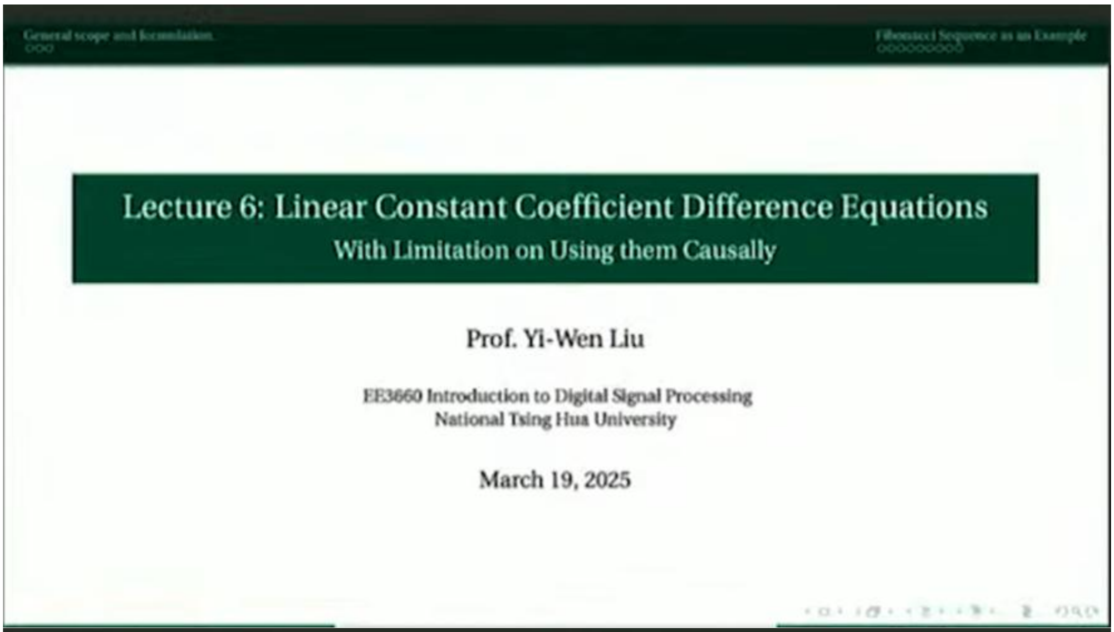
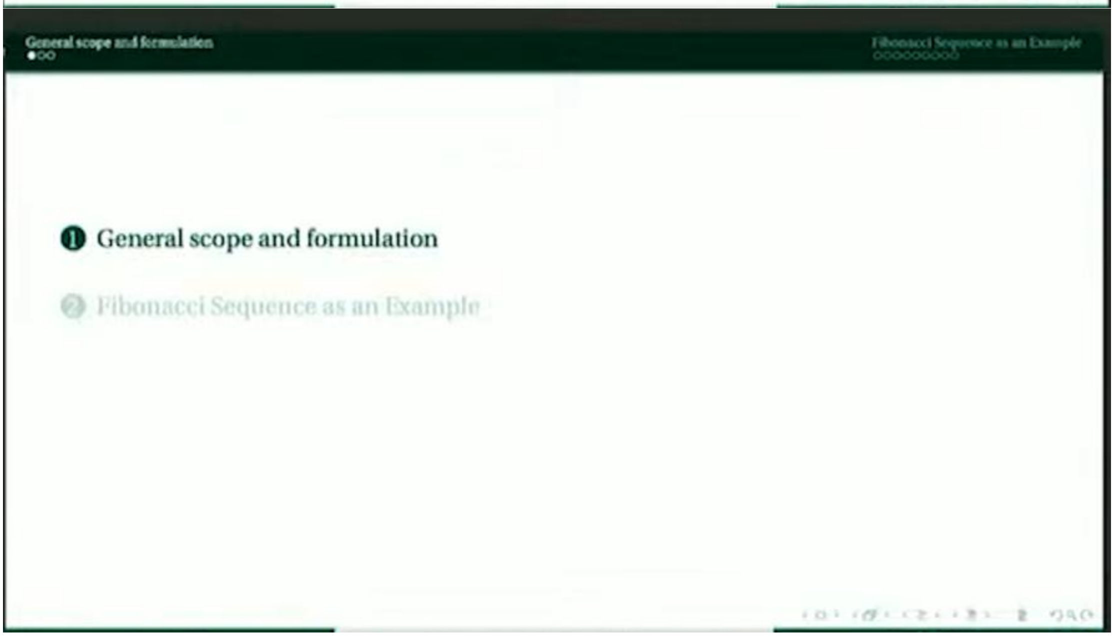
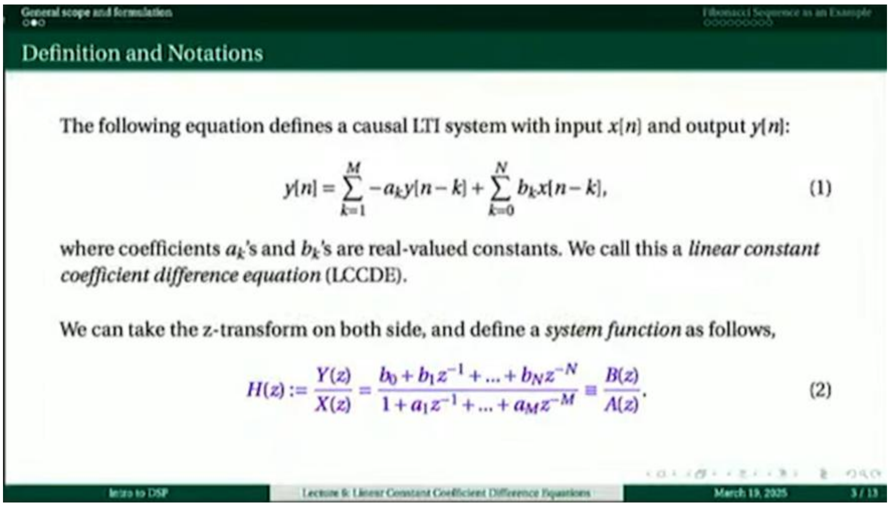
Lecture6 Linear Constant Coefficient Difference Equations With Limitation on Using them Causally







Definition and Notations

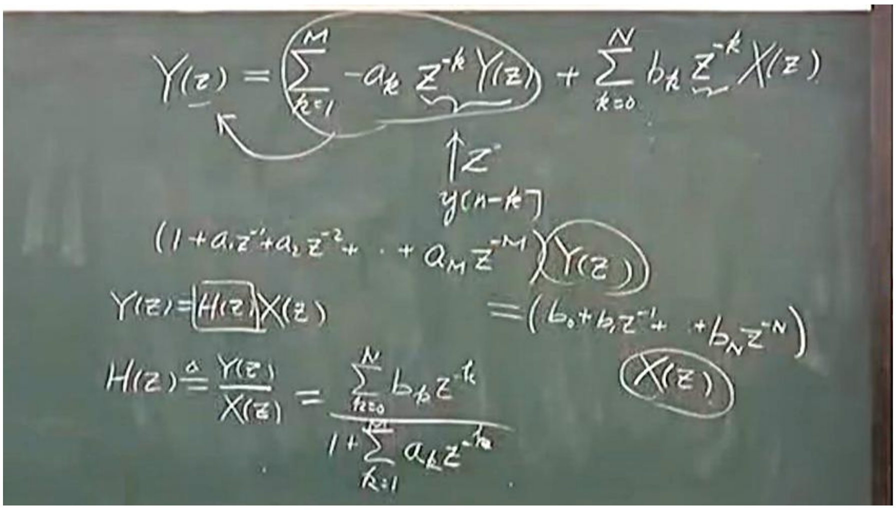
The following equation defines a causal LTI system with input $x[n]$ and output $y[n]$:

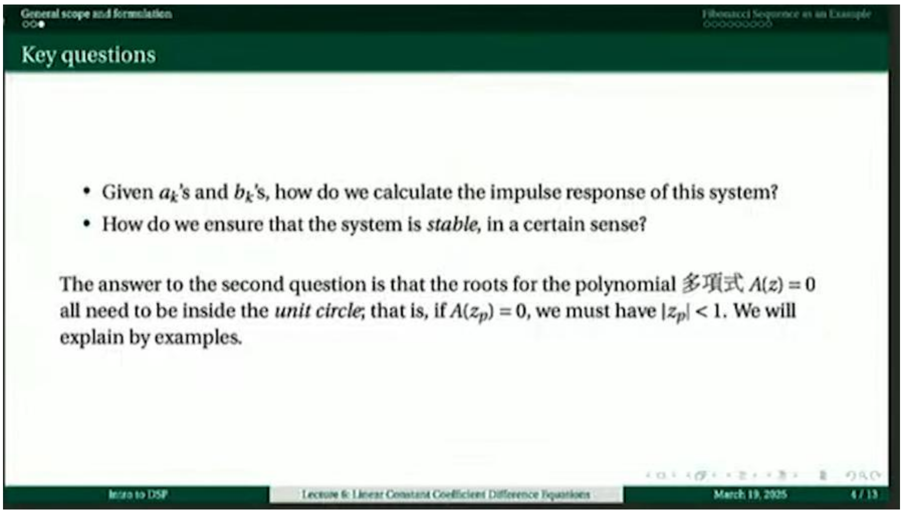
$$y[n] = \sum_{k=1}^{M} -a_k y[n-k] + \sum_{k=0}^{N} b_k x[n-k], \tag{1}$$

where coefficients a_k's and b_k's are real-valued constants. We call this a *linear constant coefficient difference equation* (LCCDE).

