Digital Signal Processing for Music

Part 8: Fourier Transform, Part 2

Andrew Beck

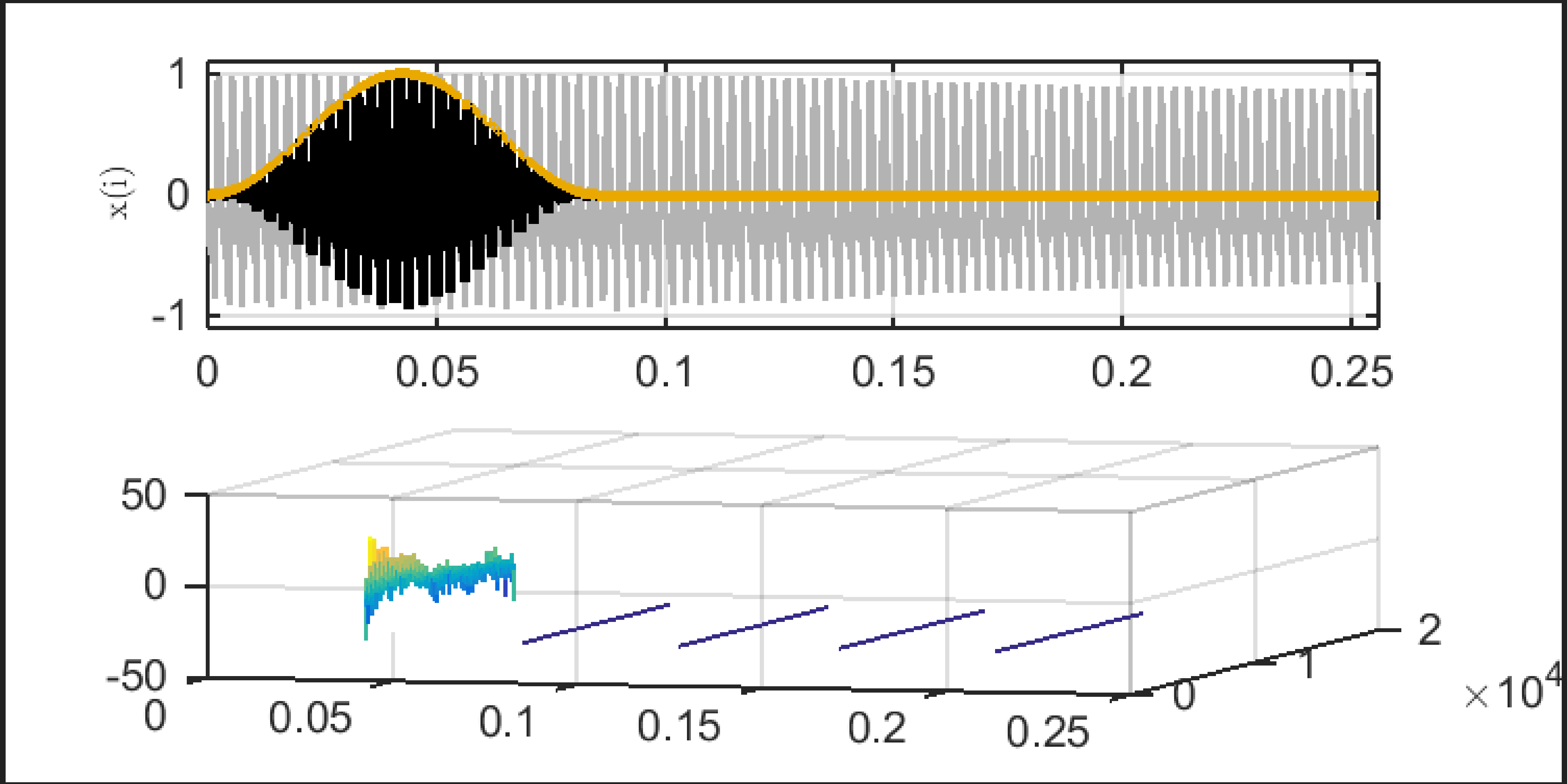
Short Time Fourier Transform (STFT)

