6.3000: Signal Processing

Short-Time Fourier Transform

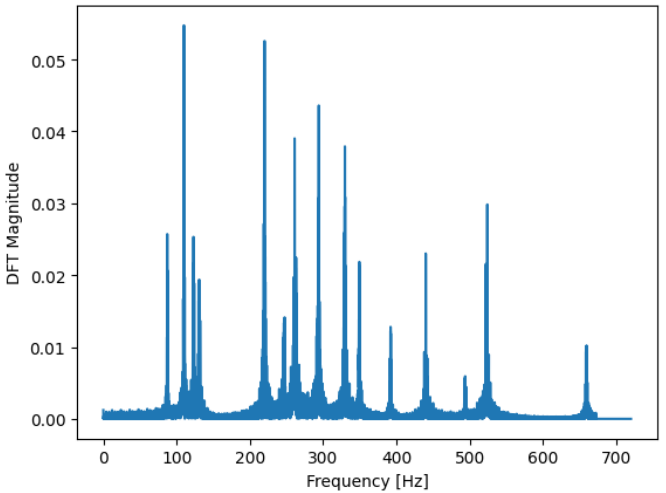
- Spectrograms
- Window Functions

November 3, 2026

Music Clip

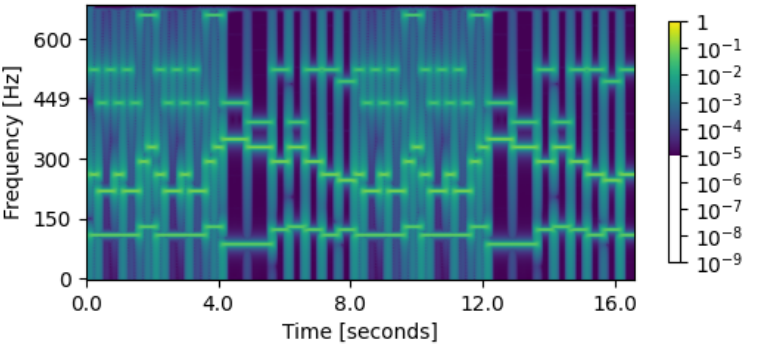
In lecture, we saw three representations for the same music clip.

The first was the magnitude of the DFT (shown below).
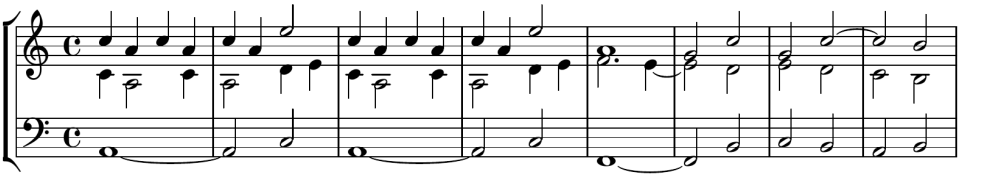


Music Clip

The second was the spectrogram.

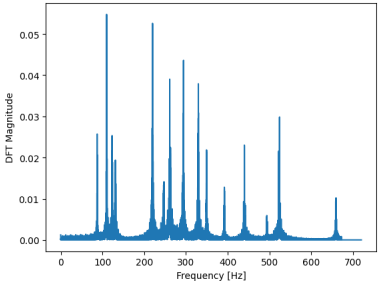


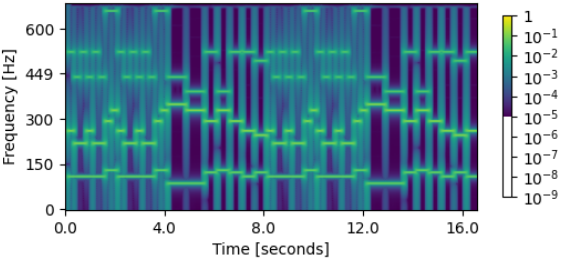
The third was the musical score.



Music Clip

Compare features of these representations.

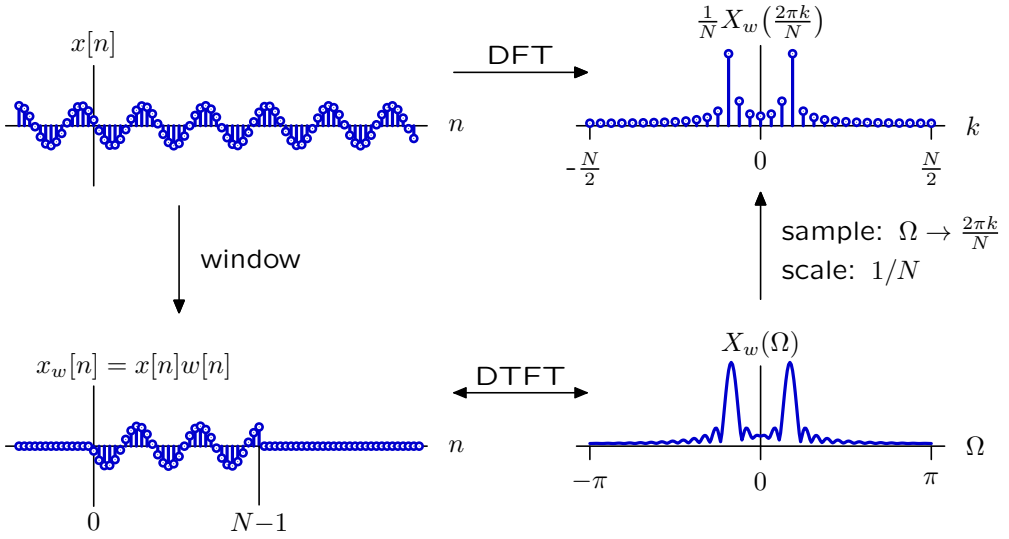




- What do these representations have in common?
  How are they different?
- Which representation has the best frequency resolution? Why?
- Which representation has the best time resolution? Why?
- The left panel has peaks with different heights. How do the different heights in the left panel correspond to features of the spectrogram?

Window Functions

A defining feature of the DFT is its finite length N, which plays a critical role in determining both time and frequency resolution.



The finite length constraint is equivalent to multiplication by a rectangular window. What would happen if we used a different type of window?

Windows

Multiplying $x[n]$ by the window function $w[n]$ corresponds to convolving the DTFT of $x[n]$ with the DTFT of $w[n]$.

Windowing

$$x_w[n] = x[n]w[n] \iff X_w(\Omega) = \frac{1}{2\pi}\big(X * W\big)(\Omega)$$

- time resolution: short $w[n] \iff$ wide $W(\Omega)$
- frequency resolution: long $w[n] \iff$ narrow $W(\Omega)$

There is a **trade-off** between time and frequency resolution.

Window Functions

Dozens of different window functions are in common use. We will look at three of them:

- rectangular window
- triangular window
- Hann window

These and other window functions have a variety of different properties. We would like to understand which properties are important in which applications.

Rectangular Window

Definition:

$$w_r[n] = \begin{cases} \frac{1}{2M-1} & 0 \leq n < 2M-1 \\ 0 & \text{otherwise} \end{cases}$$

- Make a plot of $w_r[n]$ versus n.
- Determine the DT Fourier Transform $W_r(\Omega)$.
- Make a plot of $W_r(\Omega)$ versus Ω.

Rectangular Window

Definition:

$$w_r[n] = \begin{cases} \frac{1}{2M-1} & 0 \leq n < 2M-1 \\ 0 & \text{otherwise} \end{cases}$$

One approach would be to close the following sum analytically:

$$W_r(\Omega) = \sum_{n=-\infty}^{\infty} w_r[n] e^{-j\Omega n}$$

Alternatively, we could evaluate the above sum for $\Omega = \frac{2\pi k}{N}$ using a DFT:

$$W_r\left(\frac{2\pi k}{N}\right) = \sum_{n=-\infty}^{\infty} w_r[n] e^{-j\frac{2\pi k}{N} n}$$

Since $w_r[n] = 0$ outside the range $0 \leq n \leq 2M-2$ we can reduce the infinite sum to a finite sum, which can then be evaluated with a DFT.

$$W_r\left(\frac{2\pi k}{N}\right) = \sum_{n=0}^{2M-2} w_r[n] e^{-j\frac{2\pi k}{N} n} = N \times \text{DFT}\{w_r\}$$

We can choose the analysis length N of the DFT based on our desired frequency resolution.

Rectangular Window

Definition:

$$w_r[n] = \begin{cases} \frac{1}{2M-1} & 0 \leq n < 2M-1 \\ 0 & \text{otherwise} \end{cases}$$

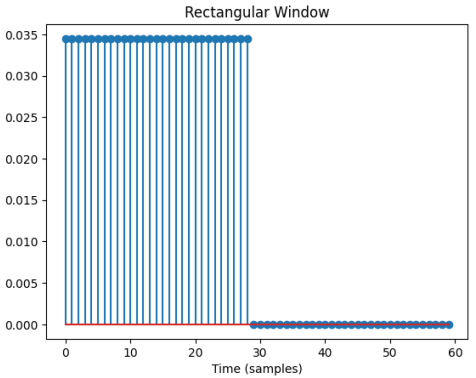
```
from matplotlib.pyplot import plot,stem,xticks,xlabel,title,legend,show
from lib6300.fft import fft
from math import pi,cos,sin

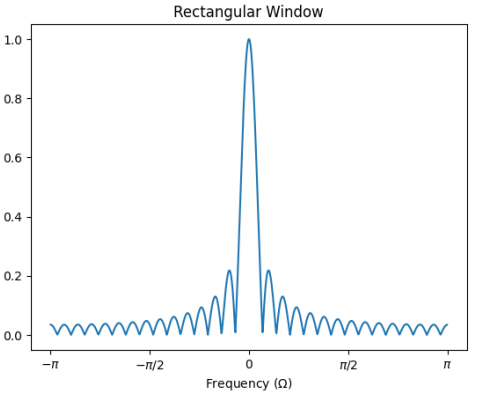
M = 15
N = 1024
w = [1/(2*M-1) for n in range(2*M-1)]
stem(w+(60-len(w))*[0])
xlabel('Time (samples)')
title('Rectangular Window')
show()
W = fft(w+(N-len(w))*[0])
plot([2*pi*k/N for k in range(-N//2,N//2)],[abs(W[k]/W[0]) for k in range(-N//2,N//2)])
xticks([-pi,-pi/2,0,pi/2,pi],['$-\pi$','$-\pi/2$','0','$\pi/2$','$\pi$'])
xlabel('Frequency ($\Omega$)')
title('Rectangular Window')
show()
```

Rectangular Window

Definition:

$$w_r[n] = \begin{cases} \frac{1}{2M-1} & 0 \le n < 2M-1 \\ 0 & \text{otherwise} \end{cases}$$





Triangular Window

Definition:

$$w_t[n] = \begin{cases} \frac{n+1}{M^2} & \text{if } 0 \le n < M \\ \frac{2M-n-1}{M^2} & \text{if } M \le n < 2M-1 \\ 0 & \text{otherwise} \end{cases}$$

- Make a plot of $w_t[n]$ versus n.
- Determine the DT Fourier Transform $W_t(\Omega)$.
- Make a plot of $W_t(\Omega)$ versus Ω.