We can take the z-transform on both side, and define a *system function* as follows,

$$H(z) := \frac{Y(z)}{X(z)} = \frac{b_0 + b_1 z^{-1} + \ldots + b_N z^{-N}}{1 + a_1 z^{-1} + \ldots + a_M z^{-M}} \equiv \frac{B(z)}{A(z)}. \tag{2}$$

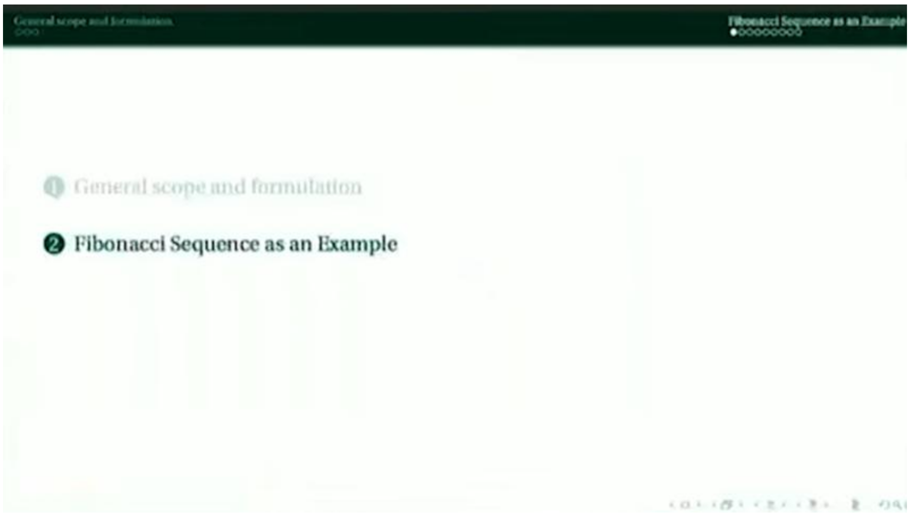


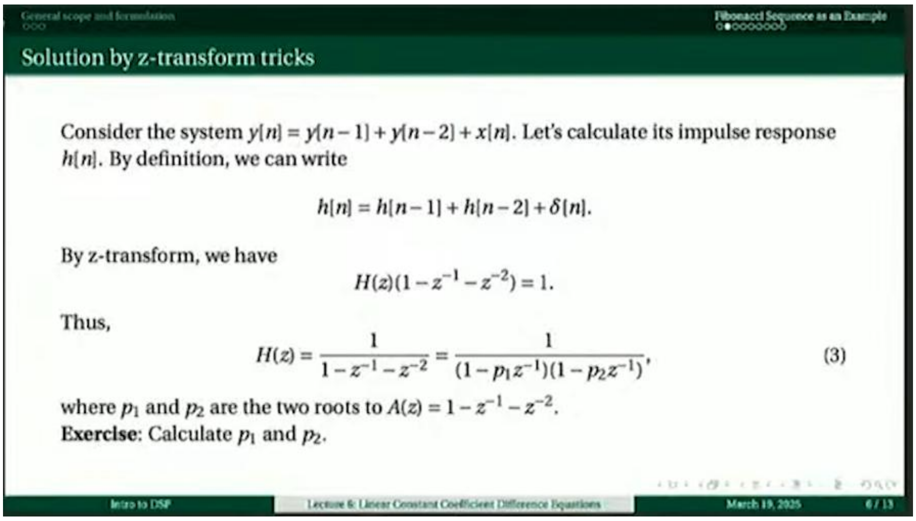


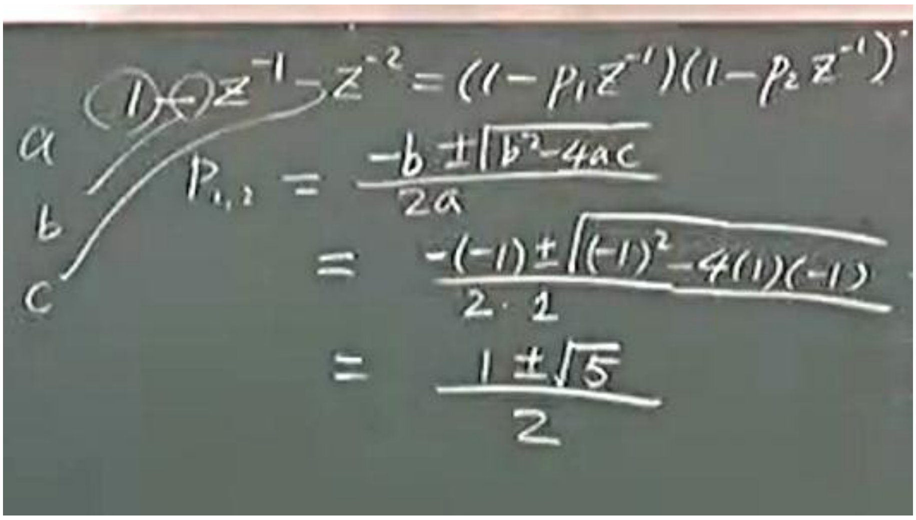
Key questions

- Given a_k's and b_k's, how do we calculate the impulse response of this system?
- How do we ensure that the system is *stable*, in a certain sense?

The answer to the second question is that the roots for the polynomial 多項式 $A(z) = 0$ all need to be inside the *unit circle*; that is, if $A(z_p) = 0$, we must have $|z_p| < 1$. We will explain by examples.


General scope and formulation
Fibonacci Sequence as an Example
1 General scope and formulation
2 Fibonacci Sequence as an Example


General scope and formulation
Fibonacci Sequence as an Example
Solution by z-transform tricks
Consider the system $y[n] = y[n-1] + y[n-2] + x[n]$. Let's calculate its impulse response $h[n]$. By definition, we can write
$$h[n] = h[n-1] + h[n-2] + \delta[n].$$
By z-transform, we have
$$H(z)(1 - z^{-1} - z^{-2}) = 1.$$
Thus,
$$H(z) = \frac{1}{1 - z^{-1} - z^{-2}} = \frac{1}{(1 - p_1 z^{-1})(1 - p_2 z^{-1})}, \quad (3)$$
where p_1 and p_2 are the two roots to $A(z) = 1 - z^{-1} - z^{-2}$.
Exercise: Calculate p_1 and p_2.
Intro to DSP
Lecture 6: Linear Constant Coefficient Difference Equations
March 19, 2025
6 / 13


$(1) - (-1)z^{-1} - z^{-2} = (1 - p_1 z^{-1})(1 - p_2 z^{-1})$
a
b
c
$p_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$
$= \frac{-(-1) \pm \sqrt{(-1)^2 - 4(1)(-1)}}{2 \cdot 1}$
$= \frac{1 \pm \sqrt{5}}{2}$

Solution by z-transform tricks (Cont'd)

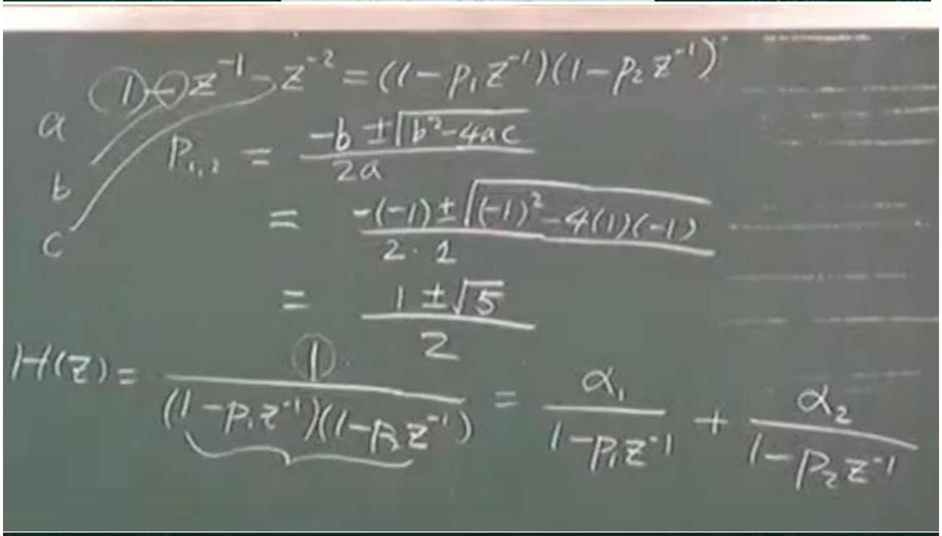
The next trick is to perform *partial fractional decomposition* — if $A(z)$ does not have any *multiple root* 重根 and $B(z)$'s *order* 階數 is lower than that of $A(z)$, then $B(z)/A(z)$ can be written in the following form,

$$\frac{B(z)}{A(z)} = \sum_{k=1}^{M} \frac{\alpha_k}{1 - p_k z^{-1}},$$

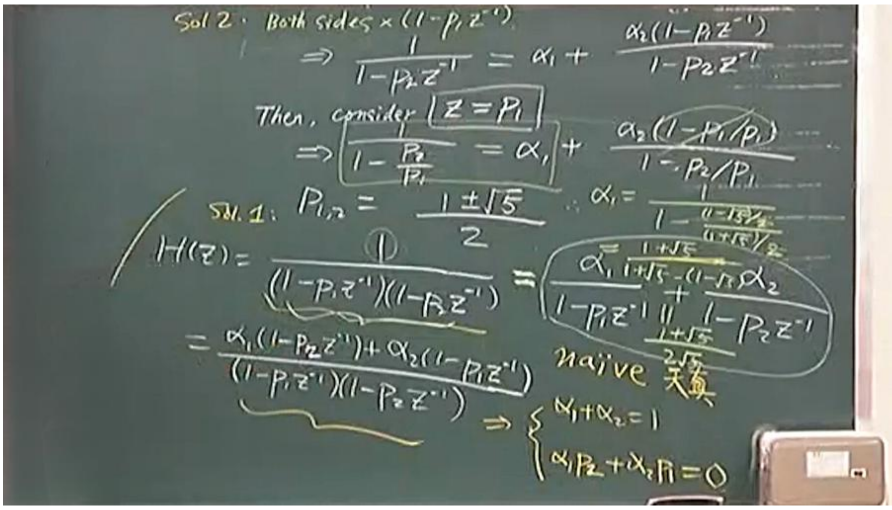
where p_k's are the roots of $A(z)$ and α_k's need to be determined.[1]

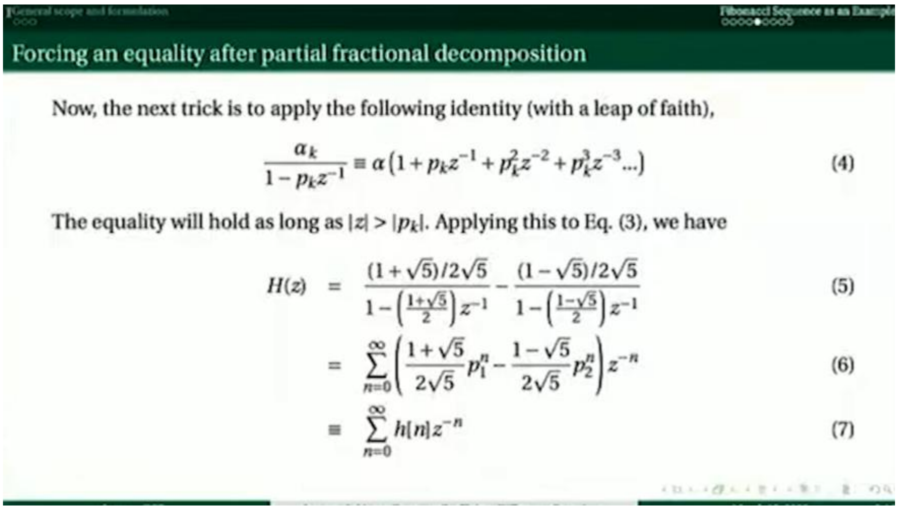
Exercise: Calculate α_k's for the case $B(z) = 1$ and $A(z) = 1 - z^{-1} - z^{-2}$.

