NTNU – Trondheim
Norwegian University of
Science and Technology

TTT4120 Digital Signal Processing
Fall 2021

Lecture: Course Arrangements & Introduction

Prof. Stefan Werner
stefan.werner@ntnu.no
Office B329

Department of Electronic Systems

1

Contents and learning outcomes

- Course arrangements and general introduction
- Discrete-time signals in time-domain

2

2

At your service

- Lecturer:

Professor Stefan Werner, stefan.werner@ntnu.no, room B329

- Teaching assistants (click to contact):

Francois

Reza

Mohammad

Hossein

Ehsan

Gianluca

3

3

At your service …

- Student assistants:
  - Solveig Reppen Lunde, solverl@stud.ntnu.no
  - Helene Markeng, volrathb@stud.ntnu.no

Check Blackboard for any updates

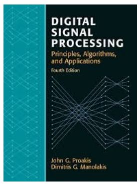
4

4

Basic course information

- Course information
  - all information through Blackboard (BB)
  - open page exists

    http://www.ntnu.edu/studies/courses/TTT4120
- Teaching:
  - lectures on Tue 16:15-18:00 and Thu 14:15 - 16:00 (EL5)
  - tutoring/exercises Thu 16:15-18:00 (KJL1)
- Course material:
  - Proakis, Manolakis Digital Signal Processing, 4th Ed.
  - lecture notes on Blackboard (Norwegian/English)
  - lecture slides



5

5

Basic course information…

- Important for Fall term 2021
  - lectures at campus, but can change abruptly to online
  - homework sessions at campus; TAs can be contacted online
  - to facilitate distance learning, most material available on BB

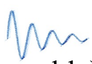
6

6

Basic course information…

- Exercises:
  - 10-11 exercises, each up to 10 points
  - total of 50 points required to participate in exam
    (at least 25 points from exercises 6-11)
- Sessions will start next week
- Exam:
  - December 13, 9-13 (home exam)

Digital signal processing (DSP)

- What is a signal? 
  - a varying (measurable) quantity
  - conveys information
- Signals can be deterministic or random
- Digital signal processing (DSP)
  - signal is a sequence of numbers
  - DSP involves transforming one signal into another signal

Signal processing is the enabling technology for the generation, transformation, and interpretation of information

Digital signal processing (DSP)

- Definition of DSP:
  - digitalization, processing, transmission and/or representation of physical (analog) signals
  - modeling of physical signals and systems
- DSP discipline provides a toolbox of methods and algorithms
  - digitalization and analysis of physical signals
  - analysis, design, and use of linear digital systems
  - mathematical treatment of discrete versions of physical signals
  - estimation and application of mathematical models for different signals and channels

9

9

Digital signal processing (DSP)…

- Discipline is quite theoretical

$$H(z) = \sum_{k=0}^{M-1} b_k z^{-k} =$$

$$= \sum_{k=0}^{(M-3)/2} h[k]z^{-k} + h\left[\frac{M-1}{2}\right] z^{-(M-1)/2} + \sum_{k=(M+1)/2}^{M-1} h[k]z^{-k}$$

$$= \sum_{k=0}^{(M-3)/2} h[k]z^{-k} + h\left[\frac{M-1}{2}\right] z^{-(M-1)/2} + \sum_{k=(M+1)/2}^{M-1} h[M-1-k]z^{-k}$$

$$= \sum_{k=0}^{(M-3)/2} h[k]z^{-k} + h\left[\frac{M-1}{2}\right] z^{-(M-1)/2} + \sum_{l=0}^{(M-3)/2} h[l]z^{l-(M-1)}$$

$$= h\left[\frac{M-1}{2}\right] z^{-(M-1)/2} + \sum_{k=0}^{(M-3)/2} h[k]\left(z^{-k} + z^{k-(M-1)}\right)$$

$$= h\left[\frac{M-1}{2}\right] z^{-(M-1)/2} + \sum_{k=0}^{(M-3)/2} h[k]z^{-(M-1)/2}\left(z^{-(k-(M-1)/2)} + z^{k-(M-1)/2}\right)$$

$$= \left(h\left[\frac{M-1}{2}\right] + \sum_{k=0}^{(M-3)/2} h[k]\left(z^{-(k-(M-1)/2)} + z^{k-(M-1)/2}\right)\right) z^{-(M-1)/2}$$

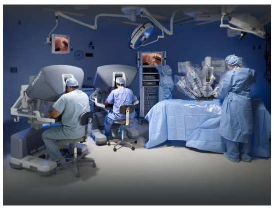
$$= \left(h\left[\frac{M-1}{2}\right] + 2\sum_{k=0}^{(M-3)/2} h[k]\cos[\omega((M-1)/2-k)]\right) e^{-j\omega(M-1)/2}$$

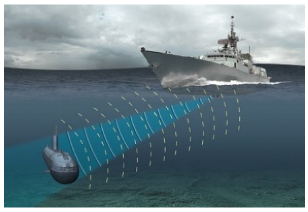
12

12

Digital signal processing (DSP)…

- … but also application oriented





13

13

Digital signal processing (DSP)…

- Traditional areas of DSP include, e.g.,

filter design

spectrum analysis

fast filtering algorithms

signal reconstruction

multirate filters

adaptive filters

14

14

Digital signal processing (DSP)…

- DSP influence on related disciplines

control

digital communication (TTT4130)

estimation, detection and classification (TTT4275)

multimedia signal processing (TTT4135)

machine learning for signal processing (TTT4185)

15

15

Typical DSP intensive problems

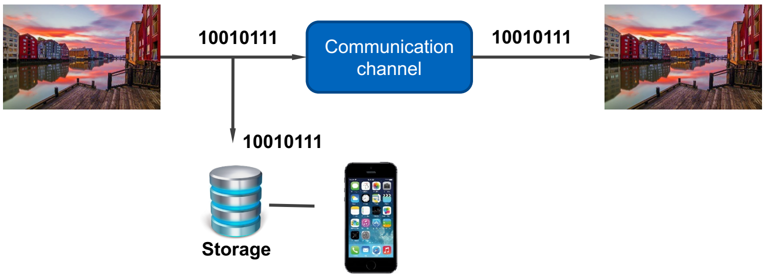
- Signal enhancement, interference reduction
- Efficient coding for transmission or storage
- Control of industrial processes
- Navigation, positioning, surveillance
- Classification, identification, detection, and verification

- Within each problem formulation you may find many applications
- A single application may also require solving a variety of problems

16

16

Applications

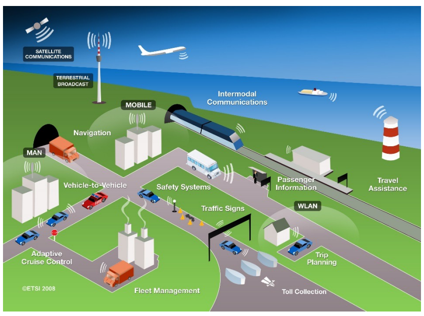


- Efficient and robust signal representation

17

17

Applications…



- Digital signal transmission

18

18

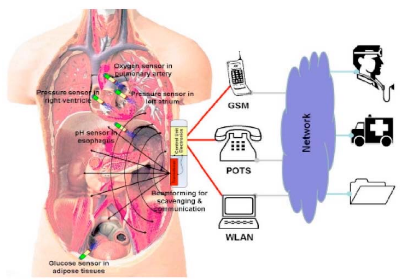
Applications…

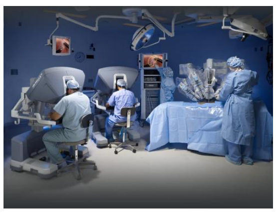
- Speech recognition

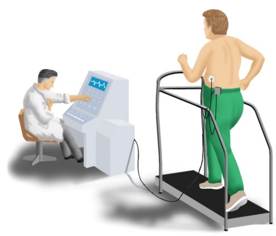
19

19

Applications…





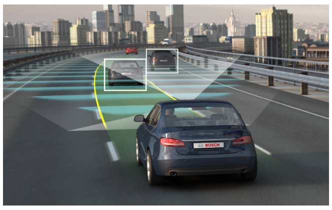


- Medicine

20

20

Applications…

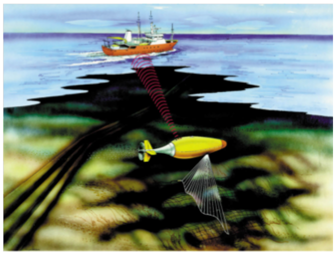


- Self-driving cars

21

21

Applications…



- Underwater communication
- Environmental monitoring

22

22

Digital versus analog signal processing

- Advantages:
  - easy to design and modify
  - possible to build complex systems
  - accuracy easier to control (word lenght of ADC)
  - errors in transmission and storage can be corrected
  - can be implemented in software
  - size decreases and less energy needed
  - same processor can time share different tasks
- Limitations:
  - signals with high bandwidth may require too fast processing and too high sampling rate

23

23

Prerequisites

- Vary with the study programme: 2-3 first weeks will be devoted to make sure all have a common understanding of basics
  - could be a lot of new things for “KYB,” and some repetition for “ELSYS”
- Central concepts during first few weeks:
  - sampling
  - signals and systems in time and frequency
  - math: Fourier analysis, Laplace transform, complex numbers, partial fractions
  - statistics: probability density functions, expectations
  - MATLAB (or Python)

24

24

Course contents

- Introduction to discrete signals and systems
- Analysis of discrete systems/filters using z-transform
- Sampling of analog signals, Nyquist rate and reconstruction
- Sampling in frequency domain, DFT and FFT
- Correlation and energy spectral density
- Stochastic signals, basic estimation theory, spectral estimation, and signal models, prediction
- Filter design and implementation
- Quantization and quantization noise
- Multirate signal processing

25

25

Example questions

- Consider the following two operations

$$y[n] = \frac{1}{4}x[n] + \frac{1}{4}x[n-1] + \frac{1}{4}x[n-2] + \frac{1}{4}x[n-3]$$
$$y[n] = \frac{1}{4}x[n] - \frac{1}{4}x[n-1] + \frac{1}{4}x[n-2] - \frac{1}{4}x[n-3]$$

- Amplifies low/high frequencies? What is digital frequency?
- What about the following two operations (feedback)

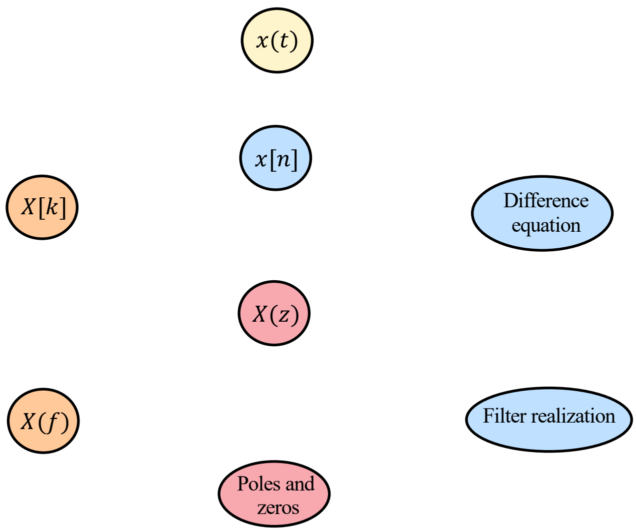
$$y[n] = 0.6y[n-1] + x[n]$$
$$y[n] = 1.1y[n-1] + x[n]$$

- Stability issues.
- How to figure out these things for more complicated signals?

26

26

First couple of weeks



27

27

What you should do

- Be active:
  - important to work throughout the whole semester
  - check lecture/homework material before sessions
  - ask questions
  - try to solve homeworks (not blindly copy someone else)
- Matlab/Python will replace pen and paper when dealing with physical signals
- Try to understand subject rather than only focus on rehearsing exam questions

28

28

Reference group

- Quality control and feeback
  - few students from course (one from each programme)
  - be contact person/link to the students in your programme
  - three meetings during the course

- Volonteers?
- More information found here*

* https://innsida.ntnu.no/wiki/-/wiki/English/Reference+groups+-+quality+assurance+of+education

29

29

NTNU – Trondheim
Norwegian University of
Science and Technology

TTT4120 Digital Signal Processing
Fall 2021

Lecture: Discrete-Time Signals in Time-Domain

Prof. Stefan Werner
stefan.werner@ntnu.no
Office B329

1

Lecture in course book*

- Proakis, Manolakis Digital Signal Processing, 4th Ed.
  - 1.1 Signals, systems and signal processing
  - 1.2 Classification of signals
  - 1.3 The concept of frequency in continuous-time and…
  - 1.4.1 Sampling of analog signals
  - 2.1 Discrete-time signals

*Level of detail is defined by lectures and problem sets

2

2

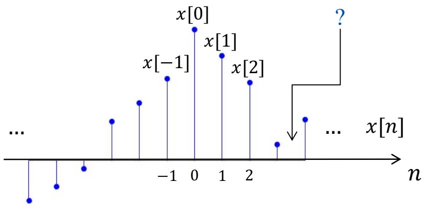
Contents and learning outcomes

- Discrete-time signals
- Power of digital signal processing (DSP)
- Properties, classification, and manipulations of sequences
- A few typical sequences
- Discrete-time sinusoids and sampling of continuous-time sinusoids

Discrete-time signals

- Continuous-time versus discrete-time signals?

Discrete-time signals…



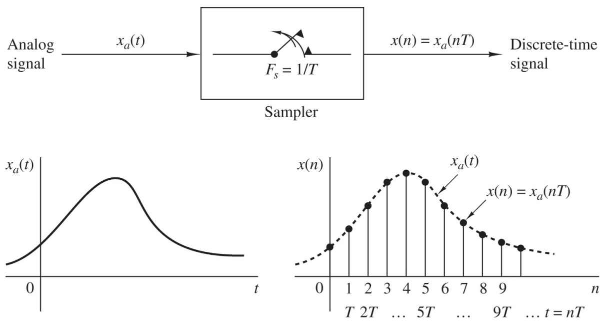
- A discrete-time signal $x[n]$ is represented by a sequence of numbers
- Sequence $x[n]$ can represent a discrete-time signal, where each number $x[n]$ corresponds to a signal amplitude at instant n

$$x[n] = \{\dots\ x[-2], x[-1], \underline{x[0]}, x[1], \dots\}$$

5

5

Discrete-time signals…



- Sometimes $x[n]$ is obtained from *sampling* an analog signal

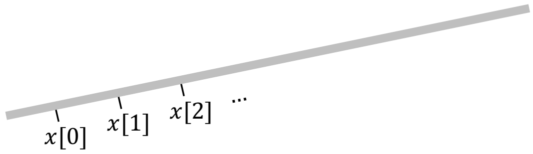
$$x[n] \triangleq x_a(nT)$$

- Interval between samples $T = \frac{1}{F_s}$, where F_s is the sampling rate

6

6

Discrete-time signals…

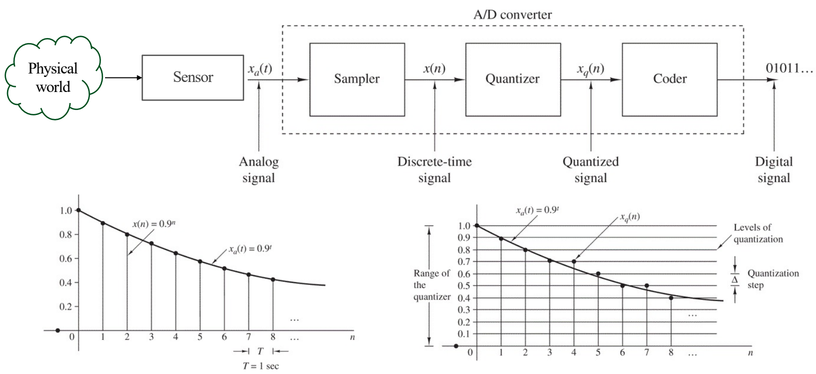


- Note that interval T need not necessarily represent time
- For example, if $x_a(t)$ is the temperature along a metal rod, then if T is a length unit, $x[n] \triangleq x_a(nT)$ represents the temperature at uniformly placed sensors along this rod
- Different choices of T lead to different discrete-time sequences

7

7

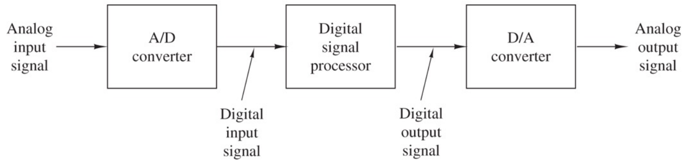
Characterization of signals



- Analog signal $x_a(t)$: continuous in time and amplitude
- Sampled-data signal $x[n]$: discrete-time and continous-amplitude
- Digital signal $x_q[n]$: discrete in both time and amplitude

8

Power of digital signal processing



- Digital signal
  - Discrete-time and discrete-valued sequence of numbers (last attribute less essential for DSP basics)
- Digital signal processing
  - Sequence is transformed to another sequence by means of arithmetic operations

9

9

Power of digital signal processing…

- Analog signal processing:
  - Process a continuously varying quantity (analog signal)
  - Can be described by differential equations
- Digital signal processing:
  - Processes sequences of numbers (discrete-time signals) using some sort of digital hardware or software
  - Power of DSP is that once a sequence of numbers is available to an appropriate digital hardware we can carry out any form of numerical processing on it

10

10

Power of digital signal processing…

- Example: Suppose we want to perform the following operation on a continuous-time signal $x(t)$:

$$y(t) = \frac{\cosh[\ln(|x(t)|)+x^3(t)+\cos^3(\sqrt{|x(t)|})]}{5x^5(t)+e^{x(t)}+\tan(x(t))}$$

- Difficult to implement using analog hardware!
- Alternatively, convert analog signal $x(t)$ into sequence $x[n]$, manipulate it on a digital computer, and generate sequence $y[n]$
- If the continuous-time signal $y(t)$ can be recovered from $y[n]$, then the desired processing has been successfully performed

11

11

Power of digital signal processing…

- Previous example highlights two important points:
  1. How powerful digital signal processing is
  2. To process analog signals using DSP, we must have a way of converting a continuous-time signal into a discrete-time one, such that the continuous-time signal can be recovered from the discrete-time signal
- Many signals are originally discrete-time, and the results of their processing are only needed in digital form

12

12

Operations on discrete-time signals

- Scaling, addition, and multiplication of sequences

$$y[n] = ax[n]$$

$$y[n] = x_1[n] + x_2[n]$$

$$y[n] = x_1[n]x_2[n]$$

- Time shifts and folding

$$y[n] = x[n-k]$$

$$y[n] = x[-n]$$

- Time shifts plus folding

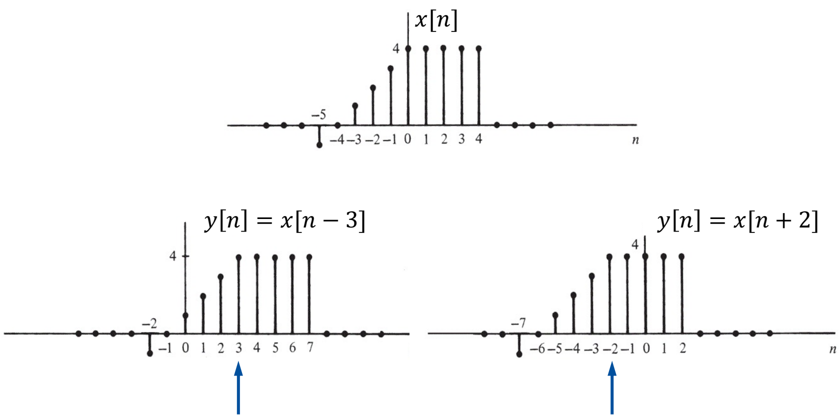
$$y[n] = x[-n+k]$$

13

13

Operations on discrete-time signals…

- Example (time-shift): Given $x[n]$ below, plot $x[n-3]$ and $x[n+2]$

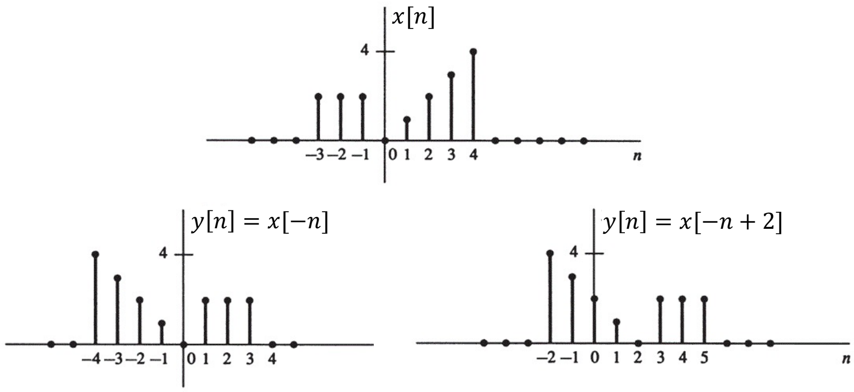


14

14

Operations on discrete-time signals…

- Example (folding): Given $x[n]$ below, plot $x[-n]$ and $x[-n+2]$



Basic properties of discrete-time signals

- A sequence $x[n]$ is causal if

$$x[n] = 0, n < 0$$

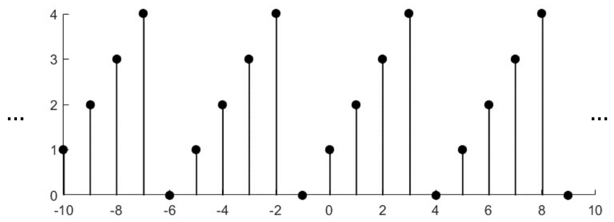
- A sequence $x[n]$ is periodic with period N if

$$x[n+N] = x[n], \forall n$$

where smallest N satisfying the above is the fundamental period

- A sequence that is not periodic is called non-periodic or aperiodic

Basic properties of discrete-time signals…



- Is the above sequence periodic?
- If so, what is the fundamental period?

17

17

Basic properties of discrete-time signals…

- A real-valued sequence $x_\mathrm{e}[n]$ is called even if

$$x_\mathrm{e}[n] = x_\mathrm{e}[-n], \forall n$$

- A real-valued sequence $x_\mathrm{o}[n]$ is called odd if

$$x[-n] = -x[n], \forall n$$

- Any real-valued sequence can be expressed as

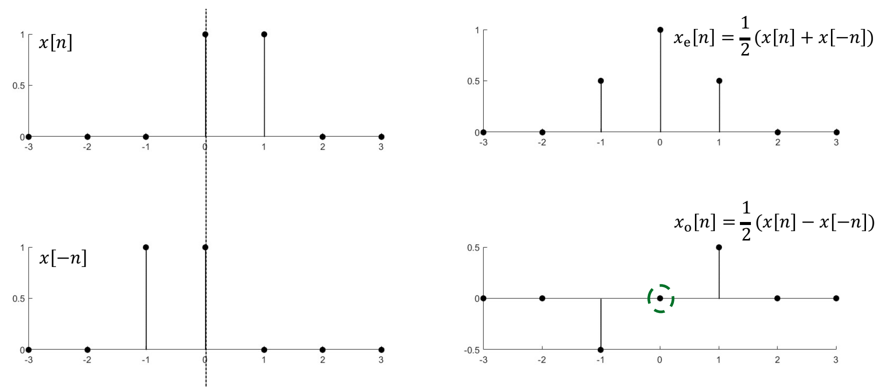
$$x[n] = x_\mathrm{e}[n] + x_\mathrm{o}[n]$$

$$x_\mathrm{e}[n] = \frac{1}{2}(x[n] + x[-n]) \qquad x_\mathrm{o}[n] = \frac{1}{2}(x[n] - x[-n])$$

18

18

Basic properties of discrete-time signals…



19

19

Classifications of discrete-time signals

- A sequence is bounded if $|x[n]| \leq B_x < \infty$ for all n
- A sequence is absolutely summable if

$$\sum_{n=-\infty}^{\infty} |x[n]| < \infty$$

- A sequence is square-summable if its energy

$$E_x = \sum_{n=-\infty}^{\infty} |x[n]|^2 < \infty$$

is bounded. Such signal is called an energy signal

20

20

Classifications of discrete-time signals…

- Not all sequences are energy signals (e.g., periodic signals)
- Average power of sequence $x[n]$ is defined as

$$P_x = \lim_{N\to\infty} \frac{1}{2N+1} \sum_{n=-N}^{N} |x[n]|^2$$

- If P_x is finite, the signal is called a power signal

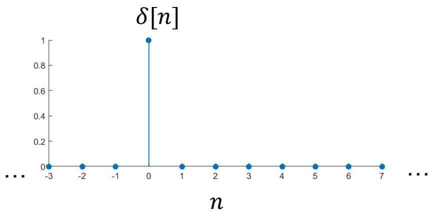
21

21

Basic types of sequences…

- Unit impulse:

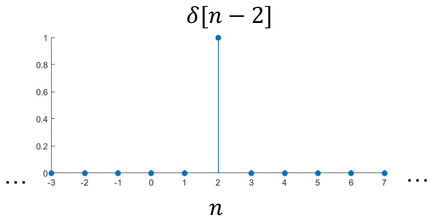
$$\delta[n] = \begin{cases} 1 & n = 0 \\ 0 & n \neq 0 \end{cases}$$



- Delayed unit impulse:

$$\delta[n-k] = \begin{cases} 1 & n = k \\ 0 & n \neq k \end{cases}$$

Matlab
```
k  = 2;
n = (-3:7)
delta = [(n-k)==0];
stem(n,delta)
```



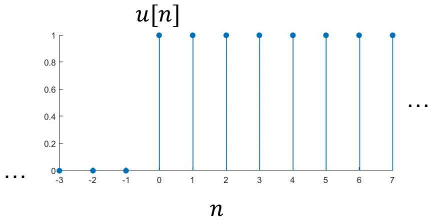
22

22

Basic types of sequences…

- Unit step:

$$u[n] = \begin{cases} 1 & n \geq 0 \\ 0 & n < 0 \end{cases}$$




```
Matlab
n = (-3:7)
u = [n>=0];
stem(n,u)
```


Basic types of sequences…

- Relationship between $u[n]$ and $\delta[n]$:
  - Unit impulse is the first-order difference of the unit step

$$\delta[n] = u[n] - u[n-1]$$

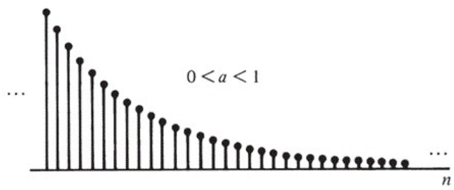
  - Unit step is the running sum of the unit impulse

$$u[n] = \sum_{k=-\infty}^{n} \delta[k]$$

Basic types of sequences…

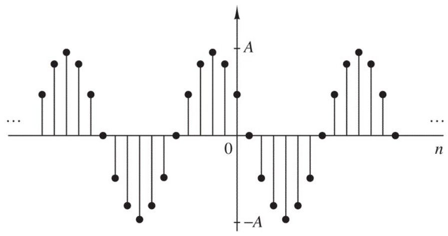
- Real-valued exponential function

$$x[n] = a^n, \forall n \text{ and } a \in \mathbb{R}$$



- What if a is complex-valued, i.e., $a \in \mathbb{C}$?

Discrete-time sinusoid

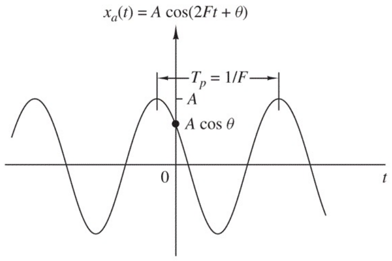


$$x[n] = A\cos[\omega n + \theta] = A\cos[2\pi f n + \theta]$$
$$= \frac{A}{2}\left(e^{j[\omega n+\theta]} + e^{-j[\omega n+\theta]}\right)$$

- What about the notion of frequency in discrete time?
- What about the notion of periodicity for discrete-time sinusoids?

Discrete-time sinusoid…

- Continuous-time sinusoid:



- Consider two signals

$$x_1(t) = A\cos(\Omega_1 t) = A\cos(2\pi F_1 t)$$

$$x_2(t) = A\cos(\Omega_2 t) = A\cos(2\pi F_2 t)$$

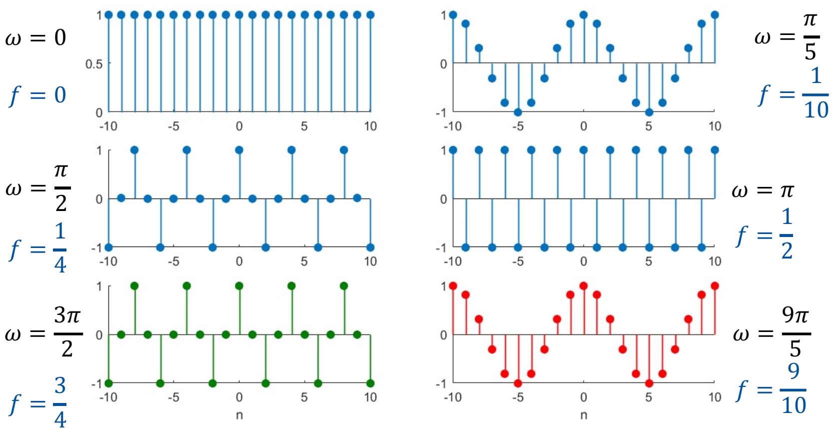
  where $F_2 > F_1$
  - Signal $x_2(t)$ will oscillate faster than $x_1(t)$
  - In general $x_2(t) \neq x_1(t)$, except at some possible points

27

27

Discrete-time sinusoid…

- Digital frequency: $x[n] = \cos[\omega n] = \cos[2\pi f n]$



28

28

Discrete-time sinusoid…

- Discrete-time sinusoid is 2π-periodic in frequency

$$\cos[(\omega + 2k\pi)n] = \cos[\omega n + 2kn\pi] = \cos[\omega n]$$

$\Longrightarrow$ Any sinusoidal sequence with $|\omega| > \pi$ is identical to a sinusoidal sequence with $|\omega| \leq \pi$!

  - Verify this for the green and red sinusoids in previous slide

- Lowest frequency at $\omega_k = 0 + 2\pi k$
- Highest frequency at $\omega_k = \pi + 2\pi k$

$\Longrightarrow$ Range of frequencies is finite

$$-\pi \leq \omega \leq \pi, \text{ or } -\frac{1}{2} \leq f \leq \frac{1}{2}$$

$$(0 \leq \omega \leq 2\pi, \text{ or } 0 \leq f \leq 1)$$

29

29

Discrete-time sinusoid…

- Is a discrete-time sinusoid a periodic sequence?

$$x[n] = x[n + N]?$$

$$\cos[2\pi f\, n] = \cos[2\pi f(n + N)]?$$

- Answer: (Yes/No/Sometimes) [Tick your option]

30

30

Discrete-time sinusoid…

- Answer: Sometimes
- A discrete-time sinusoid is periodic only if its frequency is a rational number

$$\cos[2\pi n] = \cos[2\pi f(n + N)]$$

$$\Rightarrow 2\pi f N = 2\pi k$$

$$\boxed{\Rightarrow f = \frac{k}{N}}$$

31

31

Discrete-time sinusoid…

- Example: Determine if the discrete-time signals below are periodic; if they are, determine their periods

1. $x[n] = \cos\left[\frac{12\pi}{5}n\right]$
2. $x[n] = \sin^2\left[\frac{7\pi}{12}n + \sqrt{2}\right]$
3. $x[n] = \cos[0.02n + 3]$

32

32

Complex exponential

- Complex exponential: $x[n] = Ae^{j[2\pi fn+\theta]}$
- Same properties as discrete-time sinusoids
  - 2π-periodic in (angular) frequency
  - Periodic sequence if frequency f is rational
- Used as building block for discrete-time Fourier representation

33

33

Sampling a sinusoidal signal

- Consider sampling sinusoidal signal at intervals $nT = n/F_s$

$$x_a(t) = A\cos(\Omega t) = A\cos(2\pi Ft)$$

- Discretized signal

$$x[n] = x_a(nT) = A\cos\left[2\pi\frac{F}{F_s}n\right] = A\cos[2\pi fn]$$

$$\Rightarrow f = \frac{F}{F_s} \text{ or } \omega = \Omega T \text{ (relative/normalized frequency)}$$

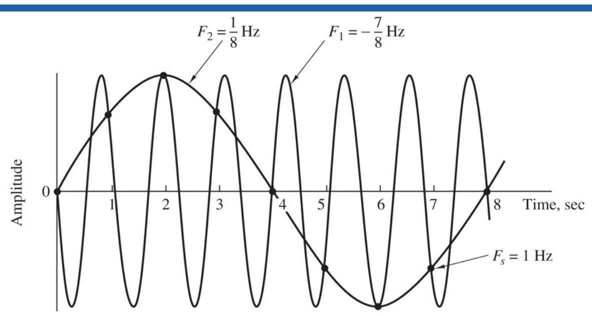
- For accurate representation we know from before

$$-\frac{1}{2} \leq f \leq \frac{1}{2} \quad \Leftrightarrow \quad -\frac{F_s}{2} \leq F \leq \frac{F_s}{2}$$

34

34

Aliasing



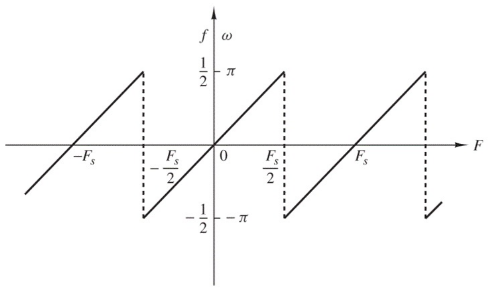
$$A\cos\left[2\pi \frac{F_1}{F_s} n\right] = A\cos\left[2\pi \frac{1}{8} n\right]$$

$$A\cos\left[2\pi \frac{F_2}{F_s} n\right] = A\cos\left[2\pi \frac{(-7)}{8} n\right] = A\cos\left[2\pi \underbrace{\left(\frac{(-7)}{8} + 1\right)}_{=\frac{1}{8}} n\right]$$

35

35

Aliasing…



- Discrete-time versus continuous-time frequency variables in periodic sampling

36

36

Summary

- Today we discussed:
  - Discrete-time signals in time-domain
- Next:
  - Discrete-time systems in time-domain

37

37

NTNU – Trondheim
Norwegian University of
Science and Technology

TTT4120 Digital Signal Processing
Fall 2021

Lecture: Discrete-Time Systems in Time-Domain

Prof. Stefan Werner
stefan.werner@ntnu.no
Office B329

1

Lecture in course book*

- Proakis, Manolakis Digital Signal Processing, 4th Ed.
  - 2.2 Discrete-time systems
  - 2.3 Analysis of discrete-time linear time-invariant systems
  - 2.4 Recursive and non-recursive discrete-time systems
  - 2.4.2 Linear time-invariant systems characterized by constant-coefficient difference equations
  - 2.5.1 Structures for the realization of linear time-invariant systems

*Level of detail is defined by lectures and problem sets

2

2

Contents and learning outcomes

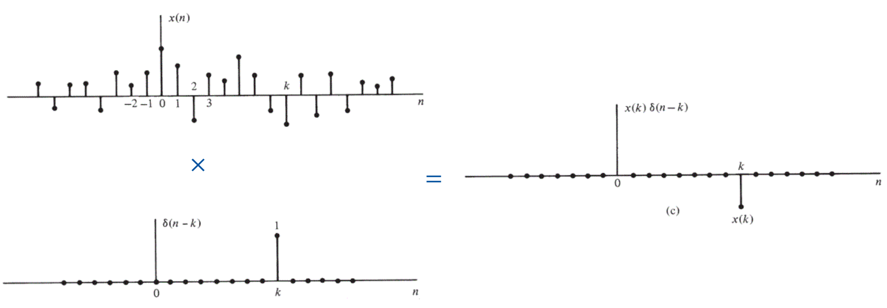
- Signal decomposition using unit impulses
- Discrete-time systems
- Classifications of discrete-time systems
- Linear time-invariant systems and the convolution sum
- Audio demo

3

3

Signal decomposition using unit impulses

- Signal decomposition using sum of delayed unit impulses by exploiting the sifting property: $x[k] = x[n]\delta[n-k]$



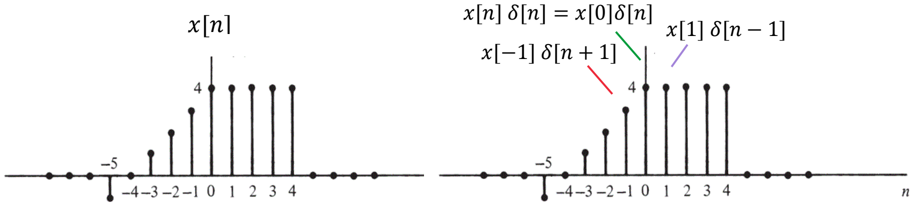
4

4

4

Signal decomposition using unit impulses…

- Discrete-time signals can be represented by scaled shifted impulses, that is, the impulse shifted by k samples is multiplied by $x[k]$



$$x[n] = \cdots x[-1]\,\delta[n+1] + x[0]\,\delta[n] + x[1]\,\delta[n-1] + x[2]\delta[n-2] + \cdots$$

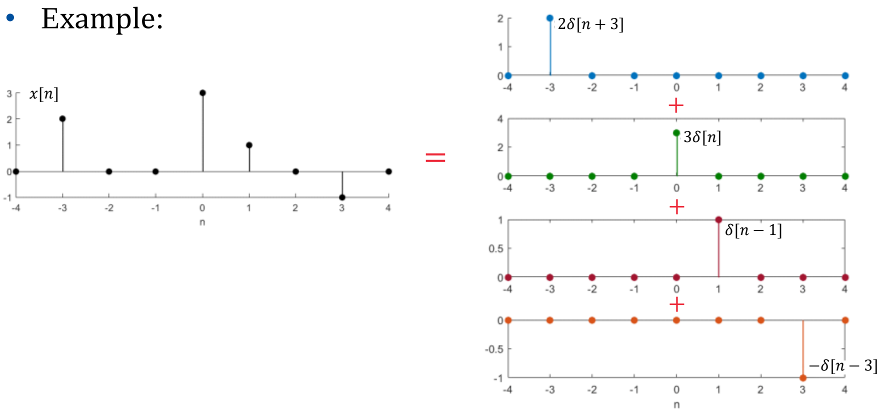
$$\boxed{= \sum_{k=-\infty}^{\infty} x[k]\delta[n-k]}$$

5

5

Signal decomposition using unit impulses…

- Example:



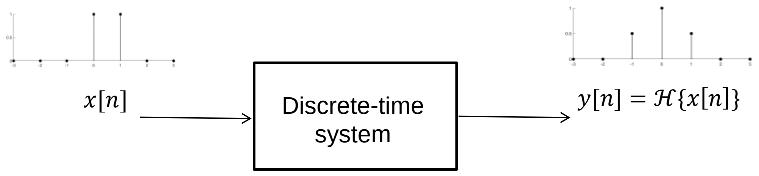
$$x[n] = 2\delta[n+3] + 3\delta[n] + \delta[n-1] - \delta[n-3]$$

6

6

Discrete-time systems

- Discrete-time systems transform (map) an input sequence $x[n]$ to an output sequence $y[n]$



- Mathematically we have

$$y[n] = \mathcal{H}\{x[n]\}$$

where operator $\mathcal{H}$ describes the discrete-time system

7

Discrete-time systems…

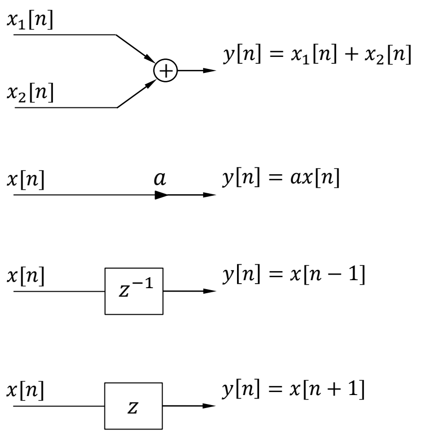
- Graphical representation of building blocks

Adder:

Constant multiplier:

Unit delay:

Unit advance:

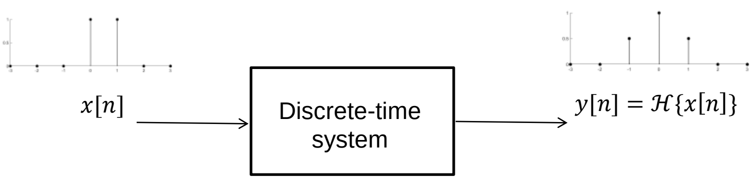


8

Classification of discrete-time systems

9

Classification of discrete-time systems

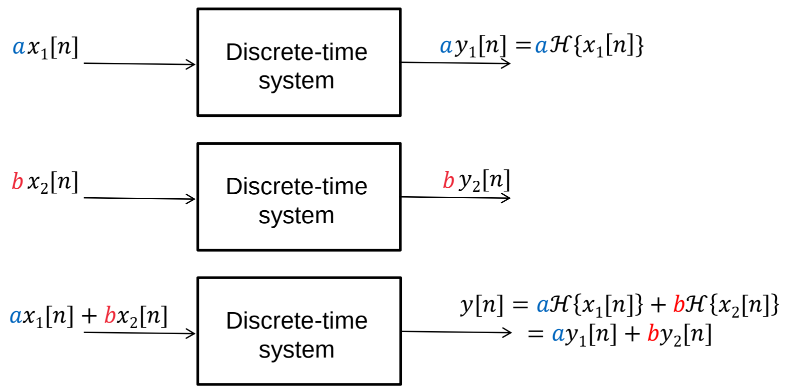


- A discrete-time system can be classified as:
  - linear or nonlinear
  - time invariant or time variant
  - causal or noncausal
- Property must hold for every possible input to the system
  - to disprove a property, need a single counter-example
  - to prove a property, need to prove for the general case

10

10

Linear discrete-time systems



- A linear system is a system for which superposition holds

11

11

Linear discrete-time systems…

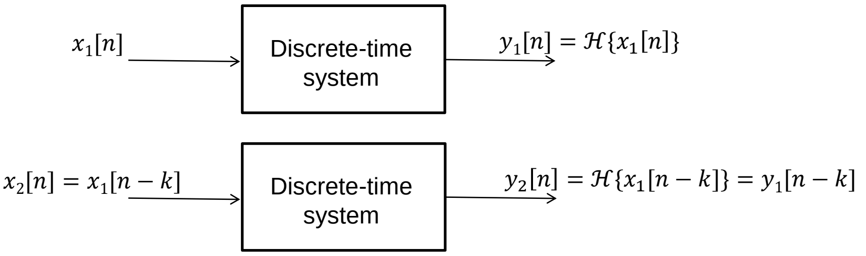
- Linear (L) or nonlinear (NL) system?

1. $y[n] = cx[n]$
2. $y[n] = (n+4)x[n]$
3. $y[n] = x[n + 1]$
4. $y[n] = x[-n]$
5. $y[n] = \sqrt{x[n]} + x^2[n-2]$
6. $y[n] = cx[n] + 3$

12

12

Time-invariant discrete-time systems



- A system whose properties do not vary in time is referred to as being time invariant

13

13

Time-invariant discrete-time systems…

- Time-invariant (TI) or time-variant (TV) system?

  1. $y[n] = cx[n]$
  2. $y[n] = (n+4)x[n]$
  3. $y[n] = x[n+1]$
  4. $y[n] = x[-n]$
  5. $y[n] = \sqrt{x[n]} + x^2[n-2]$
  6. $y[n] = cx[n] + 3$

14

14

Causal versus noncausal systems

- Causal system: output of system at any time n depends only on present and past inputs, i.e.,

$$y[n] = f\{x[n], x[n-1], x[n-2], \dots\}, \forall n$$

- Usually, in the case of a discrete-time signal, a noncausal system is not implementable in real time, since future values are unknown
- Noncausal systems are practical for processing of pre-stored values

Causal versus noncausal systems…

- Causal (C) or noncausal (NC) system?

1. $y[n] = cx[n]$
2. $y[n] = (n+4)x[n]$
3. $y[n] = x[n+1]$
4. $y[n] = x[-n]$
5. $y[n] = \sqrt{x[n]} + x^2[n-2]$
6. $y[n] = cx[n] + 3$

Stability

- A discrete-time system is stable if and only if, for every bounded input, the output is also bounded
- A system is bounded-input bounded-output stable (BIBO) iff

$$|x[n]| \leq M_x < \infty \Longrightarrow |y[n]| \leq M_y < \infty \Longrightarrow, \forall n$$

- We want our systems to behave in a predictable manner

17

17

Stability…

- Stable (S) or unstable (US) system?

1. $y[n] = cx[n]$
2. $y[n] = (n + 4)x[n]$
3. $y[n] = x[n + 1]$
4. $y[n] = x[-n]$
5. $y[n] = \sqrt{x[n]} + x^2[n - 2]$
6. $y[n] = cx[n] + 3$

18

18

Linear time-invariant system

19

Linear time-invariant systems

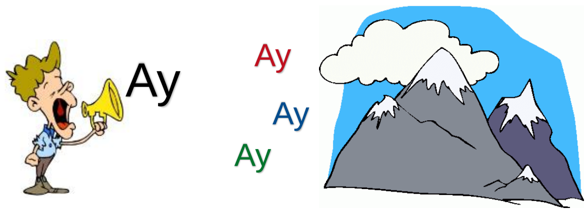
- This course is mostly dealing with linear time-invariant systems
- Knowing the system response to a unit impulse (impulse response), we can calculate the system output for an arbitrary input signal

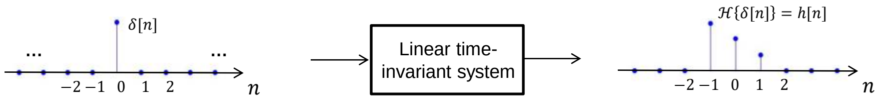
20

20

Impulse response

- Send a short impulse into the system and observe the output

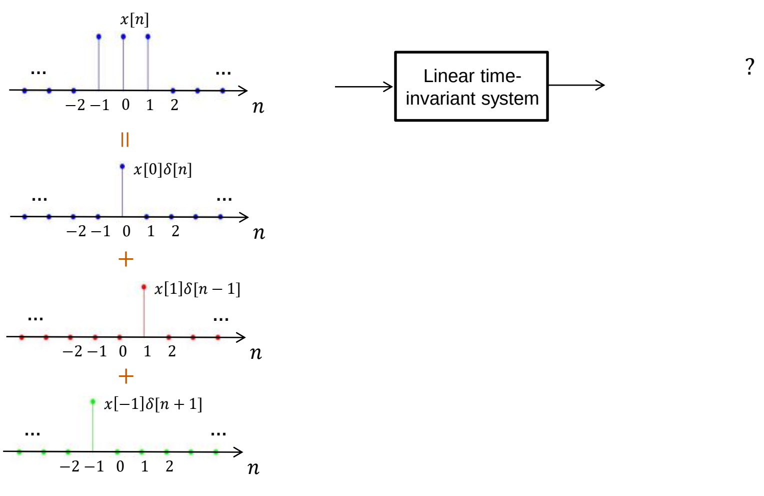




21

21

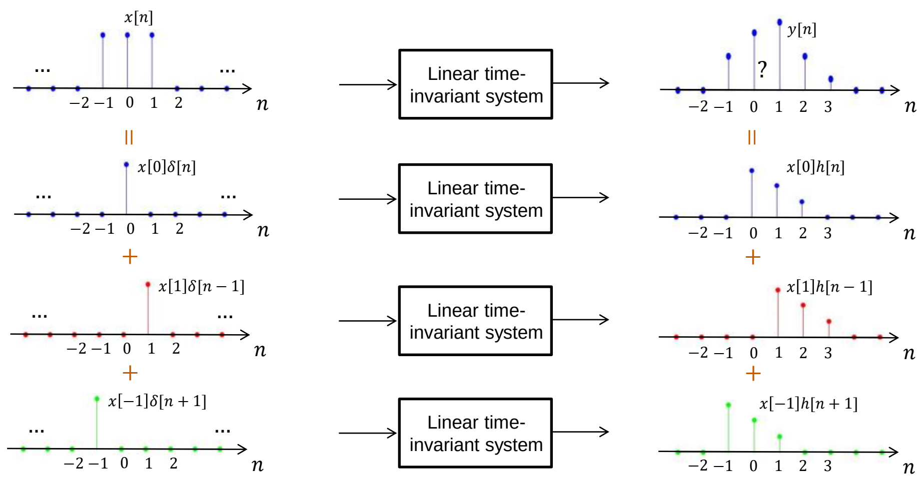
Impulse response and convolution sum



22

22

Impulse response and convolution sum



23

23

Impulse response and convolution sum…

$$x[n] = \cdots x[-1]\delta[n+1] + x[0]\delta[n] + x[1]\delta[n-1] + x[2]\delta[n-2] + \cdots$$

$\downarrow$

Linear time-invariant system

$\downarrow$

$$y[n] = \cdots + x[-1]h[n+1] + x[0]h[n] + x[1]h[n-1] + x[2]h[n-2] + \cdots$$

$$= \sum_k x[k]h[n-k]$$

24

24

Convolution sum

- More formally,

$$y[n] = \mathcal{H}\{x[n]\} = \mathcal{H}\left\{\sum_k x[k]\delta[n-k]\right\}$$

$$= \sum_k x[k]\mathcal{H}\{\delta(n-k)\}$$

$$\boxed{= \sum_k x[k]h[n-k]} = x[n] * h[n]$$

- The output of an LTI system is obtained by convolving (the asterisk operation) its *impulse response* with the *input signal*

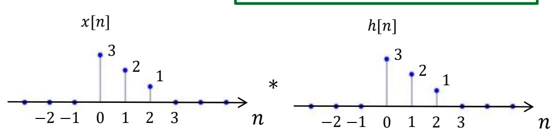
25

25

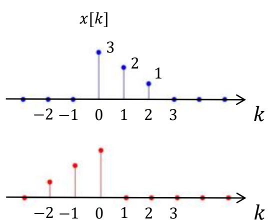
Example: Flip-shift-multiply-sum

- What is the output?

$$\boxed{y[n] = \sum_k x[k]h[n-k]}$$



$$y[0] = \sum_k x[k]h[-k]$$

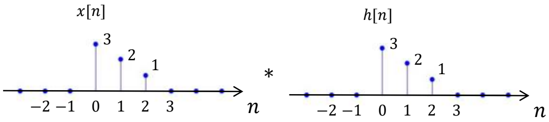


26

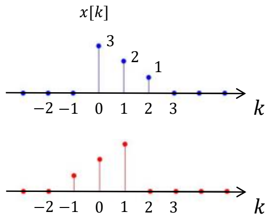
Example: Flip-shift-multiply-sum…

- What is the output?

$$y[n] = \sum_k x[k]h[n-k]$$



$$y[1] = \sum_k x[k]h[1-k]$$



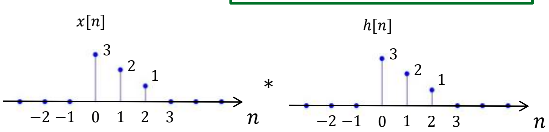
27

27

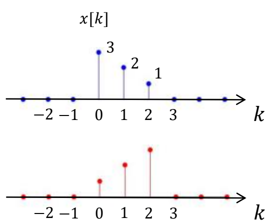
Example: Flip-shift-multiply-sum…

- What is the output?

$$y[n] = \sum_k x[k]h[n-k]$$



$$y[2] = \sum_k x[k]h[2-k]$$



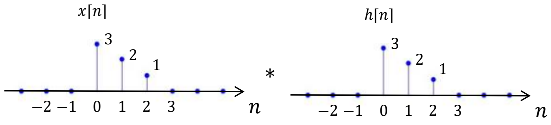
28

28

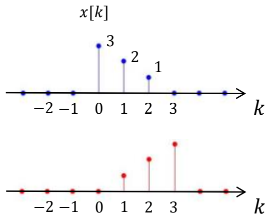
Example: Flip-shift-multiply-sum…

- What is the output?

$$y[n] = \sum_k x[k]h[n-k]$$



$$y[3] = \sum_k x[k]h[3-k]$$



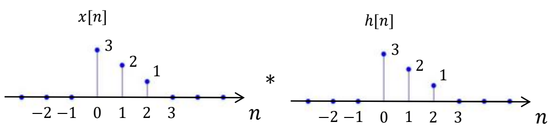
29

29

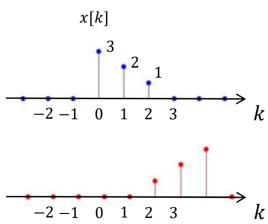
Example: Flip-shift-multiply-sum…

- What is the output?

$$y[n] = \sum_k x[k]h[n-k]$$



$$y[4] = \sum_k x[k]h[4-k]$$



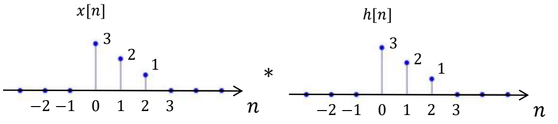
30

30

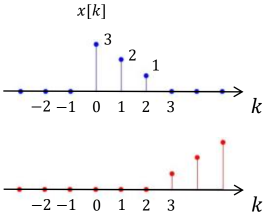
Example: Flip-shift-multiply-sum…

- What is the output?

$$y[n] = \sum_k x[k]h[n-k]$$



$$y[5] = \sum_k x[k]h[5-k]$$



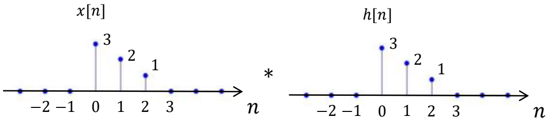
31

31

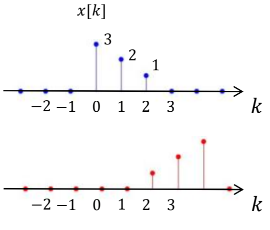
Example: Flip-shift-multiply-sum…

- What is the output?

$$y[n] = \sum_k x[k]h[n-k]$$



$$y[4] = \sum_k x[k]h[4-k]$$



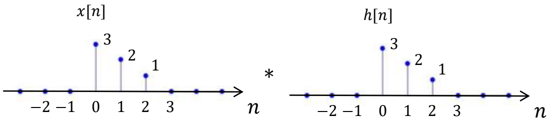
32

32

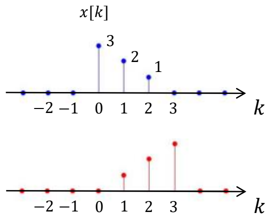
Example: Flip-shift-multiply-sum…

- What is the output?

$$y[n] = \sum_k x[k]h[n-k]$$



$$y[3] = \sum_k x[k]h[3-k]$$



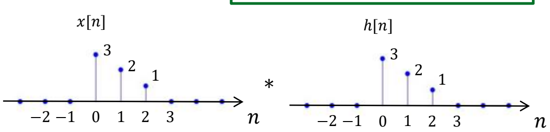
33

33

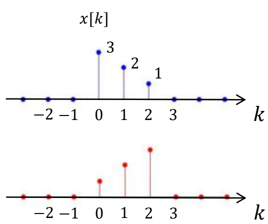
Example: Flip-shift-multiply-sum…

- What is the output?

$$y[n] = \sum_k x[k]h[n-k]$$



$$y[2] = \sum_k x[k]h[2-k]$$



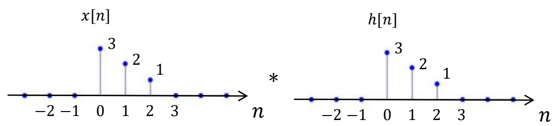
34

34

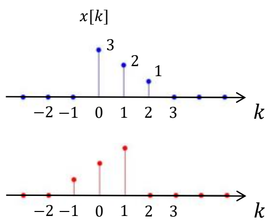
Example: Flip-shift-multiply-sum…

- What is the output?

$$y[n] = \sum_k x[k]h[n-k]$$



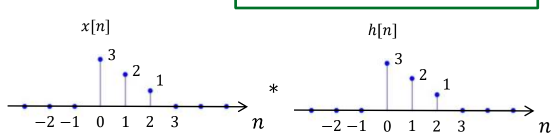
$$y[1] = \sum_k x[k]h[1-k]$$



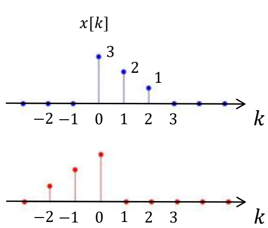
Example: Flip-shift-multiply-sum…

- What is the output?

$$y[n] = \sum_k x[k]h[n-k]$$



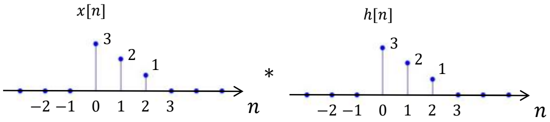
$$y[0] = \sum_k x[k]h[-k]$$



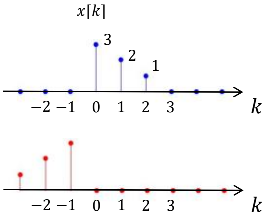
Example: Flip-shift-multiply-sum…

- What is the output?

$$y[n] = \sum_k x[k]h[n-k]$$



$$y[-1] = \sum_k x[k]h[-1-k]$$



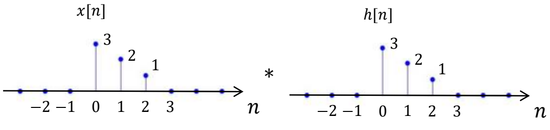
37

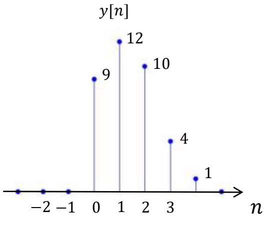
37

Example: Flip-shift-multiply-sum…

- What is the output?

$$y[n] = \sum_k x[k]h[n-k]$$



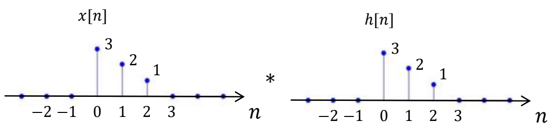


38

38

Example: the easier way

- What is the output? $y[n] = \sum_k x[k]h[n-k]$

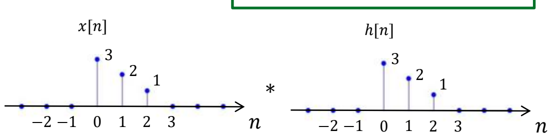


- Convolution matrix: multiply and sum anti-diagonals

$x[n]$ \ $h[n]$	3	2	1
3	9	6	3
2	6	4	2
1	3	2	1

Example: the easiest way

- What is the output? $y[n] = \sum_k x[k]h[n-k]$



- Let the computer do the job

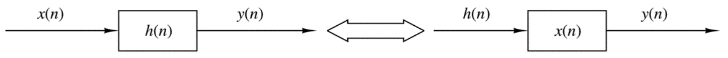
```
Matlab
x = [3 2 1];
h = [3 2 1];
y = conv(x,h)
```

Properties of convolution

- Commutative:

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

$$= \sum_{k=-\infty}^{\infty} x[n-k]h[k] = h[n] * x[n]$$



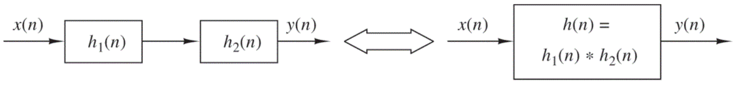
41

41

Properties of convolution…

- Associative:

$$y[n] = (x[n] * h_1[n]) * h_2[n] = x[n] * (h_1[n] * h_2[n])$$



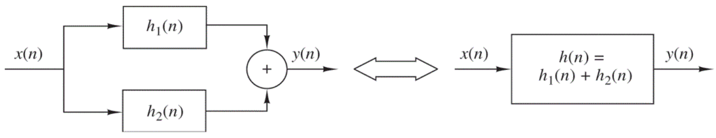
42

42

Properties of convolution…

- Distributive:

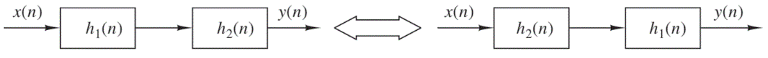
$$y[n] = x[n] * (h_1[n] + h_2[n]) = x[n] * h_1[n] + x[n] * h_2[n]$$



Properties of convolution…

- Properties can be exploited to change order of building blocks
- Order does not matter!

$$y[n] = (x[n] * h_1[n]) * h_2[n] = (x[n] * h_2[n]) * h_1[n]$$



Finite length sequences

- If $x[n]$ has finite length N_x and $h[n]$ has finite length N_h
  $\Longrightarrow y[n]$ has length $N_y = N_x + N_h - 1$

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k] = \sum_{k=0}^{N_x-1} x[k]h[n-k]$$

$$= \{l = n-k\} = \sum_{l=n-N_x+1}^{n} x[n-l]h[l]$$

$$= \sum_{l=n-N_x+1}^{N_h} x[n-l]h[l]$$

- We have $y[n] = 0$ for $n < 0$ and $n - N_x + 1 \geq N_h$

45

Causal linear time-invariant systems

- Output should depend only on past and current inputs

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

$$= \sum_{k=-\infty}^{-1} h[k]x[n-k] + \sum_{k=0}^{\infty} h[k]x[n-k]$$

- Thus, we must have $h[n] = 0,\ n < 0$, for causal systems

46

Stability of linear time-invariant systems

- Input $x[n]$ is bounded: $|x[n]| \leq M_x < \infty$
- A bounded input $x[n]$ to a linear time-invariant system yields a bounded output $y[n]$, $|y[n]| \leq M_y < \infty$ if $\sum_{k=-\infty}^{\infty} |h[k]| < \infty$

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

$$|y[n]| = |\sum_{k=-\infty}^{\infty} h[k]x[n-k]| \leq M_x \sum_{k=-\infty}^{\infty} |h[k]|$$

47

47

FIR and IIR systems

- Infinite(-duration) impulse response (IIR) system is a system whose impulse response $h[n]$ has infinite support

$$y[n] = \sum_{k=0}^{\infty} h[k]x[n-k] \text{ (causal IIR)}$$

- Finite(-duration) impulse response (FIR) system is a system whose impulse response $h[n]$ has finite length

$$y[n] = \sum_{k=0}^{N_h-1} h[k]x[n-k] \text{ (causal FIR)}$$

48

48

Systems described by difference equations

- Characterizing a system using impulse response not always feasible
- An important class of linear time-invariant (IIR) systems can be described by constant-coefficient (real-valued) difference equations

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

- Usually normalized with a_0, i.e., setting $a_0 = 1$

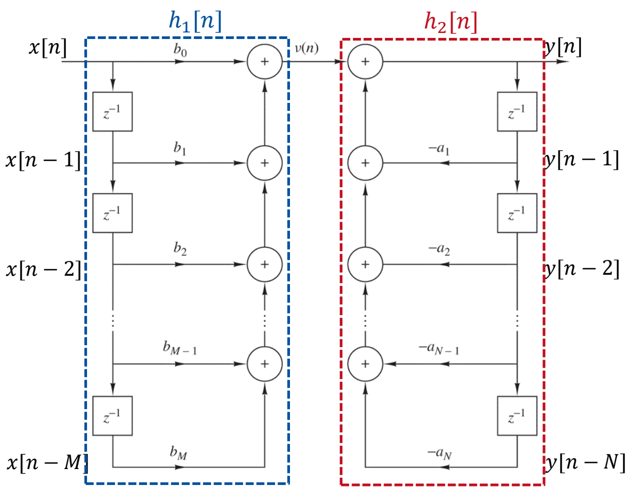
$$y[n] = \sum_{k=0}^{M} b_k x[n-k] - \sum_{k=1}^{N} a_k y[n-k]$$

- Special case of FIR when $a_k = 0, k \geq 1$ and $h[n] = b_n, 0 \leq n \leq M$

```
Matlab
b = [b0, b1,…,bM];
a = [a0,a1,…,aN];
y = filter(b,a,x)
```

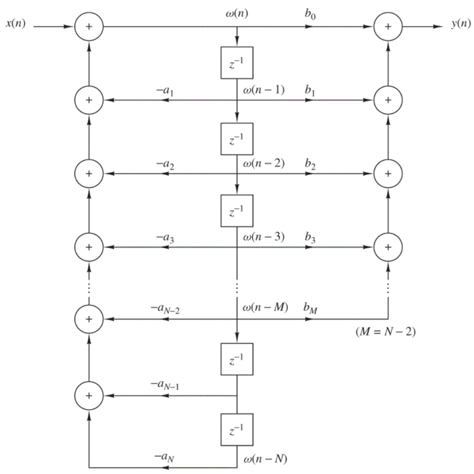

Systems described by difference…

- Graphical representation: Direct form I structure



Systems described by difference…

- Graphical representation: Direct form II structure



51

51

Systems described by difference…

- How to obtain the impulse response $y[n]$ from a difference equation?

