z Transform
(Digital Systems Made eZ)

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

Drs. Surya Singh and Paul Pounds
(Most of the slides are from Paul – **Thanks Paul!**)

Lecture 11

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

April 12, 2012

Today:

Week	Date	Lecture Title
1	27-Feb	Introduction
	1-Mar	Systems Overview
2	6-Mar	Signals & Signal Models
	8-Mar	System Models
3	13-Mar	Linear Dynamical Systems
	15-Mar	Sampling & Data Acquisition
4	20-Mar	Time Domain Analysis of Continuous Time Systems
	22-Mar	System Behaviour & Stability
5	27-Mar	Signal Representation
	29-Mar	Holiday
6	10-Apr	Frequency Response
	12-Apr	z-Transform
7	17-Apr	Analog Filters & IIR Systems
	19-Apr	FIR Systems
8	24-Apr	Discrete-Time Signals
	26-Apr	Discrete-Time Systems
9	1-May	Digital Filters
	3-May	Fourier Transform & DTFT
10	8-May	State-Space
	10-May	Controllability & Observability
11	15-May	Introduction to Digital Control
	17-May	Stability of Digital Systems
12	22-May	PID & Computer Control
	24-May	Information Theory & Communications
13	29-May	Applications in Industry
	31-May	Summary and Course Review

New!
Revised order

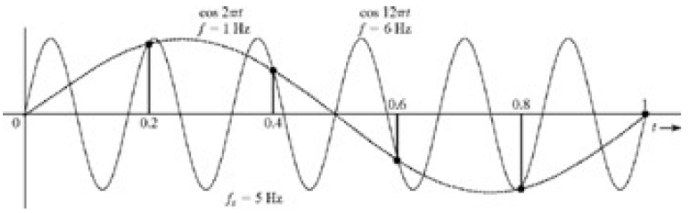
Announcements:

- Assignment 1 Solutions:
  - Questions 1-5:
    - “all or nothing”
    - Okay… **0 or 3 or 5 points**
      (+ 1-2 bonus for something truly exceptional)
- Lab 2 (Experiment 3)
  - Is to give you a “feeling” / “intuition” for digital systems
  - Feels a little “ahead” of it’s time

  ➔We are rerunning it again next week ☺
  (for those who want to do it “post-theory”)
- Lab 3 (Experiment 4)
  - **Will run on Week 9!**

  ∵ Week 8 has the ANZAC holiday

Refresher:
Aliasing & Sampling

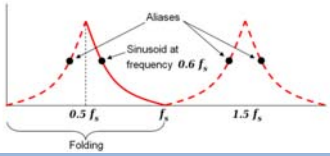


- Nyquist:

$$f_h < \frac{f_s}{2}$$

- Spectral Folding:

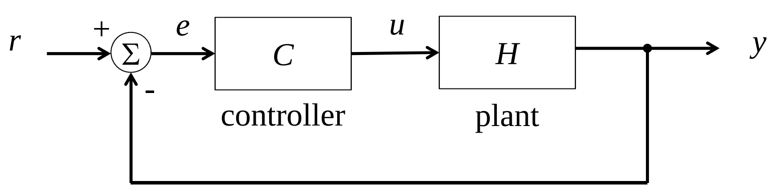
$$f_{\text{image}}(N) = f - N f_s.$$



Quick Background: Pole-Zero Diagrams & The Root Locus

- The transfer function for a closed-loop system can be easily calculated:

$$y = CH(r - y)$$
$$y + CHy = CHr$$
$$\therefore \frac{y}{r} = \frac{CH}{1 + CH}$$



Quick Background: Pole-Zero Diagrams & The Root Locus

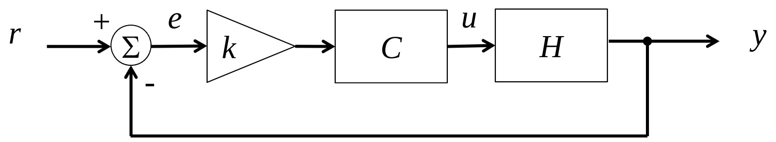
- We often care about the effect of increasing gain of a control compensator design:

$$\frac{y}{r} = \frac{kCH}{1 + kCH}$$

Multiplying by denominator:

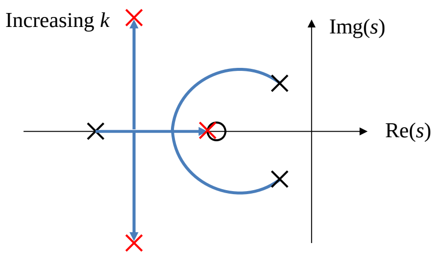
$$\frac{y}{r} = \frac{kC_n H_n}{C_d H_d + kCnHn}$$

characteristic polynomial



Quick Background:
Pole-Zero Diagrams & The Root Locus

- Pole positions change with increasing gain
  - The trajectory of **closed-loop poles** on the pole-zero plot with changing k is called the "root locus"
  - This is sometimes quite complex



(In practice you'd plot these with computers)

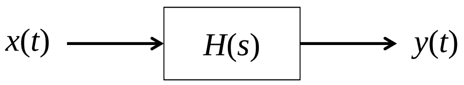
Coping with Complexity

- Transfer functions help control complexity
  - Recall the Laplace transform:

$$\mathcal{L}\{f(t)\} = \int_0^\infty f(t)e^{-st}dt = F(s)$$

where

$$\mathcal{L}\{\dot{f}(t)\} = sF(s)$$



Is there a something similar for sampled systems?

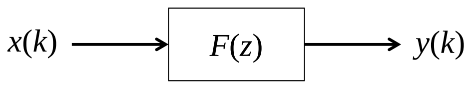
The z-transform

- The discrete equivalent is the z-Transform†:

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

and

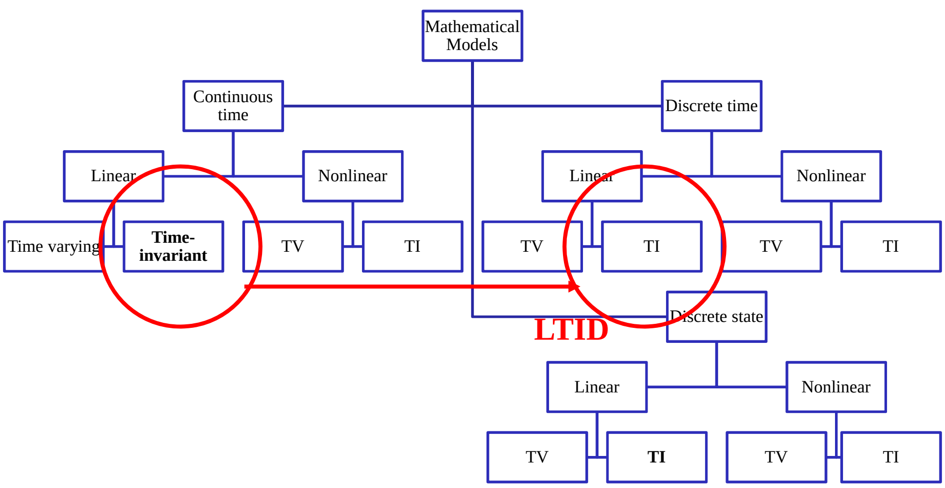
$$\mathcal{Z}\{f(k-1)\} = z^{-1} F(z)$$



Convenient!

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

What about the Discrete Domain? [Lecture 4-Slide 10]



The z-transform

- Some useful properties
  - **Delay by n samples**: $\mathcal{Z}\{f(k-n)\} = z^{-n}F(z)$
  - **Linear**: $\mathcal{Z}\{af(k) + bg(k)\} = aF(z) + bG(z)$
  - **Convolution**: $\mathcal{Z}\{f(k) * g(k)\} = F(z)G(z)$

So, all those block diagram manipulation tools you know and love will work just the same!

