Observability, Controllability & Stability of Digital Systems

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

Lecture 20
(with material from FPW and Lathi)

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

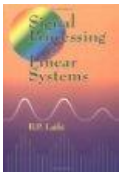
May 19, 2016

http://elec3004.com

Lecture Schedule:

Week	Date	Lecture Title
1	29-Feb	Introduction
	3-Mar	Systems Overview
2	7-Mar	Systems as Maps & Signals as Vectors
	10-Mar	Data Acquisition & Sampling
3	14-Mar	Sampling Theory
	17-Mar	Antialiasing Filters
4	21-Mar	Discrete System Analysis
	24-Mar	Convolution Review
	28-Mar	Holiday
	31-Mar	
5	4-Apr	Frequency Response & Filter Analysis
	7-Apr	Filters
6	11-Apr	Digital Filters
	14-Apr	Digital Filters
7	18-Apr	Digital Windows
	21-Apr	FFT
8	25-Apr	Holiday
	28-Apr	Introduction to Feedback Control
9	3-May	Holiday
	5-May	Feedback Control & Regulation
10	9-May	Servoregulation/PID
	12-May	Introduction to (Digital) Control
11	16-May	Digital Control Design & State-Space
	19-May	Observability, Controllability & Stability of Digital Systems
12	23-May	Digital Control Systems: Shaping the Dynamic Response & Estimation
	26-May	Applications in Industry
13	30-May	System Identification & Information Theory
	2-Jun	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 ←

- FPW
  - Ch. 5: Transfer Functions: The Digital Filter
- Lathi Ch. 13
  - § 13.2 Systematic Procedure for Determining State Equations
  - § 13.3 Solution of State Equations

- FPW
  - Chapter 6 - Design of Digital Control Systems Using State-Space Methods

Next Time

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)

Digital State Space Extended Version

Solving State Space (Extended Version)…

- Recall:

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

- For Linear Systems:

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

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

- For LTI:

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

$$\rightarrow y = Cx + Du$$

Solving State Space

- In the conventional, frequency-domain approach the differential equations are converted to transfer functions as soon as possible
  - The dynamics of a system comprising several subsystems is obtained by combining the transfer functions!

- With the state-space methods, on the other hand, the description of the system dynamics in the form of differential equations is retained throughout the analysis and design.

State-transition matrix Φ(t)

- Describes how the state *x(t)* of the system at some time *t* evolves into (or from) the state x(τ*)* at some other time *T*.

$$x(t) = \Phi\,(t, \tau)\, x\,(\tau)$$

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

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

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

Discretisation FTW!

- 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:

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

State-space control design -- Controllability

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

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

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

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

Predictably, this requires the system controllability matrix $\mathcal{C} = [\boldsymbol{\Gamma} \quad \boldsymbol{\Phi\Gamma} \quad \boldsymbol{\Phi}^2\boldsymbol{\Gamma} \quad \cdots \quad \boldsymbol{\Phi}^{n-1}\boldsymbol{\Gamma}]$ to be full-rank.

Φ: Solving State Space

- In the conventional, frequency-domain approach the differential equations are converted to transfer functions as soon as possible
  - The dynamics of a system comprising several subsystems is obtained by combining the transfer functions!

- With the state-space methods, on the other hand, the description of the system dynamics in the form of differential equations is retained throughout the analysis and design.

State-transition matrix Φ(t)

- Describes how the state *x(t)* of the system at some time *t* evolves into (or from) the state x(τ) at some other time *T*.

$$x(t) = \Phi\left(t, \tau\right) x\left(\tau\right)$$

- $\mathbf{\Phi}(\mathrm{s}) = [\mathrm{s}\mathbf{I} - \mathbf{A}]^{-1}$ ➔ $\Phi(t) = e^{\boldsymbol{At}}$

- Matrix Exponential:

$$e^{At} = \exp(At) = I + At + \frac{A^2t^2}{2!} + \cdots + \frac{A^kt^k}{k!} + \cdots$$

- Similar idea, but different result, for the control **u** ➔ Γ

Γ: Gamma: Comes from Integrating $\dot{x}$

- $\Gamma = \left(\sum_{k=0}^{\infty} \frac{A^kT^{k+1}}{(k+1)!}\right) TB \approx \left(IT + A\frac{T^2}{2}\right) B$

Why?