Compute Fourier transform only over a segment

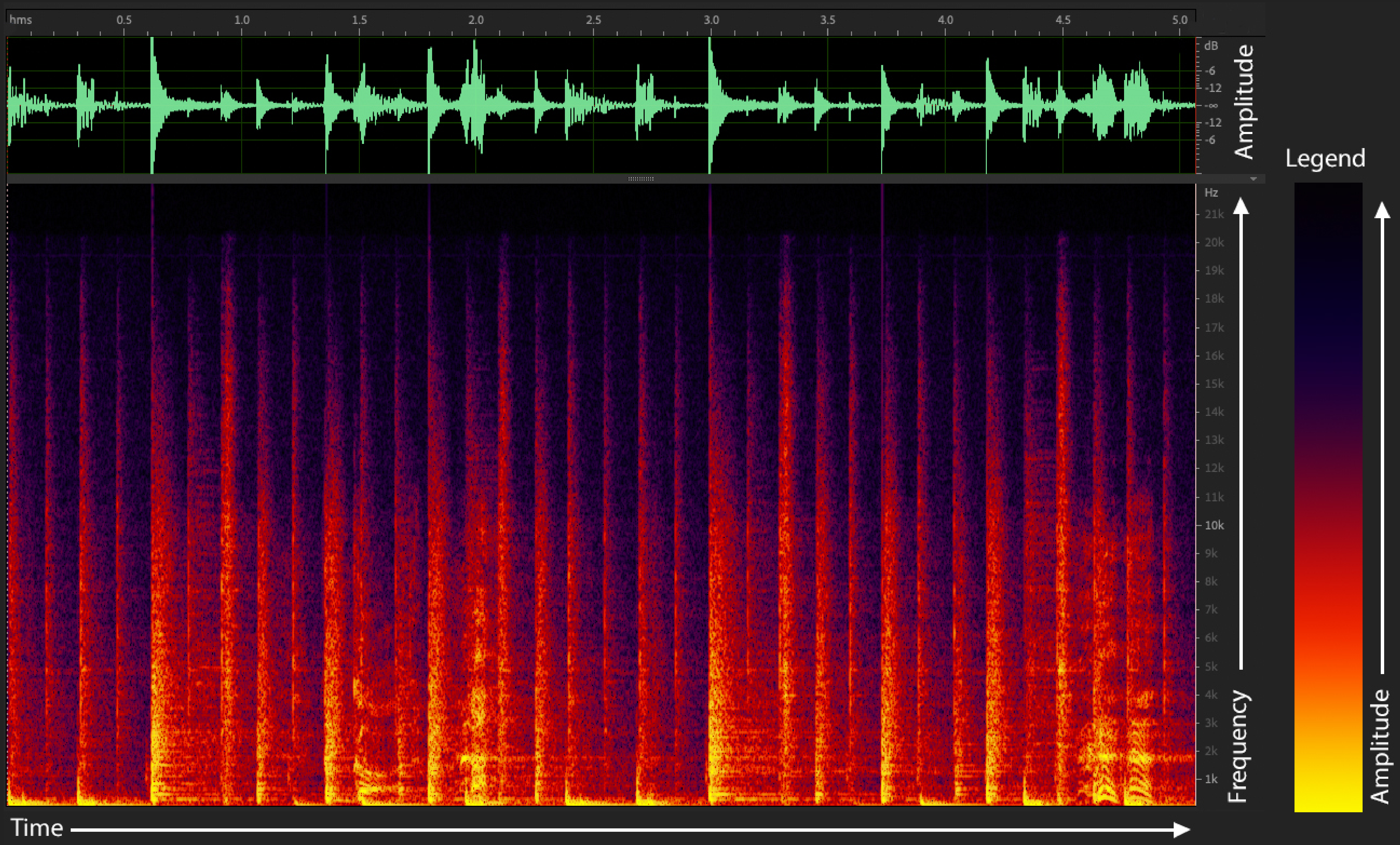
- **Signal Properties**: Choose quasi-periodic segment
- **Perception**: Ear analyzes short segments of signal
- **Hardware**: Fourier Transform is inefficient and memory consuming for very long input segments

Multiply a **window** with the signal

Animation of Process



Alternate Representation of STFT



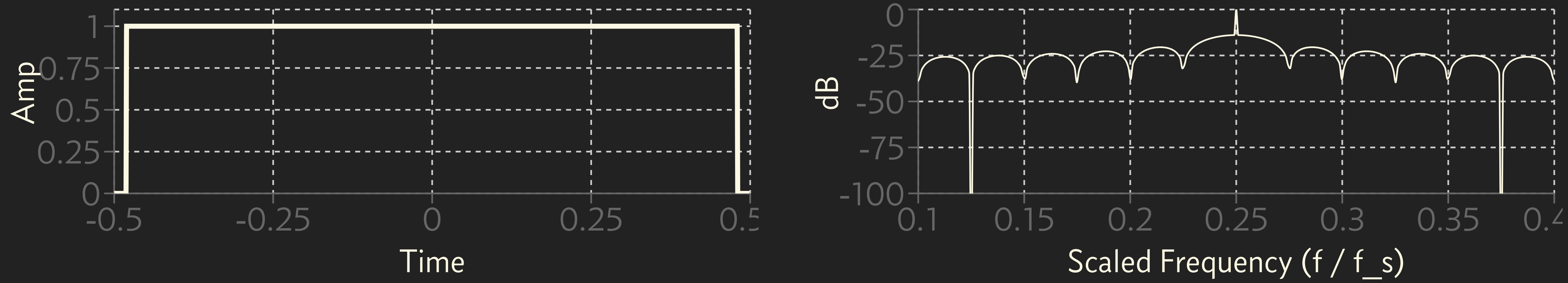
Windowing Effects in the Frequency Domain

Multiplication in time domain is convolution in the frequency domain

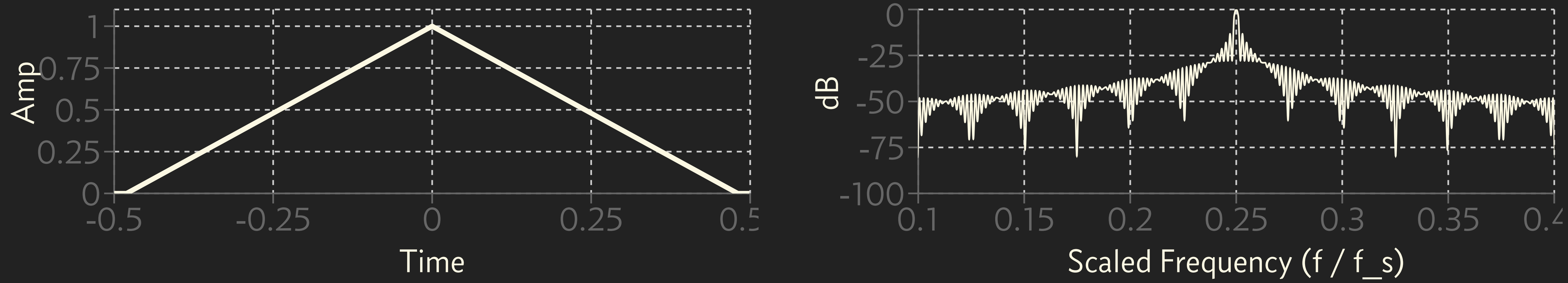
$$x_{\mathrm{W}}(t) = x(t) \cdot w(t) \rightarrow X_{\mathrm{W}}(\mathrm{j}\omega) = X(\mathrm{j}\omega) * W(\mathrm{j}\omega)$$

