http://elec3004.com

Introduction to State-Space

ELEC 3004: **Systems**: Signals & Controls
Dr. Surya Singh

Lecture 18

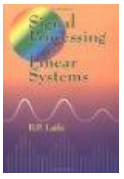
elec3004@itee.uq.edu.au
http://robotics.itee.uq.edu.au/~elec3004/

May 8, 2019

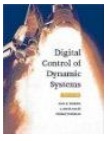
Lecture Schedule:

Week	Date	Lecture Title
1	27-Feb	Introduction
	1-Mar	Systems Overview
2	6-Mar	Systems as Maps & Signals as Vectors
	8-Mar	Systems: Linear Differential Systems
3	13-Mar	Sampling Theory & Data Acquisition
	15-Mar	Aliasing & Antialiasing
4	20-Mar	Discrete Time Analysis & Z-Transform
	22-Mar	Second Order LTID (& Convolution Review)
5	27-Mar	Frequency Response
	29-Mar	Filter Analysis
6	3-Apr	Digital Filters (IIR) & Filter Analysis
	5-Apr	PS 1: Q & A
7	10-Apr	Digital Windows
	12-Apr	Digital Filter (FIR)
8	17-Apr	Active Filters & Estimation
	19-Apr	
	24-Apr	**Holiday**
	26-Apr	
9	1-May	Introduction to Feedback Control
	3-May	Servoregulation & PID Control
10	**8-May**	**State-Space Control**
	10-May	Advanced PID
11	15-May	Guest Lecture: FFT
	17-May	State Space Control System Design
12	22-May	Digital Control Design
	24-May	Shaping the Dynamic Response
13	29-May	System Identification & Information Theory & Information Space
	31-May	Summary and Course Review

Follow Along Reading:



B. P. Lathi
Signal processing and linear systems
1998
TK5102.9.L38 1998



G. Franklin, J. Powell, M. Workman
Digital Control of Dynamic Systems
1990

TJ216.F72 1990
[Available as UQ Ebook]

Today

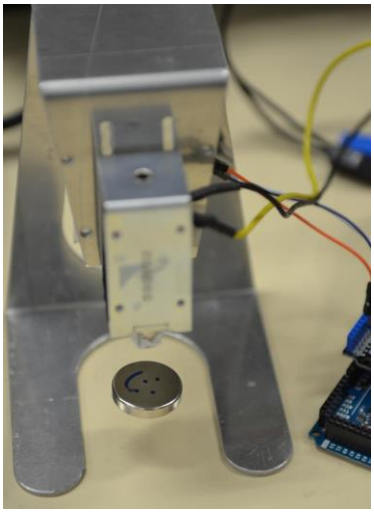
➔ **State-space** ➔ **[A stately idea! ☺]**

- FPW
  - Chapter 4: Discrete Equivalents to Continuous
  - Transfer Functions: The Digital Filter

- Lathi Ch. 13
  - § 13.2 Systematic Procedure for Determining State Equations
  - § 13.3 Solution of State Equations

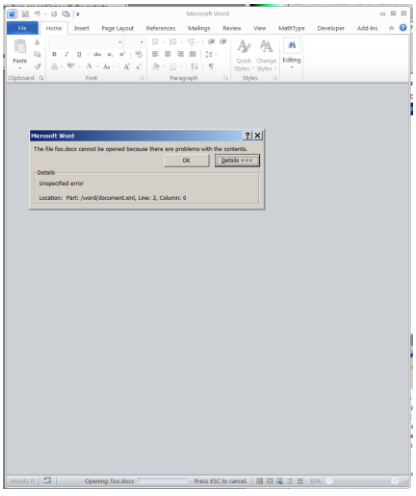
Next Time

NEXT WEEK: Lab 4 – LeviLab II:



- AKA "Revenge of the TUNING!"

Friendly computing tale…



- Please save (as) often ☺
- Use Platypus$_2$ (**Cloud-based** ☺)

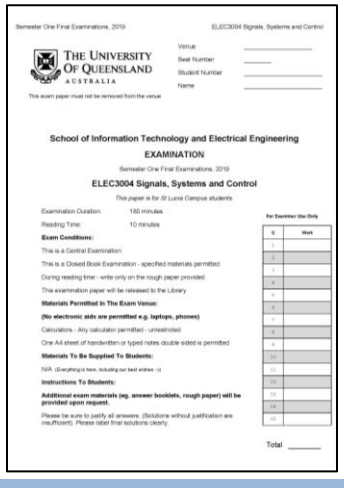
Final Exam: Room Update (49-301 or 9-217)

