

Digital Signal Processing, Fall 2006

Lecture 1: Introduction, Discrete-time signals and systems

Zheng-Hua Tan

Department of Electronic Systems
Aalborg University, Denmark
zt@kom.aau.dk

Part I: Introduction

- Introduction (Course overview)
- Discrete-time signals
- Discrete-time systems
- Linear time-invariant systems

General information

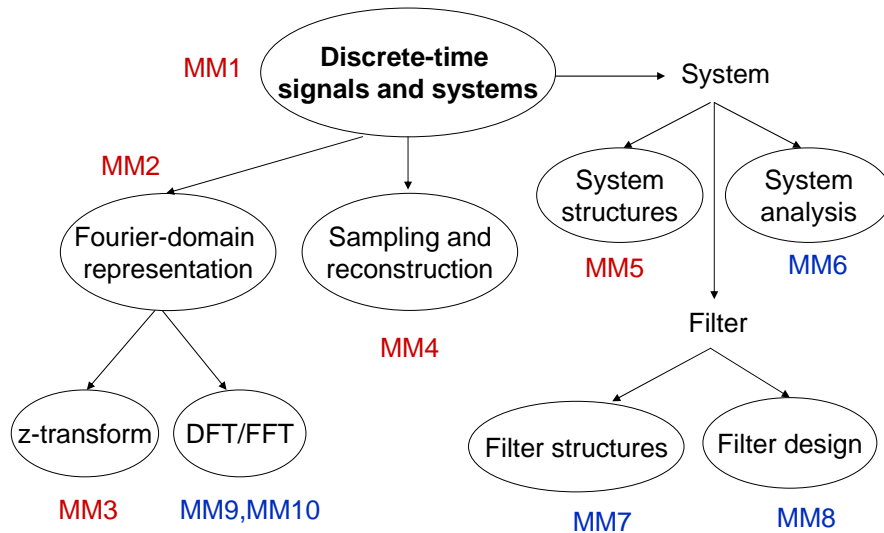
- Course website
 - <http://kom.aau.dk/~zt/courses/DSP/>
- Textbook:
 - Oppenheim, A.V., Schafer, R.W, "**Discrete-Time Signal Processing**", 2nd Edition, Prentice-Hall, 1999.
- Readings:
 - Steven W. Smith, "The Scientist and Engineer's Guide to Digital Signal Processing", California Technical Publishing, 1997. <http://www.dspguide.com/pdfbook.htm> (You can download the entire book!)
 - Kermit Sigmon, "Matlab Primer", Third Edition, Department of Mathematics, University of Florida.
 - V.K. Ingle and J.G. Proakis, "Digital Signal Processing using MATLAB", Bookware Companion Series, 2000.

General information

- Duration
 - 2 ECTS (10 Lectures)
- Prerequisites:
 - Background in advanced calculus including complex variables, Laplace- and Fourier transforms.
- Course type:
 - Study programme course (SE-course) , meaning a written exam at the end of the semester!
- Lecturer:

Associate Professor, PhD, Zheng-Hua Tan
Niels Jernes Vej 12, A6-319
zt@kom.aau.dk, +45 9635-8686

Course at a glance



5 Digital Signal Processing, I, Zheng-Hua Tan, 2006



Course objectives (Part I)

- To give the students a comprehension of the concepts of discrete-time signals and systems
- To give the students a comprehension of the Z- and the Fourier transform and their inverse
- To give the students a comprehension of the relation between digital filters, difference equations and system functions
- To give the students knowledge about the most important issues in sampling and reconstruction

6 Digital Signal Processing, I, Zheng-Hua Tan, 2006



Course objectives (Part II)

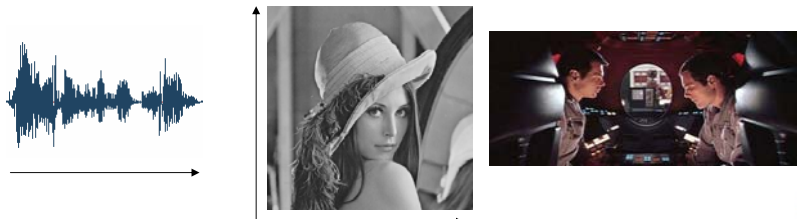
- To make the students able to apply digital filters according to known filter specifications
- To provide the knowledge about the principles behind the discrete Fourier transform (DFT) and its fast computation
- To make the students able to apply Fourier analysis of stochastic signals using the DFT
- To be able to apply the MATLAB program to digital processing problems and presentations

What is a signal ?

