4T1: The Short-Time Fourier Transform (1 of 2)

Xavier Serra

Universitat Pompeu Fabra, Barcelona

Index

- Short-time Fourier Transform equation
- Analysis window

Short-time Fourier Transform

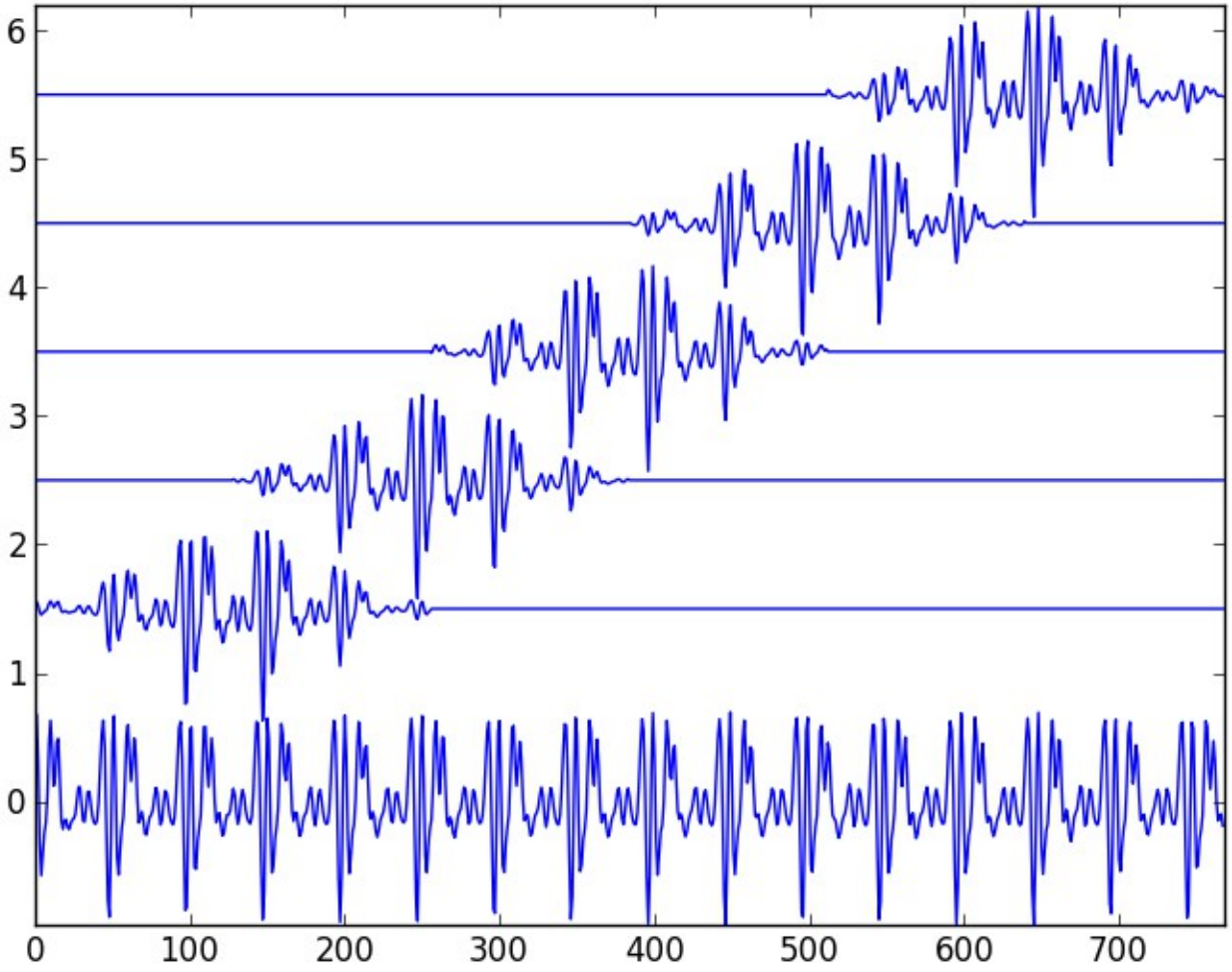
$$X_l[k] = \sum_{n=-N/2}^{N/2-1} w[n]\, x[n+lH]\, e^{-j2\pi kn/N} \qquad l = 0,1,\ldots,$$