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

- Set $x[n] = \delta[n]$ which gives $y[n] = h[n]$

$$h[n] = \sum_{k=0}^{M} b_k \delta[n-k] - \sum_{k=1}^{N} a_k h[n-k]$$

$$= b_n - \sum_{k=1}^{N} a_k h[n-k]$$

- Solve for $h[n]$ sequentially for $n = 1,2, \ldots$
- Requires initial conditions or given a causal system
- Not necessarily closed-form expression

52

52

Systems described by difference…

- General solution can be obtained (see lecture notes)
- Simpler approach is to use transform methods (later)

Matlab

```
b = [b0, b1,…,bM];
a = [a0,a1,…,aN];
h = impz(b,a,n)
```

53

53

Summary

Today:

- Signal decomposition using delayed unit impulses
- Discrete-time systems and classifications
- Linear time-invariant systems

Next:

- Discrete-time Fourier transform

54

54

Audio demo: a tiger in a cathedral

How does it sound when a tiger roars in York Minster?

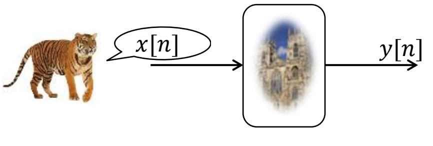
```
Matlab files in ItsLearning:
      tiger_in_york_minster.m
      tiger-growl.wav
      york-minster.wav
```

55

55

Audio demo: a tiger in a cathedral…

- The output of a linear time-invariant system is obtained by convolving its *impulse response* with the *input signal*

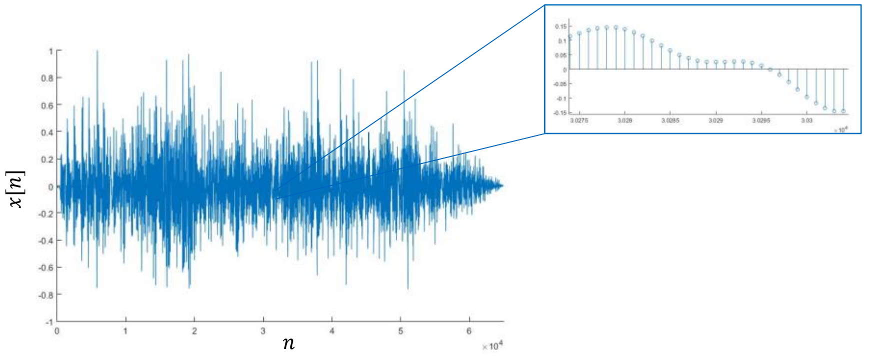


- Consequently, we need
  - The impulse response of the York Minster
  - A tiger growling

56

56

The tiger growl*



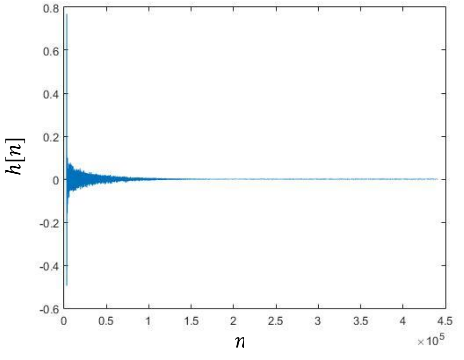
- Tiger growl sampled intervals $nT = \frac{n}{f_s} = n/44100s$

* http://soundbible.com/1485-Tiger-Growling.html

57

57

Impulse reponse of York Minster*



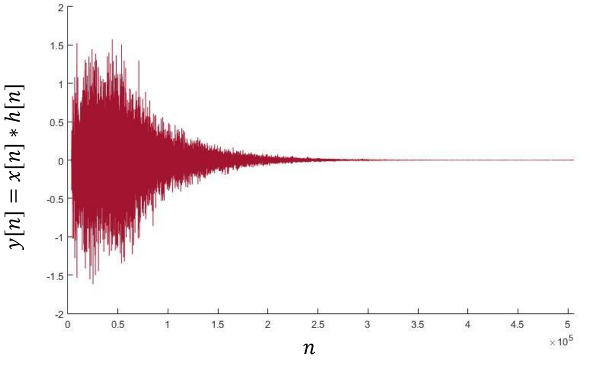
- Impulse response sampled intervals $nT = \frac{n}{f_s} = n/44100$ s

* http://www.openairlib.net/auralizationdb/content/york-minster

58

58

The tiger in York Minster



- Growl signal smears out in time (from 1.4s to 12.2s)

59

59

NTNU – Trondheim
Norwegian University of
Science and Technology

TTT4120 Digital Signal Processing
Fall 2021

Lecture: Discrete Time Systems in Frequency Domain

Prof. Stefan Werner
stefan.werner@ntnu.no
Office B329

Department of Electronic Systems

1

Lecture in course book*

- Proakis, Manolakis Digital Signal Processing, 4th Ed.
  - 4.2.1 Fourier series for discrete-time periodic signals
  - 4.2.3 Fourier transform of discrete-time aperiodic signals
  - 4.3 Frequency-domain and time-domain signal properties
  - 5.1.1 Response to complex exponential and sinusoidal…
  - 5.1.4 Response to aperiodic input signals
  - 5.4.1 Ideal filter characteristics

*Level of detail is defined by lectures and problem sets

2

2

Contents and learning outcomes

- Fourier series for periodic signals
- Fourier transform for aperiodic signals
- Signal properties in time and frequency domains
- Properties of the Fourier transform
- Frequency domain representation of LTI systems – the frequency response function $H(\omega)$

3

3

Frequency analysis of DT signals

- The impulse response of a linear time-invariant system $h[n]$ allows us to compute the response to an arbitrary input $x[n]$

$$x[n] \longrightarrow \boxed{h[n]} \longrightarrow y[n] = h[n] * x[n]$$
$$= \sum_k h[k]x[n-k]$$

- Convolution sum is based on the fact that any input sequence can be decomposed as a linear combination of scaled and delayed unit impulse sequences, $x[n] = \sum_k x[k]\delta[n-k]$
- We can choose to represent the signal using a linear combination of some other basis signals

4

4

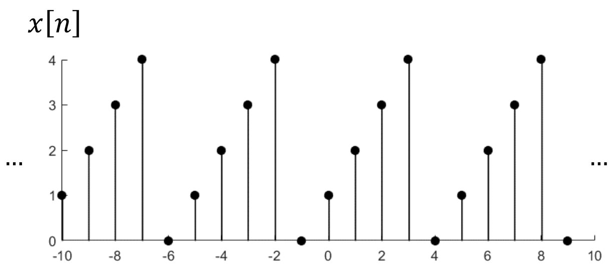
Frequency analysis of DT signals…

- Most signals of practical interest can be decomposed into a sum of sinusoidal components, or complex exponentials
- Using such a combination, a signal is said to be represented in frequency domain
  - Periodic signals $\Longrightarrow$ Fourier series
  - Finite-energy signals $\Longrightarrow$ Fourier transform
- We shall see that this decomposition is very important in the analysis of linear time-invariant systems
  - Response to a sinusoidal input signal is a sinusoid with the same frequency but different amplitude and phase
  - Linear combination of sinusoids at input produces a similar linear combination of sinusoids at output

5

5

Discrete-time Fourier series (DTFS)



- Discrete-time signal $x[n]$ periodic with period N

$$x[n+N] = x[n], \forall n$$

6

6

Discrete-time Fourier series (DTFS)

- Fourier series representation for $x[n]$ consists of a weighted sum of N harmonically related exponentials $e^{j2\pi k/N}$

$$x[n] = \sum_{k=0}^{N-1} c_k e^{j2\pi kn/N} \text{ (synthesis equation)}$$

- Fourier coefficients c_k provide frequency-domain information of $x[n]$ and are given by

$$c_k = \frac{1}{N}\sum_{n=0}^{N-1} x[n]e^{-j2\pi kn/N} \text{ (analysis equation)}$$

- Spectrum of periodic sequence is periodic

$$c_{k+N} = \frac{1}{N}\sum_{n=0}^{N-1} x[n]e^{-j2\pi(k+N)n/N} = c_k$$

7

7

Discrete-time Fourier series (DTFS)…

- Only need to concentrate on a single period in frequency

$$0 \le \omega_k \le 2\pi \quad \text{or} \quad -\pi \le \omega_k \le \pi$$

with $\omega_k = 2\pi k/N$

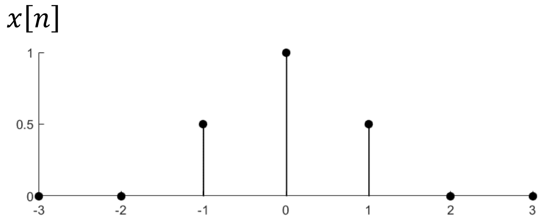
- Periodic signal in time-domain ⇒ discrete spectrum
- Example: $x_1[n] = \cos \pi^2 n$

$$x_2[n] = \cos\frac{\pi n}{4}$$

8

8

Discrete-time Fourier transform (DTFT)



- Discrete-time signal $x[n]$ is *aperiodic* but has *finite energy*

9

9

Discrete-time Fourier transform (DTFT)

- Discrete-time Fourier transform (DFTF) of $x[n]$:

$$X(\omega) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n} \text{ (analysis equation)}$$

- Represents the frequency content of $x[n]$ and is 2π-periodic

$$X(\omega + 2\pi k) = X(\omega)$$

- Frequency range for any discrete-time signal $x[n]$ is limited to $(-\pi, \pi)$ or $(0, 2\pi)$
- We may obtain $x[n]$ from $X(\omega)$

$$x[n] = \frac{1}{2\pi}\int_{-\pi}^{\pi} X(\omega)e^{j\omega n}d\omega \text{ (synthesis equation)}$$

- Notation: $x[n] \overset{\mathcal{F}}{\leftrightarrow} X(\omega)$

10

10

Discrete-time Fourier transform (DTFT)…

- Examples: $x_1[n] \; = \delta[n] \overset{\mathcal{F}}{\leftrightarrow} X_1(\omega) = ?$

$$x_2[n] = ? \overset{\mathcal{F}}{\leftrightarrow} X_2(\omega) \; = \delta(\omega)$$

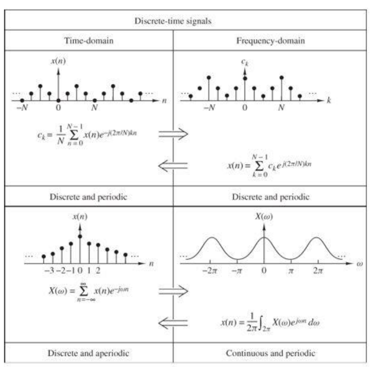
$$x_3[n] \; = a^n u[n] \overset{\mathcal{F}}{\leftrightarrow} X_3(\omega) = ?$$

$$x_4[n] = ? \overset{\mathcal{F}}{\leftrightarrow} X_4(\omega) \; = \begin{cases} 1, |\omega| \le \omega_c < \pi \\ 0, \quad \text{otherwise} \end{cases}$$

11

11

Summary DTFS and DTFT



- Discrete-time signals have periodic spectra
- Periodic signals $\Rightarrow$ discrete spectra $\omega_k = \frac{2\pi k}{N}, \;\; \Delta f \; = 1/N$
- Aperiodic signals have continuous spectra

14

14

Properties of the DTFT

- Symmetry
- Time-shift
- Time-reversal
- Convolution theorem
- Frequency shifting
- Modulation theorem
- Parseval
- Window theorem

Properties of the DTFT…

- Symmetry:
- By expressing $x[n]$ in its real and imaginary parts, i.e.,

$$x[n] = x_R[n] + jx_I[n] \overset{\mathcal{F}}{\leftrightarrow} X_R(\omega) + jX_I(\omega)$$

we can derive a number of symmetry properties

- Example: Real and even signals have real-valued even spectra

$$X(\omega) = X(-\omega) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n} = X_R(\omega) + j \cdot 0$$

- Check all possibilities: real/imag and even/odd
- Example: $x[n]$ imaginary and odd $\Longrightarrow X(\omega)$?

Properties of the DTFT…

- Answer ($x[n]$ imaginary and odd $\Longrightarrow X(\omega)$?):

$$x[n] = x_R[n] + jx_I[n] = jx_I[n] \text{ (imaginary)}$$

$$x[-n] = -x[n] \text{ (odd)}$$

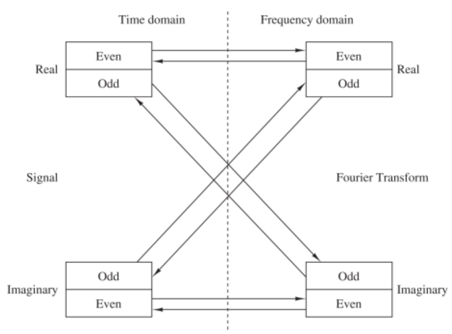
$$\begin{aligned} X(\omega) &= \textstyle\sum_{n=-\infty}^{\infty} jx_I[n]e^{-j\omega n} \\ &= \textstyle\sum_{n=-\infty}^{\infty} jx_I[n](\cos\omega n - j\sin\omega n) \\ &= \textstyle\sum_{n=-\infty}^{\infty}(jx_I[n]\cos\omega n \; - j^2 \sin\omega n) \\ &= 2\textstyle\sum_{n=0}^{\infty} x_I[n]\sin\omega n \quad \text{(Real-valued)} \end{aligned}$$

$$\begin{aligned} X(-\omega) &= 2\textstyle\sum_{n=0}^{\infty} x_I[n]\sin[-\omega n] \\ &= -2\textstyle\sum_{n=0}^{\infty} x_I[n]\sin[\omega n] = -X(\omega) \text{ (Odd)} \end{aligned}$$

17

17

Properties of the DTFT…



- Rewrite signals in terms of odd and even parts

$$x[n] = (x_R^e[n] + jx_I^e[n]) + (x_R^o[n] + jx_I^o[n])$$

$$X(\omega) = (X_R^e(\omega) + jX_I^e(\omega)) + (X_R^o(\omega) + jX_I^o(\omega))$$

18

18

Properties of the DTFT…

- Time-shift: $x[n-k] \overset{\mathcal{F}}{\leftrightarrow} e^{-j\omega k}X(\omega) = |X(\omega)|e^{j(\angle X(\omega)-\omega k)}$
- Time-reversal: $x[-n] \overset{\mathcal{F}}{\leftrightarrow} X(-\omega)$
- Convolution: $x_1[n] * x_2[n] \overset{\mathcal{F}}{\leftrightarrow} X_1(\omega)X_2(\omega)$
- Frequency shifting: $e^{j\omega_0 n}x[n] \overset{\mathcal{F}}{\leftrightarrow} X(\omega-\omega_0)$
- Modulation: $x[n]\cos\omega_0 n \overset{\mathcal{F}}{\leftrightarrow} \frac{1}{2}[X(\omega-\omega_0)+X(\omega+\omega_0)]$
- Parseval: $\sum_n |x[n]|^2 \overset{\mathcal{F}}{\leftrightarrow} \frac{1}{2\pi}\int_{-\pi}^{\pi}|X(\omega)|^2 d\omega$
- Windowing: $x_1[n]x_2[n] \overset{\mathcal{F}}{\leftrightarrow} \frac{1}{2\pi}\int_{-\pi}^{\pi}X_1(\lambda)X_2(\omega-\lambda)d\lambda$

19

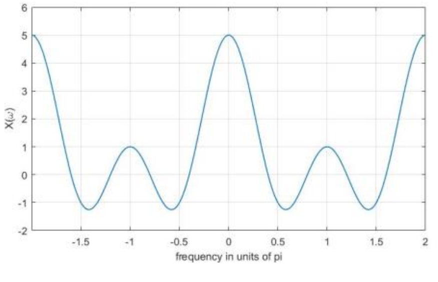
19

Properties of the DTFT…

- Example (symmetry): Pulse in time domain

$$x[n] = \{1,1,\underline{1},1,1\}$$

Sequence $x[n]$ is real and even $\Rightarrow$ $X(\omega)$ is real and even



Matlab
```
n = -2:2; x = ones(1,5);
k = -200:200; w = (pi/100)*k;
X = x * (exp(-j*pi/100)).^(n'*k);
plot(w/pi,real(X));grid
```

21

21

Properties of the DTFT…

- Example (frequency shift):

$$x[n] = \cos\frac{\pi}{2}n\,,\ 0 \le n \le 100$$

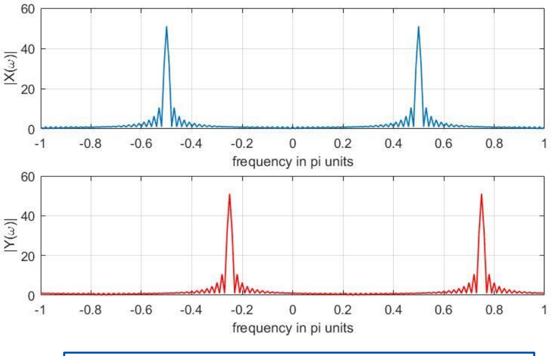
$$y[n] = e^{j\frac{\pi}{4}n}x[n]$$

- Can you guess the shape of the spectra?

22

22

Properties of the DTFT…



Matlab

```
n = 0:100; x = cos(pi*n/2);
k = -100:100; w = (pi/100)*k;
X = x * (exp(-j*pi/100)).^(n'*k);
y = exp(j*pi*n/4).*x;
Y = y * (exp(-j*pi/100)).^(n'*k);
subplot(2,1,1); plot(w/pi,abs(X));
subplot(2,1,2); plot(w/pi,abs(Y));
```

23

23

LTI systems in frequency domain

- Output of linear time-invariant system

$$x[n] \longrightarrow \boxed{h[n]} \longrightarrow y[n] = h[n] * x[n]$$

- What is the output if the input is a complex exponential?

$$x[n] = Ae^{j\omega n}, \ -\infty < n < \infty \text{ and } \omega \in [-\pi, \pi]$$

- Compute the convolution

$$y[n] = h[n] * x[n] = \sum_{k=-\infty}^{\infty} h[k]x[n-k]$$
$$= AH(\omega)e^{j\omega n}$$

with $H(\omega) = \sum_{k=-\infty}^{\infty} h[k]e^{-j\omega n}$ (frequency response)

24

24

LTI systems in frequency domain…

- Using the linearity of LTI systems

$$\sum_k A_k e^{j\omega_k n} \longrightarrow \boxed{h[n]} \longrightarrow \sum_k A_k H(\omega_k) e^{j\omega_k n}$$

- Frequency response is in general complex-valued
$$H(\omega) = |H(\omega)|e^{j\angle H(\omega)}$$
- Magnitude response: $|H(\omega)|$
- Phase response: $\angle H(\omega)$

25

25

LTI systems in frequency domain…

- Response to arbitrary input

$$x[n] \quad X(\omega) \longrightarrow \boxed{h[n]} \longrightarrow y[n] = h[n] * x[n] \quad Y(\omega) = H(\omega)\,X(\omega)$$

- Frequency response is in general complex-valued

$$H(\omega) = |H(\omega)|e^{j\angle H(\omega)}$$

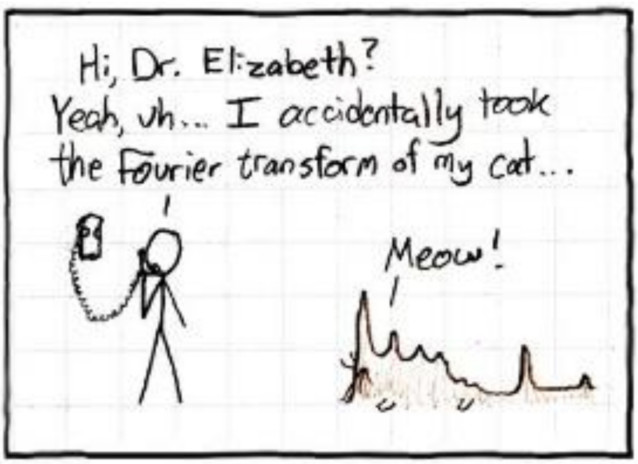
with magnitude response $|H(\omega)|$ and phase response $\angle H(\omega)$

- Frequency response acts like a spectral shaping function
- LTI system that performs spectral shaping is referred to as filter

26

26

LTI systems in frequency domain…

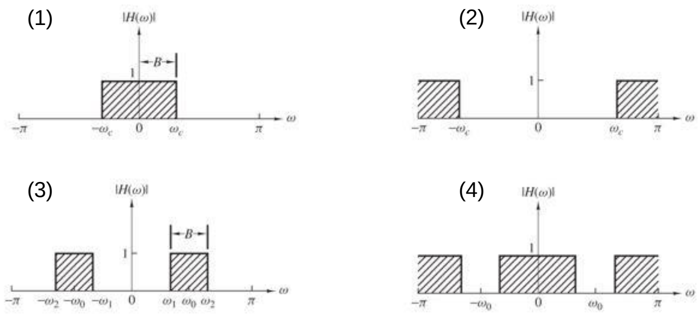


27

27

LTI systems in frequency domain…

- Ideal filter characteristics



- Practical to implement? What is the time-domain impulse response corresponding to (1)?

LTI systems in frequency domain…

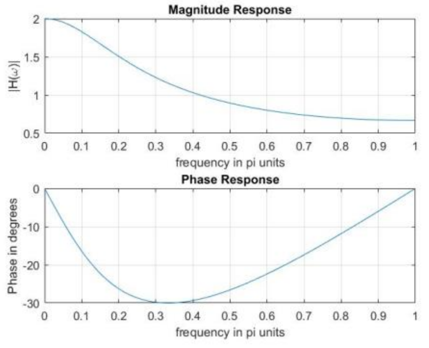
- Example: LTI system $h[n] = 0.5^n u[n]$ excited by $x[n] = e^{j\frac{\pi}{2}n}$

$$x[n] \longrightarrow \boxed{h[n]} \longrightarrow y[n]$$

1) Compute $y[n]$

2) Characterize the type of filter that $h[n]$ represents

LTI systems in frequency domain…



Matlab
```
w = [0:1:500]*pi/500
H = exp(j*w) ./ (exp(j*w) - 0.5*ones(1,501));
magH = abs(H); angH = angle(H);
subplot(2,1,1); plot(w/pi,magH); grid;
subplot(2,1,2); plot(w/pi,angH*180/pi); grid
```

25

30

Summary

Today:
- Signals and systems in frequency-domain
- Discrete-time Fourier series and transform (DTFS & DTFT)
- Filtering using LTI systems and ideal filters

Next:
- Start the journey of z-transforms

26

31

NTNU – Trondheim
Norwegian University of Science and Technology

TTT4120 Digital Signal Processing Fall 2021

Lecture: Z-Transform - Introduction

Prof. Stefan Werner
stefan.werner@ntnu.no
Office B329

1

Lecture in course book*

- Proakis, Manolakis Digital Signal Processing, 4th Ed.
  - 3.1 The z-transform
  - 3.2 Properties of the z-transform
  - 3.3 Rational z-transforms

*Level of detail is defined by lectures and problem sets

2

2

Contents and learning outcomes

- Definition of z-transform and its existence
- Some properties of the z-transform
- Rational z-transforms: poles and zeros

Motivation

- Linear time-invariant system:

$$x[n] \longrightarrow \boxed{h[n]} \longrightarrow y[n] = h[n] * x[n]$$

$$e^{j\omega n} \qquad y[n] = e^{j\omega n} H(\omega)$$

$$X(\omega) \qquad Y(\omega) = H(\omega) X(\omega)$$

- What if $h[n] = 2^n u[n]$?
  - System is unstable $\sum |h[n]|$ not finite
  - DTFT of $h[n]$ does not exist
- Can we analyze such systems using a transform method while retaining the good properties of the DTFT?

Basic idea

- Capture the source of instability or inapplicability of the DTFT
- Apply the DTFT to the modified (captured) signal

5

5

Basic idea…

- Example: Suppose we have signal $x[n] = 2^n u[n]$
  - Problem is due to the exponential growth
  - Capture the signal by multiplying it by a decaying exponential stronger than the growing one, i.e., $r^{-n}x[n]$, $r > 0$
  - What values of r allow for a DTFT for $r^{-n}x[n]$?

$$\sum_{n=-\infty}^{\infty} x[n]r^{-n}e^{-j\omega n} = \sum_{n=0}^{\infty} 2^n r^{-n} e^{-j\omega n}$$

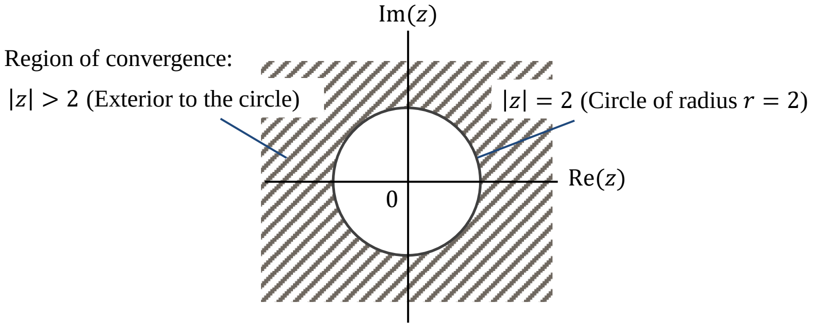
$$= \sum_{n=0}^{\infty} \left(2r^{-1}e^{-j\omega}\right)^n = \frac{1}{1-2r^{-1}e^{-j\omega}}$$

Convergence if $\left|2r^{-1}e^{-j\omega}\right| < 1$ or $r > 2$

6

6

Basic idea…



- Define complex number $z = re^{j\omega}$ in previous expression

$$\sum_{n=0}^{\infty} 2^n r^{-n} e^{-j\omega n} = \sum_{n=0}^{\infty} 2^n z^{-n} = \frac{1}{1-2z^{-1}}, \forall |z| > 2$$

- Convergence has only to do with $r = |z|$ and not ω
- We have a more general transform of the sequence $x[n]$

7

7

Definition of z-transform

- The z-transform of a discrete-time signal $x[n]$ is

$$X(z) = \mathcal{Z}\{x[n]\} = \sum_{n=-\infty}^{\infty} x[n] z^{-n}$$

- Notation: $x[n] \overset{\mathcal{Z}}{\leftrightarrow} X(z) \quad x[n] = \mathcal{Z}^{-1}\{X(z)\}$
- Transforms $x[n]$ into its complex-plane representation $X(z)$
- Transform only exists whenever power series converges
- Region of convergence (ROC) of $X(z)$ is the set of all values of z for which $X(z)$ attains a finite value

8

8

Definition of z-transform…

- Example: Z-transforms of finite-length sequences

$$x_1[n] = \{\underline{1}, 2, 5, 0, 1\}$$
$$= \delta[n] + 2\delta[n-1] + 5\delta[n-2] + \delta[n-4]$$
$$x_2[n] = \{1, \underline{2}, 5, 0, 1\}$$
$$x_3[n] = 2\delta[n]$$

- ROC for finite-length signals is entire z-plane, except possibly when $z \rightarrow 0$ or $z \rightarrow \infty$
  - either z^k or z^{-k} grow unbounded

9

9

Definition of z-transform…

- Example: Compute z-transforms of infinite-length sequences

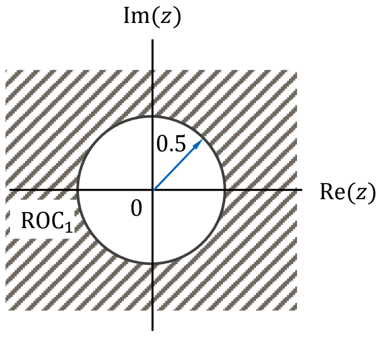
$$x_1[n] = \left(\frac{1}{2}\right)^n u[n]$$

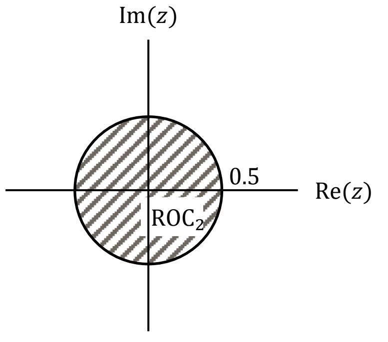
$$x_2[n] = -\left(\frac{1}{2}\right)^n u[-n-1]$$

10

10

Definition of z-transform…





$$X_1(z) = \frac{1}{1 - \frac{1}{2}z^{-1}}$$

$$X_2(z) = \frac{1}{1 - \frac{1}{2}z^{-1}}$$

Definition of z-transform…

- Observations for infinite-duration sequences:
  - z-transform expression alone does not uniquely specify the time-domain signal. ROC resolves ambiguity
  - ROC causal sequence is the exterior of a circle
  - ROC anti-causal sequence is the interior of a circle

ROC of z-transform

- In ROC of $X(z)$, we have $|X(z)| < \infty$
- Using polar form of z, i.e., $z = re^{j\omega}$, we get

$$|X(z)| = \left|\sum_{n=-\infty}^{\infty} x[n]r^{-n}e^{-j\omega n}\right| \leq \sum_{n=-\infty}^{\infty} \left|x[n]r^{-n}e^{-j\omega n}\right|$$

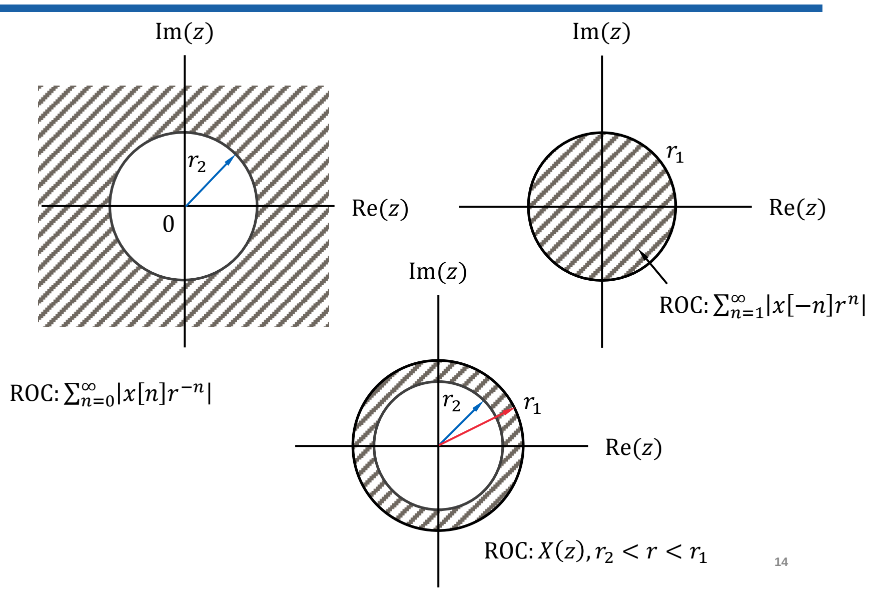
$$\boxed{\leq \sum_{n=1}^{\infty} |x[-n]r^{n}| + \sum_{n=0}^{\infty} |x[n]r^{-n}|}$$

- Observations:
  - both series should converge, ROC = $\mathrm{ROC}_1 \cap \mathrm{ROC}_2$
  - for r sufficiently small, $r < r_1 < \infty$, first sum may converge
  - for r sufficiently large, $r > r_2$, second sum may converge

13

13

ROC of z-transform…



14

14

ROC of z-transform

- Example: Two-sided infinite-length sequences

$$x_1[n] = \left(\frac{1}{2}\right)^n u[n] - \left(\frac{1}{3}\right)^n u[-n-1]$$

$$x_2[n] = \left(\frac{1}{3}\right)^n u[n] - \left(\frac{1}{2}\right)^n u[-n-1]$$

$$X_1(z) = ?, X_2(z) = ?$$

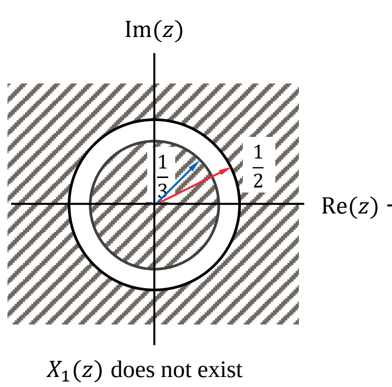
From earlier: $\alpha^n u[n] \overset{\mathcal{Z}}{\leftrightarrow} \frac{1}{(1-\alpha z^{-1})}$, ROC: $|z| > \alpha$

$$-\alpha^n u[-n-1] \overset{\mathcal{Z}}{\leftrightarrow} \frac{1}{(1-\alpha z^{-1})}, \text{ ROC: } |z| < \alpha$$

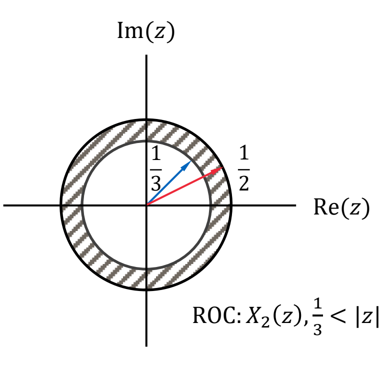
15

15

ROC of z-transform…



$X_1(z)$ does not exist



ROC: $X_2(z), \frac{1}{3} < |z| < \frac{1}{2}$

$$X_2(z) = \frac{1}{1-\frac{1}{3}z^{-1}} + \frac{1}{1-\frac{1}{2}z^{-1}}$$

16

16

Properties of the z-transform

- Linearity
- Time-shift
- Scaling
- Time-reversal
- Convolution
- Differentiation
- Initial value theorem

Properties of the z-transform…

- Linearity:

$$x_3[n] = a_1 x_1[n] + a_2 x_2[n] \overset{z}{\leftrightarrow} X_3(z) = a_1 X_1(z) + a_2 X_2(z)$$

  for <u>any</u> constants a_1 and a_2

- ROC of $X_3(z)$ at least $\mathcal{R}_{X_1} \cap \mathcal{R}_{X_2}$ but can extend beyond intersection
- Example: $x_1[n] = (3 \cdot 2^n - 4 \cdot 3^n)u[n]$

$$x_2[n] = (3 \cdot 2^n + 4 \cdot 3^n)u[n]$$

$$x_3[n] = x_1[n] + x_2[n]$$

Properties of the z-transform…

- Time-shift: $x[n-k] \overset{z}{\leftrightarrow} z^{-k}X(z)$
- ROC of $z^{-k}X(z)$ same as $X(z)$ except at $z = 0$ and $z \to \infty$
- Coefficient of z^{-n} becomes $z^{-(n+k)}$
- Example: $x[n] = \{1, \underline{2}, -1, 0, 3\}$

  $x[n+2]$

  $x[n-2]$

19

19

Properties of the z-transform…

- Scaling: $a^n x[n] \overset{z}{\leftrightarrow} X(a^{-1}z)$
- If ROC of $X(z)$ is $r_1 < |z| < r_2$, then ROC of $X(a^{-1}z)$ is

  $$|a|r_1 < |z| < |a|r_2$$
- Example: $x[n] = 2^n u[n]$

20

20

Properties of the z-transform…

- Time reversal: $x[-n] \overset{Z}{\leftrightarrow} X(z^{-1})$
- If ROC of $X(z)$ is $r_1 < |z| < r_2$, then ROC of $X(z^{-1})$ is

$$1/r_2 < |z| < 1/r_1$$

- Example: $x[n] = u[-n]$

21

21

Properties of the z-transform…

- Convolution: $x[n] = x_1[n] * x_2[n] \overset{Z}{\leftrightarrow} X_1(z)X_2(z)$
- ROC at least the intersection of that of $X_1(z)$ and $X_2(z)$
- Many cases much easier to carry out in z-domain
- Example: $x_1[n] = \{\underline{1}, -1\}$
  $x_2[n] = \{\underline{1}, 1\}$

22

22

Properties of the z-transform…

- Differentiation: $nx[n] \overset{Z}{\leftrightarrow} -z\frac{dX(z)}{dz}$
- ROC convergence stays the same
- Example: $x[n] = na^n u[n]$
- Initial value theorem: $x[0] = \lim_{z\to\infty} X(z)$, $x[n]$ causal

23

23

Rational z-transforms

- Family of transforms where $X(z)$ can be represented as the ratio of two polynomials in z^{-1} (or z)

$$X(z) = \frac{B(z)}{A(z)} = \frac{b_0+b_1z^{-1}+b_2z^{-2}+\dots+b_Mz^{-M}}{a_0+a_1z^{-1}+a_2z^{-2}+\dots+b_Nz^{-N}}$$

$$= \frac{\sum_{k=0}^{M} b_k z^{-k}}{\sum_{k=0}^{N} a_k z^{-k}}$$

$$= \frac{b_0}{a_0}\frac{\sum_{k=0}^{M}(b_k/b_0)z^{-k}}{\sum_{k=0}^{N}(a_k/a_0)z^{-k}} \qquad (a_0, b_0 \neq 0)$$

$$= \frac{b_0}{a_0}\frac{\prod_{k=1}^{M}(1-z_kz^{-1})}{\prod_{k=1}^{N}(1-p_kz^{-1})}$$

24

24

Rational z-transforms…

$$X(z) = \frac{B(z)}{A(z)} = \frac{b_0 \prod_{k=1}^{M}(1 - z_k z^{-1})}{a_0 \prod_{k=1}^{N}(1 - p_k z^{-1})}$$

- The zeros of $X(z)$: values of z for which $X(z) = 0$, $B(z) = 0$
- The poles of $X(z)$: values of z for which $X(z) \to \infty$, $A(z) = 0$
- If a_k and b_k real-valued$\Longrightarrow$ poles (zeros) are either real-valued or must occur in conjugate pairs

- Example: $x_1[n] = \{\underline{1}, 1, 1\}$

  $x_2[n] = \{-\underline{1}, -1, 1\}$

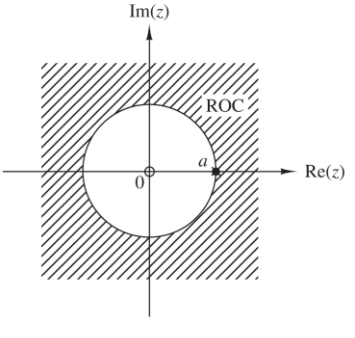
25

25

Rational z-transforms…

- Example (pole-zero plot):

$$x[n] = a^n u[n], a > 0 \overset{z}{\leftrightarrow} X(z) = \frac{1}{1 - az^{-1}}$$



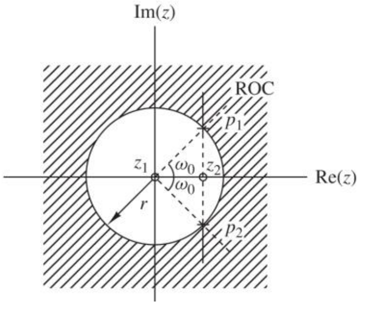
26

26

Rational z-transforms…

- Example (pole-zero plot):

$$x[n] = \left(\frac{5}{6}\right)^n \sin\left(\frac{\pi}{3}n\right) u[n] \overset{Z}{\leftrightarrow} X(z) = \frac{\frac{5}{6}\sin\left(\frac{\pi}{3}\right)z^{-1}}{\left(1-\frac{5}{6}e^{\frac{j\pi}{3}}z^{-1}\right)\left(1-\frac{5}{6}e^{-\frac{j\pi}{3}}z^{-1}\right)}$$



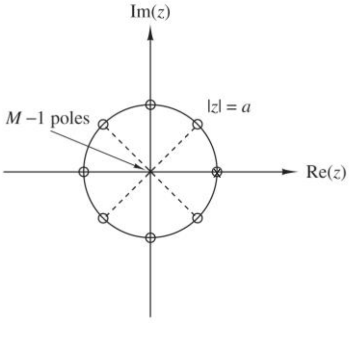
27

27

Rational z-transforms…

- Example (pole-zero plot):

$$x[n] = \begin{cases} a^n, 0 \leq n \leq M-1 \\ 0, \text{elsewhere} \end{cases} \overset{Z}{\leftrightarrow} X(z) = \frac{1-(az^{-1})^M}{1-az^{-1}}$$



28

28

Summary

Today:

- Z-transform and its existence (ROC)
- Properties of the z-transform
- Rational z-transforms: poles and zeros

Next:

- LTI systems: The system function, stability and causality
- Computation and sketching of frequency response function

29

NTNU – Trondheim
Norwegian University of
Science and Technology

TTT4120 Digital Signal Processing
Fall 2020

Lecture: Z-Transform – System Analysis

Prof. Stefan Werner
stefan.werner@ntnu.no
Office B329

Department of Electronic Systems

1

Lecture in course book*

- Proakis, Manolakis Digital Signal Processing, 4th Ed.
  - 4.2.6 Relationship of the Fourier transform to the z-transform
  - 3.5.3 Causality and stability
  - 3.5.6 Stability of second-order systems
  - 5.2.2 Computation of the frequency response

*Level of detail is defined by lectures and problem sets

2

2

Contents and learning outcomes

- LTI systems: The system function, stability and causality
- Computation and sketching of frequency response function

Linear time-invariant systems

- Output of linear time-invariant system

$$x[n] \longrightarrow \boxed{h[n]} \longrightarrow y[n] = h[n] * x[n]$$

$$X(z) \qquad\qquad\qquad Y(z) = H(z)X(z)$$

- By knowing $x[n]$ and observing $y[n]$, we can obtain

$$H(z) = \frac{Y(z)}{X(z)}$$

- Since $H(z) = \sum_n h[n]z^{-n}$, we obtain $h[n] = \mathcal{Z}^{-1}\{H(z)\}$
- Two equivalent descriptions of an LTI system

Linear time-invariant systems…

- Linear time-invariant systems described by constant-coefficient difference equations

$$x[n] \longrightarrow \boxed{h[n]} \longrightarrow \quad y[n] = -\sum_{k=1}^{N} a_k y[n-k] + \sum_{k=0}^{M} b_k x[n-k]$$

$$X(z) \qquad\qquad Y(z) = H(z)X(z)$$

- Rational system function

$$H(z) = \frac{\sum_{k=0}^{M} b_k z^{-k}}{1+\sum_{k=1}^{N} a_k z^{-k}}$$

- Special cases: $a_k = 0$ or $b_k = 0$ for $1 \leq k \leq N$

5

5

Linear time-invariant systems…

- Example: $y[n] = \frac{1}{4} y[n-2] + x[n]$

$$H(z) = \frac{Y(z)}{X(z)} = \frac{1}{1-\frac{1}{4}z^{-2}}$$

6

6

Causality and stability

- Causal linear time-invariant system: $h[n] = 0$ for $n < 0$
- ROC of $H(z)$ must be the exterior of a circle
- Stability of LTI system in terms of system function

$$|H(z)| = \left|\sum_{n=-\infty}^{\infty} h[n]z^{-n}\right| \leq \sum_{n=-\infty}^{\infty} |h[n]||z^{-n}|$$

- If the system is BIBO stable, the unit circle, $z = e^{j\omega}$, is within ROC of $H(z)$. Converse is also true.
- ROC of $H(z)$ can provide information of whether a linear time-invariant system is causal and stable

7

7

Causality and stability…

- In general, if system function is rational, and $N > M$

$$H(z) = b_0 \frac{\prod_{k=1}^{M}(1 - z_k z^{-1})}{\prod_{k=1}^{N}(1 - p_k z^{-1})} = \sum_{k=1}^{N} \frac{C_k}{1 - p_k z^{-1}}$$

- Causal if ROC is the exterior of a circle, $|z| > \max|p_k|$

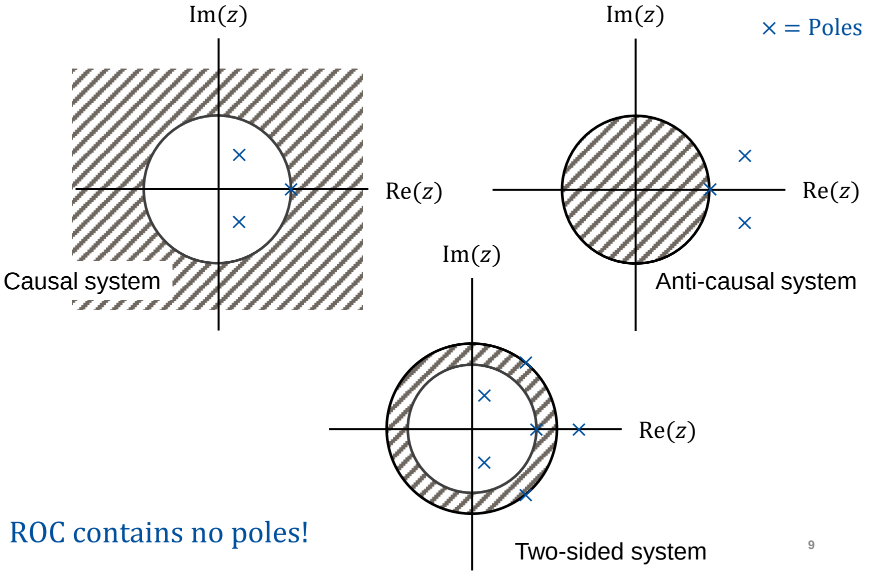
$$h[n] = \sum_{k=1}^{N} C_k p_k^n u[n]$$

- Stable if $\max|p_k| < 1$ (unit circle is included in ROC)

8

8

Causality and stability…

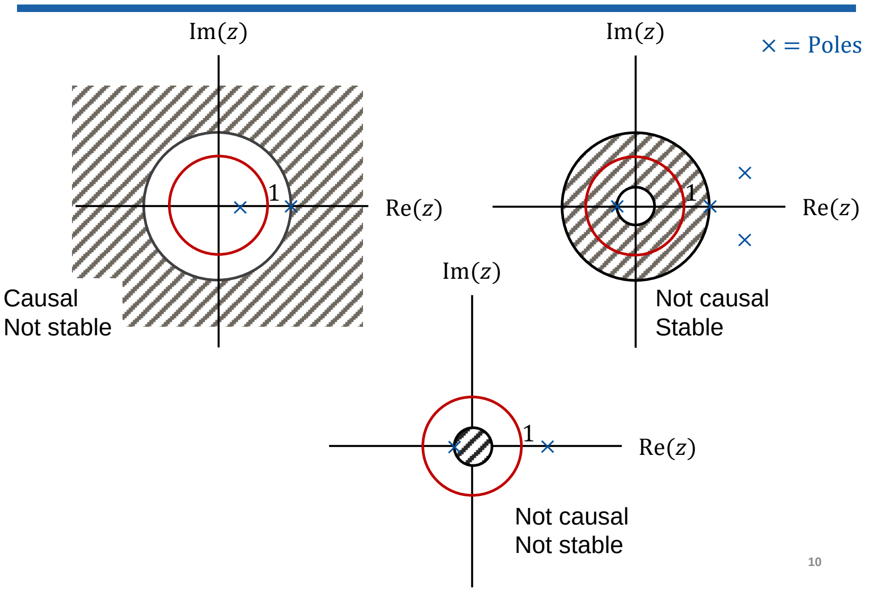


ROC contains no poles!

