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

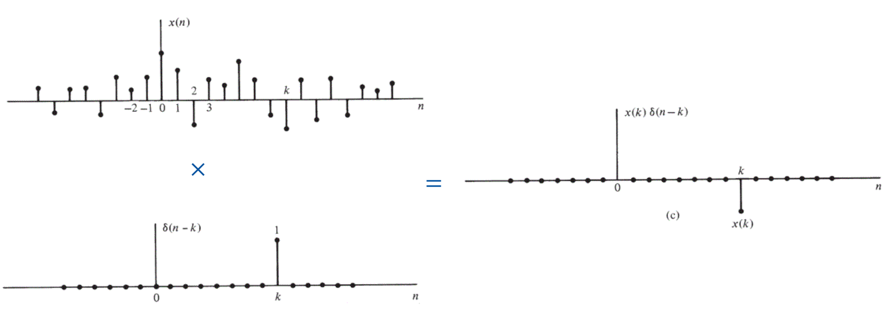
2

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

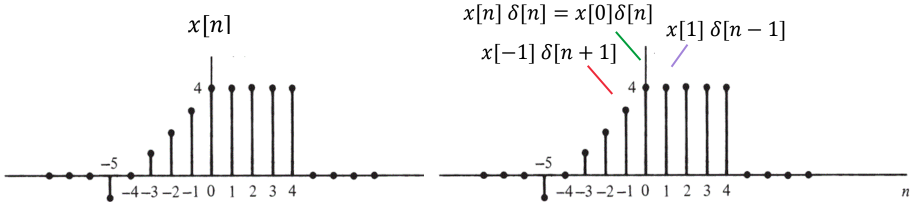
Signal decomposition using unit impulses

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



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 $x[k]$



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

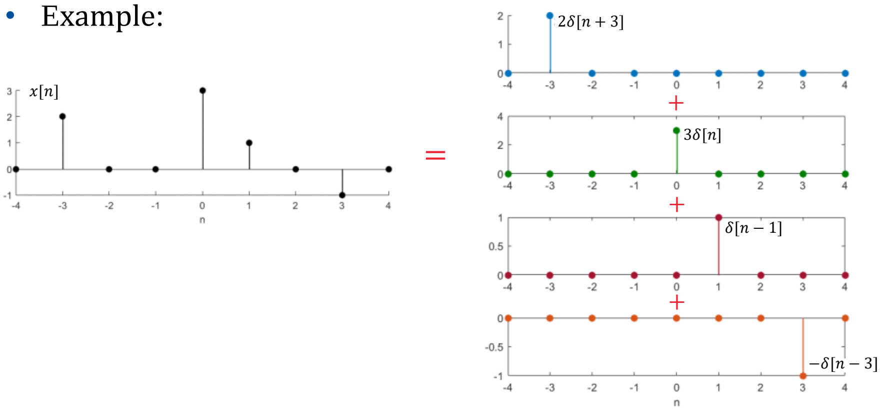
$$\boxed{= \sum_{k=-\infty}^{\infty} x[k]\delta[n-k]}$$

5

5

Signal decomposition using unit impulses…

- Example:



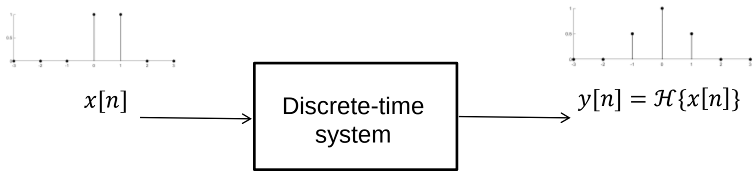
$$x[n] = 2\delta[n+3] + 3\delta[n] + \delta[n-1] - \delta[n-3]$$

6

6

Discrete-time systems

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



- Mathematically we have

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

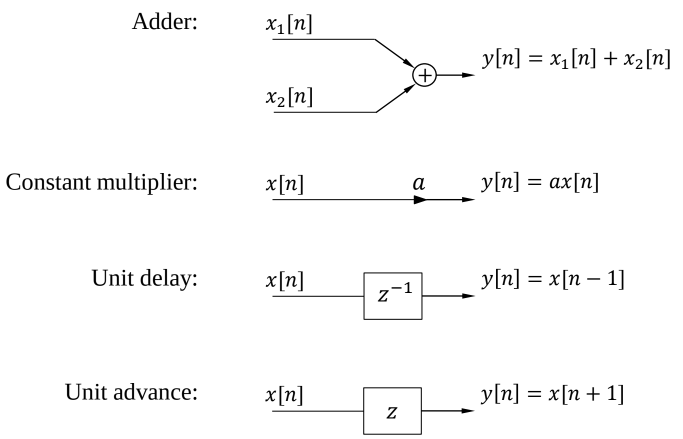
where operator $\mathcal{H}$ describes the discrete-time system