- A flow of information.
- (mathematically represented as) a function of independent variables such as time (e.g. speech signal), position (e.g. image), etc.
- A common convention is to refer to the independent variable as time, although may in fact not.

Example signals

- Speech: 1-Dimension signal as a function of time $s(t)$;
- Grey-scale image: 2-Dimension signal as a function of space $i(x,y)$
- Video: 3 x 3-Dimension signal as a function of space and time $\{r(x,y,t), g(x,y,t), b(x,y,t)\}$.



9 Digital Signal Processing, I, Zheng-Hua Tan, 2006

AALBORG UNIVERSITY

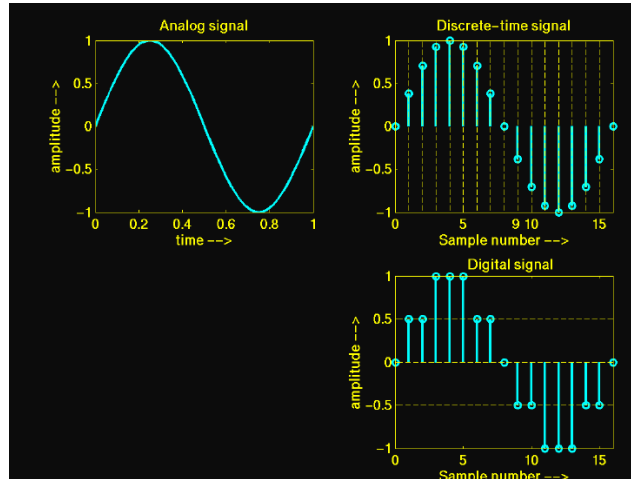
Types of signals

- The independent variable may be either continuous or discrete
 - Continuous-time signals
 - Discrete-time signals are defined at discrete times and represented as sequences of numbers
- The signal amplitude may be either continuous or discrete
 - Analog signals: both time and amplitude are continuous
 - Digital signals: both are discrete
- Computers and other digital devices are restricted to discrete time
- Signal processing systems classification follows the same lines

10 Digital Signal Processing, I, Zheng-Hua Tan, 2006

AALBORG UNIVERSITY

Types of signals



From <http://www.ece.rochester.edu/courses/ECE446>

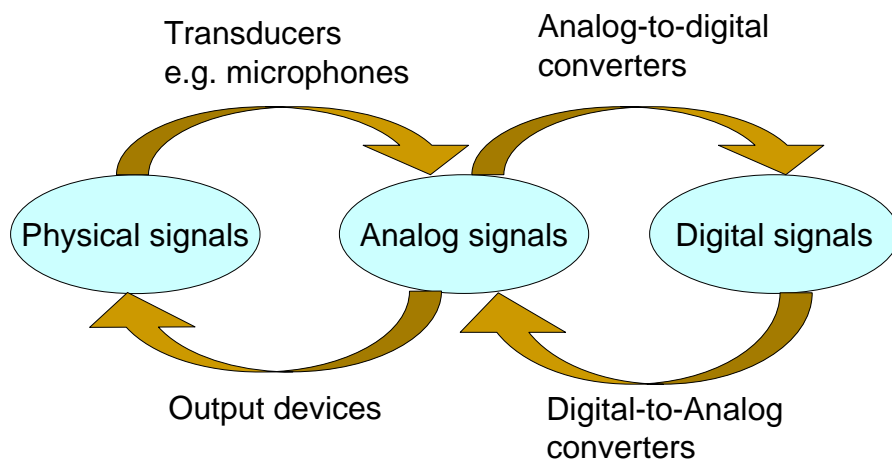
Digital signal processing

- Modifying and analyzing information with computers – so being measured as sequences of numbers.
- Representation, transformation and manipulation of signals and information they contain

Typical DSP system components

- Input lowpass filter to avoid aliasing
- Analog to digital converter (ADC)
- Computer or DSP processor
- Digital to analog converter (DAC)
- Output lowpass filter to avoid imaging

ADC and DAC



Pros and cons of DSP

- Pros
 - Easy to duplicate
 - Stable and robust: not varying with temperature, storage without deterioration
 - Flexibility and upgrade: use a general computer or microprocessor
- Cons
 - Limitations of ADC and DAC
 - High power consumption and complexity of a DSP implementation: unsuitable for simple, low-power applications
 - Limited to signals with relatively low bandwidths