9

9

Causality and stability…



10

10

Causality and stability…

- Example: $y[n] = \frac{1}{4}y[n-2] + x[n]$

$$H(z) = \frac{Y(z)}{X(z)} = \frac{1}{1-\frac{1}{4}z^{-2}} = \frac{1}{\left(1-\frac{1}{2}z^{-1}\right)\left(1+\frac{1}{2}z^{-1}\right)}$$
$$= \frac{1/2}{1-\frac{1}{2}z^{-1}} + \frac{1/2}{1+\frac{1}{2}z^{-1}}$$

- Causal if $|z| > \frac{1}{2}$ and stable since ROC contains unit circle
- Not causal if $|z| < \frac{1}{2}$ and unstable since unit circle not in ROC

Causality and stability…

- Example: $H(z) = \frac{1}{1-0.5z^{-1}} + \frac{2}{1-3z^{-1}}$

Specify ROC and determine $h[n]$ when

1) system is stable
2) system is causal
3) system is anti-causal

Computation of the frequency response

- The z-transform expressed in polar form

$$X(z)|_{z=re^{j\omega}} = \sum_{n=-\infty}^{\infty}(x[n]r^{-n})e^{-j\omega n}, r_2 < r < r_1$$

- If unit circle, $z = e^{j\omega}$, is within ROC of $X(z)$ we have

$$X(\omega) = X(z)|_{z=re^{j\omega}} = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n}$$

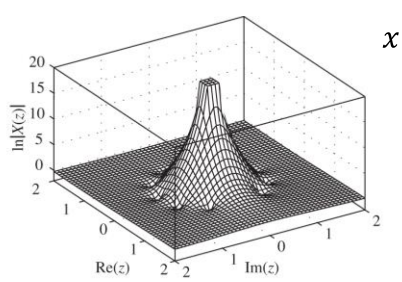
- If $X(z)$ does not converge for $|z| = 1$, Fourier transform does not exist, e.g., $r_2 > 1$

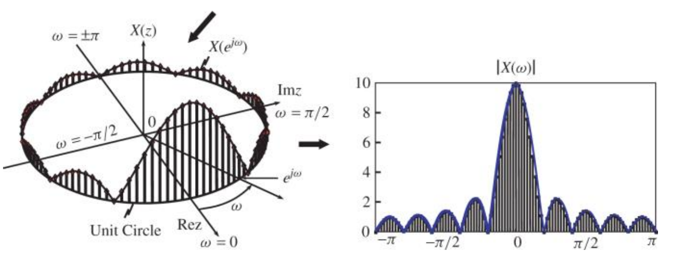
14

14

Computation of the frequency response…

$$x[n] = u[n] - u[n-10]$$





15

Computation of the frequency response…

- The frequency response

$$H(\omega) = H(z)|_{z=re^{j\omega}} = b_0 \frac{\prod_{k=1}^{M}(1 - z_k e^{-j\omega})}{\prod_{k=1}^{N}(1 - p_k e^{-j\omega})}$$

$$= b_0 e^{j(N-M)\omega} \frac{\prod_{k=1}^{M}(e^{j\omega} - z_k)}{\prod_{k=1}^{N}(e^{j\omega} - p_k)}$$

- Product of frequency-dependent distance-vectors in z-plane

$$e^{j\omega} - z_k = V_k e^{j\Theta_k(\omega)}$$

$$e^{j\omega} - p_k = U_k e^{j\Phi_k(\omega)}$$

- If we know z_k and p_k we can plot/sketch the frequency response and phase response

16

16

Computation of the frequency response…

- The magnitude of frequency response

$$|H(\omega)| = |b_0| \frac{\prod_{k=1}^{M}|e^{j\omega} - z_k|}{\prod_{k=1}^{N}|e^{j\omega} - p_k|} = |b_0| \frac{\prod_{k=1}^{M} V_k}{\prod_{k=1}^{N} U_k}$$

- Phase response:

$$\angle H(\omega) = \angle b_0 e^{j(N-M)\omega} \frac{\prod_{k=1}^{M} V_k e^{j\Theta_k(\omega)}}{\prod_{k=1}^{N} U_k e^{j\Phi_k(\omega)}}$$

$$= \angle b_0 + (N - M)\omega + \sum_{k=1}^{M} \Theta_k(\omega) - \sum_{k=1}^{N} \Phi_k(\omega)$$

17

17

Computation of the frequency response…

- Example:

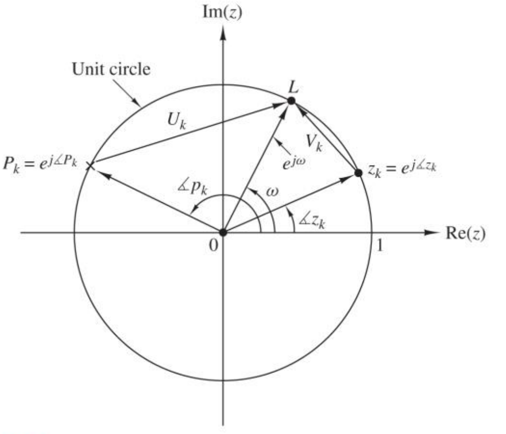


Figure 5.2.2 A zero on the unit circle causes $|H(\omega)| = 0$ and $\omega = \measuredangle z_k$. In contrast, a pole on the unit circle results in $|H(\omega)| = \infty$ at $\omega = \measuredangle p_k$.

18

18

Computation of the frequency response…

- Example: Sketch the frequency response of systems from the pole-zero plot

$$H_1(z) = \frac{1}{1 - 0.5z^{-1}}$$

Matlab

```
B = 1;
A = [1 -0.5];
figure(1)
zplane(B,A)

figure(2)
[H,W]=freqz(B,A);
plot(W/pi,abs(H));
```

19

19

Computation of the frequency response…

- Another Matlab example:
  Sketch the frequency response of system using `zplane(B,A)`

$$H(z) = \frac{B(z)}{A(z)}$$

with `B = fircls1(8,0.3,0.02,0.008);`
and `A = [1]`

- Verify using `[H,W]=freqz(B,A), plot(W/pi,abs(H))`

20

20

Summary

Today:

- LTI systems: causality and stability
- System function
- Computation and sketch of frequency response from the system function

Next:

- Some simple filters and properties
- Why do we want linear phase filters?
- Minimum-phase and inverse systems

21

21

NTNU – Trondheim
Norwegian University of
Science and Technology

TTT4120 Digital Signal Processing
Fall 2021

Lecture: Filter Properties and Inverse Systems

Prof. Stefan Werner
stefan.werner@ntnu.no
Office B329

1

Lecture in course book*

- Proakis, Manolakis Digital Signal Processing, 4^{th} Ed.
  - 5.4.2 Lowpass, highpass, and bandpass filters
  - 5.4.3 Digital resonators
  - 5.4.4 Notch filters
  - 5.4.5 Comb filters
  - 5.4.6 All-pass filters
  - 5.4.1 Ideal filter characteristics
  - 10.2.1 Symmetric and antisymmetric FIR filters
  - 5.5 Inverse systems and deconvolution

*Level of detail is defined by lectures and problem sets

2

2

Contents and learning outcomes

- Some simple filter properties
- Why linear phase?
- Minimum-phase and inverse systems

Ideal filter characteristics

- Systems may be designed to amplify or attenuate parts of the input signal (e.g., remove noise, suppress interference)

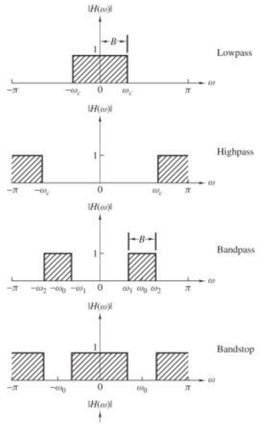
$$x[n] \longrightarrow \boxed{h[n]} \longrightarrow y[n] = h[n] * x[n]$$

$$X(\omega) \qquad Y(\omega) = H(\omega)X(\omega)$$

$$X(z) \qquad Y(z) = H(z)X(z)$$

- Frequency response $H(\omega)$ shapes the spectrum of the input signal to have a desired form

Linear time-invariant systems…



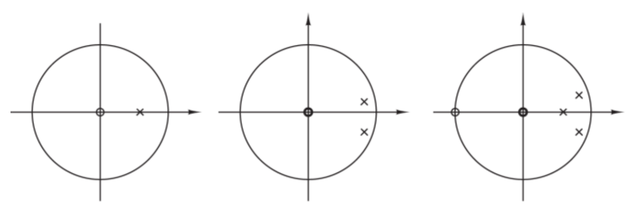
- Passband, stopband, cutoff frequencies
- Cannot get this kind of shapes using a causal impulse response with a finite number of coefficients (later)

5

5

Lowpass

- Poles close(r) to $z = 1$ and zeros close(r) to $z = -1$. Why?



Lowpass

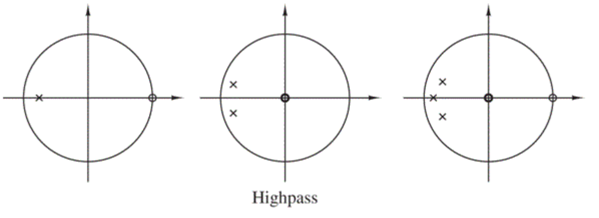
- Example: $H(z) = \frac{1-a}{2} \cdot \frac{1+z^{-1}}{1-az^{-1}}$

6

6

Highpass

- Poles close(r) to $z = -1$ and zeros close(r) to $z = 0$

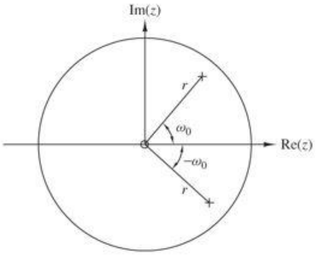


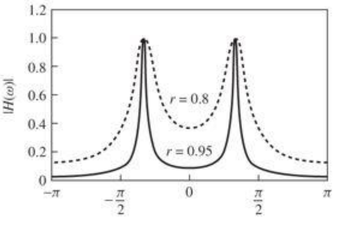
- Reflect poles-zeros of lowpass around imaginary axis
- Frequency translation: $H_{\text{hp}}(\omega) = H_{\text{lp}}(\omega - \pi)$
- Example: $H_{\text{lp}}(z) = \frac{1-a}{2} \cdot \frac{1+z^{-1}}{1-az^{-1}} \rightarrow H_{\text{hp}}(z) = \frac{1-a}{2} \cdot \frac{1-z^{-1}}{1+az^{-1}}$

7

7

Digital resonator





- Complex-conjugate poles $p_{1,2} = re^{\pm j\omega_0}$ close to $|z| = 1$

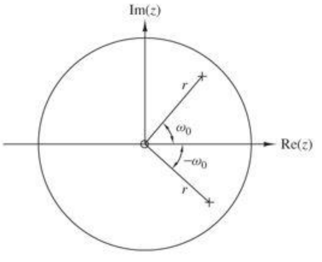
$$H(z) = \frac{b_0}{(1-re^{j\omega_0}z^{-1})(1-re^{-j\omega_0}z^{-1})}$$

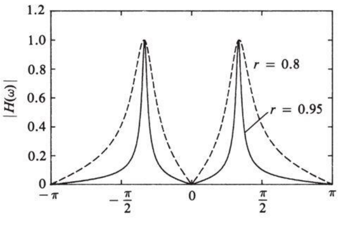
- Resonant peak can be computed: $\omega_r = \cos^{-1}\left(\frac{1+r^2}{2r}\cos\omega_0\right)$
- For $r \approx 1$, $\omega_r \approx \omega_0$

8

8

Digital resonator…





- Complex-conjugate poles $p_{1,2} = re^{\pm j\omega_0}$ and zeros $z_{1,2} = \pm 1$

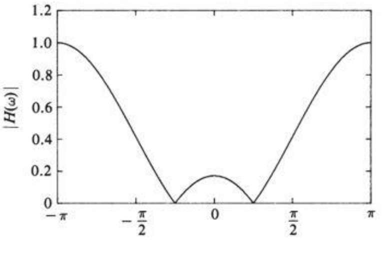
$$H(z) = \frac{(1+z^{-1})(1-z^{-1})}{(1-re^{j\omega_0}z^{-1})(1-re^{-j\omega_0}z^{-1})}$$

- Exact location of resonant peak harder to find analytically

9

9

Notch filter



- A filter that contains deep notches in its frequency response
- Removing powerline frequency disturbance
- Create nulls by complex-conjugate zeros on the unit circle

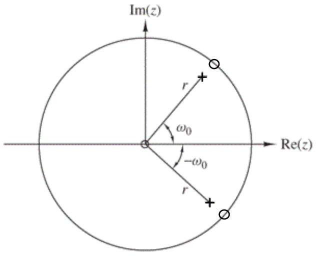
$$H(z) = b_0\left(1 - e^{j\omega_0}z^{-1}\right)\left(1 - e^{-j\omega_0}z^{-1}\right)$$

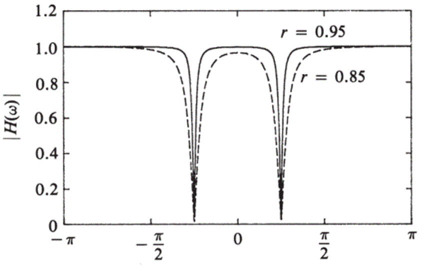
- Large bandwidth is a problem with FIR notch filters

10

10

Notch filter…





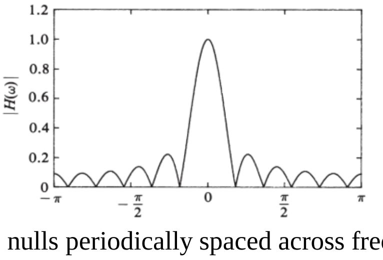
- Introduce poles close to unit circle reduces notch bandwidth

$$H(z) = \frac{b_0\left(1-e^{j\omega_0}z^{-1}\right)\left(1-e^{-j\omega_0}z^{-1}\right)}{\left(1-re^{j\omega_0}z^{-1}\right)\left(1-re^{-j\omega_0}z^{-1}\right)}$$

11

11

Comb filter



- Notch filter with nulls periodically spaced across frequency
- Simple moving average (FIR) filter

$$H(z) = \sum_{k=0}^{M-1} z^{-k} = \frac{1-z^{-M}}{1-z^{-1}}$$

- We may also construct a comb filter by replacing z with z^L

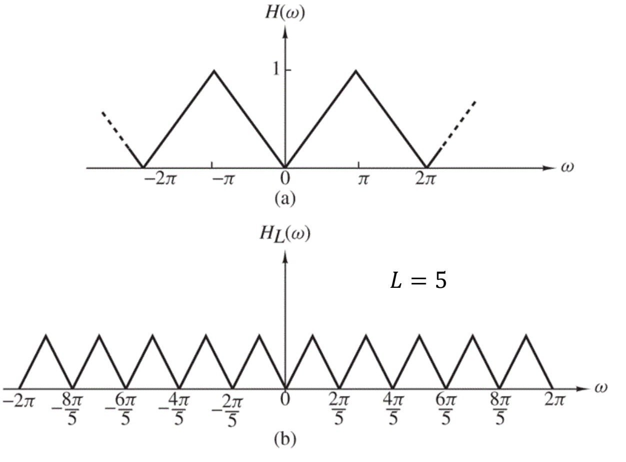
$$H_L(z) = \sum_{k=0}^{M} h[k]z^{-Lk} \Leftrightarrow H_L(\omega) = H(L\omega)$$

12

12

Comb filter…

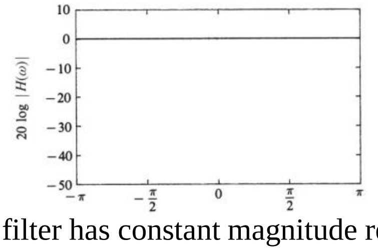
$$H_L(z) = \sum_{k=0}^{M} h[k]z^{-Lk} \Leftrightarrow H_L(\omega) = H(L\omega)$$



13

13

All-pass filters



- All-pass filter has constant magnitude response
- Can be used to compensate poor phase characteristics

$$H(z) = \frac{a_N + a_{N-1}z^{-1} + \cdots + a_1 z^{-N+1} + z^{-N}}{1 + a_1 z^{-1} + \cdots + a_{N-1}z^{-N+1} + a_N z^{-N}} = \frac{z^{-N}A(z^{-1})}{A(z)}$$

- Assuming real coefficients

$$H(z) = \frac{z^{-N}A(z^{-1})}{A(z)} \Leftrightarrow |H(\omega)|^2 = H(z)H(z^{-1})|_{z=e^{j\omega}} = 1$$

14

14

Linear phase filters…

- Why linear phase filters, i.e., $\measuredangle H(\omega) = a + b\omega$?

$$H(\omega) = |H(\omega)|e^{j\measuredangle H(\omega)}$$

- Compare the two ideal lowpass specifications

$$H_1(\omega) = \begin{cases} 1, & |\omega| \leq \omega_c \\ 0, & |\omega| \leq \omega_c \leq \pi \end{cases}$$

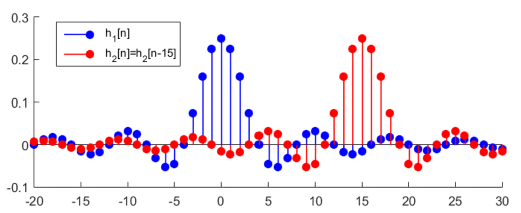
$$H_2(\omega) = \begin{cases} e^{-jn_d\omega}, & |\omega| \leq \omega_c \\ 0, & |\omega| \leq \omega_c \leq \pi \end{cases}$$

- How about the time-domain pulses?

15

15

Linear phase filters…



Matlab

```
n  = -20:30;
wc = 2*pi*1/8;
x = wc/pi;
nd = 15;
stem(n,x*sinc(x*n),'r')
hold on
stem(n,x*sinc(x*(n-nd)))
```

- How about the time-domain pulses?

$$h_{1[n]} = \frac{w_c}{\pi}\frac{\sin[w_c n]}{w_c n}$$

$$h_2[n] = \frac{w_c}{\pi}\frac{\sin[w_c(n-n_d)]}{[w_c(n-n_d)]} \Longrightarrow h_2[n] = h_1[n - n_d]$$

- Delays the output signal with n_d samples, no signal distortion!

16

16

Linear phase filters…

- Filter design in general (later in the course):

$$\min_{a,b} ||E(z)|| = \min_{a,b} \left\| H_{\text{des}}(z) - \frac{\sum_{k=0}^{M-1} b_k z^{-k}}{\sum_{k=0}^{N-1} a_k z^{-k}} \right\|$$

- Consider FIR filters having a frequency response of the form

$$H(\omega) = H_{\text{r}}(\omega) e^{-j(\omega d + c)},\ H_{\text{r}}(\omega) \text{ real-valued}$$

- We want a pure signal delay in passband
- Obtained by choosing $h[k]$ real and $h[k] = \pm h[M-1-k]$
  - Symmetric or antisymmetric

17

17

Linear phase filters…

- Example: FIR with $M = 5 \Longrightarrow N = 2$

$$H(z) = h[0] + h[1]z^{-1} + h[2]z^{-2} + h[3]z^{-3} + h[4]z^{-4}$$
$$= z^{-2}\{h[0]z^2 + h[1]z + h[2] \pm h[1]z^{-1} \pm h[0]z^{-2}\}$$
$$= z^{-2}\{h[2] + h[0][z^2 \pm z^{-2}] + h[1][z^1 \pm z^{-1}]\}$$

- Frequency response symmetric filter (take the '+' signs):

$$H(z)\Big|_{e^{j\omega}} = e^{-j2\omega}\{h[2] + 2h[0]\cos 2\omega + 2h[1]\cos\omega\}$$

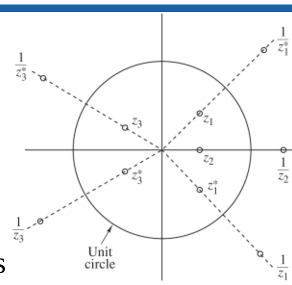
- Frequency response antisymmetric filter (take the '-' signs):

$$H(z)\Big|_{e^{j\omega}} = je^{-j2\omega}\{2h[0]\sin 2\omega + 2h[1]\sin\omega\}$$

18

18

Linear phase filters…



- Zeros of $H(z)$ occur in reciprocal pairs
- Example (cont.): Symmetric FIR with $M = 5$ $(N = 2)$

$$H(z) = h[0] + h[1]z^{-1} + h[2]z^{-2} + h[3]z^{-3} + h[4]z^{-4}$$

$$= z^{-2}\{h[0]z^{2} + h[1]z + h[2] + h[1]z^{-1} + h[0]z^{-2}\}$$

$$= z^{-2}\{h[2] + h[0][z^{2} + z^{-2}] + h[1][z^{1} + z^{-1}]\}$$

$$= z^{-4}H(z^{-1})$$

19

19

Inverse and minimum-phase systems

$x[n]$ $\quad$ $h[n]$ $\quad$ $y[n] = h[n] * x[n]$

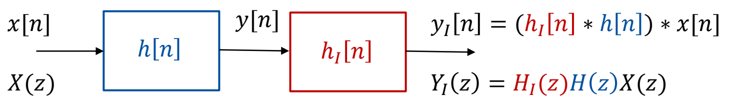
$X(z)$ $\quad\quad\quad$ $Y(z) = H(z)X(z)$

- What if we are given $y[n]$ and want to determine $x[n]$?
  - Information signal passing through communication channel

20

20

Inverse and minimum-phase systems



- If system $\mathcal{T}$ is invertible, $x[n]$ and can be recovered from $y[n]$

$$x[n]=\mathcal{T}^{-1}\{y[n]\} = \mathcal{T}^{-1}\{\mathcal{T}\ [x[n]]\}$$

- Linear time-invariant systems

$$h[n] * h_I[n] = \delta[n] \overset{z}{\leftrightarrow} H(z)H_I(z) = 1$$

- Solving for $h_I[n]$ usually simpler in z-domain, especially if $H(z)$ is rational, i.e., $H(z) = B(z)/A(z)$

21

21

Inverse and minimum-phase systems…

- Example: Determine inverse system $h[n] = \delta[n] - \frac{1}{3}\delta[n-1]$
- Time-domain solution ($h_I[n]$ causal and stable)

$$h[n] * h_I[n] = \delta[n] \ \Leftrightarrow \sum_{k=0}^{n} h[k]h_I[n-k] = \delta[n]$$

- Solution via z-transform

$$H(z)H_I(z) = 1 \ \Leftrightarrow H_I(z) = 1/H(z)$$

Remember to specify ROC of $H_I(z)$ (two possibilities)!

22

22

Inverse and minimum-phase systems…

- Solution via z-transform

$$H(z)H_I(z) = 1 \;\Leftrightarrow H_I(z) = 1/H(z)$$

Remember to specify ROC of $H_I(z)$ (two possibilities)!

$$H(z) = 1 - \frac{1}{3}z^{-1} \text{ ROC: } |z| \neq 0 \rightarrow H_I(z) = \frac{1}{1-\frac{1}{3}z^{-1}}$$

- Corresponds to either

$$h_I[n] = -\left(\frac{1}{3}\right)^n u[-n-1],\;\text{ ROC: } |z| < \frac{1}{3} \text{ (anti-causal unstable) or}$$

$$h_I[n] = \left(\frac{1}{3}\right)^n u[n],\;\text{ ROC: } |z| > \frac{1}{3} \text{ (causal stable)}$$

23

23

Inverse and minimum-phase systems…

- In general we have that if $H(z)$ is stable and causal then poles $|p_k| \;<\; 1\;\forall k$ and ROC: $|z| > \max_k |p_k|$

  $\Longrightarrow$There exists a stable and causal inverse $H_I(z) = 1/H(z)$ if zeros of $H(z)$ are within the unit circle, i.e., $|z_k| \;<\; 1\;\forall k$

- Definition: A system is called minimum-phase if all zeros and poles are inside the unit circle

  $\Longrightarrow$ a stable pole-zero system that is minimum phase has a stable inverse that is also minimum phase

24

24

Summary

Today:
- Some simple filter types and their properties
- Linear phase systems
- Inverse and minimum-phase systems

Next:
- Correlation and energy spectrum density

Inverse and minimum-phase systems…

- Example: Determine inverse system $h[n] = \delta[n] - \frac{1}{3}\delta[n-1]$
- Time-domain solution ($h_I[n]$ causal and stable)

$$h[n] * h_I[n] = \delta[n] \;\Leftrightarrow \sum_{k=-\infty}^{\infty} h[k]h_I[n-k] = \delta[n]$$

$$\sum_{k=0}^{n} h[k]h_I[n-k] = \delta[n]$$

$$n = 0: h[0]h_I[0] = 1 \;\Rightarrow h_I[0] = 1/h[0]$$

$$n = 1: h[0]h_I[1] + h[1]h_I[0] = 0 \Rightarrow h_I[1] = -h[1]h_I[0]/h[0]$$

$$n = 2: h[0]h_I[2] + h[1]h_I[1] + h[2]h_I[0] = 0 \Rightarrow$$

$$h_I[2] = -(h[1]h_I[1] + h[2]h_I[0])/h[0]$$

$$n \geq 1: h_I[n] = -\sum_{k=1}^{n} \frac{h[k]h_I[n-k]}{h[0]}$$

NTNU – Trondheim
Norwegian University of
Science and Technology

TTT4120 Digital Signal Processing
Fall 2021

Lecture: Correlation and Energy Spectral Density

Prof. Stefan Werner
stefan.werner@ntnu.no
Office B329

Department of Electronic Systems

1

Lecture in course book*

- Proakis, Manolakis Digital Signal Processing, 4th Ed.
  - 2.6.1 Crosscorrelation and autocorrelation sequences
  - 2.6.2 Properties of crosscorrelation and autocorrelation…
  - 2.6.4 Input-output correlation sequences
  - 4.2.5 Energy density spectrum of aperiodic signals
  - 5.3.1 Input-output correlation functions and spectra

*Level of detail is defined by lectures and problem sets

2

2

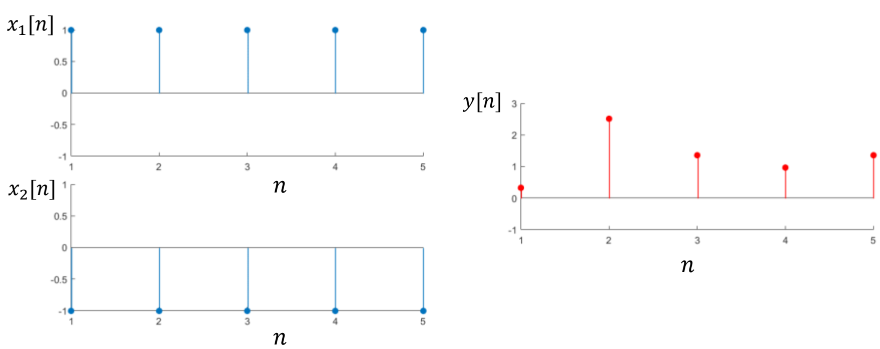
Contents and learning outcomes

- Cross- and autocorrelation sequences
- Properties of cross- and autocorrelation sequences
- Linear time-invariant systems
- Energy spectral density

3

3

Introduction



- Which of the sequences $x_1[n]$ and $x_2[n]$ resembles $y[n]$?
- How to measure similarity between signal sequences

4

4

Introduction…

- Signals transmitted over some medium, e.g., wireless channels, experience delays, echoes, and noise
  - Difficult to recognize/or detect the signals at the receiving end
- Suppose that $x_1[n]$ or $x_2[n]$ is transmitted and $y[n]$ is received
  - If $y[n]$ is more similar to $x_1[n]$ than to $x_2[n]$, we decide that $x_1[n]$ was transmitted
  - If $y[n]$ is more similar to $x_2[n]$ than to $x_1[n]$, we decide that $x_2[n]$ was transmitted

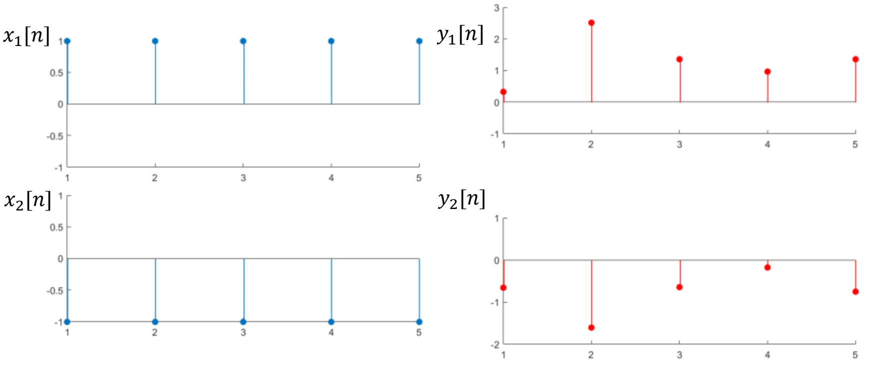
- Correlation is a measure of similarity

5

5

Introduction…

- Digital communication example: $y_i[n] = x_i[n] + w[n]$



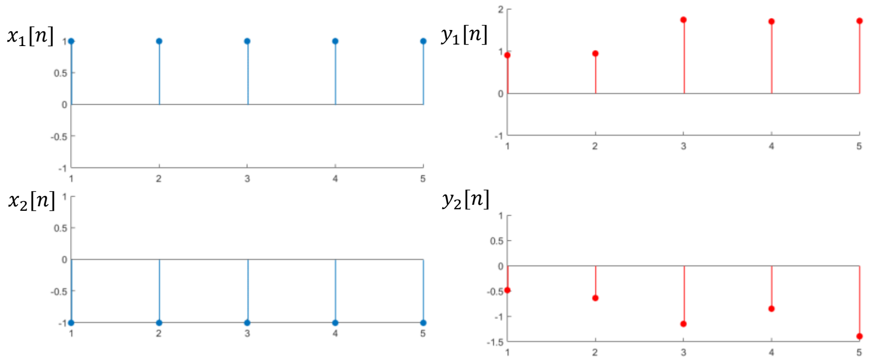
- Noise can make received signal fluctuate significantly

6

6

Introduction…

- Digital communication example: $y_i[n] = x_i[n] + w[n]$

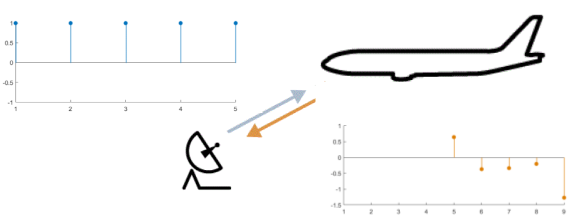


- Noise can make received signal fluctuate significantly

7

Introduction…

- Radar example: $y[n] = \alpha x[n - D] + w[n]$, find D?



- Here we need a similarity measure that gives a maximum for D, considering all possible delays

8

Crosscorrelation and autocorrelation

- Crosscorrelation of real-valued sequences $x[n]$ and $y[n]$

$$r_{xy}[l] = \sum_{n=-\infty}^{\infty} x[n]y[n-l]$$
$$= \sum_{n=-\infty}^{\infty} x[n+l]y[n]\,, l = \pm 1, \pm 2, \ldots$$

- Measure of similarity between signals $x[n]$ and $y[n]$
- Reverse role $r_{yx}[l] \neq r_{xy}[l]$

$$r_{yx}[l] = \sum_{n=-\infty}^{\infty} y[n]x[n-l] = r_{xy}[-l]$$

9

9

Crosscorrelation and autocorrelation…

- Similarity to convolution of $x[n]$ and $y[n]$

$$x[l] * y[l] = \sum_{k=-\infty}^{\infty} x[k]y[l-k]$$

$$r_{xy}[l] = \sum_{k=-\infty}^{\infty} x[k]y[k-l] = x[l] * y[-l]$$

- Relation can be exploited for efficient computation
- Autocorrelation sequence (self-similarity), $y[n] = x[n]$

$$r_{xx}[l] = \sum_{n=-\infty}^{\infty} x[n]x[n-l]$$
$$= \sum_{n=-\infty}^{\infty} x[n+l]x[n]$$

10

10

Crosscorrelation and autocorrelation…

- Finite causal sequences $x[n] = y[n] = 0, n < 0, n \geq N$

$$x[n] = \{\underline{x[0]}, x[1], x[2], \dots x[N-1]\}$$

$$y[n] = \{\underline{y[0]}, y[1], y[2], \dots y[N-1]\}$$

$$r_{xy}[l] = \sum_{n=-\infty}^{\infty} x[n]y[n-l]$$

- Try a couple of values of l and search for pattern ⇒

$$r_{xy}[l] = \sum_{n=l}^{N-1} x[n]y[n-l], l \geq 0$$

$$r_{xy}[l] = \sum_{n=0}^{N-|l|-1} x[n]y[n-l], l < 0$$

11

11

Properties of autocorrelation

- Energy of sequences $x[n]$

$$E_x = \sum_{n=-\infty}^{\infty} x^2[n] = r_x[0] \geq 0$$

- Autocorrelation is maximum at lag l=0

$$|r_{xx}[l]| \leq r_{xx}[0] = E_x$$

- Autocorrelation is even ⇒ only compute values for $l \geq 0$

$$r_{xy}[l] = r_{xy}[-l] \Rightarrow r_{xx}[l] = r_{xx}[-l]$$

12

12

Properties of autocorrelation…

- Normalized versions

$$\varrho_{xx}[l] = \frac{r_{xx}[l]}{r_{xx}[0]} \Rightarrow |\varrho_{xx}[l]| \le 1$$

$$\varrho_{xy}[l] = \frac{r_{xy}[l]}{\sqrt{r_{xx}[0]r_{yy}[0]}} \Rightarrow \left|\varrho_{xy}[l]\right| \le 1$$

13

13

Properties cross- and autocorrelation…

- Example: Compute the autocorrelation of $x[n] = \alpha^n u[n]$

- Solution:

$$r_{xx}[l] = \sum_{n=-\infty}^{\infty} x[n+l]x[n]$$

$$= \sum_{n=-\infty}^{\infty} \alpha^{n+l} u[n+l]\alpha^n u[n]$$

$$= \alpha^l \sum_{n=0}^{\infty} \alpha^{2n} = \frac{\alpha^l}{1-\alpha^2},\ l \ge 0$$

Since $r_{xx}[-l] = r_{xx}[l]$, we get the final expression

$$r_{xx}[l] = \frac{\alpha^{|l|}}{1-\alpha^2}, \forall l$$

14

14

Example 1

- Let $y[n] = Ax[n-D]$. Show that $D = \arg\max_{l} |r_{yx}[l]|$

- Solution:

$$r_{yx}[l] = \sum_{n=-\infty}^{\infty} y[n]x[n-l] = \sum_{n=-\infty}^{\infty} y[n+l]x[n]$$
$$= \sum_{n=-\infty}^{\infty} Ax[n+l-D]x[n] = Ar_{xx}[D-l]$$

From properties of autocorrelation sequences, we know

$$\left|r_{xy}[l]\right| = |A||r_{xx}[D-l]| \leq |A||r_{xx}[0]|, \forall l$$

$\therefore \left|r_{xy}[l]\right|$ reach its maximum for $l = D$

15

15

Example 2

- Let $y[n] = x[n] + Rx[n-D]$, i.e., received signal contains an echo. How to estimate R, D using the autocorrelation of $y[n]$?

16

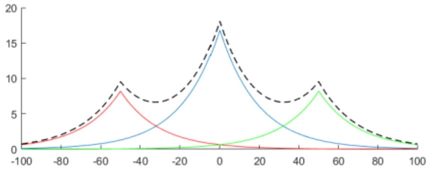
16

Example 3

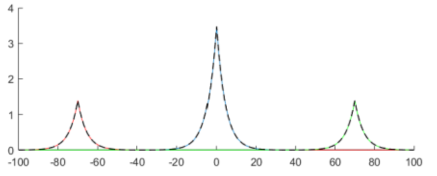
- Let $x[n] = \alpha^n u[n]$ and $y[n] = x[n] + Rx[n - D]$

$$\Longrightarrow r_{yy}[l] = (1 + R^2) r_{xx}[l] + R r_{xx}[l + D] + R r_{xx}[l - D]$$

$\alpha = 0.95, R = 0.8, D = 50$



$\alpha = 0.8, R = 0.5, D = 70$



- Shape of $r_{yy}[l]$ depends on R, D

17

17

Example 3...

- Details on how to obtain $r_{yy}[l]$ and R in previous slide
- From Slide 14: $r_{xx}[l] = \frac{\alpha^{|l|}}{1-\alpha^2}$
- $r_{yy}[l] = \sum_{n=-\infty}^{\infty} y[n + l] y[n]$

$$= \sum_{n=-\infty}^{\infty} (x[n + l] + Rx[n + l - D])(x[n] + Rx[n - D])$$

$$= \sum_{n=-\infty}^{\infty} x[n + l] x[n] + R \sum_{n=-\infty}^{\infty} x[n + l - D] x[n]$$

$$+ R \sum_{n=-\infty}^{\infty} x[n + l] x[n - D] + R^2 \sum_{n=-\infty}^{\infty} x[n - D] x[n + l - D]$$

$$= r_{xx}[l] + R r_{xx}[l - D] + R r_{xx}[l + D] + R^2 r_{xx}[l]$$

- Look at the following values (corresponding to the peaks in figure)

$$r_{yy}[0] = (1 + R^2) r_{xx}[0] + R r_{xx}[-D] + R r_{xx}[D] \approx (1 + R^2) r_{xx}[0]$$

$$r_{yy}[D] = (1 + R^2) r_{xx}[D] + R r_{xx}[0] + R r_{xx}[2D] \approx R r_{xx}[0]$$

- Given values $r_{yy}[0]$ and $r_{yy}[D]$, we can solve for R

18

18

Example 2…

Matlab

```
l = (-100:100);
a=0.8;    % decay rate
R = 0.5;  % echo strength
D = 70;   % delay of echo

rxx_l   = a.^abs(l)/(1-a^2);
rxx_lpD = a.^abs(l+D)/(1-a^2);
rxx_lmD = a.^abs(l-D)/(1-a^2);
ryy     = (1+R^2)*rxx_l+R*rxx_lpD+R*rxx_lmD;

figure
plot(l,(1+R^2)*rxx_l); hold on
plot(l,R*rxx_lpD,'r');
plot(l,R*rxx_lmD,'g')
plot(l,ryy,'k--','LineWidth',1)
```

19

19

Energy spectral density

- Quantity $S_{xx}(\omega) \geq 0$ is the energy density spectrum of $x[n]$

$$r_{xx}[l] = x[l] * x[-l] \overset{\mathcal{F}}{\leftrightarrow} S_{xx}(\omega) = X(\omega)X^*(\omega) = |X(\omega)|^2$$

- Energy of complex-valued sequence $x[n]$

$$E_x = r_{xx}[0] = \sum_{n=-\infty}^{\infty} |x[n]|^2$$
$$= \frac{1}{2\pi}\int_{-\pi}^{\pi} S_{xx}(\omega)d\omega = \frac{1}{2\pi}\int_{-\pi}^{\pi} |X(\omega)|^2 d\omega$$

- Quantity $S_{xy}(\omega)$ is the cross-energy density spectrum

$$r_{xy}[l] = x[l] * y[-l] \overset{\mathcal{F}}{\leftrightarrow} S_{xy}(\omega) = X(\omega)Y^*(\omega)$$

20

20

Input-output correlations

$x[n] \longrightarrow \boxed{\text{LTI}} \longrightarrow y[n] = h[n] * x[n]$

$r_{xx}[l]$ $\quad h[n] \quad$ $r_{yy}[l]?\ r_{yx}[l]?$

- Input-output correlations

$$r_{yx}[l] = h[l] * r_{xx}[l]$$
$$r_{yy}[l] = r_{hh}[l] * r_{xx}[l]$$
$$E_y = r_{yy}[0] = \sum_{k=-\infty}^{\infty} r_{hh}[k] r_{xx}[k]$$

- Crosscorrelation between $x[n]$ and $y[n]$ can be seen as the output signal of an LTI system when input signal is $r_{xx}[n]$

21

21

Input-output correlations and energy spectrum

$x[n] \longrightarrow \boxed{\text{LTI}} \longrightarrow y[n] = h[n] * x[n]$

$r_{xx}[l]$ $\quad h[n] \quad$ $r_{yx}[l] = h[l] * r_{xx}[l]$

$r_{yy}[l] = r_{hh}[l] * r_{xx}[l]$

- In z-transform domain

$$h[l] * h[-l] \overset{z}{\leftrightarrow} H(z)H(z^{-1})$$
$$r_{yx}[l] = h[l] * r_{xx}[l] \overset{z}{\leftrightarrow} H(z)S_{xx}(z)$$
$$r_{yy}[l] = r_{hh}[l] * r_{xx}[l] \overset{z}{\leftrightarrow} H(z)H(z^{-1})S_{xx}(z)$$

- Output- and cross-energy density spectra

$$S_{yy}(\omega) = |H(\omega)|^2 S_{xx}(\omega) = |H(\omega)|^2 |X(\omega)|^2$$
$$S_{yx}(\omega) = H(\omega)S_{xx}(\omega)$$

22

22

Input-output correlations and energy …

- We have the following relation Fourier transform pair

$$r_{yy}[l] = \frac{1}{2\pi}\int_{-\pi}^{\pi} S_{yy}(\omega)e^{j\omega l}d\omega$$

- Energy of output sequence (of an LTI system)

$$r_{yy}[0] = \frac{1}{2\pi}\int_{-\pi}^{\pi} S_{yy}(\omega)d\omega = \frac{1}{2\pi}\int_{-\pi}^{\pi} |H(\omega)|^2 S_{xx}(\omega)d\omega$$

- Determine impulse response by signal with flat spectrum

$$h[n] = \frac{1}{S_{xx}} r_{yx}[n]$$

23

23

Summary

Today:

- Crosscorrelation and autocorrelation sequences
- Linear time invariant systems
- Energy spectrum

Next:

- Inverse z-transform

24

24

Example 2 (modified)

- Let $y[n] = x[n] + Rx\,[n - D]$ be an audio signal corrupted by an echo. We would like to estimate R, D using the autocorrelation of $y[n]$, and design a filter to remove the echo.

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

$$X(z) \qquad\qquad Y(z) = H(z)X(z)$$

```
Matlab files on BB:
      Correlation.m
```

25

Example 2 (modified)…

- Model the problem using an LTI system

$$x[n] \longrightarrow \boxed{h[n]} \longrightarrow y[n] = x[n] + Rx[n-D]$$

$$X(z) \qquad\qquad Y(z) = H(z)X(z) = (1 + Rz^{-D})X(z)$$

- Estimate R and D using autocorrelation sequence $r_{yy}[n]$
- Find the inverse system $H_I(z)$ such that (see previous lecture)

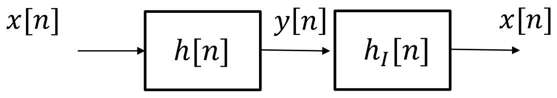
$$h[n] * h_I[n] = \delta[n] \overset{z}{\leftrightarrow} H(z)H_I(z) = 1$$

$$\Rightarrow H_I(z) = \frac{1}{1 + Rz^{-D}}$$

26

Example 2 (modified)…

- If $R < 1$, $H(z)$ is minimum phase and so is $H_I(z)$
- We can find a causal and stable filter $h_I[n]$



- We get D from inspecting the peaks of $r_{yy}[l]$
- We obtain an estimate R from the relation

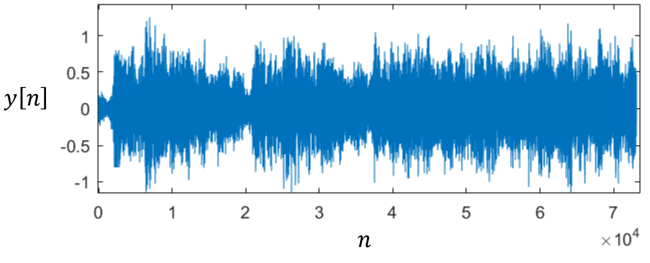
$$\frac{r_{yy}[0]}{r_{yy}[D]} = \frac{1+R^2}{R}$$

27

27

Example 2 (modified)…

- Plot of received signal $y[n]$ ($R = 0.98$, $D = 4196$):

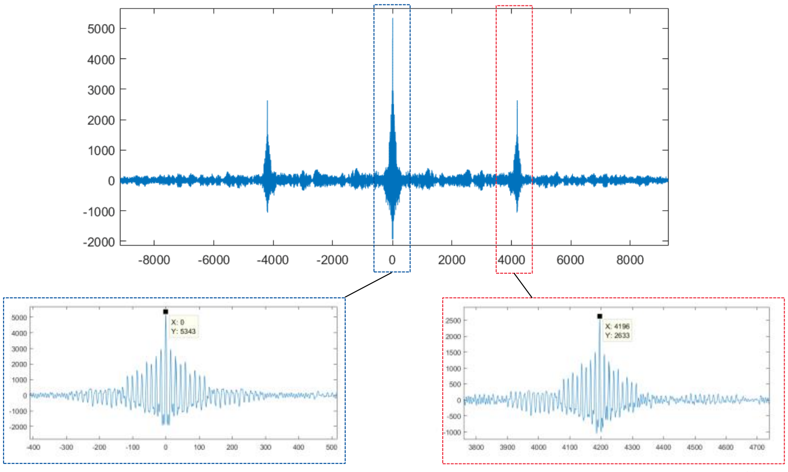


28

28

Example 2 (modified)…

- Autocorrelation of $y[n]$:



Example 2 (modified)…

- From figure we get:

$$D = 4196, \quad \frac{r_{yy}[0]}{r_{yy}[D]} = \frac{1+R^2}{R} = \frac{5343}{2633} \Rightarrow R = 0.8430$$

- Delay is correct but parameter estimate of R is not exact. Listen to the equalized signal and judge whether the echo is removed (or suppressed)

NTNU – Trondheim
Norwegian University of Science and Technology

TTT4120 Digital Signal Processing
Fall 2019

Lecture: Inverse Z-transform and Residues

Prof. Stefan Werner
stefan.werner@ntnu.no
Office B329

Department of Electronic Systems

Lecture in course book*

- Proakis, Manolakis Digital Signal Processing, 4th Ed.
  - 3.4.3 The inverse z-transform by partial-fraction expansion

*Level of detail is defined by lectures and problem sets

Contents and learning outcomes

- Inverse z-transform using partial fraction expansion
  - Mainly repetition of already covered or known topics
- Matlab implementation

Inverse z-transform

$$X(z) \overset{?}{\rightarrow} x[n]$$

- Three popular methods
  - Contour integration: $x[n] = \frac{1}{2\pi j} \oint_C X(z) z^{n-1} dz$
  - Power series expansion: $X(z) = \sum_{k=-\infty}^{\infty} c_k z^{-k}$
  - Partial fraction expansion and table lookup (rational functions):

$$X(z) = \sum_{k=1}^{N} \left(\frac{R_{k,1}}{(1-p_k z^{-1})} + \frac{R_{k,2}}{(1-p_k z^{-1})^2} + \cdots + \frac{R_{k,r}}{(1-p_k z^{-1})^r} \right)$$

Z-transform table

Sequence	Transform	ROC
$\delta[n]$	1	$\forall z$
$a^n u[n]$	$\frac{1}{1-az^{-1}}$	$\lvert z\rvert > \lvert a\rvert$
$-b^n u[-n-1]$	$\frac{1}{1-bz^{-1}}$	$\lvert z\rvert < \lvert b\rvert$
$(a^n \sin \omega_0 n)u[n]$	$\frac{(a\sin \omega_0)z^{-1}}{1-(2a\cos \omega_0)z^{-1}+a^2z^{-2}}$	$\lvert z\rvert > \lvert a\rvert$
$(a^n \cos \omega_0 n)u[n]$	$\frac{1-(a\cos \omega_0)z^{-1}}{1-(2a\cos \omega_0)z^{-1}+a^2z^{-2}}$	$\lvert z\rvert > \lvert a\rvert$
$na^n u[n]$	$\frac{az^{-1}}{(1-az^{-1})^2}$	$\lvert z\rvert > \lvert a\rvert$
$-nb^n u[-n-1]$	$\frac{bz^{-1}}{(1-bz^{-1})^2}$	$\lvert z\rvert < \lvert b\rvert$

5

Inverse z-transform by partial fractions

- Consider the rational expression

$$X(z) = \frac{b_0+b_1z^{-1}+\cdots+b_Mz^{-M}}{1+a_1z^{-1}+\cdots+a_Nz^{-N}} = \frac{B(z)}{A(z)}$$

- $B(z)$ and $A(z)$ are polynomials in variable z
- b_k and a_k are the coefficients of $B(z)$ and $A(z)$, respectively
- M is the degree of $B(z)$ and N is the degree of $A(z)$
- M roots of polynomial $B(z)$, satisfy $B(z_k) = 0$: called zeros of $H(z)$
- N roots of polynomial $A(z)$, satisfy $A(p_k) = 0$: called poles of $H(z)$

6

Inverse z-transform by partial fractions…

- Fundamental theorem of algebra:

  A polynomial of degree M' has exactly M' roots,
  counting multiplicities

- We may factor $X(z)$ as follows

$$X(z) = \frac{b_0+b_1z^{-1}+\cdots+b_Mz^{-M}}{1+a_1z^{-1}+\cdots+a_Nz^{-N}} = b_0z^{N-M}\frac{z^M+\tilde{b}_1z^{M-1}+\cdots+\tilde{b}_M}{z^N+a_1z^{N-1}+\cdots+a_N}$$

$$= b_0z^{N-M}\frac{\prod_{k=1}^{M}(z-z_k)}{\prod_{k=1}^{N}(z-p_k)} = b_0\frac{\prod_{k=1}^{M}(1-z_kz^{-1})}{\prod_{k=1}^{N}(1-p_kz^{-1})}$$

- When coefficients a_k and b_k are real, complex poles or zeros occur in complex conjugate pairs

7

Inverse z-transform by partial fractions…

- Let us assume that
  - $M < N$, i.e., $H(z)$ is proper
  - Poles are distinct, i.e., all roots of $A(z)$ have multiplicity one
- Then we can perform a partial fraction of $X(z)$ to obtain

$$X(z) = \frac{R_1}{(1-p_1z^{-1})} + \frac{R_2}{(1-p_2z^{-1})} + \cdots + \frac{R_N}{(1-p_Nz^{-1})}$$

  where p_k is the kth pole of $X(z)$ and R_k is the residue at p_k
- When $p_k = p_l^*$ we have $R_k = R_l^*$

8

Inverse z-transform by partial fractions…

- Example: $X(z) = \frac{1}{1-\frac{1}{4}z^{-2}} = \frac{1/2}{1-\frac{1}{2}z^{-1}} + \frac{1/2}{1+\frac{1}{2}z^{-1}}$

- Let us verify:

$$\frac{1/2}{1-\frac{1}{2}z^{-1}} + \frac{1/2}{1+\frac{1}{2}z^{-1}} = \frac{\frac{1}{2}\left(1+\frac{1}{2}z^{-1}\right)+\frac{1}{2}\left(1-\frac{1}{2}z^{-1}\right)}{\left(1-\frac{1}{2}z^{-1}\right)\left(1+\frac{1}{2}z^{-1}\right)}$$

Inverse z-transform by partial fractions…

- Once in partial fraction form, inverse z-transform becomes simple:

$$x[n] = \mathcal{Z}^{-1}\{X(z)\}$$
$$= \mathcal{Z}^{-1}\left\{\frac{R_1}{(1-p_1z^{-1})} + \frac{R_2}{(1-p_2z^{-1})} + \cdots + \frac{R_N}{(1-p_Nz^{-1})}\right\}$$

- Finally to complete $x[n]$, we use the relation

$$\mathcal{Z}^{-1}\left\{\frac{1}{(1-p_kz^{-1})}\right\} = \begin{cases} p_k^n u[n], & \text{ROC: } |z| > |p_k| \\ -p_k^n u[-n-1], & \text{ROC: } |z| < |p_k| \end{cases}$$

- Depending on the ROCs, we may end up with causal, anti-causal and non-causal (stable or unstable) time-domain sequence $x[n]$

Inverse z-transform by partial fractions…

- Example: Causal system and stable system ($|z| > \max_k |p_k| < 1$)

$$x[n] = \sum_{k=1}^{N} R_k \mathcal{Z}^{-1}\left\{\frac{1}{(1-p_k z^{-1})}\right\} = \sum_{k=1}^{N} R_k p_k^n u[n]$$

- Example: Complex conjugated poles $R_1 = R_2^*$, with $|p_1| < 1$

$$\begin{aligned} x[n] &= R_1 p_1^n u[n] + R_1^* (p_1^*)^n u[n] \\ &= (R_1 p_1^n + R_1^* (p_1^*)^n) u[n] \\ &= |R_1||p_1|^n \left(e^{j(\angle R_1 + \angle p_1 n)} + e^{-j(\angle R_1 + \angle p_1 n)}\right) u[n] \\ &= 2|R_1||p_1|^n \cos(\angle R_1 + \angle p_1 n)\, u[n] \end{aligned}$$

11

Inverse z-transform by partial fractions…

$$X(z) \stackrel{?}{=} \frac{R_1}{(1-p_1 z^{-1})} + \frac{R_2}{(1-p_2 z^{-1})} + \cdots + \frac{R_N}{(1-p_N z^{-1})}$$

- Finding the partial fraction expansion:
  1. Factor $A(z)$, i.e., find poles $p_1, \ldots, p_N$
  2. Find residues $R_1, \ldots, R_N$

12

Finding residues

- Method 1: Solve linear equations (always works but can be tedious)
  1. Clear denominator terms

$$B(z) = \underbrace{\prod_{k=1}^{N}(1 - p_k z^{-1})}_{A(z)} X(z) = \sum_{k=1}^{N} R_k \prod_{j=1, j \neq k}^{N} (1 - p_j z^{-1})$$

  2. Equate coefficients on both sides
- Example: $\frac{1}{\left(1-\frac{1}{2}z^{-1}\right)\left(1+\frac{1}{2}z^{-1}\right)} = \frac{R_1}{1-\frac{1}{2}z^{-1}} + \frac{R_2}{1+\frac{1}{2}z^{-1}}$

  1. $1 = R_1\left(1 + \frac{1}{2}z^{-1}\right) + R_2\left(1 - \frac{1}{2}z^{-1}\right)$
  2. z^0: $1 = R_1 + R_2$

     z^{-1}: $0 = \frac{R_1}{2} - \frac{R_2}{2}$

Finding residues…

- Method 2: Multiply both sides by $1 - p_k z^{-1}$ to get
  1. $(1 - p_k z^{-1})X(z) = \frac{R_1(1-p_k z^{-1})}{(1-p_1 z^{-1})} + \cdots + R_k + \cdots + \frac{R_N(1-p_k z^{-1})}{(1-p_N z^{-1})}$
  2. and set $z = p_k$
- In general, we have the formula

$$R_k = (1 - p_k z^{-1})X(z)|_{z=p_k}$$

Finding residues…

- Example: $X(z) = \frac{1}{\left(1-\frac{1}{2}z^{-1}\right)\left(1+\frac{1}{2}z^{-1}\right)} = \frac{R_1}{1-\frac{1}{2}z^{-1}} + \frac{R_2}{1+\frac{1}{2}z^{-1}}$

$$R_1 = \left(R_1 + \frac{R_2\left(1-\frac{1}{2}z^{-1}\right)}{1+\frac{1}{2}z^{-1}}\right)\Bigg|_{z=\frac{1}{2}} = \frac{1}{\left(1+\frac{1}{2}z^{-1}\right)}\Bigg|_{z=\frac{1}{2}} = \frac{1}{2}$$

$$R_2 = \left(\frac{R_2\left(1+\frac{1}{2}z^{-1}\right)}{1-\frac{1}{2}z^{-1}} + R_2\right)\Bigg|_{z=-\frac{1}{2}} = \frac{1}{\left(1-\frac{1}{2}z^{-1}\right)}\Bigg|_{z=-\frac{1}{2}} = \frac{1}{2}$$

Example: Inverse z-transform

- Find impulse response of system $H(z) = \frac{3-4z^{-1}}{1-3.5z^{-1}+1.5z^{-2}}$
- Solution:

$$H(z) = \frac{3-4z^{-1}}{1-3.5z^{-1}+1.5z^{-2}} = \frac{3-4z^{-1}}{(1-0.5z^{-1})(1-3z^{-1})} = \frac{R_1}{1-0.5z^{-1}} + \frac{R_2}{1-3z^{-1}}$$

$$R_1 = (1-0.5z^{-1})H(z)|_{z=0.5} = \frac{3-4z^{-1}}{1-3z^{-1}}\Bigg|_{z=0.5} = \frac{3-8}{1-6} = 1$$

$$R_2 = (1-3z^{-1})H(z)|_{z=3} = \frac{3-4z^{-1}}{1-0.5z^{-1}}\Bigg|_{z=3} = \frac{3-\frac{4}{3}}{1-\frac{0.5}{3}} = 2$$

$$\Rightarrow h[n] = \mathcal{Z}^{-1}\{H(z)\} = \mathcal{Z}^{-1}\left\{\frac{1}{1-0.5z^{-1}}\right\} + \mathcal{Z}^{-1}\left\{\frac{2}{1-3z^{-1}}\right\}$$

Example: Inverse z-transform…

$$h[n] = \mathcal{Z}^{-1}\{H(z)\} = \mathcal{Z}^{-1}\left\{\frac{1}{1-0.5z^{-1}}\right\} + \mathcal{Z}^{-1}\left\{\frac{2}{1-3z^{-1}}\right\}$$

- We may completely determine $h[n]$ after specifying ROC:
  - $h_1[n] = (0.5^n + 2 \cdot 3^n)u[n]$ with ROC $|z| > 3$
  - $h_2[n] = -(0.5^n + 2 \cdot 3^n)u[-n-1]$ with ROC $|z| < 0.5$
  - $h_3[n] = 0.5^n u[n] - 2 \cdot 3^n u[-n-1]$ with ROC $0.5 < |z| < 3$
- Summary:
  - $h_1[n]$ is causal and unstable
  - $h_2[n]$ is anti-causal and unstable
  - $h_3[n]$ is non-causal and stable

17

Final comments (optional)

- We assumed that $N > M$ so that $X(z)$ was proper
  - If $M \geq N$ can just express $X(z)$ as

$$X(z) = \frac{b_0+b_1z^{-1}+\cdots+b_Mz^{-M}}{1+a_1z^{-1}+\cdots+a_Nz^{-N}}$$
$$= \underbrace{\frac{\tilde{b}_0+\tilde{b}_1z^{-1}+\cdots+\tilde{b}_{N-1}z^{-(N-1)}}{1+a_1z^{-1}+\cdots+a_Nz^{-N}}}_{\text{Proper rational part}} + \underbrace{\sum_{k=0}^{M-N} C_k z^{-k}}_{\text{Polynomial part}}$$

  - Proper rational part handled as before while polynomial part trivial
- If a pole p_k has multiplicity r, the expansion has a more general form

$$\frac{R_{k,1}}{1-p_kz^{-1}} + \frac{R_{k,2}}{(1-p_kz^{-1})^2} + \cdots + \frac{R_{k,r}}{(1-p_kz^{-1})^r}$$

- Method 2 only provides $R_{k,r}$. Remaining residues using Method 1.