- $x(t) = e^{A(t-t_0)}x(t_0) + \int_0^t e^{A(t-\tau)}Bu(\tau)\, d\tau$
- $x(kT+T) = e^{AT}x(kT) + \int_{kT}^{kT+T} e^{A(kt+t-\tau)} Bu(\tau)d\tau$
- u(t) is specified in terms of a continuous time history, though we often assume u(t) is a ZOH:
- $u(\tau) = u(kT) \Rightarrow$ Introduce $\eta = kT + T - \tau$
- $x(kT+T) = e^{AT}x(kT) + \int_{kT}^{kT+T} e^{A\eta}\, d\eta Bu(kT)$

➔$\Phi = e^{AT}$, Γ= $\int_0^T e^{A\eta}\, d\eta B$

Solving State Space (optional notes) …

Time-invariant dynamics The simplest form of the general differential equation of the form (3.1) is the "homogeneous," i.e., unforced equation

$$\dot{x} = Ax \tag{3.2}$$

where A is a constant k by k matrix. The solution to (3.2) can be expressed as

$$x(t) = e^{At}c \tag{3.3}$$

where e^{At} is the matrix exponential function

$$e^{At} = I + At + A^2\frac{t^2}{2} + A^3\frac{t^3}{3!} + \cdots \tag{3.4}$$

and c is a suitably chosen constant vector. To verify (3.3) calculate the derivative of $x(t)$

$$\frac{dx(t)}{dt} = \frac{d}{dt}(e^{At})c \tag{3.5}$$

and, from the defining series (3.4),

$$\frac{d}{dt}(e^{At}) = A + A^2t + A^3\frac{t^2}{2!} + \cdots = A\left(I + At + A^2\frac{t^2}{2!} + \cdots\right) = A\,e^{At}$$

Thus (3.5) becomes

$$\frac{dx(t)}{dt} = Ae^{At}c = Ax(t)$$

Solving State Space (optional notes)

which was to be shown. To evaluate the constant c suppose that at some time τ the state $x(\tau)$ is given. Then, from (3.3),

$$x(\tau) = e^{A\tau}c \tag{3.6}$$

Multiplying both sides of (3.6) by the inverse of $e^{A\tau}$ we find that

$$c = (e^{A\tau})^{-1}x(\tau)$$

Thus the general solution to (3.2) for the state $x(t)$ at time t, given the state $x(\tau)$ at time τ, is

$$x(t) = e^{At}(e^{A\tau})^{-1}x(\tau) \tag{3.7}$$

The following property of the matrix exponential can readily be established by a variety of methods—the easiest perhaps being the use of the series definition (3.4)—

$$e^{A(t_1+t_2)} = e^{At_1}e^{At_2} \tag{3.8}$$

for any t_1 and t_2. From this property it follows that

$$(e^{A\tau})^{-1} = e^{-A\tau} \tag{3.9}$$

and hence that (3.7) can be written

$$x(t) = e^{A(t-\tau)}x(\tau) \tag{3.10}$$

Solving State Space (optional notes)

The matrix $e^{A(t-\tau)}$ is a special form of the *state-transition matrix* to be discussed subsequently.

We now turn to the problem of finding a "particular" solution to the nonhomogeneous, or "forced," differential equation (3.1) with A and B being constant matrices. Using the "method of the variation of the constant,"[1] we seek a solution to (3.1) of the form

$$x(t) = e^{At}c(t) \tag{3.11}$$

where $c(t)$ is a function of time to be determined. Take the time derivative of $x(t)$ given by (3.11) and substitute it into (3.1) to obtain:

$$Ae^{At}c(t) + e^{At}\dot{c}(t) = Ae^{At}c(t) + Bu(t)$$

or, upon cancelling the terms $A\,e^{At}c(t)$ and premultiplying the remainder by e^{-At},

$$\dot{c}(t) = e^{-At}Bu(t) \tag{3.12}$$

Thus the desired function $c(t)$ can be obtained by simple integration (the mathematician would say "by a quadrature")

$$c(t) = \int_T^t e^{-A\lambda}Bu(\lambda)\,d\lambda$$

The lower limit T on this integral cannot as yet be specified, because we will need to put the particular solution together with the solution to the

Solving State Space (optional notes)

homogeneous equation to obtain the complete (general) solution. For the present, let T be undefined. Then the particular solution, by (3.11), is

$$x(t) = e^{At}\int_T^t e^{-A\lambda}Bu(\lambda)\,d\lambda = \int_T^t e^{A(t-\lambda)}Bu(\lambda)\,d\lambda \tag{3.13}$$

In obtaining the second integral in (3.13), the exponential e^{At}, which does not depend on the variable of integration λ, was moved under the integral, and property (3.8) was invoked to write $e^{At}e^{-A\lambda} = e^{A(t-\lambda)}$.

