Z-Transform
Second Order LTID
(& Convolution Review)

http://elec3004.com

ELEC 3004: **Systems**: Signals & Controls

Dr. Surya Singh

Lecture 8

elec3004@itee.uq.edu.au

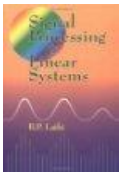
http://robotics.itee.uq.edu.au/~elec3004/

March 23, 2017

Lecture Schedule:

Week	Date	Lecture Title
1	28-Feb	Introduction
	2-Mar	Systems Overview
2	7-Mar	Systems as Maps & Signals as Vectors
	9-Mar	Systems: Linear Differential Systems
3	14-Mar	Sampling Theory & Data Acquisition
	16-Mar	Aliasing & Antialiasing
4	21-Mar	Discrete Time Analysis & Z-Transform
	23-Mar	Second Order LTID (& Convolution Review)
5	28-Mar	Frequency Response
	30-Mar	Filter Analysis
6	4-Apr	Digital Filters (IIR)
	6-Apr	Digital Windows
7	11-Apr	Digital Filter (FIR)
	13-Apr	FFT
	18-Apr	Holiday
	20-Apr	
	25-Apr	
8	27-Apr	Active Filters & Estimation
9	2-May	Introduction to Feedback Control
	4-May	Servoregulation/PID
10	9-May	Introduction to (Digital) Control
	11-May	Digitial Control
11	16-May	Digital Control Design
	18-May	Stability
12	23-May	Digital Control Systems: Shaping the Dynamic Response
	25-May	Applications in Industry
13	30-May	System Identification & Information Theory
	1-Jun	Summary and Course Review

Follow Along Reading:



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

Today

- Chapter 11 (**Discrete-Time System Analysis Using the *z*-Transform**)
  - § 11.1 The $\mathcal{Z}$-Transform
  - § 11.2 Some Properties of the *Z*-Transform

- Chapter 9 (**Time-Domain Analysis of Discrete-Time Systems**)
  - § 9.4 System Response to External Input
  - § 9.6 System Stability

Next Time

z Transforms

(Digital Systems Made eZ)

Review and Extended Explanation

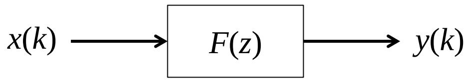
The z-transform

- The discrete equivalent is the *z*-Transform†:

$$\mathcal{Z}\{f(k)\} = \sum_{k=0}^{\infty} f(k) z^{-k} = F(z)$$

and

$$\mathcal{Z}\{f(k-1)\} = z^{-1}F(z)$$



Convenient!

†This is not an approximation, but approximations are easier to derive

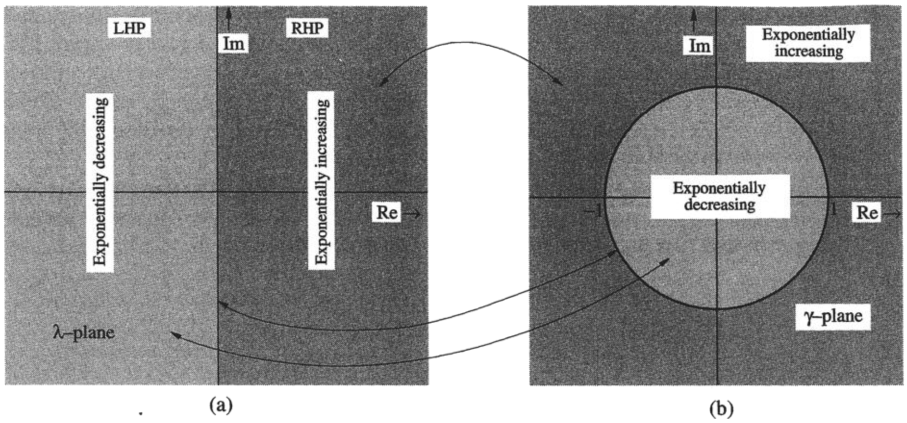
The z-transform

- In practice, you'll use look-up tables or computer tools (ie. Matlab) to find the *z*-transform of your functions

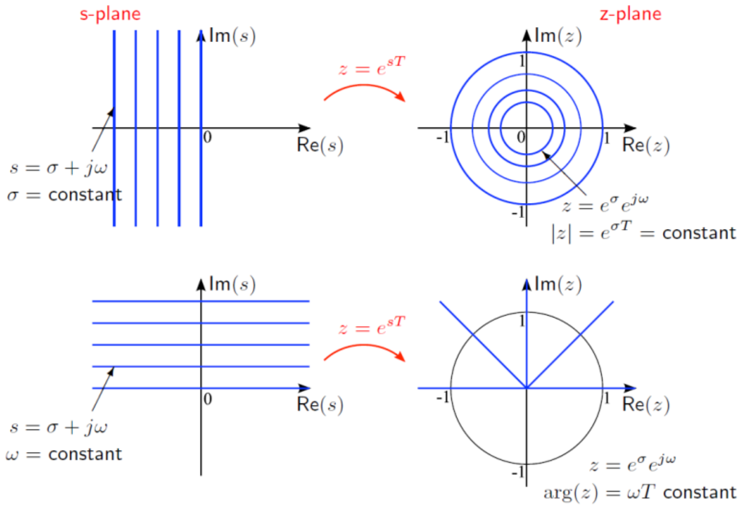
$F(s)$	$F(kt)$	$F(z)$
$\frac{1}{s}$	1	$\frac{z}{z-1}$
$\frac{1}{s^2}$	kT	$\frac{Tz}{(z-1)^2}$
$\frac{1}{s+a}$	e^{-akT}	$\frac{z}{z-e^{-aT}}$
$\frac{1}{(s+a)^2}$	kTe^{-akT}	$\frac{zTe^{-aT}}{(z-e^{-aT})^2}$
$\frac{1}{s^2+a^2}$	$\sin(akT)$	$\frac{z\sin aT}{z^2-(2\cos aT)z+1}$

S-Plane to z-Plane [1/3]: Discrete-Time Exponential γ^k

$$e^{\lambda k} = \gamma^k$$



S-Plane to z-Plane [2/3]



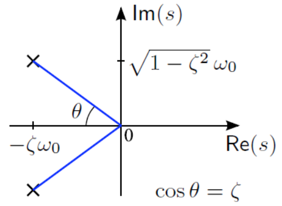
S-Plane to z-Plane [2/3]

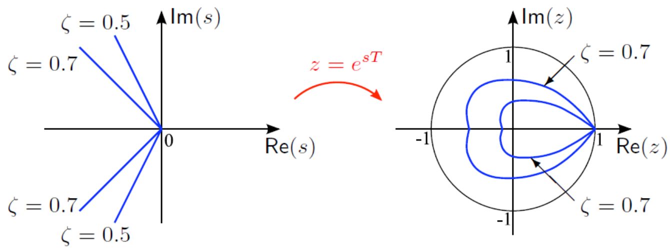
Pole locations for constant damping ratio $\zeta < 1$

$$s^2 + \zeta\omega_0 s + \omega_0^2 = 0$$

$$\Downarrow$$

$$s = -\zeta\omega_0 \pm j\sqrt{1-\zeta^2}\,\omega_0$$





$s = -\zeta\omega_0 + j\sqrt{1-\zeta^2}\,\omega_0$: $\zeta =$ constant $\qquad z = e^{-\zeta\omega_0 T} e^{-j\sqrt{1-\zeta^2}\omega_0 T}$

Relationship with s-plane poles and z-plane transforms

If $F(s)$ has a pole at $s = a$

then $F(z)$ has a pole at $z = e^{aT}$

$\uparrow$

consistent with $z = e^{sT}$

What about transfer functions?

$$G(z) = (1 - z^{-1})\mathcal{Z}\left\{\frac{G(s)}{s}\right\}$$

$\downarrow$

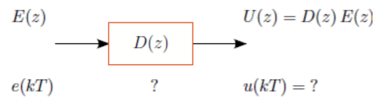
If $G(s)$ has poles $s = a_i$

then $G(z)$ has poles $z = e^{a_i T}$

but the zeros are unrelated

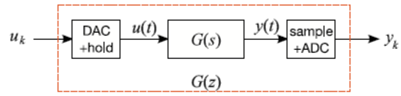
$\mathcal{F}(s)$	$f(kT)$	$F(z)$
$\frac{1}{s}$	$1(kT)$	$\frac{z}{z-1}$
$\frac{1}{s^2}$	kT	$\frac{Tz}{(z-1)^2}$
$\frac{1}{s+a}$	e^{-akT}	$\frac{z}{z-e^{-aT}}$
$\frac{1}{(s+a)^2}$	kTe^{-akT}	$\frac{Tze^{-aT}}{(z-e^{-aT})^2}$
$\frac{a}{s(s+a)}$	$1-e^{-akT}$	$\frac{z(1-e^{-aT})}{(z-1)(z-e^{-aT})}$
$\frac{b-1}{(s+a)(s+b)}$	$e^{-akT}-e^{-bkT}$	$\frac{(e^{-aT}-e^{-bT})z}{(z-e^{-aT})(z-e^{-bT})}$
$\frac{a}{s^2+a^2}$	$\sin akT$	$\frac{z\sin aT}{z^2-(2\cos aT)z+1}$
$\frac{b}{(s+a)^2+b^2}$	$e^{-akT}\sin bkT$	$\frac{ze^{-aT}\sin bT}{z^2-2e^{-aT}(\cos bT)z+e^{-2aT}}$

$s \leftrightarrow z$: Pulse Transfer Function Models



- Pulse in Discrete is equivalent to Dirac-δ

$$e_k = \begin{cases} 1 & \text{for } k = 0 \\ 0 & \text{for } k > 0 \end{cases}$$



➔

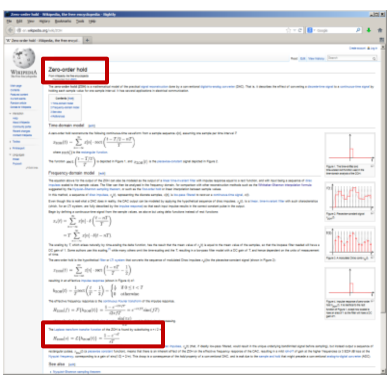
$$G(z) = (1 - z^{-1})\mathcal{Z}\left\{\mathcal{L}^{-1}\left\{\frac{G(s)}{s}\right\}_{t=kT}\right\} = (\mathbf{1} - \boldsymbol{z}^{-\mathbf{1}})\boldsymbol{\mathcal{Z}}\left\{\frac{\boldsymbol{G(s)}}{\boldsymbol{s}}\right\}$$

Source: Oxford 2A2 Discrete Systems, Tutorial Notes p. 26

$\mathcal{L}$(ZOH)=??? : What is it?

$$\frac{1 - e^{-Ts}}{Ts}$$

- Wikipedia

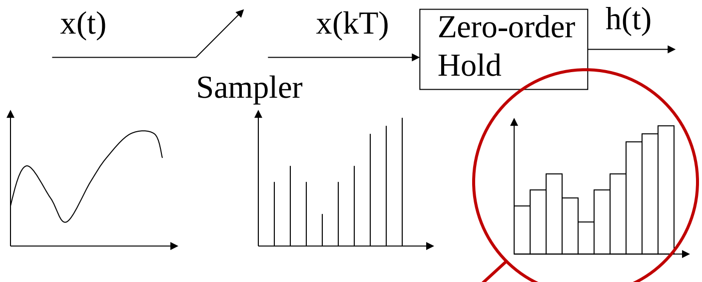


$$\frac{1 - e^{-Ts}}{s}$$

- Lathi
- Franklin, Powell, Workman
- Franklin, Powell, Emani-Naeini
- Dorf & Bishop
- Oxford Discrete Systems: (Mark Cannon)
- MIT 6.002 (Russ Tedrake)
- Matlab

Proof!