w: analysis window

l: frame number

H: hop-size

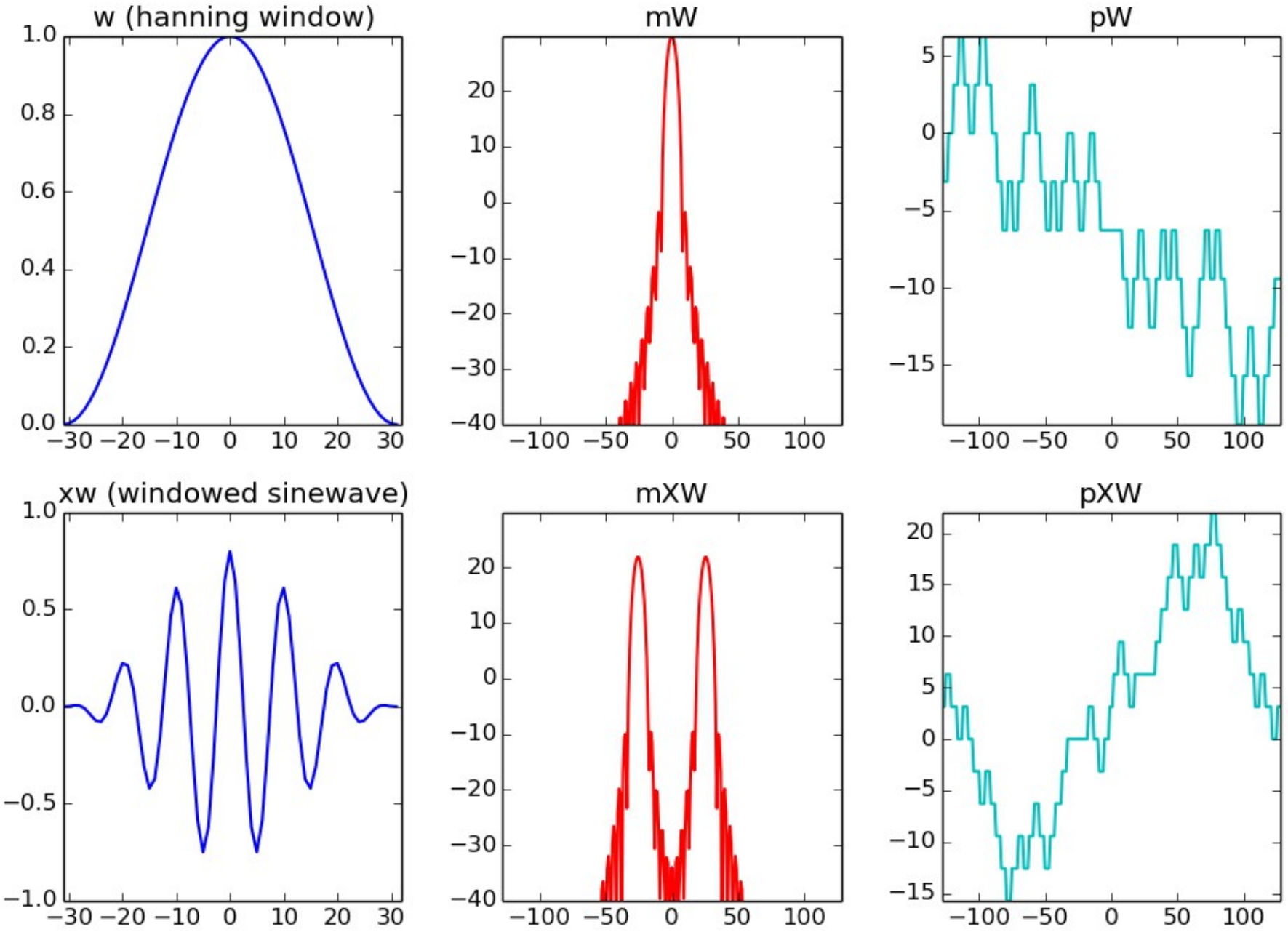
$$xw_l[n] = w[n]x[n + lH] \qquad l = 0,1,\ldots,$$