The complete solution to (3.1) is obtained by adding the "complementary solution" (3.10) to the particular solution (3.13). The result is

$$x(t) = e^{A(t-\tau)}x(\tau) + \int_T^t e^{A(t-\lambda)}Bu(\lambda)\,d\lambda \tag{3.14}$$

We can now determine the proper value for lower limit T on the integral. At $t = \tau$ (3.14) becomes

$$x(\tau) = x(\tau) + \int_T^\tau e^{A(t-\lambda)}Bu(\lambda)\,d\lambda \tag{3.15}$$

Thus, the integral in (3.15) must be zero for any $u(t)$, and this is possible only if $T = \tau$. Thus, finally we have the complete solution to (3.1) when A and B are constant matrices

$$x(t) = e^{A(t-\tau)}x(\tau) + \int_\tau^t e^{A(t-\lambda)}Bu(\lambda)\,d\lambda \tag{3.16}$$

Solving State Space (optional notes)

This important relation will be used many times in the remainder of the book. It is worthwhile dwelling upon it. We note, first of all, that the solution is the sum of two terms: the first is due to the "initial" state $x(\tau)$ and the second—the integral—is due to the input $u(\tau)$ in the time interval $\tau \leqq \lambda \leqq t$ between the "initial" time τ and the "present" time t. The terms initial and present are enclosed in quotes to denote the fact that these are simply convenient definitions. There is no requirement that $t \geqq \tau$. The relationship is perfectly valid even when $t \leqq \tau$.

Another fact worth noting is that the integral term, due to the input, is a "convolution integral": the contribution to the state $x(t)$ due to the input u is the convolution of u with $e^{At}B$. Thus the function $e^{At}B$ has the role of the impulse response[1] of the system whose output is $x(t)$ and whose input is $u(t)$.

If the output y of the system is not the state x itself but is defined by the observation equation

$$y = Cx$$

then this output is expressed by

$$y(t) = Ce^{A(t-\tau)}x(t) + \int_{\tau}^{t} Ce^{A(t-\lambda)}Bu(\lambda)\,d\lambda \tag{3.17}$$

Solving State Space (optional notes)

and the impulse response of the system with y regarded as the output is $Ce^{A(t-\lambda)}B$.

The development leading to (3.16) and (3.17) did not really require that B and C be constant matrices. By retracing the steps in the development it is readily seen that when B and C are time-varying, (3.16) and (3.17) generalize to

$$x(t) = e^{A(t-\tau)}x(\tau) + \int_{\tau}^{t} e^{A(t-\lambda)}B(\lambda)u(\lambda)\,d\lambda \tag{3.18}$$

and

$$y(t) = C(t)\,e^{A(t-\tau)}x(\tau) + \int_{\tau}^{t} C(t)\,e^{A(t-\lambda)}B(\lambda)u(\lambda)\,d\lambda \tag{3.19}$$

Controllability

Controllability matrix

- To convert an arbitrary state representation in **F**, **G**, **H** and *J* to control canonical form **A**, **B**, **C** and *D*, the “controllability matrix”

$$\mathcal{C} = [\mathbf{G} \quad \mathbf{FG} \quad \mathbf{F}^2\mathbf{G} \quad \cdots \quad \mathbf{F}^{n-1}\mathbf{G}]$$

must be nonsingular.

Why is it called the “controllability” matrix?

Controllability matrix

- If you can write it in CCF, then the system equations must be linearly independent.
- Transformation by any nonsingular matrix preserves the controllability of the system.
- Thus, a nonsingular controllability matrix means $\boldsymbol{x}$ can be driven to any value.

Can you use this for more than Control?

Can you use this for more than Control?

- Yes

Frequency Response in State Space

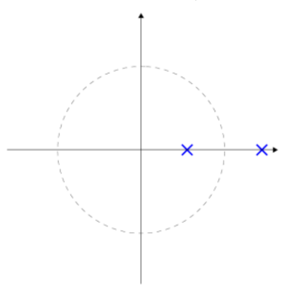
$$H(z) = C(zI - A)^{-1}B + D = \frac{1}{100z^2 - 200z + 80}$$

Poles at $\approx 0.55, 1.45$.