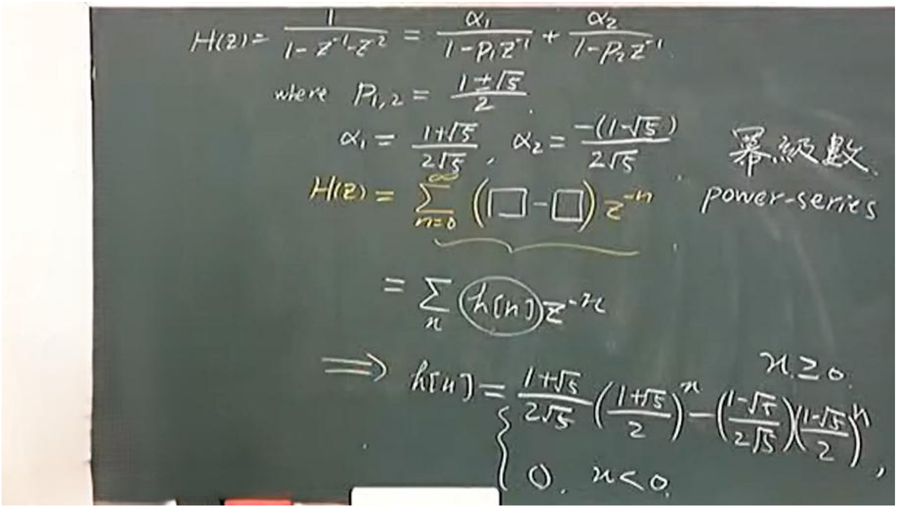
[1] The situation is a bit more complicated if $A(z)$ has any multiple roots.

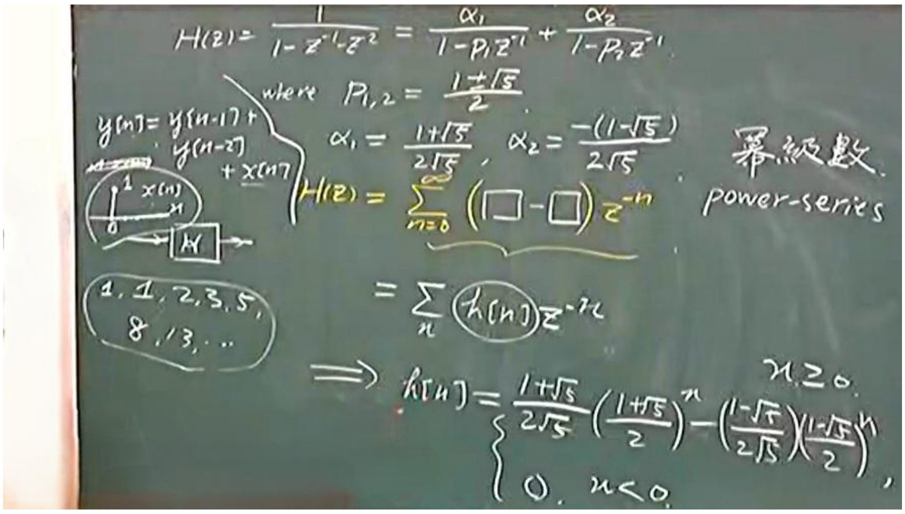


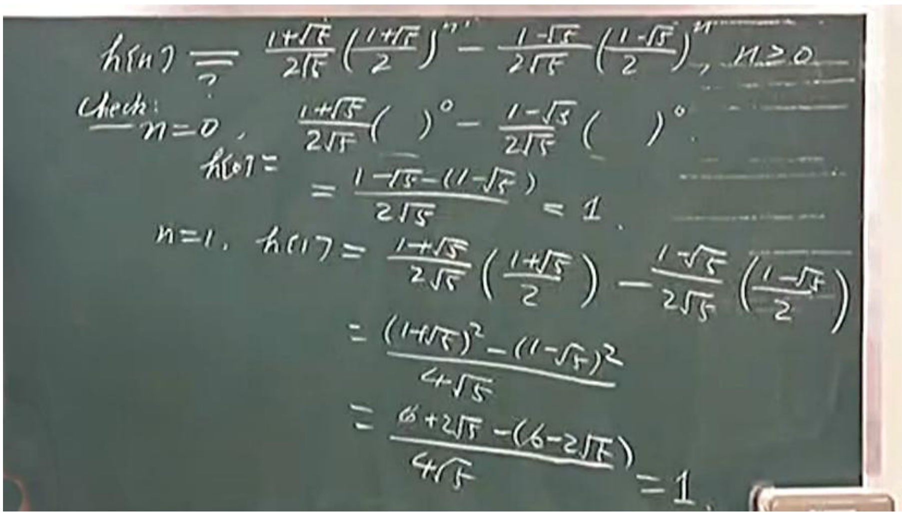
Standard trick:

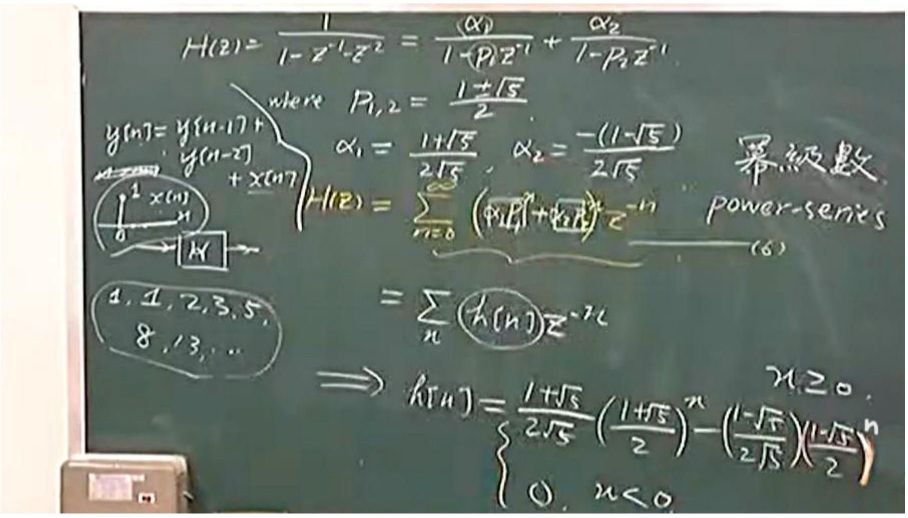

Sol 2. Both sides × (1 − p₁z⁻¹)
Then, consider z = p₁
Sol. 1: P₁,₂ = (1 ± √5)/2
H(z) =
naive 天真
α₁ + α₂ = 1
α₁p₂ + α₂p₁ = 0

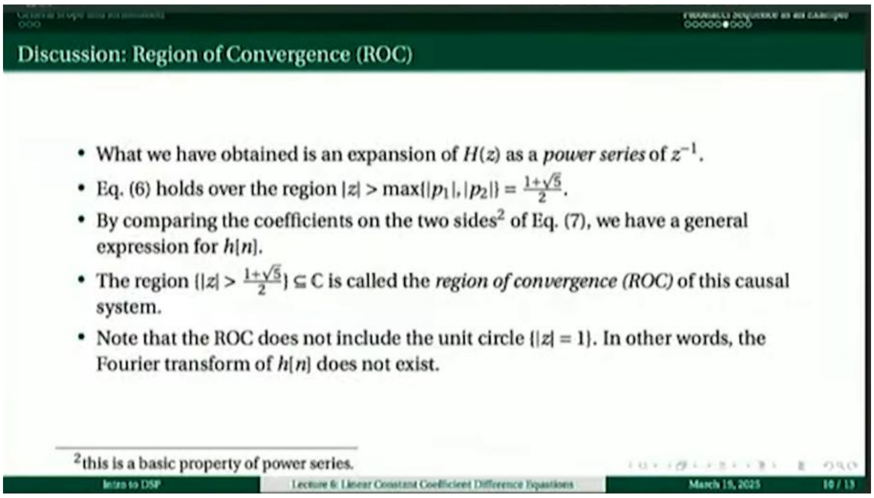

General scope and formulation
Fibonacci Sequence as an Example
Forcing an equality after partial fractional decomposition
Now, the next trick is to apply the following identity (with a leap of faith),
(4)
The equality will hold as long as |z| > |pk|. Applying this to Eq. (3), we have
(5)
(6)
(7)