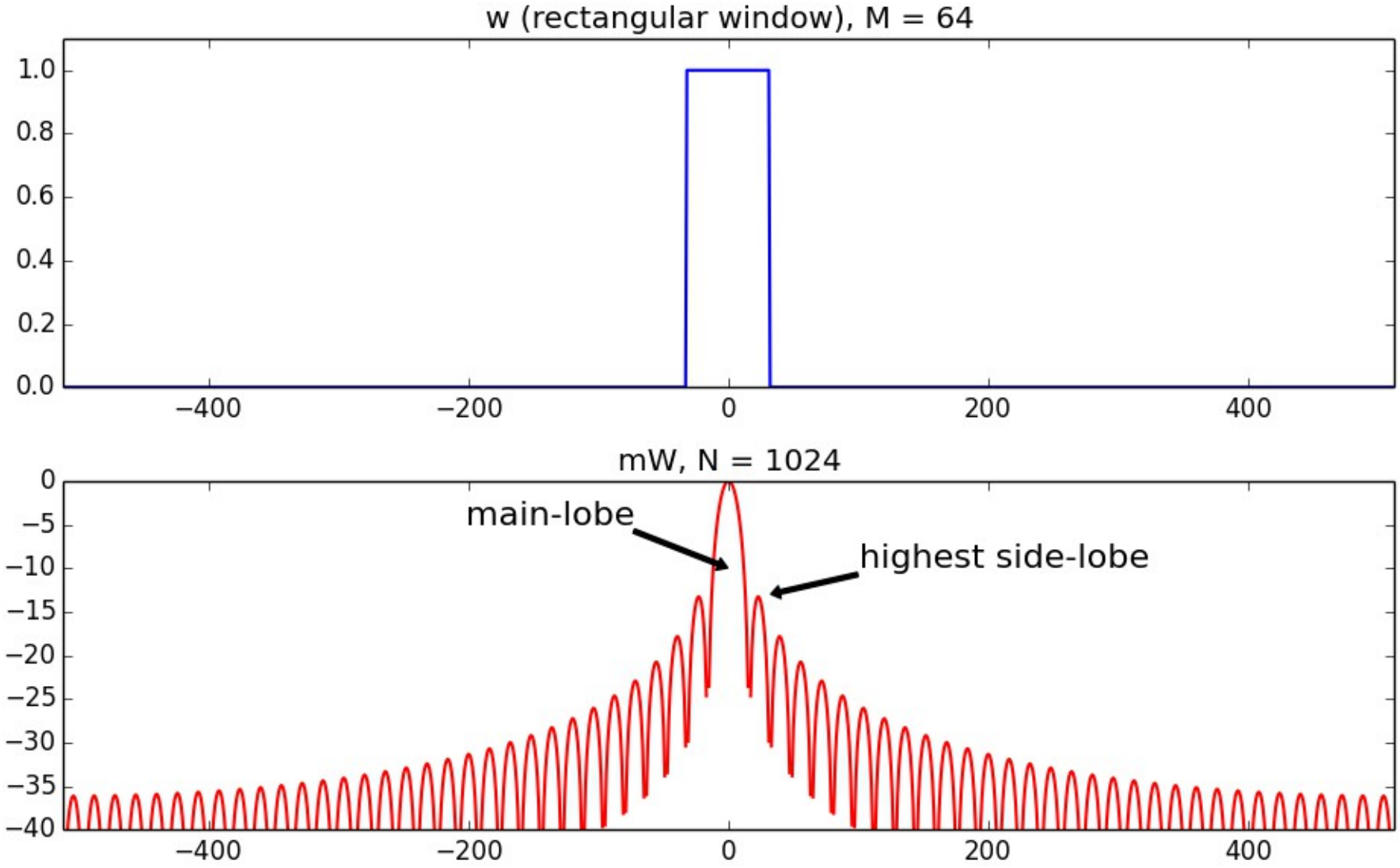
Transform of a windowed sinewave

$$x[n] = A_0 \cos(2\pi k_0 n / N) = \frac{A_0}{2} e^{j2\pi k_0 n/N} + \frac{A_0}{2} e^{-j2\pi k_0 n/N}$$

$$\begin{aligned}
X[k] &= \sum_{n=-N/2}^{N/2-1} w[n] x[n] e^{-j2\pi kn/N} \\
&= \sum_{n=-N/2}^{N/2-1} w[n] \left(\frac{A_0}{2} e^{j2\pi k_0 n/N} + \frac{A_0}{2} e^{-j2\pi k_0 n/N} \right) e^{-j2\pi kn/N} \\
&= \sum_{n=-N/2}^{N/2-1} w[n] \frac{A_0}{2} e^{j2\pi k_0 n/N} e^{-j2\pi kn/N} + \sum_{n=-N/2}^{N/2-1} w[n] \frac{A_0}{2} e^{-j2\pi k_0 n/N} e^{-j2\pi kn/N} \\
&= \frac{A_0}{2} \sum_{n=-N/2}^{N/2-1} w[n] e^{-j2\pi (k-k_0) n/N} + \frac{A_0}{2} \sum_{n=-N/2}^{N/2-1} w[n] e^{-j2\pi (k+k_0) n/N} \\
&= \frac{A_0}{2} W[k-k_0] + \frac{A_0}{2} W[k+k_0]
\end{aligned}$$