Eigenvalues of A:

$1, 1, 1.45, .55$

What are the (physical) implications?

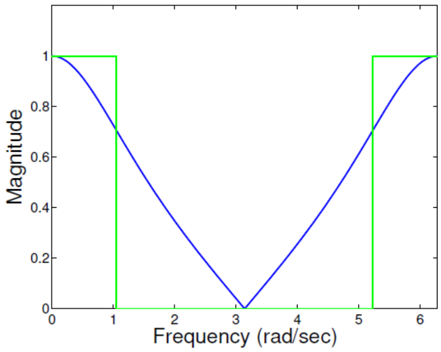


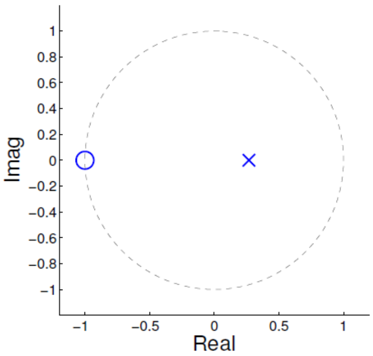
The Approach:

- Formulate the goal of control as an **optimization** (e.g. minimal impulse response, minimal effort, ...).
- You've already seen some examples of optimization-based design:
  - Used least-squares to obtain an FIR system which matched (in the least-squares sense) the desired frequency response.
  - Poles/zeros lecture: Butterworth filter

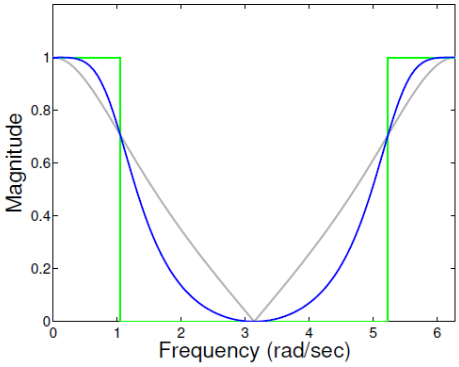
Discrete Time Butterworth Filters

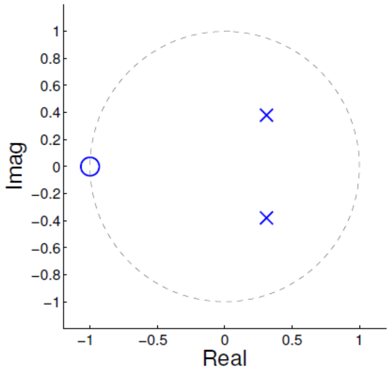
"Maximally-flat filter". Sacrifice sharpness to have flat response in pass band and stop band.



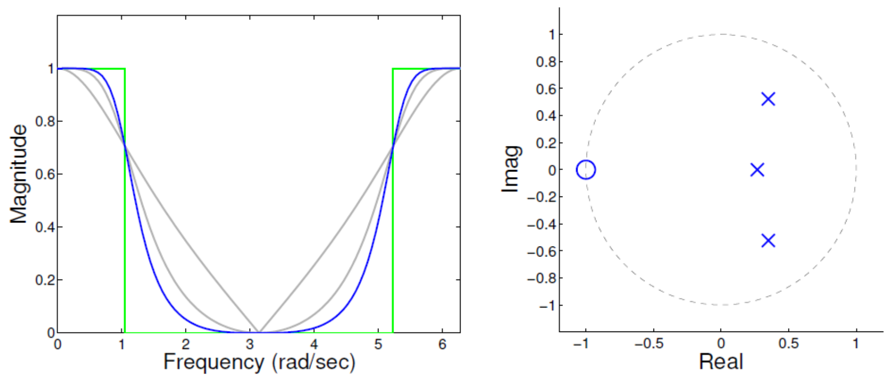


"Maximally-flat filter". Sacrifice sharpness to have flat response in pass band and stop band.