Triangular Window

Definition:

$$w_t[n] = \begin{cases} \frac{n+1}{M^2} & \text{if } 0 \le n < M \\ \frac{2M-n-1}{M^2} & \text{if } M \le n < 2M-1 \\ 0 & \text{otherwise} \end{cases}$$

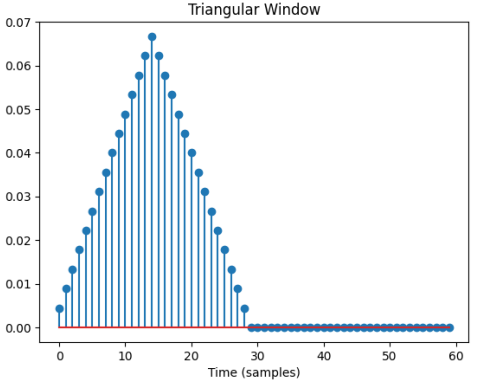
```
from matplotlib.pyplot import ion,plot,stem,xticks,xlabel,title,legend,show
from lib6300.fft import fft
from math import pi,cos,sin

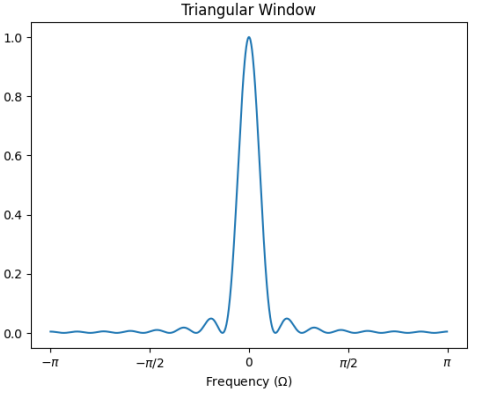
w = [(n+1)/M/M for n in range(M)]+[(2*M-n-1)/M/M for n in range(M,2*M-1)]
stem(w+100*[0])
xlabel('Time (samples)')
title('Triangular Window')
show()
W = fft(w+(N-len(w))*[0])
plot([2*pi*k/N for k in range(-N//2,N//2)],[abs(W[k]/W[0]) for k in range(-N//2,N//2)])
xticks([-pi,-pi/2,0,pi/2,pi],['$-\pi$','$-\pi/2$','0','$\pi/2$','$\pi$'])
xlabel('Frequency ($\Omega$)')
title('Triangular Window')
show()
```

Triangular Window

Definition:

$$w_t[n] = \begin{cases} \frac{n+1}{M^2} & \text{if } 0 \leq n < M \\ \frac{2M-n-1}{M^2} & \text{if } M \leq n < 2M-1 \\ 0 & \text{otherwise} \end{cases}$$





Hann Window

Definition:

$$w_h[n] = \begin{cases} \frac{1}{M}\sin^2\left(\frac{\pi(n+1)}{2M}\right) & 0 \le n < 2M-1 \\ 0 & \text{otherwise} \end{cases}$$

- Make a plot of $w_h[n]$ versus n.
- Determine the DT Fourier Transform $W_h(\Omega)$.
- Make a plot of $W_h(\Omega)$ versus Ω.

Hann Window

Definition:

$$w_h[n] = \begin{cases} \frac{1}{M}\sin^2\left(\frac{\pi(n+1)}{2M}\right) & 0 \le n < 2M-1 \\ 0 & \text{otherwise} \end{cases}$$

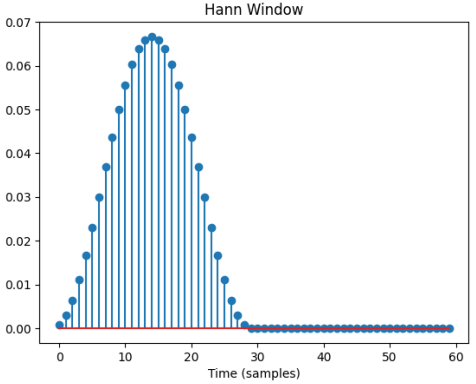
```
from matplotlib.pyplot import ion,plot,stem,xticks,xlabel,title,legend,show
from lib6300.fft import fft
from math import pi,cos,sin

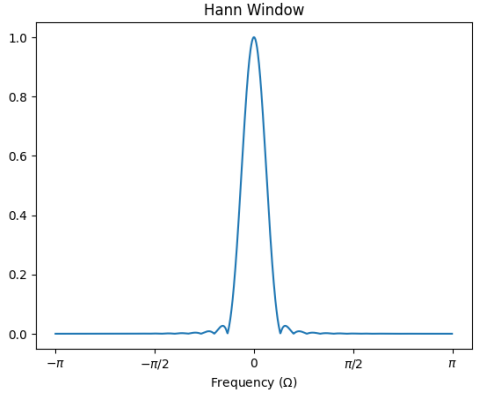
M = 15
N = 1024
w = [sin(pi*(n+1)/(2*M))**2/M for n in range(2*M-1)]
stem(w+100*[0])
xlabel('Time (samples)')
title('Hann Window')
show()
W = fft(w+(N-len(w))*[0])
plot([2*pi*k/N for k in range(-N//2,N//2)],[abs(W[k]/W[0]) for k in range(-N//2,N//2)])
xticks([-pi,-pi/2,0,pi/2,pi],['$-\pi$','$-\pi/2$','0','$\pi/2$','$\pi$'])
xlabel('Frequency ($\Omega$)')
title('Hann Window')
show()
```

Hann Window

Definition:

$$w_h[n] = \begin{cases} \frac{1}{M} \sin^2\left(\frac{\pi(n+1)}{2M}\right) & 0 \le n < 2M-1 \\ 0 & \text{otherwise} \end{cases}$$



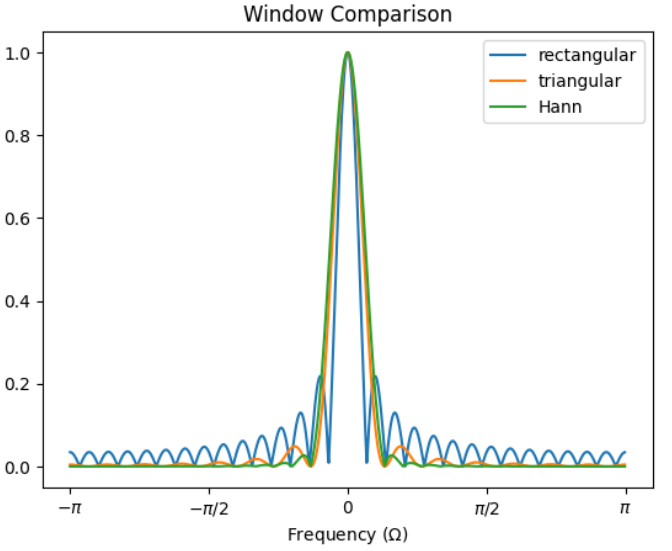


Compare

Superpose the plots of $W_r(\Omega)$, $W_t(\Omega)$, and $W_h(\Omega)$.
What are the important differences?

Compare

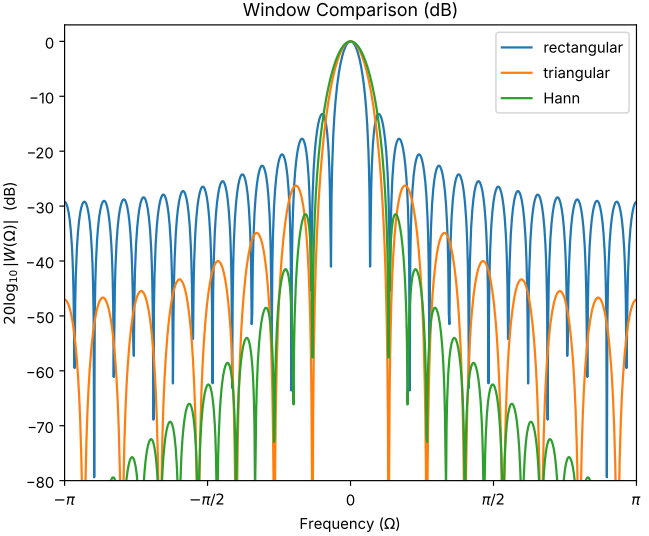
Superpose the plots of $W_r(\Omega)$, $W_t(\Omega)$, and $W_h(\Omega)$.
What are the important differences?