w (hanning window)
mW
pW
xw (windowed sinewave)
mXW
pXW

Analysis window



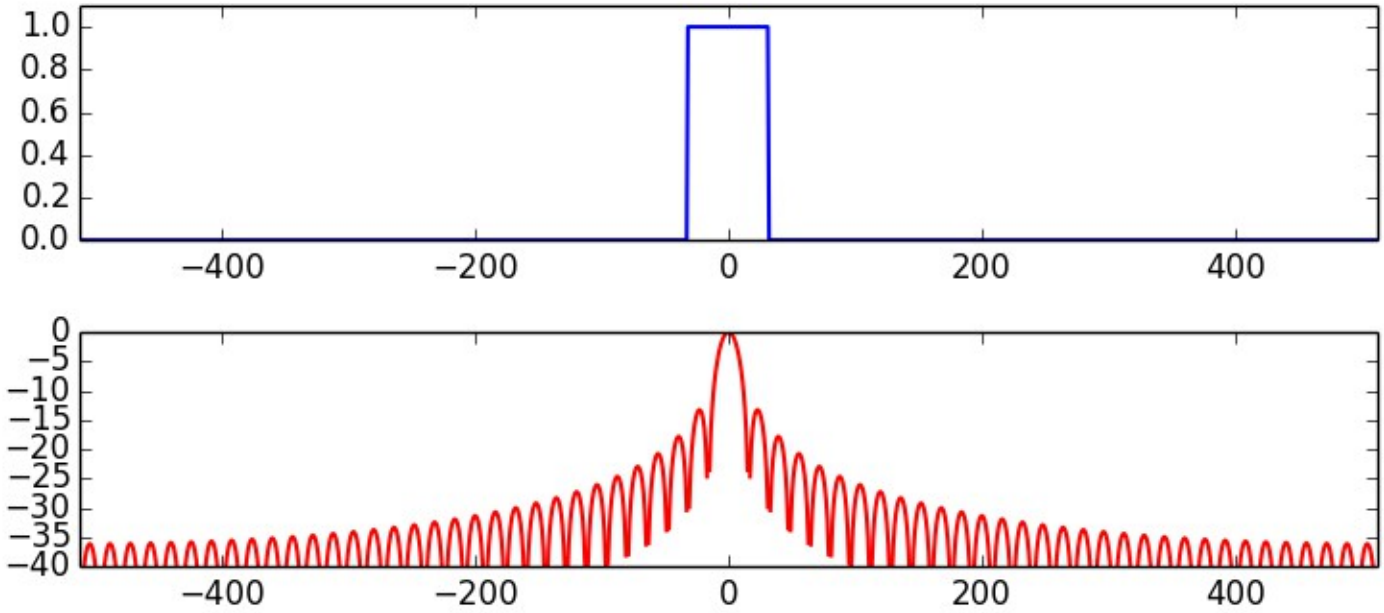
Window functions in Scipy

barthann (M[, sym])	Return a modified Bartlett-Hann window.
bartlett (M[, sym])	Return a Bartlett window.
blackman (M[, sym])	Return a Blackman window.
blackmanharris (M[, sym])	Return a minimum 4-term Blackman-Harris window.
bohman (M[, sym])	Return a Bohman window.
boxcar (M[, sym])	Return a boxcar or rectangular window.
chebwin (M, at[, sym])	Return a Dolph-Chebyshev window.
flattop (M[, sym])	Return a flat top window.
gaussian (M, std[, sym])	Return a Gaussian window.
general-gaussian (M, p, sig[, sym])	Return a window with a generalized Gaussian shape.
hamming (M[, sym])	Return a Hamming window.
hann (M[, sym])	Return a Hann window.
kaiser (M, beta[, sym])	Return a Kaiser window.
nuttall (M[, sym])	Return a minimum 4-term Blackman-Harris window according to Nuttall.
parzen (M[, sym])	Return a Parzen window.
slepian (M, width[, sym])	Return a digital Slepian window.
triang (M[, sym])	Return a triangular window.