Applications of DSP

- Speech processing
 - Enhancement – noise filtering
 - Coding
 - Text-to-speech (synthesis)
 - 🔊 Next generation TTS @ AT&T
 - Recognition
- Image processing
 - Enhancement, coding, pattern recognition (e.g. OCR)
- Multimedia processing
 - Media transmission, digital TV, video conferencing
- Communications
- Biomedical engineering
- Navigation, radar, GPS
- Control, robotics, machine vision

History of DSP

- Prior to 1950's: analog signal processing using electronic circuits or mechanical devices
- 1950's: computer simulation before analog implementation, thus cheap to try out
- 1965: Fast Fourier Transforms (FFTs) by Cooley and Tukey – make real time DSP possible
- 1980's: IC technology boosting DSP

Part II: Discrete-time signals

- Introduction
- Discrete-time signals
- Discrete-time systems
- Linear time-invariant systems

Discrete-time signals

- Sequences of numbers

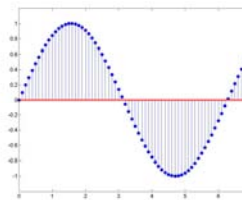
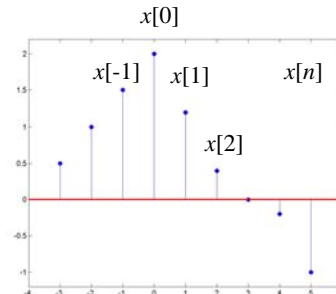
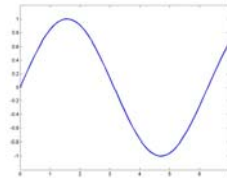
$$x = \{x[n]\}, \quad -\infty < n < \infty$$

where n is an integer

- Periodic sampling of an analog signal

$$x[n] = x_a(nT), \quad -\infty < n < \infty$$

where T is called the sampling period.



Sequence operations

- The product and sum of two sequences $x[n]$ and $y[n]$: sample-by-sample production and sum, respectively.
- Multiplication of a sequence $x[n]$ by a number α : multiplication of each sample value by α .
- Delay or shift of a sequence $x[n]$

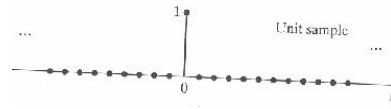
$$y[n] = x[n - n_0]$$

where n is an integer

Basic sequences

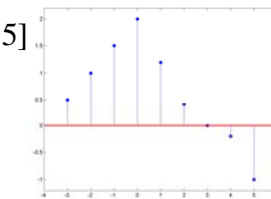
- **Unit sample sequence** (discrete-time impulse, impulse)

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



- Any sequence can be represented as a sum of scaled, delayed impulses

$$x[n] = a_{-3}\delta[n+3] + a_{-2}\delta[n+3] + \dots + a_5\delta[n-5]$$



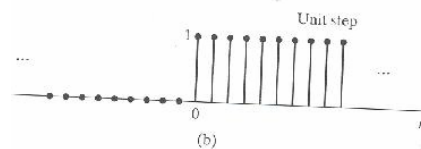
- More generally

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

Unit step sequence

- Defined as

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



- Related to the impulse by

$$u[n] = \delta[n] + \delta[n-1] + \delta[n-2] + \dots$$

or

$$u[n] = \sum_{k=-\infty}^{\infty} u[k]\delta[n-k] = \sum_{k=0}^{\infty} \delta[n-k]$$

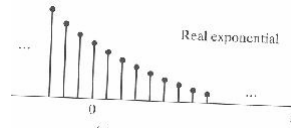
- Conversely,

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

Exponential sequences

- Extremely important in representing and analyzing LTI systems.
- Defined as

$$x[n] = A\alpha^n$$



- If A and α are real numbers, the sequence is real.
- If $0 < \alpha < 1$ and A is positive, the sequence values are positive and decrease with increasing n .
- If $-1 < \alpha < 0$, the sequence values alternate in sign, but again decrease in magnitude with increasing n .
- If $|\alpha| > 1$, the sequence values increase with increasing n .