This causes **spectral leakage**

Rectangular

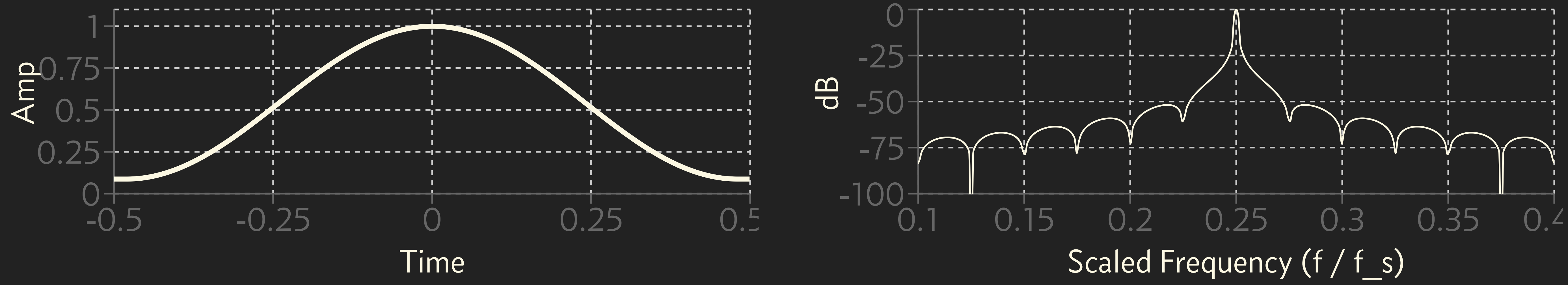


Triangle

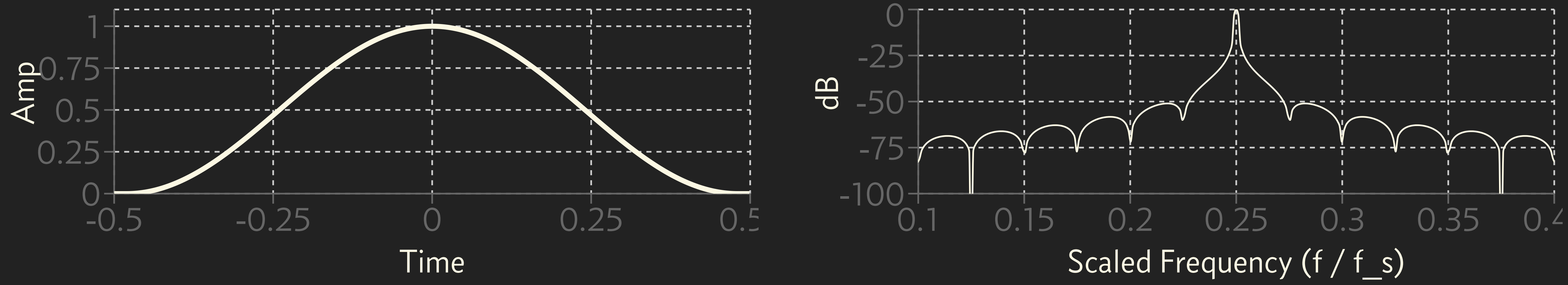


Cosine

Hamming



Von Hann



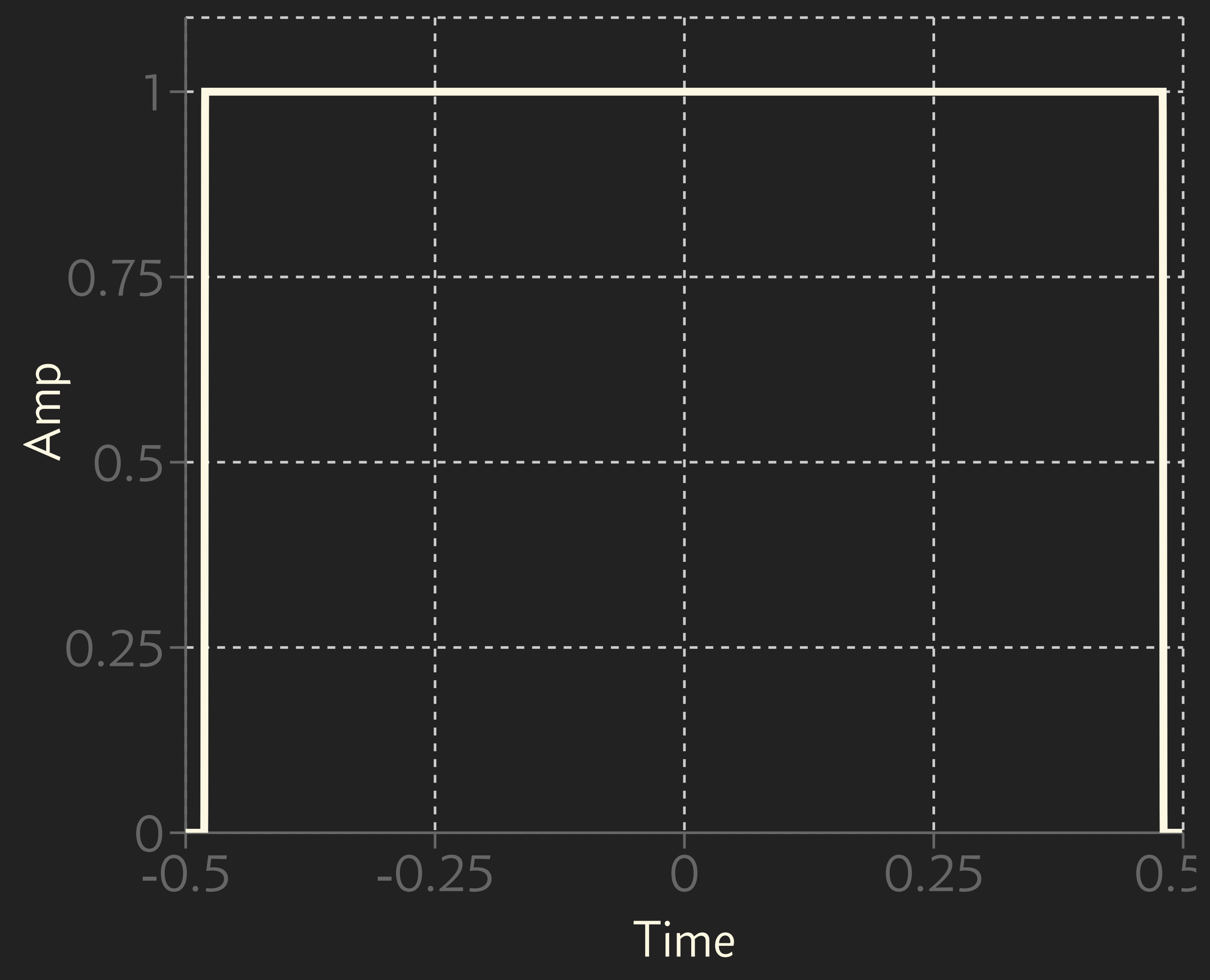
Blackman-Harris

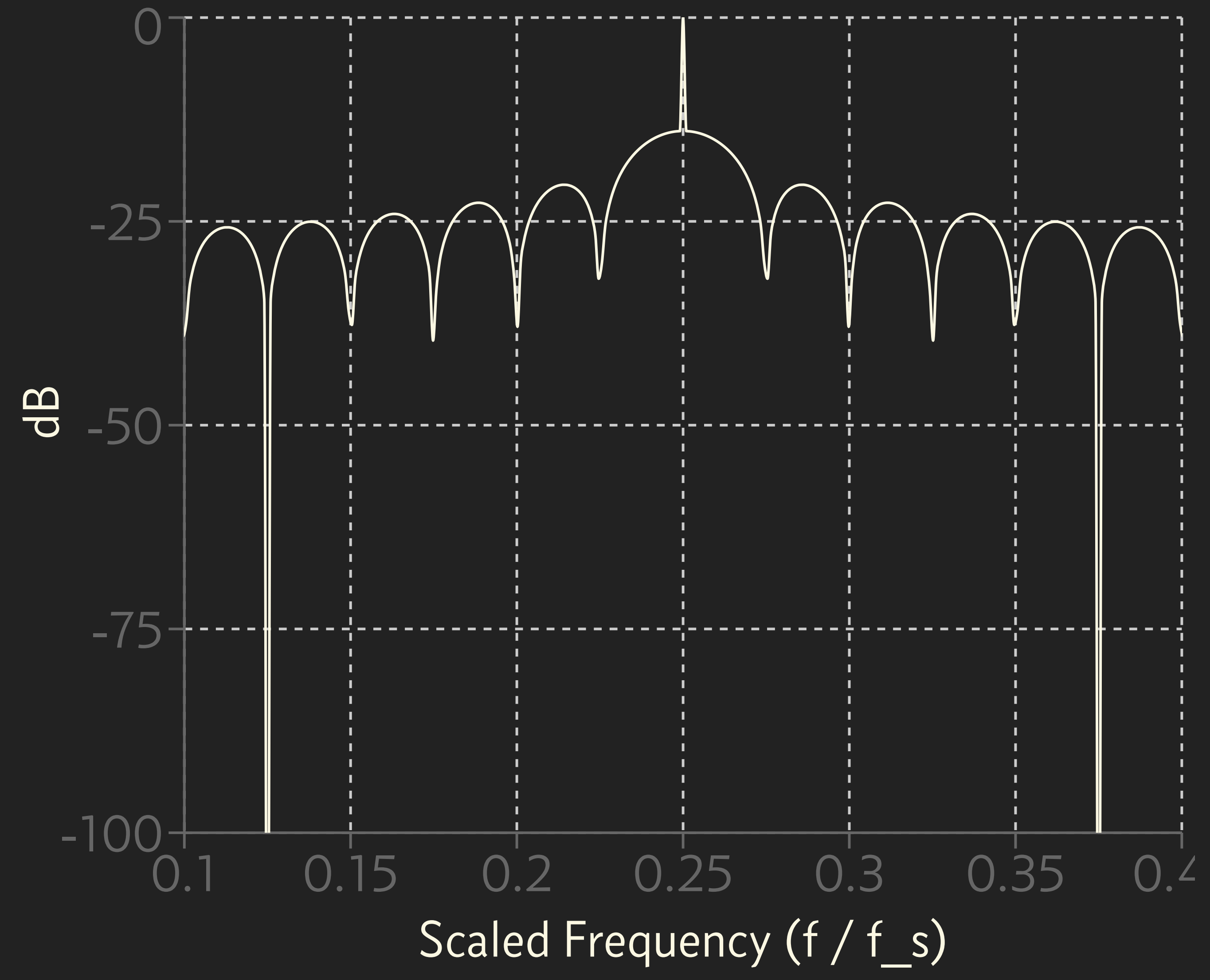
Properties of Spectral Leakage

- **Main Lobe Width**: How much does the main lobe "smear" a peak?
- **Side Lobe Height**: How domainant is the (highest) side lobe?
- **Side Lobe Attenuation/Fall-off**: How much do distant side lobes influence results?
- **Process and Scalloping Loss (DFT)**: How accurate is the amplitude (best and worst case)

Rectangular Window

$$w_{\mathrm{R}}(t) = \begin{cases} 1, & -\frac{1}{2} \le t \le \frac{1}{2} \\ 0, & \text{otherwise} \end{cases}$$