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

Final value theorem

- An important question: what is the steady-state output a stable system at $t = \infty$?
  - For continuous systems, this is found by:
  - $\lim_{t\to\infty} x(t) = \lim_{s\to 0} sX(s)$
  - The discrete equivalent is:
  - $\lim_{k\to\infty} x(k) = \lim_{z\to 1} (1 - z^{-1})X(z)$
  - (Provided the system is stable)

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?

Consider the simplest system

- Take a first-order response:

$$f(t) = e^{-at} \Rightarrow \mathcal{L}\{f(t)\} = \frac{1}{s+a}$$

- The discrete version is:

$$f(kT) = e^{-akT} \Rightarrow \mathcal{Z}\{f(k)\} = \frac{z}{z - e^{-aT}}$$

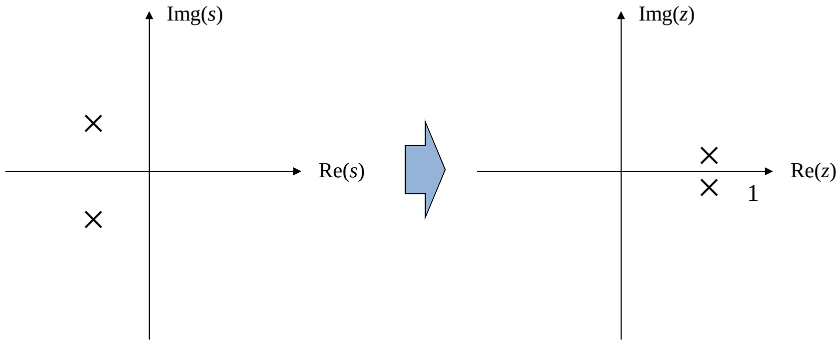
The equivalent system poles are related by

$$z = e^{sT}$$

That sounds somewhat profound… but what does it mean?

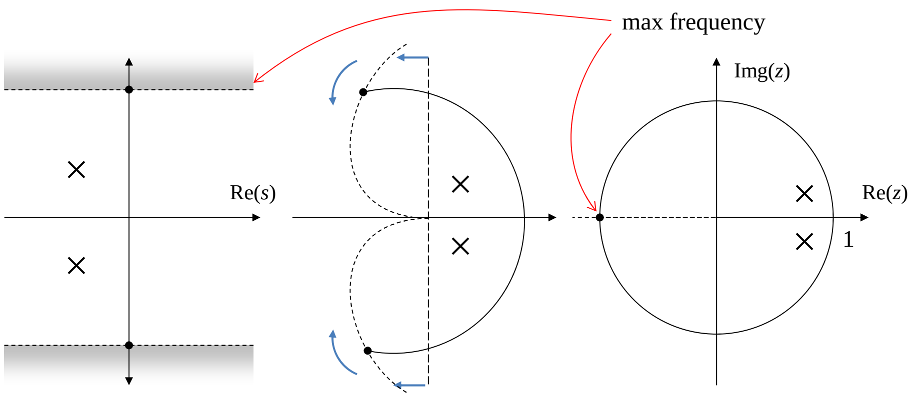
The *z*-Plane

- *z*-domain poles and zeros can be plotted just like *s*-domain poles and zeros:



Deep insight #1

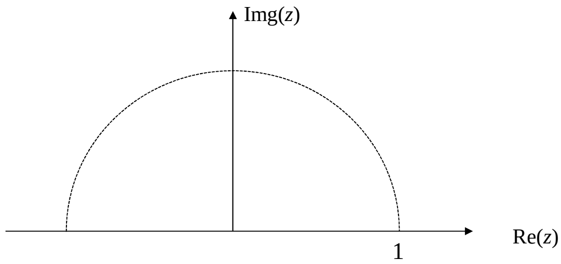
The mapping between continuous and discrete poles and zeros acts like a distortion of the plane



