\mathbf{x} + \mathbf{e}$ ← Error Vector

Vectors [3]

- ∞ bases given $\vec{\mathbf{x}}$



- Which is the best one?

$$\mathbf{f} \simeq c\mathbf{x}$$

$$c|\mathbf{x}| = |\mathbf{f}| \cos \theta$$

$$c|\mathbf{x}|^2 = |\mathbf{f}||\mathbf{x}| \cos \theta = \mathbf{f} \cdot \mathbf{x}$$

$$c = \frac{\mathbf{f} \cdot \mathbf{x}}{\mathbf{x} \cdot \mathbf{x}} = \frac{1}{|\mathbf{x}|^2} \mathbf{f} \cdot \mathbf{x}$$

$$\mathbf{f} \cdot \mathbf{x} = 0$$

- Can I allow more basis vectors than I have dimensions?

Signals **Are** Vectors

- A Vector / Signal can represent a sum of its components

  Remember (Lecture 5, Slide 10):
  Total response = Zero-input response + Zero-state response

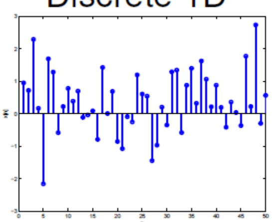
  Initial conditions | **External Input**

- Vectors are Linear
  - They have **additivity** and **homogeneity**

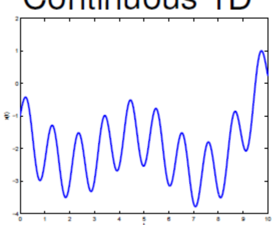
Vectors / Signals Can Be Multidimensional

- A signal is a quantity that varies as a function of an index set
- They can be multidimensional:
  - 1-dim, discrete index (time): x[n]
  - 1-dim, continuous index (time): x(t)
  - 2-dim, discrete (e.g., a B/W or RGB image): x[j; k]
  - 3-dim, video signal (e.g, video): x[j; k; n]

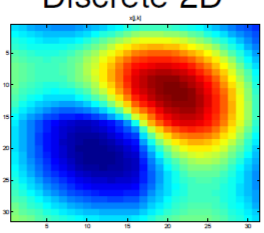
Discrete 1D



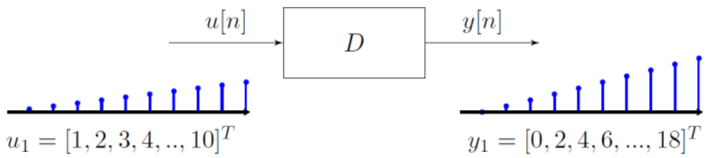
Continuous 1D



Discrete 2D



It's Just a Linear Map



- y[n]=2u[n-1] is a linear map
- BUT y[n]=2(u[n]-1) is **NOT Why?**

- **Because of homogeneity!**

  T(au)=aT(u)

Norms of signals

Can introduce a notion of signals being "nearby."

This is characterized by a metric (or distance function).

$$d(\mathbf{x}, \mathbf{y})$$

If compatible with the vector space structure, we have a norm.

$$\|\mathbf{x} - \mathbf{y}\|$$

Examples of Norms

Can use many different norms, depending on what we want to do.

The following are particularly important:

- ℓ_2 (Euclidean) norm:

$$\|x\|_2 = \left(\sum_{k=1}^{n} |x[k]|^2 \right)^{\frac{1}{2}} \qquad \texttt{norm(x,2)}$$

- ℓ_1 norm:

$$\|x\|_1 = \sum_{k=1}^{n} |x[k]| \qquad \texttt{norm(x,1)}$$

- ℓ_∞ norm:

$$\|x\|_\infty = \max_k |x[k]| \qquad \texttt{norm(x,inf)}$$

What are the differences?

Properties of norms

For any norm $\|\cdot\|$, and any signal x, we have:

1. Linearity: if C is a scalar,

$$\|C \cdot \mathrm{x}\| = |C| \cdot \|\mathrm{x}\|$$

2. Subadditivity (triangle inequality):

$$\|\mathrm{x} + \mathrm{y}\| \leq \|\mathrm{x}\| + \|\mathrm{y}\|$$

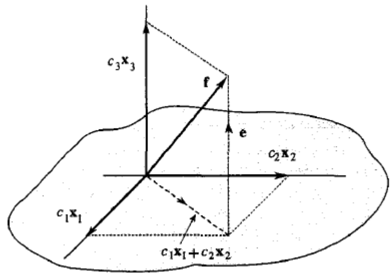
Can use norms:

- To detect whether a signal is (approximately) zero.
- To compare two signals, and determine if they are "close."

$$\|\mathrm{x} - \mathrm{y}\| \approx 0$$

Signal representation by Orthogonal Signal Set

- **Orthogonal Vector Space**



➔ A signal may be thought of as having components.