Zero-order-hold (ZOH)



- Assume that the signal x(t) is zero for t<0, then the output h(t) is related to x(t) as follows:

$$h(t) = x(0)[1(t) - 1(t-T)] + x(T)[1(t-T) - 1(t-2T)] + \cdots$$
$$= \sum_{k=0}^{\infty} x(kT)[1(t - kT) - 1(t - (k+1)T)]$$

Transfer function of Zero-order-hold (ZOH)

- Recall the Laplace Transforms ($\mathcal{L}$) of:

$$\mathcal{L}[\delta(t)] = 1 \qquad \mathcal{L}[f(t-kT)] = F(s)e^{-kTs}$$
$$\mathcal{L}\left[\delta(t-kT)\right] = e^{-kTs} \qquad \mathcal{L}[1(t-kT)] = \frac{e^{-kTs}}{s}$$

- Thus the $\mathcal{L}$ of h(t) becomes:

$$\mathcal{L}[h(t)] = \mathcal{L}[\sum_{k=0}^{\infty} x(kT)[1(t - kT) - 1(t - (k+1)T)]]$$
$$= \sum_{k=0}^{\infty} x(kT)\mathcal{L}[1(t - kT) - 1(t - (k+1)T)] = \sum_{k=0}^{\infty} x(kT)[\frac{e^{-kTs}}{s} - \frac{e^{-(k+1)Ts}}{s}]$$
$$= \sum_{k=0}^{\infty} x(kT)\frac{e^{-kTs} - e^{-(k+1)Ts}}{s} = \sum_{k=0}^{\infty} x(kT)\frac{1-e^{-Ts}}{s}e^{-kTs} = \frac{1-e^{-Ts}}{s}\sum_{k=0}^{\infty} x(kT)e^{-kTs}$$

Transfer function of Zero-order-hold (ZOH)

… Continuing the $\mathcal{L}$ of h(t) …

$$\mathcal{L}[h(t)] = \mathcal{L}[\sum_{k=0}^{\infty} x(kT)[1(t - kT) - 1(t - (k+1)T)]]$$

$$= \sum_{k=0}^{\infty} x(kT) L[1(t - kT) - 1(t - (k+1)T)] = \sum_{k=0}^{\infty} x(kT)[\frac{e^{-kTs}}{s} - \frac{e^{-(k+1)Ts}}{s}]$$

$$= \sum_{k=0}^{\infty} x(kT)\frac{e^{-kTs} - e^{-(k+1)Ts}}{s} = \sum_{k=0}^{\infty} x(kT)\frac{1-e^{-Ts}}{s}e^{-kTs} = \frac{1-e^{-Ts}}{s}\sum_{k=0}^{\infty} x(kT)e^{-kTs}$$

$$\rightarrow X(s) = \mathcal{L}\left[\sum_{k=0}^{\infty} x(kT)\delta(t-kT)\right] = \sum_{k=0}^{\infty} x(kT)e^{-kTs}$$

$$\therefore H(s) = \mathcal{L}[h(t)] = \frac{1-e^{-Ts}}{s}\sum_{k=0}^{\infty} x(kT)e^{-kTs} = \frac{1-e^{-Ts}}{s}X(s)$$

➔ Thus, giving the transfer function as:

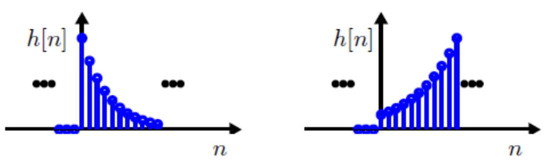
$$G_{\mathrm{ZOH}}(s) = \frac{H(s)}{X(s)} = \frac{1-e^{-Ts}}{s} \xrightarrow{\mathcal{Z}} G_{ZOH}(z) = \frac{\left(1-e^{-aT}\right)}{z-e^{-aT}}$$

z-Transforms for Difference Equations

- First-order linear constant coefficient difference equation:

First-order linear constant coefficient difference equation:

$$y[n] = ay[n-1] + bu[n]$$



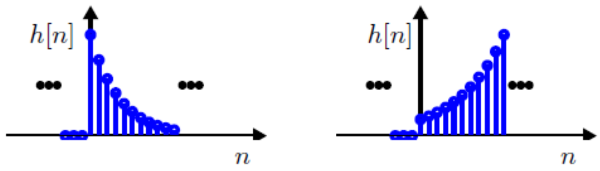
$$h[n] = \begin{cases} ba^n & n \geq 0, \\ 0 & \text{otherwise.} \end{cases}$$

$$H(z) = \sum_{k=0}^{\infty} ba^k z^{-k} = b\sum_{k=0}^{\infty}\left(\frac{a}{z}\right)^k = \frac{b}{1-az^{-1}}, \quad \text{when } |z| > |a|.$$

z-Transforms for Difference Equations

First-order linear constant coefficient difference equation:

$$y[n] = ay[n-1] + bu[n]$$



$$y[n] - ay[n-1] = bu[n]$$

$$\updownarrow$$

$$Y(z) - az^{-1}Y(z) = bU(z)$$

$$H(z) = \frac{Y(z)}{U(z)} = \frac{b}{1 - az^{-1}},$$ when does it converge?

z-Transform Example

- Obtain the z-Transform of the sequence:

$$x[k] = \{3, 0, 1, 4,1,5, \dots\}$$

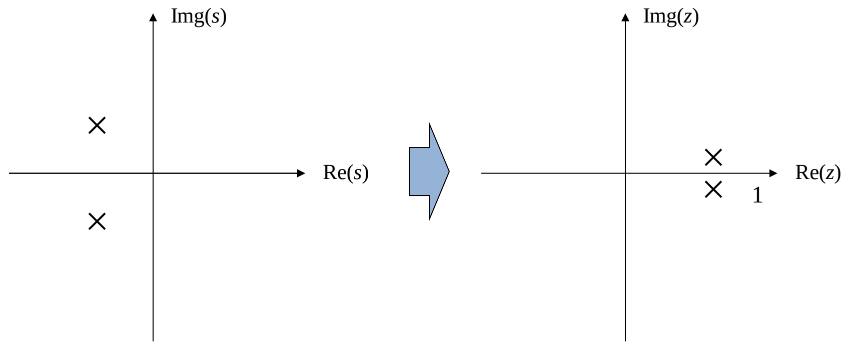
- Solution:

$$X(z) = 3 + z^{-2} + 4z^{-3} + z^{-4} + 5z^{-5}$$

The z-Plane

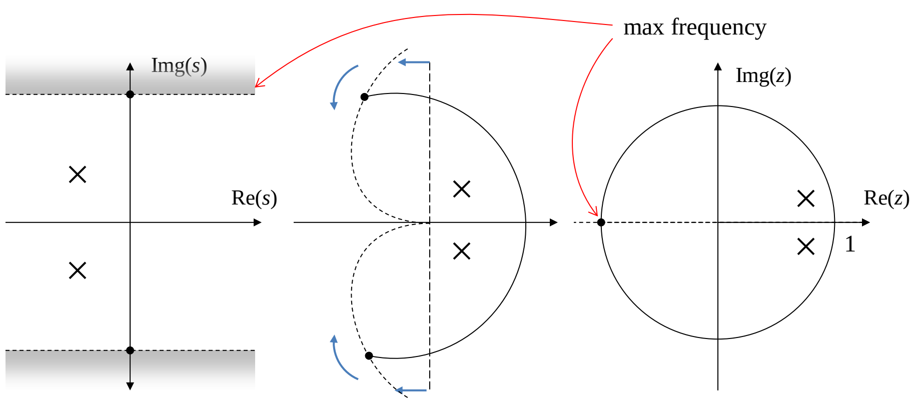
z-domain poles and zeros can be plotted just like s-domain poles and zeros (of the $\mathcal{L}$):

- S-plane:
  - λ – Plane
- $z = e^{sT}$ Plane
  - γ – Plane



Deep insight # 1

The mapping between continuous and discrete poles and zeros acts like a distortion of the plane



System Stability

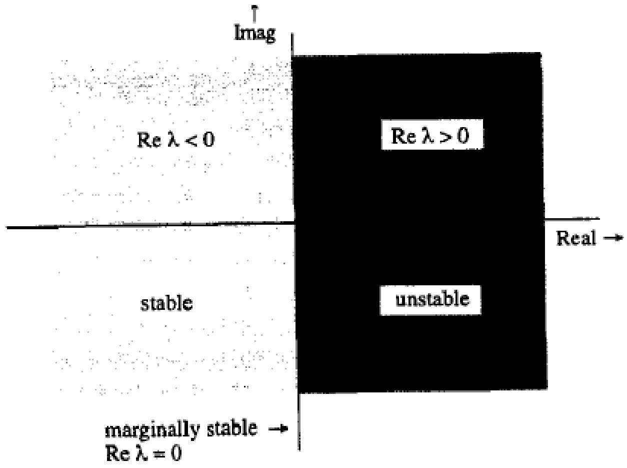
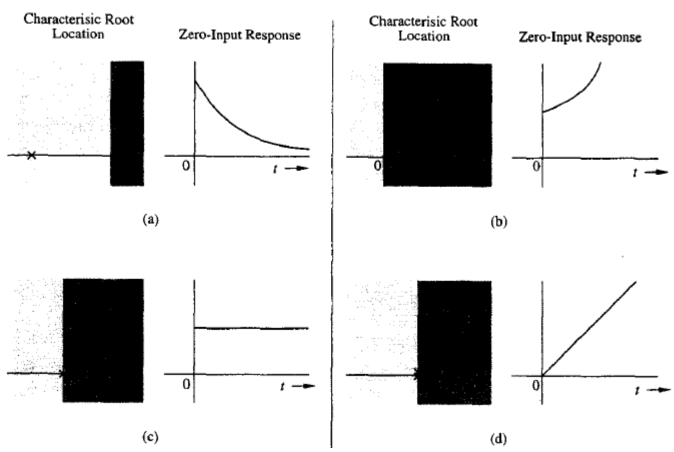


Fig. 2.15 Characteristic roots location and system stability.