18

Matlab implementation

- Find impulse partial fraction representation of $H(z) = \frac{3-4z^{-1}}{1-3.5z^{-1}+1.5z^{-2}}$

```
Matlab
B = [3 -4];
A = [1 -3.5 1.5];

[R,P,C] = residuez(B,A);

R % Residues
P % Poles
C % Direct terms (if improper)
```

- Other useful Matlab functions:
  - `roots(a), poly([p1,p2]),impz(B,A)`

Summary

Today:

- Inverse z-transform
- Calculation of residues

Next:

- Sampling theorem

Illustrating example (optional)

A causal LTI system is described by the following difference equation:

$$y[n] = 0.81y[n-2] + x[n] - x[n-2]$$

- Determine:
  a) The system function $H(z)$
  b) The unit impulse response $h[n]$
  c) The frequency response function $H(\omega)$, and plot its magnitude and phase over $0 \leq \omega \leq \pi$

Illustrating example (optional)…

- The system function $H(z)$

$$Y(z) = 0.81z^{-2}Y(z) + X(z) - z^{-2}X(z)$$

$$H(z) = \frac{Y(z)}{X(z)} = \frac{1 - z^{-2}}{1 - 0.81z^{-2}} = \frac{1 - z^{-2}}{(1 - 0.9z^{-1})(1 + 0.9z^{-1})}$$

- Causality implies ROC: $|z| > 0.9$

Illustrating example (optional)…

- The unit impulse response $h[n]$

$$h[n] = \mathcal{Z}^{-1}\{H(z)\} = \mathcal{Z}^{-1}\left\{\frac{1-z^{-2}}{1-0.81z^{-2}}\right\}$$

$$= \mathcal{Z}^{-1}\left\{1 - 0.19z^{-2}\frac{1}{1-0.81z^{-2}}\right\} = \delta[n] - 0.19 \cdot h'[n-2]$$

where

$$h'[n] = \mathcal{Z}^{-1}\left\{\frac{1}{(1-0.9z^{-1})(1+0.9z^{-1})}\right\} = \mathcal{Z}^{-1}\left\{\frac{R_1}{1-0.9z^{-1}} + \frac{R_2}{1+0.9z^{-1}}\right\}$$

- $R_1 = R_2 = \frac{1}{2} \Longrightarrow$

$$h[n] = \delta[n] - \frac{1}{2}0.19 \cdot 0.9^{n-2} \cdot (1 + (-1)^{n-2})u[n-2]$$

$$= \delta[n] - 0.1173 \cdot 0.9^{n} \cdot (1 + (-1)^{n})u[n-2]$$

Illustrating example (optional)…

- Plot the frequency response: $H(\omega) = \frac{1-e^{-j2\omega}}{1-0.81e^{-j2\omega}}$

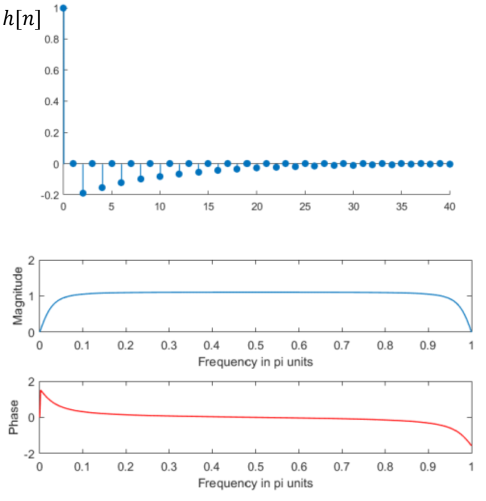
Matlab
```
B = [1 0 -1];
A = [1 0 -0.81];
W = [0:1:500]*pi/500;
H = freqz(B,A,W);

magH = abs(H); phaH = angle(H);

subplot(2,1,1); plot(W/pi,magH);
xlabel('Frequency in pi units')
ylabel('Magnitude')

subplot(2,1,2); plot(W/pi,phaH);
xlabel('Frequency in pi units')
ylabel('Phase')
```

Illustrating example (optional)…



25

NTNU – Trondheim
Norwegian University of
Science and Technology

TTT4120 Digital Signal Processing
Fall 2019

Lecture: The Sampling Theorem

Prof. Stefan Werner
stefan.werner@ntnu.no
Office B329

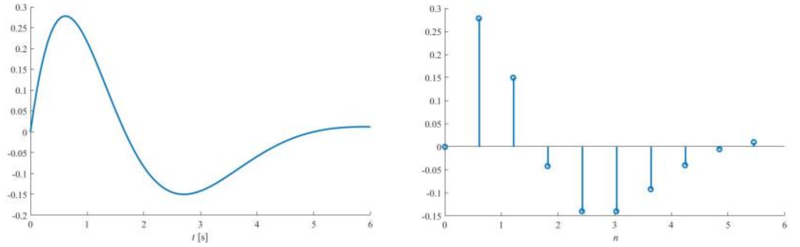
Department of Electronic Systems

Lecture in course book*

- Proakis, Manolakis Digital Signal Processing, 4th Ed.
  - 1.4.2 The sampling theorem
  - 1.4.6 Digital to analog conversion
  - 6.1 Ideal sampling and reconstruction of continuous-time signals

*Level of detail is defined by lectures and problem sets

Preliminary questions

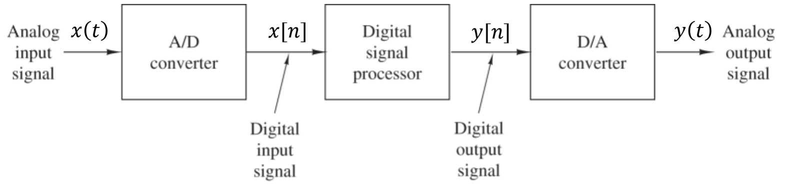


- How fast must we sample the continuous-time signal (left) without losing information?
- What continuous-time signal corresponds to the discrete-time signal (right)?

Contents and learning outcomes

- Sampling of sinusoids and aliasing (partially covered Lect.1)
- Sampling theorem:
  – Ideal reconstruction of continuous-time signals
- Wagon wheel effect

Periodic sampling



- Sampling – Processing – Reconstruction
- A signal is read (sampled) at a regular interval

$$x[n] = x(t_n) = x(nT) = x\left(\frac{n}{F_s}\right)$$

- Sampling interval $T = \frac{1}{F_s}$, F_s being the sampling frequency

5

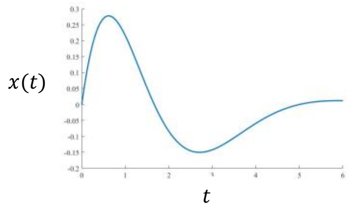
Periodic sampling…

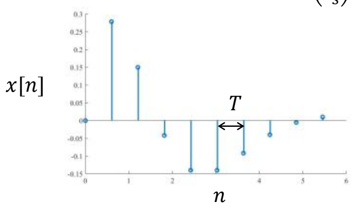
- Examples of sampling rate standards:
  - CD audio: $F_s = 44.1$ kHz
  - TV frame rate: $F_s = 100, 200, 400$ fr/s

6

Periodic sampling…

$$x[n] = x(t_n) = x(nT) = x\left(\frac{n}{F_s}\right)$$





- Under what conditions is $x[n]$ a good representation of $x(t)$?
  - Appropriate choice of T or F_s
- Under what conditions can $x(t)$ be recovered from $x[n]$?
  - Interpolation formula is needed
- Conditions are provided by the sampling theorem

7

Sampling of sinusoids and aliasing

- Why considering sinusoidal signals?
- Many practical signals can be represented by the Fourier transform (or Fourier series)

$$x(t) = \int_{-\infty}^{\infty} X(F)e^{j2\pi Ft}dF$$
$$= \int_{-\infty}^{\infty} X(F)(\cos 2\pi Ft + j\sin 2\pi Ft)dF$$

- The concepts of sampling a single sinusoidal signal carry over to the case of more complicated signals.

8

Sampling of sinusoids and aliasing…

- Consider the continuous-time signal

$$x(t) = \cos \Omega t = \cos 2\pi F t$$

with angular frequency Ω [rad/s], or frequency F [Hz]
- Periodic sampling at regular time intervals $t_n = nT = 1/F_s$

$$x[n] \equiv x(t_n) = \cos 2\pi F n T = \cos 2\pi \frac{F}{F_s} n = \cos \underbrace{2\pi f}_{\omega} n$$

- Spectrum of digital signal is periodic with period $\omega = 2\pi$ (or $f = 1$), where $f = 1/2$ represents the highest frequency

$$\therefore\ f = \frac{F}{F_s} \leq \frac{1}{2}$$

Sampling of sinusoids and aliasing…

- Example 1: Sample signal $x(t) = \cos 2\pi 400 t$ at F_s=1000

$$\begin{aligned} x[n] &= \cos 2\pi 400 nT \\ &= \cos 2\pi \tfrac{400}{1000} n = \cos 2\pi (0.4 + k)\, n \end{aligned}$$

- Spectrum of sampled signal $X(f)$ obtained directly from

$$x[n] = \tfrac{1}{2}\left(e^{j2\pi(0.4+k)n} + e^{-j2\pi(0.4+k)n}\right)$$

Sampling of sinusoids and aliasing…

- Example 2: Sample signal $x(t) = \cos 2\pi 400t + \cos 2\pi 800t$ at F_s=1000

$$x[n] = \cos 2\pi 400nT + \cos 2\pi 800nT$$
$$= \cos 2\pi(0.4 + k)n + \cos 2\pi(\underbrace{0.8}_{1-0.2} + k)\, n$$
$$= \cos 2\pi(0.4 + k)n + \cos 2\pi(-0.2 + k)\, n$$

- Spectrum of sampled signal $X(f)$ obtained directly from:

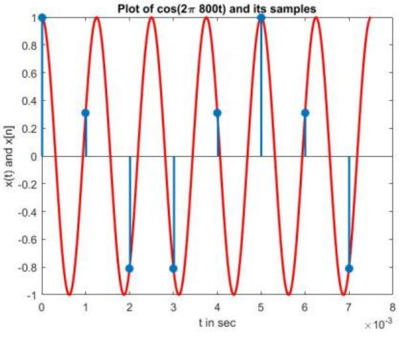
$$x[n] = \frac{1}{2}(e^{j2\pi 0.4n} + e^{-j2\pi 0.4n} + e^{j2\pi 0.2n} + e^{-j2\pi 0.2n})$$

- Distortion: highest analog frequency (800 Hz) appears as low-frequency component in digital spectrum (200 Hz)

11

Sampling of sinusoids and aliasing…

- Example 2 (cont.): Simultaneous plot of $\cos 2\pi 800t$ and its samples when $F_s = 1000$



- Samples appear to be from $\cos 2\pi 200t$

12

Sampling of sinusoids and aliasing…

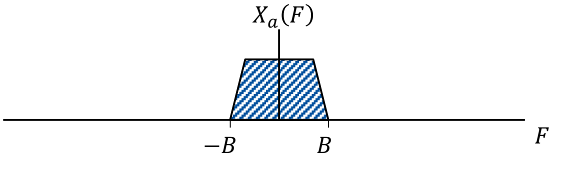
- To avoid aliasing (folding of high frequency components around $f = 1/2$, the following condition must be satisfied (Lecture 1)

$$\frac{F}{F_s} \leq \frac{1}{2}, \forall F \Rightarrow F_s \geq 2F_{\max}$$

- We shall see that any bandlimited continuous-time signal can be reconstructed if sampled above the Nyquist rate $2F_{\max}$

Ideal reconstruction of continuous-time signals

- Bandlimited signal:



A signal is bandlimited if there exists a finite frequency B (or Ω_B) such that $X_a(F)$ (or $X_a(\Omega)$) is zero for $F > B$ (or $\Omega > \Omega_B$). The frequency $B = \Omega_B/2\pi$ is called the signal bandwidth in Hz.

Ideal reconstruction of continuous-time …

- Sampling Theorem:

A bandlimited analog signal $x_a(t)$ can be reconstructed from its sample values $x[n] = x_a(nT)$ if the signal is sampled at rate

$$F_s = \frac{1}{T} \geq 2F_{\max} = 2B,$$

where $F_{max} = B$ is the highest frequency contained in $x_a(t)$. Otherwise aliasing would result in $x[n]$.

- Sampling rate $F_{\mathrm{N}} = 2F_{\max}$ is called the Nyquist rate
- Highest analog frequency represented in $x[n]$ is $\frac{F_s}{2}$

Ideal reconstruction of continuous-time …

- Example: What is the Nyquist rate for the following signals?

$$x_1(t) = \cos 2\pi 400t + \cos 2\pi 800t$$

$$x_2(t) = \cos 100\pi t + 3\cos 200\pi t$$

$$x_3(t) = \cos 150\pi t + 10\sin(600\pi t + \theta)$$

- Can the signals be sampled at rate F_{N} without problems?

Ideal reconstruction of continuous-time …

- Continuous-time signal: $x_a(t) = \int_{-\infty}^{\infty} X_a(F) e^{j2\pi Ft} dF$

  $X_a(F) = \int_{-\infty}^{\infty} x_a(t) e^{-j2\pi Ft} dt$
- Discrete-time signal: $x[n] = \int_{-\frac{1}{2}}^{\frac{1}{2}} X(f) e^{j2\pi fn} df$

  $X(f) = \sum_{n=-\infty}^{\infty} x[n] e^{-j2\pi fn}$
- Relationship between f and F

$$x[n] = \int_{-\frac{1}{2}}^{\frac{1}{2}} X(f) e^{j2\pi fn} df = x_a(nT) = \int_{-\infty}^{\infty} X_a(F) e^{j2\pi FnT} dF$$

$$= \sum_{k=-\infty}^{\infty} \int_{(k-1)\,F_s/2}^{(k+1)\,F_s/2} X_a(F) e^{j2\pi FnT} dF$$

17

Ideal reconstruction of continuous-time …

- Make use following relations

$$f = \frac{F}{F_s}, e^{j2\pi FnT} = e^{\frac{j2\pi F}{F_s}n} = e^{\frac{j2\pi n}{F_s}(F - kF_s)}$$

- Then, we can manipulate the former expression into

$$\frac{1}{F_s} \int_{-\frac{F_s}{2}}^{\frac{F_s}{2}} X(F) e^{\frac{j2\pi nF}{F_s}} dF = \int_{-\frac{F_s}{2}}^{\frac{F_s}{2}} \sum_{k=-\infty}^{\infty} X_a(F - kF_s)\, e^{\frac{j2\pi F}{F_s}n} dF$$

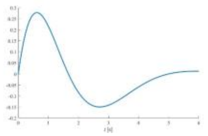
- Relation between sampled and analog spectra

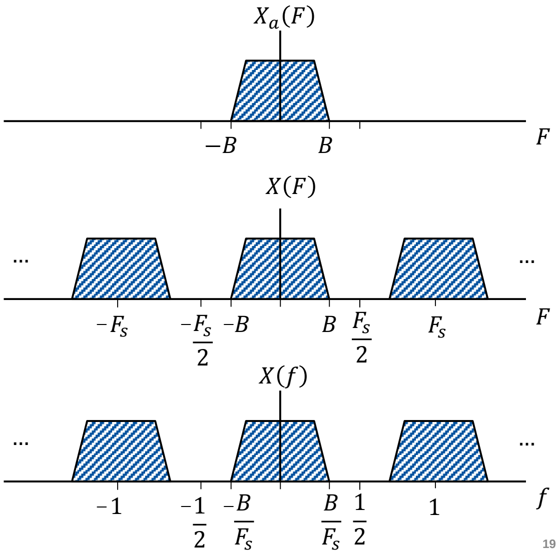
$$\boxed{\begin{gathered} X(F) = F_s \sum_{k=-\infty}^{\infty} X_a(F - kF_s), \text{ or} \\ X(f) = F_s \sum_{k=-\infty}^{\infty} X_a([f - k]F_s) \end{gathered}}$$

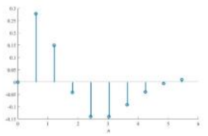
18

Ideal reconstruction of continuous-time …

- No aliasing







19

Ideal reconstruction of continuous-time …

- If discrete-time signal $x[n]$ has no aliasing in spectrum $X(F)$

$$X_a(F) = \begin{cases} \frac{1}{F_s} X(F), & |F| \leq \frac{F_s}{2} \\ 0, & |F| > \frac{F_s}{2} \end{cases}$$

- Analog signal can be reconstructed from samples $x[n]$

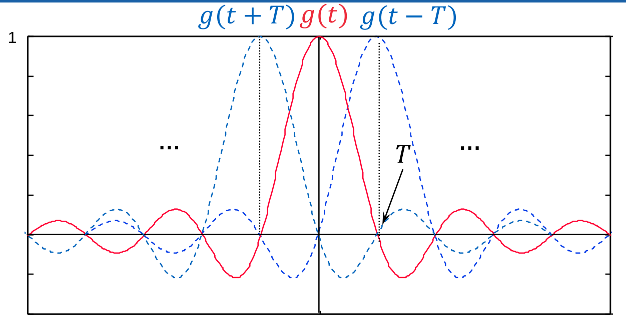
$$x_a(t) = \int_{-\frac{F_s}{2}}^{\frac{F_s}{2}} X_a(f) e^{j2\pi Ft} dF = \frac{1}{F_s} \int_{-\frac{F_s}{2}}^{\frac{F_s}{2}} X(F) e^{j2\pi Ft} dF = \cdots$$

$$\boxed{= \sum_{n=-\infty}^{\infty} x(n) \frac{\sin\left(\frac{\pi}{T}[t-nT]\right)}{\frac{\pi}{T}[t-nT]} = \sum_{n=-\infty}^{\infty} x[n] g[t-nT]}$$

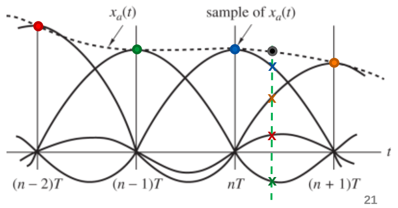
- Interpolation function is a *sinc* function

20

Ideal reconstruction of continuous-time …



$$x_a(t) = \sum_{n=-\infty}^{\infty} x(n) \frac{\sin\left(\frac{\pi}{T}[t-nT]\right)}{\frac{\pi}{T}[t-nT]}$$



Ideal reconstruction of continuous-time …

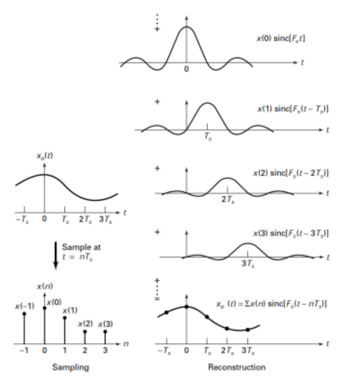
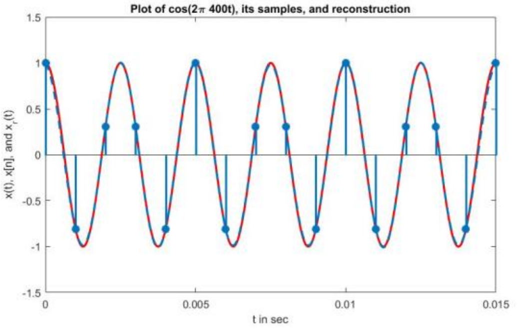


FIGURE 3.14 Reconstruction of band-limited signal from its samples

$$x_a(t) = \sum_{n=-\infty}^{\infty} x[n] g[t-nT]$$

Ideal reconstruction of continuous-time …

- Revisiting Example 1: Simultaneous plot of $\cos 2\pi 400t$, its samples, and reconstruction when $F_s = 1000$

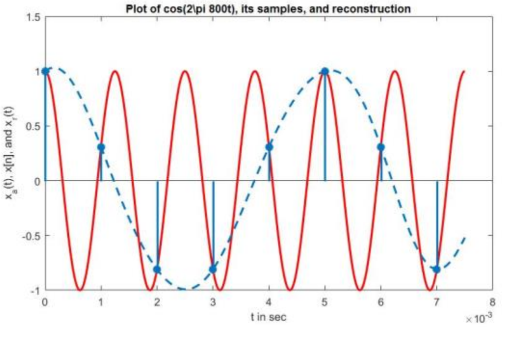


- Perfect reconstruction is possible

23

Ideal reconstruction of continuous-time …

- Revisiting Example 2: Simultaneous plot of $\cos 2\pi 800t$, its samples, and reconstruction when $F_s = 1000$

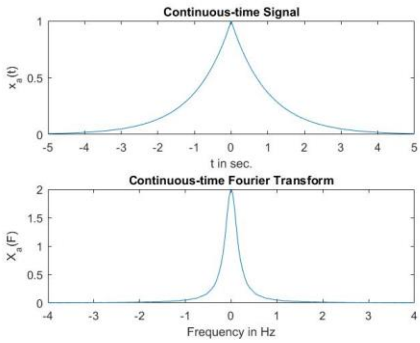


- Reconstruction of folded signal component $\cos 2\pi 200t$

24

Ideal reconstruction of continuous-time …

- Example 3: Sample $x_a(t) = e^{-|t|}$ at rates $F_{s_1} = 5$ and $F_{s_2} = 1$.



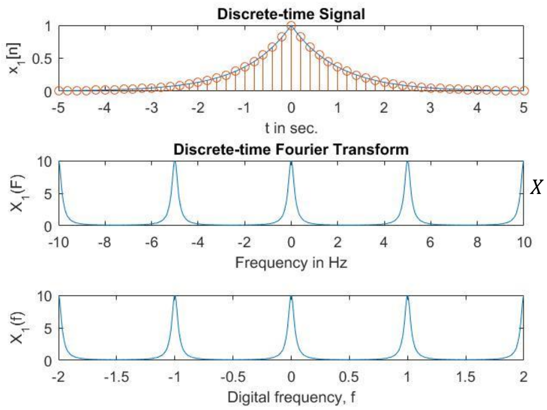
$$X(F) = \frac{2}{1 + (2\pi F)^2}$$

- How about spectra $X(F)$ and $X(f)$ for the two sampling rates? Sketch and draw conclusions about the reconstructed signals?

25

Ideal reconstruction of continuous-time …

- Example 3 (cont.): $x_1[n] = e^{-|n|T_1} = e^{-\frac{|n|}{F_{s_1}}}, F_{s_1} = 5$

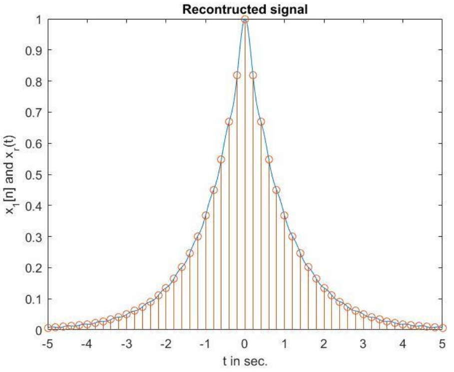


$$X\left(\frac{F}{F_s}\right) = \frac{1 - e^{\frac{-2}{F_s}}}{\left(1 - 2e^{-\frac{n}{F_s}} \cos\frac{2\pi F}{F_s} + e^{\frac{-2}{F_s}}\right)}$$

26

Ideal reconstruction of continuous-time …

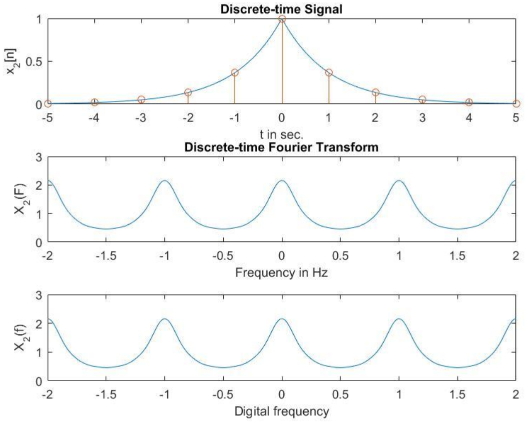
- Example 3 (cont.): $x_1[n] = e^{-nT_1} = e^{-\frac{n}{F_{S1}}}, F_{S_1} = 5$



- Excellent reconstruction.

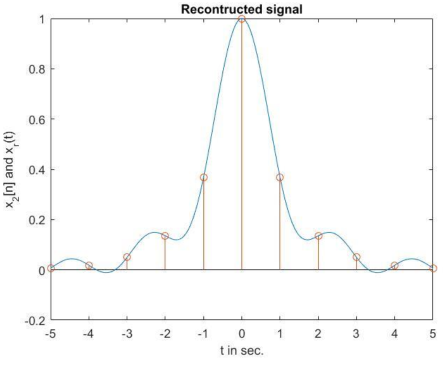
Ideal reconstruction of continuous-time …

- Example 3 (cont.): $x_2[n] = e^{-nT_2} = e^{-\frac{n}{F_{S2}}}, F_{S_2} = 1$



Ideal reconstruction of continuous-time …

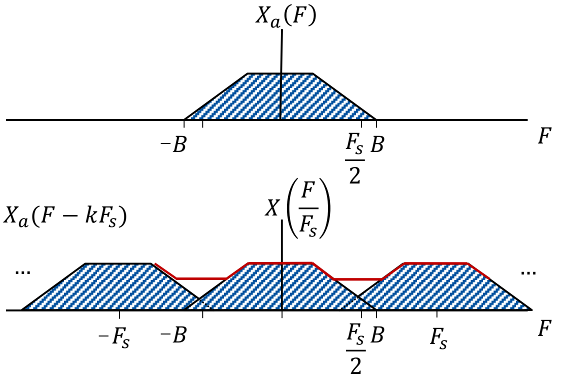
- Example 3 (cont.): $x_2[n] = e^{-nT_2} = e^{-\frac{n}{F_{S2}}}, F_{S_2} = 1$



- Reconstructed signal quite different from actual one (aliasing).

Ideal reconstruction of continuous-time …

- Aliasing

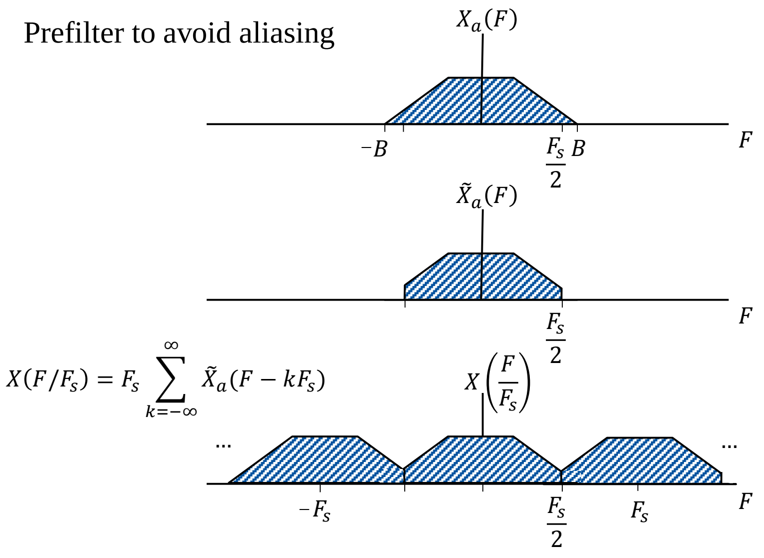


$$X(F/F_s) = F_s \sum_{k=-\infty}^{\infty} X_a(F - kF_s)$$

- Interpolation will produce $\hat{x}_a(t)$ corresponding to aliased spectrum
- Prefilter $x_a(t)$ to limit bandwidth before sampling

Ideal reconstruction of continuous-time …

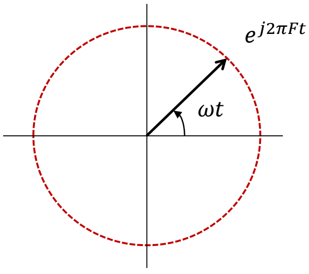
- Prefilter to avoid aliasing



31

Example: Wagon wheel effect

- Illusion of a wheel spinning in wrong direction
- Imagine phasor rotating at angular speed $\omega = 2\pi F$ rad/sec



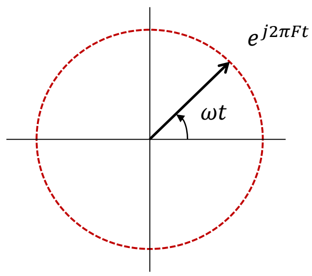
- Starting at $t = 0$, take a snapshot every T seconds, i.e., $nT = \frac{n}{F_s}$
- Find values of T such that the sampled phasor appears to rotate in clockwise direction rather than counter-clockwise?

32

Example: Wagon wheel effect…

- Demo on ItsLearning:

  `sampling_rotating_phasor.m`



Matlab

```
F = 1; Fs = 5; N = 3;
[f]=sampling_rotating_phasor(F,Fs,N);

F = 1; Fs = 4; N = 3;
[f]=sampling_rotating_phasor(F,Fs,N);

F = 1; Fs = 1.3; N = 3;
[f]=sampling_rotating_phasor(F,Fs,N);
```


Summary

Today:

- Sampling of analog and aliasing
- Sampling theorem
- Ideal reconstruction of analog signals

Next:

- Sampling in frequency domain: Discrete Fourier Transform

NTNU – Trondheim
Norwegian University of Science and Technology

TTT4120 Digital Signal Processing
Fall 2019

Lecture: The Discrete Fourier Transform

Prof. Stefan Werner
stefan.werner@ntnu.no
Office B329

Lecture in course book*

- Proakis, Manolakis Digital Signal Processing, 4th Ed.
  - 7.1.1 Frequency-domain sampling and reconstruction of discrete-time signals
  - 7.1.2 The discrete Fourier transform (DFT)
  - 7.2 Properties of the DFT

*Level of detail is defined by lectures and problem sets

2

Preliminary questions

- To perform frequency analysis of sequence $x[n]$ we need to convert it into its frequency-domain representation
- In our toolkit we find the discrete-time Fourier transform

$$X(\omega) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n}$$

- Is this a convenient representation?

Contents and learning outcomes

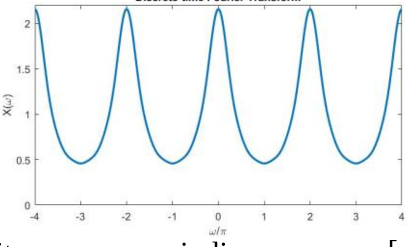
- Frequency-domain sampling and reconstruction
- Discrete Fourier Transform (DFT)
- Properties of the DFT

Motivation: Discrete Fourier transform

- Discrete Fourier transform (DFT) and inverse DFT (IDFT)
  - linear filtering of long sequences
  - frequency (spectrum) analysis
  - power spectrum estimation
- Efficient implementation using fast Fourier transform (FFT)

5

Frequency-domain sampling



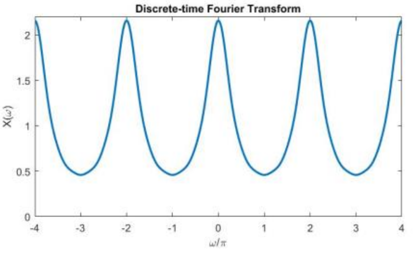
- Consider finite-energy aperiodic sequence $x[n]$ with DTFT

$$X(\omega) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n}$$

- Spectrum $X(\omega)$ is continuous but 2π-periodic
- Sample spectrum periodically in frequency
  - Benefits of performing such sampling?
  - Is sampled spectrum anymore related to $x[n]$?

6

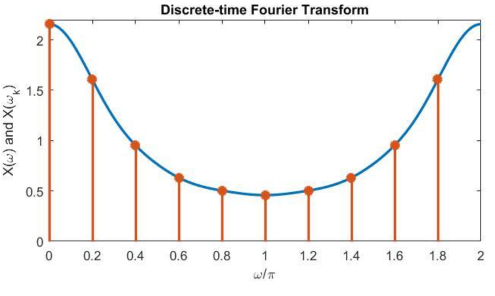
Frequency-domain sampling…



- Discussion:
  - Sampling *continuous-time* signal versus sampling *continuous-frequency* signal
  - Periodicity in transform-domain

7

Frequency-domain sampling…



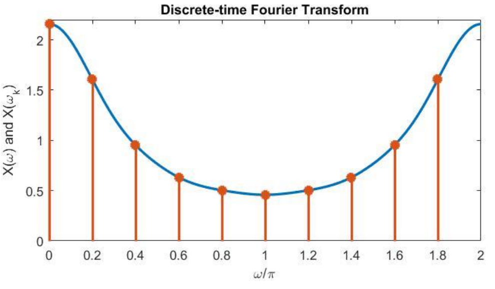
- In interval $0 \leq \omega \leq 2\pi$, take N equidistant samples,

$$X(\omega_k) = X(\omega)|_{\omega=\omega_k},$$

$$\omega_k = \frac{2\pi k}{N},\ k = 0, \dots, N-1$$

8

Frequency-domain sampling…



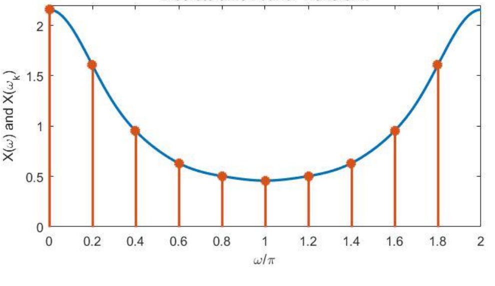
- DTFT $X(\omega)$ evaluated at ω_k

$$X(\omega_k) = \sum_{n=-\infty}^{\infty} x[n] e^{-\frac{j2\pi k}{N} n}, \; k = 0, \ldots, N-1$$

- Make use of identity $e^{-\frac{j2\pi k}{N} n} = e^{-\frac{j2\pi k}{N}(n+N)}$

9

Frequency-domain sampling…



- DTFT $X(\omega)$ evaluated at ω_k

$$X\left(\frac{2\pi k}{N}\right) = \sum_{n=-\infty}^{\infty} x[n] e^{-\frac{j2\pi k}{N} n}$$

$$= \cdots + \sum_{n=-N}^{-1} x[n] e^{-\frac{j2\pi k}{N} n}$$

$$+ \sum_{n=0}^{N-1} x[n] e^{-\frac{j2\pi k}{N} n}$$

$$+ \sum_{n=N}^{2N-1} x[n] e^{-\frac{j2\pi k}{N} n} + \cdots$$

10

Frequency-domain sampling…

- DTFT $X(\omega)$ evaluated at ω_k

$$X\left(\frac{2\pi k}{N}\right) = \sum_{n=0}^{N-1} \sum_{l=-\infty}^{\infty} x[n-lN]\, e^{-\frac{j2\pi k}{N}n}$$
$$= \sum_{n=0}^{N-1} x_p[n] e^{-\frac{j2\pi k}{N}n}$$

- Periodic extension of $x[n]$

$$x_p[n] = \sum_{l=-\infty}^{\infty} x[n-lN]$$

- Example 1: Given $x[n] = \delta[n] + 0.5\delta[n-1]$, sketch $x_p[n]$ for $N = 1$ and $N = 3$ and comment on the results

11

Frequency-domain sampling…

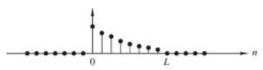
- Clearly $x_p[n] = \sum_{l=-\infty}^{\infty} x[n-lN]$ is periodic with period N
- Express as a discrete-time Fourier series $\Longrightarrow$

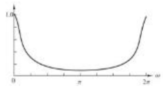
$$x_p[n] = \sum_{k=0}^{N-1} c_k e^{\frac{j2\pi k}{N}n},\ n = 0, \dots, N-1$$
$$c_k = \frac{1}{N}\sum_{n=0}^{N-1} x_p[n] e^{\frac{-j2\pi k}{N}n} = \frac{1}{N} X\left(\frac{2\pi k}{N}\right)$$

- Let us take stock and see where we stand

13

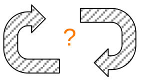
Frequency-domain sampling…





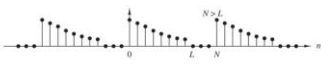
$x[n] \overset{DTFT}{\longleftrightarrow} X(\omega)$

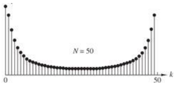
$\Uparrow$



$\Downarrow$ Sample at $\omega_k = \frac{2\pi k}{N}$

$x_p[n] = \sum_{l=-\infty}^{\infty} x[n - lN] \overset{DTFS}{\longleftrightarrow} X(\omega_k)$





- When can $x[n]$ be recovered from $x_p[n]$?
  - Duration of sequence $x[n]$ versus period of $x_p[n]$?

Frequency-domain sampling…

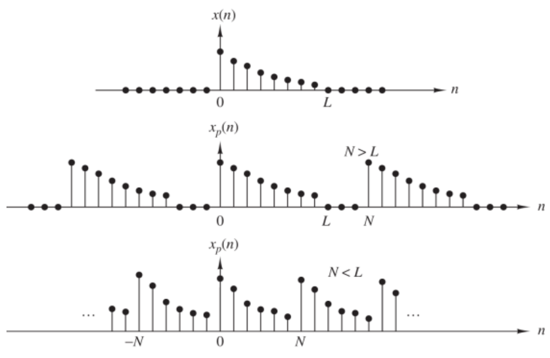
- Lesson learned:

The spectrum $X(\omega)$ of an aperiodic sequence $x[n]$ of finite duration L, can be recovered from samples $X(\omega_k)$, with $\omega_k = \frac{2\pi k}{N}$, if the number of samples $N \geq L$

- Procedure for closing the circle:
  1. Compute $x_p[n] = \frac{1}{N}\sum_{k=0}^{N-1} X\left(\frac{2\pi k}{N}\right) e^{\frac{j2\pi k}{N}n}$, $n = 0, \dots, N-1$
  2. Set $x[n] = x_p[n]$ for $0 \leq n \leq N - 1$, zero elsewhere
  3. Compute $X(\omega) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n}$

Frequency-domain sampling…

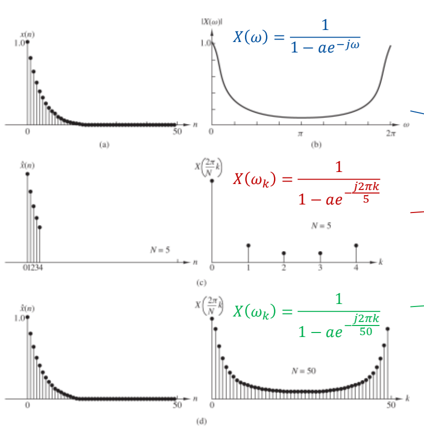
- Example 2: Which periodic extension of $x[n]$ can be used to recover spectrum $X(\omega)$?



16

Frequency-domain sampling…

- Example 3: Infinite duration sequences, reconstructed sequence will suffer from aliasing, $x[n] = a^n u[n]$, $|a| < 1$.



Matlab

```
B = [1]; A = [1 -0.9];
[X,w]=freqz(B,A,'whole');
figure, plot(w/pi,abs(X));
[x,n]=impz(B,A)
figure, stem(n,x),hold on
X1 = X(1:length(w)/5:end)
figure,
stem(ifft(X1),’r’)
X2 = X(1:length(w)/50:end)
figure,
stem(ifft(X2),’g’)
```

17

Discrete Fourier transform (DFT)

- Putting the bits and pieces together (remember)

$$x_p[n] = \sum_{k=0}^{N-1} c_k e^{\frac{j2\pi k}{N}n}, n = 0, \dots, N-1$$

$$c_k = \frac{1}{N}\sum_{n=0}^{N-1} x_p[n]e^{-\frac{j2\pi k}{N}n} = \frac{1}{N}X\left(\frac{2\pi k}{N}\right)$$

- For sequence $x[n]$ of length $L \leq N$, $x[n] = 0$, $L \leq n \leq N$

$$\text{DFT: } X(k) = \sum_{n=0}^{N-1} x[n]e^{-\frac{j2\pi k}{N}n}, k = 0, \dots, N-1$$

$$\text{IDFT: } x[n] = \frac{1}{N}\sum_{k=0}^{N-1} X(k)e^{\frac{j2\pi k}{N}n}, n = 0, \dots, N-1$$

- Notation: $X(k) \equiv X(\omega_k)$, $X(k) = \text{DFT}_N\{x[n]\}$

18

Discrete Fourier transform (DFT)

$$\text{DFT: } X(k) = \sum_{n=0}^{N-1} x[n]e^{-\frac{j2\pi k}{N}n}, k = 0, \dots, N-1$$

$$\text{IDFT: } x[n] = \frac{1}{N}\sum_{k=0}^{N-1} X(k)e^{\frac{j2\pi k}{N}n}, n = 0, \dots, N-1$$

- What happens when increasing $N > L$, L is kept fixed?
  - In frequency-domain?
  - In time-domain?
- Using $N > L$ samples for computing the DFT is commonly referred to as *zero padding* and improves resolution

19

Discrete Fourier transform (DFT)…

- Example 4: Plot N-point DFT of $x[n] = \sum_{l=0}^{3} \delta[n-l]$
  for $N = 4$ and $N = 40$

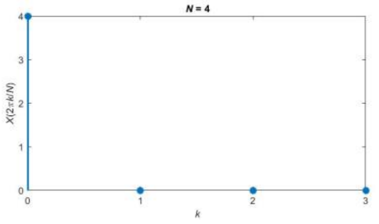
$$X(\omega) = \sum_{n=0}^{L-1=3} e^{-j\omega n} = \frac{1-e^{-j\omega L}}{1-e^{-j\omega}} = \frac{\sin(\omega L/2)}{\sin(\omega/2)} e^{-j\omega(L-1)/2}$$

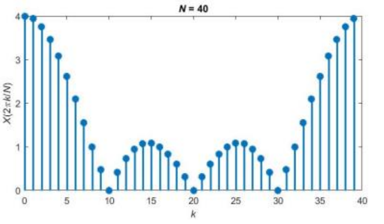
$$X(k) = \sum_{n=0}^{L-1=3} e^{-\frac{j2\pi k}{N}n} = \frac{1-e^{-\frac{j2\pi k}{N}L}}{1-e^{-\frac{j2\pi k}{N}}} = \frac{\sin(\pi kL/N)}{\sin(\pi k/N)} e^{-j\pi k(L-1)/N}$$

20

Discrete Fourier transform (DFT)…

- Example 4: Plot N-point DFT of $x[n] = \sum_{l=0}^{3} \delta[n-l]$
  for $N = 4$ and $N = 40$





Matlab
```
L = 4; x = ones(1,L);
N = L*10; x_zp = [x,zeros(1,N-L)];
stem((0:N-1),abs(fft(x_zp,N)));
```

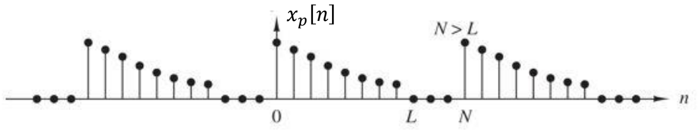
21

Properties of the DFT

- Periodicity
- Linearity
- Time reversal
- Circular time shift
- Circular frequency shift
- Conjugation
- Circular convolution
- Multiplication of two sequences
- Parseval's theorem

Properties of the DFT…

- Properties are similar to those of the DTFT
- Keep in mind is that operations on $X(k)$ in frequency domain corresponds to *operations on* $x_p[n]$ in time domain



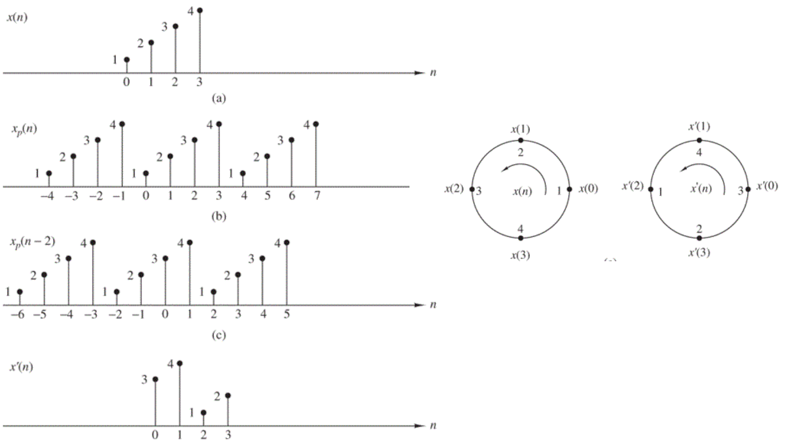
$$x_p[n] = x[n, \text{modulo } N] = x([n])_N$$

⇒ Shifting $x_p[n]$ in time by for k units, $x_p[n-k]$, is identical to a circular shift of $x[n]$ in interval $0 \le n \le N-1$

$$x([n-k])_N \equiv x[n-k, \text{modulo } N]$$

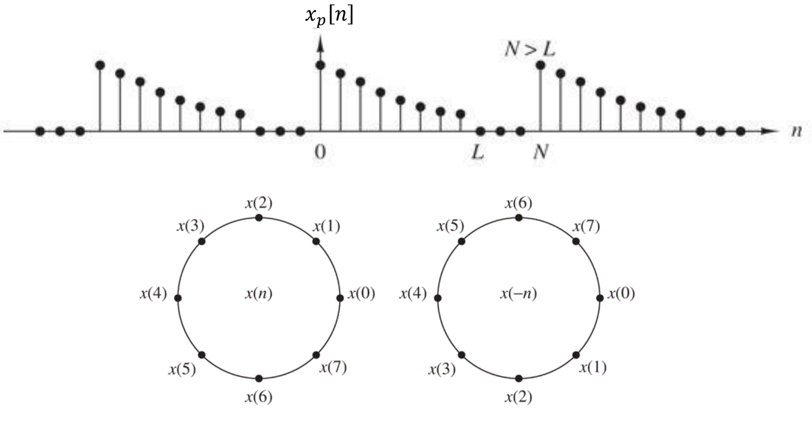
Properties of the DFT…

- Shifting:



Properties of the DFT…

- Time-reversal:



$$x([-n])_N \equiv x[-n, \text{modulo } N] = x[N-n],\ 0 \le n \le N-1$$

Properties of the DFT…

- Periodicity: $x[n] = x[n+N] \overset{\mathrm{DFT}_N}{\longleftrightarrow} X(k) = X(k+N)$
- Linearity: $a_1x_1[n] + a_2x_2[n] \overset{\mathrm{DFT}_N}{\longleftrightarrow} a_1X_1(k) + a_2X_2(k)$
- Time reversal: $x[N-n] \overset{\mathrm{DFT}_N}{\longleftrightarrow} X(N-k)$
- Circular time shift: $x([n-l])_N \overset{\mathrm{DFT}_N}{\longleftrightarrow} X(k)e^{-j2\pi kl/N}$
- Circular frequency shift: $x[n]e^{j2\pi ln/N} \overset{\mathrm{DFT}_N}{\longleftrightarrow} X((k-l))_N$
- Conjugation: $x^*[n] \overset{\mathrm{DFT}_N}{\longleftrightarrow} X^*(N-k)$
- Circular convolution: $x_1[n] \otimes_N x_2[n] \overset{\mathrm{DFT}_N}{\longleftrightarrow} X_1(k)X_2(k)$
- Parseval's theorem: $\sum_{n=0}^{N-1} x[n]y^*[n] \overset{\mathrm{DFT}_N}{\longleftrightarrow} \frac{1}{N}\sum_{n=0}^{N-1} X(k)Y^*(k)$

Properties of the DFT…

- Circular convolution: $x_1[n] \otimes_N x_2[n] \overset{\mathrm{DFT}_N}{\longleftrightarrow} X_1(k)X_2(k)$

  $x_1[n] \otimes_N x_2[n] = \sum_{k=0}^{N-1} x_1[k]x_2([n-k])_N \, , n = 0,1, \dots N-1$

- Linear convolution of causal sequences $x_1[n]$ and $x_2[n]$

$$x_1[n] * x_2[n] = \sum_{k=0}^{N-1} x_1[k]x_2[n-k]$$

- In general, $x_1[n] \otimes_N x_2[n] \neq x_1[n] * x_2[n]$

  $\Rightarrow$ important when applying the DFT to linear system analysis (next lecture)

Summary

Today:
- Frequency-domain sampling and reconstruction
- The DFT (discrete Fourier transform)
- Properties of the DFT

Next:
- Using DFT for filtering and frequency analysis

28

NTNU – Trondheim
Norwegian University of
Science and Technology

TTT4120 Digital Signal Processing
Fall 2020

Lecture: Discrete Fourier Transform for Filtering and Frequency Analysis

Prof. Stefan Werner
stefan.werner@ntnu.3no
Office B329

1

Lecture in course book*

- Proakis, Manolakis Digital Signal Processing, 4th Ed.
  - 7.3.1 Use of DFT in linear filtering
  - 7.3.2 Filtering of long sequences (overlap and add method)
  - 7.4 Frequency analysis using DFT

*Level of detail is defined by lectures and problem sets

2

2

Preliminary questions

- The discrete-time Fourier transform (DTFT) allows us to perform frequency analysis of signals and filtering of signals

$$X(\omega),\ Y(\omega),\ \text{and}\ H(\omega)$$

- What practical problems arise when applying the DTFT for these tasks?

3

3

Contents and learning outcomes

- Linear filtering using discrete Fourier transform (DFT)
- Filtering of long sequences (overlap-add)
- Frequency analysis using DFT

4

4

Linear filtering using DFT

- Remember (Lecture 3):

$$\begin{array}{c} x[n] \\ X(\omega) \end{array} \longrightarrow \boxed{h[n]} \longrightarrow \begin{array}{c} y[n] = h[n] * x[n] \\ Y(\omega) \ = H(\omega)\, X(\omega) \end{array}$$

- Convolution can sometimes be computationally demanding
- If we know $X(\omega)$ and $H(\omega)$, we can obtain $y[n]$ from

$$y[n] = \mathcal{F}^{-1}\{Y(\omega)\} \ = \mathcal{F}^{-1}\{H(\omega)\, X(\omega)\}$$

- Conceptually simpler
- How to implement these calculations on a computer?

Linear filtering using DFT…

- DFT can be implemented efficiently on a computer

$$\begin{array}{c} x[n] \\ X(\omega_k) \end{array} \longrightarrow \boxed{h[n]} \longrightarrow \begin{array}{c} y[n] = h[n] * x[n] \\ \updownarrow ? \\ Y(\omega_k) \ = H(\omega_k)\, X(\omega_k)? \end{array}$$

- Can compute $X(k) = \mathrm{DFT}_N\{x[n]\}$ and $H(k) = \mathrm{DFT}_N\{h[n]\}$
- Convenient if $y[n]$ could be obtained from

$$y[n] = \mathrm{IDFT}_N\{Y(k)\} \ = \mathrm{IDFT}_N\{H(k)\, X(k)\}$$

- Not true in general but we investigate when it can be done

Linear filtering using DFT…

- Product of two DFTs corresponds to circular convolution

$$x_1[n] \otimes_N x_2[n] \overset{\mathrm{DFT}_N}{\longleftrightarrow} X_1(k)X_2(k)$$

- Not useful to compute output $y[n]$ of linear filter $h[n]$
- Assume finite-duration input sequence $x[n]$ and impulse response $h[n]$, i.e.,

$$x[n] = 0, n < 0 \text{ and } n \geq L$$
$$h[n] = 0, n < 0 \text{ and } n \geq M$$

- Output $y[n]$ can be calculated

$$y[n] = \sum_{n=0}^{N-1} h(k)x[n-k] \overset{\mathcal{F}}{\leftrightarrow} Y(\omega) = H(\omega)X(\omega)$$

7

7

Linear filtering using DFT…

- Output has finite duration $M + L - 1$

$$y[n] = 0, n < 0 \text{ and } n \geq M + L - 1$$

- We know from before that we can restore spectrum $Y(\omega)$ from its sampled spectrum $Y(\omega_k)$, $k = 0,1, \dots N - 1$, if

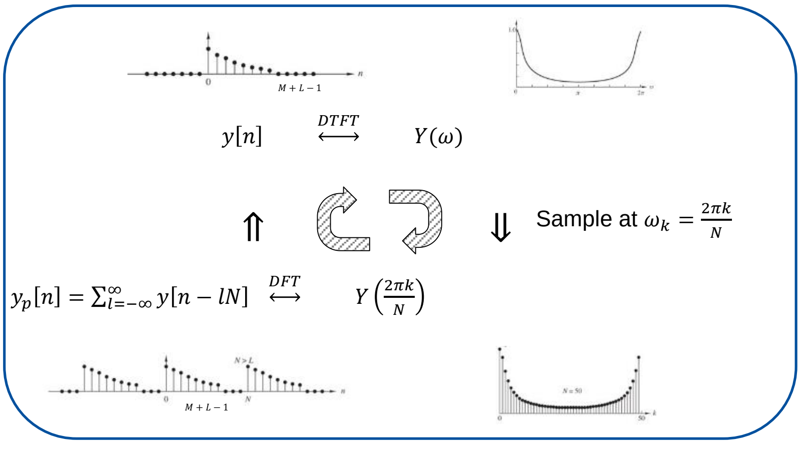
$$N \geq M + L - 1$$

∴ DFT of size $N \geq M + L - 1$ is required to uniquely represent $y[n]$ in frequency domain

8

8

Linear filtering using DFT…

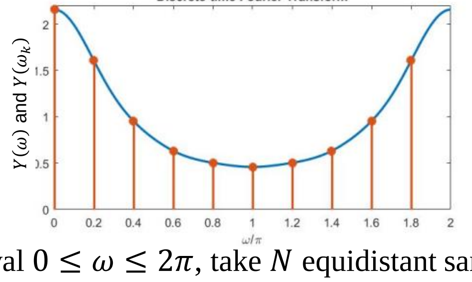


- Remember from last lecture, if $N \geq M + L - 1$,
  $y_p[n] = y[n]$ for $0 \leq n \leq N - 1$

9

9

Linear filtering using DFT…



- In interval $0 \leq \omega \leq 2\pi$, take N equidistant samples,

$$Y(\omega_k) = Y(\omega)\big|_{\omega=\frac{2\pi k}{N}} = H(\omega)X(\omega)\big|_{\omega=\frac{2\pi k}{N}}, \;\; k = 0, \dots, N-1$$

$$\Rightarrow \boxed{Y(k) = H(k)X(k), \; k = 0, \dots, N-1}$$

N-point DFT of $h[n]$ N-point DFT of $x[n]$

10

10

Linear filtering using DFT…

- Since $x[n]$ and $h[n]$ have duration less than $N \Rightarrow$ need to *pad sequences with zeros to increase lengths to* $N \geq M + L - 1$

$$x[n] = \{x[0], x[1], \dots, x[L-1], \underbrace{0, \dots, 0}_{N-L}\}$$

$$h[n] = \{h[0], h[1], \dots, h[M-1], \underbrace{0, \dots, 0}_{N-M}\}$$

- Output sequence can now be computed as

$$y[n] = \mathrm{IDFT}_N\{Y(k)\} = \mathrm{IDFT}_N\{H(k)X(k)\}$$

$$= \mathrm{IDFT}_N\left\{\mathrm{DFT}_N\{h[n]\} \cdot \mathrm{DFT}_N\{h[n]\}\right\}$$

- Note that choosing $N < M + L - 1$ will lead to time-domain aliasing $(h[n] \otimes_N x[n] \neq h[n] * x[n])$

11

11

Linear filtering using DFT…

- Example 1: Given $x[n] = \{\underline{1}, 2,2,1\}$, and $h[n] = \{\underline{1}, 2,3\}$. Which of the following calculations provide us with correct output sequence $y[n]$?

1. $y[n] = \mathrm{IDFT}_4\left\{\mathrm{DFT}_4\{x[n]\} \cdot \mathrm{DFT}_4\{h[n]\}\right\}$
2. $y[n] = \mathrm{IDFT}_3\left\{\mathrm{DFT}_3\{x[n]\} \cdot \mathrm{DFT}_3\{h[n]\}\right\}$
3. $y[n] = \mathrm{IDFT}_{16}\left\{\mathrm{DFT}_{16}\{x[n]\} \cdot \mathrm{DFT}_{16}\{h[n]\}\right\}$
4. $y[n] = \mathrm{IDFT}_6\left\{\mathrm{DFT}_6\{x[n]\} \cdot \mathrm{DFT}_6\{h[n]\}\right\}$

Matlab

```
N  = 4; % Try different N
x  = [1,2,2,1];
H  = [1,2,3];
y1 = ifft(fft(h,N)).*fft(x,N),N)
y2 = conv(x,h)
```

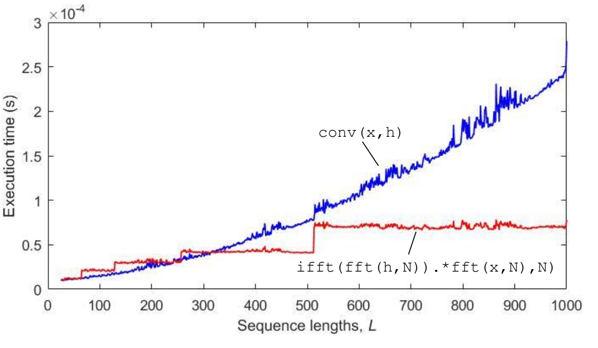
12

12

Linear filtering using DFT…

- Example 2: When does frequency-domain filtering outperform time-domain filtering?