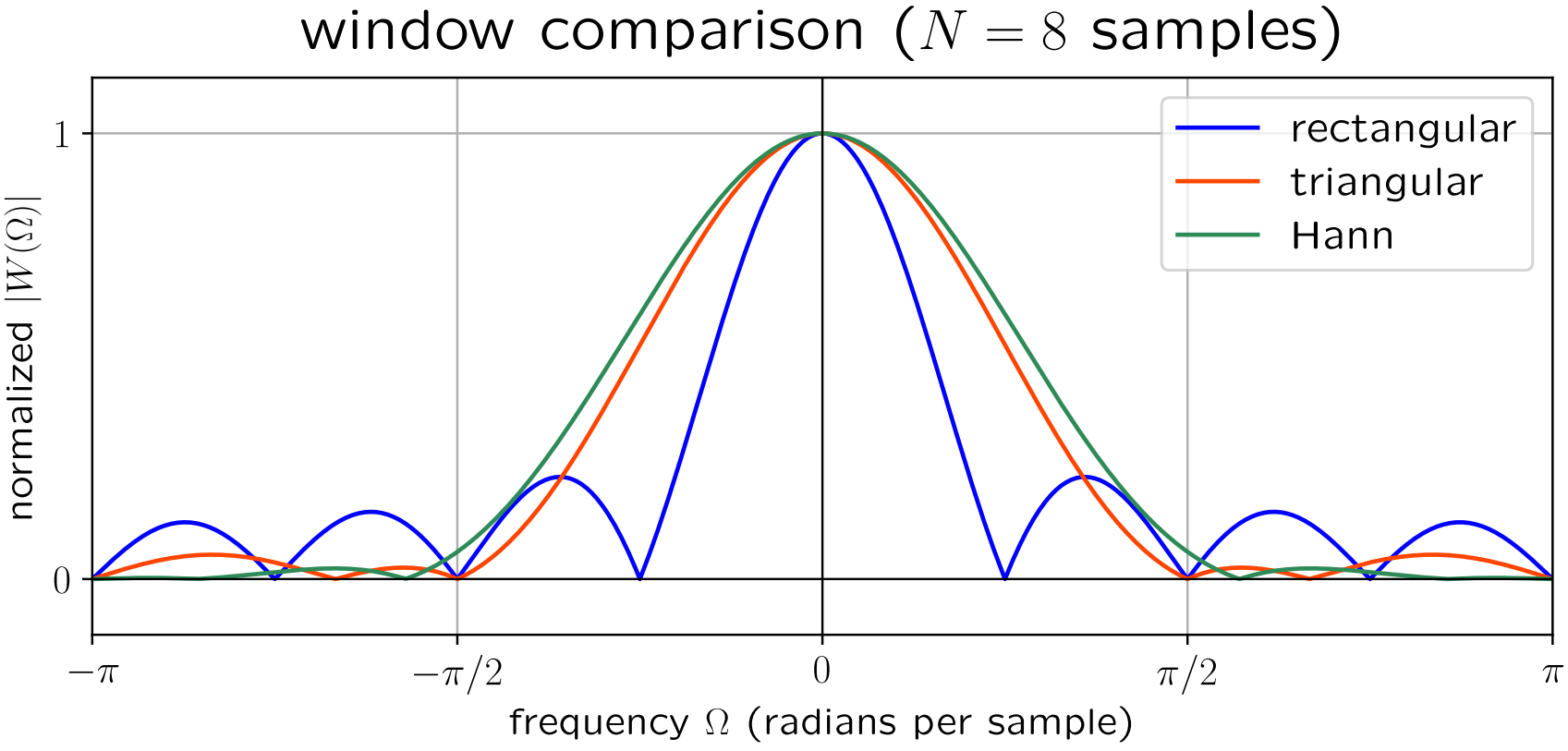
Compare

Same plots on a logarithmic scale: $20 \log_{10} |W(\Omega)|$ in dB.

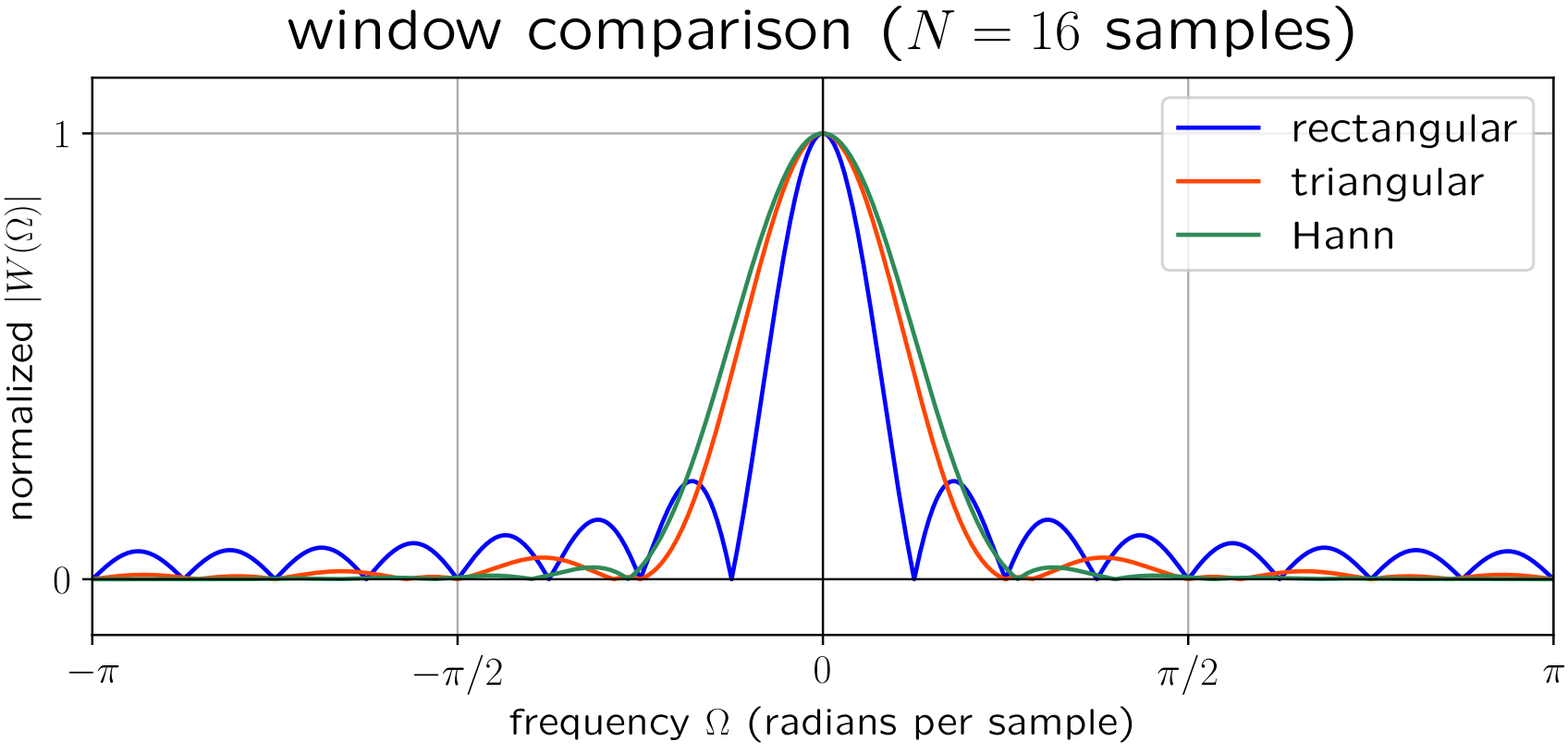
Compare the sidelobe levels and how quickly they fall off.