Lathi, p. 149

System Stability [II]



Lathi, p. 150

System Stability [III]

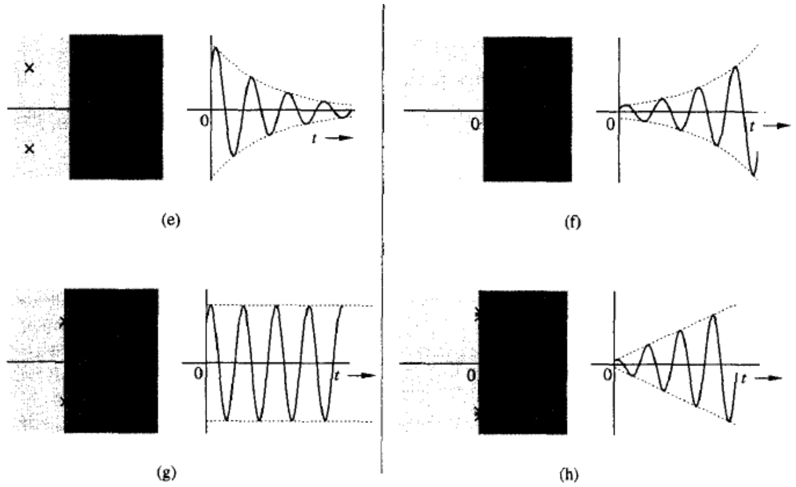
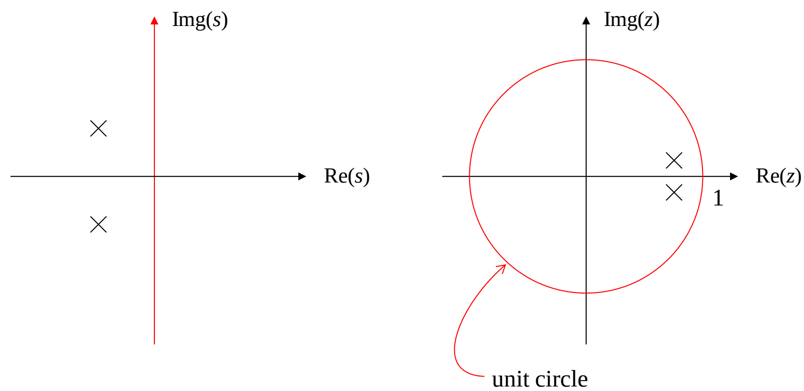


Fig. 2.16 Location of characteristic roots and the corresponding characteristic modes.

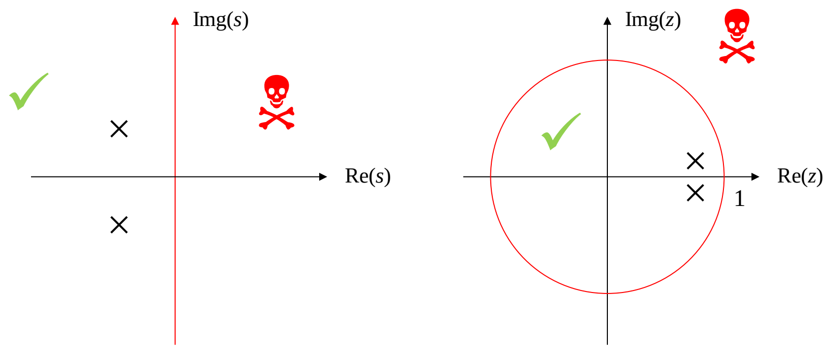
γ-plane Stability

- For a γ-Plane (e.g. the one the *z*-domain is embedded in) the unit circle is the system stability bound



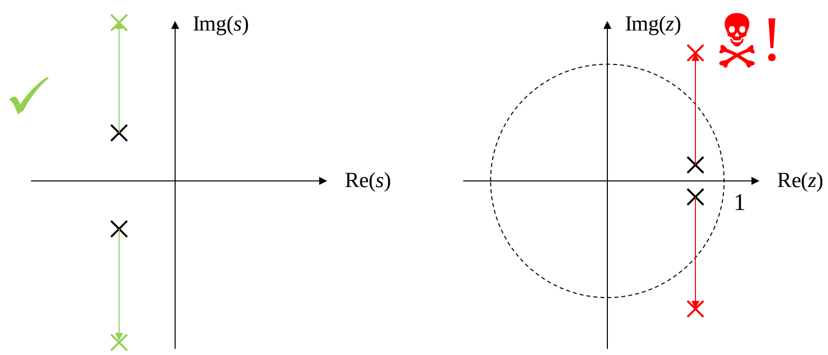
γ-plane Stability

- That is, in the *z*-domain,
  the unit circle is the system stability bound



z-plane stability

- The z-plane root-locus in closed loop feedback behaves just like the s-plane:



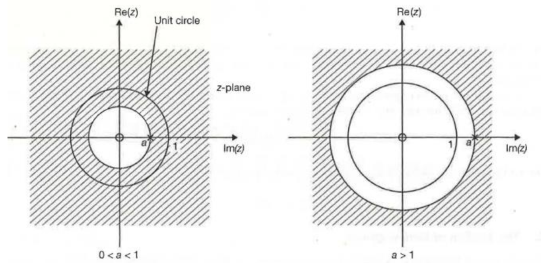
Region of Convergence

- For the convergence of X(z) we require that

$$\sum_{n=0}^{\infty} |az^{-1}|^n < \infty$$

- Thus, the ROC is the range of values of z for which $|az^{-1}| < 1$ or, equivalently, $|z| > |a|$. Then

$$X(z) = \frac{z}{z-a} \qquad |z| > |a|$$



An example!

- Back to our difference equation:

$$y(k) = x(k) + Ax(k-1) - By(k-1)$$

becomes

$$Y(z) = X(z) + Az^{-1}X(z) - Bz^{-1}Y(z)$$
$$(z+B)Y(z) = (z+A)X(z)$$

which yields the transfer function:

$$\frac{Y(z)}{X(z)} = \frac{z+A}{z+B}$$

Note: It is also not uncommon to see systems expressed as polynomials in z^{-n}

This looks familiar…

- Compare:

$$\frac{Y(s)}{X(s)} = \frac{s+2}{s+1} \text{ vs } \frac{Y(z)}{X(z)} = \frac{z+A}{z+B}$$

How are the Laplace and *z* domain representations related?

Linearity:

$$a_1 y_1[n] + a_2 y_2[n] \overset{\mathcal{Z}}{\longleftrightarrow} a_1 Y_1(z) + a_2 Y_2(z)$$

Z-Transform Properties: Time Shifting

$$y[n-n_0] \overset{\mathcal{Z}}{\longleftrightarrow} z^{-n_0} Y(z)$$

$$\begin{aligned} y_2[n] &= y[n-n_0] \\ Y_2(e^{jw}) &= \sum_{k=-\infty}^{\infty} y[k-n_0] z^{-k} \\ &= \sum_{l=-\infty}^{\infty} y[l] z^{-(l+n_0)} \\ &= z^{-n_0} Y(z) \end{aligned}$$

- Two Special Cases:
- z^{-1}: the *unit-delay operator:*

$$x[n-1] \leftrightarrow z^{-1} X(z) \qquad R' = R \cap \{0 < |z|\}$$

- z: *unit-advance operator:*

$$x[n+1] \leftrightarrow z X(z) \qquad R' = R \cap \{|z| < \infty\}$$

More Z-Transform Properties

- Time Reversal

$$x[n] \leftrightarrow X(z) \qquad \text{ROC} = R$$

$$x[-n] \leftrightarrow X\left(\frac{1}{z}\right) \qquad R' = \frac{1}{R}$$

- Multiplication by z^n

$$x[n] \leftrightarrow X(z) \qquad \text{ROC} = R$$

$$z_0^n x[n] \leftrightarrow X\left(\frac{z}{z_0}\right) \qquad R' = |z_0| R$$

- Multiplication by n (or Differentiation in z):

$$x[n] \leftrightarrow X(z) \qquad \text{ROC} = R$$

$$nx[n] \leftrightarrow -z\frac{dX(z)}{dz} \qquad R' = R$$

- Convolution

$$x_1[n] \leftrightarrow X_1(z) \qquad \text{ROC} = R_1$$
$$x_2[n] \leftrightarrow X_2(z) \qquad \text{ROC} = R_2$$

$$x_1[n] * x_2[n] \leftrightarrow X_1(z)X_2(z) \qquad R' \supset R_1 \cap R_2$$

ELEC 3004: **Systems** 23 March 2016 31

Linear Difference Equations
(a sub-set of Linear, Discrete Dynamical Systems)

ELEC 3004: **Systems** 23 March 2016 32

DT Causality & BIBO Stability [Review]

- Causality:

$$h[n] = 0,\ n < 0$$

$$\rightarrow y[n] = \sum_{k=0}^{\infty} h[k]x[n-k] \quad \text{or} \quad \Rightarrow y[n] = \sum_{k=-\infty}^{n} x[k]h[n-k]$$

- Input is Causal if: $x[n] = 0,\ n < 0$

- Then output is Causal:

$$y[n] = \sum_{k=0}^{n} h[k]x[n-k] = \sum_{k=0}^{n} x[k]h[n-k]$$

- And, DT LTI is BIBO stable if:

$$\sum_{k=-\infty}^{\infty} |h[k]| < \infty$$

Linear Difference Equations

$$u_k = f(e_0, \ldots, e_k; u_0, \ldots, u_{k-1}).$$

$$u_k = -a_1 u_{k-1} - a_2 u_{k-2} - \cdots - a_n u_{k-n} + b_0 e_k + b_1 e_{k-1} + \cdots + b_m e_{k-m}.$$

$$\begin{aligned} \nabla u_k &= u_k - u_{k-1} && \text{(first difference)}, \\ \nabla^2 u_k &= \nabla u_k - \nabla u_{k-1} && \text{(second difference)}, \\ \nabla^n u_k &= \nabla^{n-1} u_k - \nabla^{n-1} u_{k-1} && (n\text{th difference}). \end{aligned}$$

$$\begin{aligned} u_k &= u_k, \\ u_{k-1} &= u_k - \nabla u_k, \\ u_{k-2} &= u_k - 2\nabla u_k + \nabla^2 u_k. \end{aligned}$$

$$a_2 \nabla^2 u_k - (a_1 + 2a_2)\nabla u_k + (a_2 + a_1 + 1)u_k = b_0 e_k.$$

Assume a form of the solution

z^k :