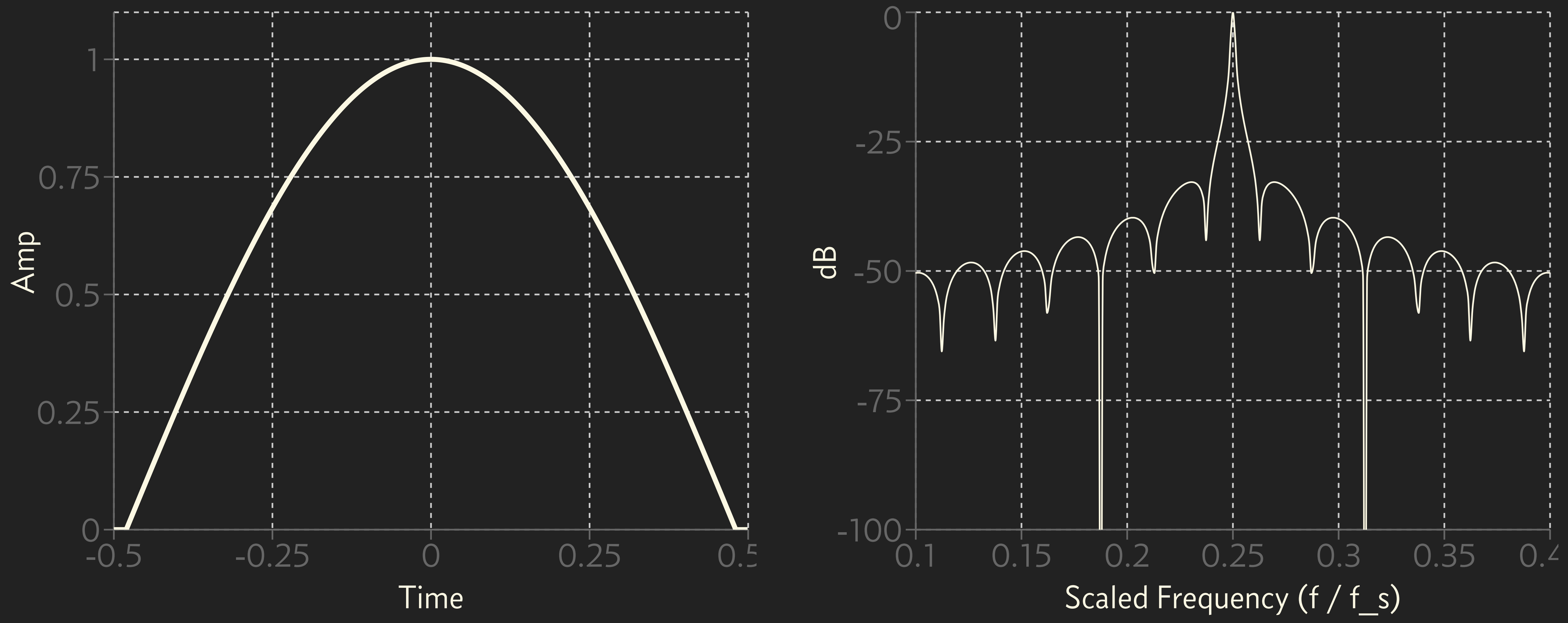


Triangle Window

$$w_{\mathrm{T}}(t) = \begin{cases} 1 + (2t), & -\frac{1}{2} \leq t < 0 \\ 2t, & 0 \leq t \leq \frac{1}{2} \\ 0, & \text{otherwise} \end{cases}$$