  Assume that both $x[n]$ and $h[n]$ have length L



13

Filtering of long sequences

$$\{\ldots, x[0], x[1], x[2], \ldots, x[1000], x[1001], \ldots\} \rightarrow \boxed{h[n]} \rightarrow \{\ldots, y[0], y[1], y[2], \ldots, y[1000], y[1001], \ldots\}$$

$X(k)$ $\qquad$ $Y(k) = X(k)H(k)$

- Assume that input sequence $x[n]$ is extremely long
- All input samples are required before we can perform DFT
- What are the implications on memory requirements and processing delay?
- Extreme case of real-time processing (no beginning or end)!

14

Filtering of long sequences…

$\{\dots, x[0], x[1], x[2], \dots, x[1000], x[1001], \dots\} \longrightarrow \boxed{h[n]} \longrightarrow \{\dots, y[0], y[1], y[2], \dots, y[1000], y[1001], \dots\}$

- All N' input samples are required before we can perform DFT
  $\Rightarrow$ Delay before output is produced increases with N'
- We need a method that can filter long sequences in time-domain that is memory- and delay-efficient
- Remember the *additivity property* of convolution

$$y[n] = h[n] * (x_1[n] + x_2[n])$$
$$= h[n] * x_1[n] + h[n] * x_2[n]$$
$$= y_1[n] + y_2[n]$$

Filtering of long sequences…

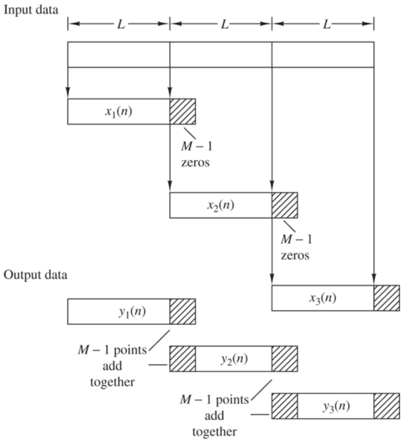
$\{\dots, x[0], x[1], x[2], \dots, x[1000], x[1001], \dots\} \longrightarrow \boxed{h[n]} \longrightarrow \{\dots, y[0], y[1], y[2], \dots, y[1000], y[1001], \dots\}$

Strategy:

1. Divide input sequence $x[n]$ into *non-overlapping blocks* $x_m[n]$ *each of length* L
2. Filter each input block $x_m[n]$ to produce output block $y_m[n]$
3. Combine outputs: $y[n] = \sum_m y_m[n]$

- If length of $h[n]$ is M, the length of $y_m[n]$ is $L + M - 1$
  $\Rightarrow$ last $M - 1$ values of $y_{m-1}[n]$ added to beginning of $y_m[n]$

Filtering of long sequences…



- Filtering using N-point DFT requires zero-padding of sequences $x_m[n]$ and $h[n]$

17

17

Filtering of long sequences…

Steps of overlap-add:

1. Divide $x[n]$ into *non-overlapping* blocks $x_m[n]$ of length L
2. Pad $h[n]$ with zeros to length $N \geq M + L - 1$
3. Compute $H(k) = \text{DFT}_N\,\{h[n]\}, k = 0, \ldots, N - 1$
4. For each block m:

   4.1 Pad $x_m[n]$ to with zeros to length $N \geq M + L - 1$

   4.2 Compute $X_m(k) = \text{DFT}_N\,\{x_m[n]\}, k = 0, \ldots, N - 1$

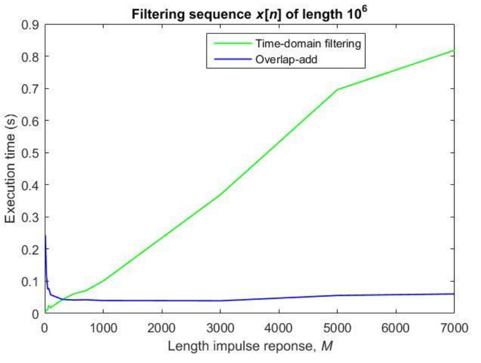
   4.3 Multiply $Y_m(k) = H(k)X_m(k), k = 0, \ldots, N - 1$

   4.4 Compute $y_m[n] = \text{IDFT}_N\,\{Y_m(k)\}, n = 0, \ldots, N - 1$
5. Form $y[n]$ by overlapping and adding the last $M - 1$ values of $y_{m-1}[n]$ and the first $M - 1$ values of $y_m[n]$

18

18

Filtering of long sequences…

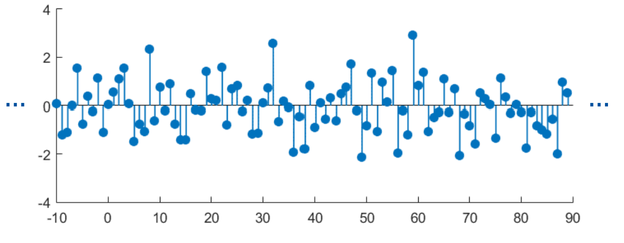



```
Matlab
M  = 2000; % Try different N
x = rand(1,1e6);
h = rand(1,M);
y1 = fftfilt(h,x);
y2 = filter(h,1,x);
```

19

19

Frequency analysis

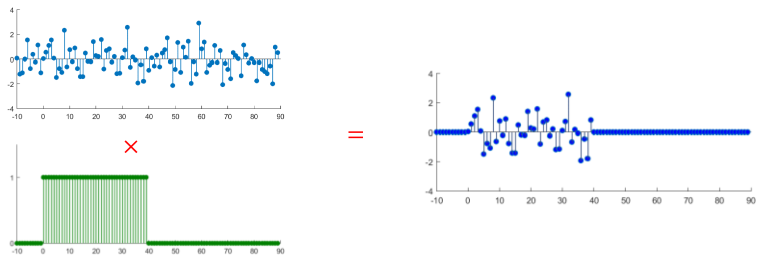


- DTFT: $X(\omega) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n}$
- In practice $x[n]$ needs to have finite duration
  $\Rightarrow$ Spectrum $X(\omega)$ approximated from a finite data record
- How does the approximation, $\hat{X}(\omega)$, depend on the number of available samples?

20

20

Frequency analysis…



- Limiting the number of samples is the same as multiplying original sequence $x[n]$ by a window $w[n]$

$$\hat{x}[n] = x[n]w[n]$$

where

$$w[n] = \begin{cases} 1 & \text{for } 0 \le n \le L-1 \\ 0 & \text{otherwise} \end{cases}$$

Frequency analysis…

- Multiplication in time-domain corresponds to

$$\hat{X}(\omega) = X(\omega) * W(\omega) = \frac{1}{2\pi}\int_{-\pi}^{\pi} X(\theta)W(\omega-\theta)d\theta$$

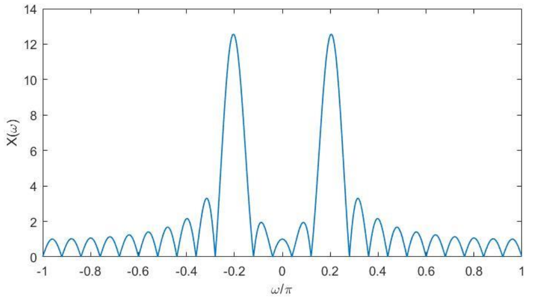
- Using DFT we would get

$$\hat{X}(k) = \sum_{n=0}^{N-1} \hat{x}[n]e^{-\frac{j2\pi k}{N}n} = \sum_{n=0}^{N-1} x[n]e^{-\frac{j2\pi k}{N}n}$$

$$= \hat{X}(\omega)\Big|_{\omega=\frac{2\pi k}{N}} = \frac{1}{2\pi}\int_{-\pi}^{\pi} X(\theta)W\left(\frac{2\pi k}{N}-\theta\right)d\theta$$

Frequency analysis…

- Example: $x[n] = \cos 0.2\pi n$ for $N = 2048$ and $L = 25$



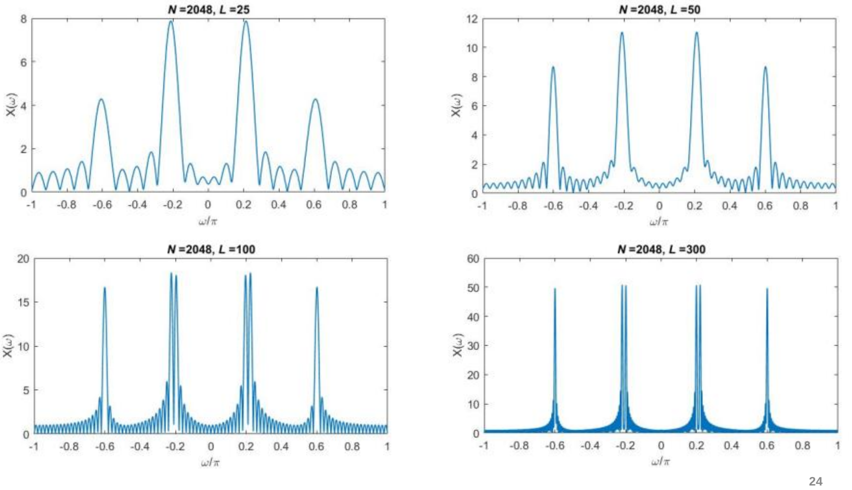

```
Matlab
N  = 2048; L  = 25;
n  = (0:N-1); k = (-1:2/N:1-2/N);
wn = [(L-n) > 0];
x  = cos(0.2*pi*n);
x_ = wn.*x;
plot(k,abs(fftshift(fft(x_hat,N))))
```

23

23

Frequency analysis…

- Example: $x[n] = \cos 0.2\pi n + \cos 0.22\pi n + \cos 0.6\pi n$



24

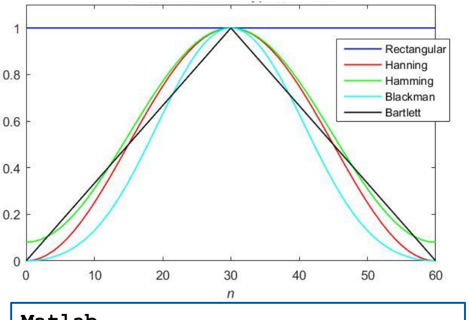
24

Frequency analysis …

- Windowing distorts the signal:
  - Spectrum peaks are smoothened out
  - Sidelobes are causing spectral leakage
- Increasing the window length, increases resolution
- Width of main lobe of rectangular window $4\pi/L$
- Use different windows to reduce spectral sidelobes
  - Width of main lobe is increasing when compared to rectangular window

Frequency analysis …

- Different window types, $L = 61$



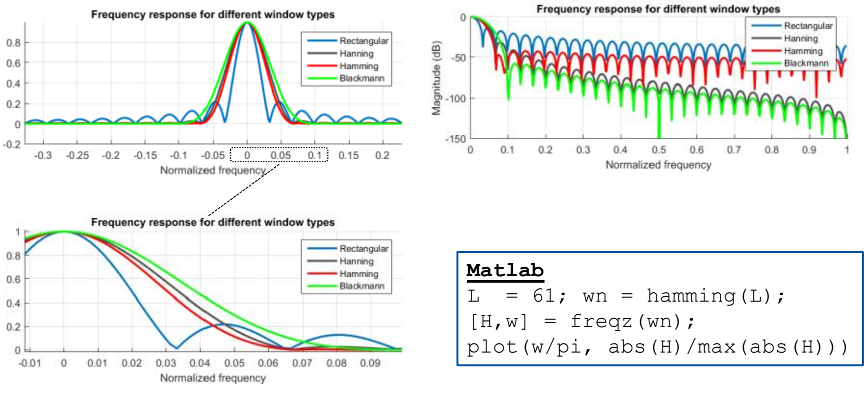
Matlab

```
% type 'help window' for options

L = 61; n = (0:L-1);
w1 = window(@hamming,L);
w2 = window(@bartlett,L);
plot(n,[w1,w2])
```

Frequency analysis…

- Frequency response for different window types

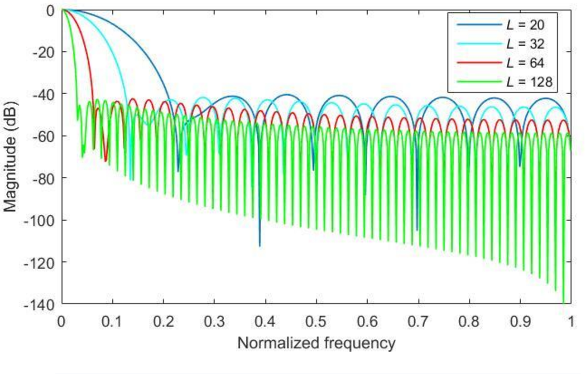


Matlab
```
L  = 61; wn = hamming(L);
[H,w] = freqz(wn);
plot(w/pi, abs(H)/max(abs(H)))
```

27

Frequency analysis…

- Frequency response for different window lengths L

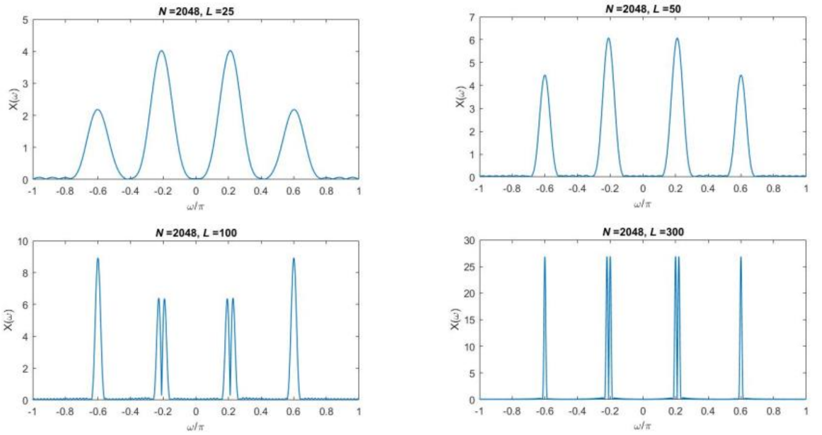


Matlab
```
L  = 20; wn = hamming(L);
[H,w] = freqz(wn);
plot(w/pi,20*log10(abs(H)/max(abs(H))))
```

28

Frequency analysis…

- Revisiting: $x[n] = \cos 0.2\pi n + \cos 0.22\pi n + \cos 0.6\pi n$ (Hamming window)



29

29

Frequency analysis …

- Increasing the window length, increases resolution
- Sidelobes are causing spectral leakage
- Width of main lobe versus sidelobe suppression
  - Use of different windows

30

30

Summary

- Today we discussed:
  - Filtering and frequency analysis using the DFT
- Next time:
  - Fast Fourier transform (FFT)

31

31

NTNU – Trondheim
Norwegian University of
Science and Technology

TTT4120 Digital Signal Processing
Fall 2019

The Fast Fourier Transform (FFT)

Prof. Stefan Werner
stefan.werner@ntnu.no
Office B329

Department of Electronic Systems

Lecture in course book*

- Proakis, Manolakis Digital Signal Processing, 4th Ed.
  - 8.1 Efficient computation of the DFT: FFT algorithms
  - 8.1.3 Radix-2 FFT algorithms (decimation in time)

*Level of detail is defined by lectures and problem sets

Contents and learning outcomes

- Complexity of the DFT
- Divide and conquer
- Properties of $e^{j2\pi nk/N}$
- Radix-2 FFT

Motivation

- *Fast Fourier transforms* (FFTs) have revolutionized DSP
- What is the FFT?
  – DFT calculation made much faster
  – Speedup increases with DFT size
- Focus on the simplest formulation: the radix-2 decimation-in-time FFT algorithm

Motivation

- Some dates:
  - Gauss 1805
  - Cooley and Tukey 1965

Principal Discoveries of Efficient Methods of Computing the DFT

Researcher(s)	Date	Lengths of Sequence	Number of DFT Values	Application
C. F. Gauss [10]	1805	Any composite integer	All	Interpolation of orbits of celestial bodies
F. Carlini [28]	1828	12	7	Harmonic analysis of barometric pressure variations
A. Smith [25]	1846	4, 8, 16, 32	5 or 9	Correcting deviations in compasses on ships
J. D. Everett [23]	1860	12	5	Modeling underground temperature deviations
C. Runge [7]	1903	$2^n K$	All	Harmonic analysis of functions
K. Stumpff [16]	1939	$2^n K$, $3^n K$	All	Harmonic analysis of functions
Danielson & Lanczos [5]	1942	2^n	All	X-ray diffraction in crystals
L. H. Thomas [13]	1948	Any integer with relatively prime factors	All	Harmonic analysis of functions
I. J. Good [3]	1958	Any integer with relatively prime factors	All	Harmonic analysis of functions
Cooley & Tukey [1]	1965	Any composite integer	All	Harmonic analysis of functions
S. Winograd [14]	1976	Any integer with relatively prime factors	All	Use of complexity theory for harmonic analysis

Michael T. Heideman, Don H. Johnson, and C. Sideny Burrus. Gauss and the History of the Fast Fourier Transform. Archive for History of Exact Sciences (Springer), 34(3):265-277, September 1985.

5

Complexity of the DFT

- DFT involves computing the sequence:

$$X(k) = \sum_{n=0}^{N-1} x[n] e^{-j2\pi nk/N}$$
$$= \sum_{n=0}^{N-1} x[n] W_N^{nk}$$

- IDFT involves computing the sequence

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X(k) W_N^{-nk}$$

- Similar calculations (same phasors but different direction)

  ⇒ concentrate on the DFT

6

Complexity of the DFT…

- What do we mean by computational efficiency?
  - Number of additions
  - Number of multiplications
  - Memory requirements
  - Scalability

Complexity of the DFT…

$$X(k) = \sum_{n=0}^{N-1} x[n] e^{-j2\pi nk/N}, k = 0, 1, \ldots, N-1$$

- For each $k, n = 0, \ldots, N-1$
  - Evaluate $e^{-j2\pi nk/N} = \cos\frac{2\pi kn}{N} - j\sin\frac{2\pi kn}{N}$
  - Multiply two complex numbers $x[n]e^{-j2\pi nk/N}$
- Total complexity for direct computation
  - $2N^2$ evaluations of trigonometric functions
  - $4N^2$ real multiplication
  - $2N(2N-1)$ real additions
- In addition indexing and addressing operations

Complexity of the DFT…

- Direct computation of the DFT is highly inefficient
- Complexity grows with the square of the signal length
- Severely limits the practical use for long signals
- How to reduce the complexity of the DFT?
  - Divide-and conquer approach
  - Exploit symmetry and periodicity properties
- Resulting algorithm will have complexity $N\log_2 N$!

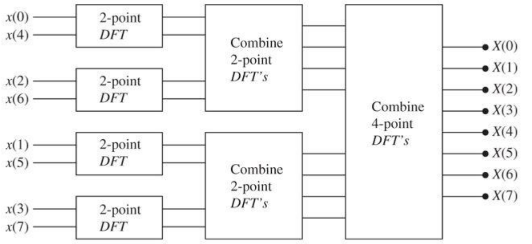
9

Divide-and-conquer

- FFT algorithms are based on a divide-and-conquer approach:
  - Divide a problem instance into subproblems
  - Conquer the subproblems by solving them recursively
  - Combine the solutions for the subproblems to a solution for the original problem
- Example 1: You are only capable of adding two numbers. How to efficiently compute the sum $S_N = \sum_{n=0}^{N-1} x[n]$? Consider the computation time for cases:
  1) Single person is assigned the task
  2) A group of persons is assigned the task

10

Divide-and-conquer…



- The approach taken in deriving the FFT is to recursively divide the N-point DFT into successively smaller DFTs
- Exploit symmetry and periodicity of $W_N^{kn} = e^{-j2\pi kn/N}$

11

Properties of $e^{j2\pi nk/N}$

- The key to reduce the complexity of the DFT is to exploit properties of $W_N = e^{-j2\pi/N}$

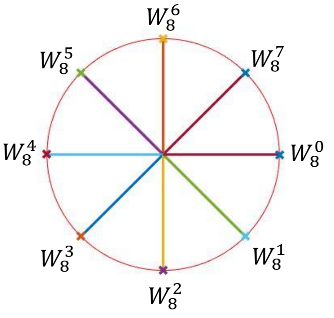
$$W_N^{k+N} = ?$$
$$W_N^{k+N/2} = ?$$
$$W_N^{2k} = ?$$

12

Properties of $e^{j2\pi nk/N}$…

- Example 2: $W_8 \;\; = e^{-j2\pi/8}$

$$W_N^{k+N} = ?$$
$$W_N^{k+N/2} = ?$$
$$W_N^{2k} = ?$$

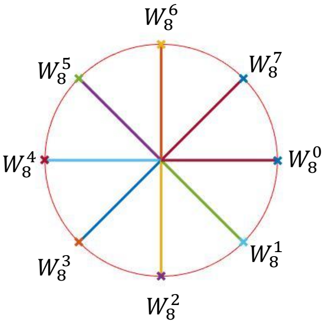


13

Properties of $e^{j2\pi nk/N}$…

- The key to reduce the complexity of the DFT is to exploit properties of $W_N \;\; = e^{-j2\pi/N}$

$$W_N^{k+N} = e^{-\frac{j2\pi(k+N)}{N}} = W_N^k$$
$$W_N^{k+N/2} = e^{-\frac{j2\pi\left(k+\frac{N}{2}\right)}{N}} = -W_N^k$$
$$W_N^{2k} = e^{-\frac{j2\pi k}{N/2}} = W_{N/2}^k$$



14

Radix-2 FFT

- Assume $N = 2^{\nu}$
- Split sequence $x[n]$ into two subsequences

$$f_1[n] = x[2n] \text{ and } f_2[n] = x[2n+1]$$

- We can write the DFT in terms of even and odd values of n

$$X(k) = \sum_{n=0}^{N-1} x[n] W_N^{nk}, k = 0, \ldots N-1$$

$$= \sum_{n \text{ even}} x[n] W_N^{nk} + \sum_{n \text{ odd}} x[n] W_N^{nk}$$

$$= \sum_{m=0}^{N/2-1} x[2m] W_N^{2mk} + \sum_{m=0}^{N/2-1} x[2m+1] W_N^{(2m+1)k}$$

15

Radix-2 FFT…

- Rewrite in terms of decimated sequences…

$$X(k) = \sum_{m=0}^{N/2-1} f_1[m] W_{N/2}^{mk} + W_N^k \sum_{m=0}^{N/2-1} f_2[m] W_{N/2}^{mk}$$

- Have we divided the problem into subproblems?

16

Radix-2 FFT…

- Problem reduced to the sum of two DFTs of size $N/2$

$$X(k) = \sum_{m=0}^{N/2-1} f_1[m] W_{N/2}^{mk} + W_N^k \sum_{m=0}^{N/2-1} f_2[m] W_{N/2}^{mk}$$

$$= F_1(k) + W_N^k F_2(k),\ \ k = 0, \dots N-1$$

- Since $F_1(k)$ and $F_2(k)$ are $N/2$-point DFTs:

$$F_1\left(k + \frac{N}{2}\right) = F_1(k)$$

$$F_2\left(k + \frac{N}{2}\right) = F_2(k)$$

17

Radix-2 FFT…

- Exploiting periodicity of $F_1(k)$ and $F_2(k)$

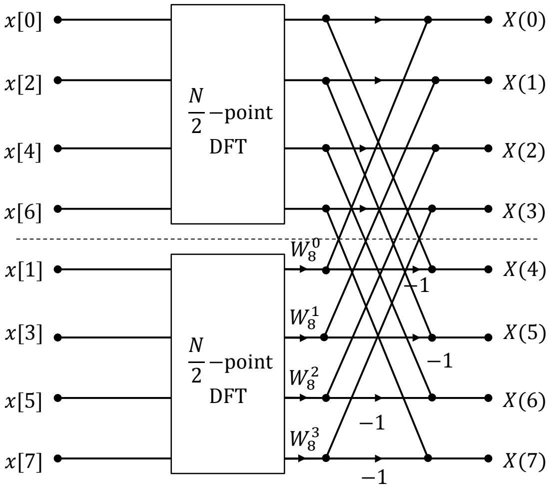
$$X(k) = F_1(k) + W_N^k F_2(k),\ \ k = 0, \dots N/2-1$$

$$X(k + N/2) = F_1(k) - W_N^k F_2(k),\ \ k = 0, \dots N/2-1$$

- Have we gained anything?

18

Radix-2 FFT…



- Number of multiplications: $2\left(\frac{N}{2}\right)^2 + \frac{N}{2}$

19

Radix-2 FFT…

- By splitting original problem into two we reduced the number of multiplications by a factor of two
- Why stop there?
- Repeat decimation for sequence $f_1[n]$

$$F_1(k) = V_{11}(k) + W_{\frac{N}{2}}^{k} V_{12}(k),\ k = 0, \ldots N/4 - 1$$

$$F_1(k + N/4) = V_{11}(k) - W_{\frac{N}{2}}^{k} V_{12}(k),\ k = 0, \ldots N/4 - 1$$

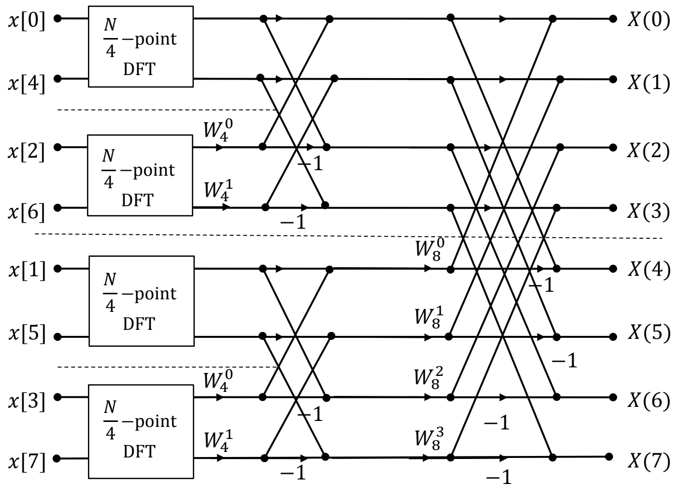
- … and for sequence $f_2[n]$

$$F_2(k) = V_{21}(k) + W_{\frac{N}{2}}^{k} V_{22}(k), \qquad k = 0, \ldots N/4 - 1$$

$$F_2(k + N/4) = V_{21}(k) - W_{\frac{N}{2}}^{k} V_{22}(k),\ k = 0, \ldots N/4 - 1$$

20

Radix-2 FFT…



- Number of multiplications: $4\left(\frac{N}{4}\right)^2 + \frac{N}{2} + \frac{N}{2} = \frac{N^2}{4} + N$

21

Radix-2 FFT…

- We can continue reducing the problem $\log_2 N = \nu$ times
- Total computational complexity
  - Number of multiplications: $\frac{N}{2}\log_2 N$
  - Number of additions: $N\log_2 N$

22

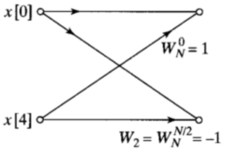
Radix-2 FFT…

- First stage is to evaluate a 2-point DFT?

$$X(k) = \sum_{n=0}^{1} x[n]e^{-\frac{j2\pi nk}{N=2}},\ k = 0, 1$$

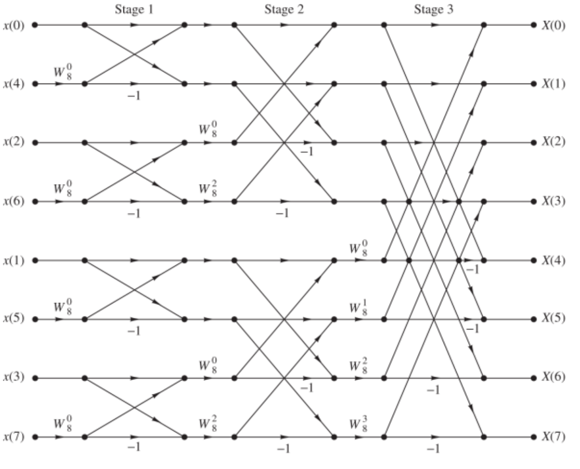
$$\Rightarrow X(0) = x[0] + x[1]$$

$$X(1) = x[0] - x[1]$$



Radix-2 FFT…

- Final flow-graph for 8-point decimation in time FFT



Radix-2 FFT…

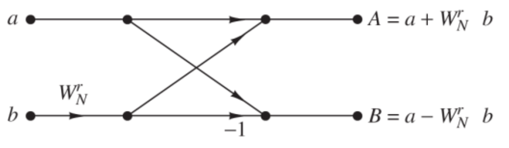
- Comparing direct-computation DFT and FFT

Number of points, N	Complex multiplications in Direct computation, N^2	Complex multiplications in FFT, $\frac{N}{2}\log_2 N$
4	16	4
8	64	12
16	256	32
32	1024	80
64	4096	192
128	16384	448
256	65536	1024
512	262144	2304
1024	1048576	5120

25

Radix-2 FFT…

- Memory requirements?
- Basic computation performed in each stage (butterfly)

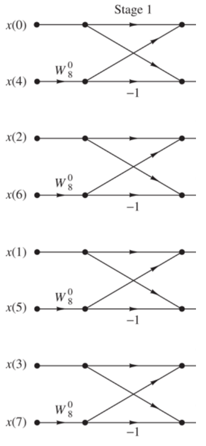


- Once butterfly operation performed, $(a, b) \longrightarrow (A, B)$, complex numbers A, B can be stored in same location as a, b
  - Computation *done in place*
  - Fixed amount of memory ($2N$ real numbers)

26

Radix-2 FFT…

- How to remember the order of the input to the FFT?
- Bit reversal - binary representation in reverse



Radix-2 IFFT

- How about the IFFT?
- Inverse DFT similar to DFT
  - Change terms W_N^k in the signal graph to W_N^{-k}
  - Divide the output of the graph by N.

Frequency-domain filtering

$$x[n] \longrightarrow \boxed{h[n]} \longrightarrow y[n] = \mathrm{IFFT}_N\{Y(k)\}$$

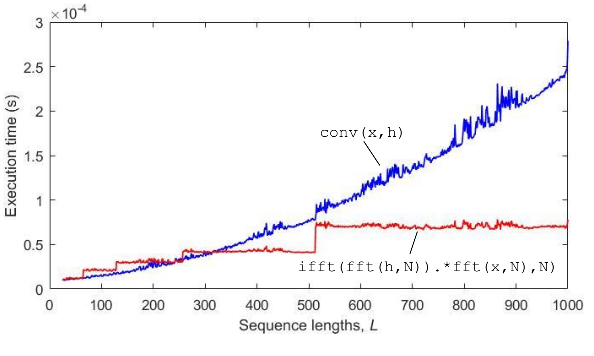
$$x[n] \updownarrow X(k) = \mathrm{FFT}_N\{x[n]\} \qquad h[n] \updownarrow H(k) = \mathrm{FFT}_N\{h[n]\} \qquad y[n] \updownarrow Y(k) = X(k)H(k)$$

- Order of complexity in terms of multiplications
  1. Transform to frequency domain FFT: $2N\log_2 N$
  2. Multiply frequency transforms: N
  3. Transform back to time domain IFFT: $N\log_2 N$

29

Frequency-domain filtering… (last week)

- Example 2: Assume that both $x[n]$ and $h[n]$ have length L



30

Summary

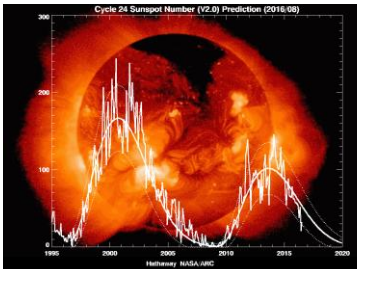
- Today we discussed:
  - Fast Fourier transform (FFT)

- Next time:
  - Stochastic processes

Example: Solar cycle

- Source: https://en.wikipedia.org/wiki/Solar_cycle

"The solar cycle or solar magnetic activity cycle is the nearly periodic 11-year change in the Sun's activity (including changes in the levels of solar radiation and ejection of solar material) and appearance (changes in the number of sunspots, flares, and other manifestations)"



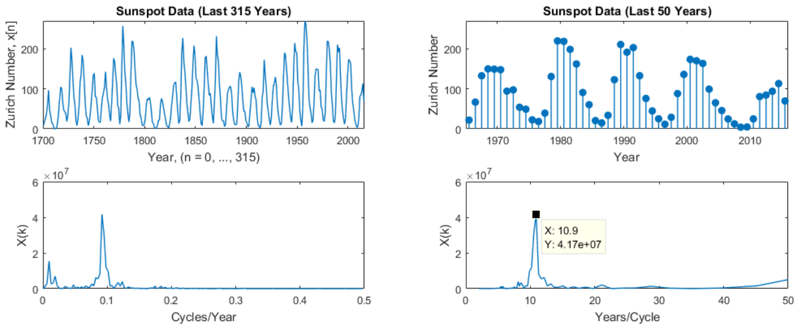
http://solarscience.msfc.nasa.gov/images/ssn_predict_l.gif

Example: Solar cycle...

- According to Wikipedia the solar cycle is “nearly periodic” with a period of approximately 11 years
- Can this be right?
- Matlab has data with 300 years of recorded sunspot numbers
- Sunspot data from 1700 until now can be downloaded at http://www.sidc.be/silso/datafiles and is available on BlackBoard

Example: Solar cycle...

http://www.sidc.be/silso/datafiles



- From figure we get a solar cycle (time between activity peaks) is approximately 11 years

Example: Solar cycle...

Matlab (Available on Blackboard)

```
load('sunspots.csv')
year = sunspots(:,1);
x = sunspots(:,2); % Relative spot number
X = fft(x);        % DFT of x
X(1) = 0; [];      % Remove DC level(sum of all numbers)
N = length(X);

% Plot some figs
figure
subplot(2,2,1);
plot(year,x) % last 316 years sunspot activity

subplot(2,2,2);
stem(year(end-50:end),x(end-50:end)) % last 51 years

freq = -0.5:1/N:(0.5-1/N);
subplot(2,2,3),
plot(freq,abs(fftshift(X)).^2) % frequency domain

period = 1./freq;
subplot(2,2,4),
plot(period,abs(fftshift(X)).^2); % show cycles/year
```

35

NTNU – Trondheim
Norwegian University of
Science and Technology

TTT4120 Digital Signal Processing
Fall 2018

Discrete Random Signals

Prof. Stefan Werner
stefan.werner@ntnu.no
Office B329

Lecture in course book*

- Proakis, Manolakis Digital Signal Processing, 4th Ed.
  - 1.2.4 Deterministic versus random signals
  - 12.1 Random signals, correlation functions, and power spectra
- A comprehensive overview of topics treated in the lecture, see "Introdukjon til statistisk signalbehandling" on Blackboard

*Level of detail is defined by lectures and problem sets

Preliminary question

- What is the Fourier transform of a sequence of coin flips?

Contents and learning outcomes

- Models
- Stochastic process
- Statistical averages
- Stationarity and wide-sense stationarity
- Ergodicity
- Power spectral density

Introduction

- Signal analysis and processing require a mathematical description of the signal itself, or so-called signal model
- Deterministic signals uniquely described by an explicit mathematical expression, well-defined rule or a table of data

$$x[n] = 2e^{-4n}, n \geq 0$$

$$x[n] = \sin 2\pi f n$$

- All past, present and future values of the signals are known precisely without any uncertainty

5

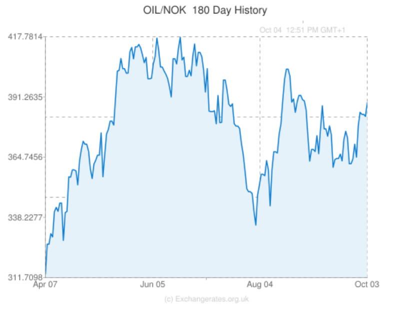
Introduction…

- Many signals cannot be described by explicit formulas
  - Speech signals, received noisy communication signals

  $\Rightarrow$ Signals evolve in time in an unpredictable manner
- Stochastic signal is a sequence of random numbers
  - Signal value at instant n unknown and modeled as a stochastic variable $X[n]$ with probability density function $p_X(x[n])$

6

Introduction…



- Models derived are usually of statistical nature
  - Find a suitable model describing the random signal
  - Estimate model parameters

7

Review stochastic variables

- First- and second-order moments
- Expected value: $m_X = E\{X\} = \int_{-\infty}^{\infty} x p_X(x)dx$
- Second-order moment: $E\{X^2\} = \int_{-\infty}^{\infty} x^2 p_X(x)dx$
- Variance: $\sigma_X^2 = E\{(X - m_X)^2\} = \int_{-\infty}^{\infty} (x - m_X)^2 p_X(x)dx$
  $= E\{X^2\} - m_X^2$
- Example: $X \sim N(m_X, \sigma_X^2) \Rightarrow p_X(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-m_X)^2}{2\sigma_X^2}}$

8

Review stochastic variables…

- Study of several stochastic variables requires joint density function, e.g., variables X_1, and X_2 described by $p_{X_1,X_2}(x_1,x_2)$
- Stochastic variables independent if

$$p_{X_1,X_2}(x_1,x_2) = p_{X_1}(x_1)p_{X_2}(x_2)$$

- Second-order moment:

$$E\{X_1X_2\} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x_1x_2p_{X_1,X_2}(x_1,x_2)dx$$

- Covariance: $\sigma^2_{X_1,X_2} = E\{(X_1 - m_{X_1})(X_2 - m_{X_2})\}$
$= E\{X_1X_2\} - m_{X_1}m_{X_2}$
- If $\sigma^2_{X_1,X_2} = 0 \Rightarrow X_1$ and X_2 are said to be uncorrelated

9

Stochastic process

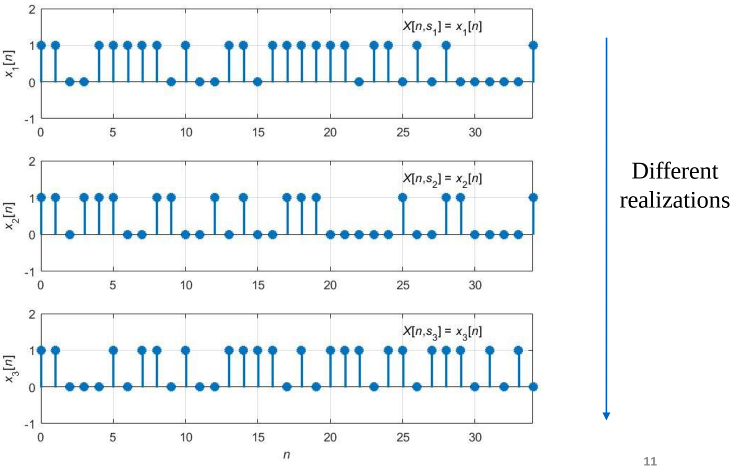
- Definition: A stochastic process is a family *or ensemble* of signals corresponding to every possible outcome of a certain signal measurement or experiment. Each signal in the ensemble is called a "realization" of the process.
- Notation: $X[n,S]$ is the ensemble of possible waveforms, where n represents time and $S = \{s_1, s_2 \dots\}$ represents the set of all possible functions
- Single waveform in ensemble denoted $x[n,s]$ or $x[n]$

- Example 1: Toss a coin 35 times and assign 1 for head and 0 for tail. Repeat the experiment.

10

Stochastic process…

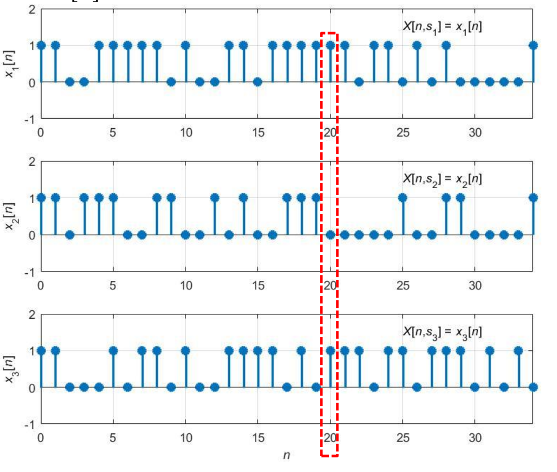
- Bernoulli process (coin flipping) with $p = 0.5$, $N = 35$



11

Stochastic process…

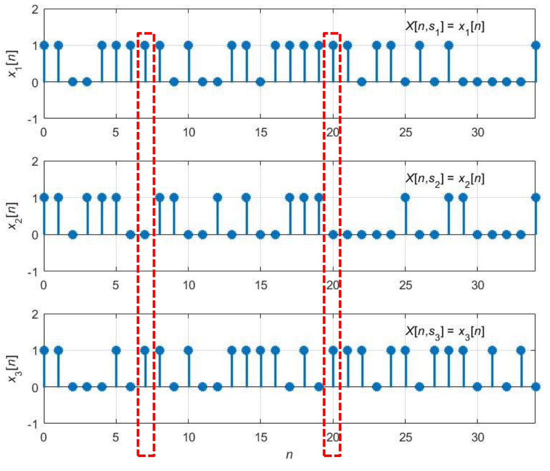
- Fixed time instant, e.g., $n = 20 \Rightarrow X(20, S)$ is a random variable defined by $p_{X[n]}(x)$



12

Stochastic process…

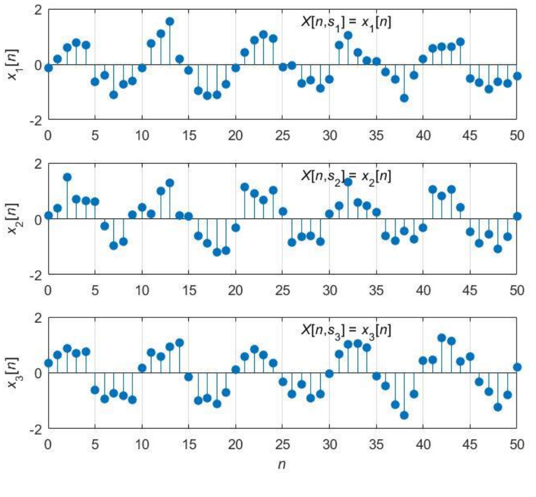
- Fixed time, e.g., $n_1 = 7$ and $n_2 = 20 \Rightarrow X(7, S)$ and $X(20, S)$ form a bivariate random vector defined by $p_{X[n_1],X[n_2]}(x_1, x_2)$



13

Stochastic process…

- Sinusoid with noise: $X(n) = \sin(2\pi f n) + W[n]$, $W[n] \sim N(0, \sigma_w^2)$



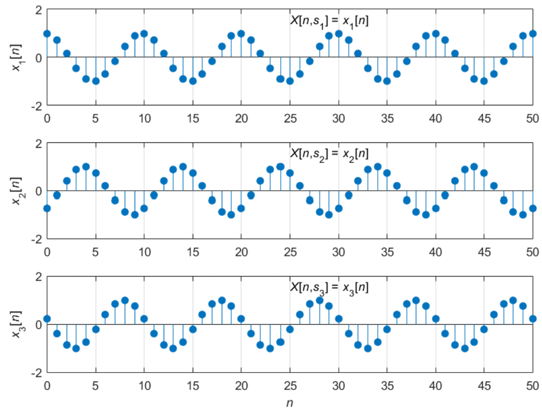
Matlab
```
nfigs = 4;
N = 51
n=(0:N-1);
x=sin(2*pi*0.1*n);

for i=1:nfigs,
   subplot(nfigs,1,i)
   w = 0.3*randn(1,N);
   stem(n,x+w),
end
```

14

Stochastic process…

- Sinusoid with random phase: $X(n) = \cos(2\pi f n + \Theta)$, $\Theta \sim \mathrm{U}[0,2\pi]$

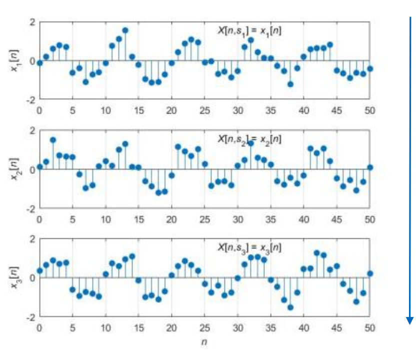


15

Statistical ensemble averages

- Definition: Mean of a stochastic process is the average of all realizations of the process

$$m_X[n] = E\{X[n]\} = \int_{-\infty}^{\infty} x p_{X[n]}(x)dx$$



Average of realizations

16

Statistical ensemble averages…

- Definition: Autocorrelation sequence of a stochastic process is the average product of a signal realization with a time-shifted version of itself
  $$\gamma_{XX}(n, n+l) = E\{X[n]X[n+l]\}$$
  $$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} x_1 x_2 p_{X[n]X[n+l]}(x_1 x_2) dx_1 dx_2$$
- Measure of temporal similarity of a single stochastic process
- Related autocovariance sequence:

$$c_{XX}(n, n+l) = E\{(X[n] - m_X[n])(X[n+l] - m_X[n+l]\}$$
$$= \gamma_{XX}(n, n+l) \ - m_X[n]m_X[n+l]$$

17

Statistical ensemble averages…

- Crosscorrelation sequence:

$$\gamma_{XY}(n, n+l) = E\{X[n]Y[n+l]\}$$

- Crosscovariance sequence:

$$c_{XY}(n, n+l) = E\{(X[n] - m_X[n])(Y[n+l] - m_Y[n+l]\}$$
$$= \gamma_{XY}(n, n+l) \ - m_X[n]m_Y[n+l]$$

18

Statistical ensemble averages…

- Example: $X(n) = \cos(2\pi f n + \Theta)$ with $\Theta \sim \mathrm{U}[0,2\pi]$
  Calculate mean and covariance sequences

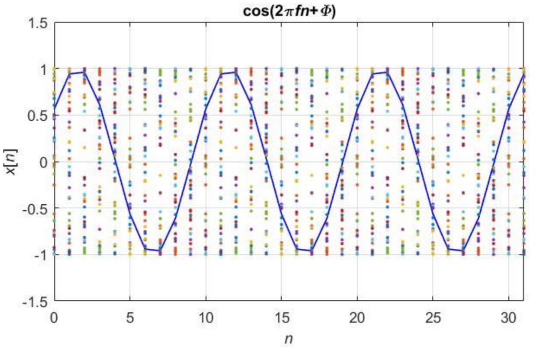
$$\mu_X[n] = \mathrm{E}[X(n)] = \mathrm{E}[\cos(2\pi f n + \Theta)]$$
$$= \int_0^{2\pi} \cos(2\pi f n + \theta) \frac{1}{2\pi} d\theta$$
$$= \frac{1}{2\pi} \sin(2\pi f n + \theta)\Big|_{\theta=0}^{2\pi} = 0$$

- Mean is constant for all *n*

19

Statistical ensemble averages…

- Example: $X(n) = \cos(2\pi f n + \Theta)$ with $\Theta \sim \mathrm{U}[0,2\pi]$
  50 realizations



Matlab
```
n     = 0:31;
nreal = 50;
f     = 0.1;
zeros(nreal,length(n));

for i = 1:nreal
  phi = 2*pi*rand;
  x(i,:) =cos(2*pi*f*n+phi);
end

figure
plot(n,x(1,:)), grid
hold on
for i=2:nreal
    plot(n,x(i,:),'.')
end
```

20

Statistical ensemble averages…

- Example: $X[n] = \cos(2\pi f n + \Theta)$ with $\Theta \sim \mathrm{U}[0,2\pi]$

  Calculate mean and covariance sequences

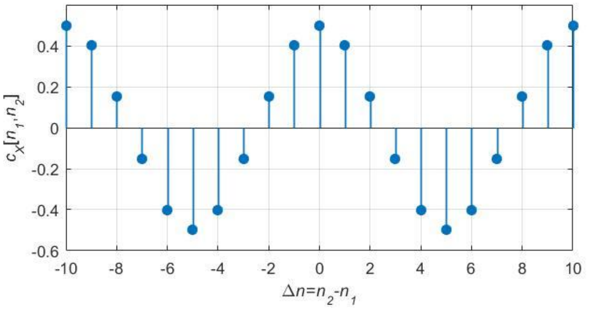
$$
\begin{aligned}
c_X[n, n+l] &= \mathrm{E}\big[X[n]X[n+l]\big] \\
&= \int_0^{2\pi} \cos(2\pi f n + \theta)\cos(2\pi f[n+l] + \theta)\,\frac{1}{2\pi}\,d\theta \\
&= \int_0^{2\pi} \left\{\frac{1}{2}\cos(2\pi f l\,) + \frac{1}{2}\cos(2\pi f[2n+l] + 2\theta)\right\}\frac{1}{2\pi}\,d\theta \\
&= \frac{1}{2}\cos(2\pi f l) + \frac{1}{8\pi}\sin(2\pi f[2n+l] + 2\theta)\Big|_{\theta=0}^{2\pi} \\
&= \frac{1}{2}\cos(2\pi f l)
\end{aligned}
$$

- Covariance sequence only depends on time difference l

21

Statistical ensemble averages…

- Example: $X[n] = \cos(2\pi f n + \Theta)$ with $\Theta \sim \mathrm{U}[0,2\pi]$
- Covariance sequence



- Covariance sequence only depends on $l = |n_2 - n_1|$

22

Stationarity

- A random process is said to be stationary in the strict sense if the statistical properties do not change over time

  ⇒ Joint density function $p_{X[n_1],\ldots,X[n_L]} = p_{X[n_1+l],\ldots,X[n_L+l]}$
- Set of samples can be shifted in time, *with each one being shifted by the same amount*, without affecting the joint PDF
- Weakly stationary (or wide-sense) process:

$$m_X[n] = m_X \text{ (a constant independent of } n\text{)}$$

$$\gamma_{XX}[n, n+l] = \gamma_{XX}[l] = \gamma_{XX}[-l] \text{ (depends only on shift } l\text{)}$$

23

Stationarity…

- Important example is white Gaussian noise (WGN) process
  - $W[n]$ are independent and zero-mean
  - Gaussian density function: $p_W(w) = \frac{1}{\sqrt{2\pi}\sigma_W} e^{-\frac{w^2}{2\sigma_W^2}}$

$$m_W = E\{W[n]\} = \int_{-\infty}^{\infty} w p_W(w) dw = 0$$

$$\sigma_W^2 = E\{W^2[n]\}$$

$$\gamma_{WW}[n, n+l] = E\{W[n]W[n+l]\} = \sigma_W^2 \delta[l]$$

- Samples are uncorrelated
- Is the GWN process wide-sense stationary? (Yes/No)

24

Ergodicity

- Random process characterized in terms of statistical averages
- In practice we observe data from a single realization
- Definition: An ergodic process is one where time averages are equal to ensemble averages

$\Longrightarrow$ We can estimate the parameters of a stationary random process through measurements

- Mean-ergodic process:

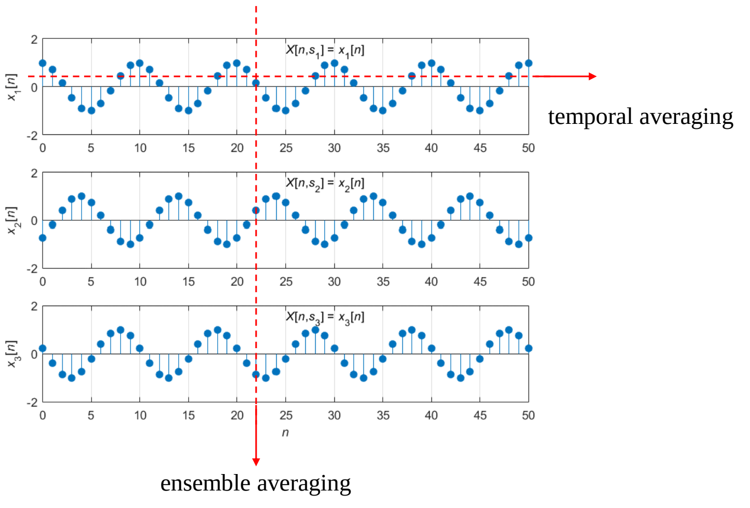
$$m_X = E\{X[n]\} = \lim_{N\to\infty} \frac{1}{2N+1} \sum_{-N}^{N} x[n]$$

- Correlation-ergodic process:

$$\gamma_{XX}[l] = E\{X[n]X[n+l]\} = \lim_{N\to\infty} \frac{1}{2N+1} \sum_{-N}^{N} x[n]x[n+l]$$

25

Ergodicity…



26

Ergodicity…

- Revisit the example: $X[n] = \cos(2\pi f n + \Theta)$ with $\Theta \sim \mathrm{U}[0,2\pi]$
- Time average mean of single realization $x[n]$ of $X[n]$

$$\hat{m}_x = \lim_{N\to\infty} \frac{1}{2N+1} \sum_{n=-N}^{N} \cos(2\pi f n + \theta)$$

$$= 0 \; (= m_X)$$

- Time average same as ensemble average

$\Longrightarrow X[n]$ is mean-ergodic

27

Ergodicity…

- Revisit the example: $X[n] = \cos(2\pi f n + \Theta)$ with $\Theta \sim \mathrm{U}[0,2\pi]$
- Time average autocorrelation of single realization $x[n]$ of $X[n]$

$$\hat{\gamma}_{xx}[l] = \lim_{N\to\infty} \frac{1}{2N+1} \sum_{n=-N}^{N} \cos(2\pi f n + \theta)\cos(2\pi f[n+l] + \theta)$$

$$= \lim_{N\to\infty} \frac{1}{2N+1} \sum_{n=-N}^{N} \frac{1}{2}\{\cos(2\pi f l) + \cos(2\pi f[2n+l] + 2\theta)\}$$

$$= \frac{1}{2}\cos(2\pi f l) \;\; (= \gamma_{XX}[l])$$

- Time average same as ensemble average

$\Longrightarrow X[n]$ is correlation-ergodic

Matlab

```
x = cos(2*pi*0.1*(0:10000)+2*pi*rand);
autocorr(x)
```

28

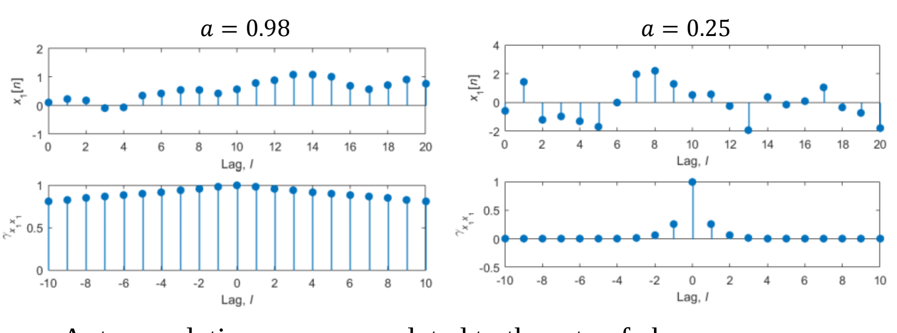
Power density spectrum

- For the rest of the course we assume wide-sense stationary processes that are both mean-ergodic and correlation-ergodic

- A stationary stochastic process is an infinite-energy signal $\Longrightarrow$ its Fourier transform does not exist
- How to measure frequency content in a random signal?
- Autocorrelation sequence measures similarity in time domain

Power density spectrum…

- Example: $X[n] = aX[n-1] + W[n]$, $W[n] \sim N(0, \sigma_w^2)$



- Autocorrelation sequence related to the rate of change
  - Realization varies slowly, $\gamma_{XX}[l]$ decays slowly
  - Realization varies rapidly, $\gamma_{XX}[l]$ decays rapidly

Power density spectrum…

- Autocorrelation sequence $\gamma_{XX}[l]$ reflects variability (frequency content) of random process
- We define the Fourier transform pair

$$\Gamma_{XX}(f) = \sum_{l=-\infty}^{\infty} \gamma_{XX}[l] e^{-j2\pi f l}$$

$$\gamma_{XX}[l] = \int_{-0.5}^{0.5} \Gamma_{XX}(f)\, e^{j2\pi f l} df$$

- Power density spectrum $\Gamma_{XX}(f)$ represents $\gamma_{XX}[l]$ in frequency
- Name power density spectrum comes from relation

$$P_X = E\{X^2[n]\} = \gamma_{XX}[0] = \int_{-0.5}^{0.5} \Gamma_{XX}(f)\, df$$

31

Power density spectrum…

- Revisiting the case of white noise sequence $W[n]$
  - Zero mean: $m_W = 0$
  - Uncorrelated samples: $\gamma_{WW}[l] = \sigma_W^2 \delta[l]$

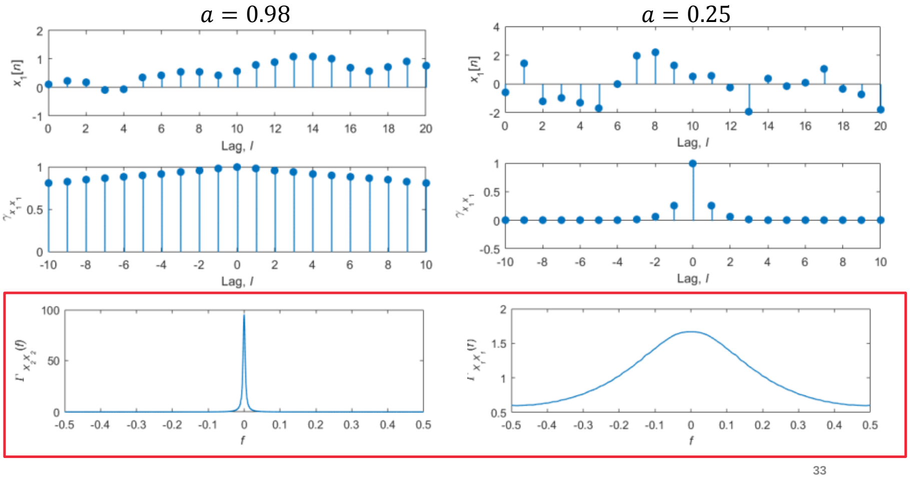
$$\Longrightarrow \Gamma_{WW}(f) = \sum_{l=-\infty}^{\infty} \gamma_{XX}[l] e^{-j2\pi f l} = \sigma_W^2 \text{ (constant } \forall f)$$

- Contains all frequencies (frequency-flat), hence the name white

32

Power density spectrum…

- Revisiting example: $X[n] = aX[n-1] + W[n]$, $W[n] \sim N(0, \sigma_W^2)$



33

Power density spectrum…

- Power density spectrum (PDS)
  - Frequency-domain interpretation of random signals
  - Information on how signal power is distributed in frequency
  - Fourier transform of the auto-correlation sequence

- Autocorrelation sequence (ACS)
  - Information of self-similarity of random signals in time-domain
  - Slow decay $\Longrightarrow$ most power is concentrated at low frequencies
  - Fast decay $\Longrightarrow$ power in high-frequency components
  - Inverse Fourier transform of the power density spectrum

34

Summary

- Today we discussed:
  - Stochastic processes and their statistical averages
  - Stationarity and wide-sense stationarity
  - Ergodicity
  - Power density spectrum

- Next time:
  - Filtering of stochastic processes (LTI systems)

35

NTNU – Trondheim
Norwegian University of
Science and Technology

TTT4120 Digital Signal Processing
Fall 2019

Filtering of Discrete Random Signals

Prof. Stefan Werner
stefan.werner@ntnu.no
Office B329

Lecture in course book*

- Proakis, Manolakis Digital Signal Processing, 4th Ed.
  - 12.1 Random signals, correlation functions, and power spectra
  - 5.3 Correlation functions and spectra at the output of LTI systems
- A comprehensive overview of topics treated in the lecture, see "Introdukjon til statistisk signalbehandling" on Blackboard

*Level of detail is defined by lectures and problem sets

Contents and learning outcomes

- Filtering of stochastic signals in time-domain
- Frequency-domain interpretation
- Example: Power density spectrum of AR(1) process

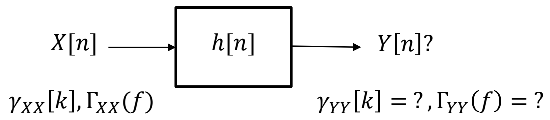
Filtering of stochastic signals

$$X[n] \longrightarrow \boxed{h[n]} \longrightarrow Y[n]?$$

$$\gamma_{XX}[k], \Gamma_{XX}(f) \qquad \gamma_{YY}[k] = ?, \Gamma_{YY}(f) = ?$$

- Let $X[n]$ be a wide-sense stationary process
- Linear time-invariant filter described by $h[n]$, $H(z)$, or $H(f)$
- Can we relate output signal $Y[n]$ to input signal $X[n]$?