- k: “order of difference”
- k: delay

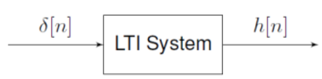
$$Az^k = Az^{k-1} + Az^{k-2}.$$

$$1 = z^{-1} + z^{-2}$$

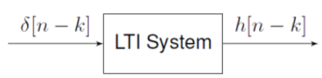
$$z^2 = z + 1.$$

Impulse Response (Graphically)

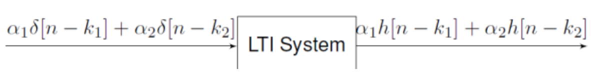
Let’s define the *impulse response*, $h[n]$, as the result of applying an LTI system to the unit impulse:

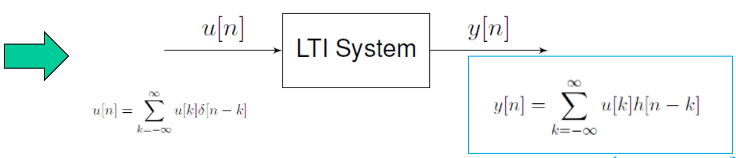


By time invariance, we know



And by linearity, we know





∞ matrix × ∞ vector?

How do you multiply an infinite matrix?

- First let's multiply circulant matrices…
  - A circulant matrix can be descibed completely by its first row or column

$$A = \begin{bmatrix} a_0 & a_1 & a_2 & \cdots & a_{n-1} \\ a_{n-1} & a_0 & a_1 & \cdots & a_{n-2} \\ a_{n-2} & a_{n-1} & a_0 & \cdots & a_{n-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_1 & a_2 & a_3 & \cdots & a_0 \end{bmatrix} = \begin{bmatrix} | & | & | & & | \\ h & Zh & Z^2h & \cdots & Z^{N-1}h \\ | & | & | & & | \end{bmatrix}$$

Z: Shift operator

- Multiply by u[k] ➔ $\begin{bmatrix} | & | & | & & | \\ h & Zh & Z^2h & \cdots & Z^{N-1}h \\ | & | & | & & | \end{bmatrix} \begin{bmatrix} u[0] \\ u[1] \\ u[2] \\ \vdots \\ u[N-1] \end{bmatrix} = \sum_{k=0}^{N-1} u[k] Z^k h$

∴ For circulant matrices, matrix multiplication reduces to a weighted combination of shifted impulse responses

Two Types of Systems

- Linear shift-invariant:

$$y = \sum_{k=0}^{N-1} u[k] Z^k h$$

Z: Shift operator

$$Z \cdot [u_0, u_1, u_2, u_3, \ldots, u_{n-1}]^T = [u_{n-1}, u_0, u_1, u_2, \ldots, u_{n-2}]^T$$

- Linear time-invariant system

$$y = \sum_{k=-\infty}^{\infty} u[k] R^k h$$

R: Unit delay operator

$$R \cdot [\ldots, u_0, u_1, u_2, u_3, \ldots]^T = [\ldots, u_{-1}, u_0, u_1, \ldots]^T$$

Impulse Response of Both Types

$$y[n] = \frac{1}{2}u[n-1] + \frac{1}{2}u[n]$$

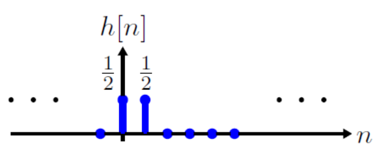
$$y[-1] = 0$$
$$y[0] = \frac{1}{2}$$
$$y[1] = \frac{1}{2}$$
$$y[2] = 0$$
$$\vdots$$

$$y[n] = \frac{1}{2}y[n-1] + u[n]$$

$$h[-1] = 0$$
$$h[0] = 1$$
$$h[1] = \frac{1}{2}$$
$$h[2] = \frac{1}{4}$$
$$\vdots$$
$$h[n] = \begin{cases} 0 & n < 0 \\ (\frac{1}{2})^n & n \geq 0 \end{cases}$$

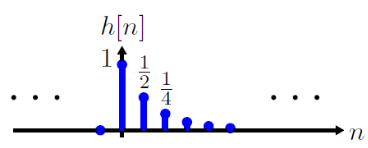
Impulse Response of Both Types

$$y[n] = \frac{1}{2}u[n-1] + \frac{1}{2}u[n]$$



'Finite impulse response" (FIR)

$$y[n] = \frac{1}{2}y[n-1] + u[n]$$



"Infinite impulse response" (IIR)

BREAK

Convolution

Convolution Definition

The convolution of two functions $f_1(t)$ and $f_2(t)$ is defined as:

$$f(t) = \int_{-\infty}^{\infty} f_1(\tau) f_2(t-\tau) d\tau$$

$$= f_1(t) * f_2(t)$$

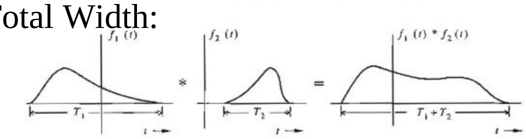
Source: URI ELE436

Convolution Properties

$$f_1(t) * f_2(t) \equiv \int_{-\infty}^{\infty} f_1(\tau) f_2(t-\tau)\, d\tau$$

Properties:

- Commutative: $f_1(t) * f_2(t) = f_2(t) * f_1(t)$
- Distributive: $f_1(t) * [f_2(t) + f_3(t)] = f_1(t) * f_2(t) + f_1(t) * f_3(t)$
- Associative: $f_1(t) * [f_2(t) * f_3(t)] = [f_1(t) * f_2(t)] * f_3(t)$
- Shift:
  if $f_1(t)*f_2(t)=c(t)$, then $f_1(t-\mathbf{T})*f_2(t)= f_1(t)*f_2(t-\mathbf{T})=c(t-\mathbf{T})$
- Identity (Convolution with an Impulse):
  $f(t) * \delta(t) = f(t)$
- Total Width:



Based on Lathi, *SPLS*, Sec 2.4-1

Convolution Properties [II]

- Convolution systems are **linear**:

$$h * (\alpha u_1 + \beta u_2) = \alpha(h * u_1) + \beta(h * u_2)$$

- Convolution systems are **causal**: the output y(t) at time t depends only on past inputs

- Convolution systems are **time-invariant** (if we shift the signal, the output similarly shifts)

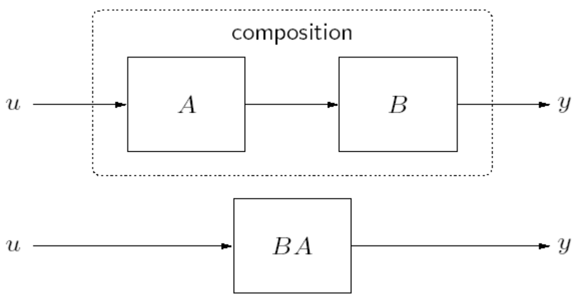
➔

$$\tilde{u}(t) = \begin{cases} 0 & t < T \\ u(t-T) & t \geq 0 \end{cases}$$

$$\tilde{y}(t) = \begin{cases} 0 & t < T \\ y(t-T) & t \geq 0 \end{cases}$$

Convolution Properties [III]

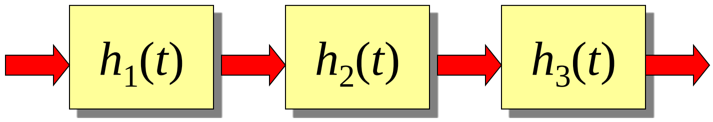
- Composition of convolution systems corresponds to:
  - multiplication of transfer functions
  - convolution of impulse responses

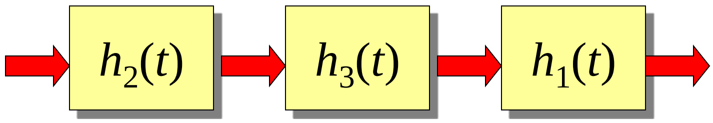


- Thus:
  - We can manipulate block diagrams with transfer functions as if they were simple gains
  - convolution systems commute with each other

Properties of Convolution: Distributive Property

$$[f_1(t) * f_2(t)] * f_3(t) = f_1(t) * [f_2(t) * f_3(t)]$$





- The two systems are identical!

Source: URI ELE436

Properties of Convolution: Commutative Property

$$f_1(t) * f_2(t) = f_2(t) * f_1(t)$$

$$f_1(t)*f_2(t) = \int_{-\infty}^{\infty} f_1(\tau) f_2(t-\tau)d\tau = \int_{\tau=-\infty}^{\tau=\infty} f_1(\tau) f_2(t-\tau)d\tau$$

$$= \int_{t-\tau=-\infty}^{t-\tau=\infty} f_1(t-\tau) f_2[t-(t-\tau)]d(t-\tau)$$

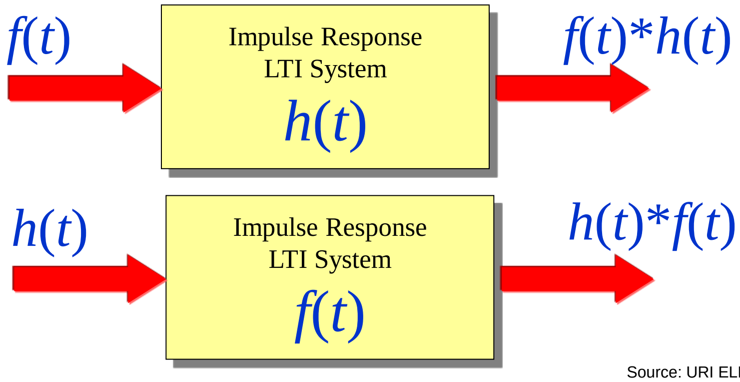
$$= -\int_{\tau=\infty}^{\tau=-\infty} f_1(t-\tau) f_2(\tau)d\tau$$