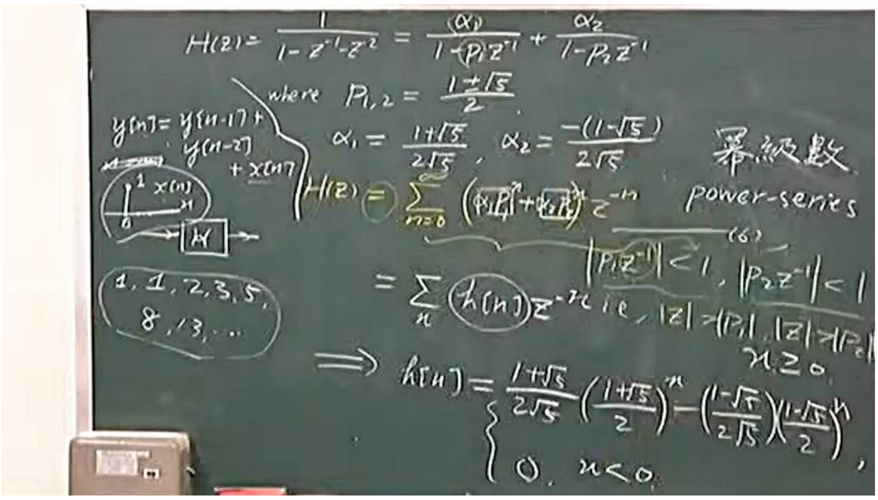

where
冪級數
power-series
n ≥ 0
0, n < 0

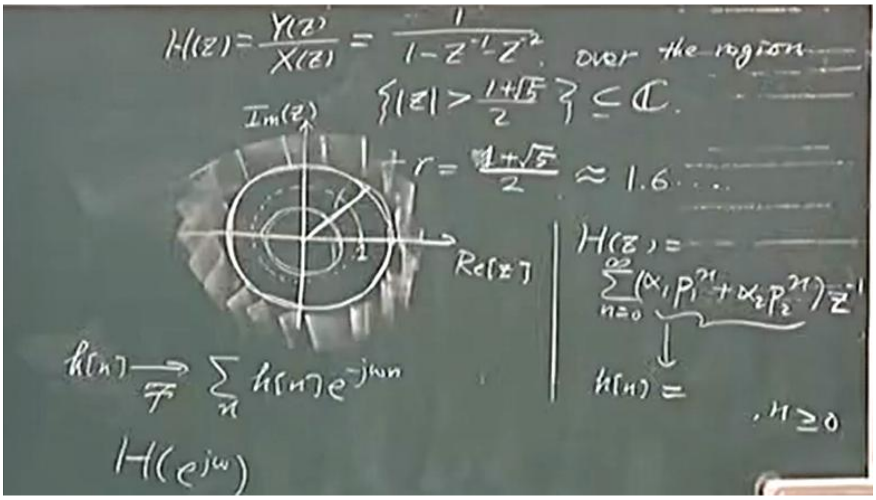

$H(z) = \frac{1}{1-z^{-1}-z^{-2}} = \frac{\alpha_1}{1-p_1 z^{-1}} + \frac{\alpha_2}{1-p_2 z^{-1}}$
where $p_{1,2} = \frac{1 \pm \sqrt{5}}{2}$
$y[n] = y[n-1] + y[n-2] + x[n]$
$\alpha_1 = \frac{1+\sqrt{5}}{2\sqrt{5}}, \quad \alpha_2 = \frac{-(1-\sqrt{5})}{2\sqrt{5}}$
冪級數
power-series
$H(z) = \sum_{n=0}^{\infty} (\square - \square) z^{-n}$
$= \sum_n h[n] z^{-n}$
1, 1, 2, 3, 5, 8, 13, ...
$\Rightarrow h[n] = \frac{1+\sqrt{5}}{2\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^n - \left(\frac{1-\sqrt{5}}{2\sqrt{5}}\right)\left(\frac{1-\sqrt{5}}{2}\right)^n, \quad n \geq 0$
$0, \quad n < 0$

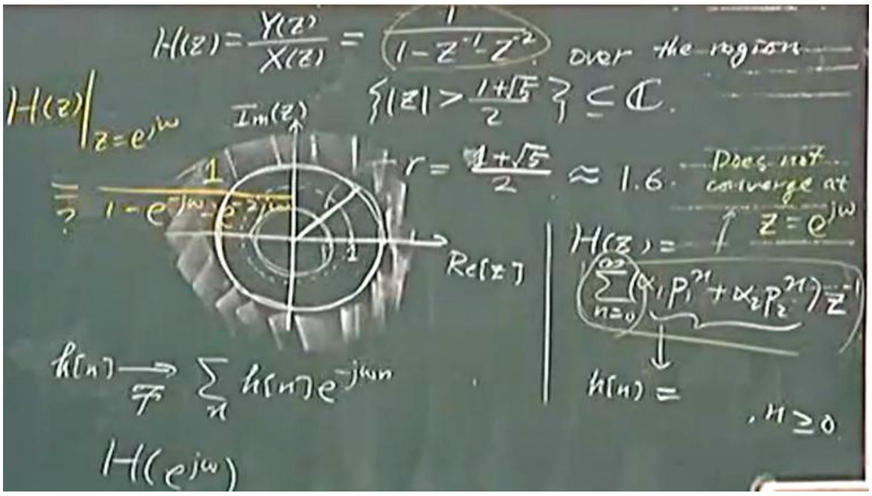

$h[n] \overset{?}{=} \frac{1+\sqrt{5}}{2\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^n - \frac{1-\sqrt{5}}{2\sqrt{5}}\left(\frac{1-\sqrt{5}}{2}\right)^n, \quad n \geq 0$
check:
$n=0, \quad h[0] = \frac{1+\sqrt{5}}{2\sqrt{5}}(\ )^0 - \frac{1-\sqrt{5}}{2\sqrt{5}}(\ )^0$
$= \frac{1+\sqrt{5}-(1-\sqrt{5})}{2\sqrt{5}} = 1$
$n=1, \quad h[1] = \frac{1+\sqrt{5}}{2\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right) - \frac{1-\sqrt{5}}{2\sqrt{5}}\left(\frac{1-\sqrt{5}}{2}\right)$
$= \frac{(1+\sqrt{5})^2 - (1-\sqrt{5})^2}{4\sqrt{5}}$
$= \frac{6+2\sqrt{5}-(6-2\sqrt{5})}{4\sqrt{5}} = 1$

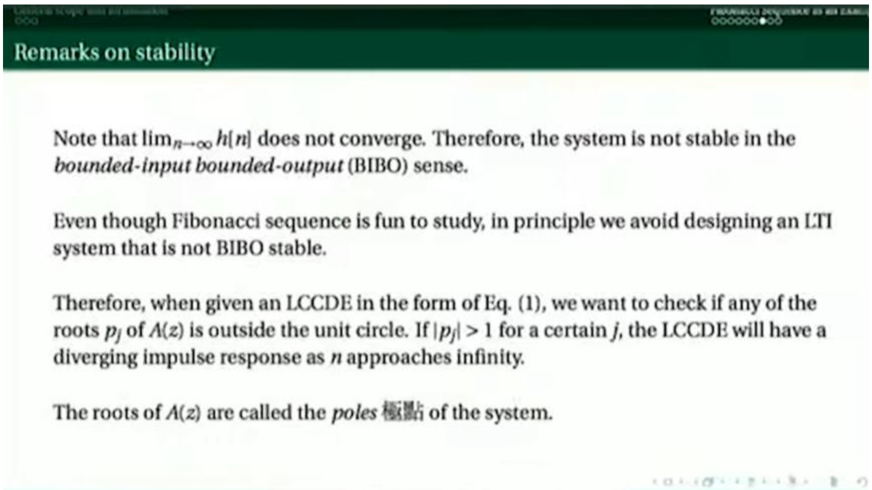

$H(z) = \frac{1}{1-z^{-1}-z^{-2}} = \frac{\alpha_1}{1-p_1 z^{-1}} + \frac{\alpha_2}{1-p_2 z^{-1}}$
where $p_{1,2} = \frac{1 \pm \sqrt{5}}{2}$
$y[n] = y[n-1] + y[n-2] + x[n]$
$\alpha_1 = \frac{1+\sqrt{5}}{2\sqrt{5}}, \quad \alpha_2 = \frac{-(1-\sqrt{5})}{2\sqrt{5}}$
冪級數
power-series
$H(z) = \sum_{n=0}^{\infty} (\alpha_1 p_1^n + \alpha_2 p_2^n) z^{-n}$ (6)
$= \sum_n h[n] z^{-n}$
1, 1, 2, 3, 5, 8, 13, ...
$\Rightarrow h[n] = \frac{1+\sqrt{5}}{2\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^n - \left(\frac{1-\sqrt{5}}{2\sqrt{5}}\right)\left(\frac{1-\sqrt{5}}{2}\right)^n, \quad n \geq 0$
$0, \quad n < 0$


Discussion: Region of Convergence (ROC)
• What we have obtained is an expansion of $H(z)$ as a *power series* of z^{-1}.
• Eq. (6) holds over the region $|z| > \max\{|p_1|, |p_2|\} = \frac{1+\sqrt{5}}{2}$.
• By comparing the coefficients on the two sides[2] of Eq. (7), we have a general expression for $h[n]$.
• The region $\{|z| > \frac{1+\sqrt{5}}{2}\} \subseteq \mathbb{C}$ is called the *region of convergence (ROC)* of this causal system.
• Note that the ROC does not include the unit circle $\{|z| = 1\}$. In other words, the Fourier transform of $h[n]$ does not exist.
[2]this is a basic property of power series.
Intro to DSP
Lecture 6: Linear Constant Coefficient Difference Equations
March 19, 2025
10 / 13


where
冪級數
power-series
1, 1, 2, 3, 5, 8, 13, ...


over the region
Im(z)
Re(z)





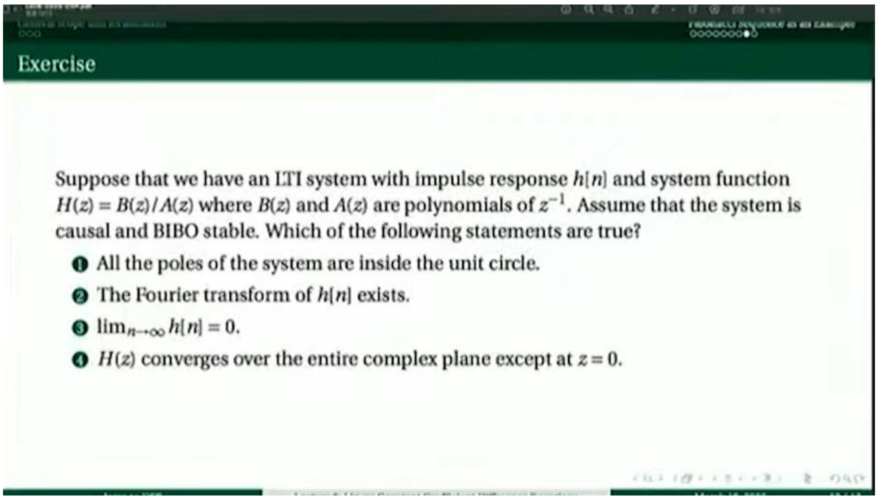
Remarks on stability

Note that $\lim_{n\to\infty} h[n]$ does not converge. Therefore, the system is not stable in the *bounded-input bounded-output* (BIBO) sense.

Even though Fibonacci sequence is fun to study, in principle we avoid designing an LTI system that is not BIBO stable.

Therefore, when given an LCCDE in the form of Eq. (1), we want to check if any of the roots p_j of $A(z)$ is outside the unit circle. If $|p_j| > 1$ for a certain j, the LCCDE will have a diverging impulse response as n approaches infinity.