Component of a Signal

$$f(t) \simeq cx(t) \qquad t_1 \le t \le t_2$$

$$c = \frac{\int_{t_1}^{t_2} f(t)x(t)\,dt}{\int_{t_1}^{t_2} x^2(t)\,dt} = \frac{1}{E_x}\int_{t_1}^{t_2} f(t)x(t)\,dt$$

$$\int_{t_1}^{t_2} f(t)x(t)\,dt = 0$$

- Let’s take an example:

$$f(t) \simeq c\sin t \qquad 0 \le t \le 2\pi$$

$$x(t) = \sin t \quad \text{and} \quad E_x = \int_0^{2\pi} \sin^2(t)\,dt = \pi$$

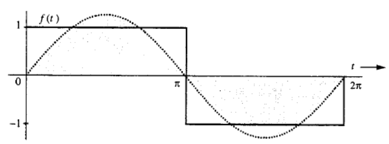


Fig. 3.3 Approximation of square signal in terms of a single sinusoid.

Thus

$$f(t) \simeq \frac{4}{\pi}\sin t \qquad (3.14)$$

Basis Spaces of a Signal

$$\int_{t_1}^{t_2} x_m(t)x_n(t)\,dt = \begin{cases} 0 & m \neq n \\ E_n & m = n \end{cases}$$

$$f(t) \simeq c_1x_1(t) + c_2x_2(t) + \cdots + c_Nx_N(t) = \sum_{n=1}^{N} c_nx_n(t)$$

$$e(t) = f(t) - \sum_{n=1}^{N} c_nx_n(t)$$

$$c_n = \frac{\int_{t_1}^{t_2} f(t)x_n(t)\,dt}{\int_{t_1}^{t_2} x_n^{\,2}(t)\,dt} = \frac{1}{E_n}\int_{t_1}^{t_2} f(t)x_n(t)\,dt \qquad n = 1, 2, \ldots, N$$

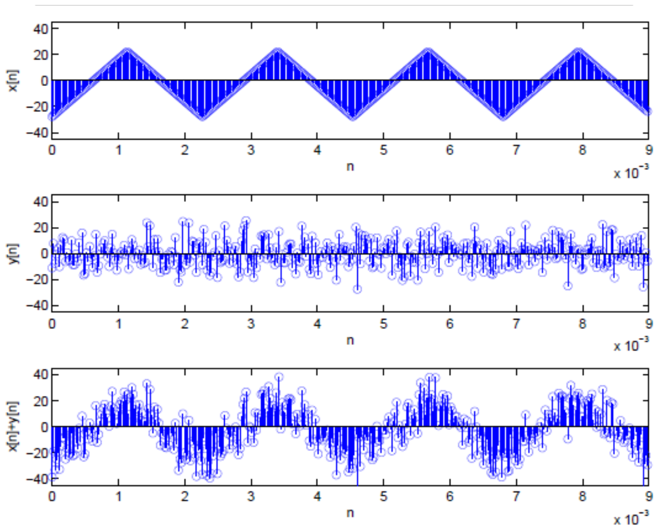
$$f(t) = c_1x_1(t) + c_2x_2(t) + \cdots + c_nx_n(t) + \cdots = \sum_{n=1}^{\infty} c_nx_n(t) \qquad t_1 \leq t \leq t_2$$

Basis Spaces of a Signal

$$f(t) = c_1x_1(t) + c_2x_2(t) + \cdots + c_nx_n(t) + \cdots = \sum_{n=1}^{\infty} c_nx_n(t) \qquad t_1 \leq t \leq t_2$$

- Observe that the error energy *Ee* generally decreases as *N*, the number of terms, is increased because the term *Ck* 2 *Ek* is nonnegative. Hence, it is possible that the error energy -> 0 as *N* -> 00. When this happens, the orthogonal signal set is said to be complete.
- In this case, it's no more an approximation but an equality

Linear combinations of signals



Application Example: Active Noise Cancellation

A "noise" signal, that we want to get rid of.

- At subject location, signal is

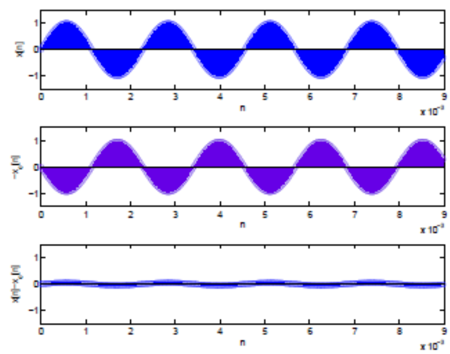
$$x[n]$$

- Microphone picks up signal

$$x_c[n]$$

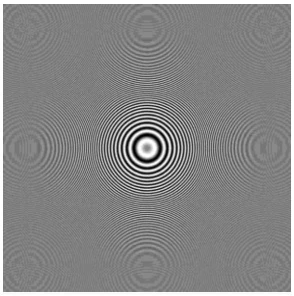
- Subtract the two signals:

$$y(t) = x(t) - x_c(t)$$



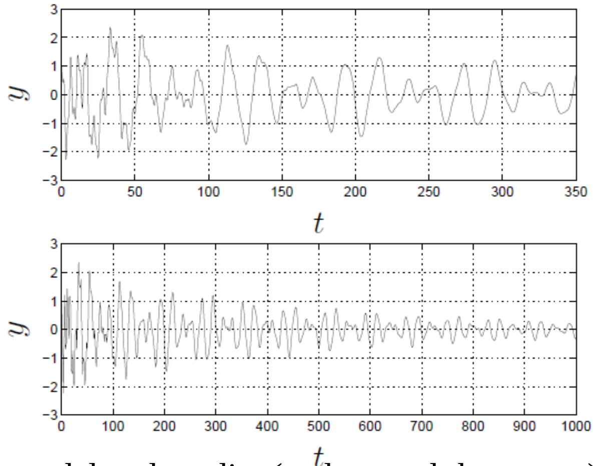
Notice careful synchronization is needed!