$$= \int_{-\infty}^{\infty} f_1(t-\tau) f_2(\tau)d\tau = f_2(t) * f_1(t)$$

Source: URI ELE436

Properties of Convolution: LTI System Response

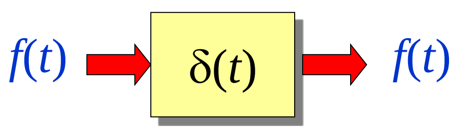
$$f_1(t) * f_2(t) = f_2(t) * f_1(t)$$



Source: URI ELE436

Properties of Convolution

$$f(t) * \delta(t) = f(t)$$



$$f(t) * \delta(t) = \int_{-\infty}^{\infty} f(\tau)\delta(t-\tau)d\tau$$

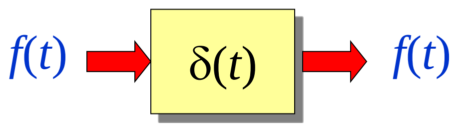
$$= \int_{-\infty}^{\infty} f(t-\tau)\delta(\tau)d\tau$$

$$= f(t)$$

Source: URI ELE436

Properties of Convolution

$$f(t) * \delta(t) = f(t)$$



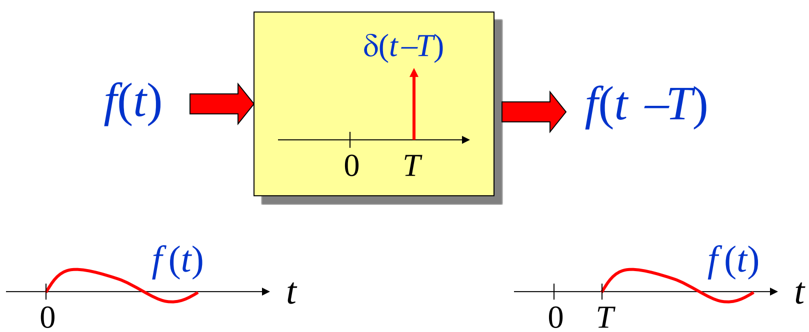
$$f(t) * \delta(t-T) = f(t-T)$$

$$\begin{aligned} f(t) * \delta(t-T) &= \int_{-\infty}^{\infty} f(\tau)\delta(t-T-\tau)d\tau \\ &= \int_{-\infty}^{\infty} f(t-T-\tau)\delta(\tau)d\tau \\ &= f(t-T) \end{aligned}$$

Source: URI ELE436

Properties of Convolution

$$f(t) * \delta(t-T) = f(t-T)$$



Source: URI ELE436

Properties of Convolution

$$f_1(t) * f_2(t) \overset{\mathcal{F}}{\longleftrightarrow} F_1(j\omega)F_2(j\omega)$$

$$F[f_1(t) * f_2(t)] = \int_{-\infty}^{\infty}\left[\int_{-\infty}^{\infty} f_1(\tau) f_2(t-\tau)d\tau\right]e^{-j\omega t}dt$$

$$= \int_{-\infty}^{\infty} f_1(\tau)\left[\int_{-\infty}^{\infty} f_2(t-\tau)e^{-j\omega t}dt\right]d\tau$$

$$= \int_{-\infty}^{\infty} f_1(\tau)F_2(j\omega)e^{-j\omega\tau}d\tau$$

$$= F_2(j\omega)\int_{-\infty}^{\infty} f_1(\tau)e^{-j\omega\tau}d\tau \quad = F_1(j\omega)F_2(j\omega)$$

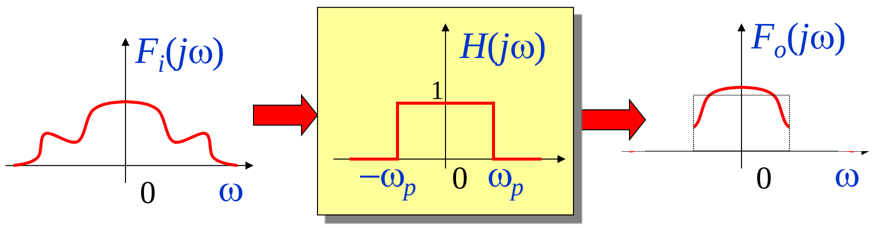
Time Domain convolution

Frequency Domain multiplication

Source: URI ELE436

Properties of Convolution

$$f_1(t) * f_2(t) \overset{\mathcal{F}}{\longleftrightarrow} F_1(j\omega)F_2(j\omega)$$

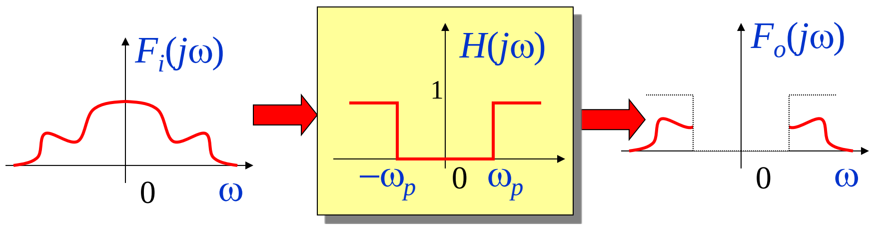


An Ideal Low-Pass Filter

Source: URI ELE436

Properties of Convolution

$$f_1(t) * f_2(t) \xleftrightarrow{\mathcal{F}} F_1(j\omega)F_2(j\omega)$$



An Ideal High-Pass Filter

Source: URI ELE436

Discrete Convolution

$$y[n] = x[n] * h[n] = \sum_{m=-\infty}^{\infty} x[m] \cdot h[n-m] = \sum_{m=-\infty}^{\infty} h[m] \cdot x[n-m]$$

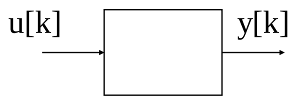
Consider this for the discrete case:

1. Rename the independent variable as **m**. You now have **x**[**m**] and **h**[**m**]. Flip **h**[**m**] over the origin. This is **h**[**-m**]
2. Shift **h**[**-m**] as far left as possible to a point "**n**", where the two signals barely touch. This is **h**[**n-m**]
3. Multiply the two signals and sum over all values of **m**. This is the convolution sum for the specific "**n**" picked above.
4. Shift / move **h**[**-m**] to the right by one sample, and obtain a new **h**[**n-m**]. Multiply and sum over all **m**.
5. 5. Repeat 2~4 until **h**[**n-m**] no longer overlaps with **x**[**m**], i.e., shifted out of the **x**[**m**] zone.

➔ The "*n*" dependency of *y*[*n*] deserves some care:
For each value of "*n*" the convolution sum must be computed ***separately*** over all values of a dummy variable "*m*".

Discrete-Time Systems & Discrete Convolution [1]

Will consider linear time-invariant (<u>LTI</u>) systems



<u>Linear</u> :

input u1[k] -> output y1[k]

input u2[k] -> output y2[k]

hence a.u1[k]+b.u2[k]-> a.y1[k]+b.y2[k]

<u>Time-invariant</u> (shift-invariant)

input u[k] -> output y[k]

hence input u[k-T] -> output y[k-T]

Discrete-Time Systems & Discrete Convolution [2]

Will consider <u>causal</u> systems

iff for all input signals with u[k]=0,k<0 -> output y[k]=0,k<0

Impulse response

K=0 K=0

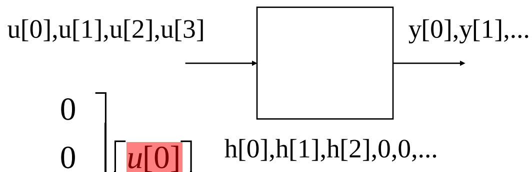
input …,0,0, 1 ,0,0,0,...-> output …,0,0, h[0] ,h[1],h[2],h[3],...

General input u[0],u[1],u[2],u[3] (cfr. linearity & shift-invariance!)

$$\begin{bmatrix} y[0] \\ y[1] \\ y[2] \\ y[3] \\ y[4] \\ y[5] \end{bmatrix} = \begin{bmatrix} h[0] & 0 & 0 & 0 \\ h[1] & h[0] & 0 & 0 \\ h[2] & h[1] & h[0] & 0 \\ 0 & h[2] & h[1] & h[0] \\ 0 & 0 & h[2] & h[1] \\ 0 & 0 & 0 & h[2] \end{bmatrix} . \begin{bmatrix} u[0] \\ u[1] \\ u[2] \\ u[3] \end{bmatrix}$$

this is called a `Toeplitz' matrix

Discrete-Time Systems & Discrete Convolution [3]



$$\begin{bmatrix} y[0] \\ y[1] \\ y[2] \\ y[3] \\ y[4] \\ y[5] \end{bmatrix} = \begin{bmatrix} h[0] & 0 & 0 & 0 \\ h[1] & h[0] & 0 & 0 \\ h[2] & h[1] & h[0] & 0 \\ 0 & h[2] & h[1] & h[0] \\ 0 & 0 & h[2] & h[1] \\ 0 & 0 & 0 & h[2] \end{bmatrix} . \begin{bmatrix} u[0] \\ u[1] \\ u[2] \\ u[3] \end{bmatrix}$$

$$y[k] = \sum_{\bar{k}} h[k-\bar{k}].u[\bar{k}] \overset{\Delta}{=} h[k] * u[k]$$

= `convolution sum'

(=more convenient than Toeplitz matrix notation when considering (infinitely) long input and impulse response sequences

Discrete-Time Systems & Discrete Convolution [4]

Z-Transform of <u>system</u> h[k] and <u>signals</u> u[k],y[k]

Definition:

$$H(z) \overset{\Delta}{=} \sum_k h[k].z^{-k} \quad U(z) \overset{\Delta}{=} \sum_k u[k].z^{-k} \quad Y(z) \overset{\Delta}{=} \sum_k y[k].z^{-k}$$

Input/output relation:

$$\underbrace{\begin{bmatrix} 1 & z^{-1} & z^{-2} & z^{-3} & z^{-4} & z^{-5} \end{bmatrix} \begin{bmatrix} y[0] \\ y[1] \\ y[2] \\ y[3] \\ y[4] \\ y[5] \end{bmatrix}}_{Y(z)} = \underbrace{\begin{bmatrix} 1 & z^{-1} & z^{-2} & z^{-3} & z^{-4} & z^{-5} \end{bmatrix} \begin{bmatrix} h[0] & 0 & 0 & 0 \\ h[1] & h[0] & 0 & 0 \\ h[2] & h[1] & h[0] & 0 \\ 0 & h[2] & h[1] & h[0] \\ 0 & 0 & h[2] & h[1] \\ 0 & 0 & 0 & h[2] \end{bmatrix}}_{H(z).\begin{bmatrix} 1 & z^{-1} & z^{-2} & z^{-3} \end{bmatrix}} . \begin{bmatrix} u[0] \\ u[1] \\ u[2] \\ u[3] \end{bmatrix}$$

$$\Rightarrow Y(z) = H(z).U(z)$$ H(z) is `transfer function'

