KU LEUVEN

Digital Signal Processing 2

Les 6: Spectrale analyse

Prof. dr. ir. Toon van Waterschoot

Faculteit Industriële Ingenieurswetenschappen
ESAT – Departement Elektrotechniek
KU Leuven, Belgium

Digital Signal Processing 2: Vakinhoud

- Les 1: Eindige woordlengte
- Les 2: Lineaire predictie
- Les 3: Optimale filtering
- Les 4: Adaptieve filtering
- Les 5: Detectieproblemen
- Les 6: Spectrale signaalanalyse
- Les 7: Schattingsproblemen 1
- Les 8: Schattingsproblemen 2
- Les 9: Sigma-Deltamodulatie
- Les 10: Transformatiecodering

Les 6: Spectrale analyse

- **Deterministic signals**

  short-time DFT, windowing, frequency measurement, spectrogram…

  \+ MATLAB exercise

- **Random signals**

  periodogram, periodogram averaging, periodogram smoothing

  \+ MATLAB exercise

Les 6: Literatuur

- **Deterministic signals**

  B. Porat, *A Course in Digital Signal Processing*

  – Ch. 6, “Practical Spectral Analysis”
    - Section 6.1, “The Effect of Rectangular Windowing”
    - Section 6.2, “Windowing”
    - Section 6.3, “Common Windows”
    - Section 6.4, “Frequency Measurement”

- **Random signals**

  B. Porat, *A Course in Digital Signal Processing*

  – Ch. 2, “Review of Frequency-Domain Analysis”
    - Section 2.9, “Discrete-Time Random Signals”

  – Ch. 13, “Analysis and Modeling of Random Signals”
    - Section 13.1, “Spectral Analysis of Random Signals”
    - Section 13.2, “SA by Smoothed Periodogram”

Les 6: Spectrale analyse

- **Deterministic signals**

  short-time DFT, windowing, frequency measurement, spectrogram…

  + MATLAB exercise

- **Random signals**

  periodogram, periodogram averaging, periodogram smoothing

  + MATLAB exercise

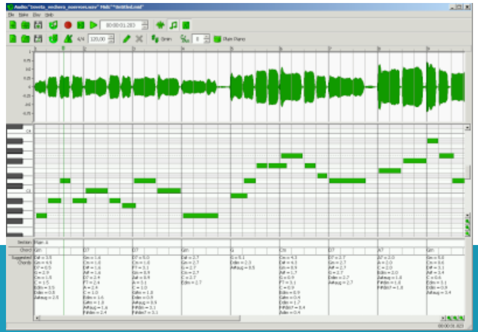
Deterministic signals: overview

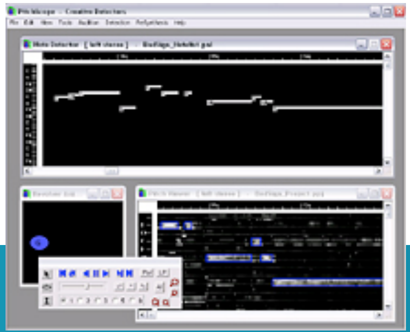
- Introduction: motivating example
- Rectangular windowing
- Windowing
- Frequency measurement
- Spectrogram

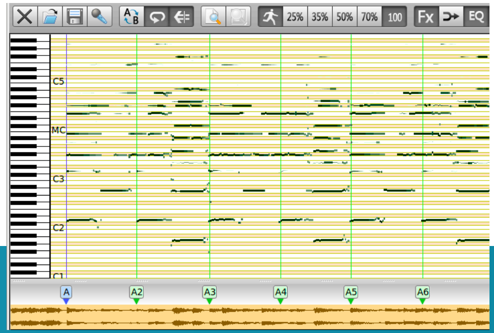
+ MATLAB exercise

Introduction: motivating example (1)

- Example from musical signal processing
  - spectral analysis of Brahms' 4^{th} symphony
  - motivation: automatic music transcription
  - some numbers:
    - 40 min of music
    - 44.1 kHz sampling rate
    - $O(10^8)$ samples
  - note:
    - [Porat, 1996] "such a task is still beyond our ability"
    - status in 2014: commercial software available, research ongoing







Introduction: motivating example (2)

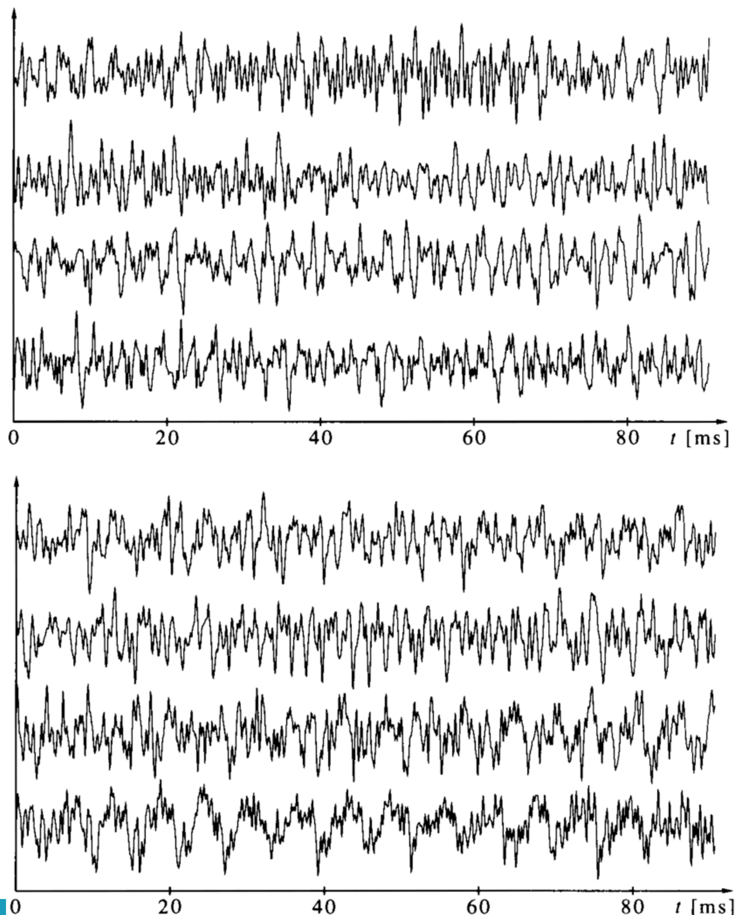
- Naïve approach: calculate N-point DFT with $N = O(10^8)$
  - extremely high frequency resolution ∼ 0.4 mHz
  - inefficient in terms of memory and processing resources
  - useless since result will be long-term spectrum average
- Meaningful approach: calculate sequence of short DFTs
  - naturally leads to time-frequency signal representation

    = essence of (short-time) spectral analysis
  - example:
    - DFT length = 4096 (∼ 92.9 ms signal segments)
    - frequency resolution = 11 Hz
    - 50 % overlap between successive signal segments
    - symphony = 52000 signal segments ⇒ 52000 length-4096 DFTs

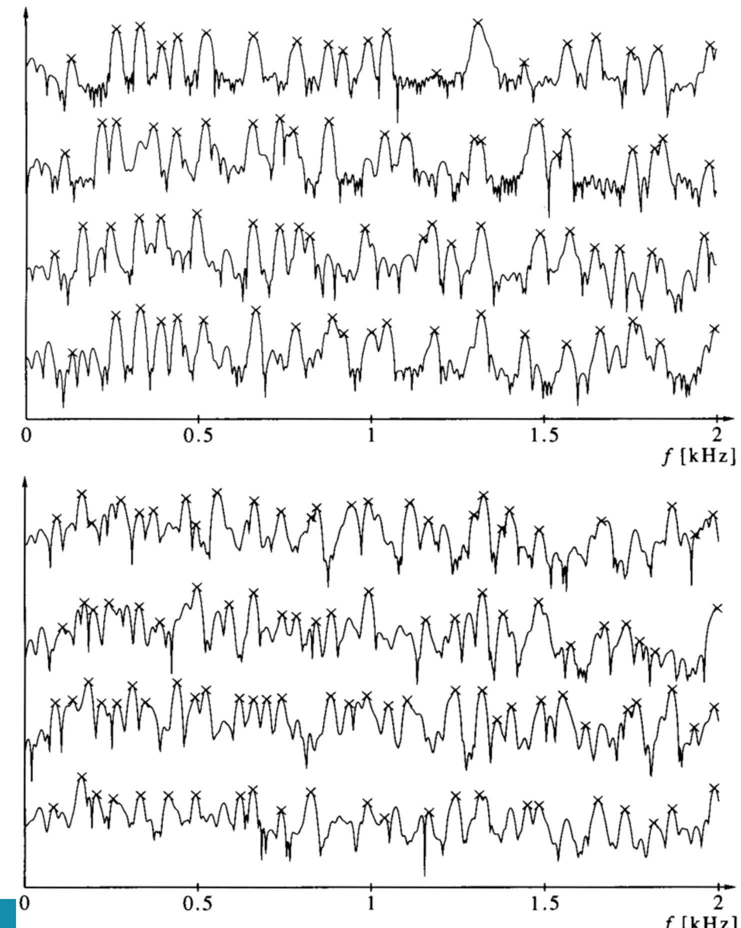
Introduction: motivating example (3)

- Example of eight 92.9 ms segments from Brahms’ symphony

time-domain waveforms



magnitude spectra



Deterministic signals: overview

- Introduction: motivating example
- Rectangular windowing
- Windowing
- Frequency measurement
- Spectrogram

+ MATLAB exercise

Rectangular windowing (1)