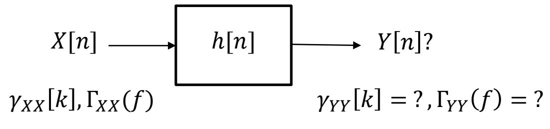
Filtering of stochastic signals…



- Consider a single realization $x[n]$ of process $X[n]$
- Each input realization $x[n]$ produces output realization $y[n]$

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

Filtering of stochastic signals…



- Since $x[n]$ is a realization of $X[n]$, $y[n]$ is a realization of the random process $Y[n]$

$$Y[n] = \sum_{k=-\infty}^{\infty} h[k]X[n-k]$$

- We want to relate the statistical properties of output process $Y[n]$ to the statistical properties of input process $X[n]$

$$E\{Y[n]\} = ?,\ \gamma_{YY}[k] = ?, \Gamma_{YY}(f) = ?$$

Filtering of stochastic signals…

- Expected value of output process $Y[n]$:

$$m_Y = E\{Y[n]\} = E\{\sum_{k=-\infty}^{\infty} h[k]X[n-k]\}$$

$$= \sum_{k=-\infty}^{\infty} h[k]E\{X[n-k]\}$$

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

$$= m_X \sum_{k=-\infty}^{\infty} h[k]e^{j2\pi 0 k}$$

$$= m_X H(0)$$

Filtering of stochastic signals…

$$X[n] \longrightarrow \boxed{h[n]} \longrightarrow Y[n]$$

- Example: WSS signal $X[n]$ with mean $m_X = 3$ is filtered by LTI system $H(f) = \frac{1}{1-0.5e^{-j2\pi f}}$. Compute the mean of output process $Y[n]$.

$$m_Y = E\{Y[n]\} = m_X H(0) = \frac{3}{1-0.5e^{-j2\pi 0}} = 6$$

Filtering of stochastic signals…

- Autocorrelation sequence of output process $Y[n]$:

$$\gamma_{YY}[l] = E\{Y[n]Y[n+l]\} = h[-l] * h[l] * \gamma_{XX}[l]$$
$$= r_{hh}[l] * \gamma_{XX}[l]$$

Filtering of stochastic signals…

- Proof:

$$\gamma_{YY}[l] = E\left\{\left(\sum_{i=-\infty}^{\infty} h[i]X[n-i]\right)\left(\sum_{j=-\infty}^{\infty} h[j]X[n+l-j]\right)\right\}$$
$$= \sum_{i=-\infty}^{\infty}\sum_{j=-\infty}^{\infty} h[i]h[j]E\{X[n-i]X[n+l-j]\}$$
$$= \sum_{i=-\infty}^{\infty}\sum_{j=-\infty}^{\infty} h[i]h[j]\gamma_{XX}[l-j+i]$$
$$= \sum_{i=-\infty}^{\infty} h[i]\sum_{j=-\infty}^{\infty} h[j]\gamma_{XX}[(l+i)-j]$$
$$= \sum_{i=-\infty}^{\infty} h[i]g[l+i] \text{ with } g[l] = h[l] * \gamma_{XX}[l]$$
$$= \sum_{k=-\infty}^{\infty} h[-k]g[l-k] = h[-l] * g[l]$$

Filtering of stochastic signals…

- Power density spectrum of $Y[n]$:

$$\Gamma_{YY}(f) = \mathcal{F}\{\gamma_{YY}[k]\} = \mathcal{F}\{r_{hh}[k] * \gamma_{XX}[k]\}$$

$$= \mathcal{F}\{r_{hh}[k]\}\mathcal{F}\{\gamma_{XX}[k]\}$$

$$= S_{hh}(f)\Gamma_{XX}(f) = |H(f)|^2\Gamma_{XX}(f)$$

- The output PDS is the input PDS multiplied by the magnitude-squared of the frequency response!

11

Filtering of stochastic signals…

- Example: What is the output power $E\{Y^2[n]\}$ of a linear system $H(f)$ when the input $X[n]$ is WGN?

$$\Gamma_{YY}(f) = |H(f)|^2\Gamma_{XX}(f) = |H(f)|^2\mathcal{F}\{\sigma_X^2\,\delta[n]\}$$

$$= \sigma_X^2|H(f)|^2$$

$$\sigma_Y^2 = E\{Y^2[n]\} = \gamma_{YY}[0]$$

$$= \int_{-0.5}^{0.5}\Gamma_{YY}(f)e^{j2\pi f 0}df = \sigma_X^2\int_{-0.5}^{0.5}|H(f)|^2df$$

12

Filtering of stochastic signals…

- Crosscorrelation sequence of processes $Y[n]$ and $X[n]$:

$$\gamma_{YX}[l] = E\{(\sum_{k=-\infty}^{\infty} h[k]X[n-k])X[n-l]\}$$

$$= \sum_{k=-\infty}^{\infty} h[k]E\{X[n-k]X[n-l]\}$$

$$= \sum_{k=-\infty}^{\infty} h[k]\gamma_{XX}[l-k] = h[l] * \gamma_{XX}[l]$$

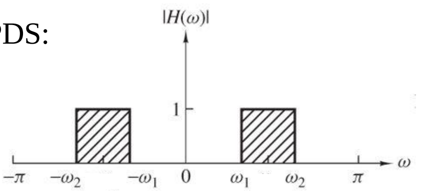
- Crosscorrelation spectrum of processes $Y[n]$ and $X[n]$:

$$\Gamma_{YX}(f) = \mathcal{F}\{h[k] * \gamma_{XX}[l]\} = \mathcal{F}\{h[k]\}\mathcal{F}\{\gamma_{XX}[k]\}$$

$$= H(f)\Gamma_{XX}(f)$$

Frequency domain interpretation

Interpretation PDS:



- Filter with narrow frequency band ⇒ $\Gamma_{YY}(f) = |H(f)|^2\Gamma_{XX}(f)$
- Average output power

$$E\{Y^2[n]\} = \gamma_{YY}[0] = \int_{-0.5}^{0.5} \Gamma_{YY}(f)df$$

$$= \int_{-\frac{\omega_2}{2\pi}}^{\frac{-\omega_1}{2\pi}} |H(f)|^2\Gamma_{XX}(f)df + \int_{\frac{\omega_1}{2\pi}}^{\frac{\omega_2}{2\pi}} |H(f)|^2\Gamma_{XX}(f)df$$

- Area under $\Gamma_{XX}(f)$ for $\omega_1 \leq |\omega| \leq \omega_2$ is the average power for that frequency band ⇒ $\Gamma_{XX}(f)$ can be viewed as density function for power in frequency domain

Example: PDS of AR(1) process

- Example: Calculate $\Gamma_{XX}(f)$ for the random process

$$X[n] = aX[n-1] + W[n],\ W[n] \sim N(0, \sigma_W^2)$$

$$\text{WGN: } E\{W[n]W[n+l]\} = \sigma_W^2 \delta[l]$$

- Approach 1: Calculate $\gamma_{XX}[l]$ and take its Fourier transform
- Approach 2: Use the idea of LTI systems

15

Example: PDS of AR(1) process…

- Approach 1: Calculate $\gamma_{XX}[l]$ and take its Fourier transform
- Consider lag $l = 0$:

$$E\{X[n]X[n]\} = \gamma_{XX}[0] = E\{(aX[n-1] + W[n])^2\}$$

$$= E\{a^2X^2[n-1] + 2aX[n-1]W[n] + W^2[n]\}$$

$$= a^2E\{X^2[n-1]\} + 2aE\{X[n-1]W[n]\} + E\{W^2[n]\}$$

$$= a^2\gamma_{XX}[0] + 0 + \sigma_W^2$$

$$\Rightarrow \boxed{\gamma_{XX}[0] = \frac{1}{1-a^2}\sigma_W^2}$$

16

Example: PDS of AR(1) process…

- Consider lag $l \geq 1$:

$$\begin{aligned}\gamma_{XX}[l] &= E\{X[n]X[n+l]\}\\ &= E\{X[n](aX[n+l-1]+W[n+l])\}\\ &= E\left\{X[n]\left(a^l X[n]+\sum_{j=0}^{l-1} a^j W[n+l-j]\right)\right\}\\ &= a^l E\{X^2[n]\}+\sum_{j=0}^{l-1} a^j E\{X[n]W[n+l-j]\}\\ &= a^l \gamma_{XX}[0]+0=\frac{a^l}{1-a^2}\sigma_W^2\end{aligned}$$

- Symmetry $\gamma_{XX}[l]=\gamma_{XX}[-l]$ provides the final answer

$$\boxed{\gamma_{XX}[l]=\frac{a^{|l|}}{1-a^2}\sigma_W^2}$$

17

Example: PDS of AR(1) process…

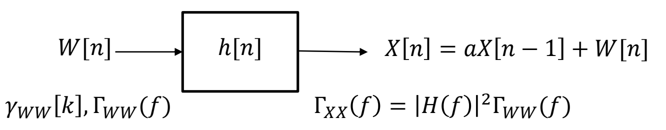
- Take the Fourier transform:

$$\begin{aligned}\Gamma_{XX}(f) &= \mathcal{F}\{\gamma_{XX}[l]\}\\ &= \sum_{l=-\infty}^{\infty}\frac{a^{|l|}\sigma_W^2}{1-a^2}e^{-j2\pi f l}\\ &= \frac{\sigma_W^2}{1-a^2}\sum_{l=-\infty}^{\infty} a^{|l|}e^{-j2\pi f l}\\ &= \cdots = \frac{\sigma_W^2}{1-a^2}\frac{1-a^2}{\left|1-ae^{-j2\pi f}\right|^2}\\ &= \frac{\sigma_W^2}{\left|1-ae^{-j2\pi f}\right|^2}\end{aligned}$$

18

Example: PDS of AR(1) process…

- Approach 2: Model the problem with a linear system

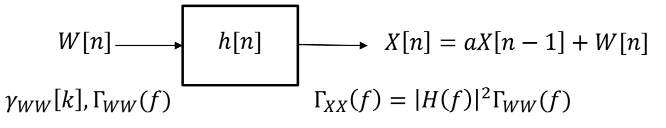


- WGN process $W[n] \Rightarrow \gamma_{WW}[k] = ?, \Gamma_{WW}(f) = ?$
- What is the system frequency response $H(f)$?

19

Example: PDS of AR(1) process…

- Find the frequency response $H(f)$:



- For any realization $x[n]$: $X(z) = az^{-1}X(z) + W(z)$

$$\Longrightarrow H(z) = \frac{1}{1-az^{-1}}$$

- Consequently we obtain the frequency response

$$H(f) = \frac{1}{1 - ae^{-j2\pi f}}$$

20

Example: PDS of AR(1) process…

- Power density spectrum of $X[n]$:

$$W[n] \longrightarrow \boxed{h[n]} \longrightarrow X[n] = aX[n-1] + W[n]$$

$\gamma_{WW}[k], \Gamma_{WW}(f)$ $\qquad$ $\Gamma_{XX}(f) = |H(f)|^2 \Gamma_{WW}(f)$

$$\Gamma_{XX}(f) = |H(f)|^2 \Gamma_{WW}(f)$$

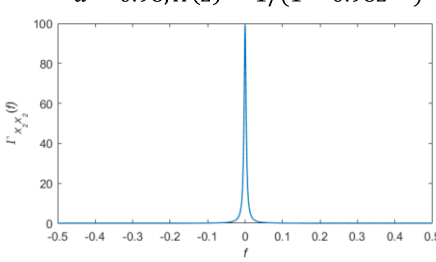
$$\boxed{= \frac{1}{\left|1 - ae^{-j2\pi f}\right|^2} \sigma_W^2}$$

21

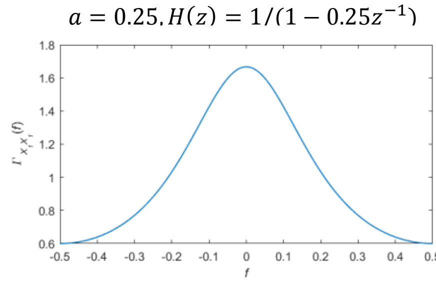
Example: PDS of AR(1) process…

- Power density spectrum of $X[n]$:

$a = 0.98, H(z) = 1/(1 - 0.98z^{-1})$



$a = 0.25, H(z) = 1/(1 - 0.25z^{-1})$



Matlab
```
[H,W]=freqz(1,[1 -0.98],(-pi:pi/500:pi))
plot(W/2/pi,(1-0.98^2)*abs(H).^2)
```

22

Summary

- Today we discussed:
  - Linear filtering of stochastic processes
  - Power density and cross-spectra
- Next:
  - Basics of parameter estimation

NTNU – Trondheim
Norwegian University of
Science and Technology

TTT4120 Digital Signal Processing
Fall 2019

Estimation Basics and Periodogram

Prof. Stefan Werner
stefan.werner@ntnu.no
Office B329

Department of Electronic Systems

Lecture in course book*

- Proakis, Manolakis Digital Signal Processing, 4th Ed.
  - 12.1 Random signals, correlation functions, and power spectra
  - 14.1.2 Estimation of the autocorrelation and power spectrum of random signals: The periodogram
  - 14.1.3 The use of DFT in power spectrum estimation
  - 14.2.1 The Bartlett method: Averaging periodograms
- A comprehensive overview of topics treated in the lecture, see "Statistisk basert signalbehandling" on Blackboard

*Level of detail is defined by lectures and problem sets

Contents and learning outcomes

- Basics of estimation theory
  - Simple example: estimating the mean
  - Properties of good estimators
- Estimating the autocorrelation sequence
- Periodogram: crude estimate of the PDS

Introduction

- Autocorrelation sequence of a random signal $X[n]$

$$\gamma_{XX}[l] = E\{X[n]X[n+l]\}$$

- Power spectrum density of a random signal $X[n]$

$$\Gamma_{XX}(f) = \sum_{l=-\infty}^{\infty} \gamma_{XX}[l] e^{-j2\pi f l}$$

- Statistical averages require knowledge of *all realizations* or *an infinitely long realization from an ergodic process*
- In practice, access to a single realization of finite duration
- Can we still estimate statistical quantities and to what accuracy?

Basics of estimation theory

- Our problem becomes to estimate an unknown quantity, θ, (e.g., a statistical average) from a discrete-time waveform or a data-set
- We have the N-point data set $\boldsymbol{x} = \{x[0], x[1], \dots, x[N-1]\}$, which is a realization of a random process containing information on θ
- Determine θ based on the data, or define an estimator

$$\hat{\theta} = g(\boldsymbol{x}) = g(x[0], x[1], \dots, x[N-1])$$

where $g(\cdot)$ is some function

- Since $x[n]$ is a realization of $X[n]$, $\hat{\theta}$ is related to random variable

$$\widehat{\Theta} = g(\boldsymbol{X}) = g(X[0], X[1], \dots, X[N-1])$$

Basics of estimation theory…

- How good is a particular estimator? How good can *any* estimate be?
- How to measure goodness of an estimate?

Simple example: estimating the mean

- Example 1: Estimate the mean m_X from an N-point realization of i.i.d. sequence $X[n] \sim N(m_X, \sigma_W^2)$
- Based on the N-point data set $\{x[0], x[1], \ldots, x[N-1]\}$, we would like to estimate m_X. Reasonable to estimate m_X as

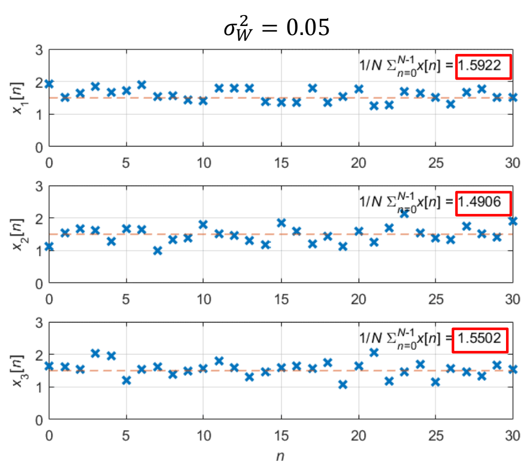
$$\widehat{m}_X = \frac{1}{N}\sum_{n=0}^{N-1} x[n]$$

(which can be seen as an outcome of $\widehat{M}_X = \frac{1}{N}\sum_{n=0}^{N-1} X[n]$)

- How close is $\widehat{m}_X$ to m_X and what is the influence of N?

Simple example: estimating the mean…

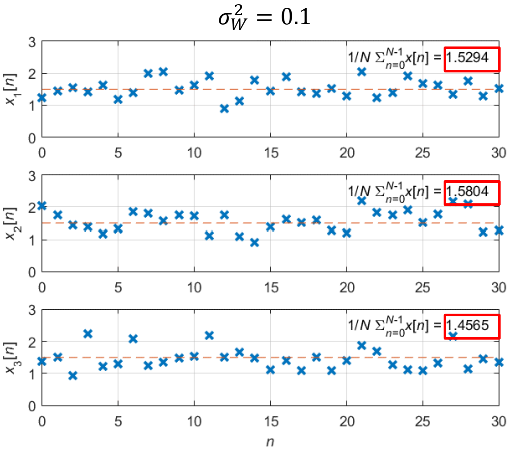
- Three different realizations of $X[n] \sim N(1.5, \sigma_W^2 = 0.05)$




```
Matlab
N = 31;
n = (0:N-1);
w = randn(1,N);
x = 1.5 + w;
plot(n,x,'x')
m_hat = mean(x)
```

Simple example: estimating the mean…

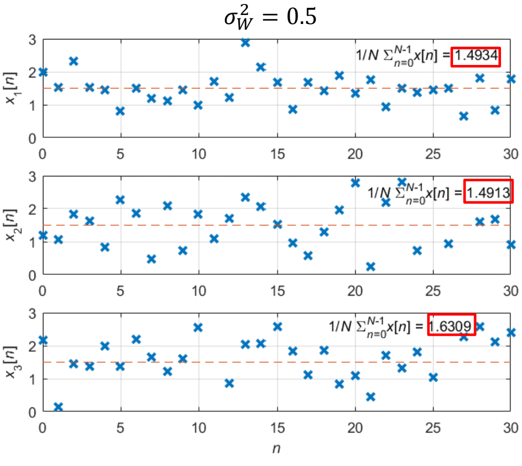
- Three different realizations of $X[n] \sim N(1.5, \sigma_W^2 = 0.1)$



9

Simple example: estimating the mean…

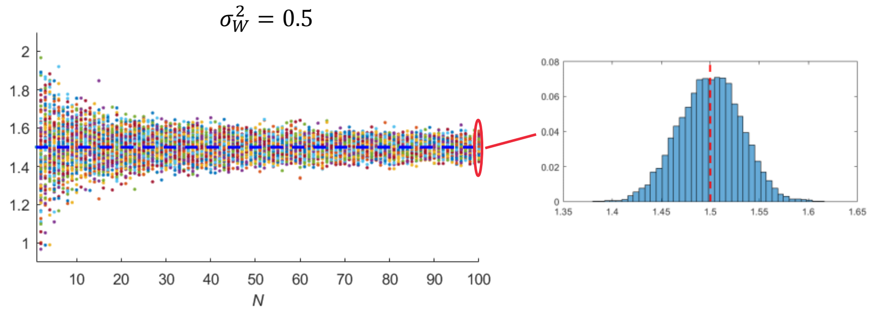
- Three different realizations of $X[n] \sim N(1.5, \sigma_W^2 = 0.5)$



10

Simple example: estimating the mean…

- Varying number of data points N used for the estimation



- Each point (for a fixed N) corresponds to the estimate from a single realization

Simple example: estimating the mean…

- Observations from this simple example
  - Estimate depends on the realization (data available)
  - True value m_X is the mid point to all realizations of $\widehat{M}_X$
  - Variability of estimates increases with uncertainty
  - Variability of estimate across realizations decreases with N
  - Estimate approaches true value as N increases

- Let us calculate the mean and variance of $\widehat{M}_X$

Simple example: estimating the mean…

- Mean value of estimate

$$E\{\widehat{M}_X\} = E\left\{\frac{1}{N}\sum_{n=0}^{N-1} X[n]\right\}$$

$$= \frac{1}{N}\sum_{n=0}^{N-1} E\{X[n]\}$$

$$= \frac{1}{N}\sum_{n=0}^{N-1} m_X = m_X$$

- On the average we get the true parameter

13

Simple example: estimating the mean…

- Variance of estimate

$$\sigma_{\widehat{M}_X}^2 = E\left\{\left(\widehat{M}_X - E\{\widehat{M}_X\}\right)^2\right\}$$

$$= E\left\{\left(\frac{1}{N}\sum_{n=0}^{N-1} X[n] - m_X\right)^2\right\}$$

$$= \frac{1}{N^2} E\{\sum_{n=0}^{N-1}(X[n] - m_X)^2\}$$

$$= \frac{\sigma_W^2}{N}$$

- Variance of estimate goes to zero as N increases

14

Properties of good estimators

- An unbiased estimator provides the true value on average

$$m_{\widehat{\Theta}} = E\{\widehat{\Theta}\} = \theta$$

- A weaker requirement is asymptotic unbiasedness

$$\lim_{N\to\infty} m_{\widehat{\Theta}} = \lim_{N\to\infty} E\{\widehat{\Theta}\}$$
$$= \lim_{N\to\infty} E\{g(\boldsymbol{X})\} = \theta$$

- Small variance $\sigma_{\hat{\theta}}^2$: The estimates $\hat{\theta}$ are close to the true value θ irrespectively of the realization $\boldsymbol{x}$
- Variance decreasing for an increased number of observations, N

15

Properties of good estimators…

- An estimator is said to be consistent whenever, the estimate approaches the true value as $N \to \infty$, i.e.,

$$\lim_{N\to\infty} m_{\widehat{\Theta}} = \theta$$

$$\lim_{N\to\infty} \sigma_{\widehat{\Theta}}^2 = 0$$

- The simple averager in previous example is a consistent estimator

16

Estimation of autocorrelation

- Goal is to estimate the PDS of a signal from a single observation of the signal over a finite time interval
- The PDS is related to the autocorrelation sequence as

$$\Gamma_{XX}(f) = \sum_{l=-\infty}^{\infty} \gamma_{XX}[l] e^{-j2\pi f l}$$

with $\gamma_{XX}[l] = E\{X[n]X[n+l]\}$

- Given an N-point realization $\boldsymbol{x} = \{x[0]\ x[1] \dots x[N-1]\}$, we would like to acquire a good estimate $\hat{\gamma}_{XX}[l]$ of $\gamma_{XX}[l]$

17

Estimation of autocorrelation…

- Approach 1: For lag l we can compute $N-|l|$ products. Compute the average over available products, i.e.,

$$\hat{\gamma}'_{XX}[l] = \frac{1}{N-|l|} \sum_{n=0}^{N-|l|-1} x[n]x[n+|l|]$$

- Is this estimator consistent?

1. $E\left\{\frac{1}{N-|l|} \sum_{n=0}^{N-|l|-1} X[n]X[n+|l|]\right\} = \gamma_{XX}[l]$
2. $\lim_{N\to\infty} \mathrm{var}\left\{\frac{1}{N-|l|} \sum_{n=0}^{N-|l|-1} X[n]X[n+|l|]\right\} = 0$

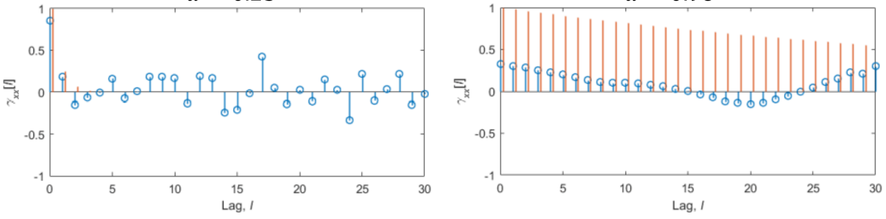
Yes!

18

Estimation of autocorrelation…

- Estimate $\gamma_{XX}[l]$ from a realization of $X[n] = aX[n-1] + W[n]$
  $0 \le n \le N-1 = 30$, $W[n] \sim N(0, \sigma_W^2)$

$a = 0.25$ $\quad$ $a = 0.98$

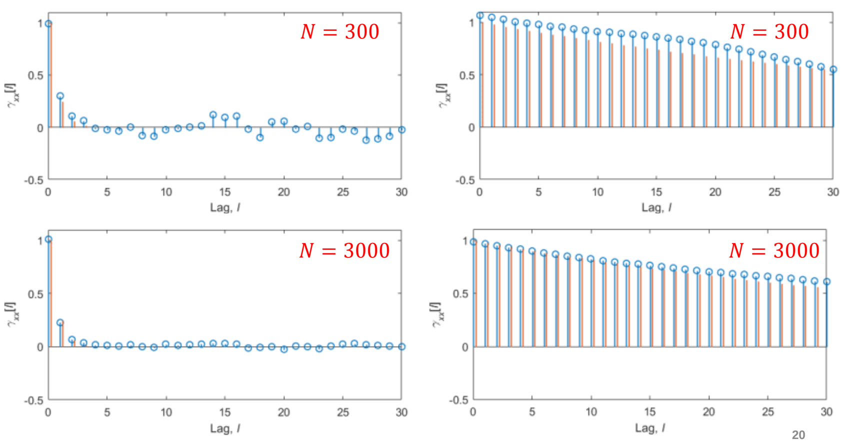


- Estimate: $\hat{\gamma}_{XX}[l] = \frac{1}{N-l} \sum_{n=0}^{N-1-l} x[n]x[n+l]\,, l = 0,1,\ldots,N-1$
- As lag l increases, less products to average over $\Rightarrow$ large errors
- Maximum lag to be estimated, $l_{\max}$, chosen such that $l_{\max} \ll N$

Estimation of autocorrelation…

- Increase the sample size: $N = 300$, and $N = 3000$

$a = 0.25$ $\quad$ $a = 0.98$



Estimation of autocorrelation…

```
Matlab
N = 30;
lmax = 21; % Max lag to compute < N
l = (-lmax:lmax);
a = 0.98;
varw = (1 - a^2);

% True autocorrelation function
gammaxx = varw/(1-a^2)*a.^abs(l);

% Generate data:
w = randn(N,1);
x = filter(1,[1 -a],sqrt(varw)*w);

% Compute ACF:
[gammaxx_est,lags] = xcorr(x,'unbiased',lmax);
stem(lags,gammaxx_est), hold on
stem(lags+.2,gammaxx,'Marker', 'none')
xlim([0 lmax])
```


Estimation of autocorrelation…

- Approach 2: For lag we can compute $N - |l|$ products. Compute the average over available products but normalize with N, i.e.,

$$\hat{\gamma}_{XX}[l] = \frac{1}{N}\sum_{n=0}^{N-|l|-1} x[n]x[n+|l|]$$

- Properties of this estimator

  1. Biased for $l \neq 0$
  2. Consistent for $|l| \ll N$

$$\lim_{N\to\infty} E\left\{\frac{1}{N}\sum_{n=0}^{N-|l|-1} X[n]X[n+|l|]\right\} = \gamma_{XX}[l]$$
$$\lim_{N\to\infty} \operatorname{var}\left\{\frac{1}{N-|l|}\sum_{n=0}^{N-|l|-1} X[n]X[n+|l|]\right\} = 0$$

Estimation of autocorrelation…

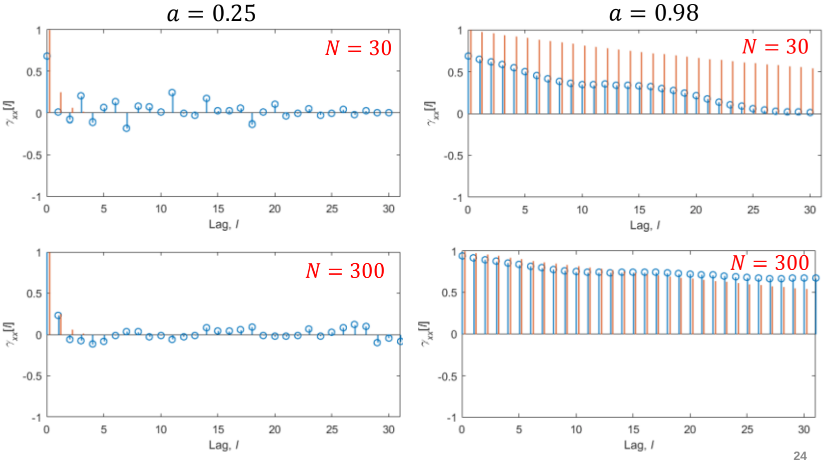
- Computing the bias

$$E\left\{\frac{1}{N}\sum_{n=0}^{N-|l|-1} X[n]X[n+|l|]\right\}$$
$$= \frac{1}{N}\sum_{n=0}^{N-|l|-1} E\{X[n]X[n+|l|]\}$$
$$= \frac{N-|l|}{N}\gamma_{XX}[l] = \left(1-\frac{|l|}{N}\right)\gamma_{XX}[l]$$
$$= w_B[l]\,\gamma_{XX}[l]$$

- Bias term disappears for fixed l when $N \to \infty$
- Triangular (Bartlett) window deemphasizes effects at lags $l \approx N$ $\Rightarrow$ lower variance

Estimation of autocorrelation…

- Revisit previous example: $N = 300$, and $N = 3000$



Estimation of autocorrelation…

```
Matlab
N = 30;
lmax = 21; % Max lag to compute < N
l = (-lmax:lmax);
a = 0.98;
varw = (1 - a^2);

% True autocorrelation function
gammaxx = varw/(1-a^2)*a.^abs(l);

% Generate data:
w = randn(N,1);
x = filter(1,[1 -a],sqrt(varw)*w);

% Compute ACF:
[gammaxx_est,lags] = xcorr(x,'biased',lmax);
stem(lags,gammaxx_est), hold on
stem(lags+.2,gammaxx,'Marker', 'none')
xlim([0 lmax])
```

25

Estimation of autocorrelation…

Comparing the of the two different estimators

- Approach 1: $\hat{\gamma}'_{XX}[l] = \frac{1}{N-|l|}\sum_{n=0}^{N-|l|-1} x[n]x[n+|l|]$
  - Consistent estimator (unbiased for any N and l)
- Approach 2: $\hat{\gamma}_{XX}[l] = \frac{1}{N}\sum_{n=0}^{N-|l|-1} x[n]x[n+|l|]$
  - Consistent estimator (asymptotically unbiased)
  - Lower variance than Approach 1
  - More effective for PDS estimation
  - Guarantees positive semidefinite autocorrelation sequence

26

Periodogram: crude estimate of the PDS

- We have the Fourier pair: $\hat{\gamma}_{XX}[l] \overset{\mathcal{F}}{\leftrightarrow} \Gamma_{XX}(f)$
- Periodogram:

$$\hat{\Gamma}_{XX}(f) = \sum_{l=-\infty}^{\infty} \hat{\gamma}_{XX}[l] e^{-j2\pi f l}$$

where $\hat{\gamma}_{XX}[l] = \frac{1}{N} \sum_{n=0}^{N-|l|-1} x[n]x[n+|l|]$

- Is the periodogram a good estimator for the PDS of $X[n]$?

27

Periodogram: crude estimate of the PDS

- With this choice of estimator, the periodogram becomes

$$\hat{\Gamma}_{XX}(f) = \frac{1}{N} \left| \sum_{n=0}^{N-1} x[n] e^{-j2\pi f n} \right|^2 = \frac{1}{N} |Y(f)|^2$$

where $Y(f)$ is the Fourier transform of

$$y[n] = \begin{cases} x[n] & \text{for } 0 \leq n \leq N-1 \\ 0 & \text{otherwise} \end{cases}$$

28

Periodogram: crude estimate of the PDS

- To see this, let us rewrite $\hat{\gamma}_{XX}[l]$

$$\hat{\gamma}_{XX}[l] = \begin{cases} \frac{1}{N}\sum_{n=0}^{N-|l|-1} x[n]x[n+|l|] & |l| < N \\ 0 & \text{otherwise} \end{cases}$$

$$= \frac{1}{N}\sum_{n=0}^{N-|l|-1} y[n]y[n+|l|]$$

$$= \frac{1}{N}\sum_{n=-\infty}^{\infty} y[n]y[n+|l|]$$

$$= \frac{1}{N}\gamma_{YY}[l]$$

Periodogram: crude estimate of the PDS

- Putting the pieces together: take the DTFT of both sides

$$\hat{\Gamma}_{XX}(f) = \mathcal{F}\{\hat{\gamma}_{XX}[l]\}$$

$$= \mathcal{F}\left\{\frac{1}{N}\gamma_{YY}[l]\right\} = S_{YY}(f)$$

$$= \mathcal{F}\left\{\frac{1}{N}y[-l] * y[l]\right\}$$

$$= \frac{1}{N}|Y(f)|^2 = \frac{1}{N}\left|\underbrace{\sum_{n=-\infty}^{\infty} y[n]e^{-j2\pi fn}}_{\sum_{n=0}^{N-1} x[n]e^{-j2\pi fn}}\right|^2$$

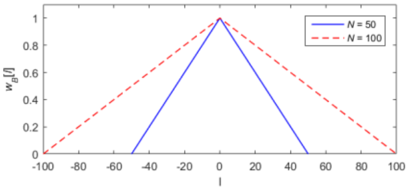
- Periodogram is obtained by taking the *N*-point DTFT of sequence $\{x[n]\}_{n=0}^{N-1}$

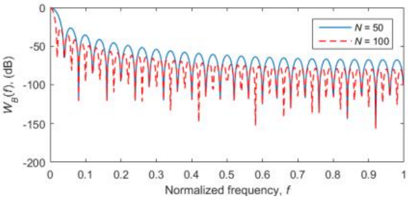
Periodogram: crude estimate of the PDS…

- Expected value of periodogram (bias)

$$E\{\hat{\Gamma}_{XX}(f)\} = E\left\{\mathcal{F}\{\hat{\gamma}_{XX}[l]\}\right\} = \mathcal{F}\left\{E\{\hat{\gamma}_{XX}[l]\}\right\}$$
$$= \mathcal{F}\{w_B[l]\gamma_{XX}[l]\} = W_B(f) * \Gamma_{XX}(f)$$

where $W_B(f)$ is the Fourier transform of the Bartlett window





$$w_B[l] = \begin{cases} 1 - \frac{|l|}{N}, & |l| \le N-1 \\ 0, & \text{otherwise} \end{cases}$$

$$W_B(f) = \frac{1}{N}\left(\frac{\sin \pi f N}{\sin \pi f}\right)^2$$

31

Periodogram: crude estimate of the PDS…

- Convolution with $W_B(f)$ results in spectrum spreading
  - Increasing window length reduces spectral leakage
- Frequency resolution is adequate for most situations
- Periodogram is asymptotically unbiased
- Peridogram is not a consistent estimator
  - That is, variance of estimate does not approach 0 as $N \to \infty$
  - For a Gaussian process $\mathrm{var}\{\hat{\Gamma}_{XX}(f)\} \ge \Gamma_{XX}^2(f)$

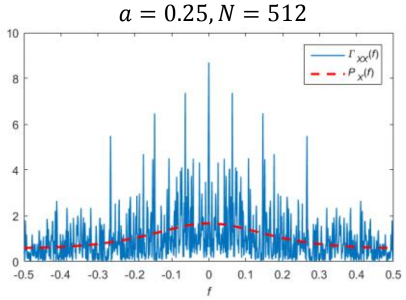
∴ Periodogram is not a good estimator for the PDS

32

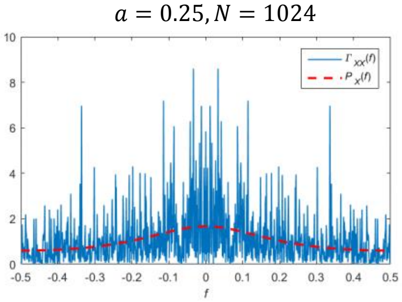
Periodogram: crude estimate of the PDS…

- Estimate $\Gamma_{XX}(f)$ from a realization of $X[n] = aX[n-1] + W[n]$
  $0 \leq n \leq N-1$, $W[n] \sim N(0, \sigma_W^2)$

$a = 0.25, N = 512$



$a = 0.25, N = 1024$



- Increasing N does not reduce variance

33

Improving the periodogram

- Use a different window function
  - Hamming, Kaiser
  - Reduces the spectral leakage and spread
  - Leads to a modified periodogram
- Take average of several periodograms
  - Split data into several blocks of length M
  - Compute periodogram for each block
  - Average over all computed periodograms
- Nonparametric methods: no assumptions made on how data were generated

34

Averaging periodogram: Bartlett method

$$\ldots, \underbrace{x[0], x[1], \ldots x[M-1]}_{M}, \underbrace{x[M], x[N+1], \ldots, x[2M-1]}_{M}, \underbrace{x[2M], x[2M+1] \ldots}_{M}.$$

- Break up $x[n]$ into K non-overlapping segments of length M

$$x_i[n] = x[n+iM], \quad \begin{matrix} i = 0, 1, \ldots, K-1 \\ n = 0, 1, \ldots, M-1 \end{matrix}$$

- Calculate the periodogram for each segment

$$\hat{\Gamma}_{XX}^{(i)}(f) = \frac{1}{M}\left|\sum_{n=0}^{M-1} x_i[n] e^{-j2\pi f n}\right|^2, i = 0, 1, \ldots, K-1$$

- Average the periodograms for the K segments

$$\hat{\Gamma}_{XX}^{B}(f) = \frac{1}{K}\sum_{n=0}^{K-1} \hat{\Gamma}_{XX}^{(i)}(f)$$

35

Averaging periodogram: Bartlett …

- Statistical properties
  - Mean value

$$E\{\hat{\Gamma}_{XX}^{B}(f)\} = \frac{1}{K}\sum_{n=0}^{K-1} E\left\{\hat{\Gamma}_{XX}^{(i)}(f)\right\} = W_B(f) * \Gamma_{XX}(f)$$

  - Variance

$$\mathrm{var}\{\hat{\Gamma}_{XX}^{B}(f)\} = \frac{1}{K}\mathrm{var}\{\hat{\Gamma}_{XX}(f)\}$$

- Bartlett window

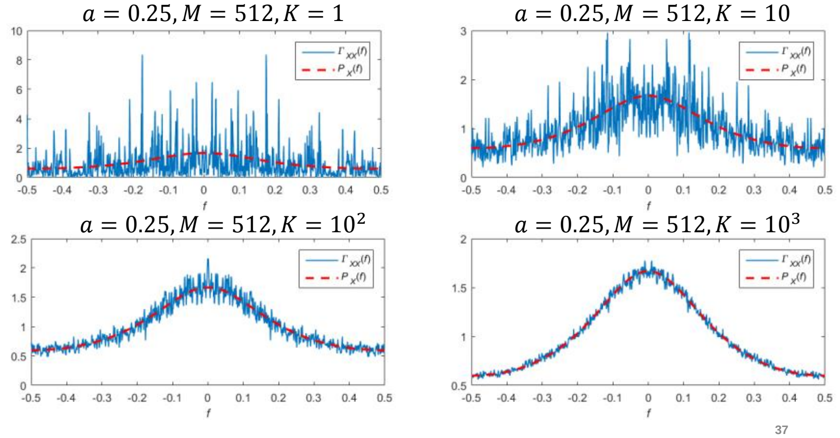
$$w_B[n] = \begin{cases} 1 - \frac{|m|}{M}, & |m| \le M-1 \\ 0, & \text{otherwise} \end{cases}$$

$$W_B(f) = \frac{1}{M}\left(\frac{\sin \pi f M}{\sin \pi f}\right)^2$$

36

Averaging periodogram: Bartlett …

- Estimate $\Gamma_{XX}(f)$ from a realization of $X[n] = aX[n-1] + W[n]$
  $0 \leq n \leq N-1$, $W[n] \sim N(0, \sigma_w^2)$



37

Summary

- Today we discussed:
  - Basics of estimation theory
  - Nonparametric power density spectrum (PDS) estimation
- Next:
  - Parametric PDS estimation

38

NTNU – Trondheim
Norwegian University of Science and Technology

TTT4120 Digital Signal Processing
Fall 2020

Modeling of Stochastic Processes: Parametric Spectral Estimation

Prof. Stefan Werner
stefan.werner@ntnu.no
Office B329

Department of Electronic Systems

1

Lecture in course book*

- Proakis, Manolakis Digital Signal Processing, 4th Ed.
  - 12.2 Innovations representation of stationary random processes
  - 14.3 Parametric methods for spectral estimation
- A comprehensive overview of topics treated in the lecture, see "Introdukjon til statistisk signalbehandling" on Blackboard

*Level of detail is defined by lectures and problem sets

2

2

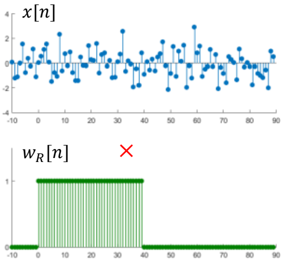
Contents and learning outcomes

- Non-parametric versus parametric models
- Innovations representation
- Rational power spectra: AR, MA, ARMA
- AR models and Yule-Walker equations

3

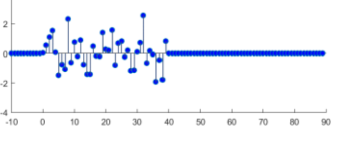
3

Nonparametric PSD estimation



=

$$\hat{\gamma}_{XX}[l] = \frac{1}{N}\sum_{n=0}^{N-|l|-1} x[n]x[n+|l|] = w_B[l]\,\gamma_{XX}[l]$$



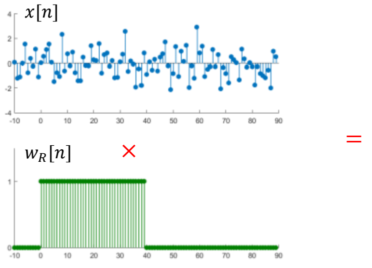
$$\hat{\Gamma}_{XX}(f) = \mathcal{F}\{\hat{\gamma}_{XX}[l]\} = \frac{1}{N}\left|\sum_{n=0}^{N-1} x[n]e^{-j2\pi f n}\right|^2$$

- Nonparametric power spectrum estimation: Assume that stochastic process $X[n]$ is wide-sense stationarity and ergodic
  - \+ Simple and easy to compute using the FFT_N
  - – Requires long data records for good frequency resolution
  - – Spectral leakage due to windowing ⇒ Can mask weak signals

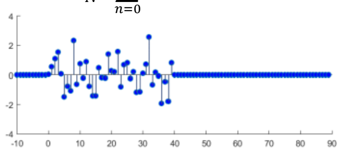
4

4

Nonparametric PSD estimation…



$$\hat{\gamma}_{XX}[l] = \frac{1}{N}\sum_{n=0}^{N-|l|-1} x[n]x[n+|l|] = w_B[l]\,\gamma_{XX}[l]$$



$$\hat{\Gamma}_{XX}(f) = \mathcal{F}\{\hat{\gamma}_{XX}[l]\} = \frac{1}{N}\left|\sum_{n=0}^{N-1} x[n]e^{-j2\pi f n}\right|^2$$

- Basic limitations of nonparametric spectrum estimation
  - Inherent assumption that $\hat{\gamma}_{XX}[l] = 0$ for some $l \geq N$
  - Inherent assumption that the data is periodic with N

5

5

Parametric PSD estimation

- Consider methods that can extrapolate the values of the autocorrelation function for $l \geq N$

$$\hat{\gamma}_{XX}[l], |l| \leq N-1 \Longrightarrow \hat{\gamma}_{XX}[l], |l| \geq N$$

- Requires *a priori* information on how data signal is generated
- A parametric model for the signal generation is constructed
  - Sufficient to find the values of the model parameters
  - Often provides us with a better description of the process, whenever the model is close to reality
  - Eliminates need for windowing and assumption that $\hat{\gamma}_{XX}[m] = 0$ for some $m \geq N$

6

6

Parametric PSD estimation…

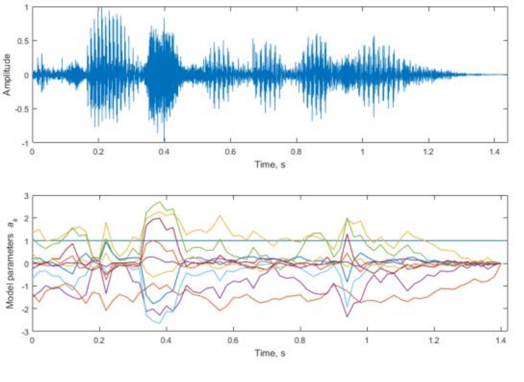
- Parametric modeling does not provide an exact representation
  - approximation characterized by a few parameters
  - enhanced spectral resolution: especially for finite data records, e.g., due to time-variant or transient phenomena
  - Efficient signal compression (e.g., LPC of speech)
- Different parametric models
  - rational models
  - all-pole models lead to linear equation systems

7

7

Parametric PSD estimation…

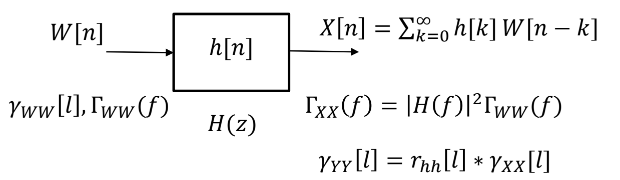
- Example: Model a speech signal (parameters change every 20ms)



8

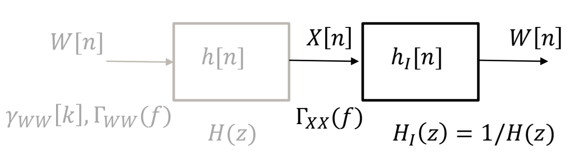
8

Innovations representations



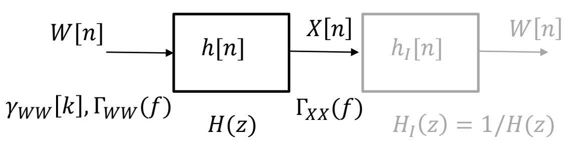
- Wide-sense stationary random processes can be represented as the output of a causal and causually invertible system excited by a white noise process
- This representation is called the Wold representation

Innovations representations…



- Consequently, a WSS random process can be represented by the output of the inverse system, which is a white process
  - a random process can be transformed into a white process by passing $X[n]$ through a linear filter
  - $H_I(z)$ is a called a whitening filter

Innovations representations…

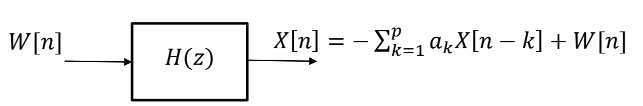


- Restrict our attention to cases where the PSD of $X[n]$ is rational
$$\Gamma_{XX}(f) = \frac{\sigma_W^2 B(z)B(z^{-1})}{A(z)A(z^{-1})}\bigg|_{z=e^{j2\pi f}} \text{ or } H(z) = \frac{B(z)}{A(z)}$$
- $H(z)$ is causal stable and minimum-phase $\Rightarrow$ $H_I(z)$ is also causal stable and minimum phase
- By knowing $H(z)$, described by a few parameters, we can go from the statistical properties of $W[n]$ to $X[n]$, and vice versa

11

11

Model types: AR process



- For an autoregressive (AR) process, filter $H(z)$ has only poles,

$$H(z) = \frac{1}{A(z)} = \frac{1}{1+\sum_{k=1}^{p} a_k z^{-k}}$$

and is described in time domain as

$$X[n] = -\sum_{k=1}^{p} a_k X[n-k] + W[n]$$

12

12

Model types: MA process

$$W[n] \longrightarrow \boxed{H(z)} \longrightarrow X[n] = \sum_{k=0}^{q} b_k W[n-k]$$

- For a moving average (MA) process, filter $H(z)$ has only zeros,

$$H(z) = B(z) = \sum_{k=0}^{q} b_k z^{-k}$$

and is described in time domain as

$$X[n] = \sum_{k=0}^{q} b_k W[n-k]$$

13

13

Model types: ARMA process

$$W[n] \longrightarrow \boxed{H(z)} \longrightarrow X[n] = -\sum_{k=1}^{p} a_k X[n-k] + \sum_{k=0}^{q} b_k W[n-k]$$

- For an autoregressive moving average (ARMA) process, filter $H(z)$ has both zeros and poles,

$$H(z) = \frac{B(z)}{A(z)} = \frac{\sum_{k=0}^{q} b_k z^{-k}}{1+\sum_{k=1}^{p} a_k z^{-k}}$$

and is described in time domain as

$$X[n] = -\sum_{k=1}^{p} a_k X[n-k] + \sum_{k=0}^{q} b_k W[n-k]$$

14

14

Parameter estimation

$$W[n] \longrightarrow \boxed{H(z)} \longrightarrow X[n] = -\sum_{k=1}^{p} a_k X[n-k] + \sum_{k=0}^{q} b_k W[n-k]$$

- The models are described by a few parameters: $\{a_k\}$, $\{b_k\}$, and σ_W^2
- Parameter values are unknown ⇒ need to estimate them from $X[n]$
- We need to find the parameters such that the model "resembles," or is "close" to, the true process
- Restrict our study to AR models

15

15

Parameter estimation…

$$W[n] \longrightarrow \boxed{H(z)} \longrightarrow X[n] = -\sum_{k=1}^{p} a_k X[n-k] + W[n]$$

- Restrict our study to AR models
  - can model processes characterized by sharp peaks in the spectrum
  - other types of spectra can be modeled by increasing the model order, i.e., the number of filter coefficients
  - suitable for many practical physical processes, e.g., speech

16

16

Statistical description of AR(p) process

- Model the problem with linear system

$$W[n] \longrightarrow \boxed{h[n]} \longrightarrow X[n] = -\sum_{k=1}^{p} a_k X[n-k] + W[n]$$

- WGN process $W[n]$: $E\{W[n]W[n-l]\} = \sigma_W^2 \delta[l]$
- The AR(p) process has p filter coefficients $\{a_k\}_{k=1}^{p}$
- Find a relation between $\gamma_{XX}[l]$ and coefficients a_k and σ_W^2

17

17

Statistical description of AR(p) process

- Autocorrelation function $\gamma_{XX}[l]$:

$$\begin{aligned}
\gamma_{XX}[l] &= E\{X[n]X[n-l]\} \\
&= E\left\{\left(-\sum_{k=1}^{p} a_k X[n-k] + W[n]\right)X[n-l]\right\} \\
&= E\left\{-\sum_{k=1}^{p} a_k X[n-k]X[n-l] + W[n]X[n-l]\right\} \\
&= -\sum_{k=1}^{p} a_k E\{X[n-k]X[n-l]\} + E\{W[n]X[n-l]\} \\
&= -\sum_{k=1}^{p} a_k \gamma_{XX}[l-k] + \gamma_{WX}[l]
\end{aligned}$$

- Take a closer look at the crosscorrelation term $\gamma_{WX}[l]$

18

18

Statistical description of AR(*p*) process…

- Crosscorrelation term $\gamma_{WX}[l]$:

$$\gamma_{WX}[l] = E\{W[n]X[n-l]\} = E\{W[n+l]X[n]\}$$
$$= E\left\{W[n+l]\left(-\sum_{k=1}^{p} a_k X[n-k] + W[n]\right)\right\}$$
$$= E\left\{-\sum_{k=1}^{p} a_k W[n+l]X[n-k] + W[n+l]W[n]\right\}$$
$$= -\sum_{k=1}^{p} a_k E\{W[n+l]X[n-k]\} + E\{W[n+l]W[n]\}$$
$$= 0 + \sigma_W^2 \delta[l]$$

- Crosscorrelation term only take non-zero value at lag $l = 0$

Statistical description of AR(*p*) process…

- Autocorrelation function $\gamma_{XX}[l]$ for an AR(p) process:

$$\boxed{\gamma_{XX}[|l|] = -\sum_{k=1}^{p} a_k \gamma_{XX}[|l|-k] + \sigma_W^2 \delta[l]}$$

- Discussion: How can we use the above equation to find $\gamma_{XX}[l]$ for all l, and what knowledge is required?