Rectangular window

$$w[n]=1, \quad n=-M/2,\ldots,0,\ldots M/2$$
$$=0, \quad n=\text{elsewhere}$$

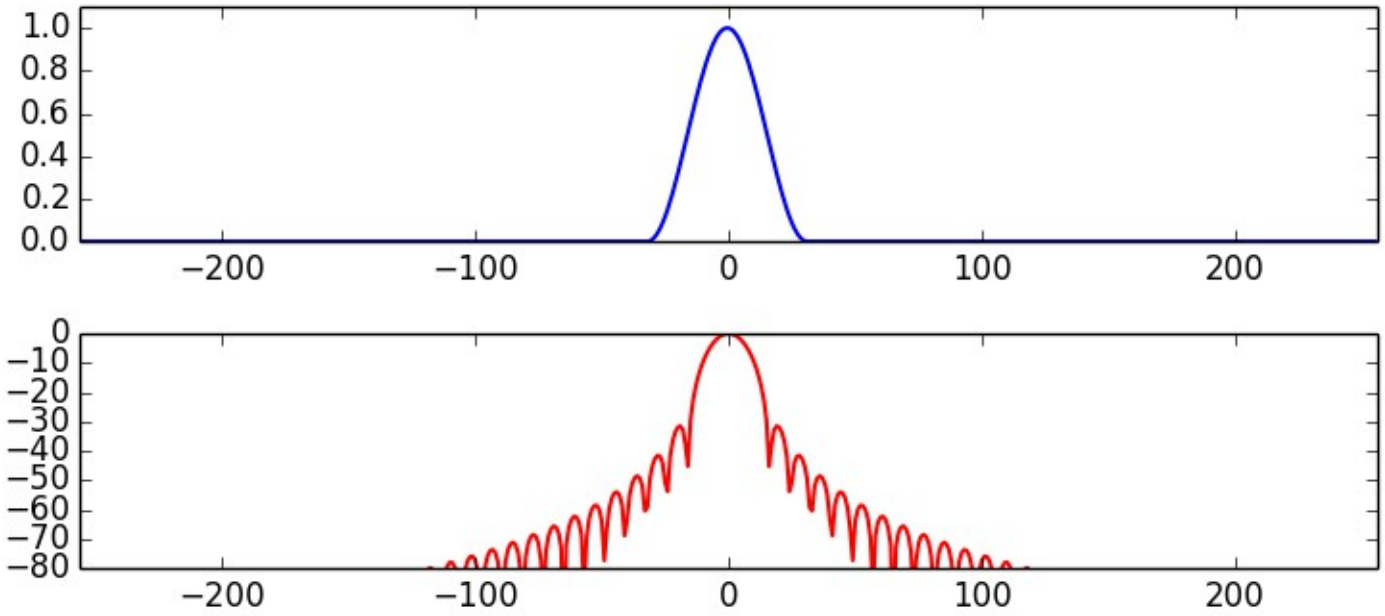
$$W[k]=\frac{\sin(\pi k)}{\sin(\pi k/M)}$$



Hanning window

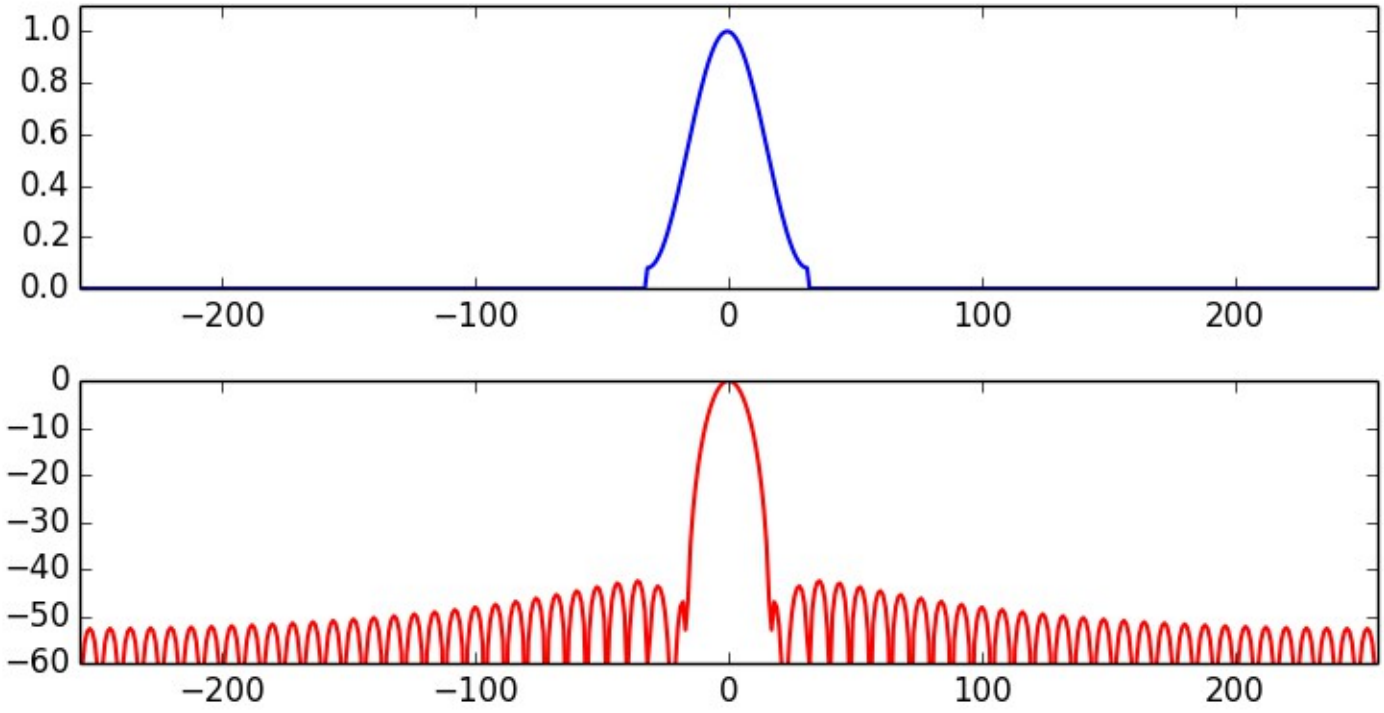
$$w[n] = .5 + .5\cos(2\pi n/M), \quad n = -M/2, \ldots, 0, \ldots M/2$$