- **Signal segmentation = rectangular windowing**
  - consider picking short segment $x[n]$ from long signal $y[n]$

$$x[n] = \begin{cases} y[n], & 0 \le n \le N-1 \\ 0, & \text{otherwise} \end{cases}$$

  - this operation can also be written as a multiplication of $y[n]$ with a rectangular window

$$x[n] = y[n]w_r[n]$$

where the window is defined as

$$w_r[n] = \begin{cases} 1, & 0 \le n \le N-1 \\ 0, & \text{otherwise} \end{cases}$$

Rectangular windowing (2)

- **Key question:** how is Fourier transform of rectangular-windowed signal related to that of original signal?
- **Example**:
  - exponential signal and its Fourier transform:

$$y[n] = \begin{cases} a^n, & n \geq 0 \\ 0, & n < 0 \end{cases} \qquad Y^{\mathrm{f}}(\theta) = \frac{1}{1 - ae^{-j\theta}}$$

  - Fourier transform of rectangular-windowed signal

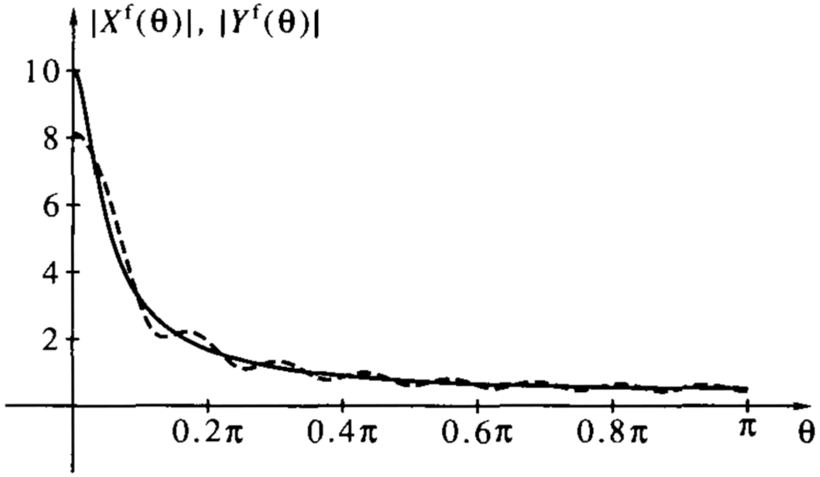
$$X^{\mathrm{f}}(\theta) = \sum_{n=0}^{N-1} a^n e^{-j\theta n} = \frac{1 - a^N e^{-j\theta N}}{1 - ae^{-j\theta}}$$

(note: apply sum formula for geometric series)

KU LEUVEN

Rectangular windowing (3)

- **Example**: $a = 0.9$, $N = 16$



$$X^{\mathrm{f}}(\theta) = \frac{1 - a^N e^{-j\theta N}}{1 - ae^{-j\theta}}$$

$$Y^{\mathrm{f}}(\theta) = \frac{1}{1 - ae^{-j\theta}}$$

  - magnitude spectrum of $x[n]$ approximates that of $y[n]$
  - magnitude spectrum of $y[n]$ exhibits smoother behavior

Rectangular windowing (4)

- **General result**:
  - time-domain multiplication = frequency-domain convolution

$$x[n] = y[n]w_r[n] \Rightarrow X^{\mathrm{f}}(\theta) = \frac{1}{2\pi}\left\{Y^{\mathrm{f}} * W_{\mathrm{r}}^{\mathrm{f}}\right\}(\theta)$$

  - Fourier transform of rectangular window:

$$W_{\mathrm{r}}^{\mathrm{f}}(\theta) = \sum_{n=0}^{N-1} e^{-j\theta n} = \frac{1-e^{-j\theta N}}{1-e^{-j\theta}}$$

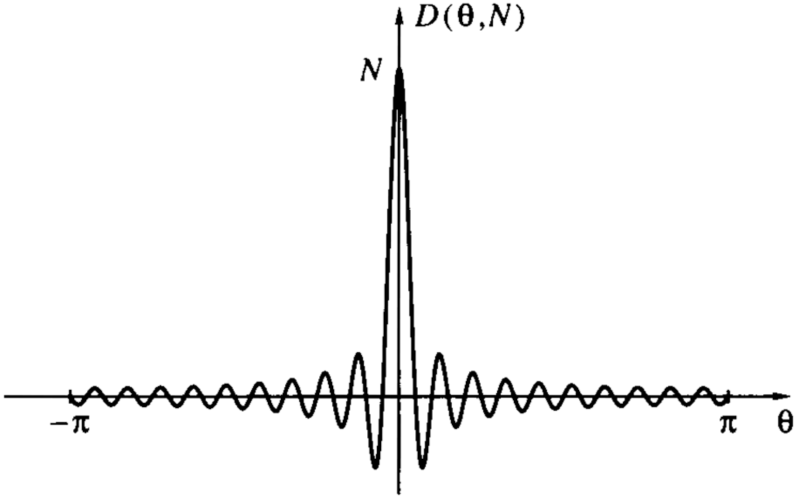
$$= \underbrace{\frac{\sin(0.5\theta N)}{\sin(0.5\theta)}}_{\text{magnitude}} \underbrace{e^{-j0.5\theta(N-1)}}_{\text{phase}}$$

Rectangular windowing (5)

- **General result**:
  - magnitude spectrum of rectangular window = Dirichlet kernel

$$D(\theta, N) = \frac{\sin(0.5\theta N)}{\sin(0.5\theta)}$$

  - example: $N = 40$

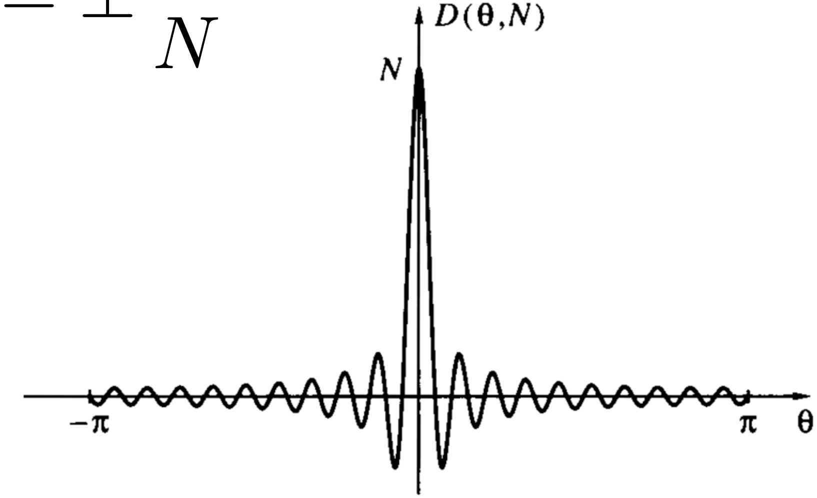


Rectangular windowing (6)

- **Dirichlet kernel:**
  - maximum value = *N*, occuring at $\theta = 0$
  - main lobe between zero crossings at $\theta = \pm\frac{2\pi}{N}$
  - side lobes between zero crossings at $\theta = \pm\frac{2m\pi}{N}, \quad m > 1$
  - highest side lobe occurs at $\theta = \pm\frac{3\pi}{N}$

    with amplitude $\approx \frac{2N}{3\pi}$

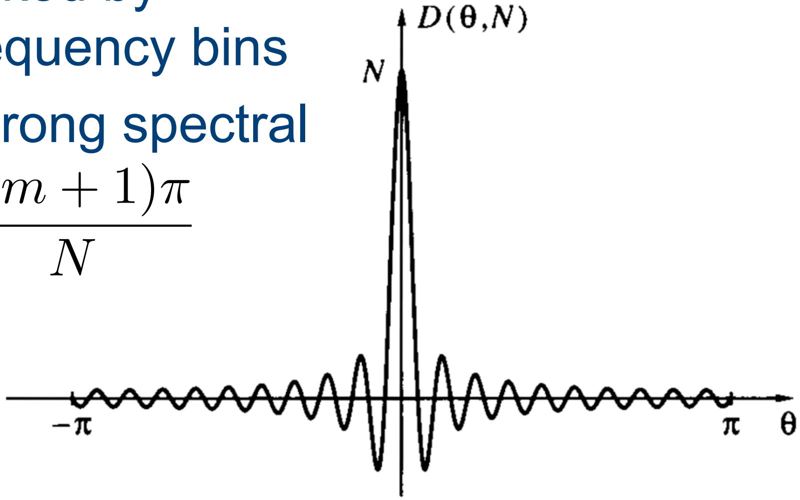
$$D(\theta, N) = \frac{\sin(0.5\theta N)}{\sin(0.5\theta)}$$



Rectangular windowing (7)

- **Distortions due to rectangular windowing:**
  - **smearing:** spectral lines become spectral lobes
    - bandwidth of main lobe is non-zero
    - loss of frequency resolution
    - neighboring spectral lines ($\Delta_\theta \leq \frac{4\pi}{N}$) will become unseparable
  - **side-lobe interference:** leakage of energy into other bins
    - weak spectral lines can be masked by strong spectral lines in other frequency bins
    - worst-case effect when weak/strong spectral lines are separated by $\Delta_\theta = \frac{(2m+1)\pi}{N}$

$$D(\theta, N) = \frac{\sin(0.5\theta N)}{\sin(0.5\theta)}$$



Deterministic signals: overview

- Introduction: motivating example
- Rectangular windowing
- Windowing
- Frequency measurement
- Spectrogram

+ MATLAB exercise

Windowing (1)