Exam Code	Exam Title	Campus Code	Date	Time	Notes	Student Split	Venue Description
ELEC3004	Signals, Systems and Control	STLUC	10/06/2019	2:30 PM		Ainslie, Lewis - Varun, Kumar	Advanced Engineering Building (49) - Room 301
ELEC3004	Signals, Systems and Control	STLUC	10/06/2019	2:30 PM		**Vecchi**, Thoma - Zemek, Reid	Michie Building (9) - Room 217
ELEC7312	Signals, Systems and Control	STLUC	10/06/2019	2:30 PM		All Students	Michie Building (9) - Room 217

- **Date: June 10**
- **Time: 2:30 pm**
- Room: See Above

- Venue may change – Please see notifications box on https://my.uq.edu.au/ for updates

- *Thank You!*


Semester One Final Examinations, 2019 ELEC3004 Signals, Systems and Control

THE UNIVERSITY OF QUEENSLAND AUSTRALIA

This exam paper must not be removed from the venue

Venue
Seat Number
Student Number
Name

School of Information Technology and Electrical Engineering

EXAMINATION

Semester One Final Examinations, 2019

ELEC3004 Signals, Systems and Control

This paper is for St Lucia Campus students

Examination Duration: 180 minutes
Reading Time: 10 minutes

For Examiner Use Only

Q	Mark
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	
Total	

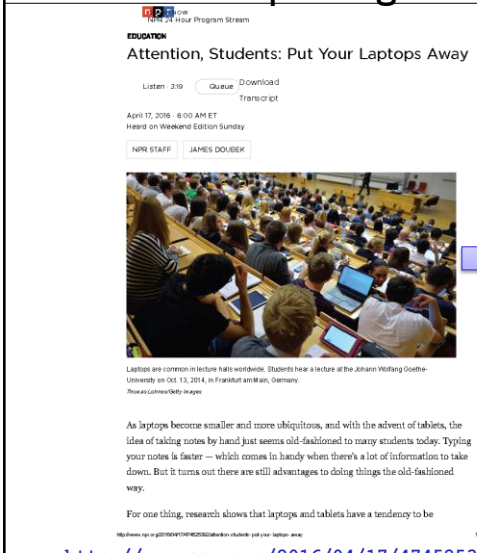
Exam Conditions:
This is a Central Examination
This is a Closed Book Examination - specified materials permitted
During reading time - write only on the rough paper provided
This examination paper will be released to the Library

Materials Permitted In The Exam Venue:
(No electronic aids are permitted e.g. laptops, phones)
Calculators - Any calculator permitted - unrestricted
One A4 sheet of handwritten or typed notes double sided is permitted

Materials To Be Supplied To Students:
N/A (Everything is here, including our best wishes :-))

Instructions To Students:
Additional exam materials (eg. answer booklets, rough paper) will be provided upon request.
Please be sure to justify all answers. (Solutions without justification are insufficient). Please label final solutions clearly.

Final Exam Tip: Longhand Notes > Typed



EDUCATION

Attention, Students: Put Your Laptops Away

Listen 3:19 Queue Download Transcript

April 17, 2016 · 6:00 AM ET
Heard on Weekend Edition Sunday

NPR STAFF JAMES DOUBEK

Laptops are common in lecture halls worldwide. Students hear a lecture at the Johann Wolfang Goethe-University on Oct. 13, 2014, in Frankfurt am Main, Germany.
Thomas Lohnes/Getty Images

As laptops become smaller and more ubiquitous, and with the advent of tablets, the idea of taking notes by hand just seems old-fashioned to many students today. Typing your notes is faster — which comes in handy when there's a lot of information to take down. But it turns out there are still advantages to doing things the old-fashioned way.

For one thing, research shows that laptops and tablets have a tendency to be

Psychological Science OnlineFirst, published on May 22, 2014 as doi:10.1177/0956797614524581

Research Article

The Pen Is Mightier Than the Keyboard: Advantages of Longhand Over Laptop Note Taking

Pam A. Mueller[1] and Daniel M. Oppenheimer[2]
[1]Princeton University and [2]University of California, Los Angeles

Abstract

Taking notes on laptops rather than in longhand is increasingly common. Many researchers have suggested that laptop note taking is less effective than longhand note taking for learning. Prior studies have primarily focused on students' capacity for multitasking and distraction when using laptops. The present research suggests that even when laptops are used solely to take notes, they may still be impairing learning because their use results in shallower processing. In three studies, we found that students who took notes on laptops performed worse on conceptual questions than students who took notes longhand. We show that whereas taking more notes can be beneficial, laptop note takers' tendency to transcribe lectures verbatim rather than processing information and reframing it in their own words is detrimental to learning.