$$W[k] = .5\,D[k] + .25\,(D[k-1] + D[k+1]) \quad \text{where } D[k] = \frac{\sin(\pi k)}{\sin(\pi k/M)}$$



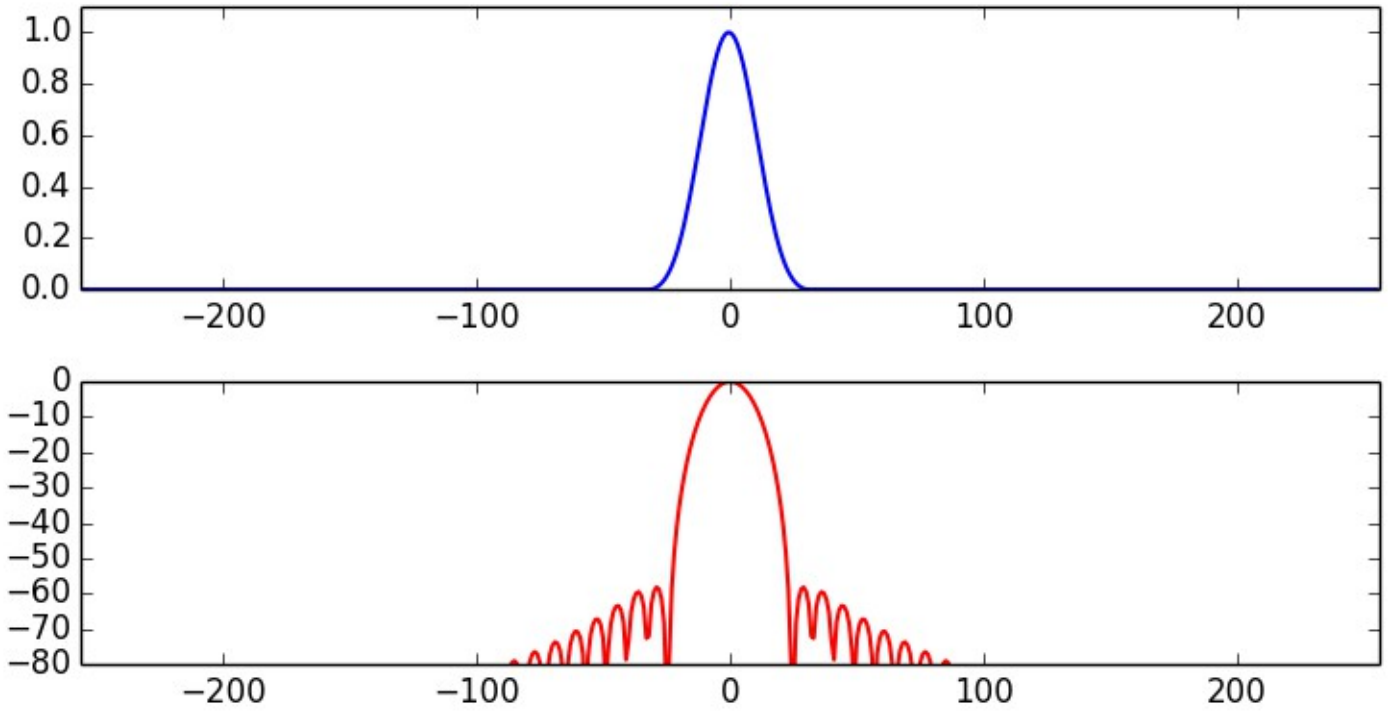
Hamming window