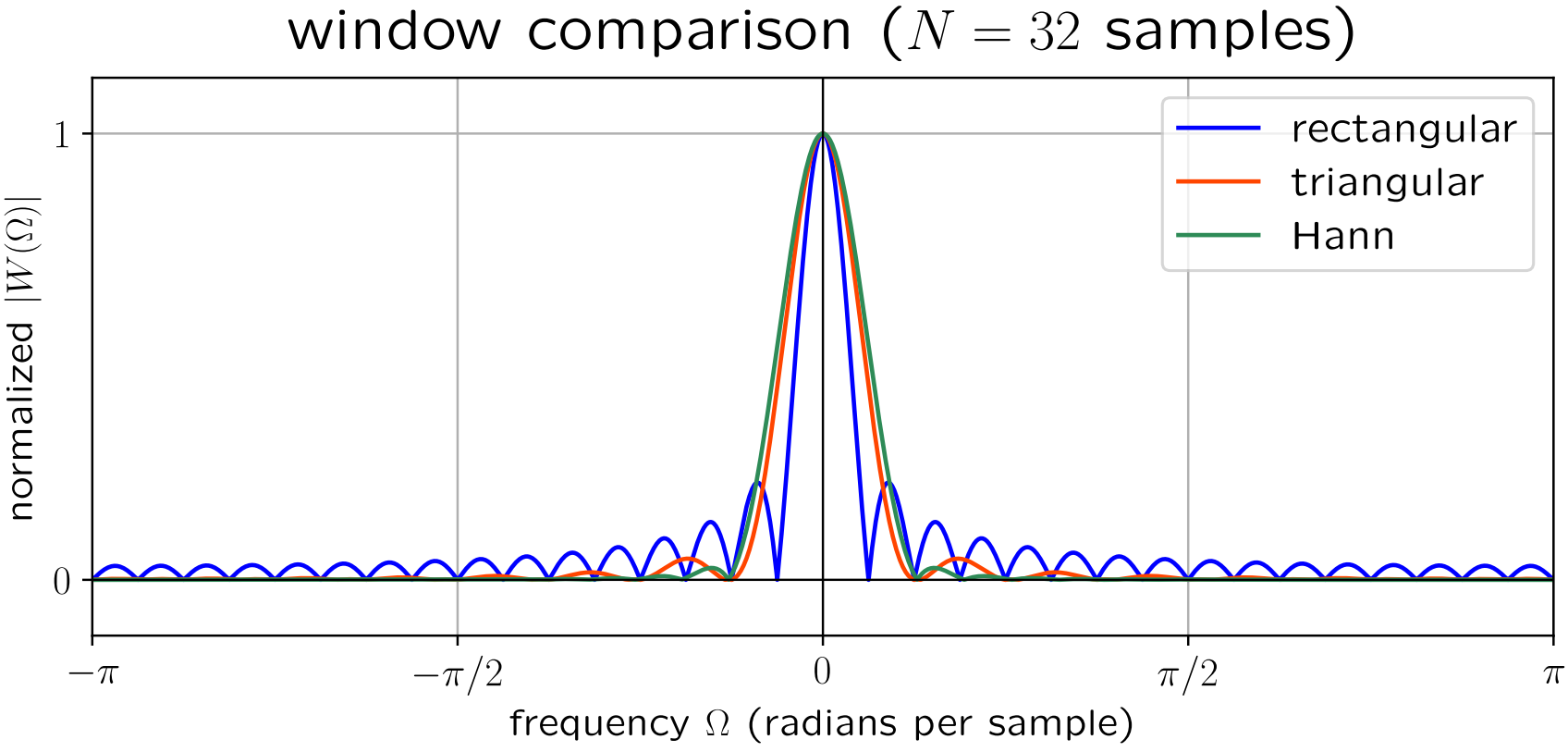
Windows


window comparison ($N = 8$ samples)
rectangular
triangular
Hann
normalized $|W(\Omega)|$
1
0
$-\pi$
$-\pi/2$
0
$\pi/2$
π
frequency Ω (radians per sample)

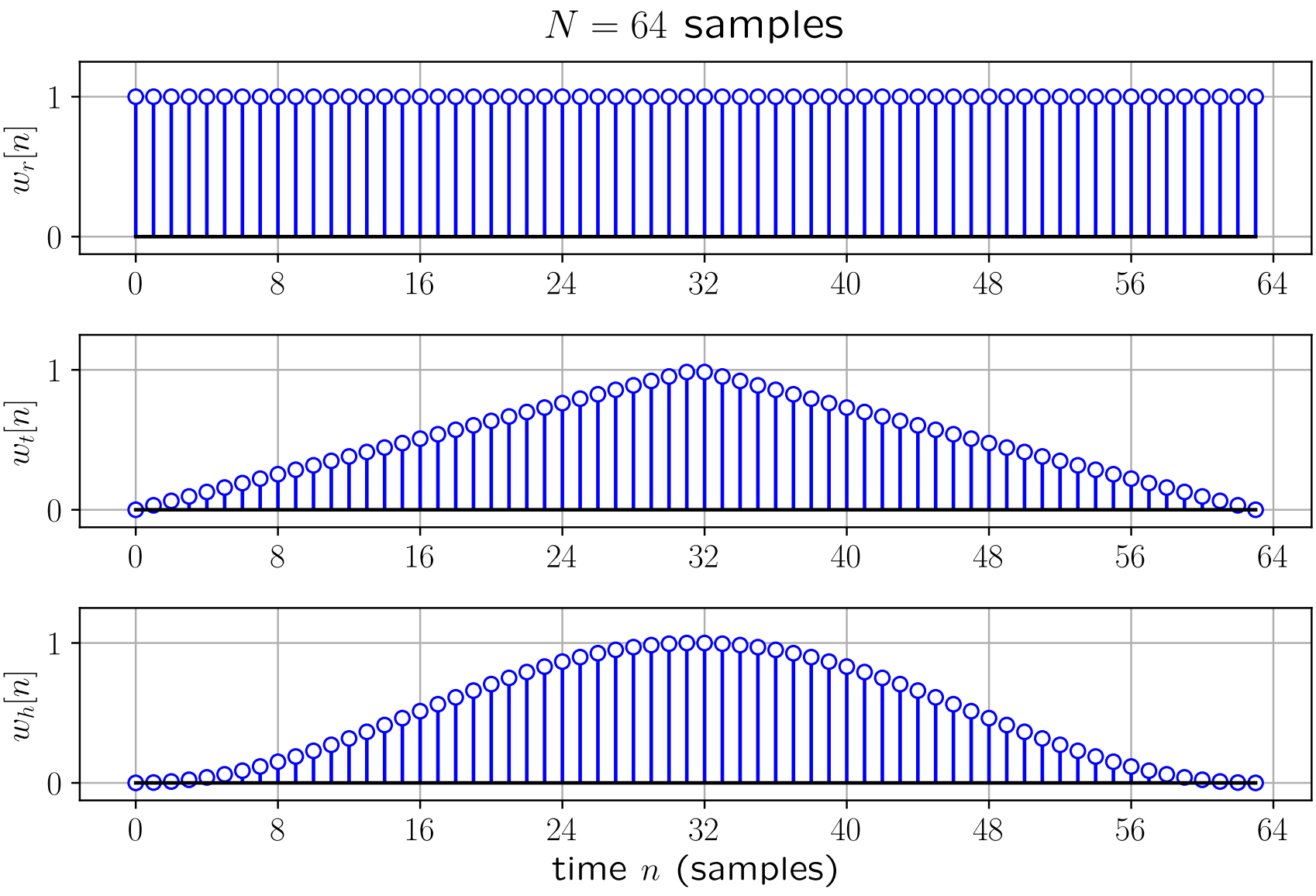
Windows


window comparison ($N = 16$ samples)
rectangular
triangular
Hann
normalized $|W(\Omega)|$
1
0
$-\pi$
$-\pi/2$
0
$\pi/2$
π
frequency Ω (radians per sample)

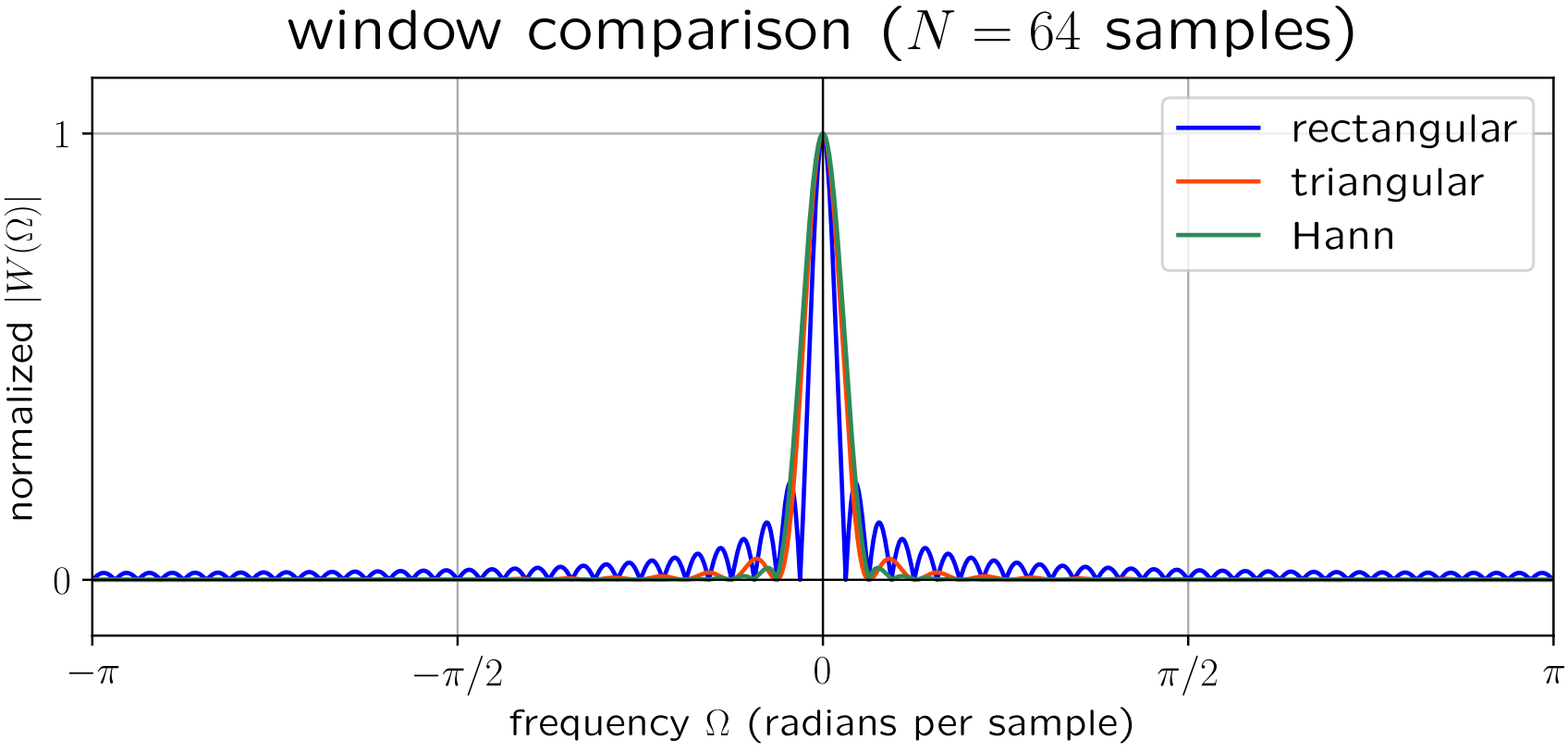
Windows


window comparison ($N = 32$ samples)
rectangular
triangular
Hann
normalized $|W(\Omega)|$
1
0
$-\pi$
$-\pi/2$
0
$\pi/2$
π
frequency Ω (radians per sample)

Windows



Windows


window comparison ($N = 64$ samples)
rectangular
triangular
Hann
normalized $|W(\Omega)|$
1
0
$-\pi$
$-\pi/2$
0
$\pi/2$
π
frequency Ω (radians per sample)