7

7

Discrete-time systems…

- Graphical representation of building blocks



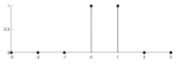
8

8

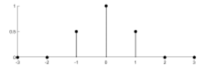
Classification of discrete-time systems

9

Classification of discrete-time systems



$x[n]$ → Discrete-time system → $y[n] = \mathcal{H}\{x[n]\}$

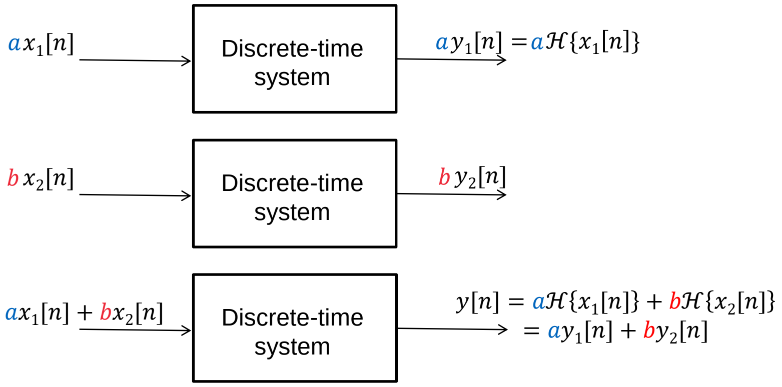


- A discrete-time system can be classified as:
  - linear or nonlinear
  - time invariant or time variant
  - causal or noncausal
- Property must hold for every possible input to the system
  - to disprove a property, need a single counter-example
  - to prove a property, need to prove for the general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

10

10

Linear discrete-time systems



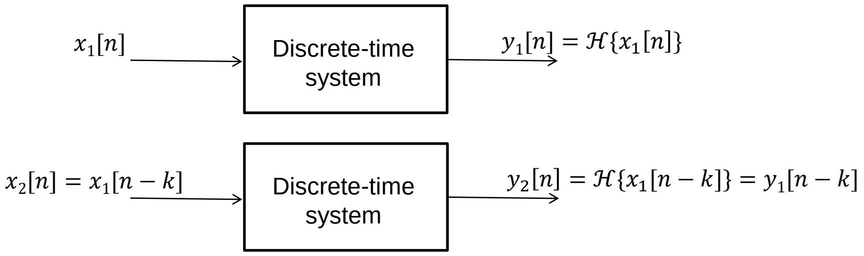
- A linear system is a system for which superposition holds

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. $y[n] = cx[n] + 3$

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.,

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

- 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

15

15

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. $y[n] = cx[n] + 3$

16

16

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

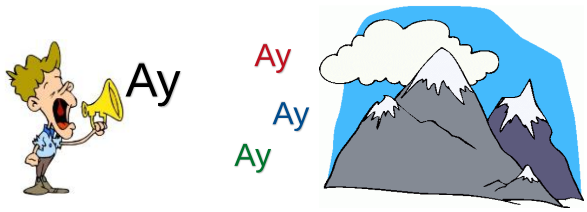
- This course is mostly dealing with linear time-invariant systems
- Knowing the system response to a unit impulse (impulse response), we can calculate the system output for an arbitrary input signal