Where are we going with this?



This can help simplify matters… An Example

Consider the following system:

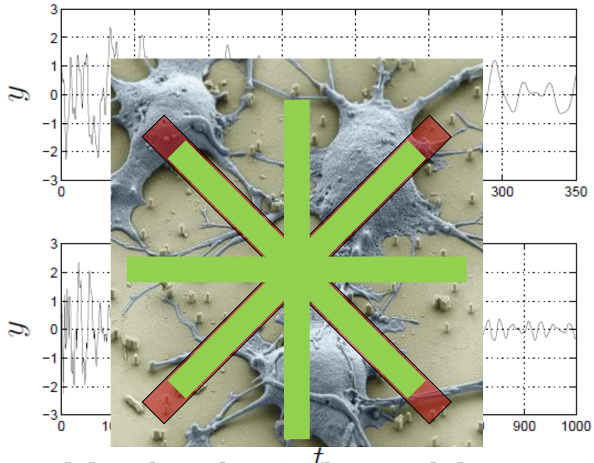


- How to model and predict (and control the output)?

Source: EE263 (s.1-13)

This can help simplify matters… An Example

Consider the following system:



- How to model and predict (and control the output)?

Source: EE263 (s.1-13)

This can help simplify matters… An Example

- Consider the following system:

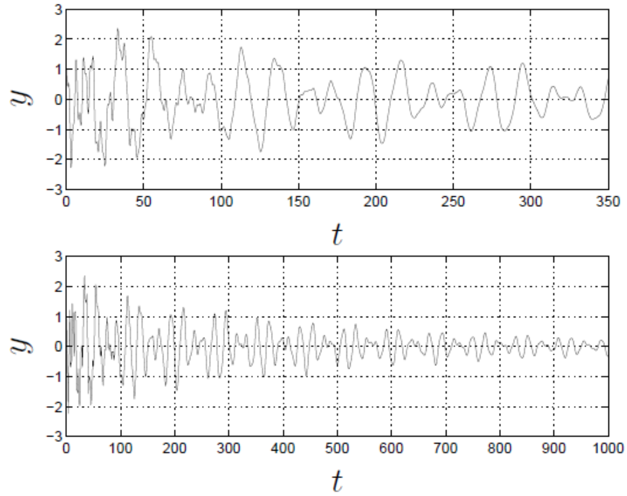
$$\dot{x} = Ax, \qquad y = Cx$$

- $x(t) \in \mathbb{R}^8$, $y(t) \in \mathbb{R}^1$ → 8-state, single-output system
- Autonomous: No input yet! ($u(t) = 0$)

Source: EE263 (s.1-13)

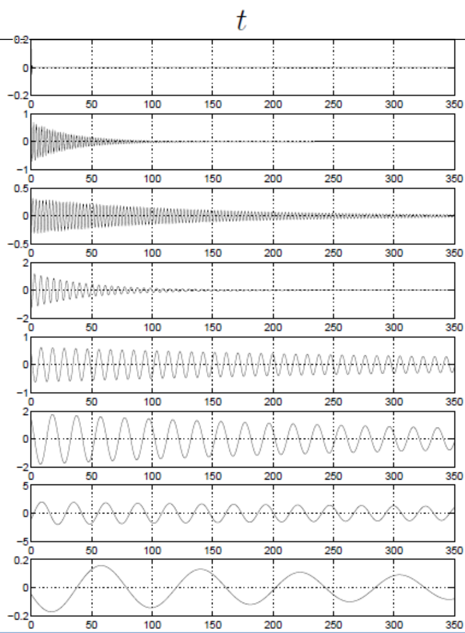
This can help simplify matters… An Example

- Consider the following system:



Source: EE263 (s.1-13)

This can help simplify matters… An Example



Source: EE263 (s.1-13)