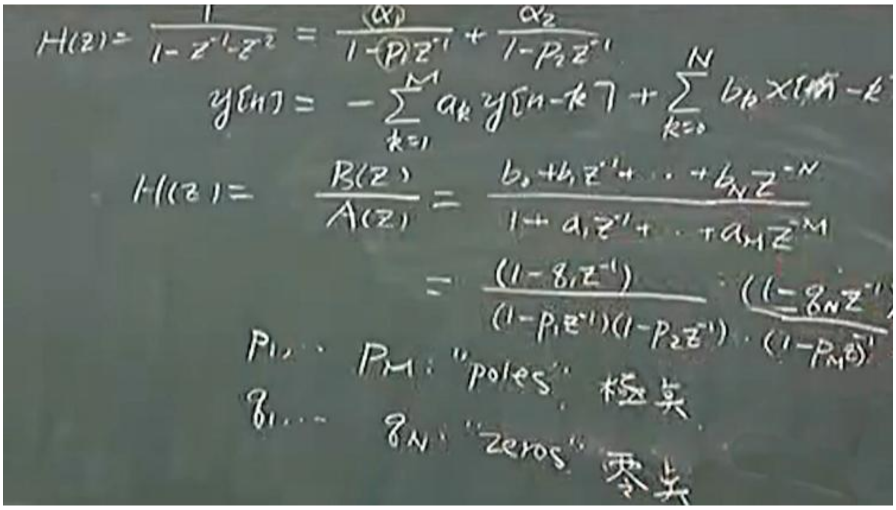
The roots of $A(z)$ are called the *poles* 極點 of the system.

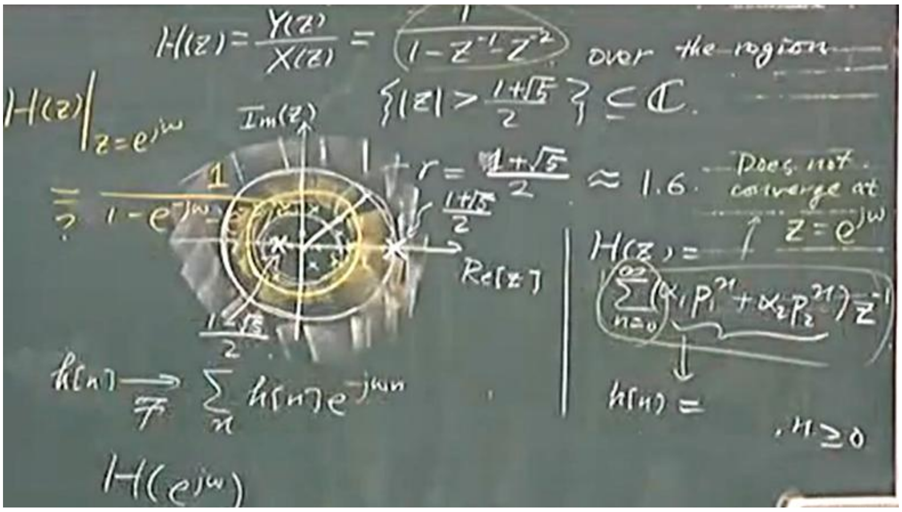


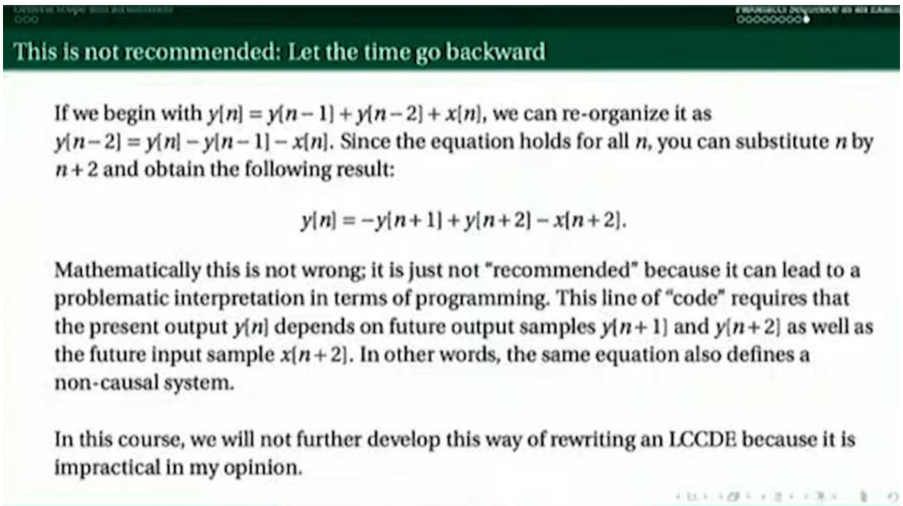
Exercise

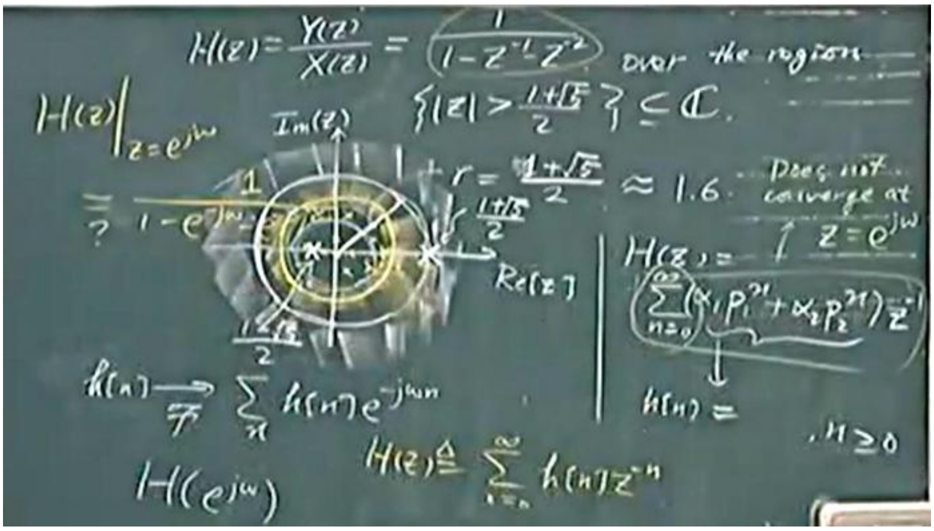
Suppose that we have an LTI system with impulse response $h[n]$ and system function $H(z) = B(z)/A(z)$ where $B(z)$ and $A(z)$ are polynomials of z^{-1}. Assume that the system is causal and BIBO stable. Which of the following statements are true?

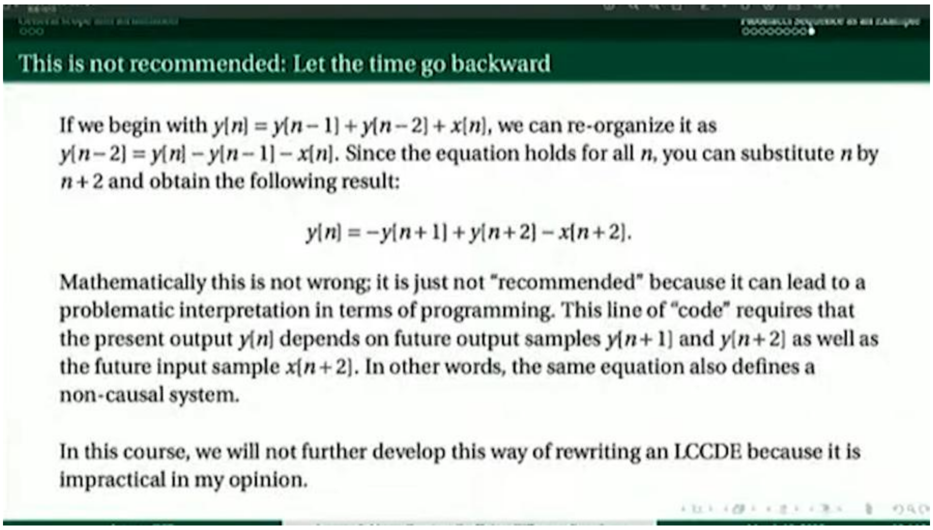
1. All the poles of the system are inside the unit circle.
2. The Fourier transform of $h[n]$ exists.
3. $\lim_{n\to\infty} h[n] = 0$.
4. $H(z)$ converges over the entire complex plane except at $z = 0$.