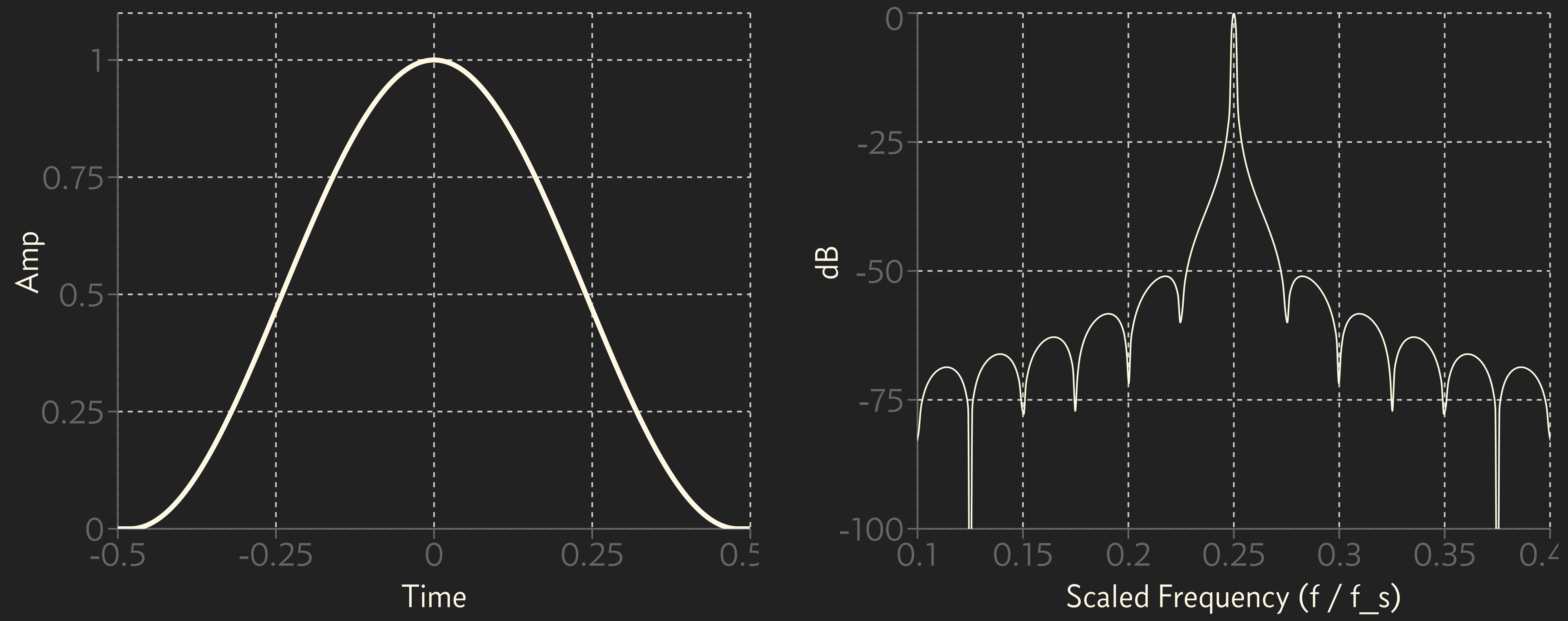
Cosine Window

$$w_{\mathrm{C}}(t) = w_{\mathrm{R}}(t) \cdot \cos(\pi t)$$



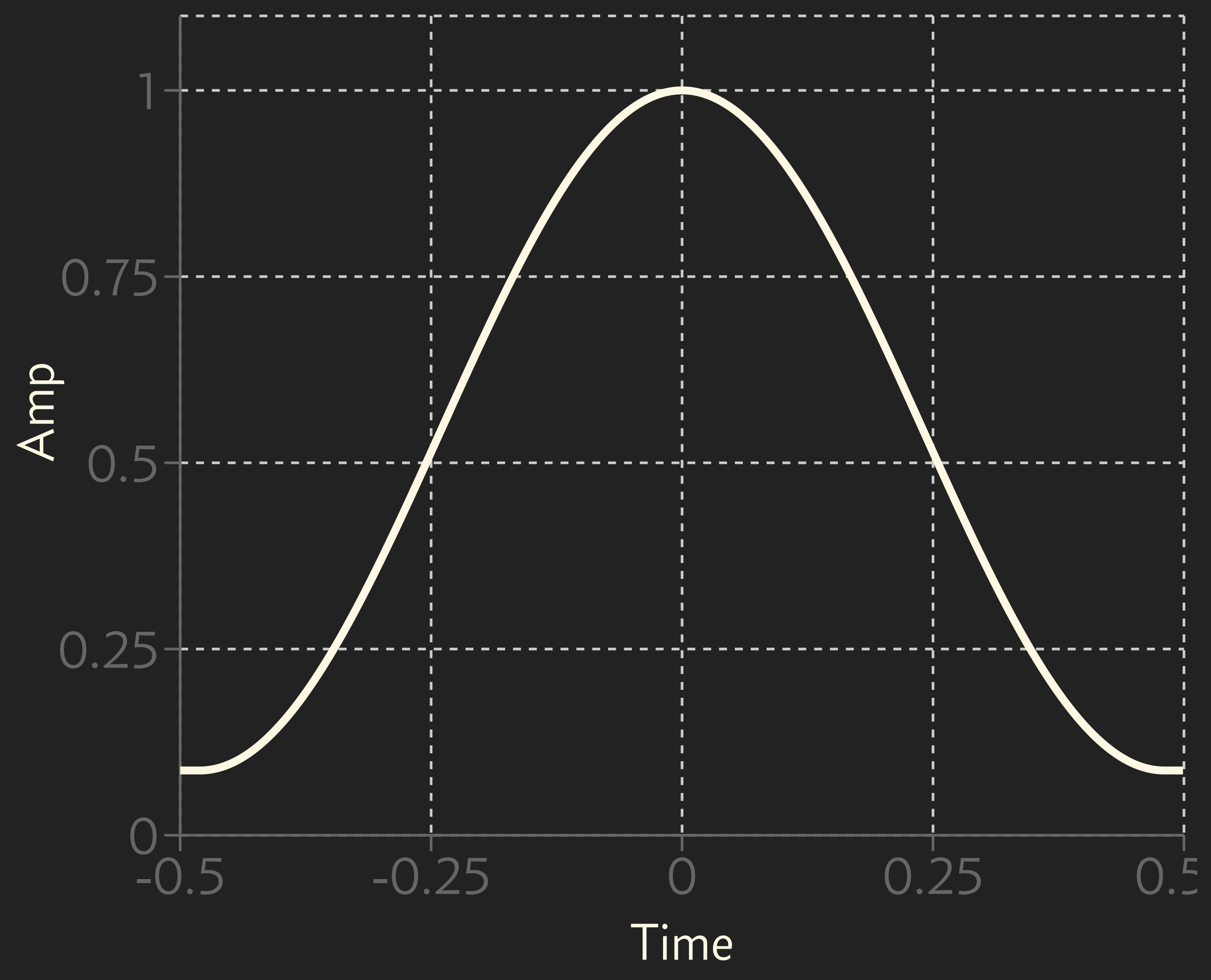
Von-Hann Window

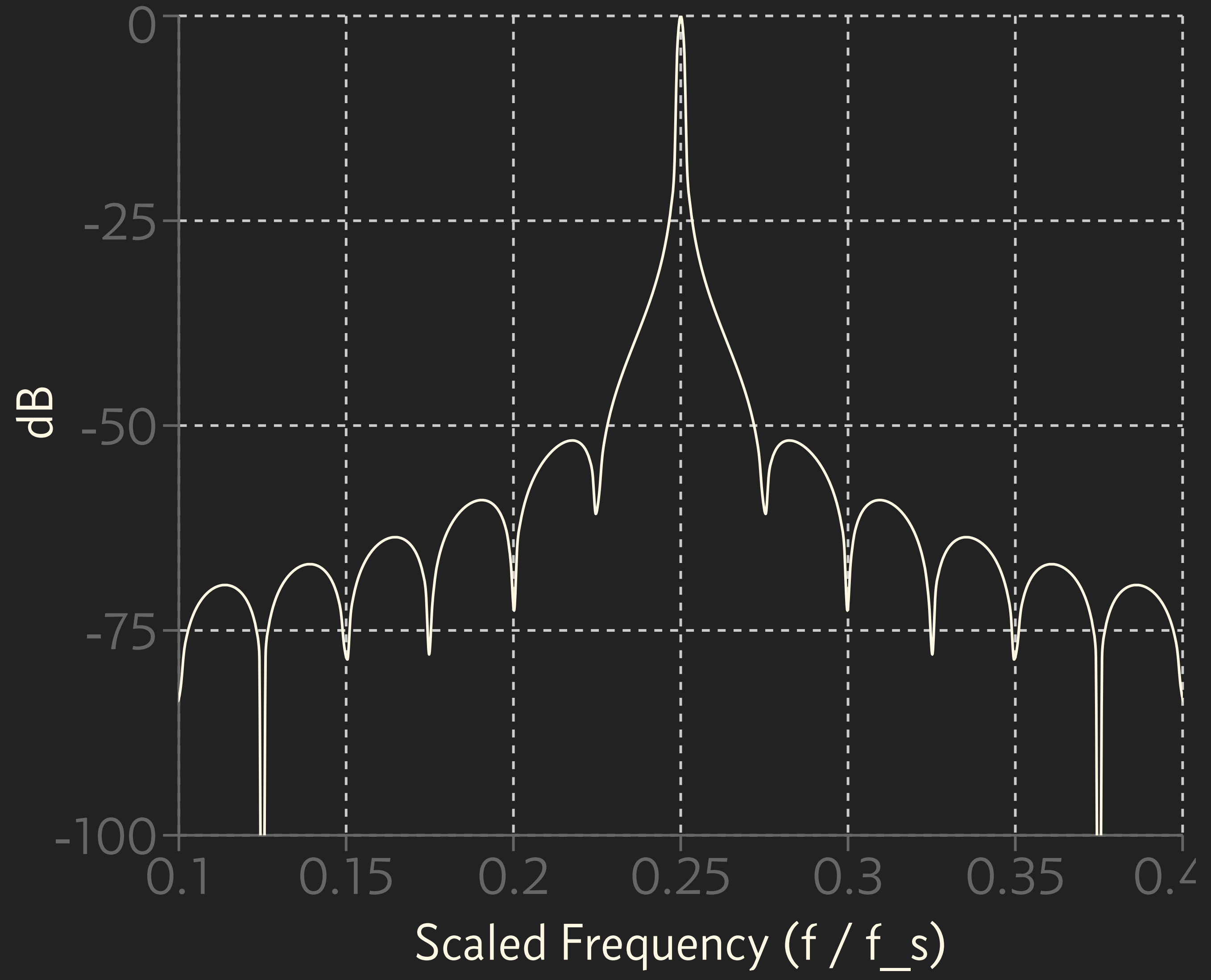
$$w_{\mathrm{H}}(t) = w_{\mathrm{R}}(t) \cdot \frac{1}{2}(1 + \cos(2\pi t))$$



Hamming Window

$$w_{\mathrm{Hm}}(t) = w_{\mathrm{R}}(t) \cdot \left(\frac{25}{46} + \frac{21}{46}\cos(2\pi t)\right)$$