The *z*-plane

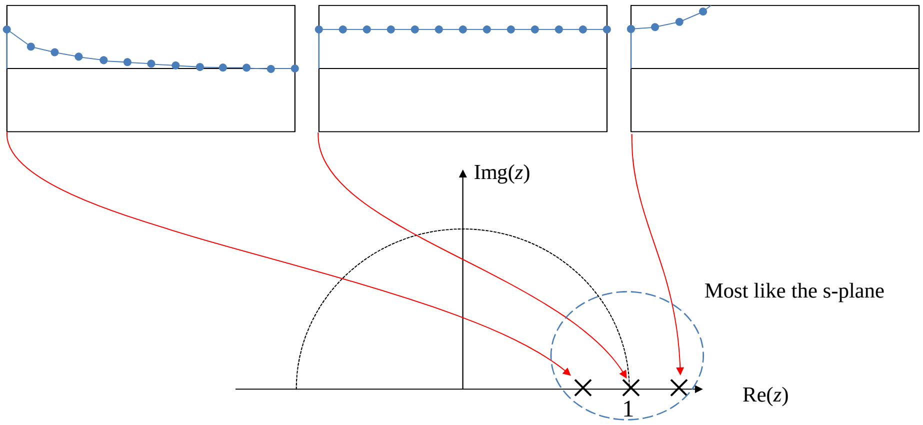
- We can understand system response by pole location in the *z*-plane

[Adapted from Franklin, Powell and Emami-Naeini]