Example: Let's consider the control…

Expand the system to have a control input…

- $B \in \mathbb{R}^{8\times2}$, $C \in \mathbb{R}^{2\times8}$ (note: the 2nd dimension of C)

$$\dot{x} = Ax + Bu, \quad y = Cx, \quad x(0) = 0$$

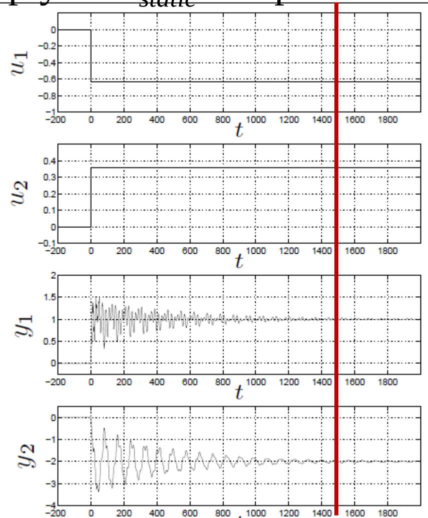
- Problem: Find **u** <u>such that</u> $y_{des}(t)$**=(1,-2)**
- A simple (and rational) approach:
  - solve the above equation!
  - Assume: static conditions (u, x, y constant)

$$\dot{x} = 0 = Ax + Bu_{\text{static}}, \quad y = y_{\text{des}} = Cx$$

➔ Solve for u:

$$u_{\text{static}} = \left(-CA^{-1}B\right)^{-1} y_{\text{des}} = \begin{bmatrix} -0.63 \\ 0.36 \end{bmatrix}$$

Example: Apply $\boldsymbol{u}=u_{static}$ and presto!

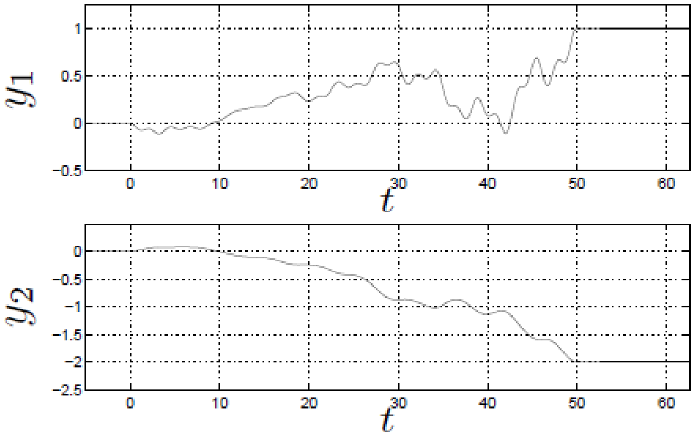


- Note: It takes 1500 seconds for the y(t) to converge … but that's natural … can we do better?

Source: EE263 (s.1-13)

Example: Yes we can!

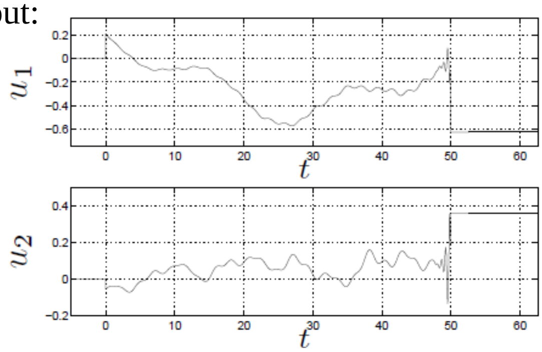
- How about:



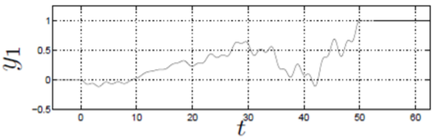
Source: EE263 (s.1-13)

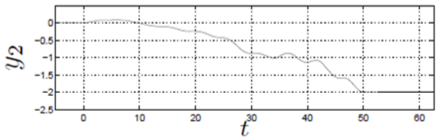
Example: **How?** How about a more clever input?

- How about:



➔

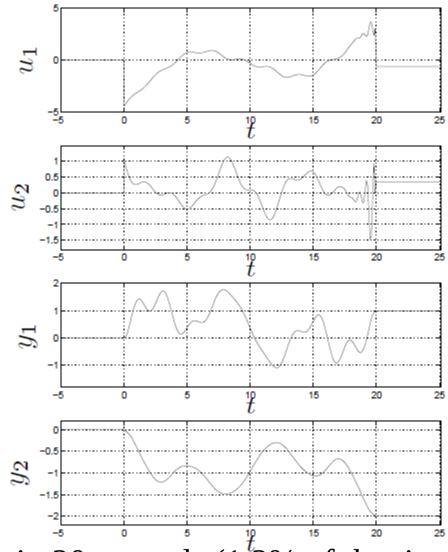




- Converges in 50 seconds (3.3% of the time ☺)

Source: EE263 (s.1-13)

Example: Can we beat it? Larger inputs & LDS



Source: EE263 (s.1-13)

- Converges in 20 seconds (1.3% of the time ☺)

Dynamical Systems…

- A system with a memory
  - Where past history (or derivative states) are **relevant** in determining the response
- Ex:
  - RC circuit: Dynamical
    - Clearly a function of the "capacitor's past" (initial state) and
    - Time! (charge / discharge)
  - R circuit: is memoryless ∵ the output of the system (recall V=IR) at some time **t** only depends on the input at time **t**

- Lumped/Distributed
  - Lumped: Parameter is constant through the process & can be treated as a "point" in space
- Distributed: System dimensions ≠ small over signal
  - Ex: waveguides, antennas, microwave tubes, etc.