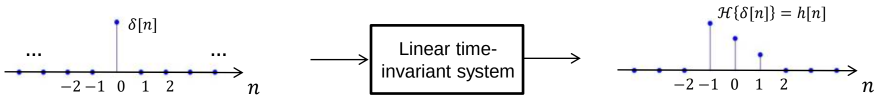
20

20

Impulse response

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

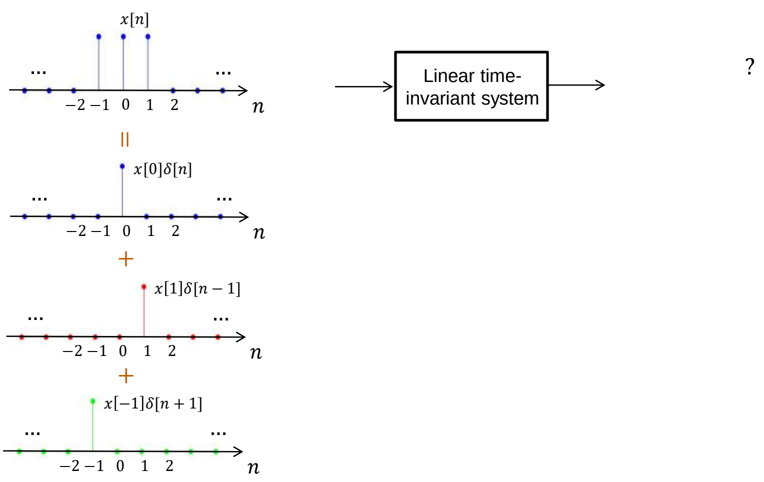




21

21

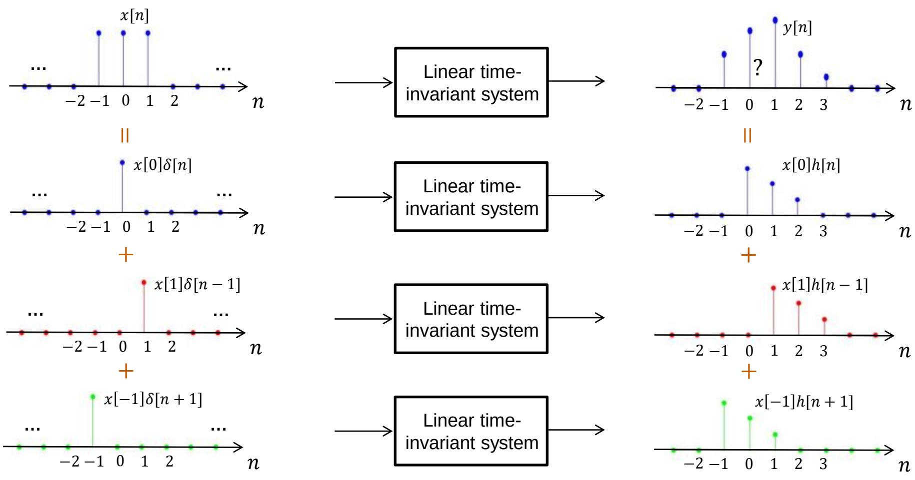
Impulse response and convolution sum



22

22

Impulse response and convolution sum



23

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

Convolution sum

- More formally,

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

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

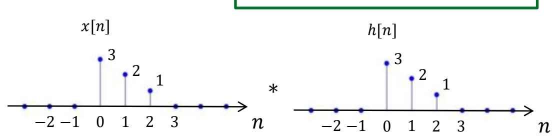
$$\boxed{= \sum_k x[k]h[n-k]} = x[n] * h[n]$$

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

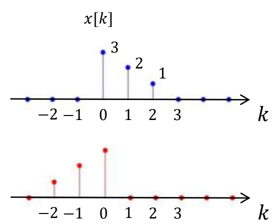
Example: Flip-shift-multiply-sum

- What is the output?

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



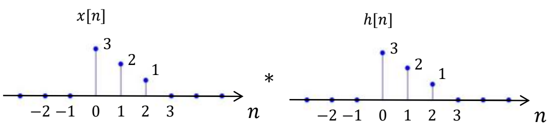
$$y[0] = \sum_k x[k]h[-k]$$



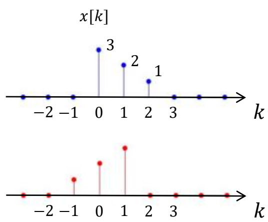
Example: Flip-shift-multiply-sum…

- What is the output?

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



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



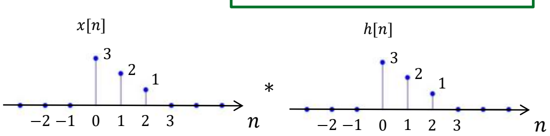
27

27

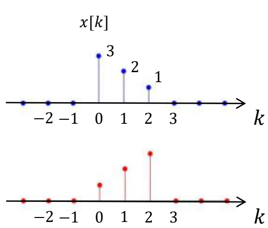
Example: Flip-shift-multiply-sum…

- What is the output?