Matrix Formulation of Convolution

$$\mathbf{y} = \mathbf{Hx}$$

$$\begin{bmatrix} 3 \\ 8 \\ 14 \\ 20 \\ 26 \\ 14 \\ 5 \end{bmatrix} = \begin{bmatrix} 1 & 2 & 3 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 2 & 3 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 2 & 3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 2 & 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 2 & 3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 2 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 3 \end{bmatrix} . \begin{bmatrix} 0 \\ 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 0 \\ 0 \end{bmatrix}$$

Toeplitz Matrix

Graphical Understanding of Convolution

➔ For c(τ)= $(f * g)(t) = \int_{-\infty}^{\infty} f(\tau) g(t-\tau)\, d\tau$:

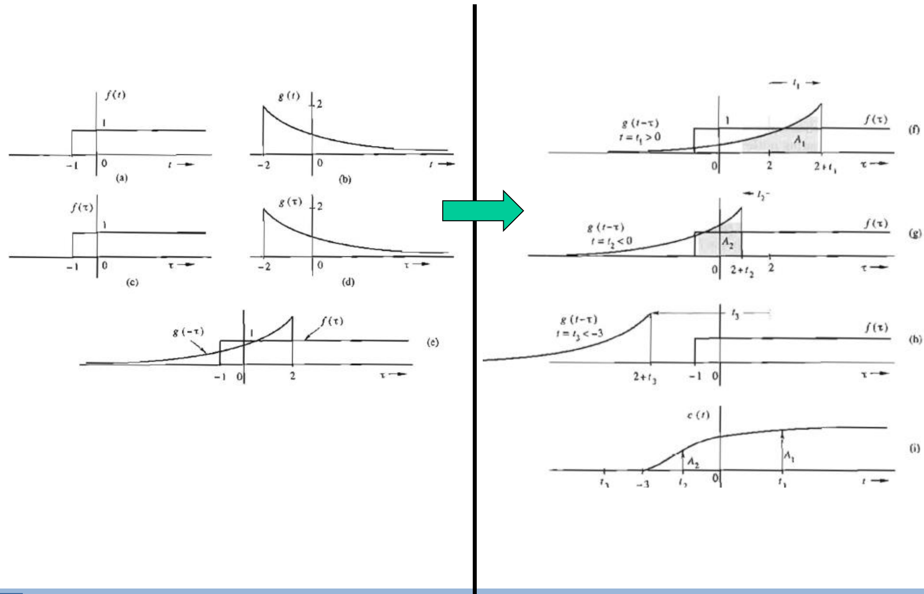
1. Keep the function ***f*** **(τ)** fixed
2. **Flip** (invert) the function ***g*****(τ)** about the vertical axis (τ=0)
   = this is ***g*****(-τ)**
3. **Shift** this frame (***g*****(-τ)**) along τ (horizontal axis) by $\mathbf{t_0}$.
   = this is ***g*****($\mathbf{t_0}$ -τ)**

➔ For c($\mathbf{t_0}$):

4. **c($\mathbf{t_0}$)** = the area under the product of ***f*** **(τ) and *g*($\mathbf{t_0}$ -τ)**

5. Repeat this procedure, shifting the frame by different values (positive and negative) to obtain **<u>c(t) for all values of t</u>**.

Graphical Understanding of Convolution (Ex)



Another View

e.g. convolution

x(n) = 1 2 3 4 5

h(n) = 3 2 1

x(k)	0 0 1 | 2 3 4 5	0 0 1 2 | 3 4 5	0 0 1 2 3 | 4 5	
h(n,k)	1 2 3 | 0 0 0 0	0 1 2 3 | 0 0 0	0 0 1 2 3 | 0 0	h(n-k)
y(n,k)	3	2 6	1 4 9	
y(n)	3	8	14	

Sum over all k

Notice the gain

Convolution & Systems

- Convolution system with input u (u(t) = 0, t < 0) and output y:

$$y(t) = \int_0^t h(\tau)u(t-\tau)\, d\tau = \int_0^t h(t-\tau)u(\tau)\, d\tau$$

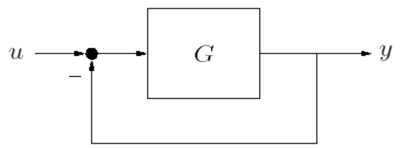
- abbreviated:

$$y = h * u$$

- in the frequency domain:

$$Y(s) = H(s)U(s)$$

Convolution & Feedback



- In the time domain:

$$y(t) = \int_0^t g(t-\tau)(u(\tau) - y(\tau))\, d\tau$$

- In the frequency domain:
  - Y=G(U-Y)

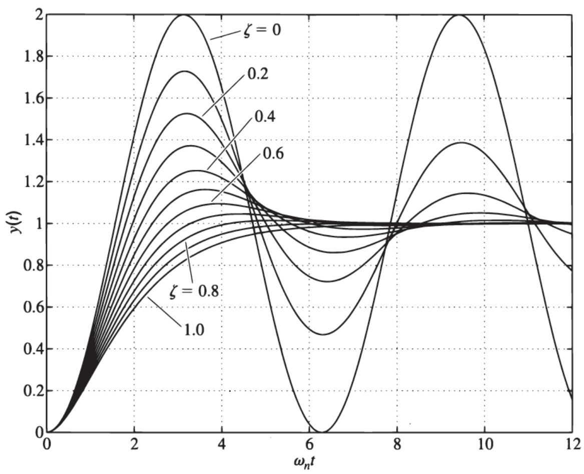
  ➔Y(s) = H(s)U(s)

$$H(s) = \frac{G(s)}{1 + G(s)}$$

2^{nd} Order LTID

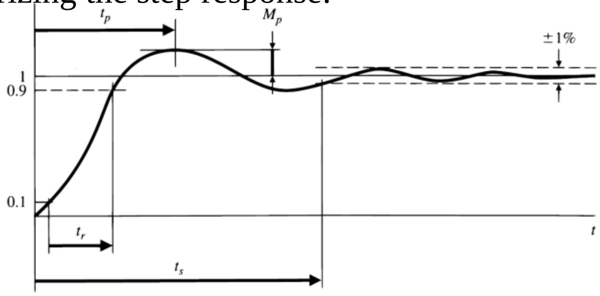
2^{nd} Order System Response

- Response of a 2^{nd} order system to increasing levels of damping:



2^{nd} Order System Specifications

Characterizing the step response:

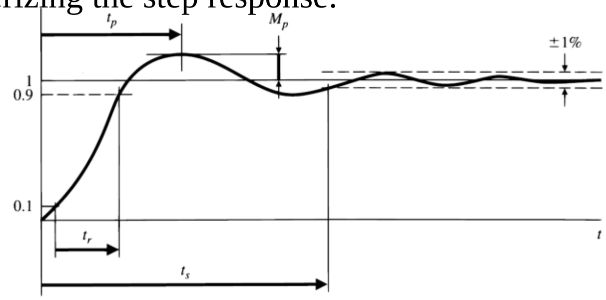


- Rise time (10% → 90%): $t_r \approx \frac{1.8}{\omega_0}$
- Overshoot: $M_p \approx \frac{e^{-\pi\zeta}}{\sqrt{1-\zeta^2}}$
- Settling time (to 1%): $t_s = \frac{4.6}{\zeta\omega_0}$
- Steady state error to unit step: e_{ss}
- Phase margin:

$$\phi_{PM} \approx 100\zeta$$

2^{nd} Order System Specifications

Characterizing the step response:

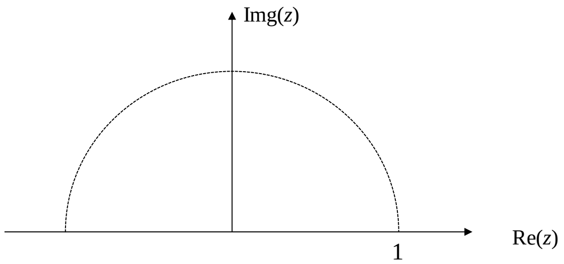


- Rise time (10% → 90%) & Overshoot:
  t_r, M_p → ζ, ω_0 : Locations of dominant poles
- Settling time (to 1%):
  t_s → radius of poles: $|z| < 0.01^{\frac{T}{t_s}}$
- Steady state error to unit step:
  e_{ss} → final value theorem $e_{ss} = \lim_{z\to 1}\{(z-1)F(z)\}$

The z-plane [for all pole systems]

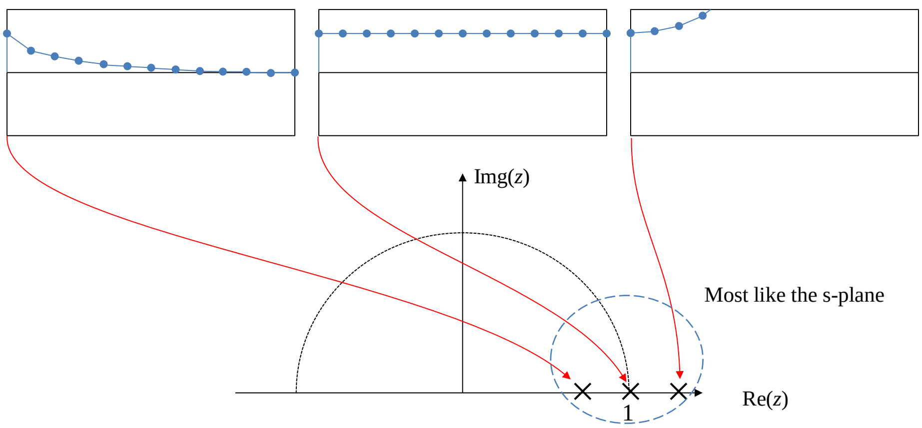
- We can understand system response by pole location in the *z*-plane

[Adapted from Franklin, Powell and Emami-Naeini]