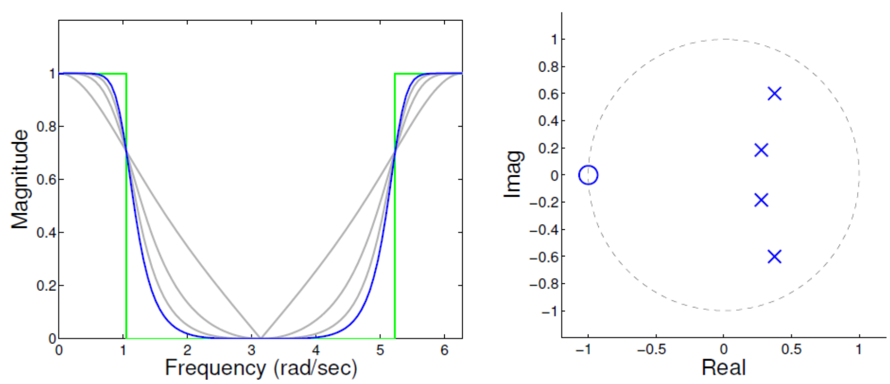




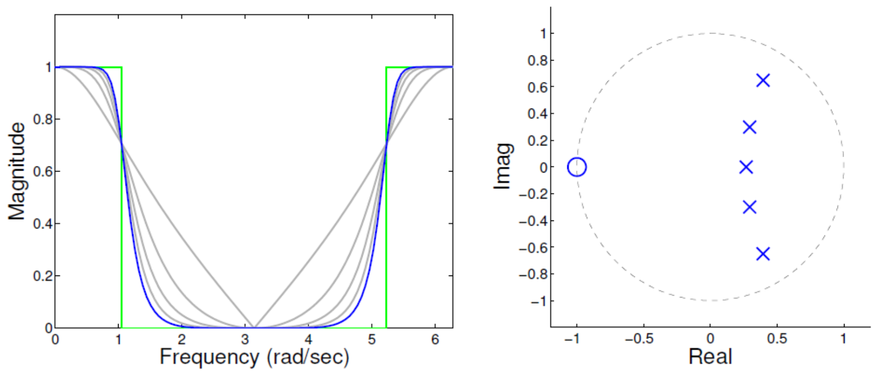
“Maximally-flat filter”. Sacrifice sharpness to have flat response in pass band and stop band.



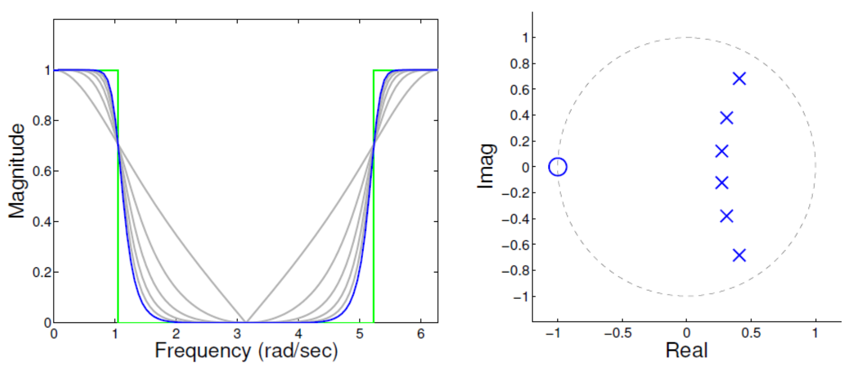
“Maximally-flat filter”. Sacrifice sharpness to have flat response in pass band and stop band.



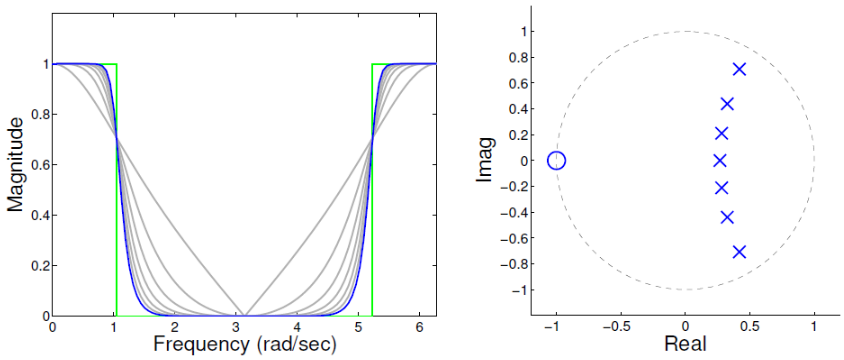
"Maximally-flat filter". Sacrifice sharpness to have flat response in pass band and stop band.



"Maximally-flat filter". Sacrifice sharpness to have flat response in pass band and stop band.