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



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



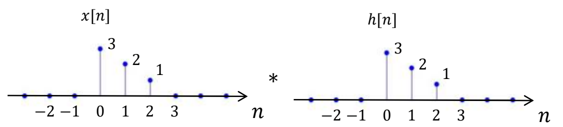
28

28

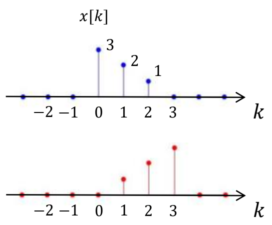
Example: Flip-shift-multiply-sum…

- What is the output?

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



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



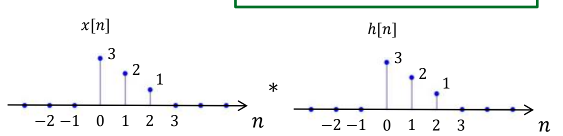
29

29

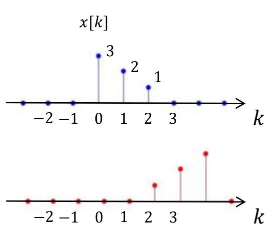
Example: Flip-shift-multiply-sum…

- What is the output?

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



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



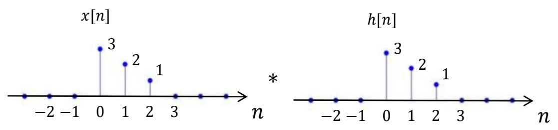
30

30

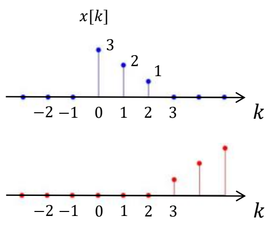
Example: Flip-shift-multiply-sum…

- What is the output?

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



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



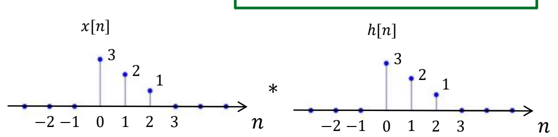
31

31

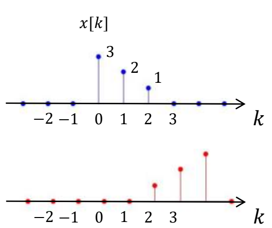
Example: Flip-shift-multiply-sum…

- What is the output?

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



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



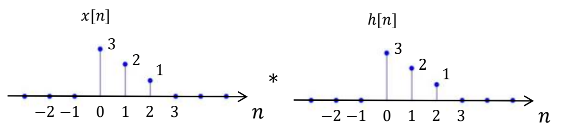
32

32

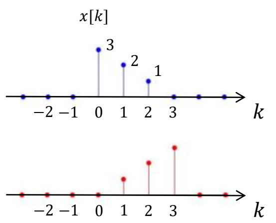
Example: Flip-shift-multiply-sum…

- What is the output?

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



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



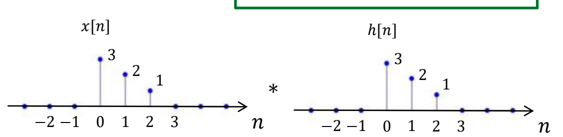
33

33

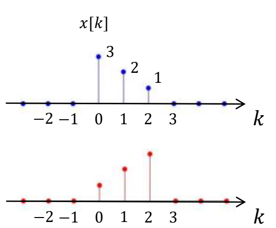
Example: Flip-shift-multiply-sum…

- What is the output?

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



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



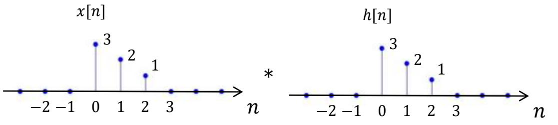
34

34

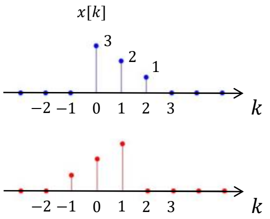
Example: Flip-shift-multiply-sum…

- What is the output?