Keywords

academic achievement, cognitive processes, memory, educational psychology, open data, open materials

Received 5/31/13; Revision accepted 1/16/14

The use of laptops in classrooms is controversial. Many professors believe that computers (and the Internet) serve as distractions, detracting from class discussion and student learning (e.g., Yamamoto, 2007). Conversely, students often self-report a belief that laptops in class are beneficial (e.g., Barak, Lipson, & Lerman, 2006; Mitra & Steffensmeier, 2000; Skolnick & Prato, 2008). Even when students admit that laptops are a distraction, they believe the benefits outweigh the costs (Kay & Lauricella, 2011). Empirical research tends to support the professors' view, finding that students using laptops are not on task during lectures (Kay & Lauricella, 2011; Kraushaar & Novak, 2010; Skolnick & Prato, 2008; Sovern, 2013), show decreased academic performance (Fried, 2008; Grace-Martin & Gay, 2001; Kraushaar & Novak, 2010), and are actually less satisfied with their education than their peers who do not use laptops in class (Wurst, Smarkola, & Gaffney, 2008).

These correlational studies have focused on the capacity of laptops to distract and to invite multitasking. Experimental tests of immediate retention of class material have also found that Internet browsing impairs performance (Hembrooke & Gay, 2003). These findings are important but relatively unsurprising, given the literature on decrements in performance when multitasking or task switching (e.g., Iqbal & Horvitz, 2007; Rubinstein, Meyer, & Evans, 2001).

However, even when distractions are controlled for, laptop use might impair performance by affecting the manner and quality of in-class note taking. There is a substantial literature on the general effectiveness of note taking in educational settings, but it mostly predates laptop use in classrooms. Prior research has focused on two ways in which note taking can affect learning: encoding and external storage (see DiVesta & Gray, 1972; Kiewra, 1989). The encoding hypothesis suggests that the processing that occurs during the act of note taking improves learning and retention. The external-storage hypothesis touts the benefits of the ability to review material (even from notes taken by someone else). These two theories are not incompatible; students who both take and review

Corresponding Author:
Pam A. Mueller, Princeton University, Psychology Department, Princeton, NJ 08544
E-mail: pamuelle@princeton.edu

- http://www.npr.org/2016/04/17/474525392/attention-students-put-your-laptops-away
- doi: 10.1177/0956797614524581

PID Recap

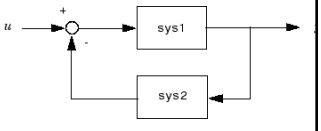
Effects of Increasing Gain

Effects of *increasing* a parameter independently

Parameter	Rise time	Overshoot	Settling time	Steady-state error	Stability[11]
K_p	Decrease	Increase	Small change	Decrease	Degrade
K_i	Decrease	Increase	Increase	Eliminate	Degrade
K_d	Minor change	Decrease	Decrease	No effect in theory	Improve if K_d small

Matlab helps with PID tuning:

```
G_system=[???]; H=[1];
D_compensator = pidtune(G_system, 'PIDF')
CL_system = feedback(series(D_compensator,G_system), H)
step(CL_system)
```



When Can PID Control Be Used?

When:

- "Industrial processes" such that the demands on the performance of the control are not too high.
  - Control authority/actuation
  - Fast (clean) sensing
- PI: Most common
  - All stable processes can be controlled by a PI law (modest performance)
  - First order dynamics

PID (PI + Derivative):

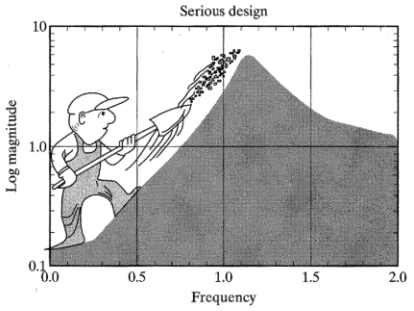
- Second order
  (A double integrator cannot be controlled by PI)
- Speed up response
  When time constants differ in magnitude
  (Thermal Systems)

Something More Sophisticated:

- Large time delays
- Oscillatory modes between inertia and compliances

Seeing PID – No Free Lunch

- The energy (and sensitivity) moves around (in this case in "frequency")



- Sensitivity reduction at low frequency unavoidably leads to sensitivity increase at higher frequencies.

Source: Gunter Stein's interpretation of the water bed effect – G. Stein, *IEEE Control Systems Magazine*, 2003.