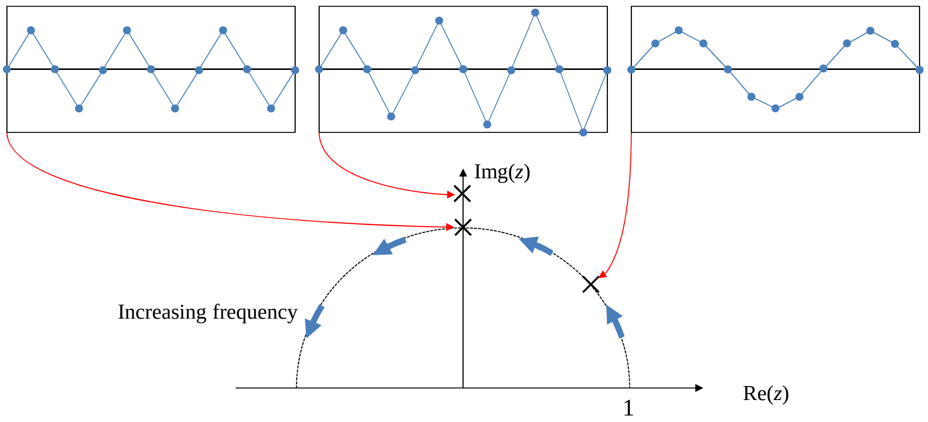
Effect of pole positions

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



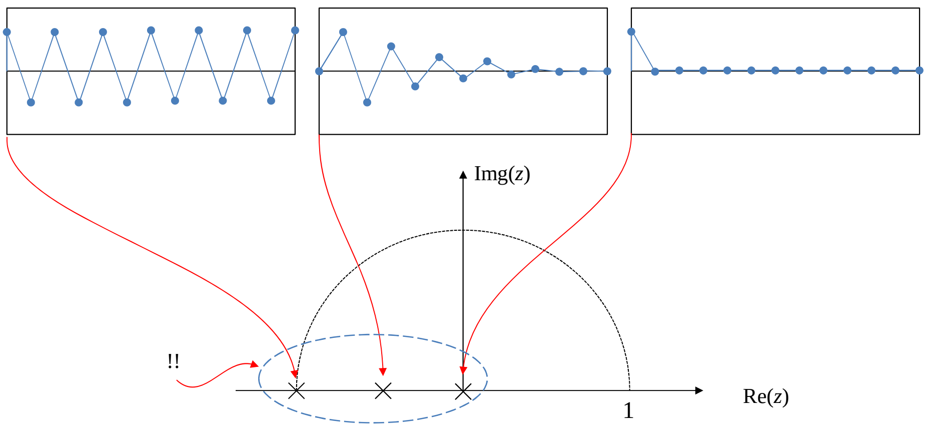
Effect of pole positions

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



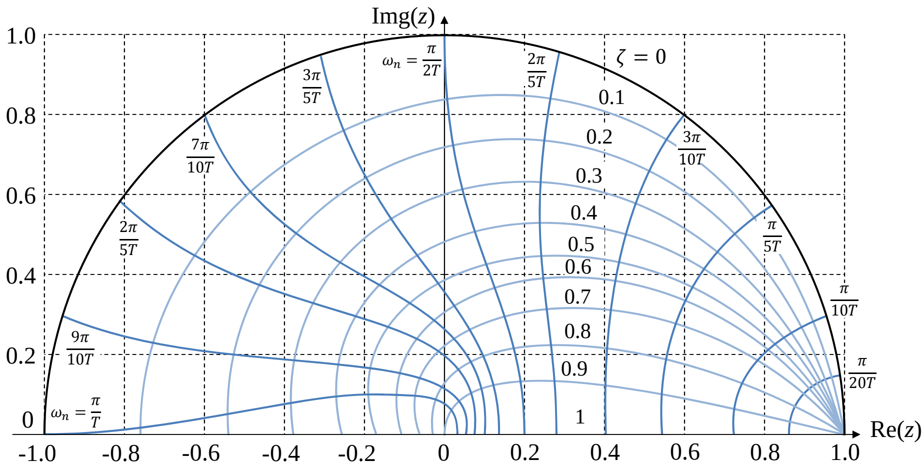
Effect of pole positions

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



Damping and natural frequency

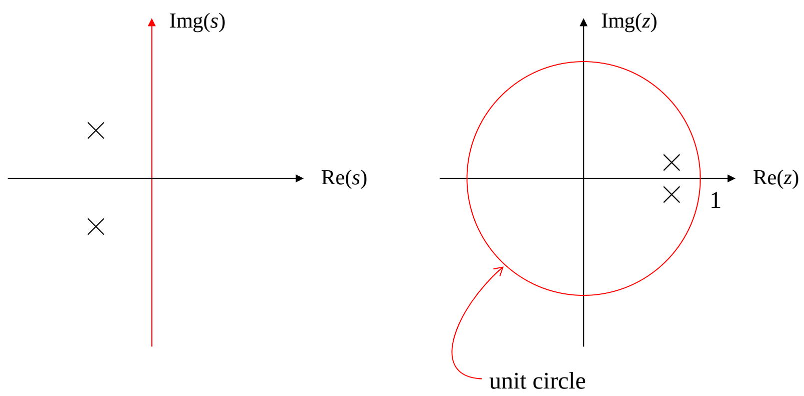
$$z = e^{sT} \text{ where } s = -\zeta\omega_n \pm j\omega_n\sqrt{1-\zeta^2}$$



[Adapted from Franklin, Powell and Emami-Naeini]