Blackman-Harris Window

$$w_{\mathrm{BH}}(t) = w_{\mathrm{R}}(t) \cdot \sum_{m=0}^{3} b_m \cos\left(2\pi m t\right)$$

with $b_0 = 0.35875$, $b_1 = 0.48829$, $b_2 = 0.14128$, $b_3 = 0.01168$

Sampled Time Signals

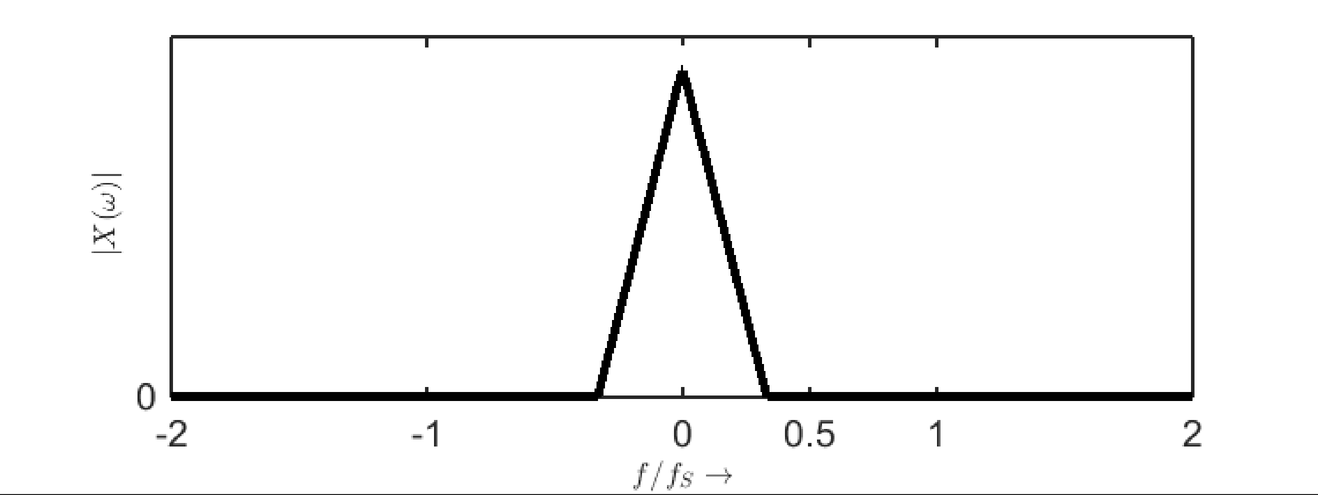
$$\mathfrak{F}[x(i)] = \mathfrak{F}[x(t) \cdot \delta_{\mathrm{T}}(t)]$$