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



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



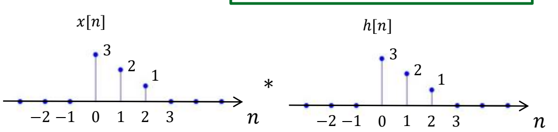
35

35

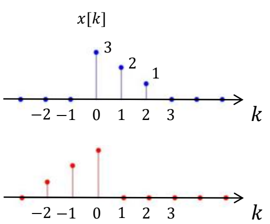
Example: Flip-shift-multiply-sum…

- What is the output?

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



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



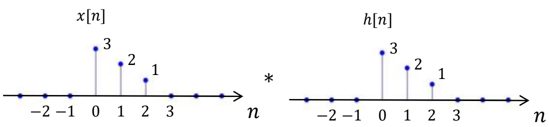
36

36

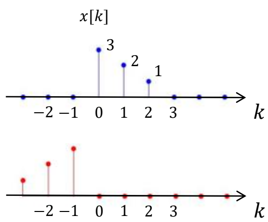
Example: Flip-shift-multiply-sum…

- What is the output?

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



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



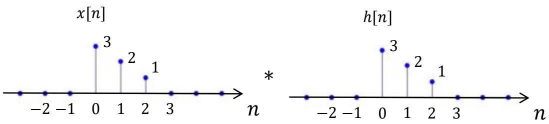
37

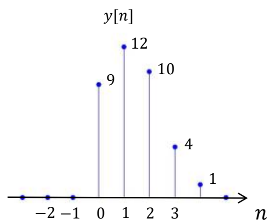
37

Example: Flip-shift-multiply-sum…

- What is the output?

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





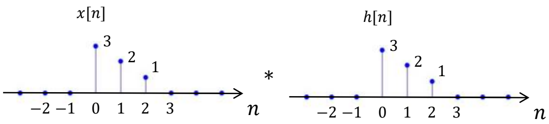
38

38

Example: the easier way

- What is the output?

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



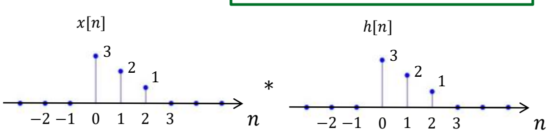
- Convolution matrix: multiply and sum anti-diagonals

$x[n]$ \ $h[n]$	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

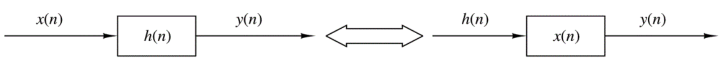
Matlab
```
x = [3 2 1];
h = [3 2 1];
y = conv(x,h)
```

Properties of convolution

- Commutative:

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

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



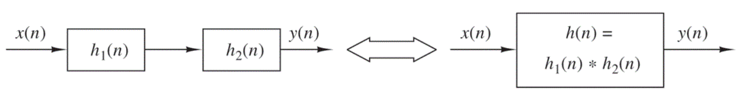
41

41

Properties of convolution…

- Associative:

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



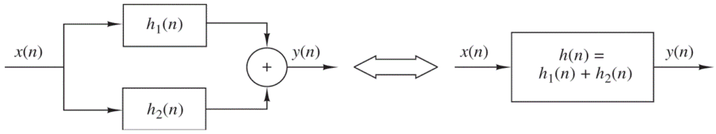
42

42

Properties of convolution…

- Distributive:

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



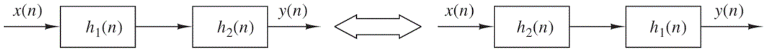
43

43

Properties of convolution…

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

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



44

44

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

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

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 $h[n] = b_n, 0 \leq n \leq M$

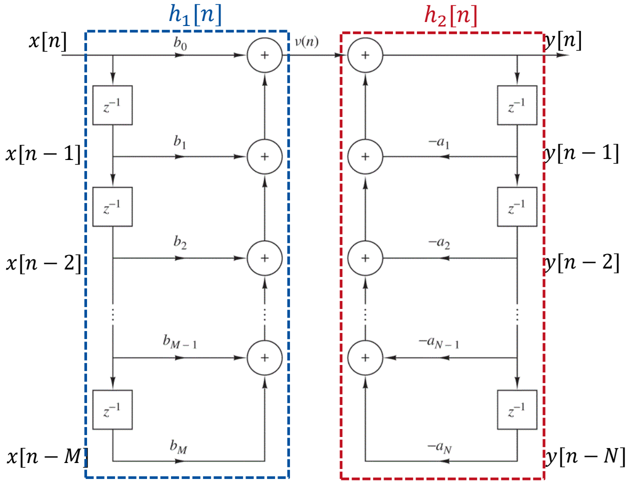
Matlab
```
b = [b0, b1,…,bM];
a = [a0,a1,…,aN];
y = filter(b,a,x)
```