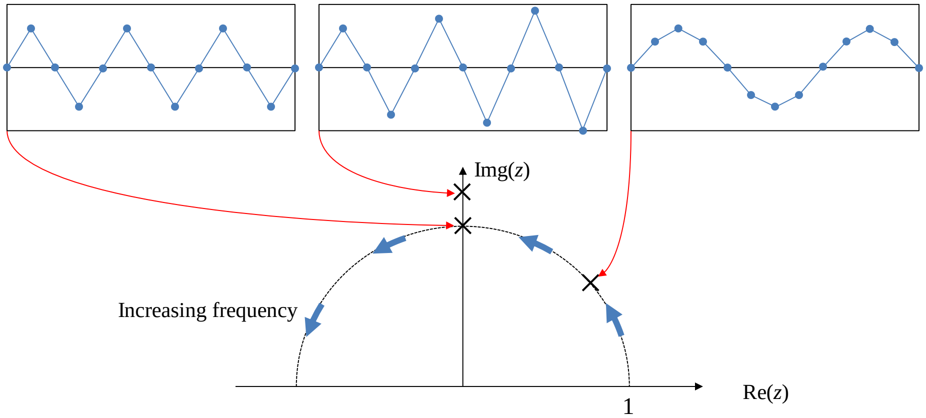
Effect of pole positions

- We can understand system response by pole location in the *z*-plane



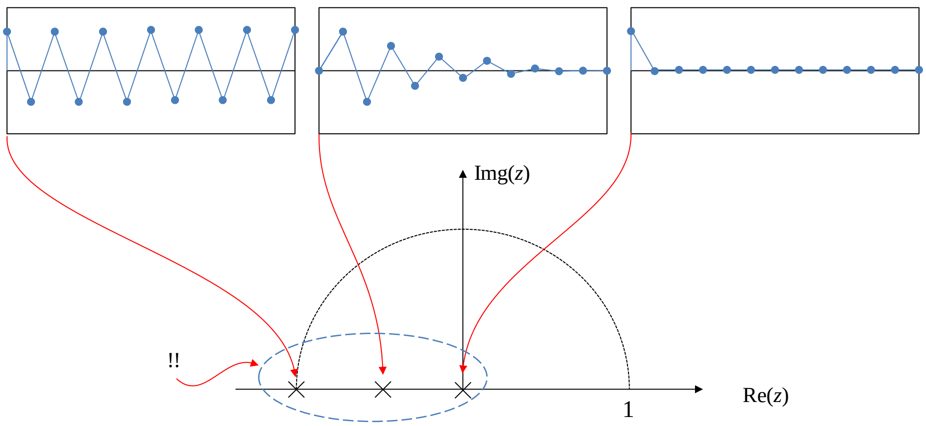
Effect of pole positions

- We can understand system response by pole location in the *z*-plane



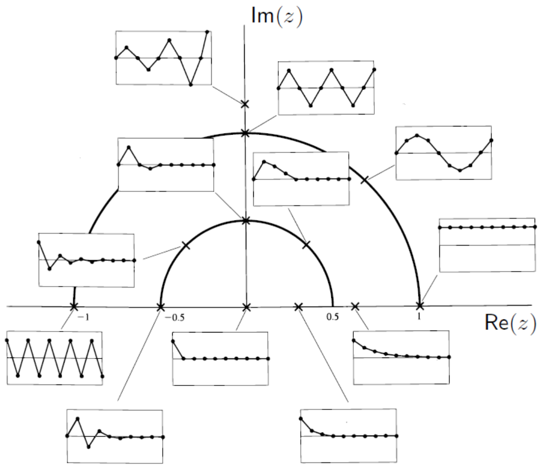
Effect of pole positions

- We can understand system response by pole location in the *z*-plane



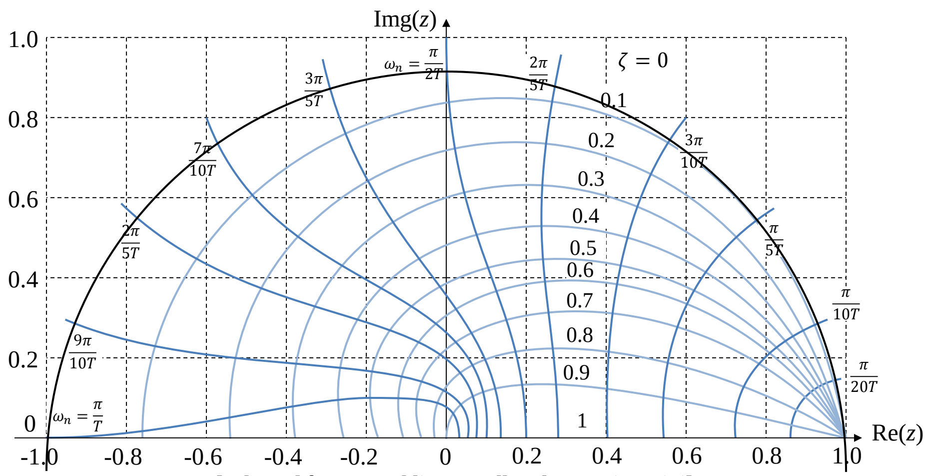
Pole positions in the z-plane

- Poles inside the unit circle are **stable**
- Poles outside the unit circle **unstable**
- Poles on the unit circle are oscillatory
- Real poles at 0 < z < 1 give exponential response
- Higher frequency of oscillation for larger
- Lower apparent damping for larer and r



Damping and natural frequency

$$z = e^{sT} \text{ where } s = -\zeta\omega_n \pm j\omega_n\sqrt{1-\zeta^2}$$



[Adapted from Franklin, Powell and Emami-Naeini]

Second Order Digital Systems

Consider the z-transform of a decaying exponential signal:

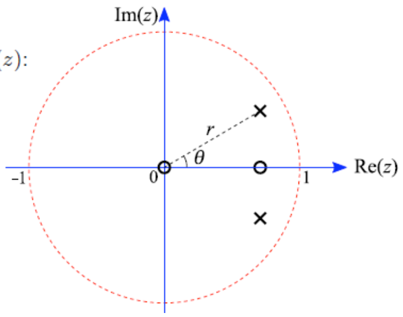
$y(t) = e^{-at}\cos(bt)\,\mathcal{U}(t)$ ($\mathcal{U}(t)$ = unit step)

- ⋆ sample: $y(kT) = r^k \cos(k\theta)\,\mathcal{U}(kT)$ with $r = e^{-aT}$ & $\theta = bT$
- ⋆ transform: $$Y(z) = \frac{1}{2}\frac{z}{(z - re^{j\theta})} + \frac{1}{2}\frac{z}{(z - re^{-j\theta})} = \frac{z(z - r\cos\theta)}{(z - re^{j\theta})(z - re^{-j\theta})}$$
- ⋆ e.g. y_k is the pulse response of $G(z)$:

$$G(z) = \frac{z(z - r\cos\theta)}{(z - re^{j\theta})(z - re^{-j\theta})}$$

poles: $\begin{cases} z = re^{j\theta} \\ z = re^{-j\theta} \end{cases}$

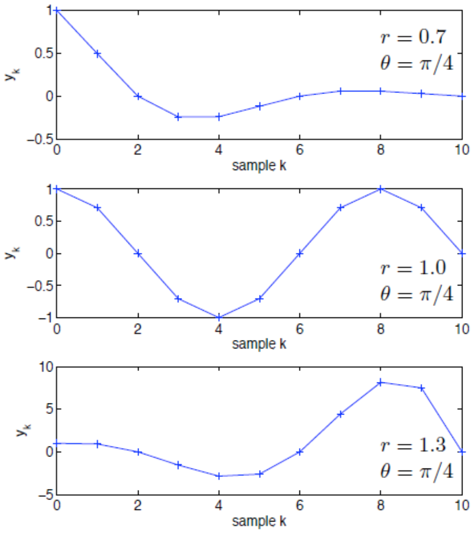
zeros: $\begin{cases} z = 0 \\ z = r\cos\theta \end{cases}$



Response of 2nd order system [1/3]

Responses for varying r:

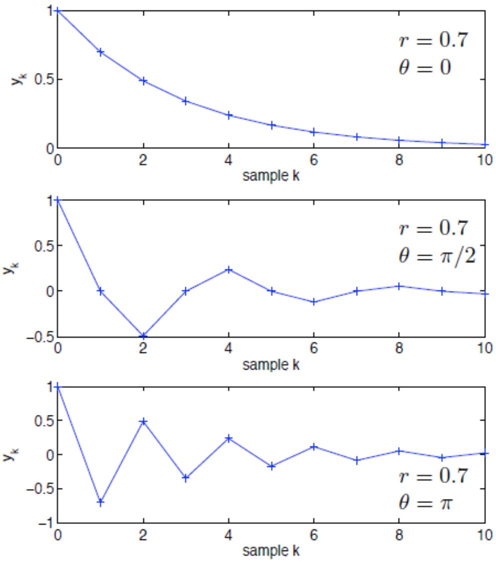
- ▷ $r < 1$
  ⇓
  exponentially decaying envelope
- ▷ $r = 1$
  ⇓
  sinusoidal response with $2\pi/\theta$ samples per period
- ▷ $r > 1$
  ⇓
  exponentially increasing envelope



Response of 2nd order system [2/3]

Responses for varying θ:

- $\theta = 0$
  $\Downarrow$
  decaying exponential
- $\theta = \pi/2$
  $\Downarrow$
  $2\pi/\theta = 4$ samples per period
- $\theta = \pi$
  $\Downarrow$
  2 samples per period



Response of 2nd order system [3/3]

Some special cases:

- for $\theta = 0$, $Y(z)$ simplifies to:

$$Y(z) = \frac{z}{z-r}$$

$\Longrightarrow$ exponentially decaying response

- when $\theta = 0$ and $r = 1$:

$$Y(z) = \frac{z}{z-1}$$

$\Longrightarrow$ unit step

- when $r = 0$:

$$Y(z) = 1$$

$\Longrightarrow$ unit pulse

- when $\theta = 0$ and $-1 < r < 0$:

samples of alternating signs

Discrete-time transfer function

take $\mathcal{Z}$-transform of system equations

$$x(t+1) = Ax(t) + Bu(t), \qquad y(t) = Cx(t) + Du(t)$$

yields

$$zX(z) - zx(0) = AX(z) + BU(z), \qquad Y(z) = CX(z) + DU(z)$$

solve for $X(z)$ to get

$$X(z) = (zI - A)^{-1}zx(0) + (zI - A)^{-1}BU(z)$$

(note extra z in first term!)

hence

$$Y(z) = H(z)U(z) + C(zI - A)^{-1}zx(0)$$

where $H(z) = C(zI - A)^{-1}B + D$ is the *discrete-time transfer function*

note power series expansion of resolvent:

$$(zI - A)^{-1} = z^{-1}I + z^{-2}A + z^{-3}A^2 + \cdots$$

Source: Boyd, Lecture Notes for EE263, 13-39

Ex: System Specifications → Control Design [1/4]

Design a controller for a system with:

- A continuous transfer function: $G(s) = \frac{0.1}{s(s+0.1)}$
- A discrete ZOH sampler
- Sampling time (T_s): T_s= 1s
- Controller:

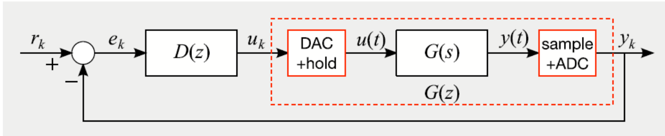
$$u_k = -0.5u_{k-1} + 13(e_k - 0.88e_{k-1})$$

The closed loop system is required to have:

- $M_p < 16\%$
- $t_s < 10$ s
- $e_{ss} < 1$

Ex: System Specifications → Control Design [2/4]

1. (a) Find the pulse transfer function of $G(s)$ plus the ZOH



$$G(z) = (1 - z^{-1})\mathcal{Z}\left\{\frac{G(s)}{s}\right\} = \frac{(z-1)}{z}\mathcal{Z}\left\{\frac{0.1}{s^2(s+0.1)}\right\}$$

e.g. look up $\mathcal{Z}\{a/s^2(s+a)\}$ in tables:

$$G(z) = \frac{(z-1)}{z}\frac{z\Big((0.1 - 1 + e^{-0.1})z + (1 - e^{-0.1} - 0.1e^{-0.1})\Big)}{0.1(z-1)^2(z-e^{-0.1})}$$
$$= \frac{0.0484(z+0.9672)}{(z-1)(z-0.9048)}$$

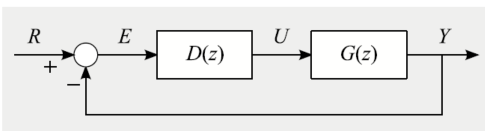
(b) Find the controller transfer function (using z = shift operator):

$$\frac{U(z)}{E(z)} = D(z) = 13\frac{(1-0.88z^{-1})}{(1+0.5z^{-1})} = 13\frac{(z-0.88)}{(z+0.5)}$$

Ex: System Specifications → Control Design [3/4]

2. Check the steady state error e_{ss} when r_k = unit ramp

$$e_{ss} = \lim_{k\to\infty} e_k = \lim_{z\to 1}(z-1)E(z)$$



$$\frac{E(z)}{R(z)} = \frac{1}{1+D(z)G(z)}$$

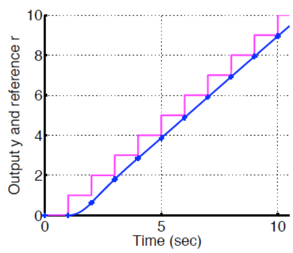
$$R(z) = \frac{Tz}{(z-1)^2}$$

so $$e_{ss} = \lim_{z\to 1}\left\{(z-1)\frac{Tz}{(z-1)^2}\frac{1}{1+D(z)G(z)}\right\} = \lim_{z\to 1}\frac{T}{(z-1)D(z)G(z)}$$

$$= \lim_{z\to 1}\frac{T}{(z-1)\frac{0.0484(z+0.9672)}{(z-1)(z-0.9048)}D(1)}$$

$$= \frac{1-0.9048}{0.0484(1+0.9672)D(1)} = 0.96$$

$\Longrightarrow \quad e_{ss} < 1$ (as required)



Ex: System Specifications → Control Design [4/4]

3. Step response: overshoot $M_p < 16\% \implies \zeta > 0.5$

settling time $t_s < 10 \implies |z| < 0.01^{1/10} = 0.63$

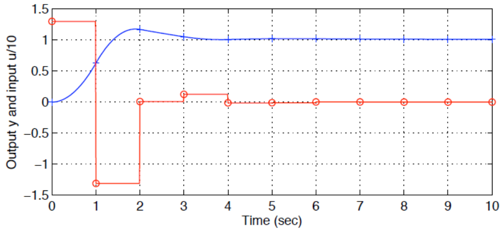
The closed loop poles are the roots of $1 + D(z)G(z) = 0$, i.e.

$$1 + 13\frac{(z-0.88)}{(z+0.5)}\frac{0.0484(z+0.9672)}{(z-1)(z-0.9048)} = 0$$

$$\implies z = 0.88,\ -0.050 \pm j0.304$$

But the pole at $z = 0.88$ is cancelled by controller zero at $z = 0.88$, and

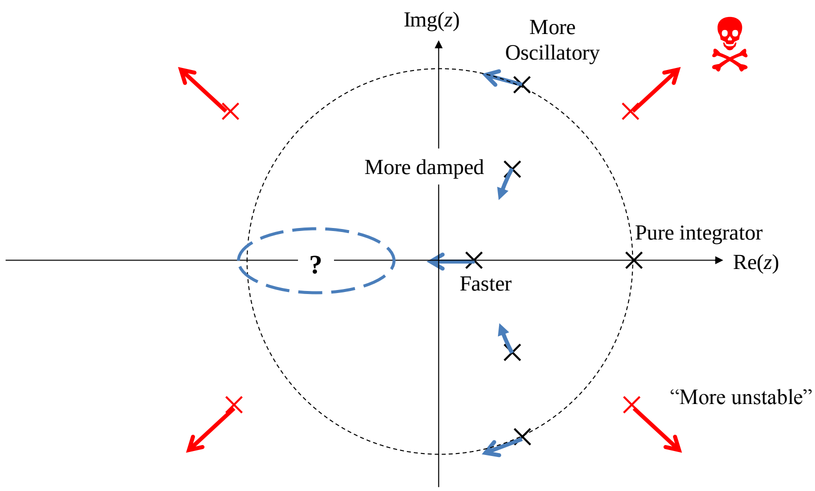
$$z = -0.050 \pm j0.304 = re^{\pm j\theta} \implies \begin{cases} r = 0.31,\ \theta = 1.73 \\ \zeta = 0.56 \end{cases}$$



all specs satisfied!

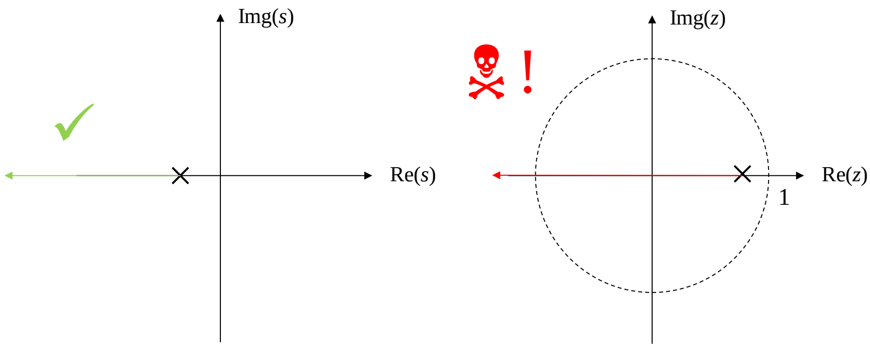
Recall dynamic responses

- Ditto the z-plane:



Deep insight #2

- Gains that stabilise continuous systems can actually ***destabilise*** digital systems!

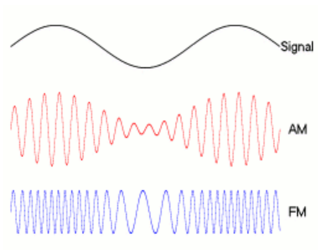


Ex: Modulation

Modulation

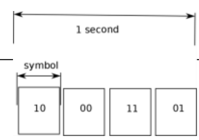
Analog Methods:

- AM - Amplitude modulation
  - Amplitude of a (carrier) is modulated to the (data)
- FM - Frequency modulation
  - Frequency of a (carrier) signal is varied in accordance to the amplitude of the (data) signal
- PM – Phase Modulation



Source: http://en.wikipedia.org/wiki/Modulation

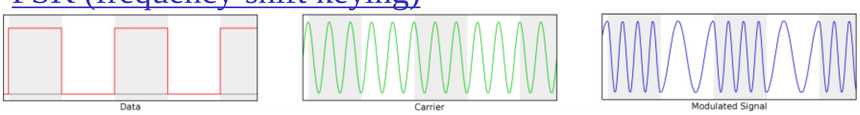
Modulation [Digital Methods]



Start with a “**symbol**” & place it on a channel

- ASK (amplitude-shift keying)

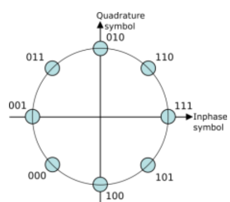
- FSK (frequency-shift keying)



- PSK (phase-shift keying)
- QAM (quadrature amplitude modulation)

$s(t) = A \cdot cos(\omega_c + \phi_i(t))$

$= x_i(t)\cos(\omega_c t) + x_q(t)\sin(\omega_c t)$



Source: http://en.wikipedia.org/wiki/Modulation | http://users.ecs.soton.ac.uk/sqc/EL334 | http://en.wikipedia.org/wiki/Constellation_diagram

Modulation [Example – V.32bis Modem]

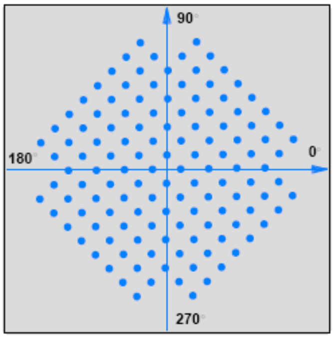
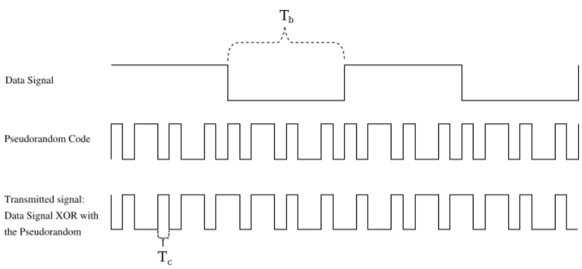


Figure 10.13 Illustration of the QAM constellation for a V.32bis dialup modem.

Source: Computer Networks and Internets, 5e, Douglas E. Comer

Multiple Access (Channel Access Method)

- Send multiple signals on 1 to N channel(s)
  - Frequency-division multiple access (FDMA)
  - Time-division multiple access (TDMA)
  - Code division multiple access (CDMA)
  - Space division multiple access (SDMA)
- CDMA:
  - Start with a pseudorandom code (the noise doesn't know your code)



Source: http://en.wikipedia.org/wiki/Code_division_multiple_access

Next Time…

- **Digital Systems**

- Review:
  - Chapter 8 of Lathi

- A signal has many signals ☺
  [Unless it's bandlimited. Then there is the one ω]