Short-Time Fourier Transform

STFT: Analyze a signal "block by block." An STFT is a sequence of windowed Fourier transforms.

$$X[k,m] = \frac{1}{N} \sum_{n=0}^{N-1} x[n+ms]w[n]e^{-jk\frac{2\pi}{N}n}$$

Spectrogram: $|X[k,m]|^2$

Parameters:

- type of window: $w[n]$
- DFT size: N
- step size: s

Short-Time Fourier Transform

Parameters:

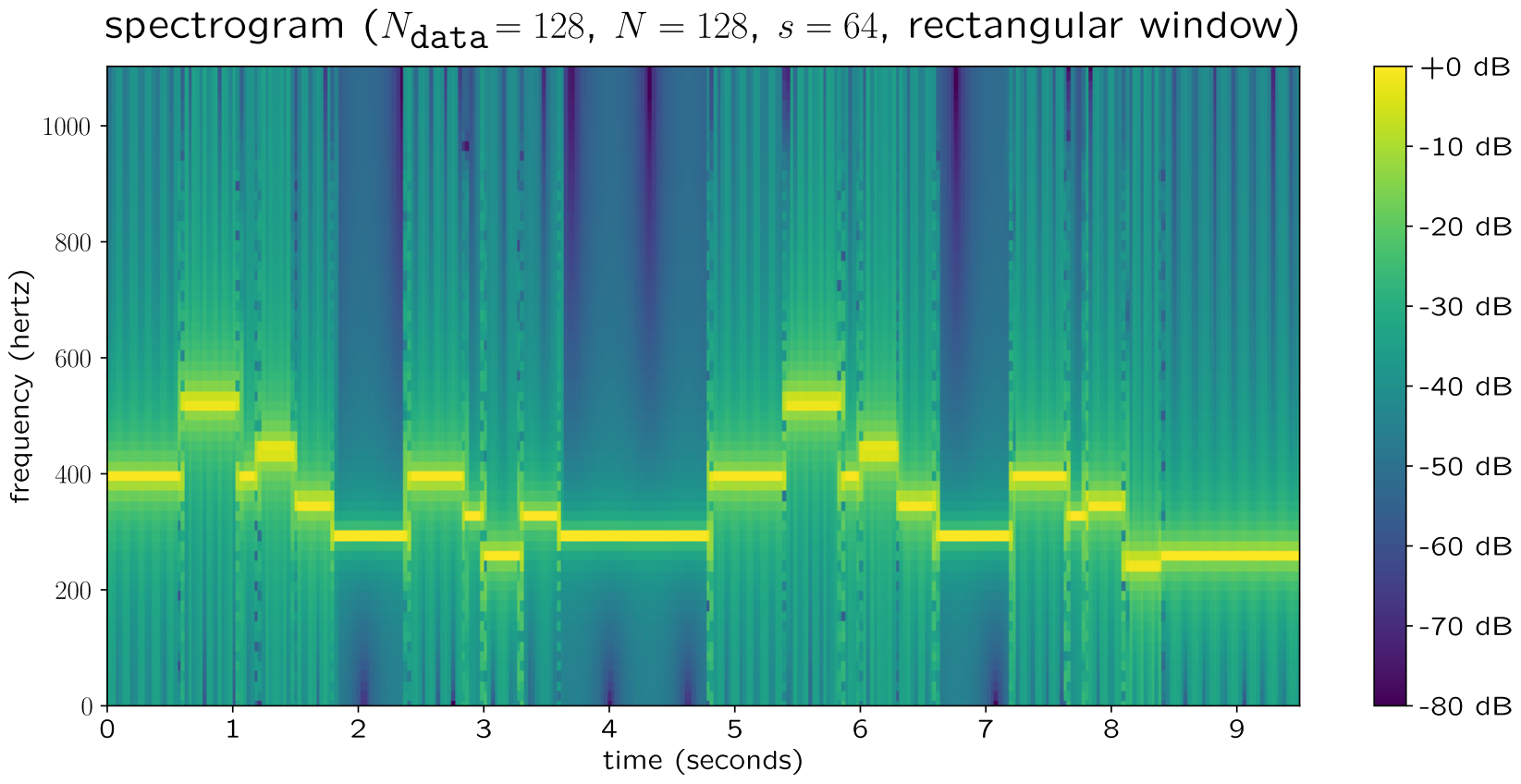
- type of window: $w[n]$
- DFT size: N
- step size: s

Short-Time Fourier Transform

Parameters:

- type of window: $w[n]$
- DFT size: N
- step size: s

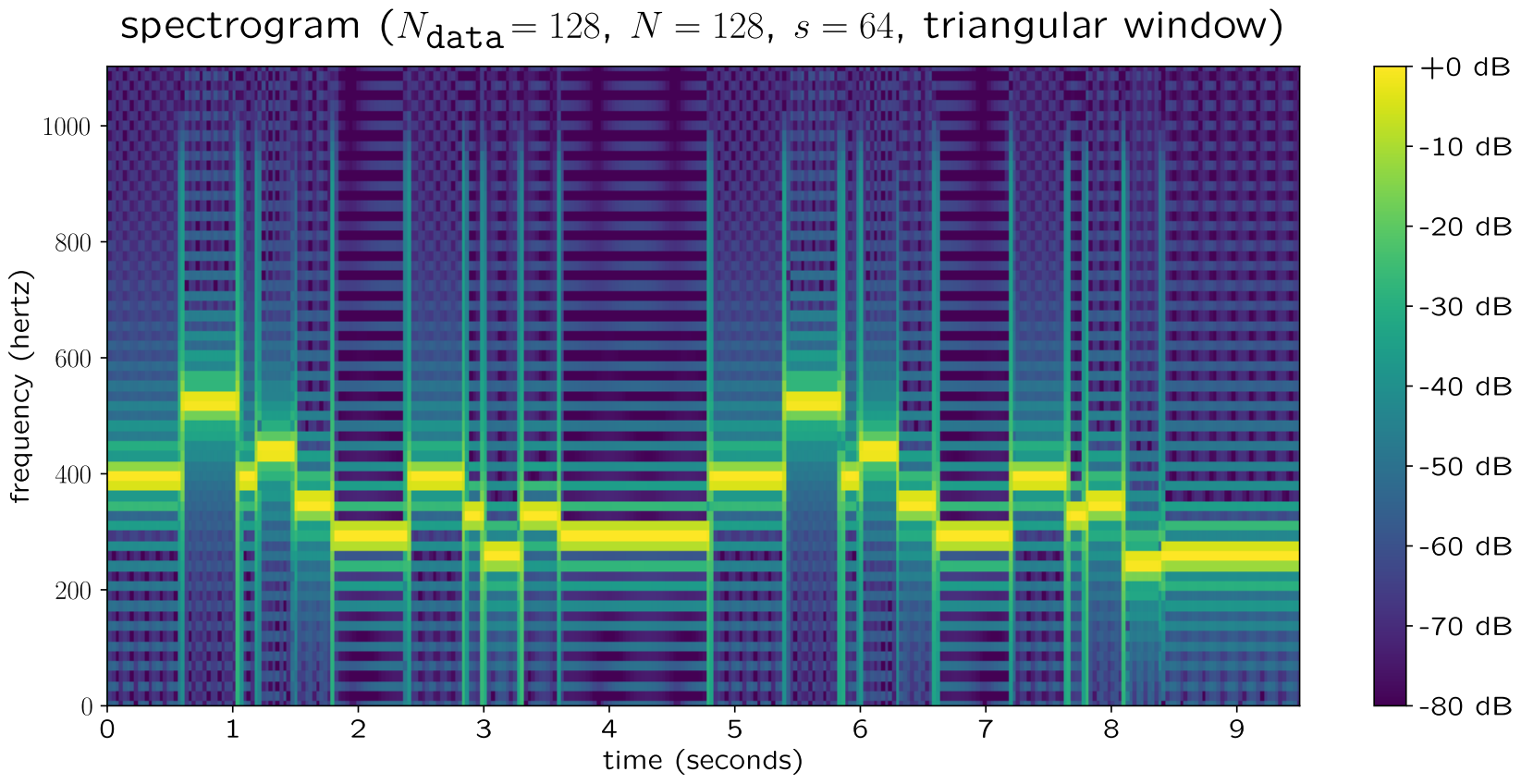
STFT: Windows



Decibels: $x_{\mathsf{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

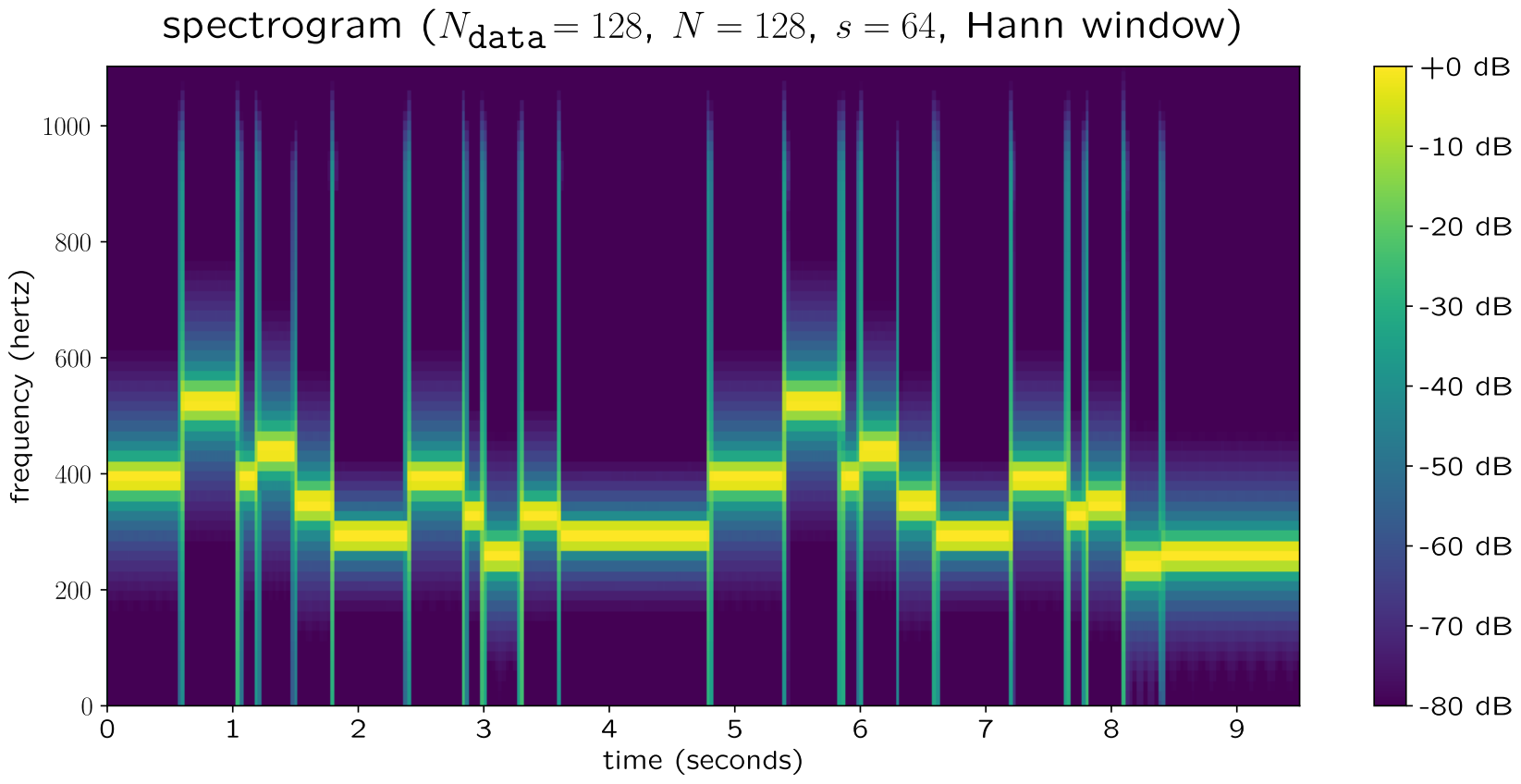
STFT: Windows



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

STFT: Windows



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

Short-Time Fourier Transform

Parameters:

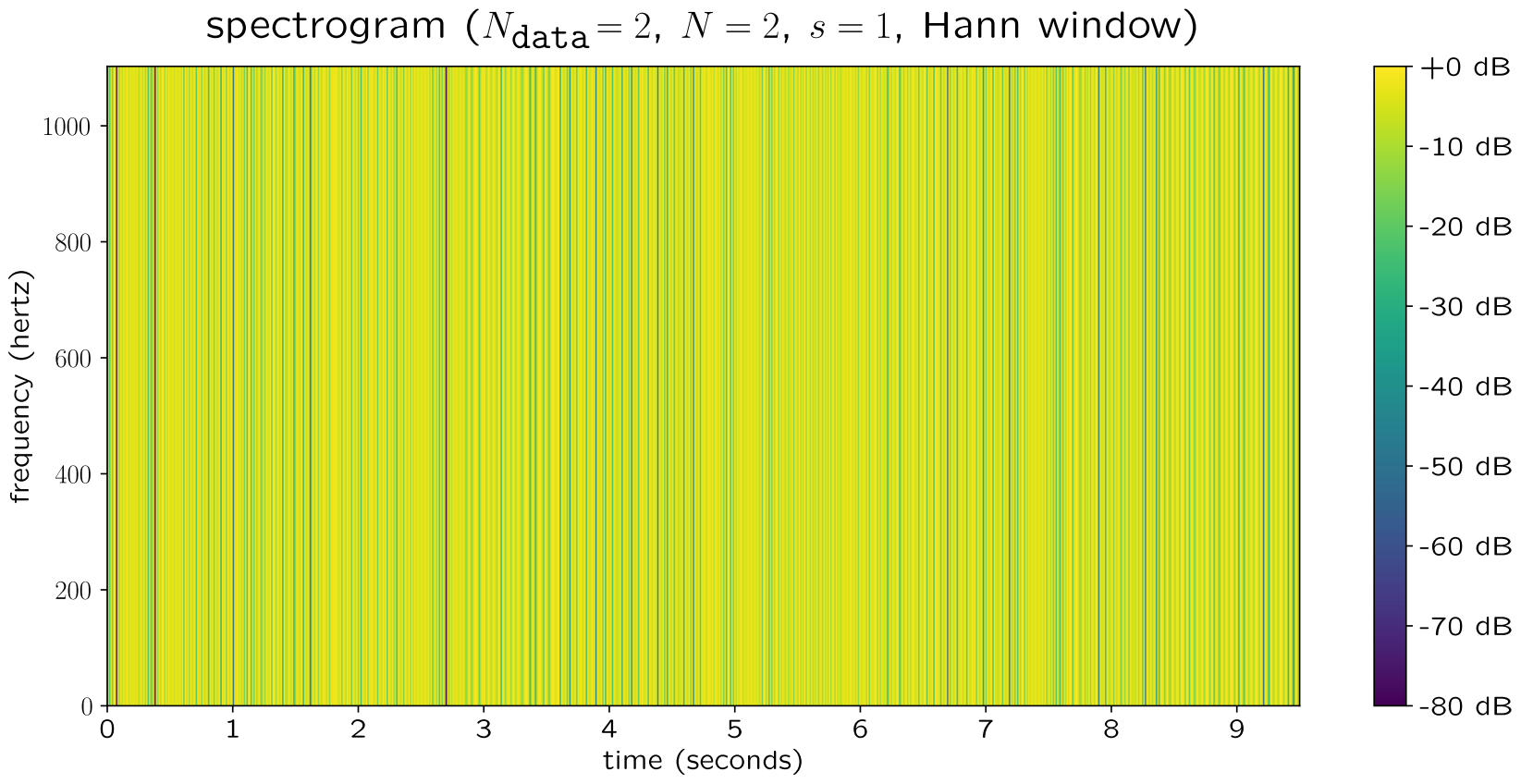
- type of window: $w[n]$
- DFT size: N
- step size: s

Short-Time Fourier Transform

Parameters:

- type of window: $w[n]$
- DFT size: N
- step size: s

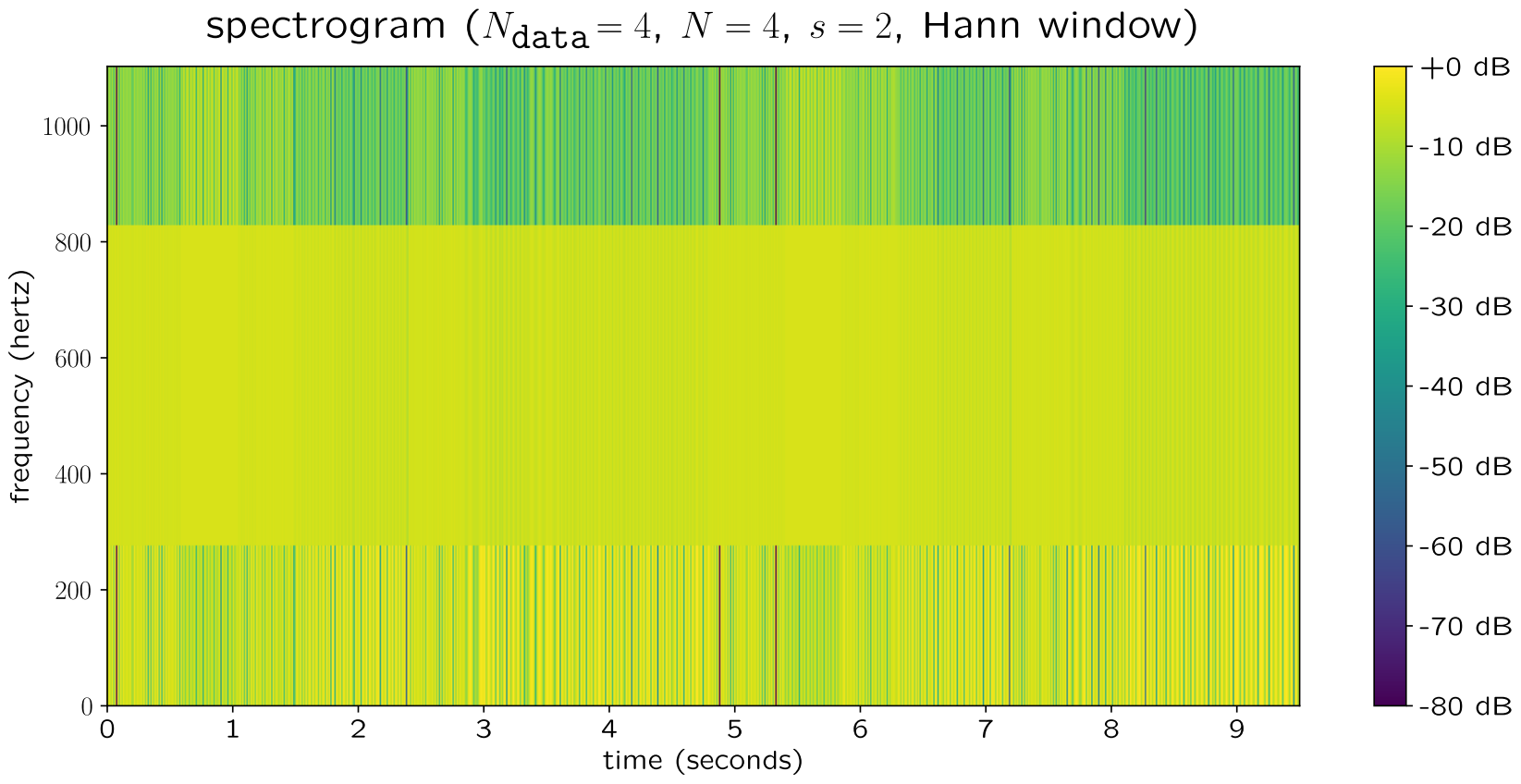
STFT: DFT Size



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

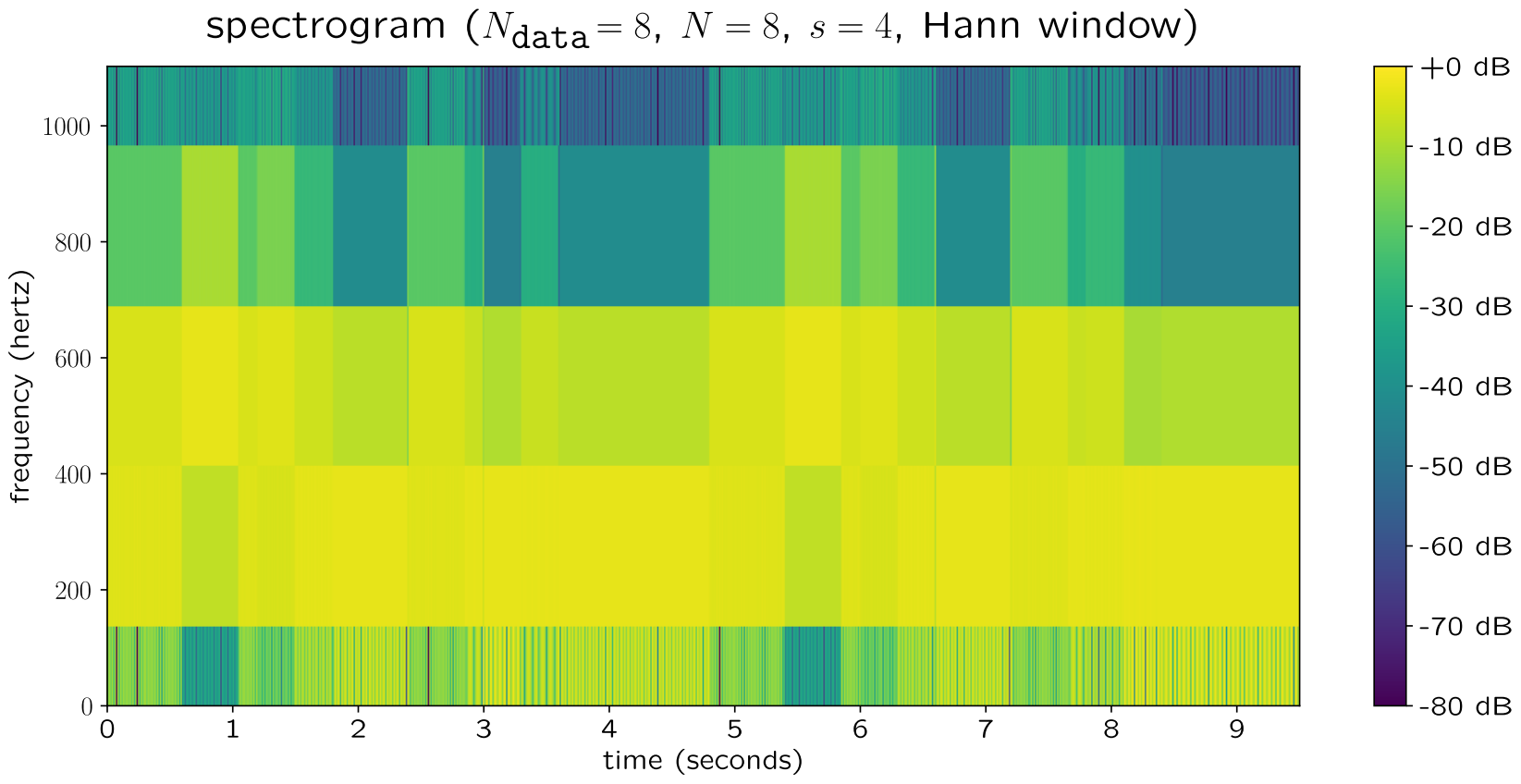
STFT: DFT Size



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

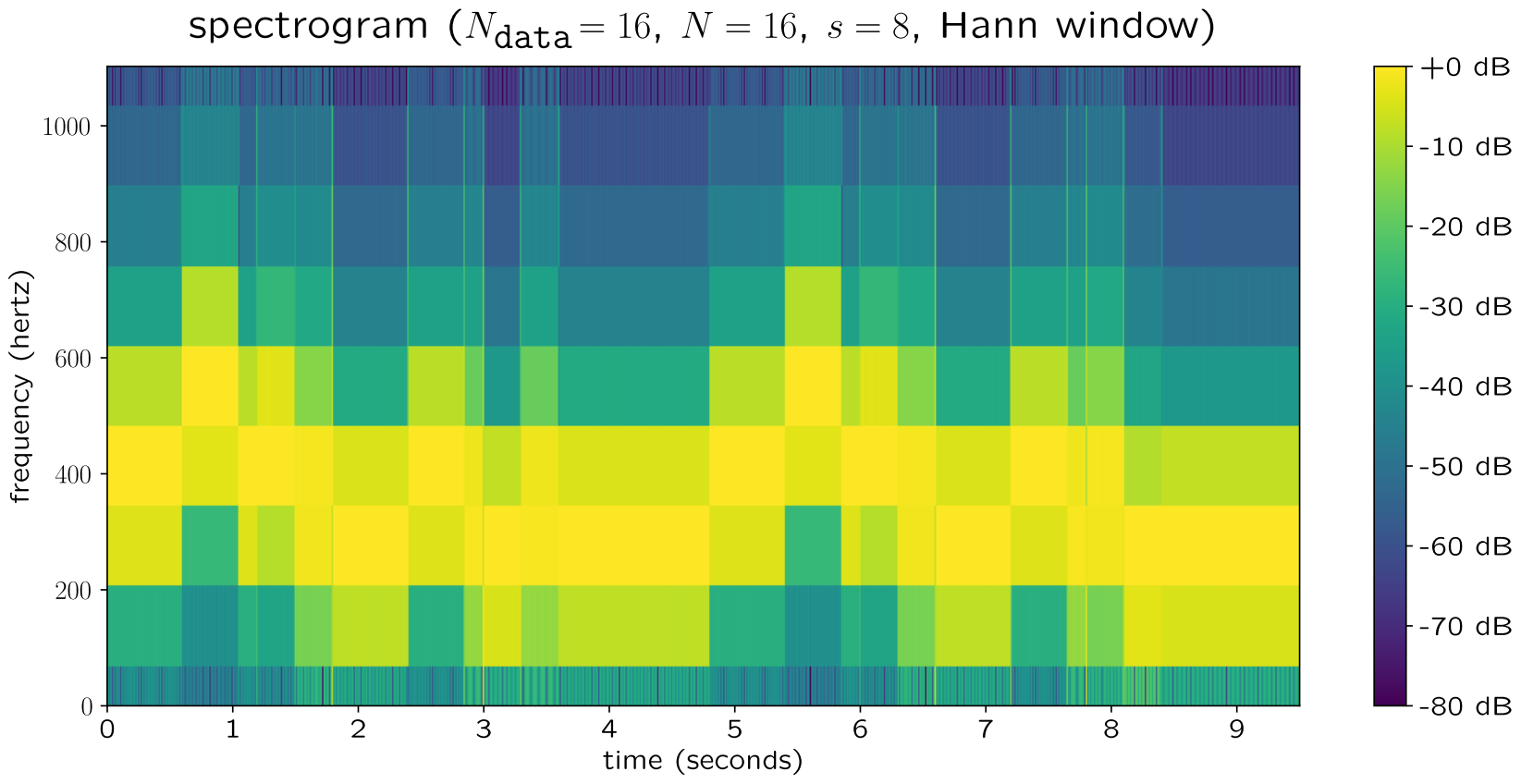
STFT: DFT Size