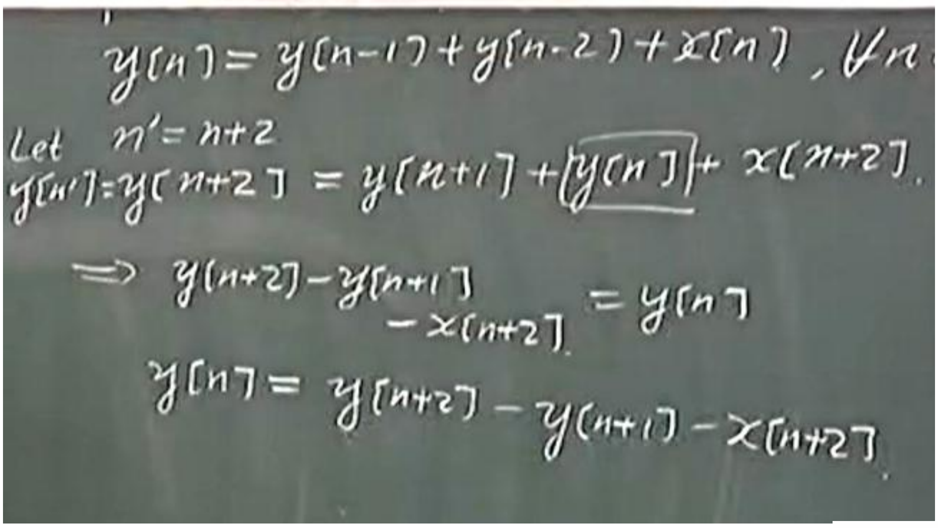

"poles" 極點
"zeros" 零點

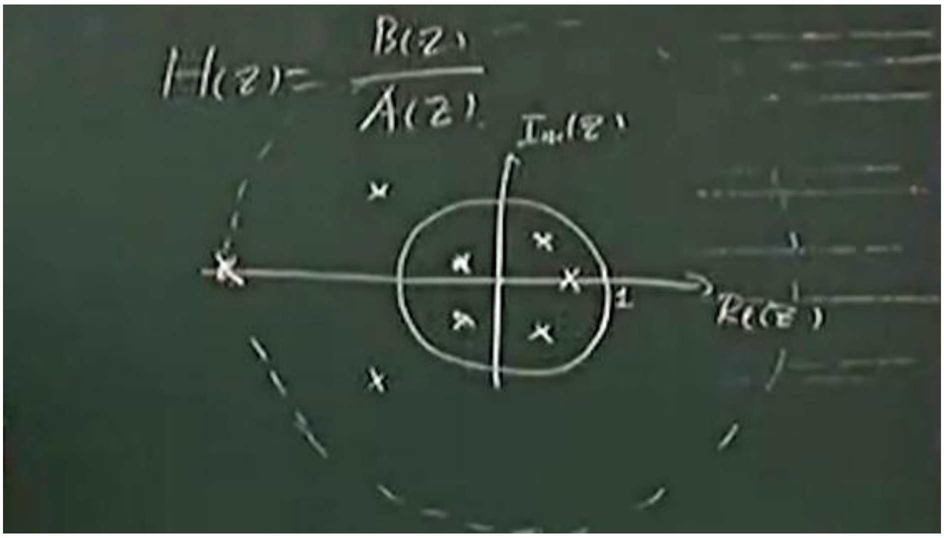

over the region
Im(z)
Re(z)
≈ 1.6
Does not converge at
z = e^jω

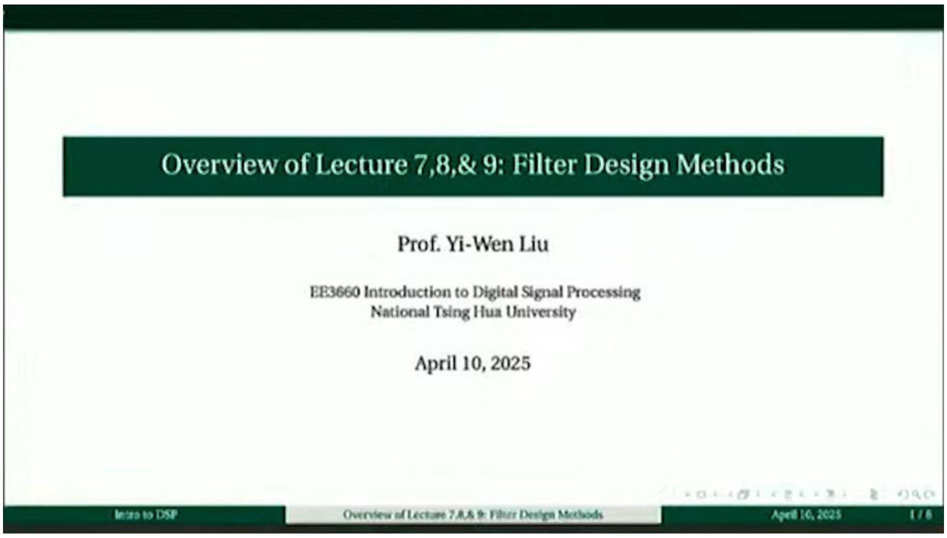

This is not recommended: Let the time go backward
If we begin with y[n] = y[n−1] + y[n−2] + x[n], we can re-organize it as y[n−2] = y[n] − y[n−1] − x[n]. Since the equation holds for all n, you can substitute n by n+2 and obtain the following result:
y[n] = −y[n+1] + y[n+2] − x[n+2].
Mathematically this is not wrong; it is just not "recommended" because it can lead to a problematic interpretation in terms of programming. This line of "code" requires that the present output y[n] depends on future output samples y[n+1] and y[n+2] as well as the future input sample x[n+2]. In other words, the same equation also defines a non-causal system.
In this course, we will not further develop this way of rewriting an LCCDE because it is impractical in my opinion.

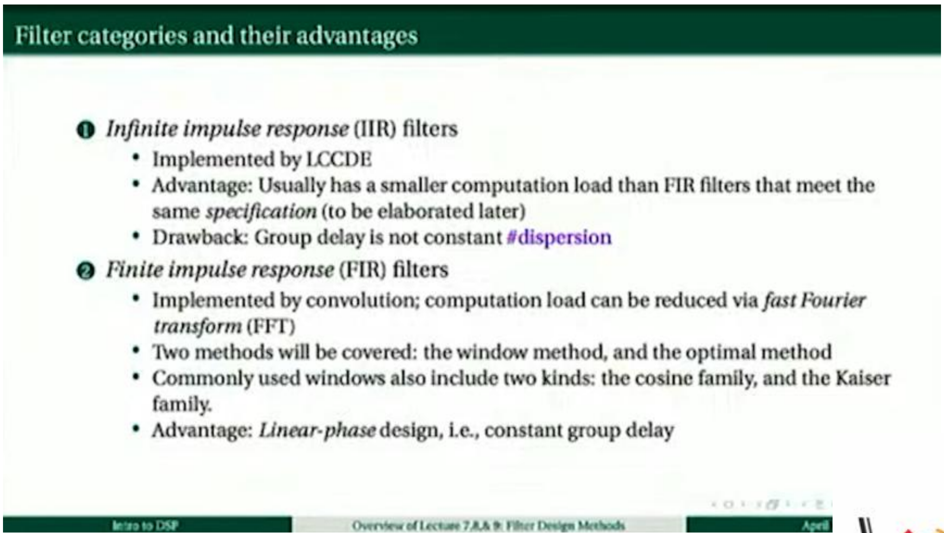


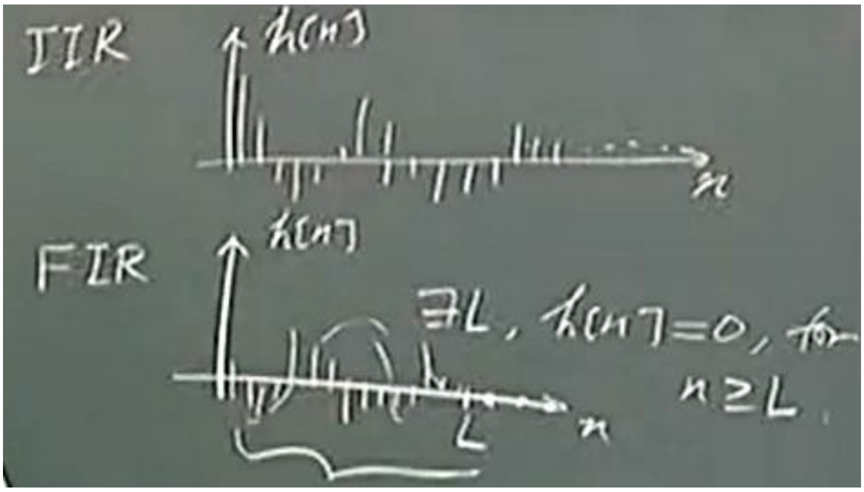

This is not recommended: Let the time go backward
If we begin with $y[n] = y[n-1] + y[n-2] + x[n]$, we can re-organize it as $y[n-2] = y[n] - y[n-1] - x[n]$. Since the equation holds for all n, you can substitute n by $n+2$ and obtain the following result:
$$y[n] = -y[n+1] + y[n+2] - x[n+2].$$
Mathematically this is not wrong; it is just not "recommended" because it can lead to a problematic interpretation in terms of programming. This line of "code" requires that the present output $y[n]$ depends on future output samples $y[n+1]$ and $y[n+2]$ as well as the future input sample $x[n+2]$. In other words, the same equation also defines a non-causal system.
In this course, we will not further develop this way of rewriting an LCCDE because it is impractical in my opinion.

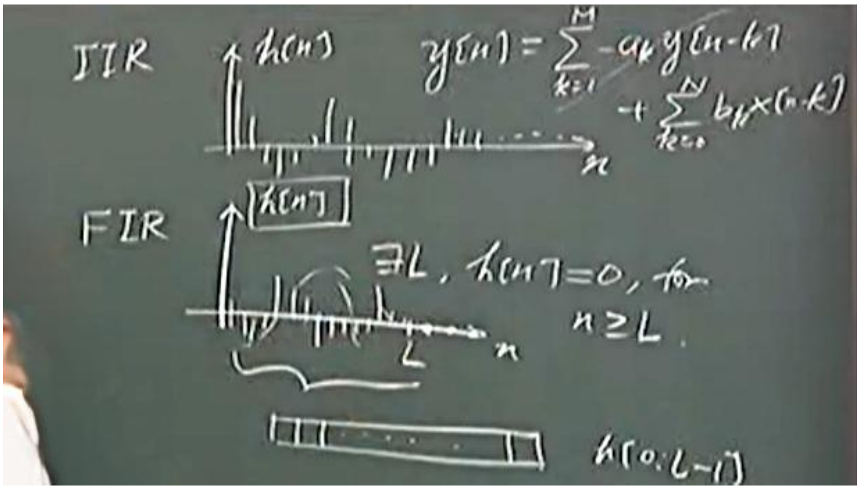

$y[n] = y[n-1] + y[n-2] + x[n]$, $\forall n$
Let $n' = n+2$
$y[n'] = y[n+2] = y[n+1] + y[n] + x[n+2]$
$\Rightarrow y[n+2] - y[n+1] - x[n+2] = y[n]$
$y[n] = y[n+2] - y[n+1] - x[n+2]$

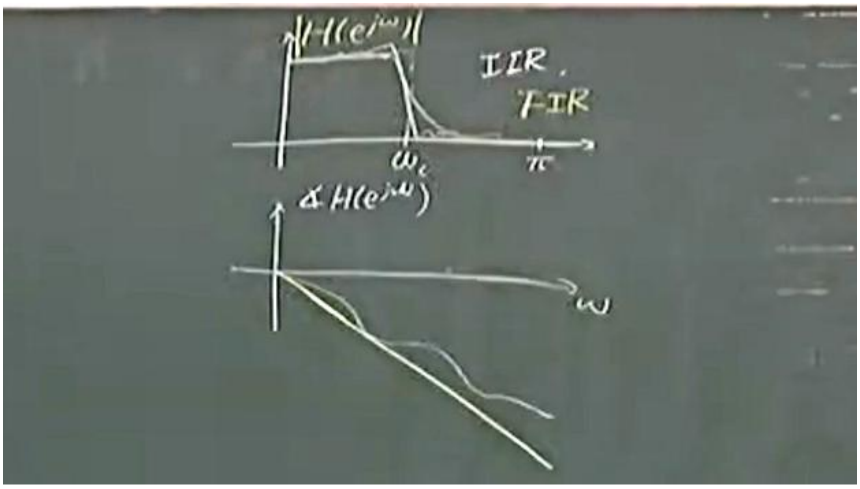

H(z) = B(z) / A(z)
Im(z)
1
Re(z)