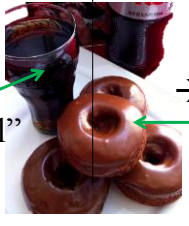
Another way to see P I | D

- Derivative

  D provides:
  - High sensitivity
  - Responds to change
  - Adds "damping" & ∴ permits larger K_P
  - Noise sensitive
  - Not used alone (∵ its on rate change of error – by itself it wouldn't get there)

→ "**D**iet Coke of control"

- Integral
  - Eliminates offsets (makes regulation ☺)
  - Leads to Oscillatory behaviour
  - Adds an "order" but instability (Makes a 2nd order system 3rd order)



→ "**I**nteresting cake of control"

State-Space

Or more aptly…

Welcome to

State-Space!

(It be stated -- Hallelujah !)

- More general mathematical model
  - MIMO, time-varying, nonlinear
- Matrix notation (think LAPACK ➔ MATLAB)
- Good for discrete systems
- More design tools!

State-Space Control

$$\dot{\mathbf{x}} = \mathbf{F}\mathbf{x}$$

(That can not be all of it? There has to be more to it than this…)

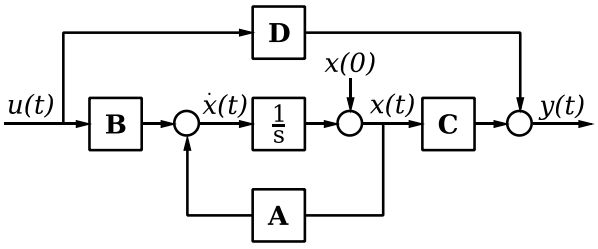
State-Space Control

$$\dot{\mathbf{x}} = \mathbf{F}\mathbf{x} + \mathbf{G}\mathbf{u}$$

Benefits:

- Characterises the process by systems of coupled, first-order differential equations
- More general mathematical model
  - MIMO, time-varying, nonlinear
- Mathematically esoteric (who needs practical solutions)
- Yet, well suited for digital computer implementation
  - That is: based on vectors/matrices (think LAPACK ➔ MATLAB)

State-Space Terminology



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

LTI State-Space

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

- **If** the system is **linear and time invariant**, **then** A,B,C,D are constant coefficient

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

Discrete Time State-Space

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

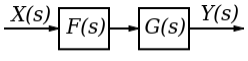
- **If** the system is **discrete**,
  then x and u are given by difference equations

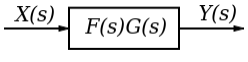
$$\rightarrow x[k+1] = A[k]\,x[k] + B[k]\,u[k]$$
$$y[k] = C[k]\,x[k] + D[k]\,u[k]$$

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

Block Diagram Algebra in State Space

- Series:





$$\begin{bmatrix} x'_G \\ x'_F \end{bmatrix} = \begin{bmatrix} A_G & B_G C_F \\ 0 & A_F \end{bmatrix} \begin{bmatrix} x_G \\ x_F \end{bmatrix} + \begin{bmatrix} B_G D_F \\ B_F \end{bmatrix} u$$

$$\begin{bmatrix} y_G \\ y_F \end{bmatrix} = \begin{bmatrix} C_G & D_G C_F \\ 0 & C_F \end{bmatrix} \begin{bmatrix} x_G \\ x_F \end{bmatrix} + \begin{bmatrix} D_G D_F \\ D_F \end{bmatrix} u$$

System 1:

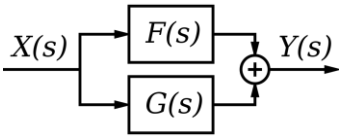
$$x'_F = A_F x_F + B_F u$$
$$y_F = C_F x_F + D_F u$$

System 2:

$$x'_G = A_G x_G + B_G y_F$$
$$y_G = C_G x_G + D_G y_F$$

Block Diagram Algebra in State Space

- Parallel:



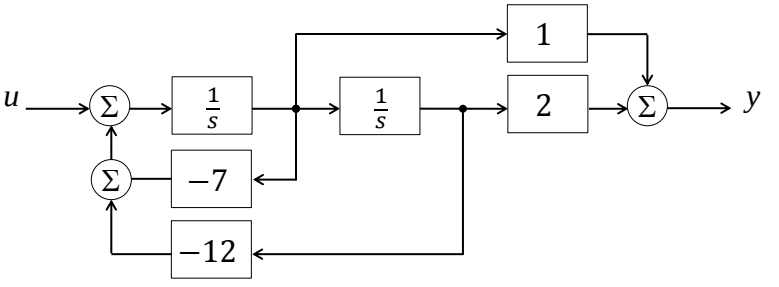
$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} u$$

$$y = \begin{bmatrix} C_1 & C_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + (D_1 + D_2)u$$