Causality:
Causal (physical or nonanticipative) systems

- Is one for which the output at any instant t_0 depends only on the value of the input x(t) for $t \le t_0$. Ex:

$u(t) = x(t-2) \Rightarrow \text{causal}$ $u(t) = x(t-2) + x(t+2) \Rightarrow \text{noncausal}$

- A "*real-time*" system must be causals
  - How can it respond to future inputs?
- A prophetic system: knows future inputs and acts on it (now)
  - The output would begin before t_0
- In some cases Noncausal maybe modelled as causal with delay
- Noncausal systems provide **an upper bound** on the performance of causal systems

Causality:
Looking at this from the output's perspective…

- **Causal** = The output *before* some time *t* does not depend on the input *after* time *t.*

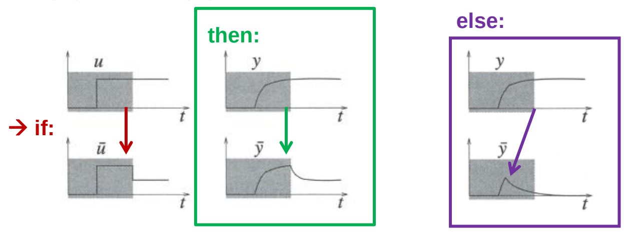
Given: $y(t) = F(u(t))$

For:

$\hat{u}(t) = u(t), \forall 0 \le t < T \; or \; [0, T)$

Then for a T>0:

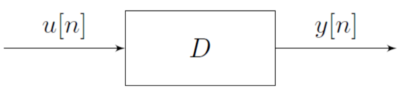
$\rightarrow \hat{y}(t) = y(t), \; \forall 0 \le t < T$



Causal Noncausal

Systems as Maps

Then a System is a **MATRIX**

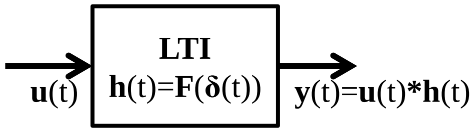


$$y = Du.$$

$$\begin{bmatrix} y[1] \\ y[2] \\ \vdots \\ y[M] \end{bmatrix} = \begin{bmatrix} D_{11} & D12 & \cdots & D_{1N} \\ D_{21} & D_{22} & \cdots & D_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ D_{M1} & D_{M2} & \cdots & D_{MN} \end{bmatrix} \begin{bmatrix} u[1] \\ u[2] \\ \vdots \\ u[N] \end{bmatrix}.$$

$$y[i] = \sum_j D_{ij} u[j].$$

Linear Time Invariant



- Linear & Time-invariant (of course - tautology!)
- Impulse response: **h**(t)=**F**(**δ**(t))
- Why?
  - Since it is linear the output response (**y**) to any input (**x**) is:

$$x(t) = \int_{-\infty}^{\infty} x(\tau)\,\delta(t-\tau)\,d\tau$$
$$y(t) = F\left[\int_{-\infty}^{\infty} x(\tau)\,\delta(t-\tau)\,d\tau\right] \overset{linear}{\to} \int_{-\infty}^{\infty} x(\tau)\,F[\delta(t-\tau)]\,d\tau$$
$$h(t-\tau) \overset{TI}{\equiv} F[\delta(t-\tau)]$$
$$\Rightarrow y(t) = \int_{-\infty}^{\infty} x(\tau)\,h(t-\tau)\,d\tau = x(t) * h(t)$$

- The output of any continuous-time LTI system is the **convolution** of input **u**(t) with the impulse response **F**(**δ**(t)) of the system.

Linear Dynamic [Differential] System

≡ LTI systems for which the input & output are linear ODEs

$$a_0 y + a_1\frac{dy}{dt} + \cdots + a_n\frac{d^n y}{dt^n} = b_0 x + b_1\frac{dx}{dt} + \cdots + b_m\frac{d^m x}{dt^m}$$

Laplace:

$$a_0 Y(s) + a_1 sY(s) + \cdots + a_n s^n Y(s) = b_0 X(s) + b_1 sX(s) + \cdots + b_m s^m X(s)$$
$$A(s)Y(s) = B(s)X(s)$$

- Total response = Zero-input response + Zero-state response

  Zero-input response: **Initial conditions**; Zero-state response: **External Input**

Linear Systems and ODE’s

- Linear system described by differential equation

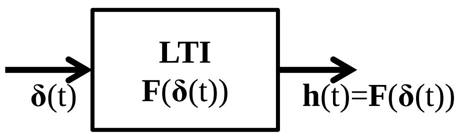
$$a_0 y + a_1 \frac{dy}{dt} + \cdots + a_n \frac{d^n y}{dt^n} = b_0 x + b_1 \frac{dx}{dt} + \cdots + b_m \frac{d^m x}{dt^m}$$

- Which using Laplace Transforms can be written as

$$a_0 Y(s) + a_1 sY(s) + \cdots + a_n s^n Y(s) = b_0 X(s) + b_1 sX(s) + \cdots + b_m s^m X(s)$$
$$A(s)Y(s) = B(s)X(s)$$