$$= \mathfrak{F}[x(t)] * \mathfrak{F}[\delta_{\mathrm{T}}(t)]$$

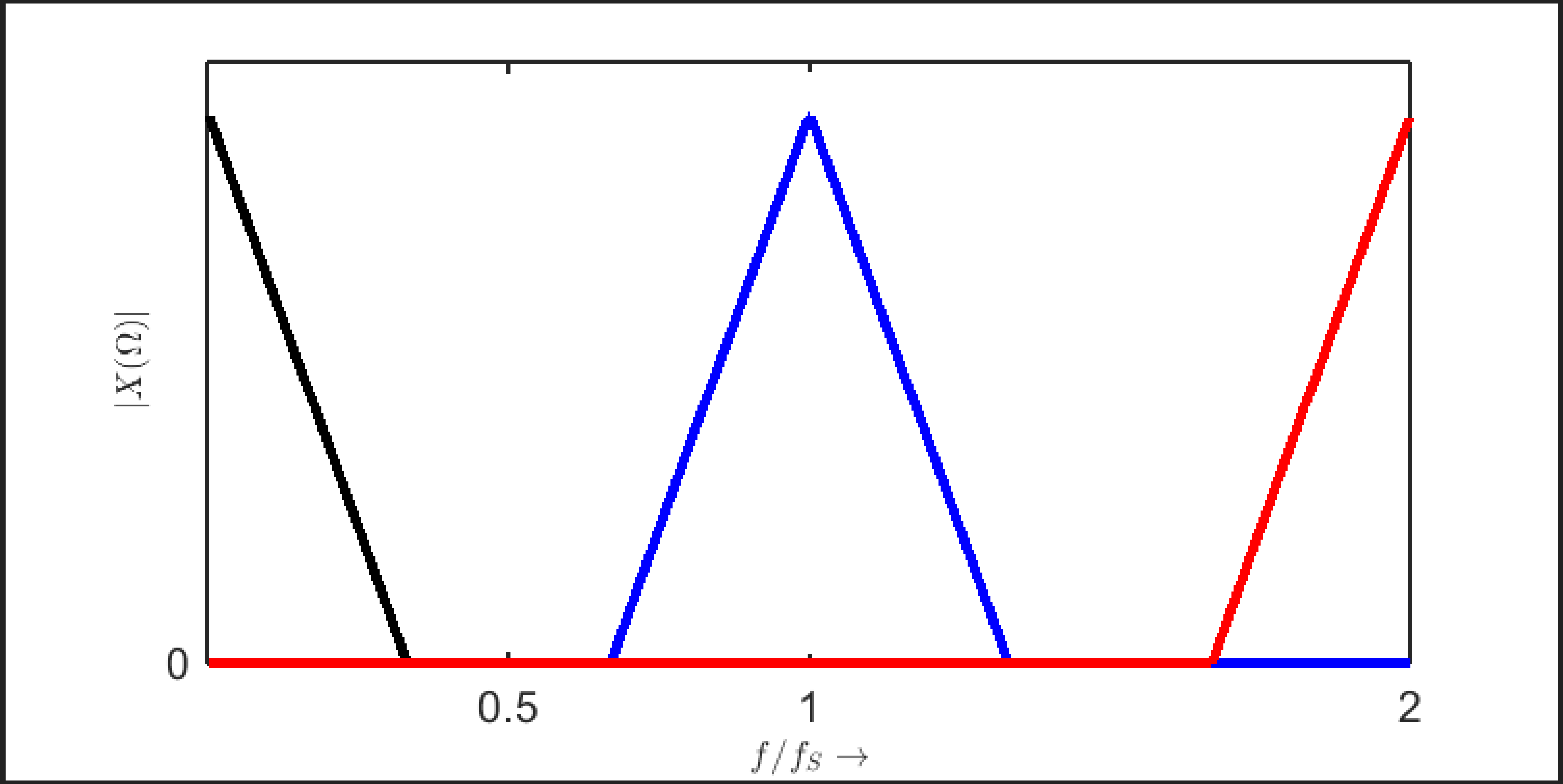
$$= X(\mathrm{j}\omega) * \Delta_{\mathrm{T}}(\mathrm{j}\omega)$$

Transformed signal is

- still **continuous**
- **periodic**



Spectral Aliasing in Sampled Signals



Discrete Fourier Transform (DFT)

Digital domain requires working with discrete frequency values

$$X(k) = \sum_{i=0}^{\mathcal{K}-1} x(i) \mathrm{e}^{-\mathrm{j} k i \frac{2\pi}{\mathcal{K}}}$$

2 Interpretations:

- Sampled continuous Fourier transform
- Continuous Fourier transform of periodically extended time domain segment

DFT Frequency Resolution

Depends on:

- Block length $\mathcal{K}$
- Sample rate ω_T (spectrum is periodic with ω_T)

$$\Delta\omega = \frac{\omega_T}{\mathcal{K}}$$

- Increasing DFT length increases frequency resolution, decreasing time resolution
- Zero-padding can increase resolution without decreasing time resolution

DFT vs FFT

FFT is an algorithm to efficiently calculate the DFT

Result is **identical**

- DFT: $\mathcal{K}^2$ complex multiplications
- FFT: $\frac{\mathcal{K}}{2}\log_2(\mathcal{K})$ complex multiplications

K	DFT Calcs	FFT Calcs	Efficiency
256	2^16	1024	64 : 1
512	2^18	2034	114 : 1
1024	2^20	5120	205 : 1
2048	2^22	11264	372 : 1
4096	2^24	24576	683 : 1

Summary

1. Fourier Series can describe any periodic function *(discrete "spectrum")*
2. Continuous FT transforms any continuous function *(continuous spectrum)*
3. STFT transforms a segment of the signal *(convolution with window spectrum)*
4. FT of sampled signals *(periodic)*
5. DFT *(sampled FT of periodic continuation)*

» **Where spectrum is periodic, time signal is discrete**

» **Where spectrum is discrete, time signal is periodic**