“Maximally-flat filter”. Sacrifice sharpness to have flat response in pass band and stop band.



How?

- Constrained Least-Squares …

One formulation: Given $x[0]$

$$\underset{u[0],u[1],\ldots,u[N]}{\text{minimize}} \quad ||\vec{u}||^2, \quad \text{where } \vec{u} = \begin{bmatrix} u[0] \\ u[1] \\ \vdots \\ u[N] \end{bmatrix}$$

$$\text{subject to} \quad x[N] = 0.$$

Note that

$$x[n] = A^n x[0] + \sum_{k=0}^{n-1} A^{(n-1-k)} B u[k],$$

so this problem can be written as

$$\underset{x_{ls}}{\text{minimize}} \, ||A_{ls}x_{ls} - b_{ls}||^2 \quad \text{subject to} \quad C_{ls}x_{ls} = D_{ls}.$$

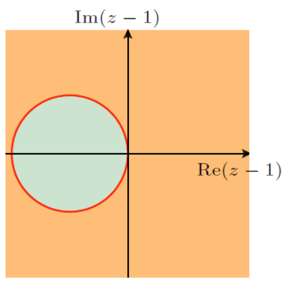
Stability

Fast sampling revisited

- For small T:

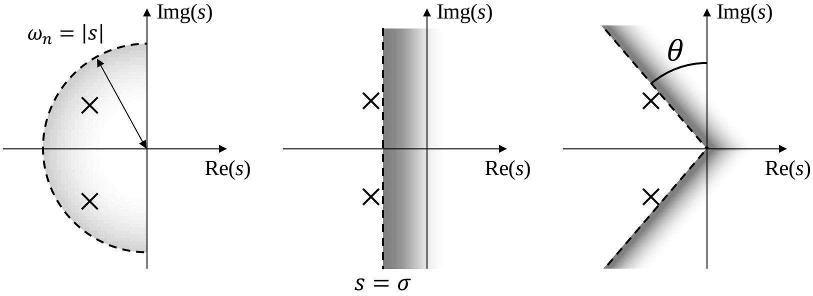
$$z = e^{sT} = 1 + sT + \frac{(sT)^2}{2} + \ldots \approx 1 + sT$$
$$\rightarrow z \approx 1 + sT \rightarrow s = \frac{z-1}{T}$$

- Hence, the unit circle under the map from z to s-plane becomes:



Specification bounds

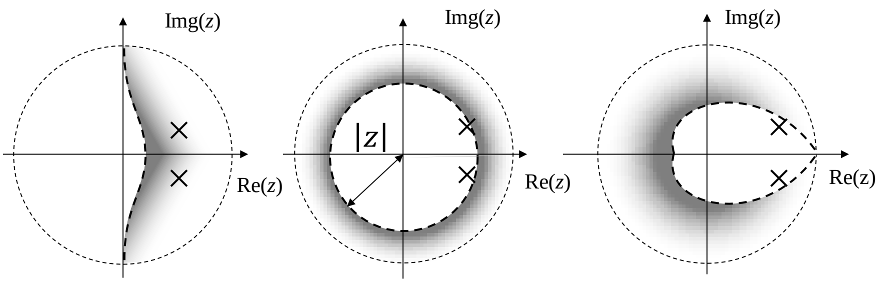
- Recall in the continuous domain, response performance metrics map to the s-plane:



$|s| = \frac{1.8}{t_r}$ $\quad s = \frac{4.6}{t_s}$ $\quad \theta = \sin^{-1}\zeta$

Discrete bounds

- These map to the discrete domain:



$|z| = e^{-t_s T}$

In practice, you'd use Matlab to plot these, and check that the spec is satisfied

Example Code:

```
%% Input System Model G
numg=5; deng=[1 20 0]; sysg=tf(numg, deng);

%% Approximate the ZOH (1-e^-sT)/(s)
[nd, dd]=pade(1,2);  %pade gives us the "hold" or -e^-sT of a ZOH
sysp=tf(nd, dd); sysi=tf([1],[1,0]); %Now we need the "1/s" portion
sys1=series(1-sysp, sysi); % Approximation as a series

%% Open loop response
syso=series(sys1, sysg); % computer the open loop G with the ZOH
sys=feedback(syso,1); % Computer the unity feedback response
step(sys) % Display the step response
```


Next Time…

- **Digital Control via Emulation!**

- Review:
  - Chapter 5 of FPW

- Deeper Pondering??