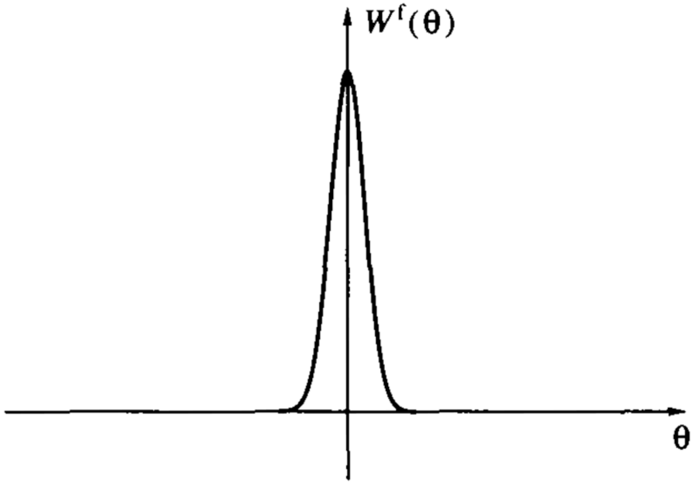
- **Rationale of window design**:
  - time-domain multiplication = frequency-domain convolution

$$x[n] = y[n]w[n] \Rightarrow X^{\mathrm{f}}(\theta) = \frac{1}{2\pi}\left\{Y^{\mathrm{f}} * W^{\mathrm{f}}\right\}(\theta)$$

  - constraints on window sequence *w*[*n*]
    - finite duration
    - length *N* must agree with desired segmentation length
    - non-negative
  - desirable properties of window sequence *w*[*n*]
    - main lobe as narrow as possible
    - side lobes as low as possible

Windowing (2)

- **Rationale of window design**:
  - Fourier transform of window sequence = kernel function
  - desired kernel shape = frequency-domain delta function (why don't we choose $W^{\mathrm{f}}(\theta) = 2\pi\delta(\theta)$?)

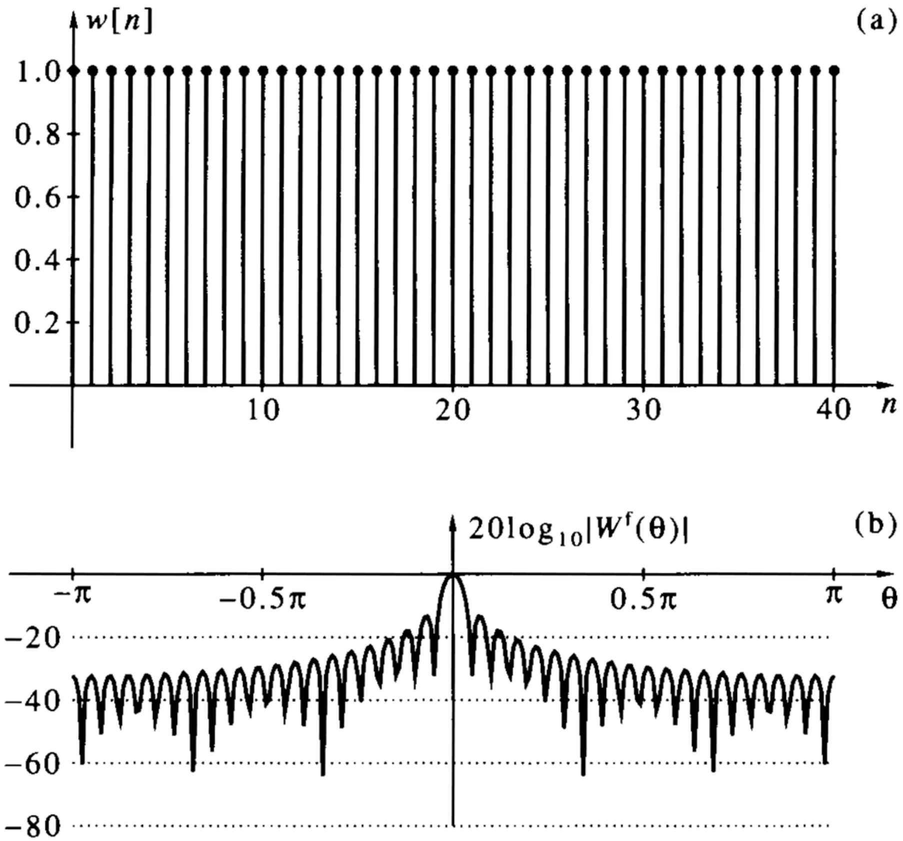


  - window design = **trade-off** between main lobe width and side lobe levels

Windowing (3)

1. **Rectangular window (revisited)**:

- main lobe width = $4\pi/N$
  = narrowest of all length-N windows!
- side lobe level = −13.5 dB
- example: $N = 41$



Windowing (4)

2. **Bartlett window (= triangular window)**:

- rationale: squaring kernel function halves side lobe level
- implementation:
  - convolve rectangular time-domain window sequence with itself
  - in case *N* is odd, start from length-(*N*+1)/2 rectangular window:

$$w_{\mathrm{t}}[n] = \frac{2}{N+1}\left\{w_{\mathrm{r}} * w_{\mathrm{r}}\right\}[n] = 1 - \frac{|2n - N + 1|}{N+1}, \quad 0 \leq n \leq N-1$$

  - corresponding kernel function:

$$W_{\mathrm{t}}^{\mathrm{f}}(\theta) = \frac{2}{N+1} D^2\left(\theta, 0.5(N+1)\right) e^{-j0.5\theta(N-1)} = \frac{2\sin^2\left\{0.25\theta(N+1)\right\}}{(N+1)\sin^2(0.5\theta)} e^{-j0.5\theta(N-1)}$$

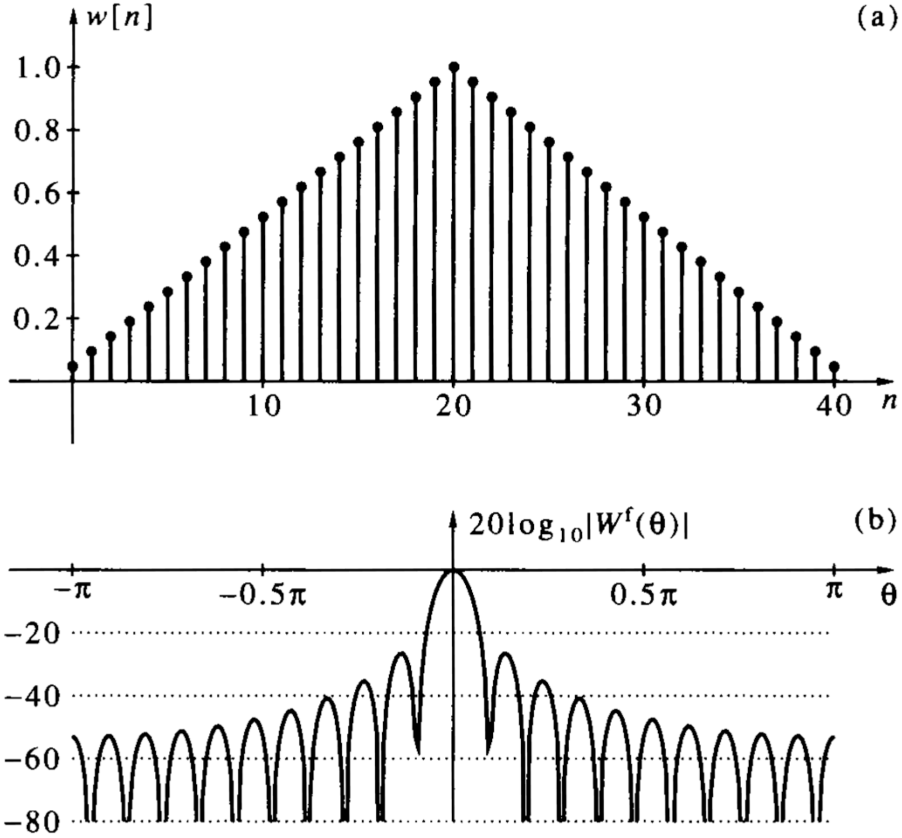
  - (note: similar expressions exist for case of even *N*)

KU LEUVEN

Windowing (5)

2. Bartlett window (= triangular window):

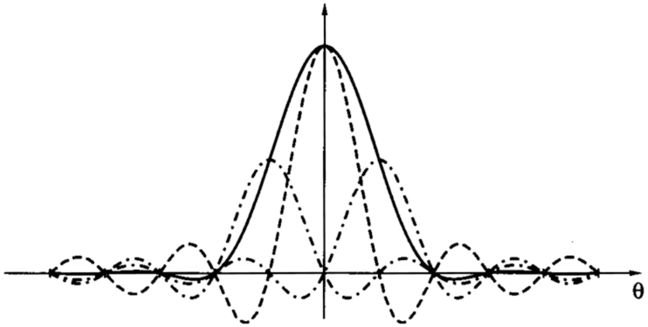
- main lobe width = $8\pi/(N+1)$
- side lobe level = −27 dB
- example: $N = 41$



Windowing (6)

3. Hann window (= cosine window):

- rationale: superposition of 3 frequency-shifted Dirichlet kernels to reduce side lobe level
- implementation:
  - exploit modulation theorem of Fourier transform:



$$W^{\mathrm{f}}_{\mathrm{hn}}(\theta) = 0.5W^{\mathrm{f}}_{\mathrm{r}}(\theta) - 0.25W^{\mathrm{f}}_{\mathrm{r}}\left(\theta - \frac{2\pi}{N-1}\right) - 0.25W^{\mathrm{f}}_{\mathrm{r}}\left(\theta + \frac{2\pi}{N-1}\right)$$