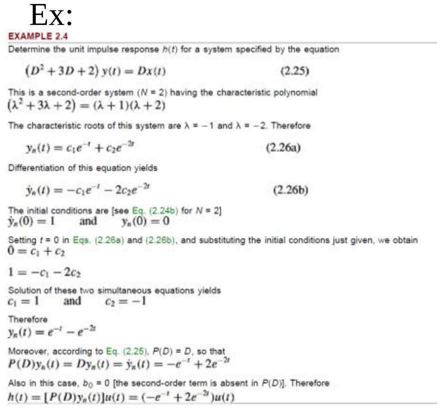
where *A(s)* and *B(s)* are polynomials in *s*

Unit Impulse Response



- **δ**(t): Impulsive excitation
- h(t): characteristic mode terms

Ex:


EXAMPLE 2.4

Determine the unit impulse response $h(t)$ for a system specified by the equation

$$(D^2 + 3D + 2)\,y(t) = Dx(t) \qquad (2.25)$$

This is a second-order system ($N = 2$) having the characteristic polynomial
$(\lambda^2 + 3\lambda + 2) = (\lambda + 1)(\lambda + 2)$

The characteristic roots of this system are $\lambda = -1$ and $\lambda = -2$. Therefore

$$y_n(t) = c_1 e^{-t} + c_2 e^{-2t} \qquad (2.26a)$$

Differentiation of this equation yields

$$\dot{y}_n(t) = -c_1 e^{-t} - 2c_2 e^{-2t} \qquad (2.26b)$$

The initial conditions are [see Eq. (2.24b) for $N = 2$]
$\dot{y}_n(0) = 1$ and $y_n(0) = 0$

Setting $t = 0$ in Eqs. (2.26a) and (2.26b), and substituting the initial conditions just given, we obtain
$0 = c_1 + c_2$

$1 = -c_1 - 2c_2$

Solution of these two simultaneous equations yields
$c_1 = 1$ and $c_2 = -1$

Therefore
$y_n(t) = e^{-t} - e^{-2t}$

Moreover, according to Eq. (2.25), $P(D) = D$, so that
$P(D)y_n(t) = Dy_n(t) = \dot{y}_n(t) = -e^{-t} + 2e^{-2t}$

Also in this case, $b_0 = 0$ [the second-order term is absent in $P(D)$]. Therefore
$h(t) = [P(D)y_n(t)]u(t) = (-e^{-t} + 2e^{-2t})u(t)$

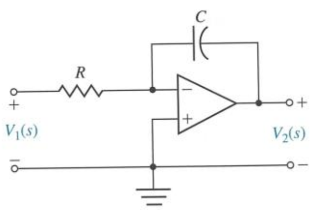
System Models

- Various things – all the same!

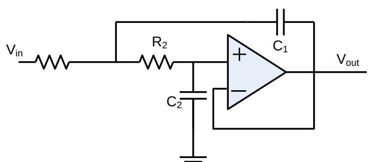
Table 2.1 Summary of Through- and Across-Variables for Physical Systems

System	Variable Through Element	Integrated Through-Variable	Variable Across Element	Integrated Across-Variable
Electrical	Current, i	Charge, q	Voltage difference, v_{21}	Flux linkage, λ_{21}
Mechanical translational	Force, F	Translational momentum, P	Velocity difference, v_{21}	Displacement difference, y_{21}
Mechanical rotational	Torque, T	Angular momentum, h	Angular velocity difference, ω_{21}	Angular displacement difference, θ_{21}
Fluid	Fluid volumetric rate of flow, Q	Volume, V	Pressure difference, P_{21}	Pressure momentum, γ_{21}
Thermal	Heat flow rate, q	Heat energy, H	Temperature difference, $\mathcal{T}_{21}$	

Circuits



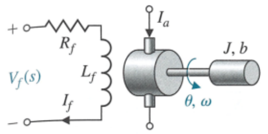
$$\frac{V_2(s)}{V_1(s)} = -\frac{1}{RCs}$$



$$\frac{v_{out}}{v_{in}} = \frac{1}{C_1C_2R_1R_2s^2+C_2(R_1+R_2)s+1}$$

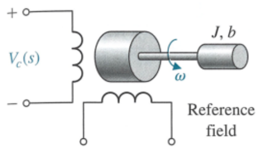
Motors

5. DC motor, field-controlled, rotational actuator



$$\frac{\theta(s)}{V_f(s)} = \frac{K_m}{s(Js + b)(L_f s + R_f)}$$

7. AC motor, two-phase control field, rotational actuator



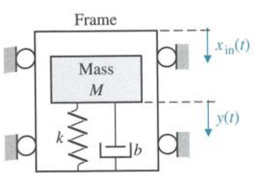
$$\frac{\theta(s)}{V_c(s)} = \frac{K_m}{s(\tau s + 1)}$$

$$\tau = J/(b - m)$$

m = slope of linearized torque-speed curve (normally negative)

Mechanical Systems

15. Accelerometer, acceleration sensor



$$x_o(t) = y(t) - x_{in}(t),$$

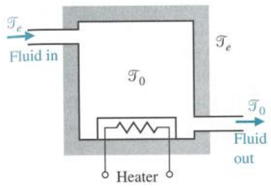
$$\frac{X_o(s)}{X_{in}(s)} = \frac{-s^2}{s^2 + (b/M)s + k/M}$$