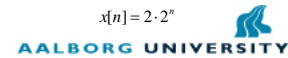
$$x[n] = 2 \cdot (0.5)^n$$

$$x[n] = 2 \cdot (-0.5)^n$$

$$x[n] = 2 \cdot 2^n$$

23

Digital Signal Processing, I, Zheng-Hua Tan, 2006



Combining basic sequences

- An exponential sequence that is zero for $n < 0$

$$x[n] = \begin{cases} A\alpha^n, & n \geq 0, \\ 0, & n < 0 \end{cases}$$

$$x[n] = A\alpha^n u[n]$$

24

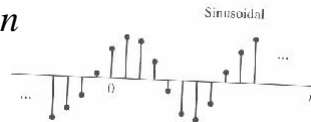
Digital Signal Processing, I, Zheng-Hua Tan, 2006



Sinusoidal sequences

$$x[n] = A \cos(\omega_0 n + \phi), \quad \text{for all } n$$

with A and ϕ real constants.



- The $A\alpha^n$ with complex α has real and imaginary parts that are exponentially weighted sinusoids.

If $\alpha = |\alpha| e^{j\omega_0}$ and $A = |A| e^{j\phi}$, then

$$\begin{aligned} x[n] = A\alpha^n &= |A| e^{j\phi} |\alpha|^n e^{j\omega_0 n} \\ &= |A| |\alpha|^n e^{j(\omega_0 n + \phi)} \\ &= |A| |\alpha|^n \cos(\omega_0 n + \phi) + j |A| |\alpha|^n \sin(\omega_0 n + \phi) \end{aligned}$$

Complex exponential sequence

When $|\alpha| = 1$,

$$x[n] = |A| e^{j(\omega_0 n + \phi)} = |A| \cos(\omega_0 n + \phi) + j |A| \sin(\omega_0 n + \phi)$$

- By analogy with the continuous-time case, the quantity ω_0 is called the **frequency** of the complex sinusoid or complex exponential and ϕ is called the **phase**.
- n is always an integer \rightarrow differences between discrete-time and continuous-time

An important difference – frequency range

- Consider a frequency $(\omega_0 + 2\pi)$

$$x[n] = Ae^{j(\omega_0 + 2\pi)n} = Ae^{j\omega_0 n} e^{j2\pi n} = Ae^{j\omega_0 n}$$

- More generally $(\omega_0 + 2\pi r)$, r being an integer,

$$x[n] = Ae^{j(\omega_0 + 2\pi r)n} = Ae^{j\omega_0 n} e^{j2\pi r n} = Ae^{j\omega_0 n}$$

- Same for sinusoidal sequences

$$x[n] = A \cos[(\omega_0 + 2\pi r)n + \phi] = A \cos(\omega_0 n + \phi)$$

- So, only consider frequencies in an interval of 2π such as

$$-\pi < \omega_0 \leq \pi \quad \text{or} \quad 0 \leq \omega_0 < 2\pi$$

27

Digital Signal Processing, I, Zheng-Hua Tan, 2006



Another important difference – periodicity

- In the continuous-time case, a sinusoidal signal and a complex exponential signal are both periodic.
- In the discrete-time case, a periodic sequence is defined as

$$x[n] = x[n + N], \quad \text{for all } n$$

where the period N is necessarily an integer.

- For sinusoid,

$$A \cos(\omega_0 n + \phi) = A \cos(\omega_0 n + \omega_0 N + \phi)$$

which requires that $\omega_0 N = 2\pi k$ or $N = 2\pi k / \omega_0$

where k is an integer.

28

Digital Signal Processing, I, Zheng-Hua Tan, 2006



Another important difference – periodicity

- Same for complex exponential sequence

$$e^{j\omega_0(n+N)} = e^{j\omega_0 n},$$

which is true only for $\omega_0 N = 2\pi k$

- So, complex exponential and sinusoidal sequences
 - are not necessarily periodic in n with period $(2\pi / \omega_0)$
 - and, depending on the value of ω_0 , may not be periodic at all.

- Consider

$$x_1[n] = \cos(\pi n / 4), \quad \text{with a period of } N = 8$$

$$x_2[n] = \cos(3\pi n / 8), \quad \text{with a period of } N = 16$$

Increasing frequency \rightarrow increasing period!