$$w[n] = .54 + .46\cos(2\pi n/M), \quad n = -M/2, \ldots, 0, \ldots M/2$$



Blackman window

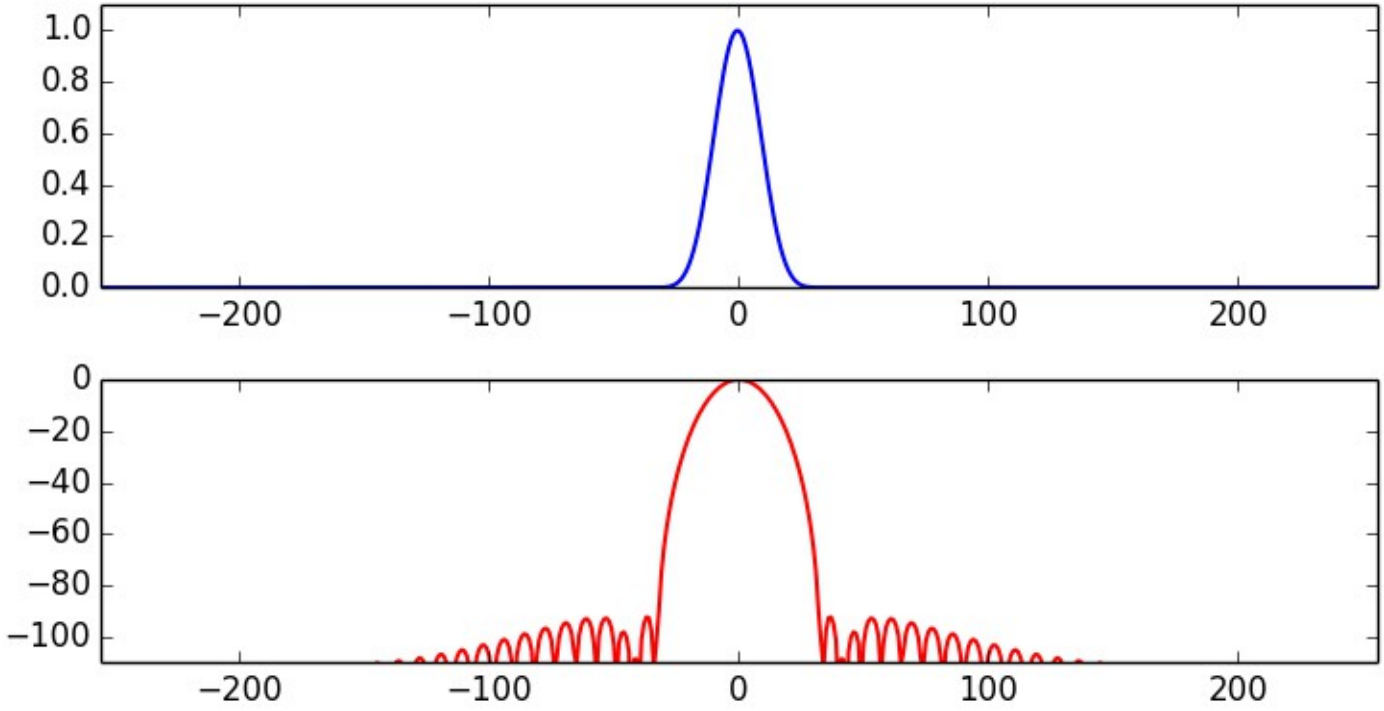
$$w[n] = 0.42 - 0.5\cos(2\pi n/M) + 0.08\cos(4\pi n/M)$$