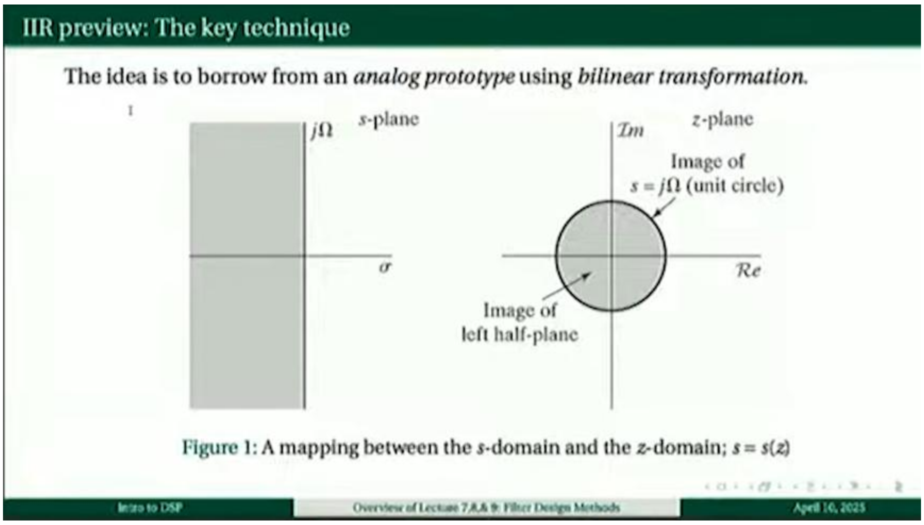

Overview of Lecture 7,8,& 9: Filter Design Methods
Prof. Yi-Wen Liu
EE3660 Introduction to Digital Signal Processing
National Tsing Hua University
April 10, 2025
Intro to DSP
Overview of Lecture 7,8,& 9: Filter Design Methods
April 10, 2025
1 / 8

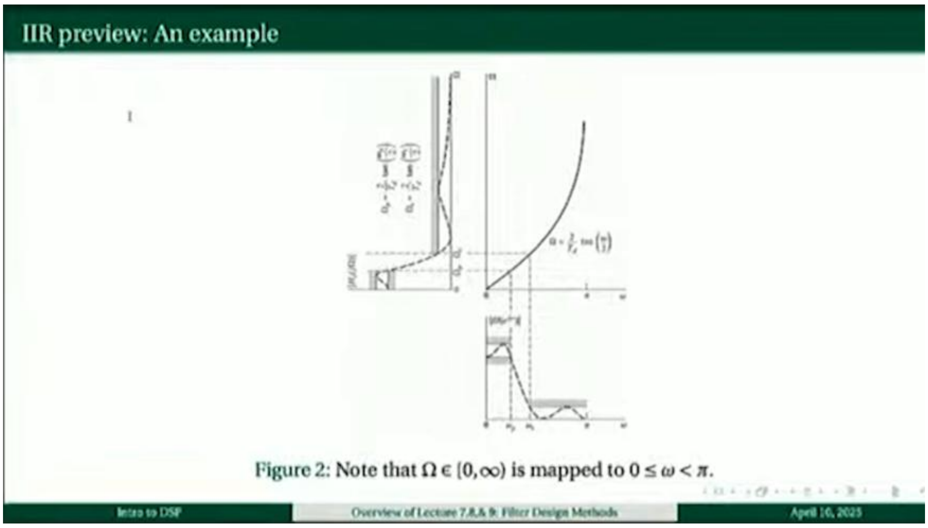

Filter categories and their advantages
1. Infinite impulse response (IIR) filters
• Implemented by LCCDE
• Advantage: Usually has a smaller computation load than FIR filters that meet the same specification (to be elaborated later)
• Drawback: Group delay is not constant #dispersion
2. Finite impulse response (FIR) filters
• Implemented by convolution; computation load can be reduced via fast Fourier transform (FFT)
• Two methods will be covered: the window method, and the optimal method
• Commonly used windows also include two kinds: the cosine family, and the Kaiser family.
• Advantage: Linear-phase design, i.e., constant group delay
Intro to DSP
Overview of Lecture 7,8,& 9: Filter Design Methods
April

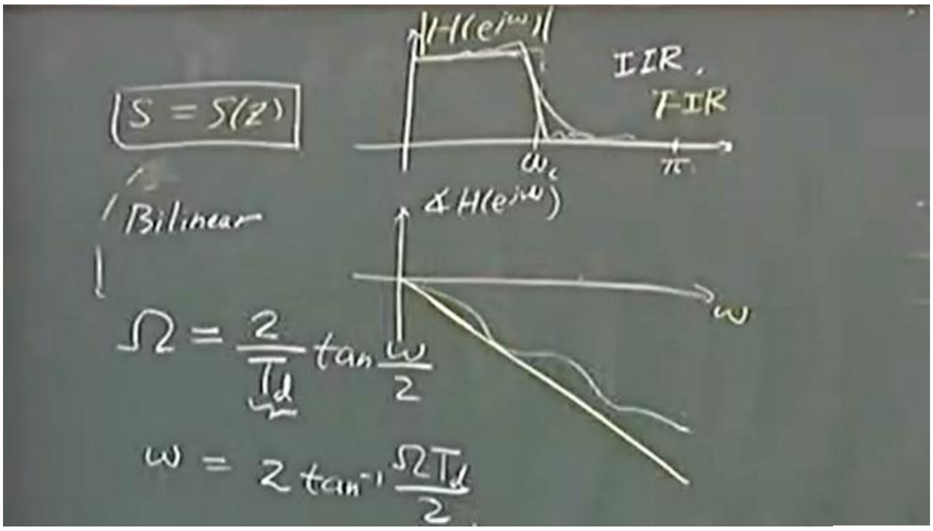

IIR
h[n]
n
FIR
h[n]
∃L, h[n]=0, for
n≥L.
L
n


IIR
h[n]
y[n] = Σ_{k=1}^{M} -a_k y[n-k] + Σ_{k=0}^{N} b_k x[n-k]
n
FIR
h[n]
∃L, h[n]=0, for
n≥L.
L
n
h[0:L-1]


|H(e^{jω})|
IIR
FIR
ω_c
π
∡H(e^{jω})
ω


IIR preview: The key technique
The idea is to borrow from an analog prototype using bilinear transformation.
jΩ
s-plane
σ
Im
z-plane
Image of
s = jΩ (unit circle)
Re
Image of
left half-plane
Figure 1: A mapping between the s-domain and the z-domain; s = s(z)
Intro to DSP
Overview of Lecture 7,8,& 9: Filter Design Methods
April 16, 2025


IIR preview: An example
Figure 2: Note that Ω ∈ [0,∞) is mapped to 0 ≤ ω < π.
Intro to DSP
Overview of Lecture 7,8,& 9: Filter Design Methods
April 16, 2025


S = S(z)
Bilinear
|H(e^{jω})|
IIR
FIR
ω_c
π
∡H(e^{jω})
ω
Ω = 2/T_d tan ω/2
ω = 2 tan^{-1} ΩT_d/2

FIR overview: the first step

The idea is to approximate the ideal filter, but make its length finite. In Homework 2 Problem 1, you have seen that

$$g[n] = \begin{cases} \frac{\sin \omega_c n}{\pi n}, & n \neq 0 \\ \frac{\omega_c}{\pi}, & n = 0 \end{cases}$$

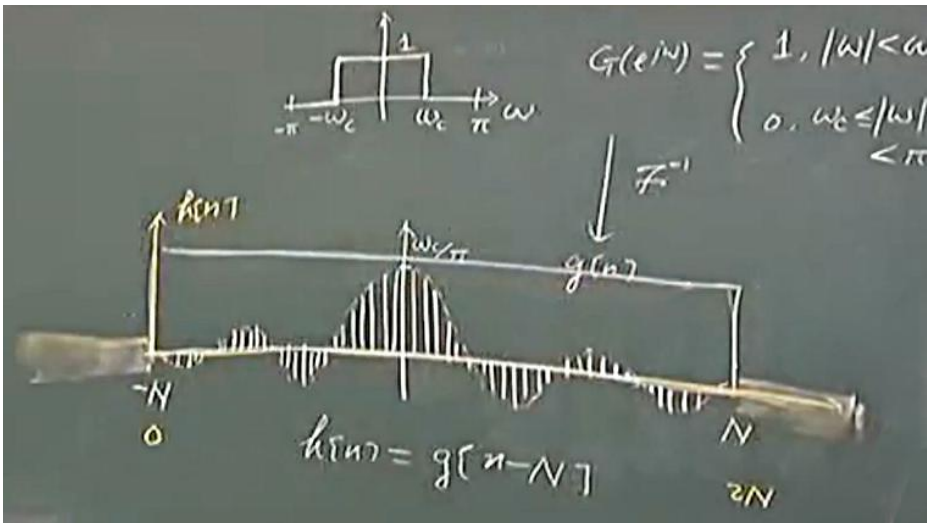
is the impulse response for an ideal low-pass filter with a cutoff frequency at ω_c.

Then, truncate $g[n]$ at $-N \le n \le N$; that is, let's define

$$h_N[n] = \begin{cases} g[n], & -N \le n \le N \\ 0, & \text{elsewhere.} \end{cases}$$

This can already be called a "design", but we can do better by multiplying $h_N[n]$ with a window function $w_{zp}[n]$. (zp stands for "zero-phase")

Intro to DSP | Overview of Lecture 7,8,& 9: Filter Design Methods | April 10, 2025