Statistical description of AR(*p*) process…

- Autocorrelation function $\gamma_{XX}[l]$ for an AR(p) process:

$$\boxed{\gamma_{XX}[|l|] = -\sum_{k=1}^{p} a_k \gamma_{XX}[|l|-k] + \sigma_W^2 \delta[l]}$$

- Discussion: How can we use the above equation to find…
  - Crosscorrelation function $\gamma_{XX}[l]$ is specified for all l when
    1) the p filter coefficients $\{a_k\}_{k=1}^{p}$, and;
    2) the p+1 first values of $\gamma_{XX}[l]$, i.e., $\gamma_{XX}[0], \gamma_{XX}[1], \dots, \gamma_{XX}[p]$
- How to find model parameters to the model $\{a_k\}_{k=1}^{p}$ given $\gamma_{XX}[l]$

21

21

Statistical description of AR(*p*) process…

$$\boxed{\gamma_{XX}[|l|] = -\sum_{k=1}^{p} a_k \gamma_{XX}[|l|-k] + \sigma_W^2 \delta[l]}$$

- Linear equation system ($\gamma_{XX}[l] = \gamma_{XX}[-l]$):

$l = 1$: $-\gamma_{XX}[1] = a_1\gamma_{XX}[0] + a_2\gamma_{XX}[1] + \cdots + a_p\gamma_{XX}[p-1]$

$l = 2$: $-\gamma_{XX}[2] = a_1\gamma_{XX}[1] + a_2\gamma_{XX}[0] + \cdots + a_p\gamma_{XX}[p-2]$

$l = 3$: $-\gamma_{XX}[3] = a_1\gamma_{XX}[2] + a_2\gamma_{XX}[1] + \cdots + a_p\gamma_{XX}[p-3]$

$$\vdots$$

$l = p$: $-\gamma_{XX}[p] = a_1\gamma_{XX}[p-1] + a_2\gamma_{XX}[p-2] + \cdots + a_p\gamma_{XX}[0]$

22

22

Statistical description of AR(p) process…

- Linear equation system in matrix form (Yule-Walker equations):

$$\underbrace{\begin{bmatrix} \gamma_{XX}[0] & \gamma_{XX}[1] & \dots & \gamma_{XX}[p-1] \\ \gamma_{XX}[1] & \gamma_{XX}[0] & \dots & \gamma_{XX}[p-2] \\ \vdots & \ddots & \ddots & \vdots \\ \gamma_{XX}[p-1] & \gamma_{XX}[p-2] & \dots & \gamma_{XX}[0] \end{bmatrix}}_{\boldsymbol{\Gamma}_{XX}} \underbrace{\begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix}}_{\boldsymbol{a}} = -\underbrace{\begin{bmatrix} \gamma_{XX}[1] \\ \gamma_{XX}[2] \\ \vdots \\ \gamma_{XX}[p] \end{bmatrix}}_{\boldsymbol{\gamma}_{XX}}$$

and

$$\sigma_W^2 = \gamma_{XX}[0] + \sum_{k=1}^{p} a_k \gamma_{XX}[k]$$

- Solve for coefficients

$$\boldsymbol{a} = -\boldsymbol{\Gamma}_{XX}^{-1} \boldsymbol{\gamma}_{XX}$$

23

23

Statistical description of AR(p) process…

- Sanity check: Model process $X[n] = -aX[n-1] + W[n]$ using an AR(2) process, and find parameters a_1 and a_2

AR(1) process: $\gamma_{XX}[l] = \sigma_W^2 \frac{(-a)^{|l|}}{1-a^2}$, $|l| \geq 0$

Yule-Walker equations:

$$\underbrace{\begin{bmatrix} \gamma_{XX}[0] & \gamma_{XX}[1] \\ \gamma_{XX}[1] & \gamma_{XX}[0] \end{bmatrix}}_{\boldsymbol{\Gamma}_{XX}} \underbrace{\begin{bmatrix} a_1 \\ a_2 \end{bmatrix}}_{\boldsymbol{a}} = -\underbrace{\begin{bmatrix} \gamma_{XX}[1] \\ \gamma_{XX}[2] \end{bmatrix}}_{\boldsymbol{\gamma}_{XX}}$$

$$\begin{bmatrix} 1 & -a \\ -a & 1 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = -\begin{bmatrix} -a \\ a^2 \end{bmatrix} \Rightarrow \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = -\begin{bmatrix} 1 & -a \\ -a & 1 \end{bmatrix}^{-1} \begin{bmatrix} -a \\ a^2 \end{bmatrix}$$

Solve for coefficients: $a_1 = ?$, $a_2 = ?$

24

24

PSD of an AR(p) process

- Once we have filter coefficients $\{a_k\}_{k=1}^{p}$ we can compute the PSD

$$W[n] \longrightarrow \boxed{h[n]} \longrightarrow X[n] = -\sum_{k=1}^{p} a_k X[n-k] + W[n]$$

$\gamma_{WW}[k] = \sigma_W^2 \delta[k]$,
$\Gamma_{WW}(f) = \sigma_W^2$

$$\Gamma_{XX}(f) = |H(f)|^2 \Gamma_{WW}(f) = |H(f)|^2 \sigma_W^2$$

- Frequency response of filter

$$H(f) = \frac{1}{1+\tilde{A}(f)}, \text{ with } \tilde{A}(z) = \sum_{k=1}^{p} a_k z^{-k}$$

- Finally we obtain the PSD as

$$\Gamma_{XX}(f) = \frac{\sigma_W^2}{\left|1+\tilde{A}(f)\right|^2}$$

PSD of an AR(p) process…

- Example: Model $X[n] = W[n] - bW[n-1]$, $W[n] \sim N(0,1)$ using an AR(2) process, and find parameters a_1 and a_2

- Autocorrelation: $\gamma_{XX}[l] = (1+b^2)\delta[l] - b\delta[l-1] - b\delta[l+1]$
- Yule-Walker equations:

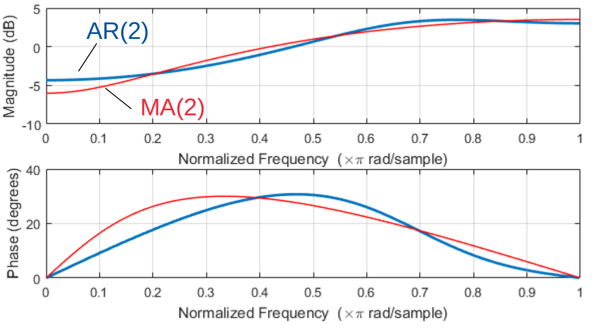
$$\begin{bmatrix} 1+b^2 & -b \\ -b & 1+b^2 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = -\begin{bmatrix} -b \\ 0 \end{bmatrix} \Rightarrow \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} 1+b^2 & -b \\ -b & 1+b^2 \end{bmatrix}^{-1} \begin{bmatrix} b \\ 0 \end{bmatrix}$$

Solve for coefficients: $a_1 = \frac{b(1+b^2)}{b^2(1+b^2)+1}$, $a_2 = \frac{b^2}{b^2(1+b^2)+1}$

$$\sigma_W^2 = \gamma_{XX}[0] + a_1\gamma_{XX}[1] + a_2\gamma_{XX}[2]$$

PSD of an AR(p) process…

- Example: Model $X[n] = W[n] - 0.5W[n-1]$, $W[n] \sim N(0,1)$

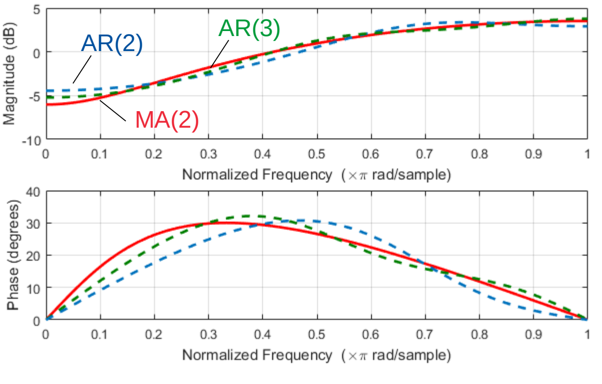


27

27

PSD of an AR(p) process…

- Example: Model $X[n] = W[n] - 0.5W[n-1]$, $W[n] \sim N(0,1)$



28

28

PSD of an AR(*p*) process…

- Example: Power spectrum of an AR process is given by

$$\Gamma_{XX}(f) = \frac{\sigma_W^2}{|A(f)|^2} = \frac{25}{\left|1 - e^{-j2\pi f} + \frac{1}{2}e^{-j4\pi f}\right|^2}$$

where σ_W^2 is the variance of the input sequence.

a) Determine the difference equation that generates the AR process when the excitation is white noise

b) Determine the system function for the whitening filter

29

29

PSD of an AR(*p*) process…

- Only access to finite-length realization, $x[n]$, of process $X[n]$
  - True $\gamma_{XX}[l]$ must be estimated from $x[n] \Rightarrow \hat{\gamma}_{XX}[l]$
  - Parameter values computed using $\hat{\gamma}_{XX}[l]$ becomes parameter estimates $\{\hat{a}_k\} \Rightarrow$ Power spectrum estimate

$$\hat{\Gamma}_{XX}(f) = \frac{\hat{\sigma}_W^2}{\left|\hat{A}(f)\right|^2} = \frac{\hat{\sigma}_W^2}{\left|1 + \sum_{k=1}^{p} \hat{a}_k e^{-j2\pi f k}\right|^2}$$

$$\boxed{\begin{array}{c} \gamma_{XX}[l] \longrightarrow \{a_k\} \longrightarrow \Gamma_{XX}(f) \\ \downarrow \text{(estimation)} \\ \hat{\gamma}_{XX}[l] \longrightarrow \{\hat{a}_k\} \longrightarrow \hat{\Gamma}_{XX}(f) \end{array}}$$

30

30

PSD of an AR(p) process…

- Example: Estimate $\Gamma_{XX}(f)$ from an N-point realization of
  $X[n] = aX[n-1] + W[n]$, $W[n] \sim N(0, \sigma_w^2)$

1. Compute estimates of $\hat{\gamma}_{XX}[0]$ and $\hat{\gamma}_{XX}[1]$:

$$\hat{\gamma}_{XX}[l] = \frac{1}{N}\sum_{n=0}^{N-|l|-1} x[n]x[n+|l|]\,, l = 0,1$$

2. Estimate AR(1) parameter and noise variance (Yule-Walker):

$$\hat{a} = -\frac{\hat{\gamma}_{XX}[1]}{\hat{\gamma}_{XX}[0]}, \sigma_W^2 = \hat{\gamma}_{XX}[0] + \hat{a}\hat{\gamma}_{XX}[1]$$

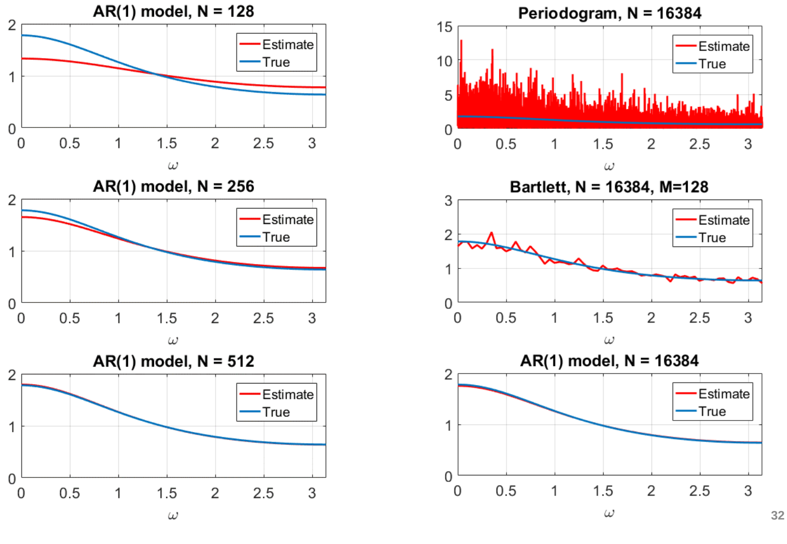
3. Power spectrum estimate:

$$\hat{\Gamma}_{XX}(f) = \frac{\hat{\sigma}_W^2}{|\hat{A}(f)|^2} = \frac{\hat{\sigma}_W^2}{|1+\hat{a}e^{-j2\pi fk}|^2}$$

31

31

PSD of an AR(1) process…



32

32

PSD of an AR(1) process…

```
Matlab
% AR prosess
a = [1 -0.25];
[H,Omega] = freqz(1,a,1024); % True spectrum

% WGN
N = 2^14;
W = randn(N,1);

% Observed AR prosess
X = filter(1,a,W);

% Estimate AR process
[a_e, sigmaW2_e] = aryule(X,1); % Estimate
[He,W]=freqz(sigmaW2_e, a_e, 1024);

plot(W/pi,10*log10(abs(H))), hold on
plot(W/pi,10*log10(abs(He)))
```

33

33

Summary

- Today we discussed:
  - Parametric spectral estimation

- Next:
  - Linear prediction

34

34

NTNU – Trondheim
Norwegian University of
Science and Technology

TTT4120 Digital Signal Processing
Fall 2018

Linear prediction

Prof. Stefan Werner
stefan.werner@ntnu.no
Office B329

Lecture in course book*

- Proakis, Manolakis Digital Signal Processing, 4th Ed.
  - 14.3.1 Forward linear prediction
  - 14.3.2 The Yule-Walker method for AR model parameters

- A comprehensive overview of topics treated in the lecture, see "Introdukjon til statistisk signalbehandling" on Blackboard

*Level of detail is defined by lectures and problem sets

Contents and learning outcomes

- How to find the AR parameters for a general process
- Linear prediction
- How many coefficients to choose? Model order estimation

Estimation in practice

- Only access to finite-length realization, $x[n]$, of process $X[n]$
  - True $\gamma_{XX}[l]$ must be estimated from $x[n] \Rightarrow \hat{\gamma}_{XX}[l]$
  - Parameter values computed using $\hat{\gamma}_{XX}[l]$ becomes parameter estimates $\{\hat{a}_k\} \Rightarrow$ Power spectrum estimate

$$\hat{\Gamma}_{XX}(f) = \frac{\hat{\sigma}_f^2}{\left|\hat{A}(f)\right|^2} = \frac{\hat{\sigma}_f^2}{\left|1+\sum_{k=1}^{p} \hat{a}_k e^{-j2\pi f k}\right|^2}$$

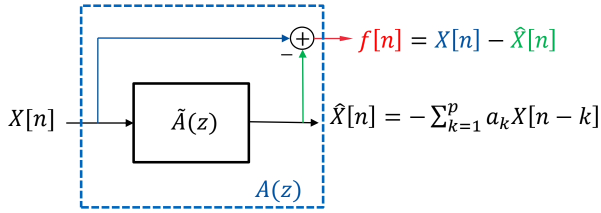
$$\boxed{\begin{array}{c} \gamma_{XX}[l] \longrightarrow \{a_k\} \longrightarrow \Gamma_{XX}(f) \\ \downarrow \text{(estimation)} \\ \hat{\gamma}_{XX}[l] \longrightarrow \{\hat{a}_k\} \longrightarrow \hat{\Gamma}_{XX}(f) \end{array}}$$

Estimation in practice…

- In practice process $X[n]$ may not be a true AR(p) process
  - How to choose parameters $\{\hat{a}_k\}$ to closely model $X[n]$ using an AR(p) process?
  - How do we measure closeness between model process and physical process?
- We will design pth-order linear predictor:
  - We observe/measure process $X[n]$
  - Store p prior values of $X[n]$, i.e., $\{X[n-1], \dots, X[n-p]\}$
  - Make linear combination of past values to estimate of $X[n]$

$$\hat{X}[n] = -\sum_{k=1}^{p} a_k X[n-k]$$

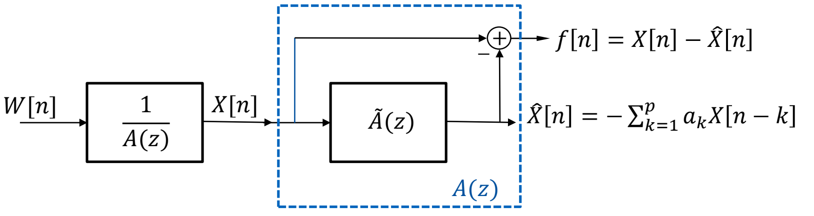
Linear prediction



- Design a_k to match $X[n]$ as good as possible in some sense
  - We can compute the prediction error
  - Error $f[n]$ should be small
  - Find predictor coefficients that minimize mean-square error

$$\sigma_f^2 = E\left\{\left(X[n] - \hat{X}[n]\right)^2\right\} = E\left\{\left(X[n] + \sum_{k=1}^{p} a_k X[n-k]\right)^2\right\}$$

Linear prediction…



- If $X[n]$ is a true AR(p) process then $f[n] = W[n]$ whenever the prediction coefficients a_k match those of the AR(p) process
- In practice this assumption leads to an approximation

7

Linear prediction…

- Elaborate the MSE

$$\sigma_f^2 = E\left\{\left(X[n] - \hat{X}[n]\right)^2\right\} = E\left\{\left(X[n] + \sum_{k=1}^{p} a_k X[n-k]\right)^2\right\}$$

$$= E\{X^2[n] + 2\sum_{k=1}^{p} a_k X[n-k]X[n] + \sum_{l=1}^{p}\sum_{k=1}^{p} a_k a_l X[n-k]X[n-l]\}$$

$$= \gamma_{XX}[0] + 2\sum_{k=1}^{p} a_k \gamma_{XX}[k] + \sum_{l=1}^{p}\sum_{k=1}^{p} a_k a_l \gamma_{XX}[l-k]$$

- MSE is minimum if we choose a_k such that

$$\frac{d\sigma_f^2}{da_k} = 0, k = 1, 2, \dots, p$$

8

Linear prediction…

- Example: Find optimal predictor for $p = 1$, i.e., $\hat{X}[n] = -a_1 X[n-1]$

$$\sigma_f^2 = E\left\{\left(X[n] - \hat{X}[n]\right)^2\right\} = E\{(X[n] + a_1 X[n-1])^2\}$$
$$= \gamma_{XX}[0] + 2a_1\gamma_{XX}[1] + a_1^2\gamma_{XX}[0]$$
$$= \gamma_{XX}[0] - \frac{\gamma_{XX}^2[1]}{\gamma_{XX}[0]} + \gamma_{XX}[0]\left(a_1 + \frac{\gamma_{XX}[1]}{\gamma_{XX}[0]}\right)^2$$

- Prediction error variance minimized for value a_1 that gives $\frac{d\sigma_f^2}{da_1} = 0$:

$$\frac{d\sigma_f^2}{da_1} = 2\gamma_{XX}[1] + 2a_1\gamma_{XX}[0] = 0 \Rightarrow a_1 = -\frac{\gamma_{XX}[1]}{\gamma_{XX}[0]}$$

- Resulting prediction variance: $\sigma_f^2 = \gamma_{XX}[0] + a_1\gamma_{XX}[1]$

Linear prediction…

- In vector notation: $\sigma_f^2 = \gamma_{XX}[0] + 2\boldsymbol{a}^{\mathrm{T}}\boldsymbol{\gamma}_{XX} + \boldsymbol{a}^{\mathrm{T}}\boldsymbol{\Gamma}_{XX}\boldsymbol{a}$

$$\boldsymbol{a} = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix}, \boldsymbol{\Gamma}_{XX} = \begin{bmatrix} \gamma_{XX}[0] & \gamma_{XX}[1] & \dots & \gamma_{XX}[p-1] \\ \gamma_{XX}[1] & \gamma_{XX}[0] & \dots & \gamma_{XX}[p-2] \\ \vdots & \ddots & \ddots & \vdots \\ \gamma_{XX}[p-1] & \gamma_{XX}[p-2] & \dots & \gamma_{XX}[0] \end{bmatrix}, \boldsymbol{\gamma}_{XX} = \begin{bmatrix} \gamma_{XX}[1] \\ \gamma_{XX}[2] \\ \vdots \\ \gamma_{XX}[p] \end{bmatrix}$$

- Set the gradient $\nabla_{\boldsymbol{a}}\sigma_f^2 = \boldsymbol{0}$, i.e.,

$$\nabla_{\boldsymbol{a}}\sigma_f^2 = \begin{bmatrix} \frac{\partial\sigma_f^2}{\partial a_1} & \cdots & \frac{\partial\sigma_f^2}{\partial a_p} \end{bmatrix}^{\mathrm{T}} = [0 \quad \cdots \quad 0]^{\mathrm{T}}$$

Linear prediction…

- $\nabla_{\boldsymbol{a}}\sigma_f^2 = \mathbf{0}$:

$$\nabla_{\boldsymbol{a}}\sigma_f^2 = 2\boldsymbol{\gamma}_{XX} + 2\boldsymbol{\Gamma}_{XX}\boldsymbol{a} = \mathbf{0}$$
$$\Rightarrow \boldsymbol{a} = -\boldsymbol{\Gamma}_{XX}^{-1}\boldsymbol{\gamma}_{XX}$$

- Minimum MSE:

$$\sigma_f^2 = \gamma_{XX}[0] + 2\boldsymbol{a}^T\boldsymbol{\gamma}_{XX} + \boldsymbol{a}^T\boldsymbol{\Gamma}_{XX}\boldsymbol{a}$$
$$= \gamma_{XX}[0] + 2\boldsymbol{a}^T\boldsymbol{\gamma}_{XX} - \boldsymbol{a}^T\boldsymbol{\gamma}_{XX}$$
$$= \gamma_{XX}[0] + \boldsymbol{a}^T\boldsymbol{\gamma}_{XX} = \gamma_{XX}[0] + \sum_{k=1}^{p} a_k\gamma_{XX}[k]$$

- Same solution as we had for a pure AR(p) process

11

Linear prediction…

- Alternative approach by completing the square:

$$\sigma_f^2 = \gamma_{XX}[0] + 2\boldsymbol{a}^{\mathrm{T}}\boldsymbol{\gamma}_{XX} + \boldsymbol{a}^{\mathrm{T}}\boldsymbol{\Gamma}_{XX}\boldsymbol{a}$$
$$= \gamma_{XX}[0] - \boldsymbol{\gamma}_{XX}^{\mathrm{T}}\boldsymbol{\Gamma}_{XX}^{-1}\boldsymbol{\gamma}_{XX} + (\boldsymbol{a} + \boldsymbol{\Gamma}_{XX}^{-1}\boldsymbol{\gamma}_{XX})^{\mathrm{T}}\boldsymbol{\Gamma}_{XX}(\boldsymbol{a} + \boldsymbol{\Gamma}_{XX}^{-1}\boldsymbol{\gamma}_{XX})$$

- The above holds true whenever $\boldsymbol{\Gamma}_{XX}$ is positive definite, i.e.,

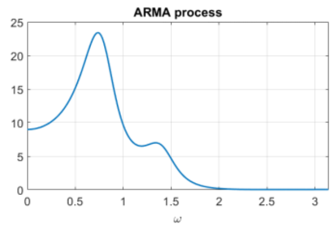
$$\boldsymbol{x}^{\mathrm{T}}\boldsymbol{\Gamma}_{XX}\boldsymbol{x} > 0, \forall \boldsymbol{x} \neq \mathbf{0}$$

- Consequently, σ_f^2 is minimized when last term equals zero

$$\boldsymbol{a} = -\boldsymbol{\Gamma}_{XX}^{-1}\boldsymbol{\gamma}_{XX}$$

12

Linear prediction…



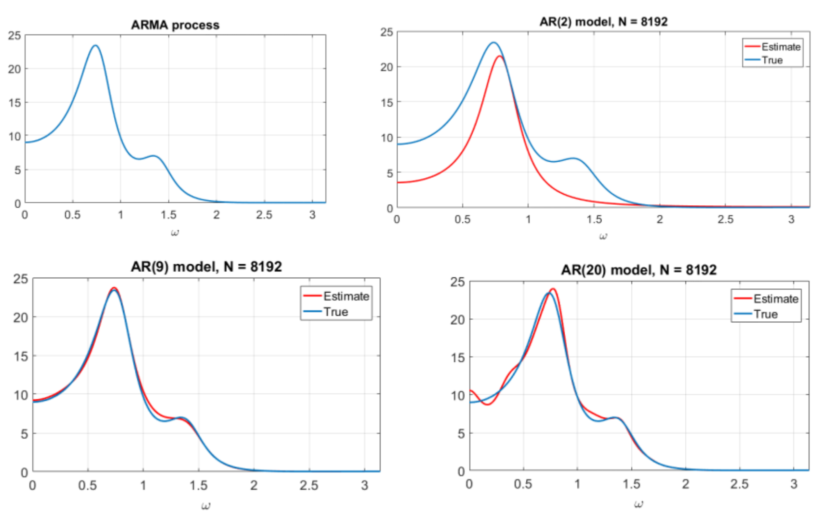
- Example: Estimate $\Gamma_{XX}(f)$ from a realization of an N-point ARMA process,

$$X[n] = -\sum_{k=1}^{p} a_k X[n-k] + \sum_{k=0}^{q} b_k W[n-k],\ W[n] \sim N(0, \sigma_W^2)$$

- Approximate with an AR(p) process and estimate model coefficients, $\hat{a}_k$, by minimizing prediction error variance, σ_f^2
  - What model order should I use?

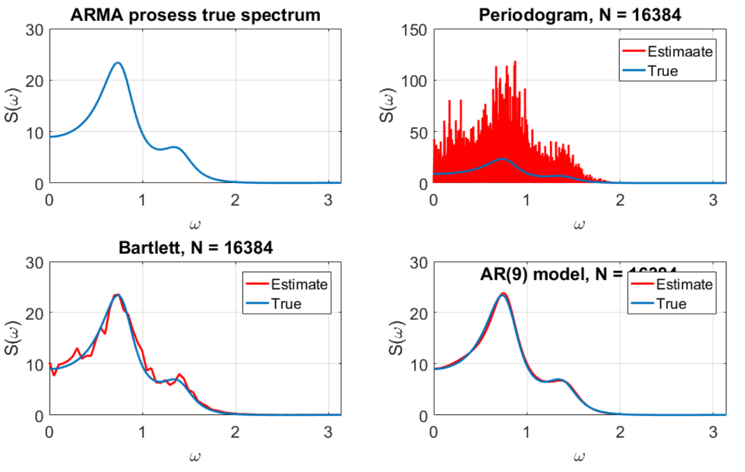
13

Linear prediction…



14

Linear prediction…



15

Determining model order *p*

- Model order not known when we shall model a physical process
- Proper choice of order p is necessary for good modelling capability
  - Too small p leads to smoothened spectrum
  - Too large p leads to spurious low-level peaks in the spectrum
- Prediction variance $\sigma_f^2(p)$ could be an indicator
  - Monotonically decreasing with p
  - Need to decide when changes are sufficiently small
  - Usually imprecise: in general no clear knee visible in plot $\sigma_f^2(p)$

16

Determining model order *p*

- Different criteria that penalizes high model order p:

$$\mathrm{FPE}(p) = \sigma_f^2(p)\frac{N+p+1}{N-p-1}$$

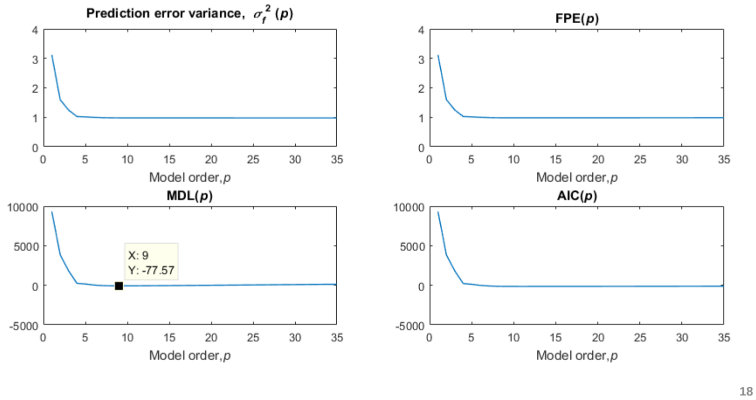
$$\mathrm{MDL}(p) = N\log\sigma_f^2(p) + p\log N$$

$$\mathrm{AIC}(p) = N\log\sigma_f^2(p) + 2p$$

Determining model order *p*…

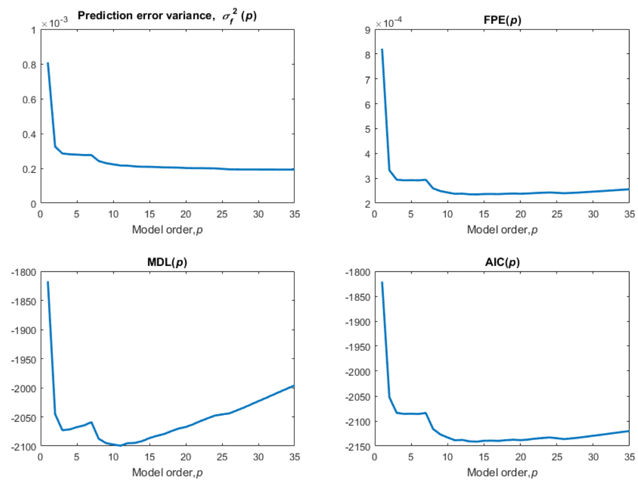
- Example: Estimate $\Gamma_{XX}(f)$ from a realization of an ARMA process

$$X[n] = -\sum_{k=1}^{p} a_k X[n-k] + \sum_{k=0}^{q} b_k W[n-k],\ W[n] \sim N(0,\sigma_w^2)$$



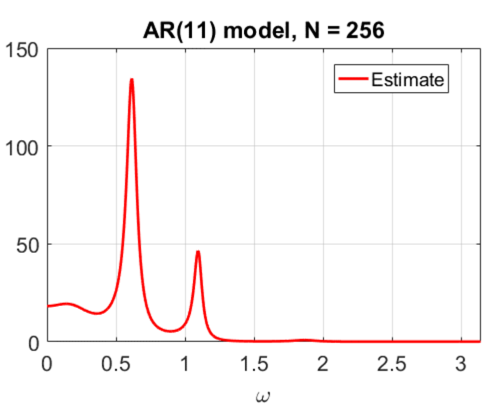
Determining model order *p*…

- Example: Vowel ‘æ’, $N = 256$:

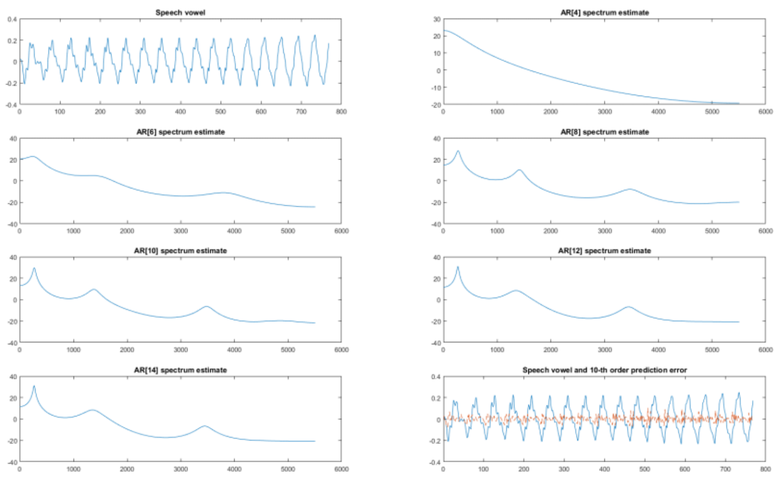


Determining model order *p*…

- Example: Vowel ‘æ’, $N = 256$:



Determining model order *p…*



21

Final notes on estimation in practice

- All methods looked at so far assume
  - Random processes to be stationary and ergodic
  - Random processes are autoregressive (AR)
- In practice, all physical processes of interest are nonstationary
  - Short-time stationarity: process varies slowly and within a certain time window, statistical properties are constant
  - Assume stationarity over M times and we need $N < M$ points
- Other methods for finding estimates
  - Usually lead to similar performance. Main differences are in the performance with few data points

22

Summary

- Today we discussed:
  - Linear prediction
  - Model order
- Next time:
  - FIR filter design

NTNU – Trondheim
Norwegian University of Science and Technology

TTT4120 Digital Signal Processing
Fall 2018

Design of Digital Filters: FIR

Prof. Stefan Werner
stefan.werner@ntnu.no
Office B329

Department of Electronic Systems

Lecture in course book*

- Proakis, Manolakis Digital Signal Processing, 4th Ed.
  - 10.2.2 Design of linear-phase FIR filters using windows
  - 10.2.4 Design of optimum equiripple linear-phase FIR filters
- A compressed overview of topics treated in the lecture, see "Design av digitale filtre" on Blackboard

*Level of detail is defined by lectures and problem sets

2

Contents and learning outcomes

- Filter specifications
- FIR versus IIR
- Window method
- Equirippel design

Filter design

- Systems may be designed to amplify or attenuate parts of the input signal (e.g., remove noise, suppress interference)

$x[n]$ → $h[n]$ → $y[n] = h[n] * x[n]$

$X(\omega), X(z)$ $H(\omega), H(z)$

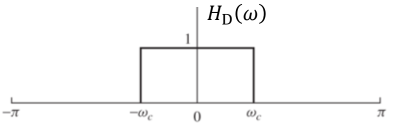
$$Y(\omega) = H(\omega)X(\omega)$$
$$Y(z) = H(z)X(z)$$

- A discrete-time filter modifies the Fourier representation of $x[n]$
  - Lowpass
  - Highpass
  - Bandpass
  - Bandstop, etc.

Filter design...

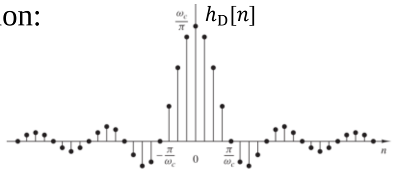
- Ideal lowpass filter:

$$H_D(\omega) = \begin{cases} 1, & |\omega| \le \omega_c \\ 0, & \omega_c \le \omega \le \pi \end{cases}$$



- Impulse response in the sinc function:

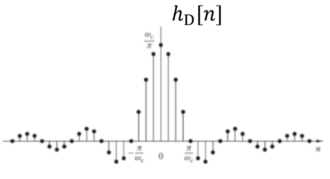
$$h_D[n] = \frac{\omega_c}{\pi} \frac{\sin \omega_c n}{\omega_c n}$$



- Problems:
  - Ideal filters are not causal $\Longrightarrow$ not physically realizable
  - Infinite complexity and delay, not BIBO stable, etc.

5

Filter design...

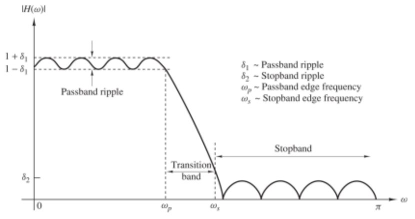


$\approx$?

- We want causal linear-phase filters $\Longrightarrow$ approximations needed
  - Truncate time-domain pulse (windowing)
  - Control frequency response (equirippel design)

6

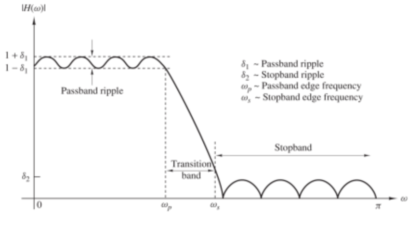
Filter design...



- In practice, ideal filter characteristics are not absolutely necessary
- Find filter of minimum complexity satisfying a given specification
  - Nonconstant magnitude in passband (small ripple)
  - Non-zero stopband (small value or small amount of ripple)
  - Allow for non-zero transition band from passband to stopband
- The more restrictions on the design, the more complex it will be

7

Filter design...

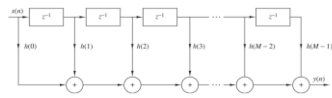


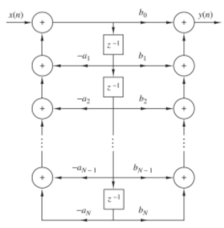
- Real-valued, causal filters of the form: $H(z) = \frac{b_0+b_1z^{-1}+\cdots+b_Mz^{-M}}{1+a_1z^{-1}+\cdots+a_Nz^{-N}}$
- FIR: $H(z) = \sum_{k=0}^{M-1} b_k z^{-k} \Rightarrow y[n] = \sum_{k=0}^{M-1} b_k x[n-k]$
- IIR: $H(z) = \frac{\sum_{k=0}^{M-1} b_k z^{-k}}{1+\sum_{k=1}^{N-1} a_k z^{-k}} \Rightarrow y[n] = -\sum_{k=1}^{N-1} a_k z^{-k} + \sum_{k=0}^{M-1} b_k x[n-k]$
- Find $\{a_k\}$ and $\{b_k\}$ that satisfy filter specification

8

FIR versus IIR

- FIR filters:
  - Always stable
  - Can achieve exactly linear phase
  - Easily designed with linear methods
  - Easy to implement
- IIR filters:
  - Fewer parameters (low filter order)
  - Less memory
  - Low delay
  - Lower computational complexity
  - Typically designed by transforming an analog filter design



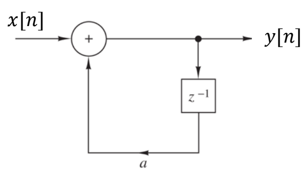


9

FIR versus IIR...

- Example: $H(z) = \frac{1}{1-az^{-1}}$, $|a| < 1$

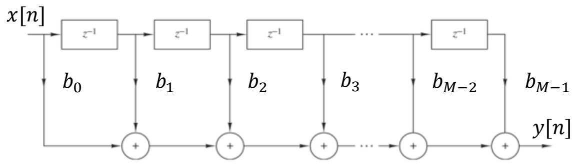
- IIR implementation: $y[n] = ay[n-1] + x[n]$



- FIR approximation: $y[n] = \sum_{k=0}^{M} a^k \, x[n-k]$, M large

10

Linear-phase FIR filters



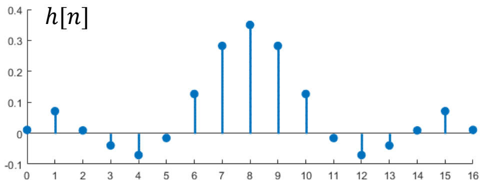
- Moving average filter, or an all-zero filter, of order M

$$H(z) = \sum_{k=0}^{M-1} b_k z^{-k} = b_0 z^{-(M-1)} \prod_{k=1}^{M-1} (z - z_k)$$

- Design of $\{b_k\}$ $\Leftrightarrow$ moving zeros in the z-plane
  - Can be designed using some optimality criterion
- Impulse response $h[n]$ of an FIR filter given by the filter weights
  - Easily verified by setting $x[n] = \delta[n]$

11

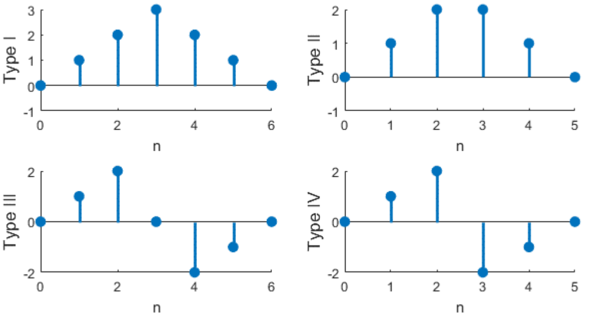
Linear-phase FIR filters...



- FIR filters can be causal and have linear phase
  - Implies a linear shift in time domain (no distortion)
  - Exact linear phase not possible in IIR filters
- Linear phase filters must have symmetric impulse response
  - Four possibilities: M even/odd, $h[n]$ symmetric/antisymmetric

12

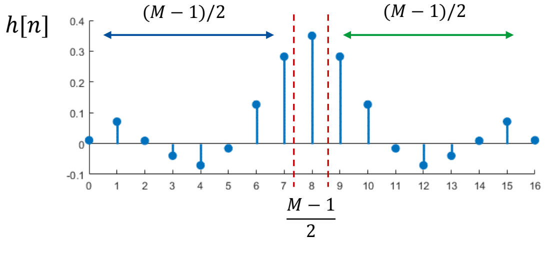
Linear-phase FIR filters...



- Let us review the case of M odd and $h[n]$ symmetric (Lecture 7)
  $\Rightarrow M-1$ even and $h[n] = h[M-1-n]$

13

Linear-phase FIR filters...



$$H(z) = \sum_{k=0}^{M-1} b_k z^{-k} =$$

$$= \sum_{k=0}^{(M-3)/2} h[k] z^{-k} + h\left[\frac{M-1}{2}\right] z^{-(M-1)/2} + \sum_{k=(M+1)/2}^{M-1} h[k] z^{-k}$$

$$= \sum_{k=0}^{(M-3)/2} h[k] z^{-k} + h\left[\frac{M-1}{2}\right] z^{-(M-1)/2} + \sum_{k=(M+1)/2}^{M-1} h[M-1-k] z^{-k}$$

$$= \sum_{k=0}^{(M-3)/2} h[k] z^{-k} + h\left[\frac{M-1}{2}\right] z^{-(M-1)/2} + \sum_{l=0}^{(M-3)/2} h[l] z^{l-(M-1)}$$

14

Linear-phase FIR filters...

$$H(z) = \sum_{k=0}^{(M-3)/2} h[k]z^{-k} + h\left[\frac{M-1}{2}\right] z^{-(M-1)/2} + \sum_{k=0}^{(M-3)/2} h[k]z^{k-(M-1)}$$

$$= h\left[\frac{M-1}{2}\right] z^{-(M-1)/2} + \sum_{k=0}^{(M-3)/2} h[k]\left(z^{-k} + z^{k-(M-1)}\right)$$

$$= h\left[\frac{M-1}{2}\right] z^{-(M-1)/2} + \sum_{k=0}^{(M-3)/2} h[k]z^{-(M-1)/2}\left(z^{-(k-(M-1)/2)} + z^{k-(M-1)/2}\right)$$

$$= \left(h\left[\frac{M-1}{2}\right] + \sum_{k=0}^{(M-3)/2} h[k]\left(z^{-(k-(M-1)/2)} + z^{k-(M-1)/2}\right)\right) z^{-(M-1)/2}$$

- Frequency response obtained by substituting $z = e^{j\omega}$

$$H(\omega) = \left(h\left[\frac{M-1}{2}\right] + 2\sum_{k=0}^{(M-3)/2} h[k]\cos[\omega((M-1)/2-k)]\right) e^{-j\omega(M-1)/2}$$

15

Linear-phase FIR filters...

- Frequency response M odd and $h[n]$ symmetric

$$H(\omega) = \left(h\left[\frac{M-1}{2}\right] + 2\sum_{k=0}^{\frac{M-3}{2}} h[k]\cos\left[\omega\left(\frac{M-1}{2}-k\right)\right]\right) e^{-\frac{j\omega(M-1)}{2}}$$

$$= H_R(\omega)\, e^{-\frac{j\omega(M-1)}{2}}$$

- Amplitude of filter $H_R(\omega) \in \mathbb{R}$ similar to $|H(\omega)|$ since $|H_R(\omega)| = |H(\omega)|$
- However, note that $H_R(\omega)$ can be less than 0
- Linear shift $e^{-\frac{j\omega(M-1)}{2}}$: $H(\omega)$ has piecewise linear phase
  - When $H_R(\omega)$ changes sign, phase jumps π radians (usually in stopband)

16

Linear-phase FIR filters...

- All possibilities (similar derivations):
  - Type I. Symmetric, $h[n] = h[M-1-n]$, M odd:

$$H(\omega) = \left(h\left[\frac{M-1}{2}\right] + 2\sum_{n=0}^{(M-3)/2} h[n]\cos\left[\frac{M-1}{2} - n\right]\right) e^{-j\omega(M-1)/2}$$

  - Type II. Symmetric, $h[n] = h[M-1-n]$, M even:

$$H(\omega) = \left(2\sum_{n=0}^{(M-2)/2} h[n]\cos\left[\frac{M-1}{2} - n\right]\right) e^{-j\omega(M-1)/2}$$

  - Type III. Antisymmetric, $h[n] = -h[M-1-n]$, M odd:

$$H(\omega) = \left(2\sum_{n=0}^{(M-3)/2} h[n]\cos\left[\frac{M-1}{2} - n\right]\right) e^{-j\left[\frac{\omega(M-1)}{2}+\frac{\pi}{2}\right]}$$

  - Type IV. Antisymmetric, $h[n] = -h[M-1-n]$, M even:

$$H(\omega) = \left(2\sum_{n=0}^{(M-2)/2} h[n]\sin\left[\frac{M-1}{2} - n\right]\right) e^{-j\left[\frac{\omega(M-1)}{2}+\frac{\pi}{2}\right]}$$

17

Linear-phase design using windowing

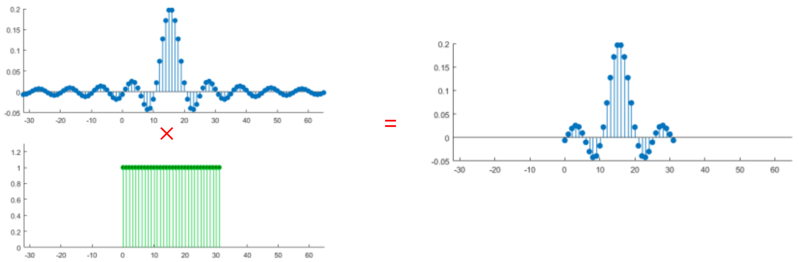
- Basic design principle: Start with a desired frequency specification $H_D(\omega)$ and determine impulse response

$$h_D[n] = \frac{1}{2\pi}\int_{-\pi}^{\pi} H_D(\omega)e^{j\omega n}d\omega$$

- In general $h_D[n]$ is of infinite length and need to be truncated
- To obtain causal FIR filter of length M we can multiply $h_D[n]$ with a rectangular window

18

Linear-phase design using windowing...



- Truncation of $h_{\mathrm{D}}[n] \Leftrightarrow$ multiplying $h_{\mathrm{D}}[n]$ by window $w[n]$

$$h[n] = h_{\mathrm{D}}[n] w_{\mathrm{R}}[n]$$

where

$$w_{\mathrm{R}}[n] = \begin{cases} 1 & \text{for } 0 \leq n \leq L-1 \\ 0 & \text{otherwise} \end{cases}$$

Linear-phase design using windowing...

- Multiplication in time-domain corresponds to

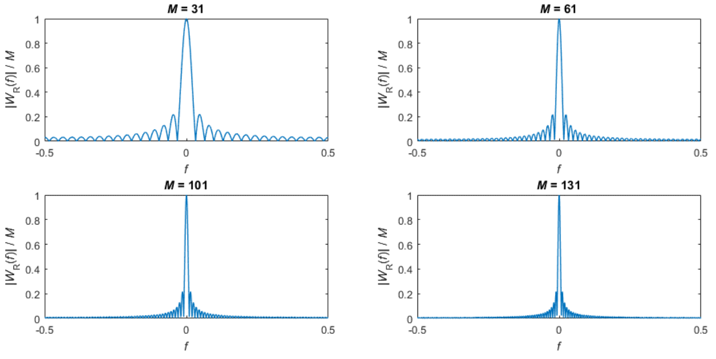
$$H(\omega) = H_{\mathrm{D}}(\omega) * W(\omega) = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(\lambda) W(\omega - \lambda) d\lambda$$

with $W(\omega) = e^{-j\omega(M-1)/2} \frac{\sin\frac{\omega M}{2}}{\sin\frac{\omega}{2}}$

- Rectangular window has a mainlobe and sidelobes
  - Mainlobe smoothens desired frequency response
  - Sidelobes introduce ringing effects

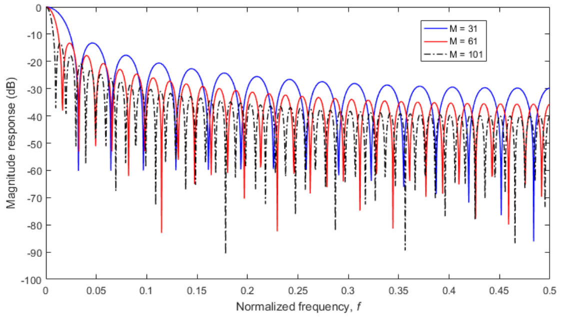
Linear-phase design using windowing...

- Illustration: $\frac{1}{M}|W(\omega)| = \frac{1}{M}\frac{|\sin\frac{\omega M}{2}|}{|\sin\frac{\omega}{2}|}$



21

Linear-phase design using windowing



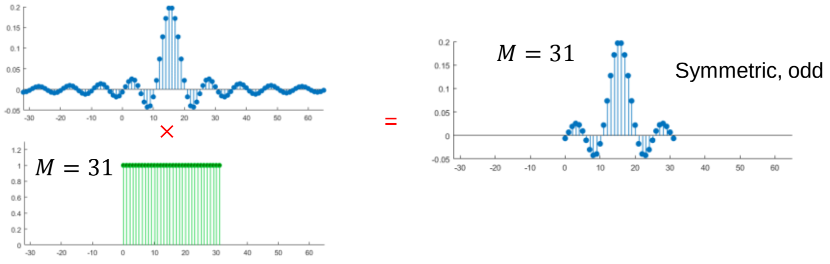
- Large dynamic range $\Longrightarrow$ plot magnitude response in dB

Matlab
```
M   = 31;
wR  = window(@rectwin,M);
[WR,w]=freqz(wR,1,1024);
plot(w/2/pi,20*log10(abs(WR)/M))
```

22

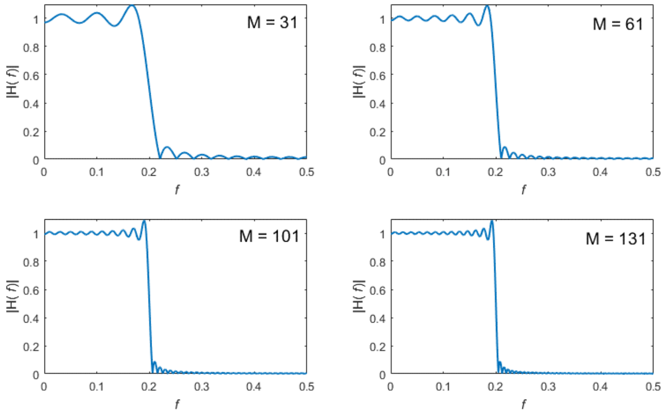
Linear-phase design using windowing...

- Design example: $H_\mathrm{D}(\omega) = \begin{cases} 1 \cdot e^{-j\omega(M-1)/2}, & |\omega| \le \omega_c \\ 0, & \omega_c \le \omega \le \pi \end{cases}$
- Corresponding impulse response $h_\mathrm{D}[n] = \frac{\omega_c}{\pi} \frac{\sin \omega_c[n-(M-1)/2]}{\omega_c[n-(M-1)/2]}$
- Truncated response: $h[n] = w_\mathrm{R}[n] h_\mathrm{D}[n]$



23

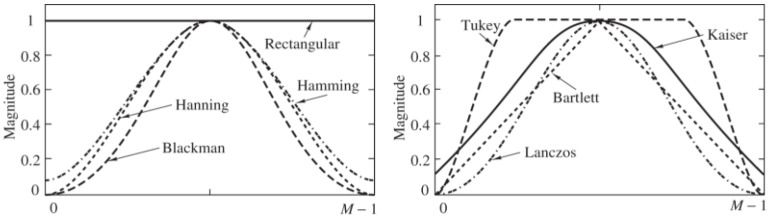
Linear-phase design using windowing...



- Oscillations do not disappear as M increases (Gibbs)
- Use other windows to reduce ripples in passband and stopband

24

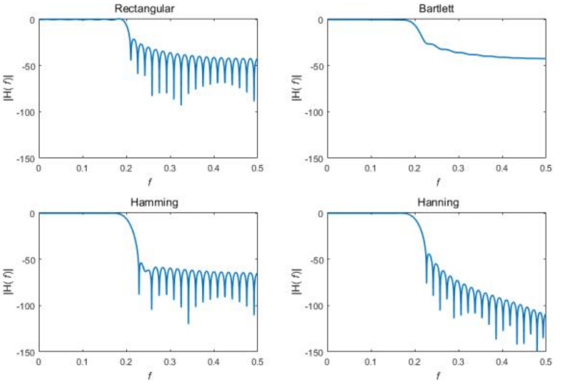
Different windows in time domain



- Type 'window' at Matlab command prompt
- Transition bandwidth depends on window length and type
- Passband attenuation
  - Depends on window chosen
- Rectangular window narrowest mainlobe
  - Smallest transition region but worst attenuation in stopband

25

Different windows in time domain...



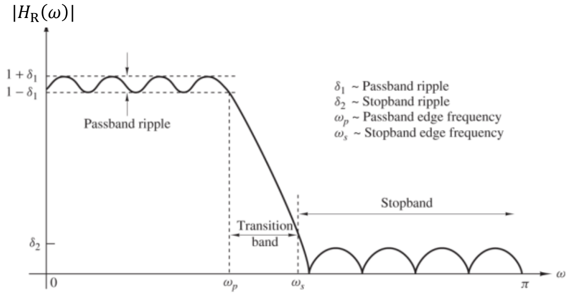
Matlab

```
M  = 61;
wc = 2*pi*0.2;
hB = fir1(M-1,wc/pi,bartlett(M));
[HB,w]=freqz(B,1,1024);
plot(w/2/pi,20*log10(abs(HB)))
```

26

Equiripple design of linear-phase filters

- Major disadvantage of window method is the lack of precise control of the critical frequencies at band edges, i.e., ω_p and ω_s
- Instead, find filter coefficients $h_R[n]$ to minimize the maximal deviation from a desired response $H_D(\omega)$



Equiripple design of linear-phase filters...

- Define an error function $E(\omega) = W(\omega)[H_D(\omega) - H_R(\omega)]$
  - $H_D(\omega)$ is the desired frequency response
  - $H_R(\omega)$ is the frequency response with filter coefficients $h[n] = b_n$
  - $W(\omega)$ is a weight function given by the filter specs

$$W(\omega) = \begin{cases} \frac{\delta_2}{\delta_1} & \omega \leq \omega_p \\ 0 & \omega \geq \omega_s \end{cases}$$

- Find filter coefficients $h[n]$ that minimizes maximal deviation

$$\min_{h[n]} \max_{\omega} |W(\omega)[H_D(\omega) - H_R(\omega)]|$$

- Result of minimization is a filter with equirippel characteristic

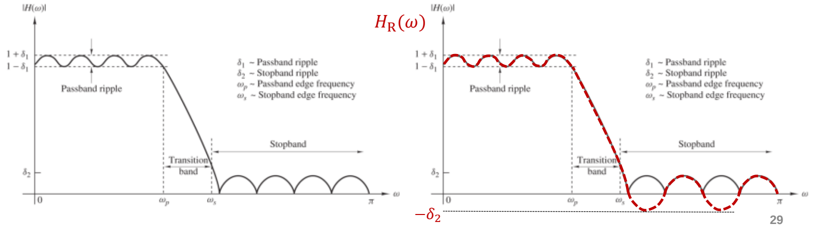
Equiripple design of linear-phase filters...

- Let us consider linear-phase filter of Type 1 (symmetric, M odd):

$$H(\omega) = H_{\mathrm{R}}(\omega)\, e^{-\frac{j\omega(M-1)}{2}},$$

$$\text{with } H_{\mathrm{R}}(\omega) = h\left[\frac{M-1}{2}\right] + 2\sum_{k=0}^{(M-3)/2} h[k]\cos\left[\omega\left(\frac{M-1}{2} - k\right)\right]$$

- Design $H(\omega)$ equivalent to design $H_{\mathrm{R}}(\omega)$: slightly different specs



Equiripple design of linear-phase filters

- Goal is to find optimal $H_{\mathrm{R}}(\omega)$ that complies with specifications

$$H_{\mathrm{R}}(\omega) = h\left[\frac{M-1}{2}\right] + 2\sum_{k=0}^{(M-3)/2} h[k]\cos\left[\omega\left(\frac{M-1}{2} - k\right)\right]$$

- Optimization over the filter taps $h[n],\ n = 0, \dots, (M-1)/2 + 1$
- The weighted error function is

$$E(\omega) = W(\omega)[H_{\mathrm{D}}(\omega) - H_{\mathrm{R}}(\omega)]$$
$$= W(\omega)\left[H_{\mathrm{D}}(\omega) - \left(h\left[\frac{M-1}{2}\right] + 2\sum_{k=0}^{(M-3)/2} h[k]\cos\left[\omega\left(\frac{M-1}{2} - k\right)\right]\right)\right]$$

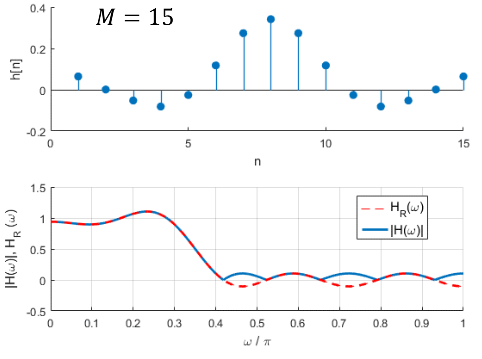
- Alternation theorem: The optimal $H_{\mathrm{R}}(\omega)$ will touch the error bounds at $(M-1)/2 + 2$ frequencies in interval $[0, \pi]$

Equiripple design of linear-phase filters...

- Alternation theorem: The optimal $H_{\mathrm{R}}(\omega)$ will touch the error bounds at $(M-1)/2+2$ frequencies in interval $[0,\pi]$
- Remez Exchange algorithm finds coefficients $h[k]$ such that $H_{\mathrm{R}}(\omega)$ satisfies the alternation theorem
  - Always converges to an equiripple solution
  - May not have the passband/stopband characteristics needed for a given M $\Rightarrow$ increase M

31

Equiripple design of linear-phase filters...