For low-frequency oscillations, where $\omega < \omega_n$,

$$\frac{X_o(j\omega)}{X_{in}(j\omega)} \simeq \frac{\omega^2}{k/M}$$

Thermal Systems

16. Thermal heating system



$$\frac{\mathcal{T}(s)}{q(s)} = \frac{1}{C_t s + (QS + 1/R_t)},$$ where

$\mathcal{T} = \mathcal{T}_o - \mathcal{T}_e$ = temperature difference due to thermal process

C_t = thermal capacitance

Q = fluid flow rate = constant

S = specific heat of water

R_t = thermal resistance of insulation

$q(s)$ = transform of rate of heat flow of heating element

First Order Systems

First order systems

$$ay' + by = 0 \quad (\text{with } a \neq 0)$$

righthand side is zero:

- called *autonomous system*
- solution is called *natural* or *unforced response*

can be expressed as

$$Ty' + y = 0 \quad \text{or} \quad y' + ry = 0$$

where

- $T = a/b$ is a *time* (units: seconds)
- $r = b/a = 1/T$ is a *rate* (units: 1/sec)

First Order Systems

Solution by Laplace transform

take Laplace transform of $Ty' + y = 0$ to get

$$T\underbrace{(sY(s) - y(0))}_{\mathcal{L}(y')} + Y(s) = 0$$

solve for $Y(s)$ (algebra!)

$$Y(s) = \frac{Ty(0)}{sT + 1} = \frac{y(0)}{s + 1/T}$$

and so $y(t) = y(0)e^{-t/T}$

First Order Systems

solution of $Ty' + y = 0$: $y(t) = y(0)e^{-t/T}$

if $T > 0$, y *decays* exponentially

- T gives time to decay by $e^{-1} \approx 0.37$
- $0.693T$ gives time to decay by half $(0.693 = \log 2)$
- $4.6T$ gives time to decay by 0.01 $(4.6 = \log 100)$

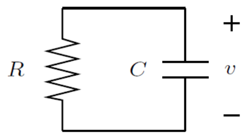
if $T < 0$, y *grows* exponentially

- $|T|$ gives time to grow by $e \approx 2.72$;
- $0.693|T|$ gives time to double
- $4.6|T|$ gives time to grow by 100

First Order Systems

Examples

simple RC circuit:



circuit equation: $RCv' + v = 0$

solution: $v(t) = v(0)e^{-t/(RC)}$

population dynamics:

- $y(t)$ is population of some bacteria at time t
- growth (or decay if negative) rate is $y' = by - dy$ where b is birth rate, d is death rate
- $y(t) = y(0)e^{(b-d)t}$ (grows if $b > d$; decays if $b < d$)

Second Order Systems

Second order systems

$$ay'' + by' + cy = 0$$

assume $a > 0$ (otherwise multiply equation by -1)

solution by Laplace transform:

$$a(\underbrace{s^2Y(s) - sy(0) - y'(0)}_{\mathcal{L}(y'')}) + b(\underbrace{sY(s) - y(0)}_{\mathcal{L}(y')}) + cY(s) = 0$$

solve for Y (just algebra!)

$$Y(s) = \frac{asy(0) + ay'(0) + by(0)}{as^2 + bs + c} = \frac{\alpha s + \beta}{as^2 + bs + c}$$

where $\alpha = ay(0)$ and $\beta = ay'(0) + by(0)$

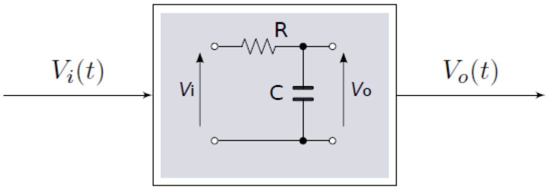
Second Order Systems

so solution of $ay'' + by' + cy = 0$ is

$$y(t) = \mathcal{L}^{-1}\left(\frac{\alpha s + \beta}{as^2 + bs + c}\right)$$

- $\chi(s) = as^2 + bs + c$ is called *characteristic polynomial* of the system
- form of $y = \mathcal{L}^{-1}(Y)$ depends on roots of characteristic polynomial χ
- coefficients of numerator $\alpha s + \beta$ come from initial conditions

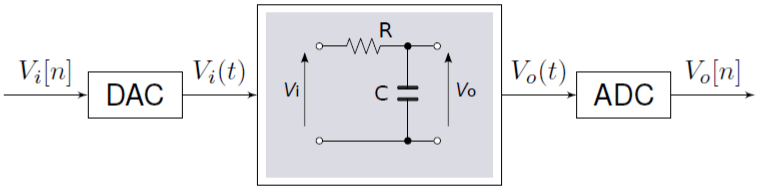
Example: Speaking of Circuits



$$C\frac{dV_o(t)}{dt} = \frac{V_i(t) - V_o(t)}{R}.$$

Source: ELEC6.003 (s.3-42)

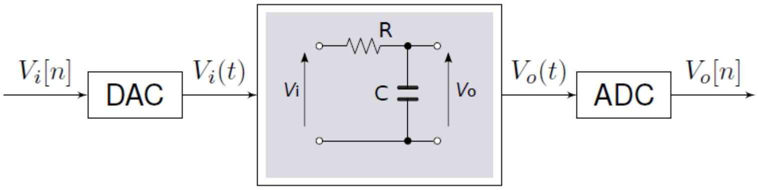
What about the DIGITAL case?