49

49

Systems described by difference…

- Graphical representation: Direct form I structure

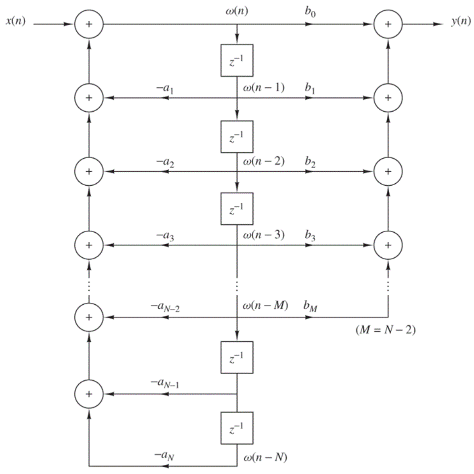


50

50

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 $n = 1,2, \dots$
- 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

Summary

Today:

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

Next:

- Discrete-time Fourier transform

54

Audio demo: a tiger in a cathedral

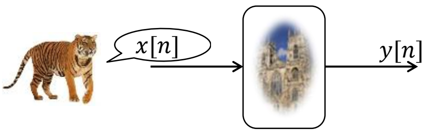
How does it sound when a tiger roars in York Minster?

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


55

Audio demo: a tiger in a cathedral…

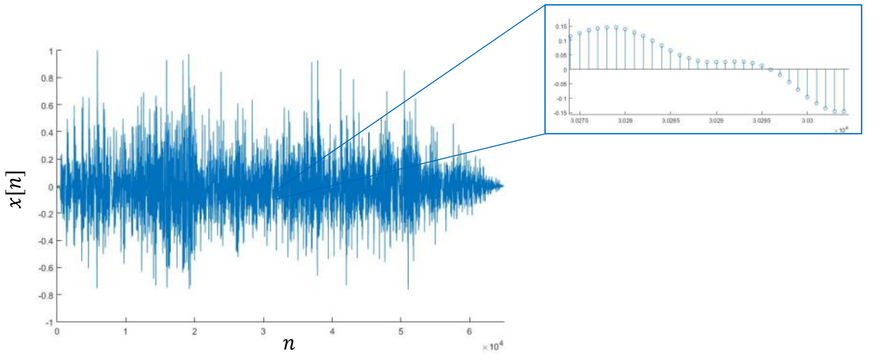
- The output of a linear time-invariant system is obtained by convolving its *impulse response* with the *input signal*



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

56

The tiger growl*



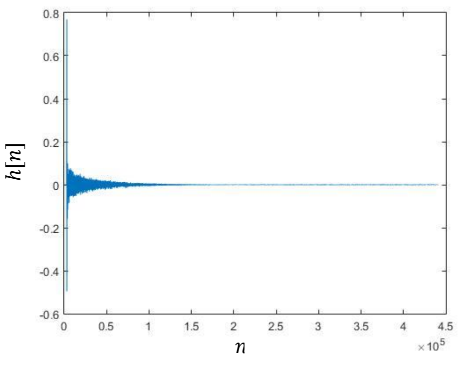
- Tiger growl sampled intervals $nT = \frac{n}{f_s} = n/44100s$

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

57

57

Impulse reponse of York Minster*



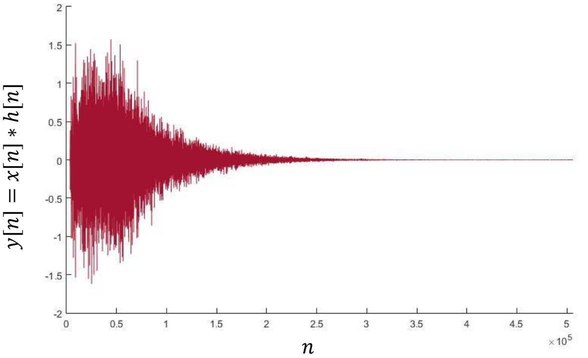
- Impulse response sampled intervals $nT = \frac{n}{f_s} = n/44100$ s

* 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