Introduction to state-space

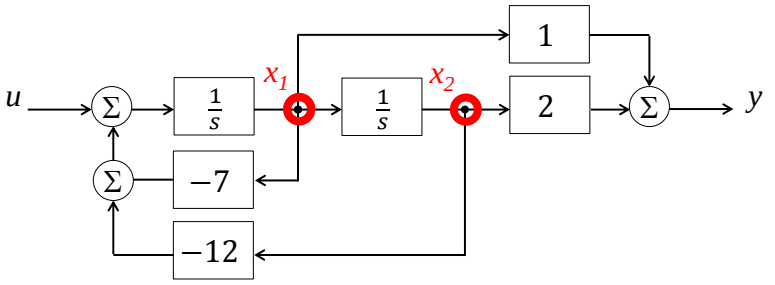
- Linear systems can be written as networks of simple dynamic elements:

$$H = \frac{s+2}{s^2+7s+12} = \frac{2}{s+4} + \frac{-1}{s+3}$$



Introduction to state-space

- We can identify the nodes in the system
  - These nodes contain the integrated time-history values of the system response
  - We call them "states"



Linear system equations

- We can represent the dynamic relationship between the states with a linear system:

$$\begin{aligned} \dot{x_1} &= -7x_1 - 12x_2 + u \\ \dot{x_2} &= x_1 + 0x_2 + 0u \\ y &= x_1 + 2x_2 + 0u \end{aligned}$$

State-space representation

- We can write linear systems in matrix form:

$$\dot{\boldsymbol{x}} = \begin{bmatrix} -7 & 12 \\ 1 & 0 \end{bmatrix} \boldsymbol{x} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u$$
$$\boldsymbol{y} = [1 \quad 2]\boldsymbol{x} + 0u$$

Or, more generally:

$$\dot{\boldsymbol{x}} = \mathbf{A}\boldsymbol{x} + \mathbf{B}u$$
$$y = \mathbf{C}\boldsymbol{x} + Du$$

"State-space equations"

State evolution

- Consider the system matrix relation:

$$\dot{\boldsymbol{x}} = \mathbf{F}\boldsymbol{x} + \mathbf{G}u$$
$$y = \mathbf{H}\boldsymbol{x} + Ju$$

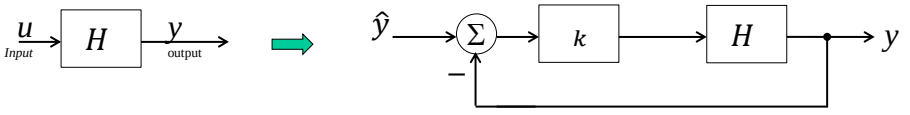
The time solution of this system is:

$$\boldsymbol{x}(t) = e^{\mathbf{F}(t-t_0)}\,\boldsymbol{x}(t_0) + \int_{t_0}^{t} = e^{\mathbf{F}(t-\tau)}\,\mathbf{G}u(\tau)d\tau$$

If you didn't know, the matrix exponential is:

$$e^{\mathbf{K}t} = \mathbf{I} + \mathbf{K}t + \frac{1}{2!}\mathbf{K}^2t^2 + \frac{1}{3!}\mathbf{K}^3t^3 + \cdots$$

Difference Equations & Feedback



- Start with the Open-Loop:

$$y = kHu$$

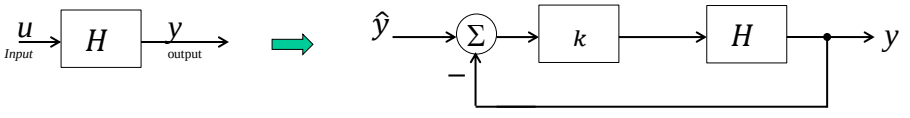
- Close the loop:

$$u = ke = k(\hat{y} - y) \rightarrow y = H[k(\hat{y} - y)]$$

$$\rightarrow \mathrm{y} = \frac{Hk}{1+Hk}\hat{y}$$

- All easy! (yesa!)

Difference Equations & Feedback



- Now add delay (image the plant is a replica with a delay τ)

$$y(t) = u(t - \tau)$$

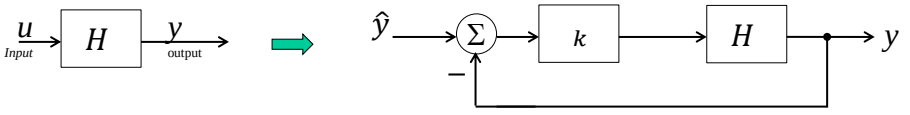
- Close the loop:

$$u(t - \tau) = ke(t - \tau) = k\,[\hat{y}(t - \tau) - \mathrm{y}(t - \tau)]$$

$$\rightarrow \mathrm{y(t)} = k\,[\hat{y}(t - \tau) - \mathrm{y}(t - \tau)]$$

- Notice we have a difference equation!