Blackman-Harris window

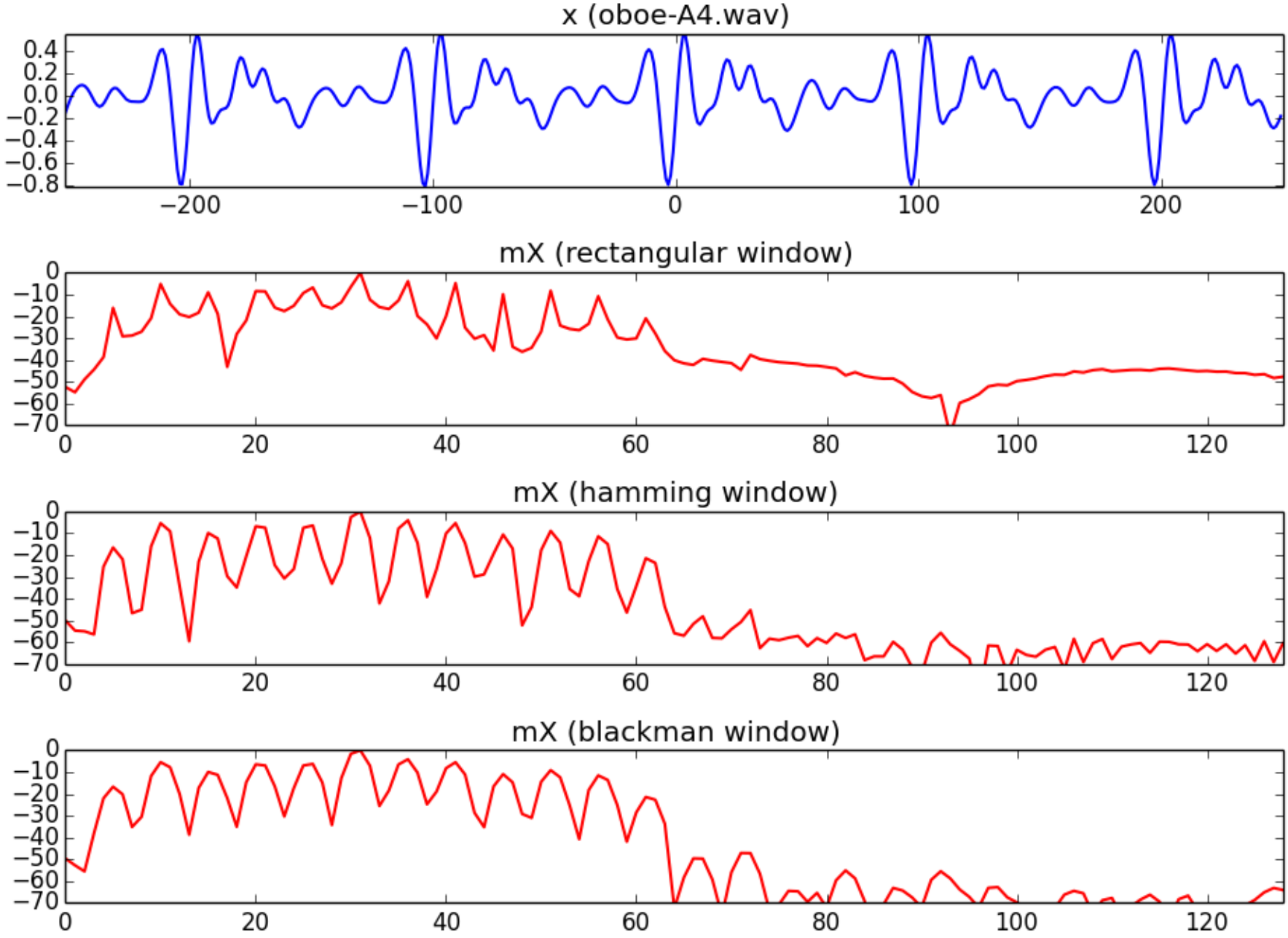
$$w(n)=\frac{1}{M}\sum_{l=0}^{3}\alpha_l\cos(2nl\pi/M),\quad n=-M/2,\ldots 0,\ldots M/2$$

where $\alpha_0=0.35875,\alpha_1=0.48829,\alpha_2=0.14128,\alpha_3=0.01168$



main lobe width : 8 bins

side-lobe level : −92dB


x (oboe-A4.wav)
0.4
0.2
0.0
−0.2
−0.4
−0.6
−0.8
−200
−100
0
100
200
mX (rectangular window)
0
−10
−20
−30
−40
−50
−60
−70
0
20
40
60
80
100
120
mX (hamming window)
0
−10
−20
−30
−40
−50
−60
−70
0
20
40
60
80
100
120
mX (blackman window)
0
−10
−20
−30
−40
−50
−60
−70
0
20
40
60
80
100
120

References and credits

- More information in:
  https://en.wikipedia.org/wiki/STFT
  https://en.wikipedia.org/wiki/Window_function
- Reference on the STFT by Julius O. Smith:
  https://ccrma.stanford.edu/~jos/sasp/
- Sounds from:
  http://www.freesound.org/people/xserra/packs/13038/
- Slides and code released using the CC Attribution-Noncommercial-Share Alike license or the Affero GPL license and available from
  https://github.com/MTG/sms-tools

4T1: The Short-Time Fourier Transform (1 of 2)

Xavier Serra
Universitat Pompeu Fabra, Barcelona