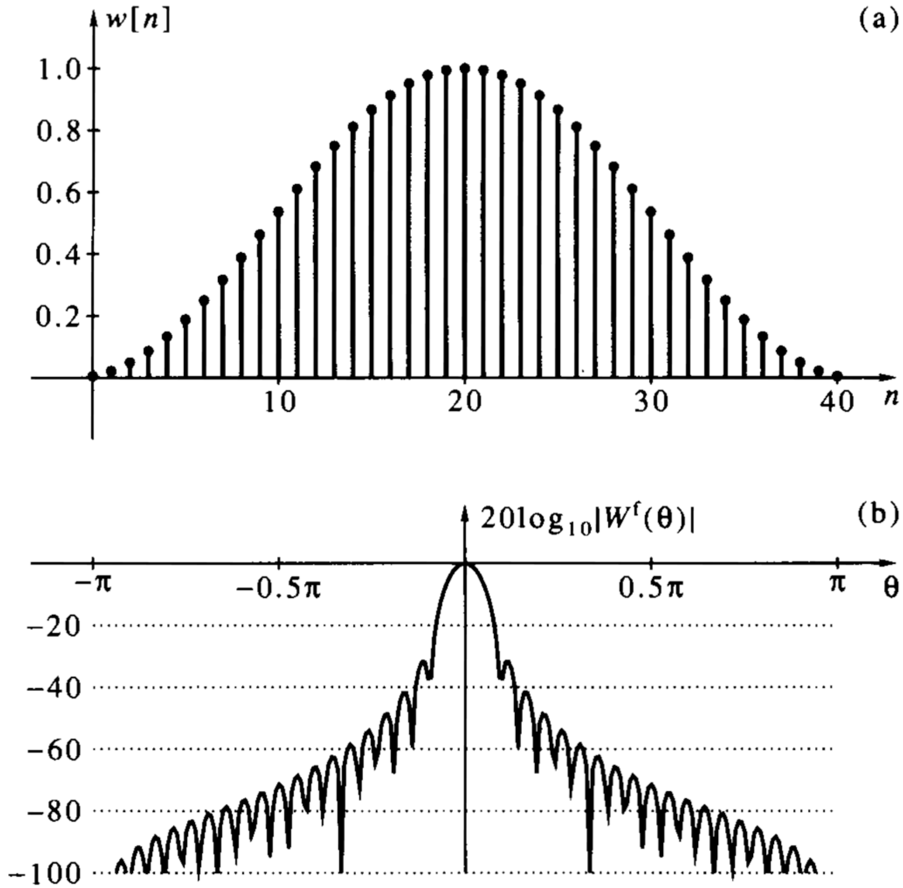
$$\begin{aligned} w_{\mathrm{hn}}[n] &= 0.5 - 0.25\exp\left(\frac{j2\pi n}{N-1}\right) - 0.25\exp\left(-\frac{j2\pi n}{N-1}\right) \\ &= 0.5\left[1 - \cos\left(\frac{2\pi n}{N-1}\right)\right], \quad 0 \leq n \leq N-1 \end{aligned}$$

KU LEUVEN

Windowing (7)

3. Hann window (= cosine window):

- main lobe width = $8\pi/N$
- side lobe level = −32 dB
- example: $N = 41$
- note: two end points = 0



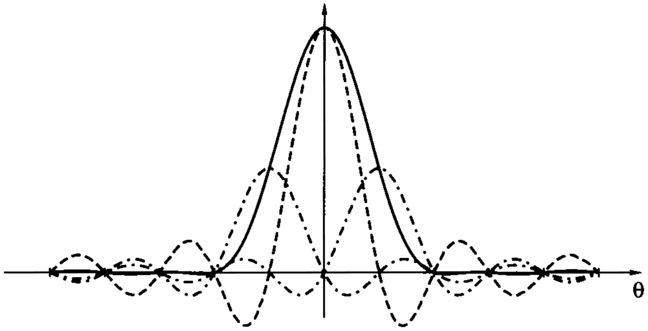
- **Modified Hann window:**
  - start from length-(N+2) rectangular window
  - delete zero end points

$$w_{\mathrm{hn}}[n] = 0.5\left\{1 - \cos\left[\frac{2\pi(n+1)}{N+1}\right]\right\}, \quad 0 \le n \le N-1$$

Windowing (8)

4. Hamming window (= raised cosine window):

- rationale: superposition of 3 frequency-shifted Dirichlet kernels to reduce side lobe level
- implementation:
  - similar to Hann window
  - weights chosen to minimize side lobe level



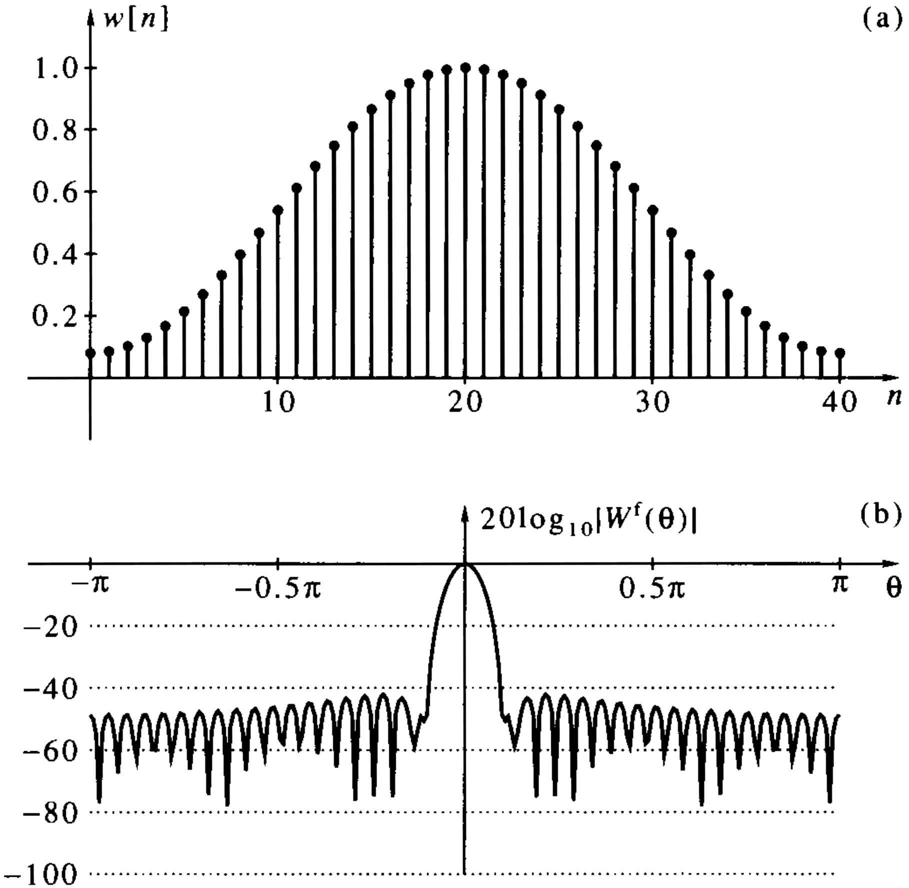
$$W^{\mathrm{f}}_{\mathrm{hm}}(\theta) = 0.54 W^{\mathrm{f}}_{\mathrm{r}}(\theta) - 0.23 W^{\mathrm{f}}_{\mathrm{r}}\left(\theta - \frac{2\pi}{N-1}\right) - 0.23 W^{\mathrm{f}}_{\mathrm{r}}\left(\theta + \frac{2\pi}{N-1}\right)$$

$$w_{\mathrm{hm}}[n] = 0.54 - 0.46 \cos\left(\frac{2\pi n}{N-1}\right), \ 0 \leq n \leq N-1$$

Windowing (9)

4. Hamming window (= raised cosine window):

- main lobe width = $8\pi/N$
- side lobe level = −43 dB
- example: N = 41
- note: two end points ≠ 0



Windowing (10)

5. Blackman window:

- rationale: superposition of 5 frequency-shifted Dirichlet kernels to reduce side lobe level
- implementation:
  - similar to Hamming window
  - 5 instead of 3 Dirichlet kernels

$$W_{\mathrm{b}}^{\mathrm{f}}(\theta) = 0.42 W_{\mathrm{r}}^{\mathrm{f}}(\theta) - 0.25 W_{\mathrm{r}}^{\mathrm{f}}\left(\theta + \frac{2\pi}{N-1}\right) - 0.25 W_{\mathrm{r}}^{\mathrm{f}}\left(\theta - \frac{2\pi}{N-1}\right)$$
$$+0.04 W_{\mathrm{r}}^{\mathrm{f}}\left(\theta + \frac{4\pi}{N-1}\right) + 0.04 W_{\mathrm{r}}^{\mathrm{f}}\left(\theta - \frac{4\pi}{N-1}\right)$$

↓

$$w_{\mathrm{b}}[n] = 0.42 - 0.5\cos\left(\frac{2\pi n}{N-1}\right) + 0.08\cos\left(\frac{4\pi n}{N-1}\right), \quad 0 \le n \le N-1$$

KU LEUVEN

Windowing (11)