- Is it still linear?

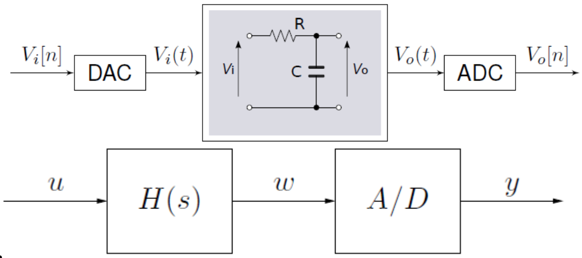
Source: ELEC6.003 (s.3-46)

What about the DIGITAL case?



- Can LDS help do better than quantization?

What about the DIGITAL case?

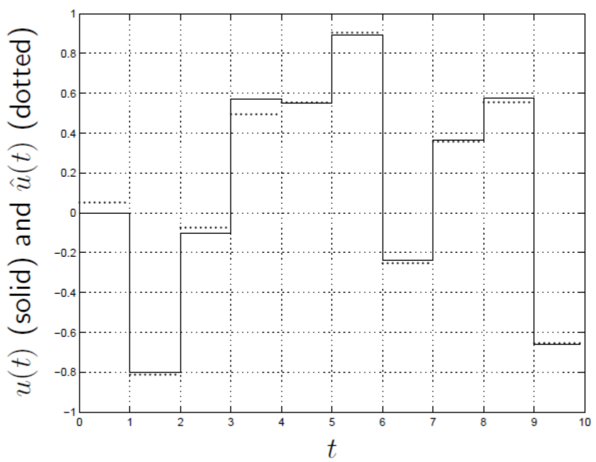


➔

- Problem:
  Estimate signal ***u***, given quantized, filtered signal ***y***
- Some solutions:
  - ignore quantization
  - design equalizer *G(s)* for *H(s)* (i.e., *GH* ≅ 1)
  - approximate *u* as *G(s)y*
  - ➔ Pose as an estimation problem

Source: EE263 (s.1-124)

What about the DIGITAL case?

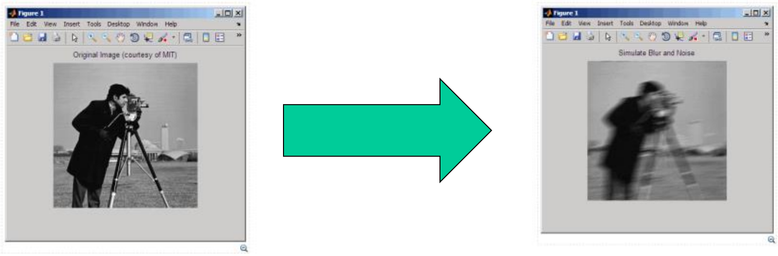


- RMS error 0.03, well below quantization error (!)

Source: EE263 (s.1-124)

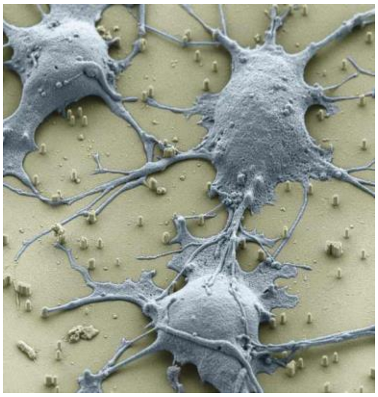
Ex: Deblurring

perfect image → [Moving Camera Optics] → blurry image

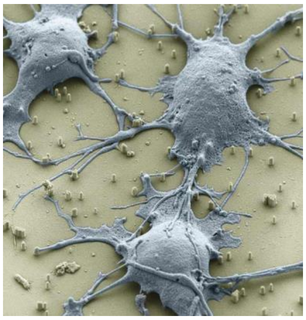


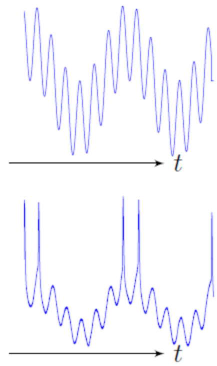
- Matlab: **deconvwnr**

What about …



What about …





- For small current inputs, neuron membrane potential output response is surprisingly **linear**.
- Though this has limits …
  neurons "spike" are (quite) nonlinear (truly)

Source: ELEC6.003 (s.3-49)

Next Time…

- **Sampling**
  - Measurements at regular intervals of a continuous signal
  - Not to be confused with
    " How to try regional dishes without indigestion"
- Review:
  - Chapter 8 of Lathi
- Send (and you shall receive) a positive signal ☺