Difference Equations & Feedback



- What happens with a single delay and a unit step?

$$u(t) = k \text{ for } 0<t<\tau$$

$$y(t) = u(t-\tau) \text{ for } \tau<t<2\tau$$

- Then with feedback we get:

$$u(t) = k(1-k) = k - k^2$$

$$y(t) = k - k^2 + k^3 + \cdots + (-1)^{n-1}k^{n-1}$$

- If k<1: then:

$$\rightarrow \lim y(t) = \frac{k}{1+k}$$

Break ☺

Solving State Space

Solving State Space…

- Recall:

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

- For Linear Systems:

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

$$y(t) = C(t)\,x(t) + D(t)\,u(t)$$

- For LTI:

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

$$\rightarrow y = Cx + Du$$

➔ Solutions to State Equations

$$\dot{x} = Ax + Bu$$
$$sX(s) - x(0) = AX(s) + BU(s)$$
$$X(s) = (sI - A)^{-1}x(0) + (sI - A)^{-1}BU(s)$$

$$X(s) = \mathcal{L}[e^{At}]x(0) + \mathcal{L}[e^{At}]BU(s)$$

$$x(t) = \int_0^t e^{At}\, Bu(\tau)d\tau$$

$$\Rightarrow\ e^{At}$$

➔ State-Transition Matrix Φ

- $\Phi(t) = e^{At} = \mathcal{L}^{-1}[(sI - A)^{-1}]$

- It contains all the information about the free motions of the system described by $\dot{x} = Ax$

LTI Properties:

- $\Phi(0) = e^{0t} = I$
- $\Phi^{-1}(t) = \Phi(-t)$
- $\Phi(t_1 + t_2) = \Phi(t_1)\Phi(t_2) = \Phi(t_2)\Phi(t_1)$
- $[\Phi(t)]^n = \Phi(nt)$

➔ The closed-loop poles are the eignvalues of the system matrix

Great, so how about control?

- Given $\dot{\boldsymbol{x}} = \mathbf{F}\boldsymbol{x} + \mathbf{G}u$, if we know $\mathbf{F}$ and $\mathbf{G}$, we can design a controller $u = -\mathbf{K}\boldsymbol{x}$ such that

$$\text{eig}(\mathbf{F} - \mathbf{GK}) < 0$$

- In fact, if we have full measurement and control of the states of $\boldsymbol{x}$, we can position the poles of the system in arbitrary locations!

Digital State Space:

- Difference equations in state-space form:

$$x[n+1] = Ax[n] + Bu[n]$$
$$y[n] = Cx[n] + Du[n]$$

- Where:
  - u[n], y[n]: input & output (scalars)
  - x[n]: state vector

Digital Control Law Design

In Chapter 2, we saw that the state-space description of a continuous system is given by (2.43),

$$\dot{\mathbf{x}} = \mathbf{F}\mathbf{x} + \mathbf{G}u, \tag{6.1}$$

and (2.44),

$$y = \mathbf{H}\mathbf{x}. \tag{6.2}$$

We assume the control is applied from the computer by a ZOH as shown in Fig. 1.1. Therefore, (6.1) and (6.2) have an exact discrete representation as given by (2.57),

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{\Phi}\mathbf{x}(k) + \mathbf{\Gamma}u(k), \\ y(k) &= \mathbf{H}\mathbf{x}(k), \end{aligned} \tag{6.3}$$

where

$$\mathbf{\Phi} = e^{\mathbf{F}T}, \tag{6.4a}$$

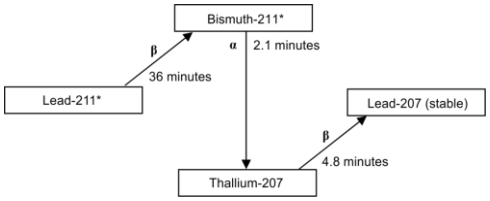
$$\mathbf{\Gamma} = \int_0^T e^{\mathbf{F}\eta} d\eta \mathbf{G}, \tag{6.4b}$$

A Systematic Procedure for Determining State Eqs.

1. Choose all independent capacitor voltages and inductor currents to be the state variables.
2. Choose a set of loop currents; express the state variables and their first derivatives in terms of these loop currents.
3. Write the loop equations and eliminate all variables other than state variables (and their first derivatives) from the equations derived in Steps 2 and 3.