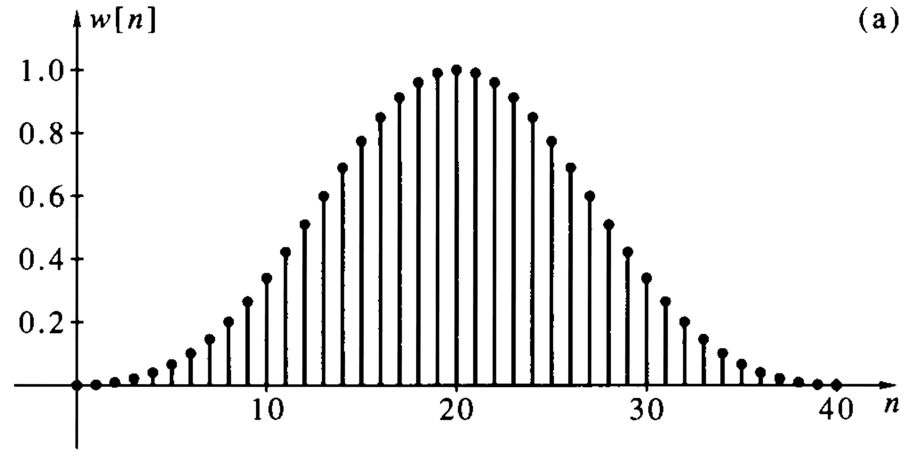
5. Blackman window:

– main lobe width = $12\pi/N$

– side lobe level = −57 dB

– example: N = 41

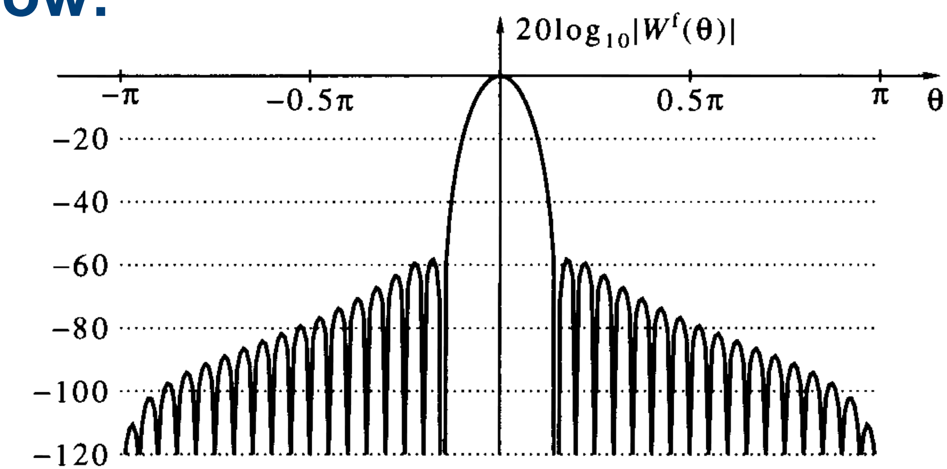
– note: two end points = 0



- **Modified Blackman window:**

  – start from length-(N+2) rectangular window

  – delete zero end points



Windowing (12)

6. Kaiser window:

- rationale: calculate family of windows as solution to constrained optimization problem

$$\min_{w_{\mathrm{k}}[n]} \quad \text{main lobe width}$$

$$\text{s.t.} \quad \frac{\text{side lobe energy}}{\text{total energy}} \leq \beta$$

- solution (with I_0 = modified Bessel function of order zero):

$$w_{\mathrm{k}}[n] = \frac{I_0\left[\alpha\sqrt{1-\left(\frac{|2n-N+1|}{N-1}\right)^2}\right]}{I_0[\alpha]}, \; 0 \leq n \leq N-1$$

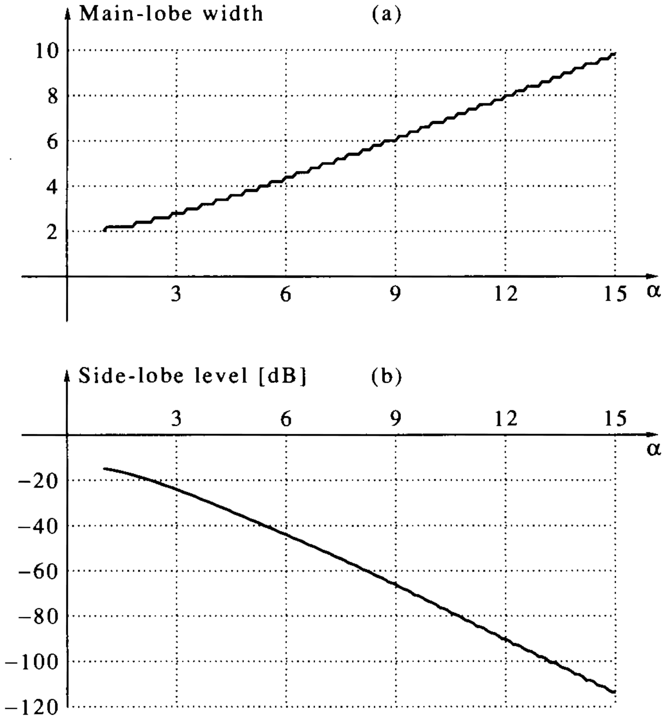
- parameter α determines trade-off between main lobe width and side lobe energy

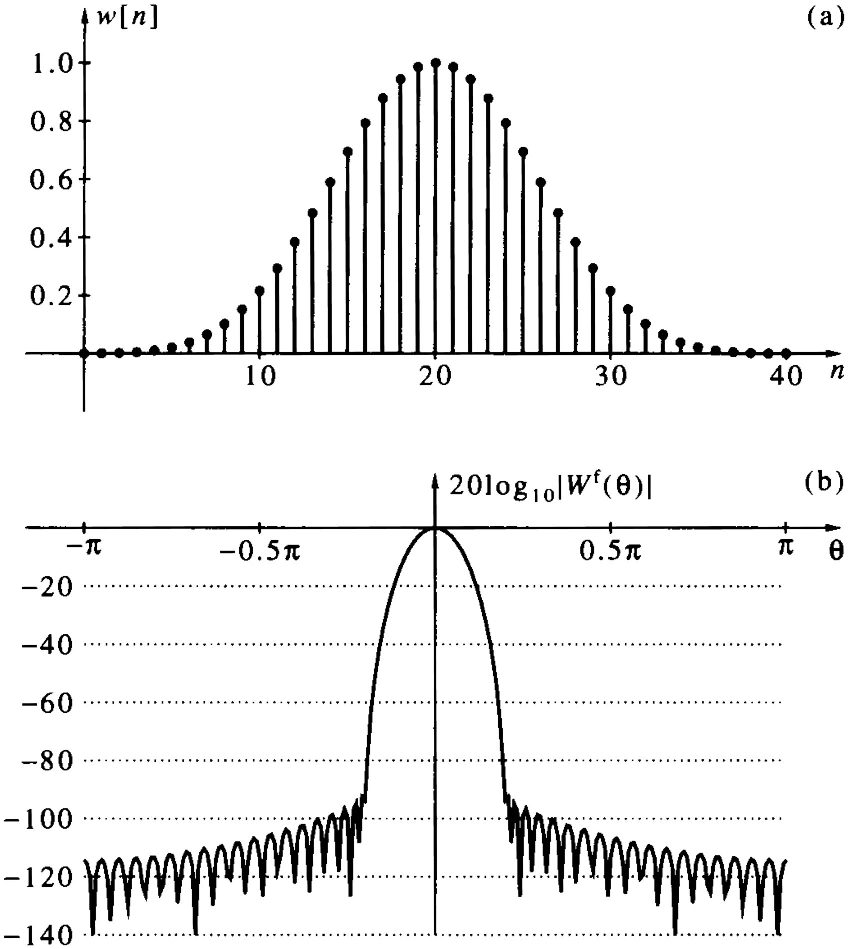
KU LEUVEN

Windowing (13)

6. Kaiser window ($\alpha = 12$):

- main lobe width = $16\pi/N$
- side lobe level = −90 dB (!)
- example: $N = 41$





Deterministic signals: overview

- Introduction: motivating example
- Rectangular windowing
- Windowing
- Frequency measurement
- Spectrogram

+ MATLAB exercise

Frequency measurement (1)

- **Frequency measurement of single complex sinusoid:**
  - in theory, frequency of single complex sinusoid

$$y[n] = Ae^{j(\theta_0 n + \phi_0)}, \ 0 \leq n \leq N-1$$

can be uniquely determined from its Fourier transform

$$Y^{\mathrm{f}}(\theta) = Ae^{-j[0.5(\theta-\theta_0)(N-1)-\phi_0]} D(\theta-\theta_0, N)$$

by finding maximum of magnitude spectrum

$$\theta_0 = \arg\min_{\theta} |Y^{\mathrm{f}}(\theta)|$$

Frequency measurement (2)

- **Frequency measurement of single complex sinusoid:**
  - in practice, however, DFT is calculated on frequency grid generally not including θ_0
  - simple frequency measurement:
    - find frequency bin with maximum magnitude

$$k_0 = \arg\min_k |Y^{\mathrm{d}}[k]| \Rightarrow \theta[k_0] = \frac{2\pi k_0}{N}$$

  - improved frequency measurement:
    - zero pad sequence $y[n]$ and increase DFT length

Frequency measurement (3)

- **Frequency measurement of two complex sinusoids:**
  - sum of two complex sinusoids and its frequency spectrum

$$y[n] = A_1 e^{j(\theta_1 n + \phi_1)} + A_2 e^{j(\theta_2 n + \phi_2)},\ 0 \leq n \leq N-1$$

$$\begin{aligned} Y^{\mathrm{f}}(\theta) &= A_1 e^{-j[0.5(\theta-\theta_1)(N-1)-\phi_1]} D(\theta-\theta_1, N) \\ &\quad + A_2 e^{-j[0.5(\theta-\theta_2)(N-1)-\phi_2]} D(\theta-\theta_2, N) \end{aligned}$$