Another important difference – frequency

- For a continuous-time sinusoidal signal

$$x(t) = A \cos(\Omega_0 t + \phi),$$

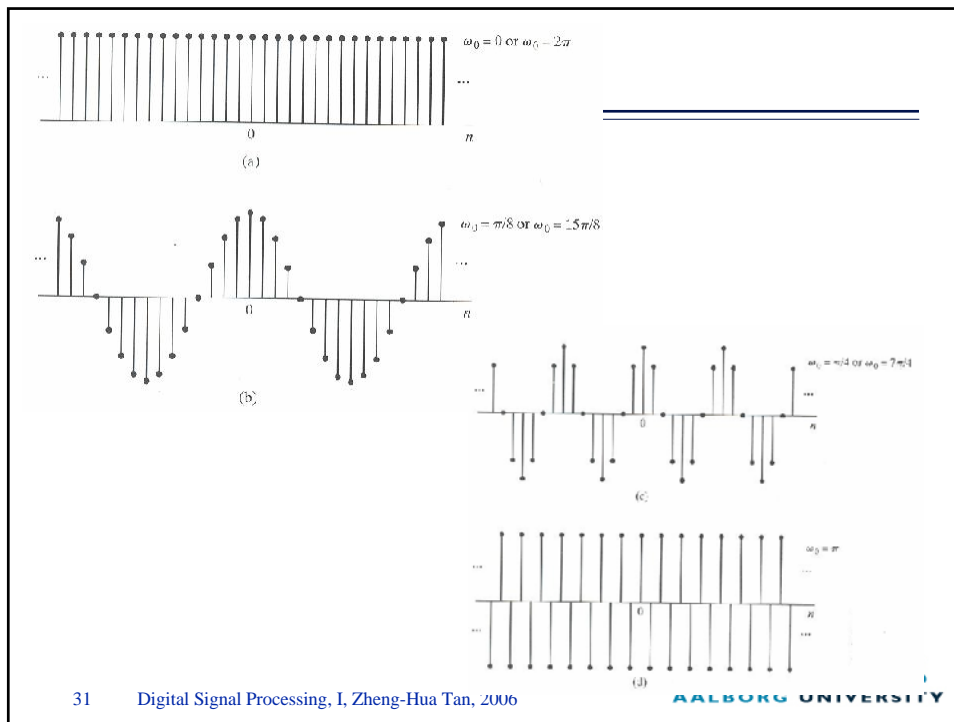
as Ω_0 increases, $x(t)$ oscillates more and more rapidly

- For the discrete-time sinusoidal signal

$$x[n] = A \cos(\omega_0 n + \phi),$$

as ω_0 increases from 0 towards π , $x[n]$ oscillates more and more rapidly

as ω_0 increases from π towards 2π , the oscillations become slower.



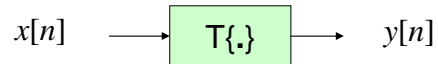
Part II: Discrete-time systems

- Introduction
- Discrete-time signals
- Discrete-time systems
- Linear time-invariant systems

Discrete-time systems

- A transformation or operator that maps input into output

$$y[n] = T\{x[n]\}$$



- Examples:

- The ideal delay system

$$y[n] = x[n - n_d], \quad -\infty < n < \infty$$

- A memoryless system

$$y[n] = (x[n])^2, \quad -\infty < n < \infty$$

Linear systems

- A system is linear if and only if

additivity property

$$T\{x_1[n] + x_2[n]\} = T\{x_1[n]\} + T\{x_2[n]\} = y_1[n] + y_2[n]$$

and

$$T\{ax[n]\} = aT\{x[n]\} = ay[n] \quad \text{scaling property}$$

where a is an arbitrary constant

- Combined into superposition

$$T\{ax_1[n] + bx_2[n]\} = aT\{x_1[n]\} + bT\{x_2[n]\} = ay_1[n] + by_2[n]$$

- Example 2.6, 2.7 pp. 19

Time-invariant systems

- For which a time shift or delay of the input sequence causes a corresponding shift in the output sequence.

$$x_1[n] = x[n - n_0] \Rightarrow y_1[n] = y[n - n_0]$$

- Example 2.8 pp. 20

Causality

- The output sequence value at the index $n=n_0$ depends only on the input sequence values for $n \leq n_0$.

- Example $y[n] = x[n - n_d]$, $-\infty < n < \infty$

- Causal for $n_d \geq 0$
- Noncausal for $n_d < 0$

Stability

- A system is stable in the BIBO sense if and only if every bounded input sequence produces a bounded output sequence.