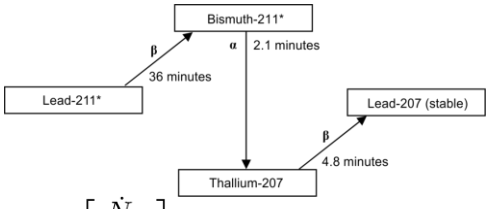
See also: Lathi § 13.2-1 (p. 788)

An Example



- $\frac{dN1(t)}{dt} = -\lambda_1 N1(t)$
- $\frac{dN2(t)}{dt} = -\lambda_2 N2(t) + \lambda_1 N1(t)$
- $\frac{dN3(t)}{dt} = -\lambda_3 N3(t) + \lambda_2 N2(t)$
- $\frac{dN4(t)}{dt} = \lambda_3 N3(t)$

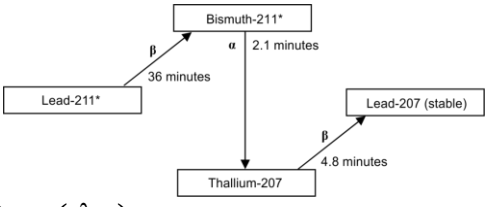
An Example



$$X = \begin{bmatrix} N_1 \\ N_2 \\ N_3 \\ N_4 \end{bmatrix} \rightarrow \dot{X} = \begin{bmatrix} \dot{N}_1 \\ \dot{N}_2 \\ \dot{N}_3 \\ \dot{N}_4 \end{bmatrix}$$

$$\dot{X} = FX \rightarrow \begin{bmatrix} \dot{N}_1 \\ \dot{N}_2 \\ \dot{N}_3 \\ \dot{N}_4 \end{bmatrix} = \begin{bmatrix} -\lambda_1 & 0 & 0 & 0 \\ \lambda_1 & -\lambda_2 & 0 & 0 \\ 0 & \lambda_2 & -\lambda_3 & 0 \\ 0 & 0 & \lambda_3 & 0 \end{bmatrix} \begin{bmatrix} N_1 \\ N_2 \\ N_3 \\ N_4 \end{bmatrix}$$

An Example

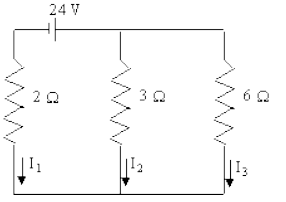


- $N_1(t)=N_1(0)\exp(-\lambda_1 t)$
- $N2(t) = N2(0)exp(-\lambda_2 t) - N1(0)\frac{\lambda_1}{\lambda_2-\lambda_1}(exp(-\lambda_2 t) - exp(-\lambda_1 t))$
- $N3(t) = \lambda_1\lambda_2 N1(0)\left[\frac{exp(-\lambda_1 t)}{(\lambda_2-\lambda_1)(\lambda_3-\lambda_1)} + \frac{exp(-\lambda_2 t)}{(\lambda_1-\lambda_2)(\lambda_3-\lambda_2)} + \frac{exp(-\lambda_3 t)}{(\lambda_1-\lambda_3)(\lambda_2-\lambda_3)}\right]$
- $N4(t) = \lambda_1\lambda_2\lambda_3 N1(0)\left[\frac{exp(-\lambda_1 t)}{(\lambda_2-\lambda_1)(\lambda_3-\lambda_1)(-\lambda_1)} + \frac{exp(-\lambda_2 t)}{(\lambda_1-\lambda_2)(\lambda_3-\lambda_2)(-\lambda_2)} + \frac{exp(-\lambda_3 t)}{(\lambda_1-\lambda_3)(\lambda_2-\lambda_3)(-\lambda_3)} + \frac{1}{(\lambda_1\lambda_2\lambda_3)}\right]$

Another Ex: Linear Algebra & KVL!

We can write this as:

$$\begin{pmatrix} 1 & 1 & 1 \\ -2 & 3 & 0 \\ 0 & -3 & 6 \end{pmatrix} \begin{pmatrix} I_1 \\ I_2 \\ I_3 \end{pmatrix} = \begin{pmatrix} 0 \\ 24 \\ 0 \end{pmatrix}$$



So we have:

$$\begin{pmatrix} I_1 \\ I_2 \\ I_3 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 \\ -2 & 3 & 0 \\ 0 & -3 & 6 \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ 24 \\ 0 \end{pmatrix}$$

Using a computer algebra system to perform the inverse and multiply by the constant matrix, we get:

$I_1 = -6$ A

$I_2 = 4$ A

$I_3 = 2$ A

We observe that I_1 is negative, as expected from the circuit diagram.