  - evaluation of Fourier transform at one sinusoid frequency θ_1

$$Y^{\mathrm{f}}(\theta_1) = N A_1 e^{j\phi_1} + A_2 e^{-j[0.5(\theta_1-\theta_2)(N-1)-\phi_2]} D(\theta_1-\theta_2, N)$$

  - frequency measurement only feasible when

$$|A_2 D(\theta_1-\theta_2, N)| \ll N A_1 \Leftrightarrow \left\{ \begin{array}{rcl} |\theta_2-\theta_1| & \geq & 2\pi/N \\ O(A_2) & = & O(A_1) \end{array} \right.$$

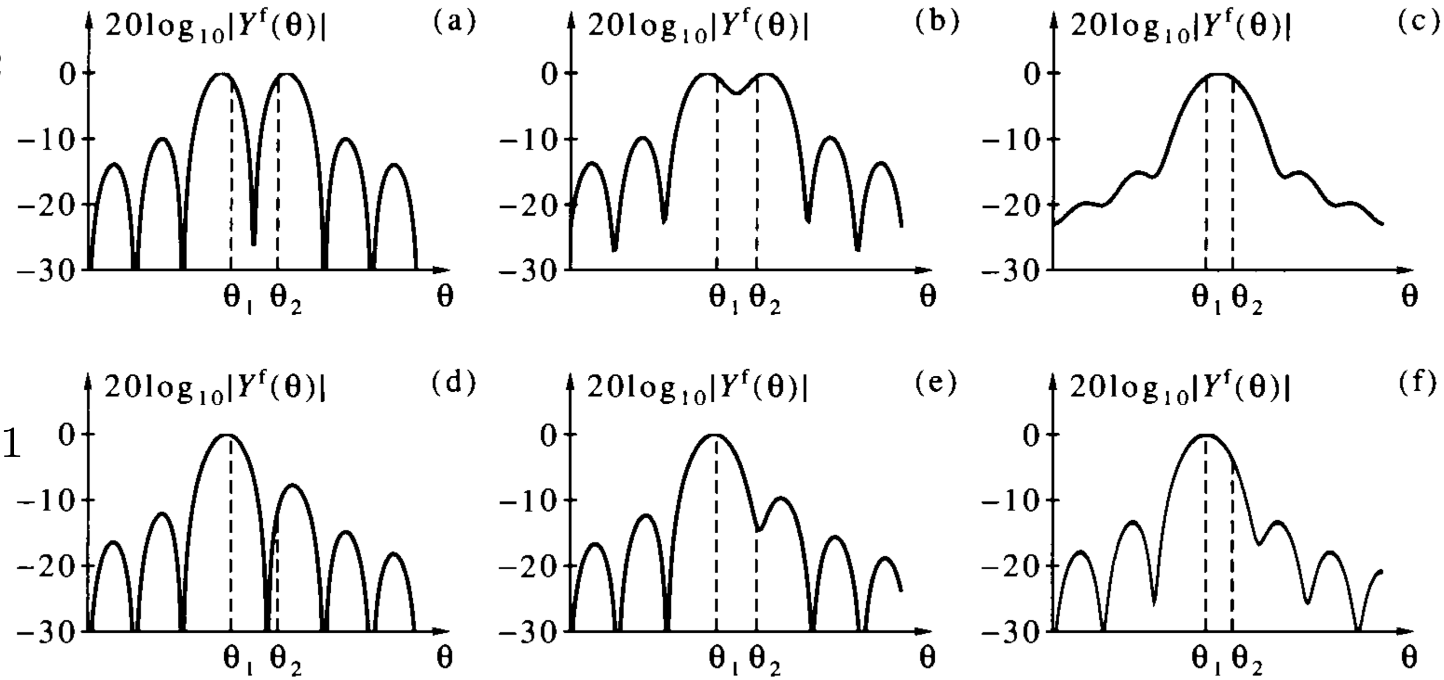
KU LEUVEN

Frequency measurement (4)

- **Frequency measurement of two complex sinusoids:**
  - example:

$A_1 = A_2$

$A_2 = 0.25 A_1$



$|\theta_2 - \theta_1| = 2\pi/N$ $1.5\pi/N$ π/N

KU LEUVEN

Frequency measurement (5)

- **Frequency measurement of two complex sinusoids:**
  - windowed sum of two complex sinusoids $x[n] = y[n]w[n]$ has Fourier transform

$$X^{\mathrm{f}}(\theta) = A_1 e^{j\phi_1} W^{\mathrm{f}}(\theta - \theta_1, N) + A_2 e^{j\phi_2} W^{\mathrm{f}}(\theta - \theta_2, N)$$

  - evaluation of Fourier transform at one sinusoid frequency θ_1

$$X^{\mathrm{f}}(\theta_1) = A_1 e^{j\phi_1} W^{\mathrm{f}}(0, N) + A_2 e^{j\phi_2} W^{\mathrm{f}}(\theta_1 - \theta_2, N)$$

  - frequency measurement feasible with window for which

$$|A_2 W^{\mathrm{f}}(\theta_1 - \theta_2, N)| \ll A_1 \sum_{n=0}^{N-1} w[n]$$

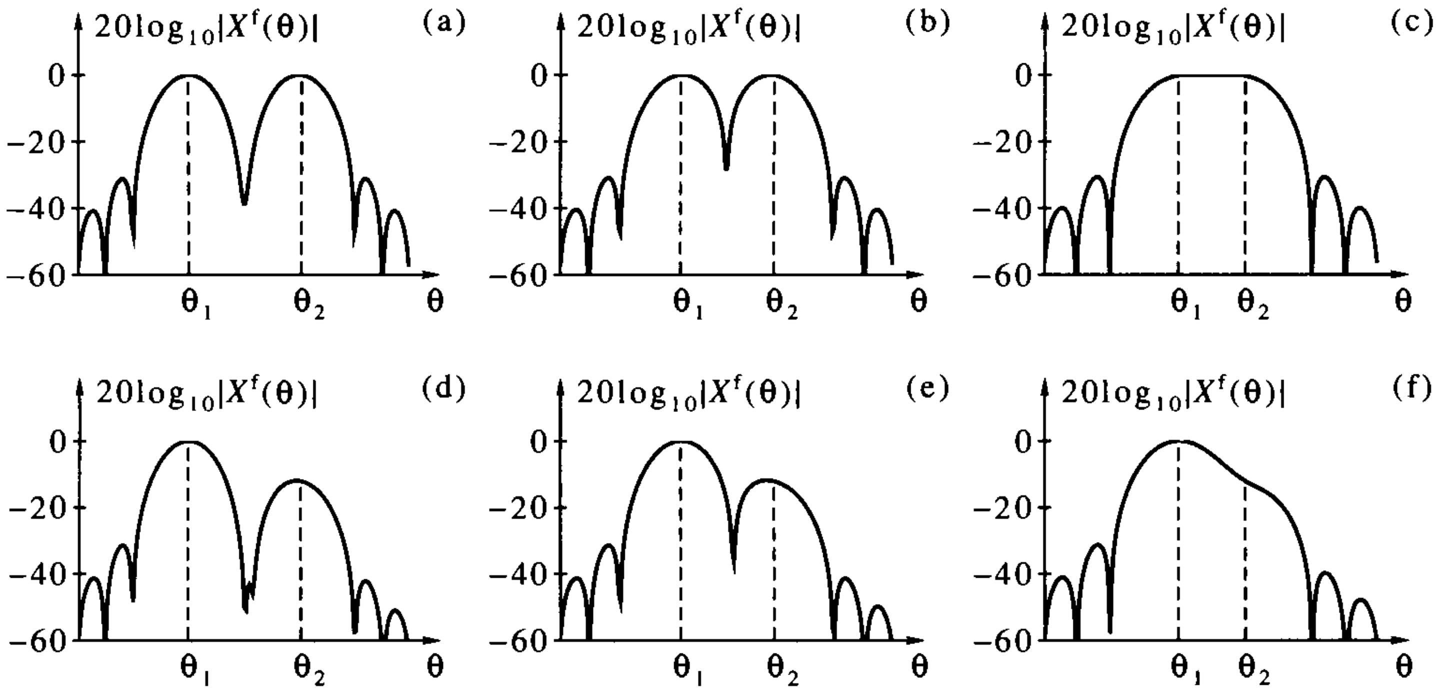
    - $|\theta_2 - \theta_1|$ greater than kernel main lobe width
    - $20 \log_{10}(A_1 / A_2)$ larger than side-lobe level

Frequency measurement (6)

- **Frequency measurement of two complex sinusoids:**
  - example: Hann window

$A_1 = A_2$

$A_2 = 0.25 A_1$



$|\theta_2 - \theta_1| = 8\pi/N$ $6\pi/N$ $4\pi/N$

Frequency measurement (7)

- **Frequency measurement of *M* real sinusoids:**
  - real sinusoids can be decomposed as sum of complex sinusoids, hence previous results still hold
  - frequency measurement is feasible without windowing if
    - sinusoid frequencies are separated by at least $2\pi/N$
    - π/N < sinusoid frequencies < $\pi(1 - 1/N)$
    - $O(A_1) = \ldots = O(A_M)$
  - frequency measurement is feasible with windowing if
    - sinusoid frequencies are separated by at least ½ main lobe width
    - ½ main lobe width < sinusoid frequencies < π – ½ main lobe width
    - sinusoid amplitude differences > side lobe level

Frequency measurement (8)

- **Practice of frequency measurement**
  1. multiply sampled sequence by window
  2. compute DFT (using FFT algorithm)
  3. search for local maxima in magnitude spectrum