- Example

stable $y[n] = (x[n])^2, \quad -\infty < n < \infty$

Part III: Linear time-invariant systems

- Course overview
- Discrete-time signals
- Discrete-time systems
- Linear time-invariant systems

Linear time-invariant systems

- Important due to convenient representations and significant applications
- A **linear** system is completely characterised by its impulse response

$$y[n] = T\{x[n]\} = T\left\{\sum_{k=-\infty}^{\infty} x[k]\delta[n-k]\right\}$$

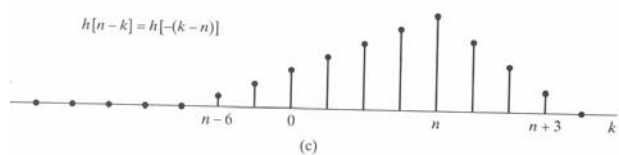
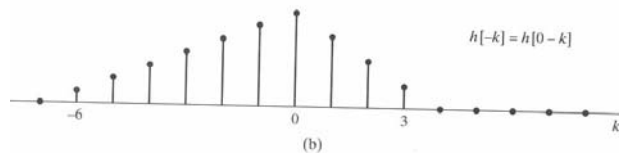
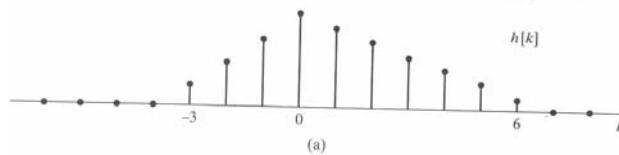
$$= \sum_{k=-\infty}^{\infty} x[k]T\{\delta[n-k]\} = \sum_{k=-\infty}^{\infty} x[k]h_k[n]$$

- **Time invariance** $h_k[n] = h[n-k]$

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

$$= x[n] * h[n] \quad \text{Convolution sum}$$

Forming the sequence $h[n-k]$



Computation of the convolution sum

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

- Obtain the sequence $h[n-k]$
 - Reflecting $h[k]$ about the origin to get $h[-k]$
 - Shifting the origin of the reflected sequence to $k=n$
- Multiply $x[k]$ and $h[n-k]$ for $-\infty < k < \infty$
- Sum the products to compute the output sample $y[n]$

Computing a discrete convolution

Example 2.13 pp.26

Impulse response

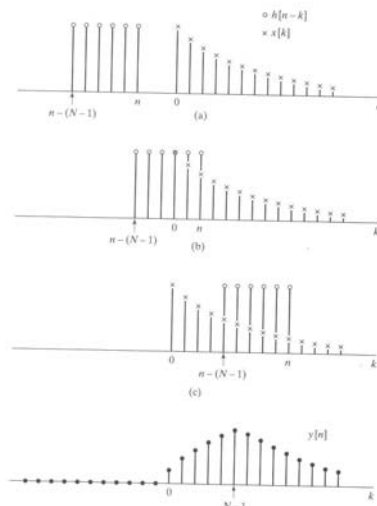
$$h[n] = u[n] - u[n - N]$$

$$= \begin{cases} 1, & 0 \leq n \leq N-1, \\ 0, & \text{otherwise.} \end{cases}$$

input

$$x[n] = a^n u[n]$$

$$y[n] = \begin{cases} 0, & n < 0, \\ \frac{1-a^{n+1}}{1-a}, & 0 \leq n \leq N-1, \\ a^{n-N+1} \left(\frac{1-a^N}{1-a} \right), & N-1 < n. \end{cases}$$



Properties of LTI systems

- Defined by discrete-time convolution

- Commutative

$$x[n]*h[n] = h[n]*x[n]$$

- Linear

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

- Cascade connection (Fig. 2.11 pp.29)

$$h[n] = h_1[n]*h_2[n]$$

- Parallel connection (Fig. 2.12 pp.30)

$$h[n] = h_1[n]+h_2[n]$$

Properties of LTI systems

- Defined by the impulse response

- Stable

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

- Causality

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

MATLAB

- An interactive, matrix-based system for numeric computation and visualization
- Kermit Sigmon, "**Matlab Primer**", Third Edition, Department of Mathematics, University of Florida.
- **Matlab Help** (>> *doc*)

Summary

- Course overview
- Discrete-time signals
- Discrete-time systems
- Linear time-invariant systems

- Matlab functions for the exercises in this lecture are available at
http://kom.aau.dk/~zt/cources/DSP_E/MM1/
Thanks Borge Lindberg for providing the functions.