Source: http://www.intmath.com/matrices-determinants/6-matrices-linear-equations.php

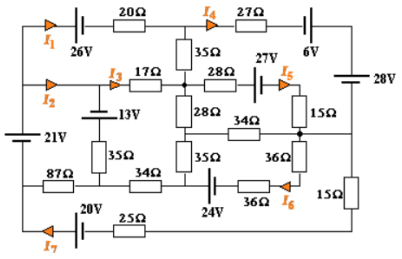
Another Example: Linear Algebra & KCL!

We solve this using a computer as follows. We just write the coefficient matrix on the left, find the inverse (raise the matrix to the power -1) and multiply the result by the constant matrix.

You can use Matlab, Mathcad or similar math software to do this. Wolfram|Alpha is a free alternative.

$$X = \begin{bmatrix} 72 & 0 & -17 & -35 & 0 & 0 & 0 \\ 0 & 122 & -35 & 0 & 0 & 0 & -87 \\ 0 & -87 & -34 & 0 & 0 & -72 & 233 \\ -17 & -35 & 149 & 0 & -28 & -35 & -34 \\ 0 & 0 & -28 & -43 & 105 & -34 & 0 \\ 0 & 0 & -35 & 0 & -34 & 141 & -72 \\ -35 & 0 & 0 & 105 & -43 & 0 & 0 \end{bmatrix}^{-1} \begin{bmatrix} -26 \\ 34 \\ -4 \\ -13 \\ -27 \\ 24 \\ 5 \end{bmatrix}$$

$$= \begin{bmatrix} -0.46801 \\ 0.42932 \\ 5.193 \times 10^{-3} \\ -0.22243 \\ -0.27848 \\ 0.21115 \\ 0.20914 \end{bmatrix}$$



Source: http://www.intmath.com/matrices-determinants/6-matrices-linear-equations.php

Discretisation (FPW!)

- We can use the time-domain representation to produce difference equations!

$$\boldsymbol{x}(kT+T) = e^{\mathbf{F}T}\,\boldsymbol{x}(kT) + \int_{kT}^{kT+T} e^{\mathbf{F}(kT+T-\tau)}\mathbf{G}u(\tau)d\tau$$

Notice $\boldsymbol{u}(\tau)$ is not based on a discrete ZOH input, but rather an integrated time-series.

We can structure this by using the form:

$$u(\tau) = u(kT), \qquad kT \le \tau \le kT+T$$

State-space z-transform

We can apply the z-transform to our system:

$$(z\mathbf{I} - \mathbf{\Phi})\boldsymbol{X}(z) = \mathbf{\Gamma}U(k)$$
$$Y(z) = \mathbf{H}\boldsymbol{X}(z)$$

which yields the transfer function:

$$\frac{Y(z)}{\boldsymbol{X}(z)} = G(z) = \mathbf{H}(z\mathbf{I} - \mathbf{\Phi})^{-1}\mathbf{\Gamma}$$

Ex: State-space control design

- Design for discrete state-space systems is just like the continuous case.
  - Apply linear state-variable feedback:

$$u = -\mathbf{K}\boldsymbol{x}$$

such that $\det(z\mathbf{I} - \mathbf{\Phi} + \mathbf{\Gamma}\mathbf{K}) = \alpha_c(z)$

where $\alpha_c(z)$ is the desired control characteristic equation

Predictably, this requires the system controllability matrix

$\mathcal{C} = [\mathbf{\Gamma} \quad \mathbf{\Phi}\mathbf{\Gamma} \quad \mathbf{\Phi}^2\mathbf{\Gamma} \quad \cdots \quad \mathbf{\Phi}^{n-1}\mathbf{\Gamma}]$ to be full-rank.

Example: PID control

- Consider a system parameterised by three states:
  - x_1, x_2, x_3
  - where $x_2 = \dot{x}_1$ and $x_3 = \dot{x}_2$

$$\dot{\boldsymbol{x}} = \begin{bmatrix} 1 & & \\ & 1 & \\ & & -2 \end{bmatrix} \boldsymbol{x} - \mathbf{K}u$$
$$y = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix} \boldsymbol{x} + 0u$$

x_2 is the output state of the system;
x_1 is the value of the integral;
x_3 is the velocity.

Example: PID control [2]

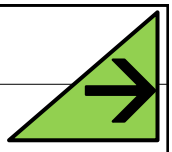
- We can choose **K** to move the eigenvalues of the system as desired:

$$\det \begin{bmatrix} 1 - K_1 & & \\ & 1 - K_2 & \\ & & -2 - K_3 \end{bmatrix} = \mathbf{0}$$

All of these eigenvalues must be positive.

It's straightforward to see how adding derivative gain K_3 can stabilise the system.

Next Time…



- **Digital Feedback Control**

- Review:
  - Chapter 2 of FPW

- More Pondering??