Deterministic signals: overview

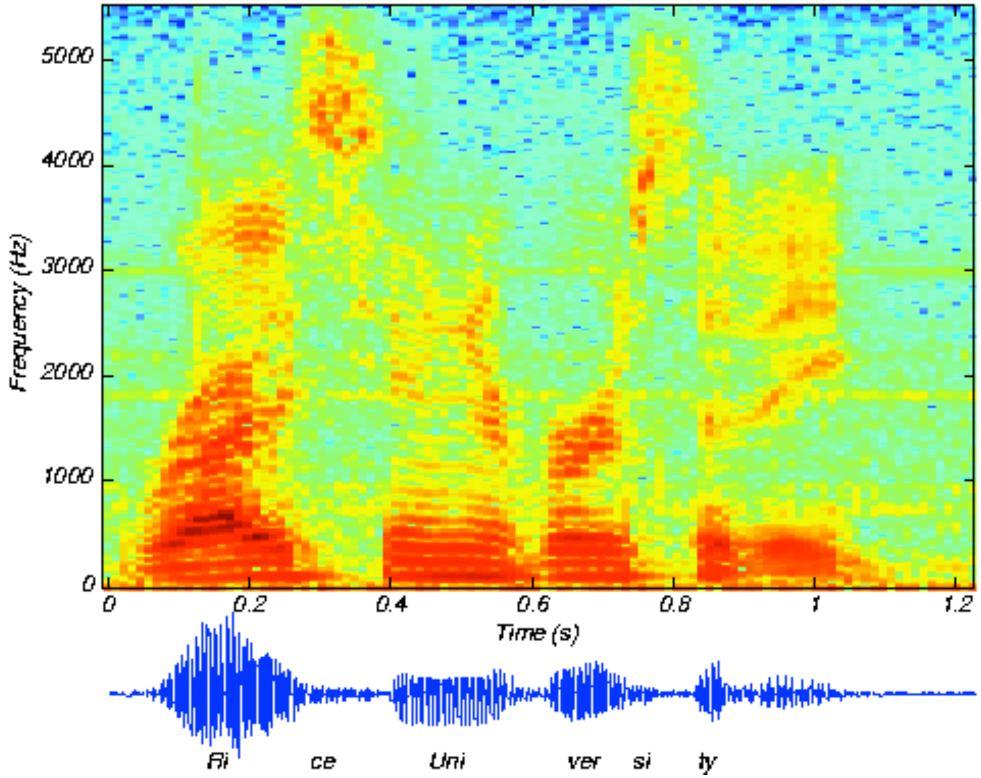
- Introduction: motivating example
- Rectangular windowing
- Windowing
- Frequency measurement
- Spectrogram

+ MATLAB exercise

KU LEUVEN

Spectrogram

- Spectrogram = 2-D color plot of DFT log-magnitude for number of overlapping windowed signal segments
  - example: speech signal [Rice University]



Deterministic signals: overview

- Introduction: motivating example
- Rectangular windowing
- Windowing
- Frequency measurement
- Spectrogram

+ MATLAB exercise

MATLAB exercise (1)

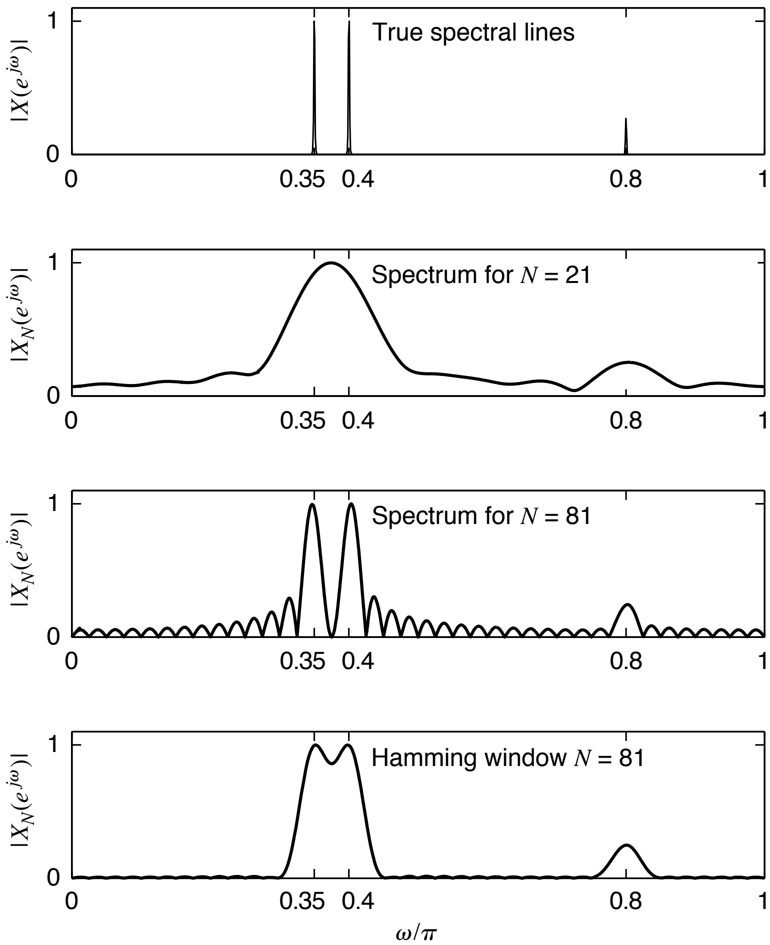
Consider signal $y[n] = \cos 0.35\pi n + \cos 0.4\pi n + 0.25 \cos 0.8\pi n$

1. Draw magnitude spectrum by hand
2. Generate length-*N* segment of this signal in Matlab
3. Calculate and plot magnitude spectrum for *N* = 21
4. Calculate and plot magnitude spectrum for *N* = 21 with zero padding up till 2048 samples
5. Calculate and plot magnitude spectrum for *N* = 81 with zero padding up till 2048 samples
6. Calculate and plot magnitude spectrum for N = 81 with zero padding up till 2048 samples and Hamming windowing
7. Compare and explain results

MATLAB exercise (2)

Consider signal $y[n] = \cos 0.35\pi n + \cos 0.4\pi n + 0.25 \cos 0.8\pi n$

- Solution:



Les 6: Spectrale analyse

- **Deterministic signals**

  short-time DFT, windowing, frequency measurement, spectrogram…

  + MATLAB exercise

- **Random signals**

  periodogram, periodogram averaging, periodogram smoothing

  + MATLAB exercise

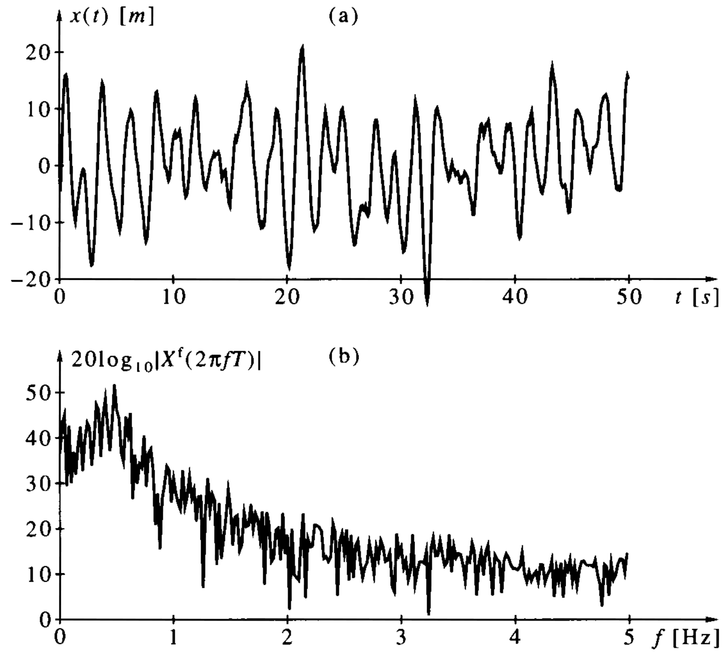
Random signals: overview

- Introduction: motivating example
- Averaged periodogram
- Smoothed periodogram

+ MATLAB exercise

Introduction: motivating example (1)

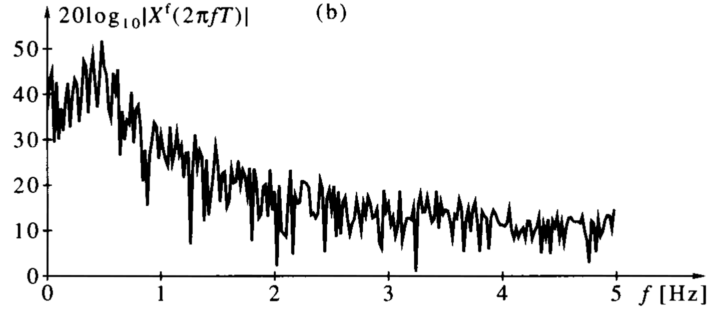
- Example from oceanography
  - measurement of time variation in height of ocean waves



  - signal shows oscillatory behavior but not sinusoidal behavior
  - repeating measurement yields similar but not same signal

Introduction: motivating example (2)

- Example from oceanography
  - how do we perform spectral analysis of random signals?
  - short-time spectral analysis (as for deterministic signals)? No!
    - random signals have random Fourier transform
    - DFT magnitude spectrum will look very noisy
    - spectrum details vary from experiment to experiment
    - zero padding does not solve this problem



  - some sort of averaging or smoothing is required

Random signals: overview

- Introduction: motivating example
- Averaged periodogram
- Smoothed periodogram

+ MATLAB exercise

KU LEUVEN

Averaged periodogram (1)

- Averaged periodogram
  - periodogram = square magnitude of DFT
  - divide length-*NL* signal *x*[*n*] into *L* length-*N* segments
  - averaged periodogram (AP) over L segments:

$$\hat{K}_x^{\mathrm{f}}(\theta) = \frac{1}{L}\sum_{l=0}^{L-1}\left\{\frac{1}{N}\left|\sum_{n=0}^{N-1} x[n+lN]e^{-j\theta n}\right|^2\right\}$$

  - key property: AP converges to power spectral density (PSD)

$$\lim_{\substack{N\to\infty \\ L\to\infty}} \hat{K}_x^{\mathrm{f}}(\theta) = K_x^{\mathrm{f}}(\theta) = E\left\{\frac{1}{N}\left|\sum_{n=0}^{N-1} x[n+lN]e^{-j\theta n}\right|^2\right\}$$

  - note: why average periodogram instead of DFT?