FIR: Commonly used windows

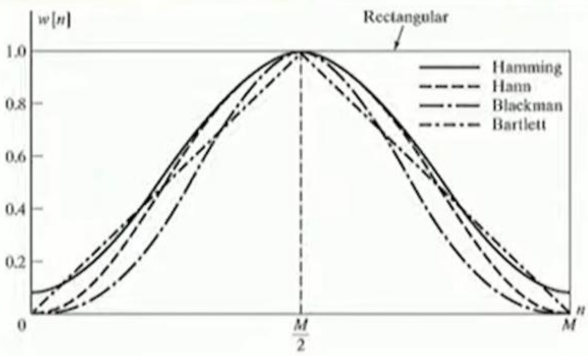
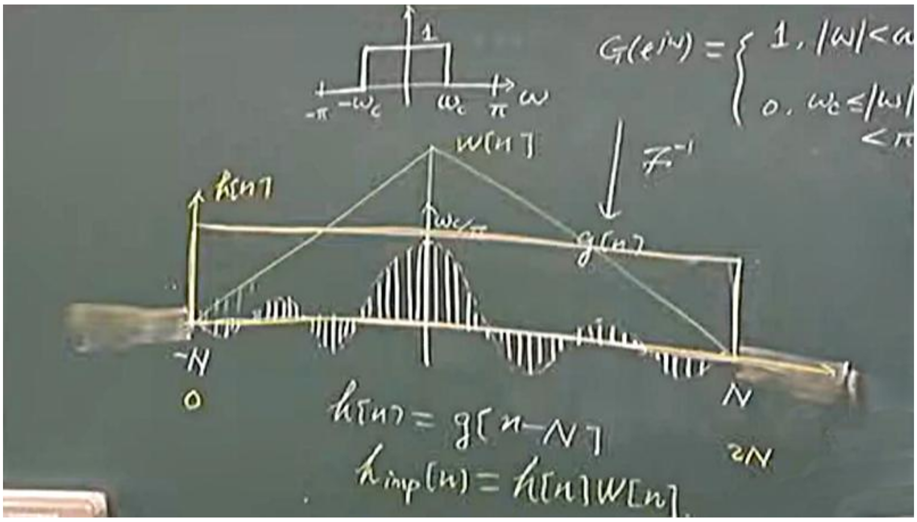
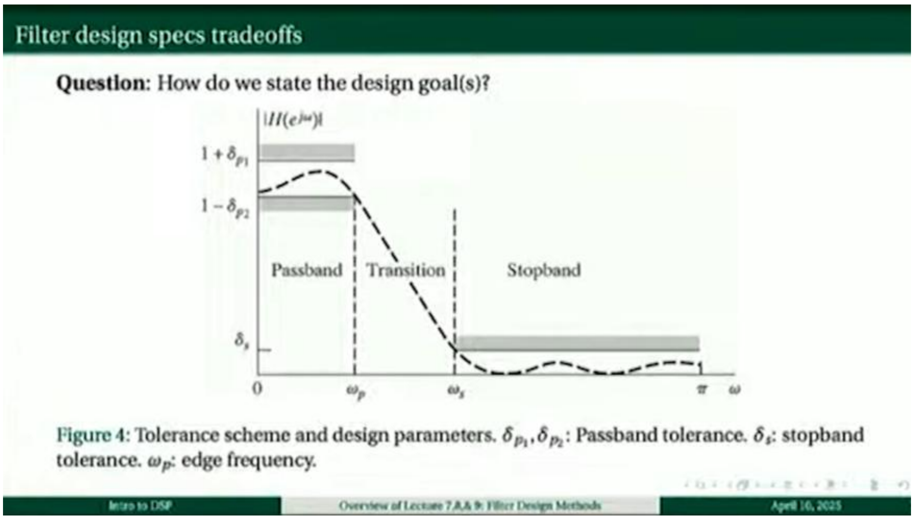
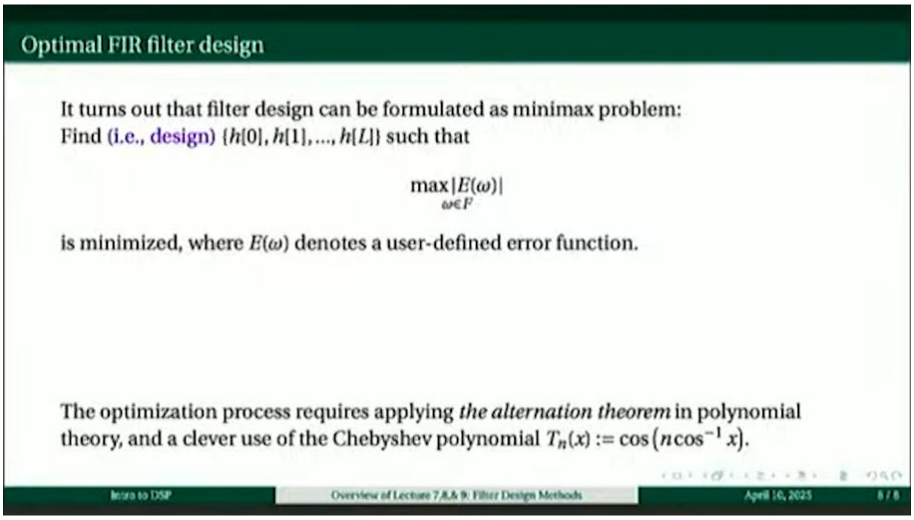


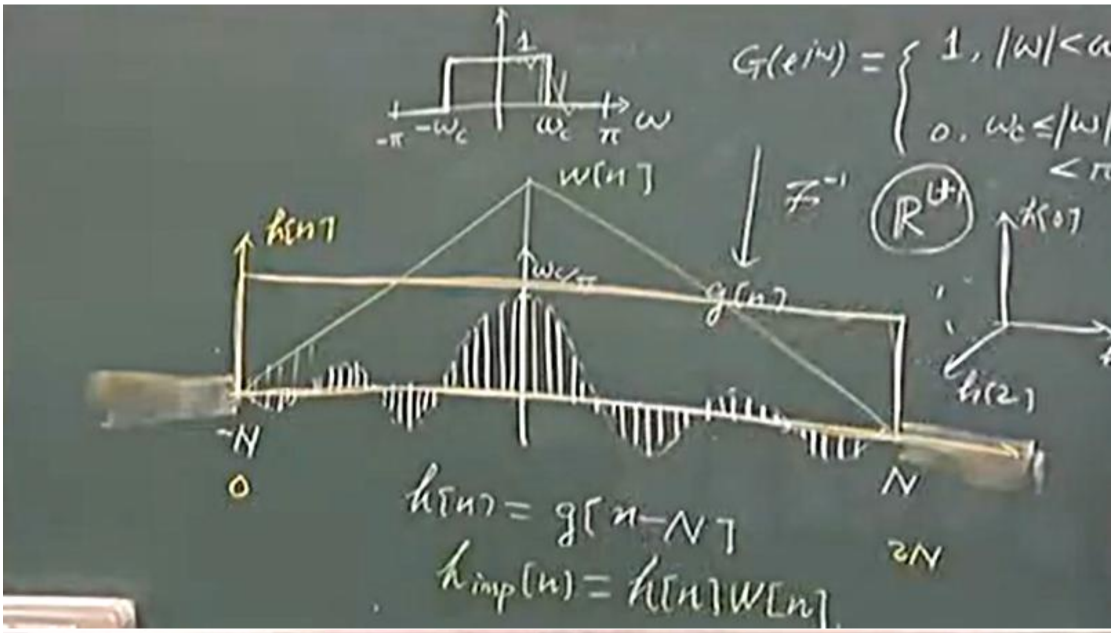
Figure 3: Here, $M = 2N$, and $w_{zp}[n] = w[n-N]$.

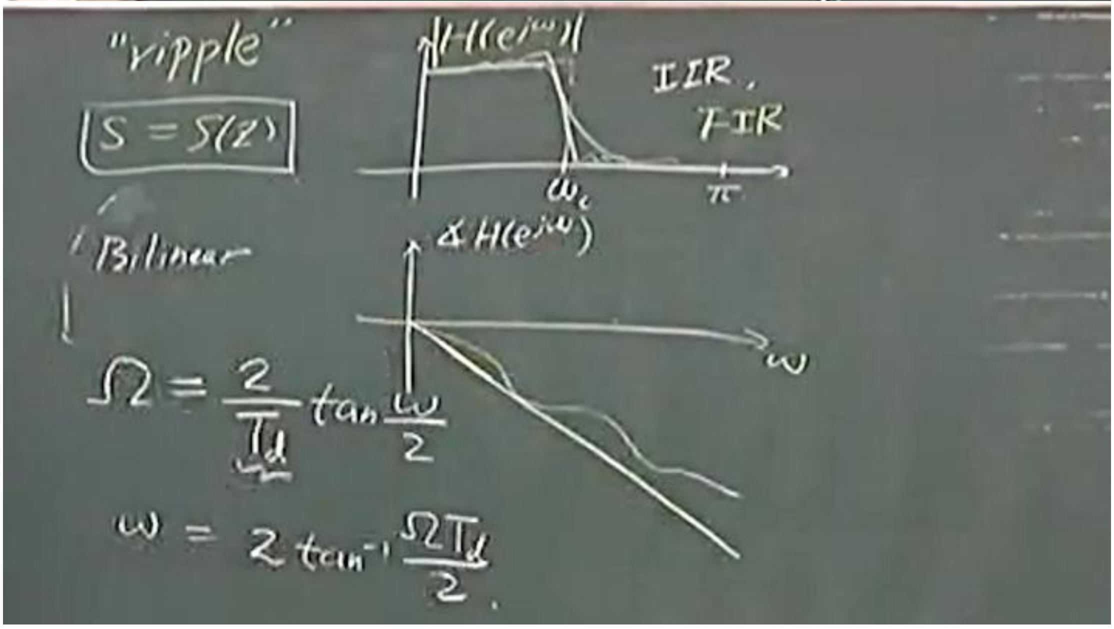
Intro to DSP | Overview of Lecture 7,8,& 9: Filter Design Methods | April 10, 2025 | 6 / 8


$G(e^{j\omega}) = \begin{cases} 1, & |\omega| < \omega_c \\ 0, & \omega_c \le |\omega| < \pi \end{cases}$
$\mathcal{F}^{-1}$
$g[n]$
$W[n]$
$h[n]$
ω_c/π
$-N$
0
N
$2N$
$h[n] = g[n-N]$
$h_{imp}(n) = h[n]W[n]$


Filter design specs tradeoffs
Question: How do we state the design goal(s)?
$|H(e^{j\omega})|$
$1+\delta_{p_1}$
$1-\delta_{p_2}$
δ_s
Passband
Transition
Stopband
0
ω_p
ω_s
π
ω
Figure 4: Tolerance scheme and design parameters. $\delta_{p_1}, \delta_{p_2}$: Passband tolerance. δ_s: stopband tolerance. ω_p: edge frequency.
Intro to DSP
Overview of Lecture 7,8,& 9: Filter Design Methods
April 10, 2025


Optimal FIR filter design
It turns out that filter design can be formulated as minimax problem:
Find (i.e., design) $\{h[0], h[1], \ldots, h[L]\}$ such that
$\max_{\omega \in F} |E(\omega)|$
is minimized, where $E(\omega)$ denotes a user-defined error function.
The optimization process requires applying *the alternation theorem* in polynomial theory, and a clever use of the Chebyshev polynomial $T_n(x) := \cos(n \cos^{-1} x)$.
Intro to DSP
Overview of Lecture 7,8,& 9: Filter Design Methods
April 10, 2025
8 / 8


G(e^{jω}) = { 1, |ω| < ω
0, ωc ≤ |ω|
ℱ⁻¹
w[n]
h[n]
g[n]
h(z)
ωc/π
-N
0
N
2N
h[n] = g[n−N]
h_imp[n] = h[n]w[n]


"ripple"
S = S(z)
|H(e^{jω})|
IIR,
FIR
ωc
π
∡H(e^{jω})
Bilinear
ω
Ω = 2/Td tan ω/2
ω = 2 tan⁻¹ ΩTd/2