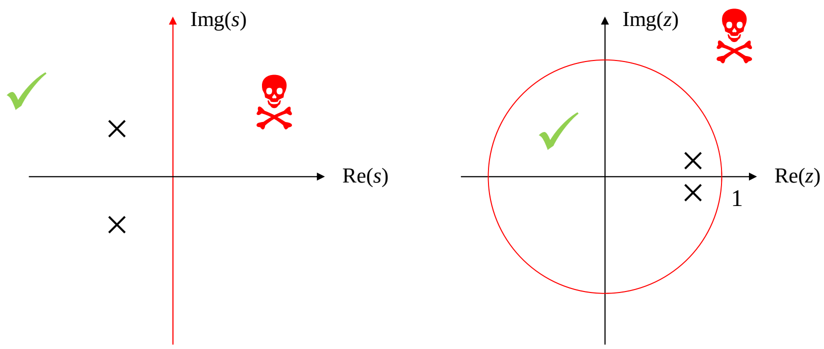
z-plane stability

- In the z-domain, the unit circle is the system stability bound



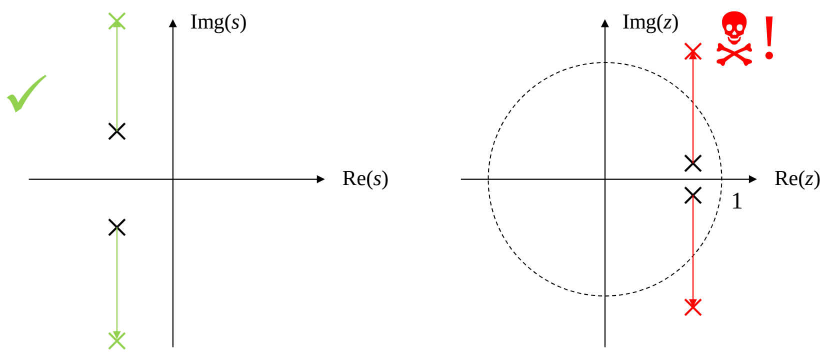
z-plane stability

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



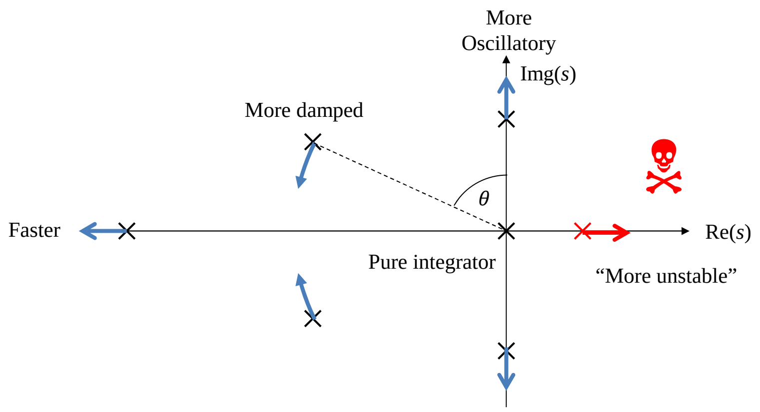
z-plane stability

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



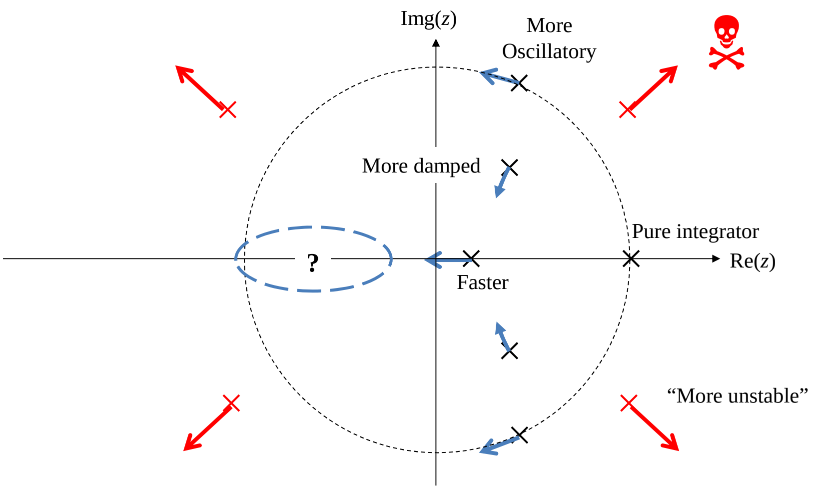
Recall dynamic responses

- Moving pole positions change system response characteristics



Recall dynamic responses

- Ditto the z-plane:



Deep insight #2

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