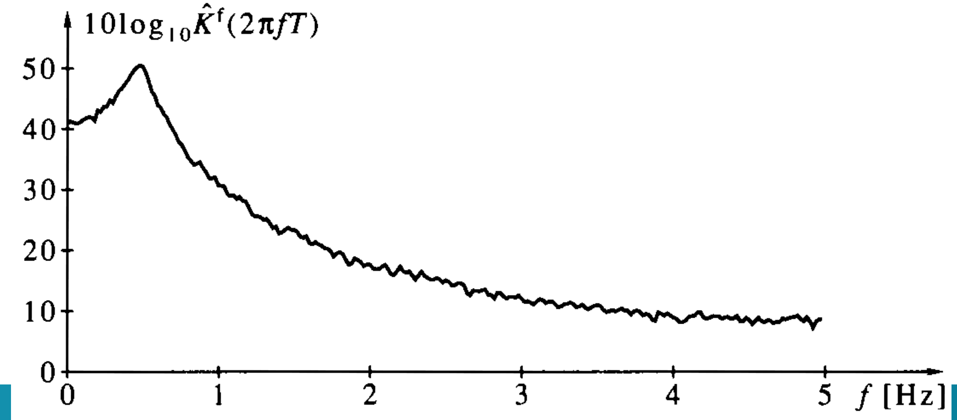
KU LEUVEN

Averaged periodogram (2)

- Windowed averaged periodogram
  - since *N,L* ≠ ∞ averaged periodogram is still (slightly) random
    - increasing *N* makes AP more detailed and more random
    - increasing *L* makes AP smoother with less randomness
  - additional smoothing can be obtained by windowing

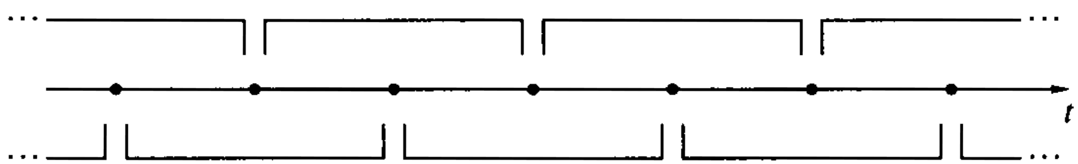
$$\hat{K}_x^{\mathrm{f}}(\theta) = \frac{1}{L}\sum_{l=0}^{L-1}\left\{\frac{1}{N}\left|\sum_{n=0}^{N-1} w[n]x[n+lN]e^{-j\theta n}\right|^2\right\}$$

  - example: ocean wave AP
    - $L = 100$
    - $N = 500$
    - $w[n]$ = Hann window



Averaged periodogram (3)

- Welch periodogram
  - windowing reduces importance of end-of-segment samples
  - compensated by using (50%) overlap in signal segmentation



  - Welch periodogram = AP with windowing and 50% overlap

$$\hat{K}_x^{\mathrm{f}}(\theta) = \frac{1}{L}\sum_{l=0}^{L-1}\left\{\frac{1}{N}\left|\sum_{n=0}^{N-1} w[n]x[n+0.5lN]e^{-j\theta n}\right|^2\right\}$$

= standard tool for spectral analysis of stationary random signals

Random signals: overview

- Introduction: motivating example
- Averaged periodogram
- Smoothed periodogram

+ MATLAB exercise

Smoothed periodogram (1)

- Smoothed periodogram
  - consider case when data sequence is too short for averaging
    - because limited amount of data measurement is available
    - because signal has non-stationary behavior
  - smoothed periodogram (without segmentation or averaging)
    - frequency-domain convolution of periodogram and window kernel

$$\hat{K}_x^{\mathrm{f}}(\theta) = \frac{1}{2\pi}\int_{-\pi}^{\pi} \frac{1}{N}|X^{\mathrm{f}}(\theta-\lambda)|^2 W^{\mathrm{f}}(\lambda)d\lambda$$

    - time-domain implementation by length-(2M+1) windowing

$$\hat{K}_x^{\mathrm{f}}(\theta) = \sum_{m=-M}^{M} \hat{\kappa}_x[m]w[m]e^{-j\theta m}$$

KU LEUVEN

Smoothed periodogram (2)

- Smoothed periodogram
  - $\hat{\kappa}_x[m]$ ~ inverse Fourier transform of periodogram $|X^{\mathbf{f}}(\theta)|^2$
  - $\hat{\kappa}_x[m]$ is estimate of covariance $\kappa_x[m]$ of random signal $x[n]$

$$\hat{\kappa}_x[m] = \frac{1}{N} \sum_{i=0}^{N-1-|m|} x[i]x[i+|m|] \approx E\left\{x[i]x[i+|m|]\right\} = \kappa_x[m]$$

  - smoothed periodogram computational procedure:
    1. estimate covariance
    2. multiply with window
    3. compute DFT
  - window length 2(M+1) should always be smaller than 2N-1, and typically $0.2 < M/N < 0.5$

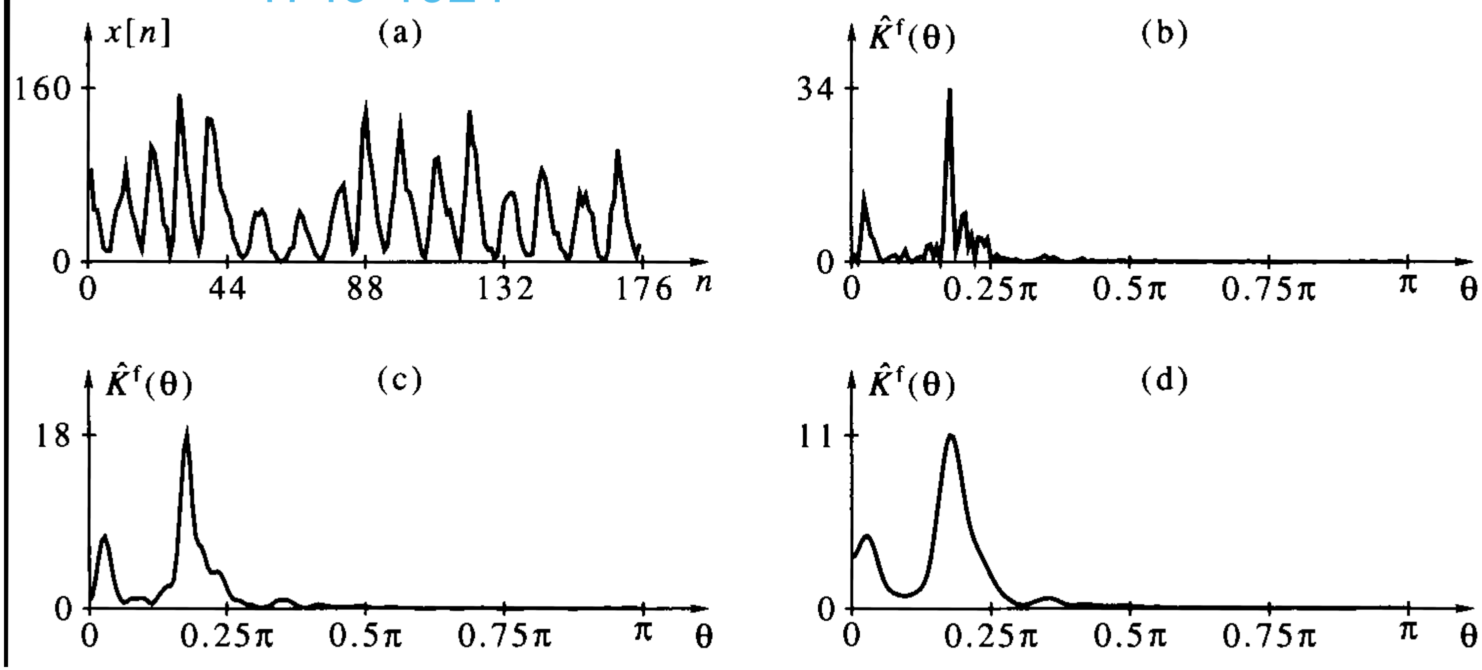
KU LEUVEN

Smoothed periodogram (3)

- Example: sunspot statistics

annual sunspot numbers
1749-1924

periodogram



smoothed periodogram
M = 88

smoothed periodogram
M = 44

Random signals: overview

- Introduction: motivating example
- Averaged periodogram
- Smoothed periodogram

+ MATLAB exercise

MATLAB exercise

1. Generate zero-mean unit-variance Gaussian white noise signal (length $NL=2^{16}$)
2. Filter this signal with FIR filter $H(z) = 1 + 0.8z^{-1}$
3. Calculate and plot periodogram of original and filtered signal
4. Calculate and plot averaged periodogram of original and filtered signal (try out different combinations of N and L)