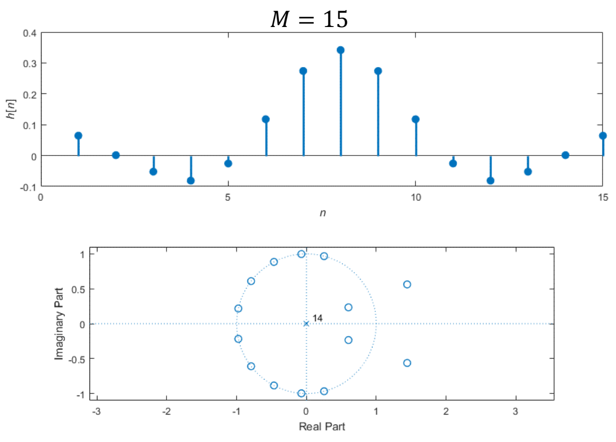
Matlab

```
E = [0 0.3 0.4 1];
A = [1 1 0 0];
M = 15;
B = firpm(M-1, E, A)
w = linspace(0,pi,500);
H = freqz(B,1,w);
figure
subplot(2,1,1),
stem(B);
subplot(2,1,2),
plot(w/pi,abs(H));
```

- $(M-1)/2+2=(15-1)/2+2=9$ alternations
- Change width of transition band (comment on result)

32

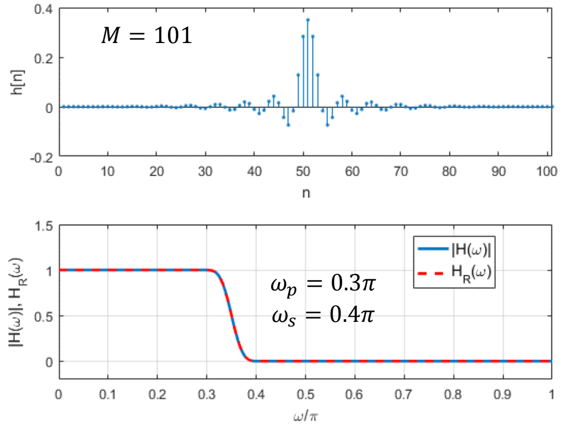
Equiripple design of linear-phase filters...



- Check pole-zero plot with `zplane(B,1)`

33

Equiripple design of linear-phase filters...



- $(M-1)/2+2=(101-1)/2+2=52$ alterations

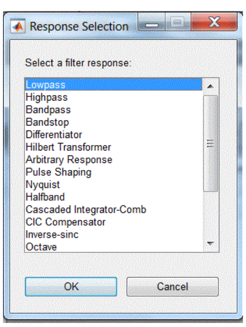
34

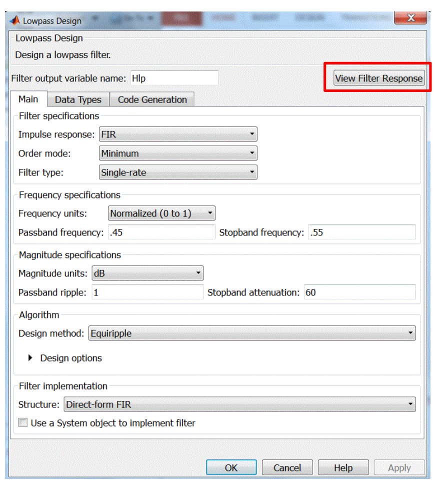
Summary

- Today we discussed:
  - Basics of filter design
  - Linear phase filters using windowing and equiripple designs
- Next:
  - IIR filter design

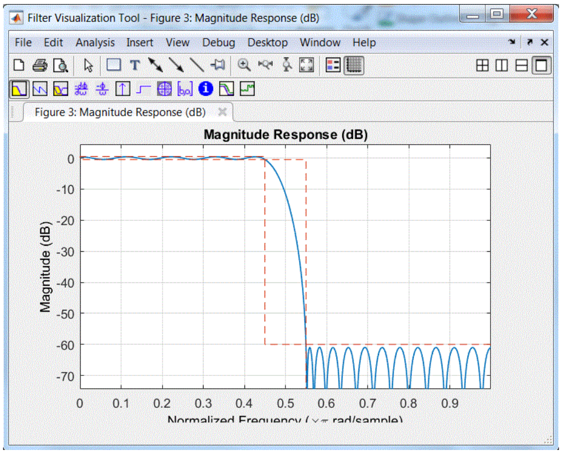
Matlab: filterbuilder…

- Type `filterbuilder` at Matlab command prompt:

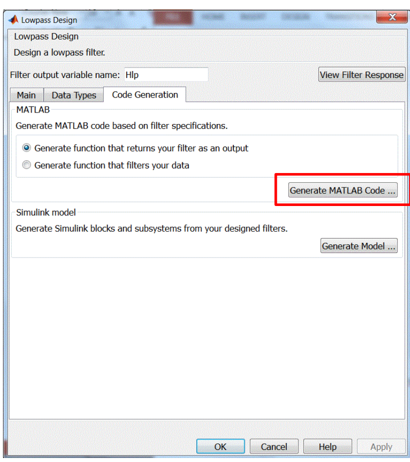




Matlab: filterbuilder



Matlab: filterbuilder…




```
function Hd = test
%TEST Returns a discrete-time filter object.

% MATLAB Code
% Generated by MATLAB(R) 8.6 and the Signal Processing Toolbox 7.1.
% Generated on: 07-Oct-2016 14:29:33

Fpass = 0.45;  % Passband Frequency
Fstop = 0.55;  % Stopband Frequency
Apass = 1;     % Passband Ripple (dB)
Astop = 60;    % Stopband Attenuation (dB)

h = fdesign.lowpass('fp,fst,ap,ast', Fpass, Fstop, Apass, Astop);

Hd = design(h, 'equiripple', ...
    'MinOrder', 'any', ...
    'StopbandShape', 'flat');
```

NTNU – Trondheim
Norwegian University of Science and Technology

TTT4120 Digital Signal Processing
Fall 2020

Design of Digital Filters: IIR

Prof. Stefan Werner
stefan.werner@ntnu.no
Office B329

1

Lecture in course book*

- Proakis, Manolakis Digital Signal Processing, 4th Ed.
  - 10.3.3 IIR filter design by the bilinear transformation
  - 10.3.4 Characteristics of commonly used analog filters
- A compressed overview of topics treated in the lecture, see "Design av digitale filtre" on Blackboard

*Level of detail is defined by lectures and problem sets

2

2

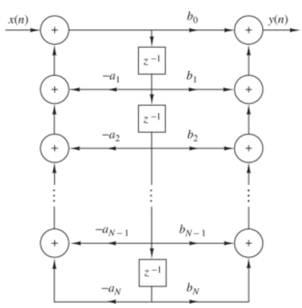
Contents and learning outcomes

- IIR filter
- Bilinear transformation
- Examples

IIR filters

- Moving and recursive averages
- Filter has both poles and zeros

$$H(z) = \frac{\sum_{k=0}^{M} b_k z^{-k}}{1+\sum_{k=1}^{N} a_k z^{-k}}$$



- IIR filters designed, $\{a_k\}$ and $\{b_k\}$, by specifying poles and zeros in the z-plane
- In general IIR filters can, for a given filter order, satisfy a tighter specification than FIR filters (lower computational complexity)

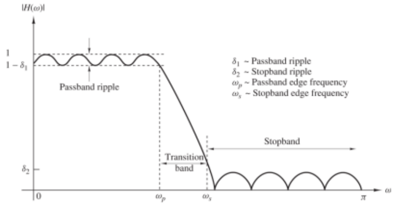
IIR filters...

- In contrast to FIR filter design, IIR filters are typically designed by utilizing known analog filter design
  - Take an analog design and transform it to the digital domain
  - Nice thing: closed-form solutions exist
  - How to transform the analog solutions to discrete-time?
- Three ways of describing an analog filter
  - System function: $H_a(s) = \frac{B(s)}{A(s)} = \frac{\sum_{k=0}^{M} \beta_k s^k}{\sum_{n=0}^{N} \alpha_k s^k}$
  - Impulse response: $H_a(s) = \int_{-\infty}^{\infty} h_a(t) e^{-st} dt$
  - Differential equations: $\sum_{k=0}^{N} \alpha_k \frac{d^k y(t)}{dt^k} = \sum_{k=0}^{M} \beta_k \frac{d^k x(t)}{dt^k}$
- We will be using the system function

5

5

IIR filters...



- IIR filter design quite different from the FIR design
  - Step 1: State filter specs of digital filter, $\{\omega_p, \omega_s, \delta_1, \delta_2\}$
  - Step 2: Map the specs to analog domain, $\omega_p \rightarrow \Omega_p$, $\omega_s \rightarrow \Omega_s$
  - Step 3: Design an analog filter (for resistors, capacitors, and inductors) using the Laplace transform $H(s)$
  - Step 4: The design in digital domain is obtained using mapping $s = f(z)$, or $H(z) = H(s)|_{s=f(z)}$

6

6

Transformation between s- and z-planes

- We need to transform an analog design into a digital design
  - How to go from the s-plane to the z-plane?

$$H(z) = H_a(s)|_{s=f(z)}$$

- Demands on the mapping?
  - Stable analog filters need to be mapped to stable digital filters

$$Re\{s\} < 0 \;\Rightarrow |z| < 1$$

  - Imaginary axis in s-plane mapped to unit circle in z-plane

$$Re\{s\} = 0 \;\Rightarrow |z| = 1 \;\Leftrightarrow j\Omega \rightarrow e^{j\omega}$$

- The bilinear tranformation satisfies these conditions
- Alternative method is to sample analog impulse respose

Impulse invariance method

- Sample analog impulse response

$$h[n] = h_a(t)|_{t=nT} \overset{\mathcal{F}}{\leftrightarrow} H(\Omega T) = \frac{1}{T}\sum_{k=-\infty}^{\infty} H_a\left(\Omega - \frac{2\pi k}{T}\right)$$

- Frequency mapping: $\omega = \Omega T$
  - Simple and linear
  - Suffers from potential aliasing $\Rightarrow$ not useful for highpass
- Transfer function:

$$H_a(s) = \sum_{k=1}^{N} \frac{c_k}{s-p_k} \rightarrow H(z) = \sum_{k=1}^{N} \frac{c_k}{1-e^{p_k T} z^{-1}}$$

- No explicit mapping $s = f(z)$, but mapping of poles

$$s_k = p_k \rightarrow z_k = e^{p_k T}$$

Impulse invariance method...

- Procedure for transforming $H_a(s)$ to $H(z)$:
  1. Find poles of $H_a(s)$, p_k
  2. Express $H_a(s) = \sum_{k=1}^{N} \frac{c_k}{s-p_k}$
  3. Finally, $H(z) = \sum_{k=1}^{N} \frac{c_k}{1-e^{p_k T} z^{-1}}$

9

9

Bilinear transformation...

- The bilinear transform, a conformal mapping, provides an explicit mapping between s-plane and z-plane

$$s = \frac{2}{T}\frac{z-1}{z+1}, \text{ or } z = \frac{\frac{2}{T}+s}{\frac{2}{T}-s}$$

- Setting $s = \sigma + j\Omega$ and $z = e^{j\omega}$, we get the frequency mapping

$$\omega = 2 \arctan \frac{\Omega T}{2} \text{ or } \Omega = \frac{2}{T} \tan \frac{\omega}{2}$$

- The discrete-time filter's system function is given by

$$H(z) = H_a(s)\Big|_{s=\frac{2}{T}\frac{z-1}{z+1}}$$

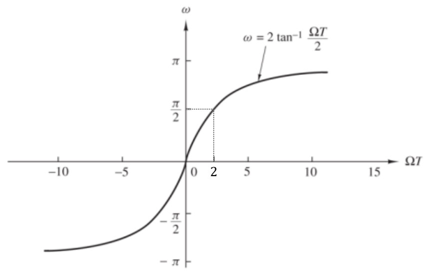
10

10

Bilinear transformation...

- Example: Fill in the table using $z = \frac{\frac{2}{T}+s}{\frac{2}{T}-s}$ and $\omega = 2\arctan\frac{\Omega T}{2}$

s	z	ω
0		
∞		
$\frac{2j}{T}$		
$-\frac{2}{T}$		



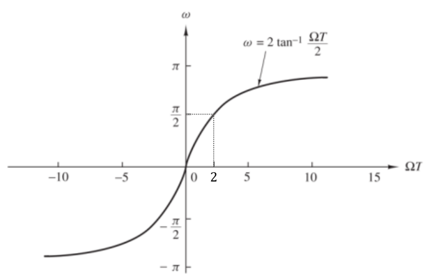
11

11

Bilinear transformation...

- Example: Fill in the table using $z = \frac{\frac{2}{T}+s}{\frac{2}{T}-s}$ and $\omega = 2\arctan\frac{\Omega T}{2}$

s	z	ω
0	1	0
∞	-1	π
$\frac{2j}{T}$	j	$\pi/2$
$-\frac{2}{T}$	0	N/A



12

12

Bilinear transformation...

- Substitute $s = \sigma + j\Omega$ in $|z| = \frac{\left|\frac{2}{T}+s\right|}{\left|\frac{2}{T}-s\right|}$, look at $\sigma < 0, \sigma = 0, \sigma > 0$
  - For $\sigma < 0 \Rightarrow |z| = \left|\frac{\frac{2}{T}+\sigma+j\Omega}{\frac{2}{T}-\sigma-j\Omega}\right| < 1$
  - For $\sigma = 0 \Rightarrow |z| = \left|\frac{\frac{2}{T}+j\Omega}{\frac{2}{T}-j\Omega}\right| = 1$
  - For $\sigma > 0 \Rightarrow |z| = \left|\frac{\frac{2}{T}+\sigma+j\Omega}{\frac{2}{T}-\sigma-j\Omega}\right| > 1$
- Entire left half-plane maps into the inside of unit circle
- Imaginary axis maps onto the unit circle

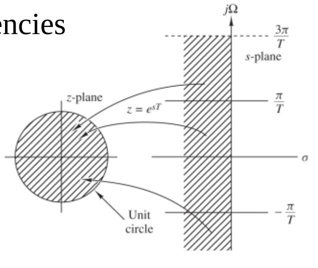
13

Bilinear transformation...

- The mapping satisfies the conditions for stability and mapping of the $j\Omega$-axis to the unit circle
- Reversible mapping of frequency axis, i.e.,

$$\Omega \in (-\infty, \infty) \leftrightarrow \omega \in (-\pi, \pi]$$

- Nonlinear relation between analog and digital frequencies
  - Need to pre-warp digital frequencies
- Magnitude levels unaffected
- No aliasing
  - Can design all filter types

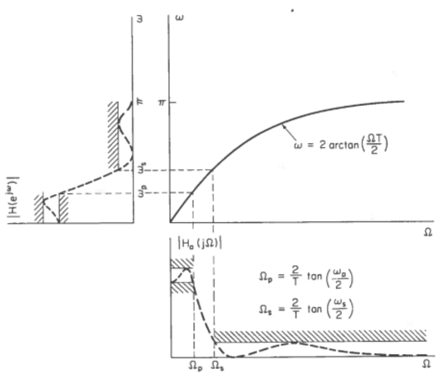


14

Bilinear transformation...

- Transformation of $H_a(s)$ to $H(z)$:

$$H(z) = H_a(s)|_{s=\frac{2}{T}\frac{1-z^{-1}}{1+z^{-1}}}$$



15

15

Bilinear transformation...

- Example: Transform $H_a(s) = \frac{s+1}{s^2+5s+6}$ into a digital filter using the bilinear transformation. You may choose $T = 1$

$$H(z) = H_a(s)|_{s=2\frac{1-z^{-1}}{1+z^{-1}}} = \frac{2\frac{1-z^{-1}}{1+z^{-1}}+1}{\left(2\frac{1-z^{-1}}{1+z^{-1}}\right)^2+5\left(2\frac{1-z^{-1}}{1+z^{-1}}\right)+6}$$

$$= \frac{2\frac{1-z^{-1}}{1+z^{-1}}+1}{\left(2\frac{1-z^{-1}}{1+z^{-1}}\right)^2+5\left(2\frac{1-z^{-1}}{1+z^{-1}}\right)+6} = \frac{3+2z^{-1}-z^{-2}}{20+4z^{-1}}$$

16

16

Example: Bandpass filter

- Design three digital IIR bandpass filters with resonance frequencies

$$\omega_{r_1} = \frac{\pi}{4}, \omega_{r_2} = \frac{\pi}{2}, \text{ and } \omega_{r_3} = \frac{3\pi}{4}$$

by converting the analog filter with system function

$$H_a(s) = \frac{s+0.1}{(s+0.1)^2+9}$$

using the the bilinear transformation

17

17

Example: Bandpass filter...

- Poles of analog filter

$$H_a(s) = \frac{s+0.1}{(s+0.1)^2+9} = \frac{s+0.1}{(s-[-0.1+j3])(s-[-0.1-j3])} = \frac{s+0.1}{(s-p_r)(s-p_r^*)}$$

reveals analog resonance frequency $\Omega_r = 3$ rad/s

- Use frequency relation between Ω and ω_{r_i} to obtain T_i

$$\Omega = \frac{2}{T_i}\tan\frac{\omega_i}{2} \Rightarrow T_i = \frac{2}{\Omega}\tan\frac{\omega_i}{2}$$

$$T_1 = \frac{2}{3}\tan\frac{\pi}{8}, T_2 = \frac{2}{3}, \text{ and } T_3 = \frac{2}{3}\tan\frac{\pi}{8}$$

18

18

Example: Bandpass filter...

- For each T_i apply the bilinear transform:

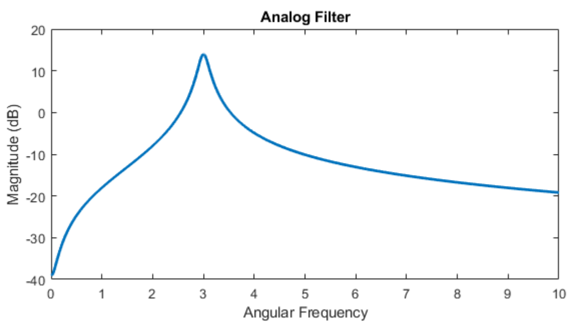
$$H_i(z) = H_a(s)\Big|_{s=\frac{2}{T_i}\frac{1-z^{-1}}{1+z^{-1}}} = \frac{\left(\frac{2}{T_i}\frac{1-z^{-1}}{1+z^{-1}}\right)+0.1}{\left(\frac{2}{T_i}\frac{1-z^{-1}}{1+z^{-1}}+0.1\right)^2+9}$$

$$= \frac{\left(2T_i+0.1T_i^2\right)+0.2T_i^2z^{-1}+\left(0.1T_i^2-2T_i\right)z^{-2}}{\left(4+0.4T+9.01T_i^2\right)+\left(18.02T_i^2-8\right)z^{-1}+\left(4-0.4T_i+9.01T_i^2\right)z^{-2}}$$

19

19

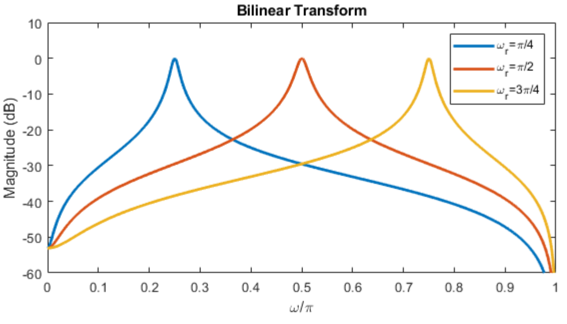
Example: Bandpass filter...



20

20

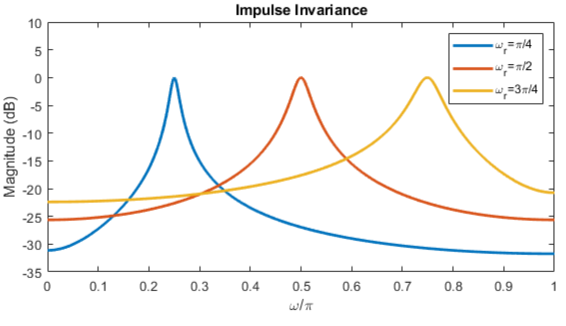
Example: Bandpass filter...



21

21

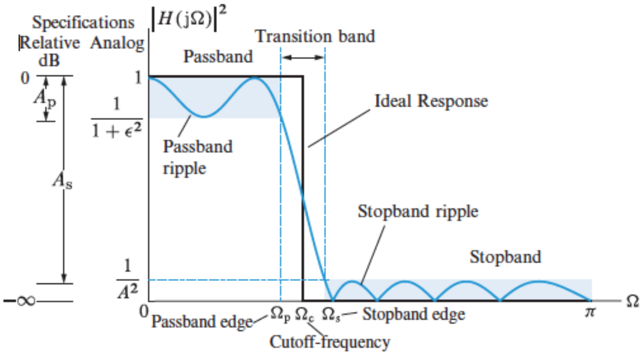
Example: Bandpass filter...



22

22

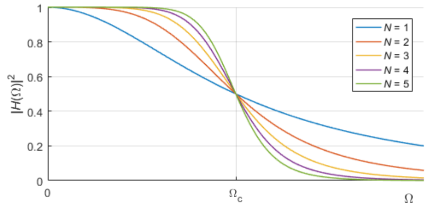
Analog filter specifications



Three classes of IIR filters

- Butterworth filters
  - In Matlab: `butter`
  - No ripples (oscillations) in $|H(\omega)|$, maximally flat
  - Smoothest transition from passband to stopband
- Chebyshev filters (two types)
  - `cheby1` and `cheby2` commands in Matlab
  - Ripples in either passband or stopband
- Elliptic filters
  - `ellip` in Matlab
  - Ripples in both passband and stopband
  - Sharpest transition from passband to stopband for a given order

Butterworth filter



- Frequency response:

$$|H(\Omega)|^2 = \frac{1}{1+(\Omega/\Omega_c)^{2N}} = \frac{1}{1+\epsilon^2(\Omega/\Omega_p)^{2N}}$$

- N poles on a circle with radius Ω_c in the s-plane
- Notice: $|H(0)|^2 = 1$, $|H(\Omega_c)|^2 = 0.5$ for all N

  $|H(\Omega)|^2$ monotonically decreasing
- Choose filter order depending on flatness of passband and how rapid decay in stopband

25

Butterworth filter...

- How to find $H(s)$:

$$|H(\Omega)|^2 = H(\Omega)H^*(\Omega) = H(s)H(-s)|_{s=j\Omega} = \left.\frac{1}{1+(-s^2/\Omega_c^2)^N}\right|_{s=j\Omega}$$

- Poles can be found from

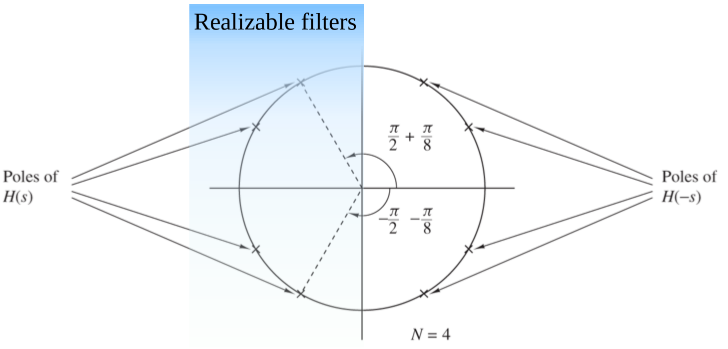
$$1 + \left(-p_k^2/\Omega_c^2\right)^N = 0 \;\Rightarrow p_k = \Omega_c e^{\frac{j\pi}{2}} e^{\frac{j(2k+1)\pi}{2N}}, k = 0, \dots, 2N-1$$

- Poles in $H(s)$: p_k in the left half-plane $k = 0, \dots, N-1$

$$H(s) = \frac{1}{(s-p_0)(s-p_1)\dots(s-p_{N-1})}$$

26

Butterworth filter...



$$p_k = \Omega_C e^{\frac{j\pi}{2}} e^{\frac{j(2k+1)\pi}{2N}}, k = 0, \dots, N-1$$

27

27

Butterworth filter...

- Example: Design a Butterworth filter, order $N = 2$, with half-power (digital) frequency $\omega_c = \frac{\pi}{4}$
- In analog frequency domain, $\omega_c = \frac{\pi}{4}$ corresponds to

$$\Omega = \frac{2}{T}\tan\frac{\pi}{8} = \frac{2(\sqrt{2}-1)}{T}$$

- Poles and system function:

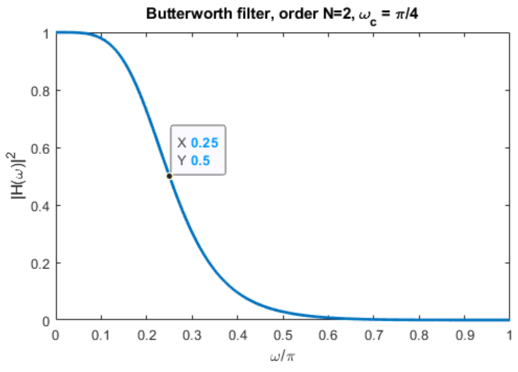
$$p_0 = \Omega_C e^{\frac{j\pi}{2}} e^{\frac{j\pi}{4}} = \frac{2(\sqrt{2}-1)}{T} e^{\frac{j3\pi}{4}}, p_1 = \Omega_C e^{\frac{j\pi}{2}} e^{\frac{j3\pi}{4}} = \frac{2(\sqrt{2}-1)}{T} e^{\frac{-j3\pi}{4}}$$

$$H(z) = \frac{1}{(s-p_0)(s-p_1)}\bigg|_{\frac{\frac{2}{T}(1-z^{-1})}{1+z^{-1}}} = \frac{T^2}{4} \cdot \frac{1+2z^{-1}+z^{-2}}{(6-3\sqrt{2}) - 4(\sqrt{2}-1)z^{-1} + (2-\sqrt{2})z^{-2}}$$

28

28

Butterworth filter...

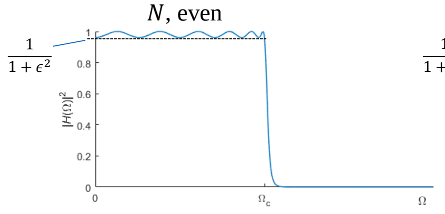


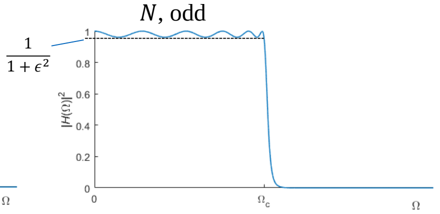

```
[B,A] = butter(2,1/4);
[H,w]=freqz(B,A,(0:pi/500:pi));
plot(w/pi,(abs(H)).^2/max(abs(H)).^2)
```

29

29

Chebyshev I





- Frequency response:

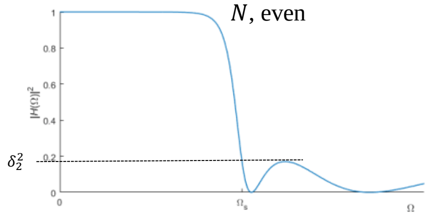
$$|H(\Omega)|^2 = \frac{1}{1+\epsilon^2 T_N^2(\Omega/\Omega_c)}, \ (T_N\ (x)\ N\text{th-order Chebyshev pol.})$$

- Parameter ϵ decides ripple in passband
- Poles lying on an ellipse in the s-plane

30

30

Chebyshev II



- Frequency response:

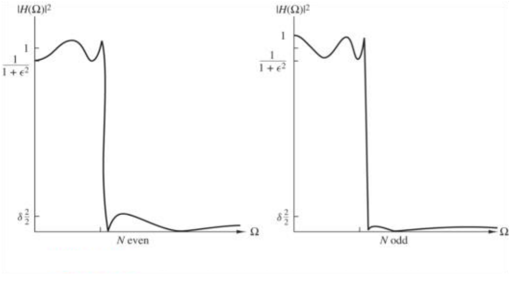
$$|H(\Omega)|^2 = \frac{1}{1+\epsilon^2\left[T_N^2(\Omega_s/\Omega_p)/T_N^2(\Omega_s/\Omega)\right]}$$

- Parameter ϵ decides ripple in stopband
- Poles on an ellipse in s-plane
- Zeros on the imaginary axis ($j\Omega$-axis)

31

31

Elliptic filter



- Frequency response:

$$|H(\Omega)|^2 = \frac{1}{1+\epsilon^2 U_N^2(\Omega/\Omega_c)}$$ ($U_N\ (x)$ Nth-order Jacobi elliptic function)

- Parameter ϵ decides ripple in passband
- Sharpest transition from passband to stopband among discussed filters
- Zeros on the imaginary axis ($j\Omega$-axis)

32

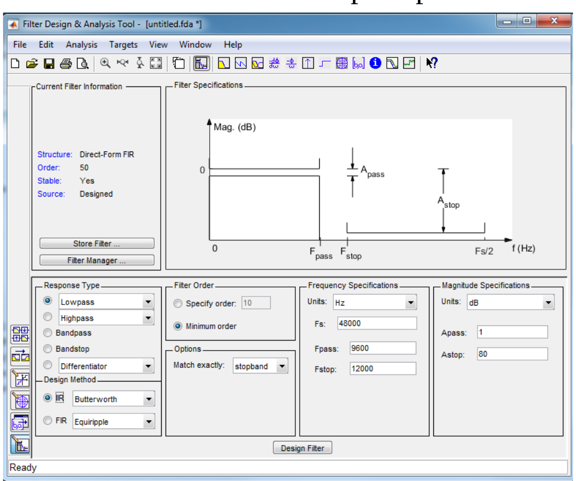
32

Summary

- Today we discussed:
  - IIR filter design
- Next:
  - Wiener filters

Matlab: fdatool

- Type `fdatool` at Matlab command prompt:



NTNU – Trondheim
Norwegian University of Science and Technology

TTT4120 Digital Signal Processing
Fall 2020

Wiener Filter Design

Prof. Stefan Werner
stefan.werner@ntnu.no
Office B329

Department of Electronic Systems

1

Lecture in course book*

- Proakis, Manolakis Digital Signal Processing, 4th Ed.
  - 12.7.1 FIR Wiener filter
  - 12.7.3 IIR Wiener filter
  - 12.7.4 Noncausal Wiener filter

- For a compressed overview of topics treated in the lecture, see "Wiener filter design" on Blackboard

*Level of detail is defined by lectures and problem sets

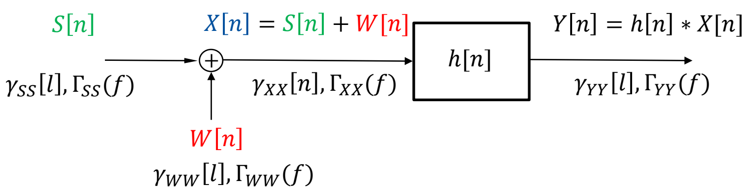
2

2

Contents and learning outcomes

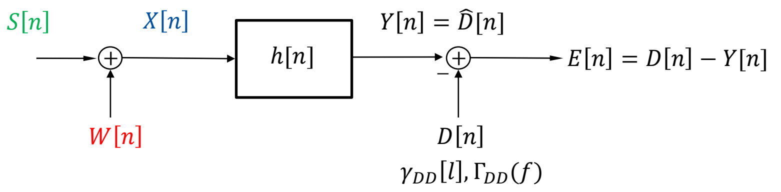
- Optimum MSE filter
  - Non-causal Wiener filter
  - Causal FIR Wiener filter
  - Causal IIR Wiener filter

Signal estimation



- Input signal $X[n]$ consists of a desired signal $S[n]$ and an undesired interference $W[n]$
- Design a filter $h[n]$ that suppress the undesired signal component
- Objective: Filter out the additive interference $W[n]$ while preserving the characteristics of desired signal $S[n]$
  - Interference suppression turns into the problem of signal estimation in the presence of noise

Signal estimation…

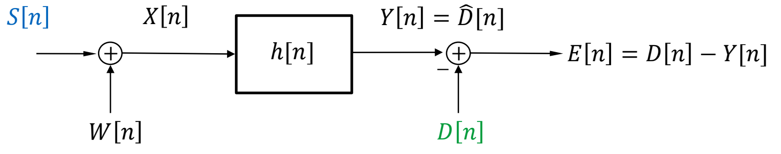


- Estimator is constrained to be a linear filter whose output approximates some desired signal sequence $D[n]$
  - Input to filter: $X[n] = S[n] + W[n]$
  - Sequence $S[n]$ stationary with known $\gamma_{SS}[l], \Gamma_{SS}(f)$
  - Sequence $D[n]$ stationary with known properties $\gamma_{DD}[l], \Gamma_{DD}(f)$
  - Sequence $W[n]$ white with known (or estimated) σ_W^2
- Error between $Y[n]$ and $D[n]$ measures similarity

5

5

Choice of target sequence $D[n]$



Three important choices of target sequence $D[n]$:

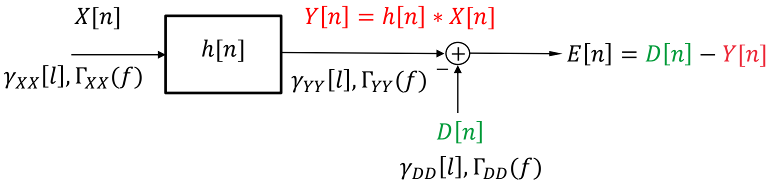
1. Noise reduction or filtering: $D[n] = S[n] \Longrightarrow \gamma_{DS}[l] = \gamma_{SS}[l]$
2. Smoothing: $D[n] = S[n - n_d], n_d > 0 \Longrightarrow \gamma_{DS}[l] = \gamma_{SS}[l - n_d]$
3. Prediction in noise: $D[n] = S[n + n_d], n_d > 0 \Rightarrow \gamma_{DS}[l] = \gamma_{SS}[l + n_d]$

- Remember definition: $\gamma_{DS}[l] = E\{D[n]S[n-l]\} = E\{D[n+l]S[n]\}$
  $= \gamma_{SD}[-l]$

6

6

Optimal MSE filtering



- Find filter $h[n]$ that minimizes mean-square error (MSE)

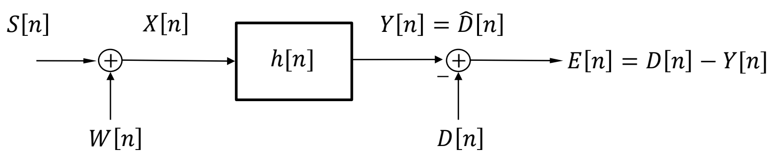
$$h_{\text{opt}}[n] = \arg\min_h E\{(D[n] - Y[n])^2\}$$

- Possible solutions depend on conditions set on filter $h[n]$
  - IIR and noncausal
  - IIR and causal, or FIR and causal

7

7

Optimum MSE noncausal IIR filter



- Filter $h[n]$ allowed to include both infinite past and infinite future of sequence $X[n]$ in forming output $Y[n]$

$$Y[n] = \sum_{k=-\infty}^{\infty} h[k]X[n-k]$$

- Filter $h[n]$ is unrealizable but serves as a best-case scenario
- Design filter to minimize $\sigma_E^2 = E\{(D[n] - Y[n])^2\}$

8

8

Optimum MSE noncausal IIR filter…

- Mean-square error (MSE);

$$\sigma_E^2 = E\{(D[n] - Y[n])^2\}$$
$$= E\{(D[n] - \sum_{k=-\infty}^{\infty} h[k]X[n-k])^2\}$$
$$= \gamma_{DD}[0] - 2\sum_{k=-\infty}^{\infty} h[k]\gamma_{DX}[k] +$$
$$+\sum_{k=-\infty}^{\infty}\sum_{l=-\infty}^{\infty} h[k]h[l]\gamma_{XX}[k-l]$$

- Minimum MSE (MMSE) when

$$\frac{d\sigma_E^2}{dh[k]} = 0, -\infty < k < \infty$$
$$\Longrightarrow \frac{d}{dh[l]} E\{(D[n] - \sum_{k=-\infty}^{\infty} h[k]X[n-k])^2\}$$
$$= E\{-2(D[n] - \sum_{k=-\infty}^{\infty} h[k]X[n-k])X[n-l]\} = 0$$

9

9

Optimum MSE noncausal IIR filter…

- Minimum MSE (MMSE) attained for $h[n]$ satisfying equation

$$\sum_{k=-\infty}^{\infty} h[k]\gamma_{XX}[l-k] = \gamma_{DX}[l], |l| \geq 0$$

- Minimum achievable MSE obtained by the above filter

$$\sigma_E^2 = \gamma_{DD}[0] - \sum_{k=-\infty}^{\infty} h[k]\gamma_{DX}[k]$$

- Equation system for $h[k]$ not solvable in time domain
- Take z-transform (or DTFT):

$$\Gamma_{DX}(z) = H(z)\Gamma_{XX}(z)$$
$$\boxed{\Longrightarrow H(z) = \frac{\Gamma_{DX}(z)}{\Gamma_{XX}(z)}}$$

10

10

Optimum MSE noncausal IIR filter…

- White noise $W[n]$ is uncorrelated with all other signals, i.e.,

$$\gamma_{XX}[l] = \gamma_{SS}[l] + \sigma_W^2 \delta[l], |l| \geq 0$$

$$\gamma_{DX}[l] = \gamma_{DS}[l], |l| \geq 0$$

- Optimal filter given by:

$$H(z) = \frac{\Gamma_{DS}(z)}{\Gamma_{SS}(z) + \sigma_W^2}$$

- Time-domain impulse response:

$$h[n] = \mathcal{Z}^{-1}\{H(z)\}$$

11

Optimum MSE noncausal IIR filter…

- Example: $X[n] = S[n] + W[n]$, and $W[n] \sim N(0, \sigma_W^2 = 1)$

  $S[n] = 0.6S[n-1] + N[n]$, and $N[n] \sim N(0, \sigma_N^2 = 0.64)$

  Design a noncausal IIR Wiener filter to estimate $S[n]$
- From earlier lectures:

$$\gamma_{SS}[l] = \frac{0.64}{1-0.6^2} 0.6^{|l|} = 0.6^{|l|} = \gamma_{DS}[l]$$

$$\Gamma_{SS}(z) = \frac{0.64}{(1-0.6z^{-1})(1-0.6z)} = \Gamma_{DS}(z)$$

$$\Gamma_{XX}(z) = \frac{0.64}{(1-0.6z^{-1})(1-0.6z)} + 1$$

- Optimum filter:

$$H(z) = \frac{\Gamma_{SS}(z)}{\Gamma_{XX}(z)} = -\frac{0.64}{0.6} \frac{z^{-1}}{\left(1-\frac{1}{3}z^{-1}\right)(1-3z^{-1})} = \frac{0.4}{\left(1-\frac{1}{3}z^{-1}\right)} + \frac{-0.4}{(1-3z^{-1})}$$

12

Optimum MSE noncausal IIR filter…

- Impulse response $h[n]$:

$$h[n] = \mathcal{Z}^{-1}\{H(z)\} = \mathcal{Z}^{-1}\left\{\frac{0.4}{\left(1-\frac{1}{3}z^{-1}\right)} + \frac{-0.4}{(1-3z^{-1})}\right\}$$

$$= \ 0.4\left(\frac{1}{3}\right)^{n} u[n] + 0.4 \cdot 3^{n} u[-n-1]$$

$$= 0.4\left(\frac{1}{3}\right)^{|n|}$$

- Minimum MSE

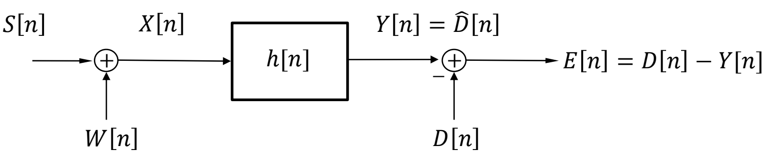
$$\sigma_E^2 = \gamma_{SS}[0] - \sum_{k=-\infty}^{\infty} h[k]\gamma_{SS}[k]$$

$$= 1 - \sum_{k=-\infty}^{\infty} 0.4\left(\frac{1}{3}\right)^{|k|} 0.6^{|k|} = 0.4$$

13

13

Optimum MSE causal FIR filter



- Filter $h[n]$ constrained to be causal and length M
- Output $Y[n]$ depends on $X[n], X[n-1], \ldots X[n-M+1]$

$$Y[n] = \sum_{k=0}^{M-1} h[k]X[n-k]$$

- Design filter to minimize $\sigma_E^2 = E\{(D[n] - Y[n])^2\}$

14

14

Optimum MSE causal FIR filter…

- Mean-square error (MSE);

$$\sigma_E^2 = E\{(D[n] - Y[n])^2\}$$

$$= E\left\{\left(D[n] - \sum_{k=0}^{M-1} h[k]X[n-k]\right)^2\right\}$$

$$= \gamma_{DD}[0] - 2\sum_{k=0}^{M-1} h[k]\gamma_{DX}[k] +$$

$$+\sum_{k=0}^{M-1}\sum_{l=0}^{M-1} h[k]h[l]\gamma_{XX}[k-l]$$

- Minimum MSE (MMSE) when

$$\frac{d\sigma_E^2}{dh[k]} = 0, 0 < k < M-1$$

15

15

Optimum MSE FIR filter

- Minimum MSE (MMSE) attained for $h[n]$ satisfying equation

$$\sum_{k=0}^{M-1} h[k]\gamma_{XX}[l-k] = \gamma_{DX}[l], l = 0, 1, \dots, M-1$$

- In matrix notation:

$$\underbrace{\begin{bmatrix} \gamma_{XX}[0] & \cdots & \gamma_{XX}[M-1] \\ \vdots & \ddots & \vdots \\ \gamma_{XX}[M-1] & \cdots & \gamma_{XX}[0] \end{bmatrix}}_{\boldsymbol{\Gamma}_{XX}} \underbrace{\begin{bmatrix} h[0] \\ \vdots \\ h[M-1] \end{bmatrix}}_{\mathbf{h}} = \underbrace{\begin{bmatrix} \gamma_{DX}[0] \\ \vdots \\ \gamma_{DX}[M-1] \end{bmatrix}}_{\boldsymbol{\gamma}_{DX}}$$

where $M \times M$ autocorrelation matrix $(\boldsymbol{\Gamma}_{XX})_{lk} = \gamma_{XX}[l-k]$ and $M \times 1$ cross-correlation vector $(\boldsymbol{\gamma}_{DX})_l = \gamma_{DX}[l]$

- Minimum MSE: $\sigma_E^2 = \gamma_{DD}[0] - \sum_{k=0}^{M-1} h[k]\gamma_{DX}[k]$

16

16

Optimum MSE FIR filter

- Can be solved directly in time-domain

$$\mathbf{h} = \mathbf{\Gamma}_{XX}^{-1}\boldsymbol{\gamma}_{DX}$$

- Matrix $\mathbf{\Gamma}_{XX}$ symmetric and Toeplitz $\Longrightarrow$ Efficient algorithms exist
- Minimum achievable MSE obtained by above filter

$$\sigma_E^2 = \gamma_{DD}[0] - \sum_{k=0}^{M-1} h[k]\gamma_{DX}[k]$$
$$= \gamma_{DD}[0] - \mathbf{h}^{\mathrm{T}}\boldsymbol{\gamma}_{DX}$$

- FIR filters are popular for signal estimation as they can be adapted continuously in dynamic environments (adaptive filters)

17

17

Optimum MSE FIR filter…

- Example: $X[n] = S[n] + W[n]$, and $W[n]\sim N(0, \sigma_W^2 = 1)$

  $S[n] = 0.6S[n-1] + N[n]$, and $N[n]\sim N(0, \sigma_N^2 = 0.64)$

  Design FIR filter with $M = 2$ coefficients

- From before $\gamma_{SS}[l] = 0.6^{|l|} = \gamma_{DX}[l], \gamma_{XX}[l] = \gamma_{SS}[l] + \delta[l]$

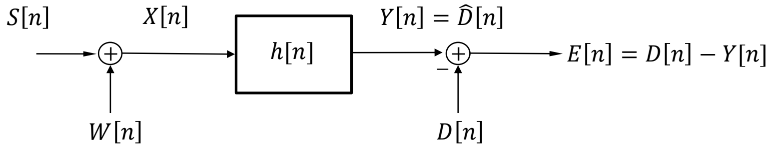
$$\begin{bmatrix}\gamma_{XX}[0] & \gamma_{XX}[1]\\ \gamma_{XX}[1] & \gamma_{XX}[0]\end{bmatrix}\begin{bmatrix}h[0]\\ h[1]\end{bmatrix} = \begin{bmatrix}\gamma_{DX}[0]\\ \gamma_{DX}[1]\end{bmatrix} \Rightarrow \begin{bmatrix}2 & 0.6\\ 0.6 & 2\end{bmatrix}\begin{bmatrix}h[0]\\ h[1]\end{bmatrix} = \begin{bmatrix}1\\ 0.6\end{bmatrix}$$

- We get $h[0] = 0.451$ and $h[1] = 0.165$
- Minimum MSE: $\sigma_E^2 = 1 - \sum_{k=0}^{1} h[k]\gamma_{XX}[k] = 0.45$

18

18

Optimum MSE causal IIR filter



- Filter $h[n]$ constrained to be causal but can be of infinite duration
- Output $Y[n]$ depends on $X[n], X[n-1], \ldots$

$$Y[n] = \sum_{k=0}^{\infty} h[k]X[n-k]$$

- Design filter to minimize $\sigma_E^2 = E\{(D[n] - Y[n])^2\}$

19

19

Optimum MSE IIR causal filter…

- Mean-square error (MSE):

$$\sigma_E^2 = E\{(D[n] - Y[n])^2\}$$

$$= E\{(D[n] - \sum_{k=0}^{\infty} h[k]X[n-k])^2\}$$

$$= \gamma_{DD}[0] - 2\sum_{k=0}^{\infty} h[k]\gamma_{DX}[k] +$$

$$+ \sum_{k=0}^{\infty}\sum_{l=0}^{\infty} h[k]h[l]\gamma_{XX}[k-l]$$

- Minimum MSE when

$$\frac{d\sigma_E^2}{dh[k]} = 0, k = 0, 1, \ldots$$

20

20

Optimum MSE IIR causal filter…

- Minimum MSE (MMSE) attained for $h[n]$ satisfying equation

$$\sum_{k=0}^{\infty} h[k]\gamma_{XX}[l-k] = \gamma_{DX}[l], l \geq 0$$

- Minimum achievable MSE obtained by above filter

$$\sigma_E^2 = \gamma_{DD}[0] - \sum_{k=0}^{\infty} h[k]\gamma_{DX}[k]$$

- Cannot directly solve for $h[k]$ using z-transform, since equations only consider $l \geq 0$
- Alternative solution via the innovations representation of $X[n]$

21

21

Optimum MSE IIR causal filter…

- Definition: Let $[A(z)]_+$ denote the causal part of $A(z)$, i.e.,

$$A(z) = \sum_{k=-\infty}^{\infty} a[k]z^{-k} \Rightarrow [A(z)]_+ = \sum_{k=0}^{\infty} a[k]z^{-k}$$

- Example: $A(z) = \frac{1}{1-0.5z^{-1}} + \frac{0.5z}{1-0.5z}$, ROC: $0.5 < |z| < 2$

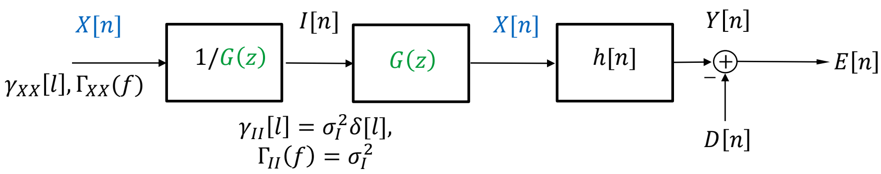
$$\Rightarrow [A(z)]_+ = \sum_{k=0}^{\infty} a[k]z^{-k} = \frac{1}{1-0.5z^{-1}}, \quad \text{ROC: } |z| > 0.5$$

$$a[n] = \left(\frac{1}{2}\right)^n u[n]$$

22

22

Optimum MSE IIR causal filter…

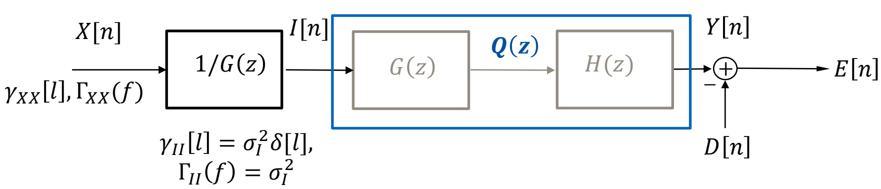


- Express $\Gamma_{XX}(z) = \sigma_I^2 G(z)G(z^{-1})$ with $G(z)$ being minimum-phase
  - Remember definition that $G(z)$ *causal and stable with causal and stable inverse* $1/G(z) \Rightarrow G(z)$ must be minimum-phase
- Use the innovations representation of $X[n]$ to simplify the design

23

23

Optimum MSE IIR causal filter…



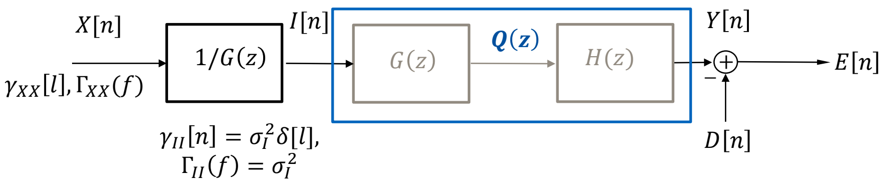
- Study system $Q(z) = G(z)H(z)$ with input $I[n]$ and derive optimal $Q(z)$ using the MSE formulation
- Once $Q(z)$ obtained we can get $H(z)$ from relation

$$H(z) = Q(z)/G(z)$$

24

24

Optimum MSE IIR causal filter…



- Filter $q[n]$ constrained to be causal but can be of infinite duration
- Output $Y[n]$ depends on $I[n], I[n-1], \ldots$

$$Y[n] = \sum_{k=0}^{\infty} q[k]I[n-k]$$

- Design filter to minimize $\sigma_E^2 = E\{(D[n]-Y[n])^2\}$

25

25

Optimum MSE IIR causal filter…

- Minimum MSE (MMSE) attained for $q[n]$ satisfying equation

$$\sum_{k=0}^{\infty} q[k]\gamma_{II}[l-k] = \gamma_{DI}[l], l \geq 0$$

- We know that $I[n]$ is white noise with $\gamma_{II}[l] = \sigma_I^2\delta[l]$

$$q[l]\gamma_{II}[0] = \gamma_{DI}[l], l \geq 0$$

$$\Rightarrow q[l] = \frac{\gamma_{DI}[l]}{\gamma_{II}[0]} = \frac{\gamma_{DI}[l]}{\sigma_I^2}, l \geq 0$$

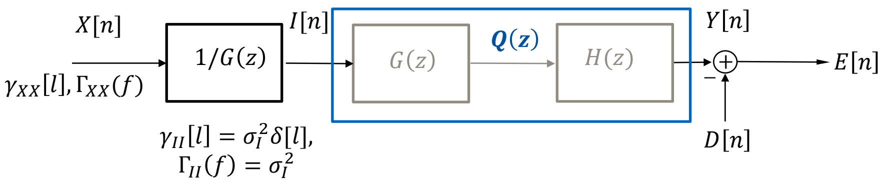
- Coefficients of filter $q[n]$ is related to $\Gamma_{DI}(z)$ as

$$Q(z) = \sum_{k=0}^{\infty} q[k]z^{-k} = \frac{1}{\sigma_I^2}\sum_{k=0}^{\infty} \gamma_{DI}[k]z^{-k} = \frac{1}{\sigma_I^2}[\Gamma_{DI}(z)]_+$$

26

26

Optimum MSE IIR causal filter…



- To find $[\Gamma_{DI}(z)]_+$ we express $I[n]$ in terms of $X[n]$
- Let $v[n]$ denote the impulse response of $1/G(z)$

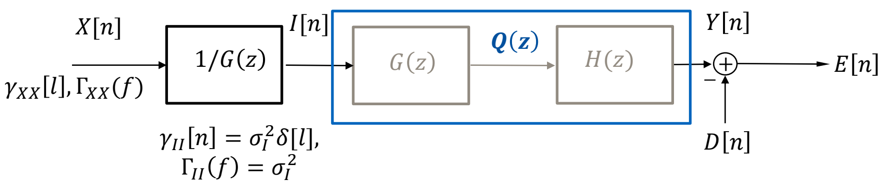
$$I[n] = \sum_{k=0}^{\infty} v[k]X[n-k]$$

$$\gamma_{DI}[l] = E\{D[n]I[n-l]\} = \sum_{k=0}^{\infty} v[k]E\{D[n]X[n-k-l]\}$$

$$= \sum_{k=0}^{\infty} v[k]\gamma_{DX}[k+l]$$

27

Optimum MSE IIR causal filter…



- $\Gamma_{DI}(z)$ in terms of $X[n]$

$$\Gamma_{DI}(z) = \sum_{l=-\infty}^{\infty} \gamma_{DI}[l]z^{-l} = \sum_{l=-\infty}^{\infty}\left(\sum_{k=0}^{\infty} v[k]\gamma_{DX}[k+l]\right)z^{-l}$$

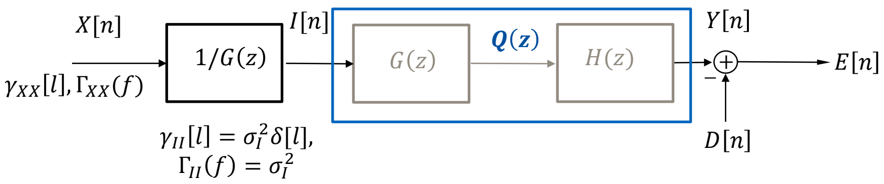
$$= V(z^{-1})\Gamma_{DX}(z) = \Gamma_{DX}(z)/G(z^{-1})$$

- Consequently

$$\boxed{H_{opt}(z) = \frac{Q(z)}{G(z)} = \frac{\frac{1}{\sigma_I^2}[\Gamma_{DI}(z)]_+}{G(z)} = \frac{1}{\sigma_I^2 G(z)}\left[\frac{\Gamma_{DX}(z)}{G(z^{-1})}\right]_+}$$

28

Optimum MSE IIR causal filter…



- Summary of steps:
  1. Express $\Gamma_{XX}(z)$ as $\Gamma_{XX}(z) = \sigma_I^2 G(z) G(z^{-1})$
  2. Compute $H_{opt}(z) = \frac{1}{\sigma_I^2 G(z)} \left[\frac{\Gamma_{DX}(z)}{G(z^{-1})} \right]_+$

29

Optimum MSE IIR causal filter…

- Example: $X[n] = S[n] + W[n]$, and $W[n] \sim N(0, \sigma_W^2 = 1)$

  $S[n] = 0.6S[n-1] + N[n]$, and $N[n] \sim N(0, \sigma_N^2 = 0.64)$

  Design a causal IIR Wiener filter to estimate $S[n]$
- From before:

$$\gamma_{SS}[l] = 0.6^{|l|} = \gamma_{DX}[l]$$

$$\Gamma_{SS}(z) = \frac{0.64}{(1-0.6z^{-1})(1-0.6z)} = \Gamma_{DX}(z) = \Gamma_{DS}(z)$$

$$\Gamma_{XX}(z) = \Gamma_{SS}(z) + \Gamma_{WW}(z)$$

$$= \frac{1.8\left(1-\frac{1}{3}z^{-1}\right)\left(1-\frac{1}{3}z\right)}{(1-0.6z^{-1})(1-0.6z)} = \sigma_I^2 G(z) G(z^{-1})$$

30

Optimum MSE IIR causal filter…

- System function of optimal IIR filter:

$$H_{opt}(z) = \frac{1}{\sigma_I^2 G(z)}\left[\frac{\Gamma_{DX}(z)}{G(z^{-1})}\right]_+ = \frac{(1-0.6z^{-1})}{1.8\left(1-\frac{1}{3}z^{-1}\right)}\left[\frac{0.64(1-0.6z)}{(1-0.6z^{-1})(1-0.6z)\left(1-\frac{1}{3}z\right)}\right]_+$$

$$= \frac{(1-0.6z^{-1})}{1.8\left(1-\frac{1}{3}z^{-1}\right)}\left[\frac{0.8}{(1-0.6z^{-1})} + \frac{0.266z}{\left(1-\frac{1}{3}z\right)}\right]_+ = \frac{(1-0.6z^{-1})}{1.8\left(1-\frac{1}{3}z^{-1}\right)}\frac{0.8}{(1-0.6z^{-1})}$$

$$= \frac{4}{9}\frac{1}{1-\frac{1}{3}z^{-1}}$$

- Impulse response:

$$h[n] = \frac{4}{9}\left(\frac{1}{3}\right)^n u[n]$$

Optimum MSE IIR causal filter…

- Minimum MSE:

$$\sigma_E^2 = \gamma_{DD}[0] - \sum_{k=0}^{\infty} h[k]\gamma_{DX}[k]$$

with

$$\gamma_{SS}[l] = 0.6^{|l|} = \gamma_{DX}[l]$$

$$h[n] = \frac{4}{9}\left(\frac{1}{3}\right)^n u[n]$$

we finally obtain

$$\sigma_E^2 = 1 - \frac{4}{9}\sum_{k=0}^{\infty} 0.6^k \left(\frac{1}{3}\right)^k = 1 - \frac{4}{9}\sum_{k=0}^{\infty}\left(\frac{1}{5}\right)^k = \frac{4}{9} \approx 0.44$$

Summary

- Today we discussed:
  – Wiener filters (noncausal and causal design)
- Next:
  – Filter implementation