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

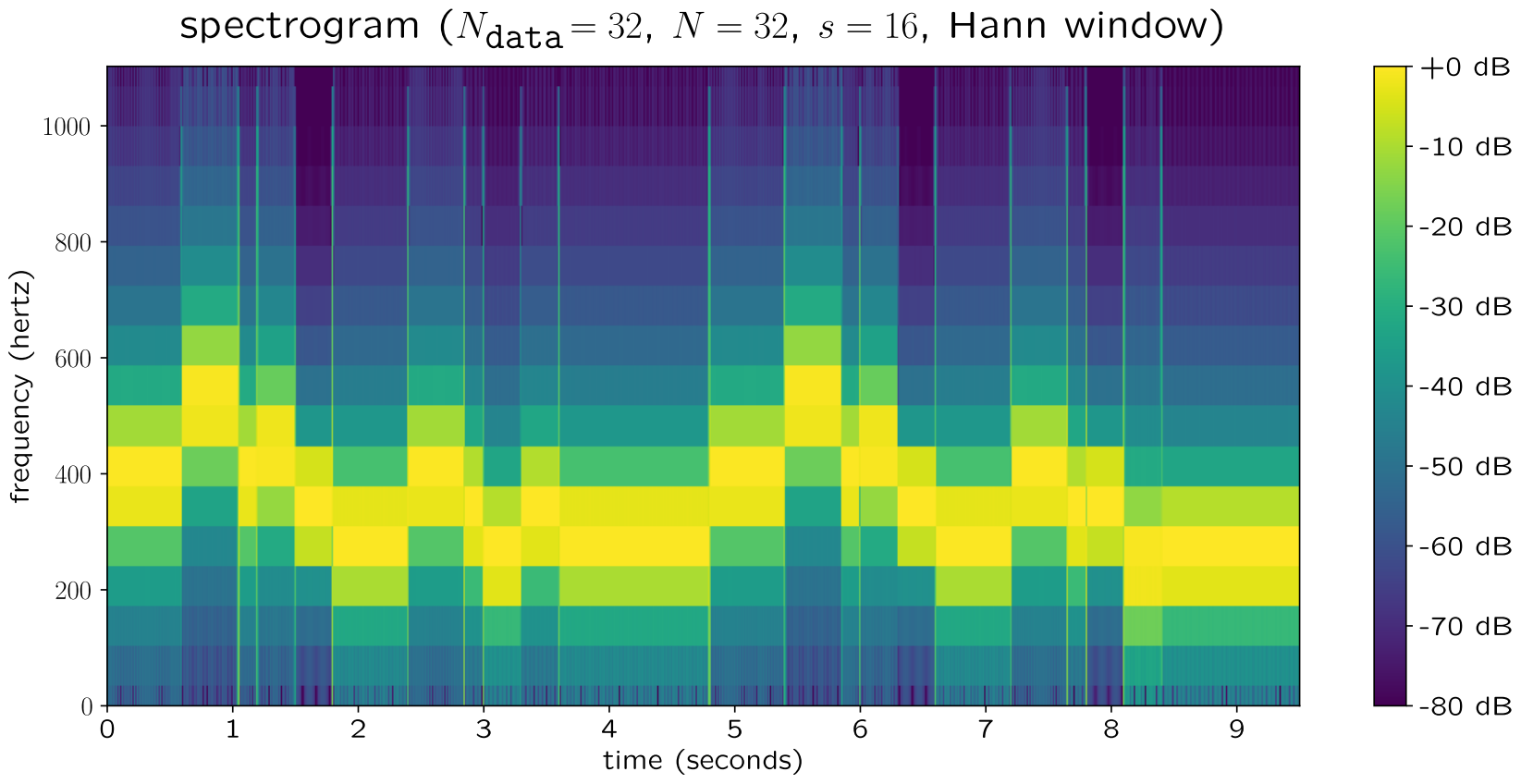
STFT: DFT Size



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

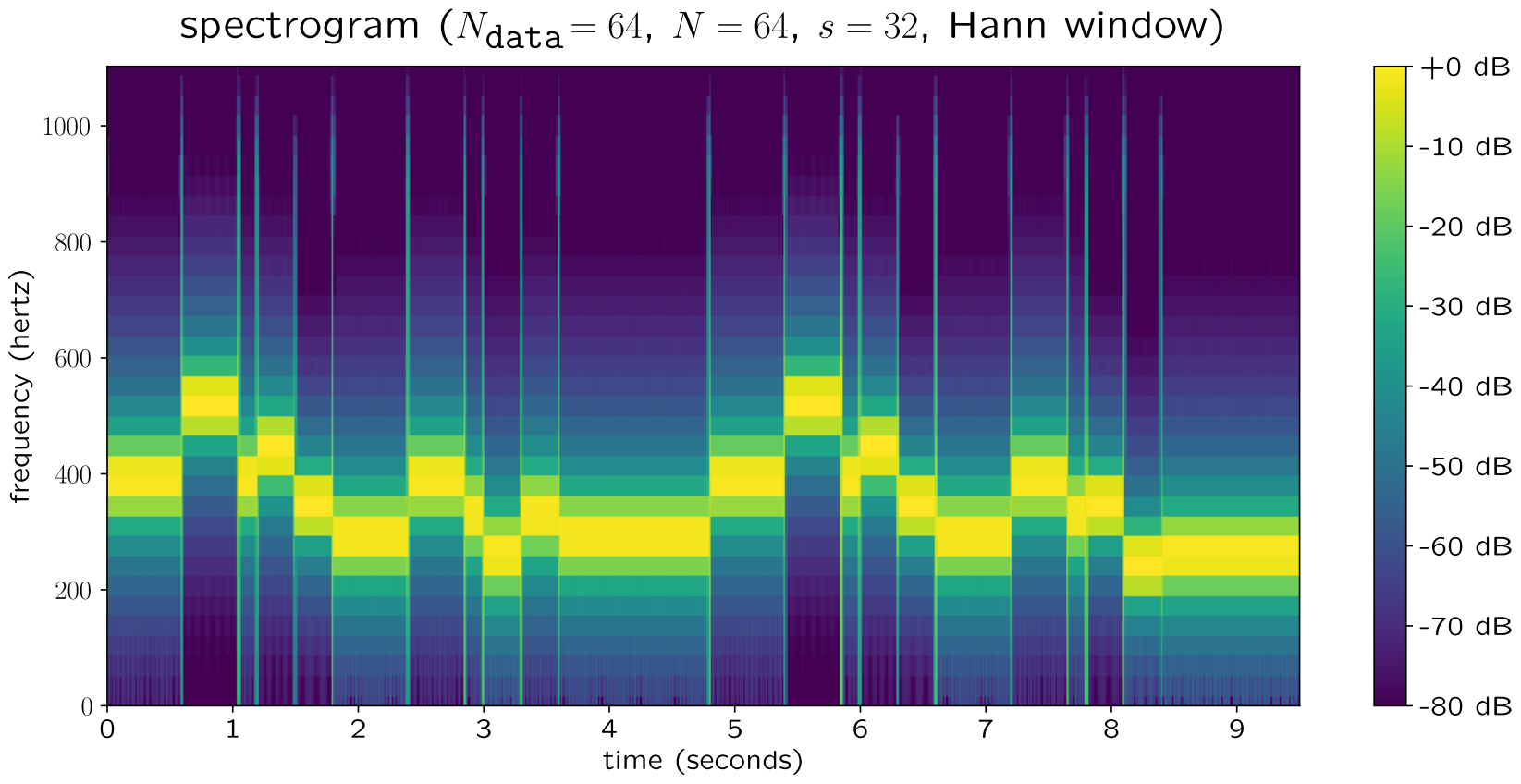
STFT: DFT Size



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

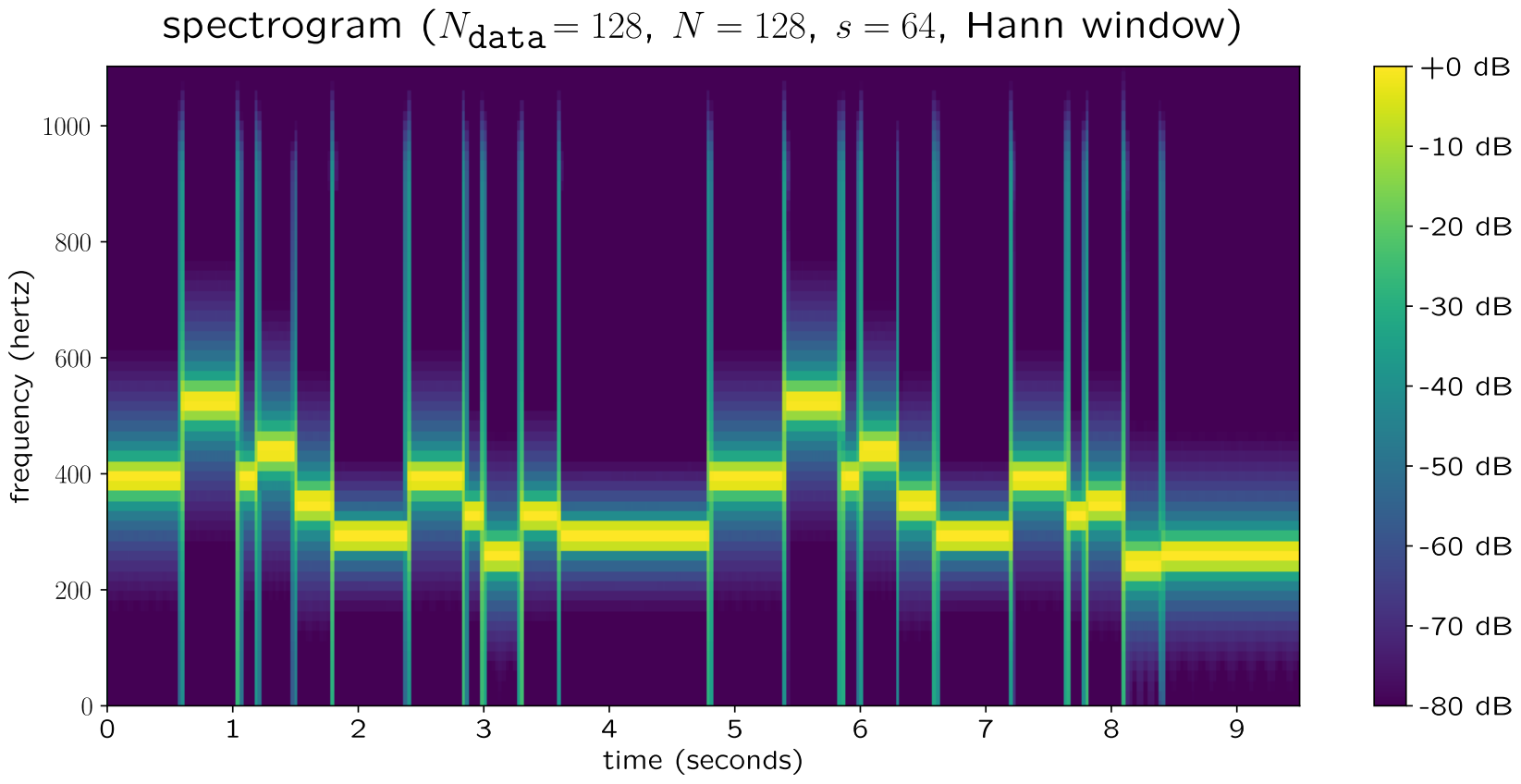
STFT: DFT Size



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

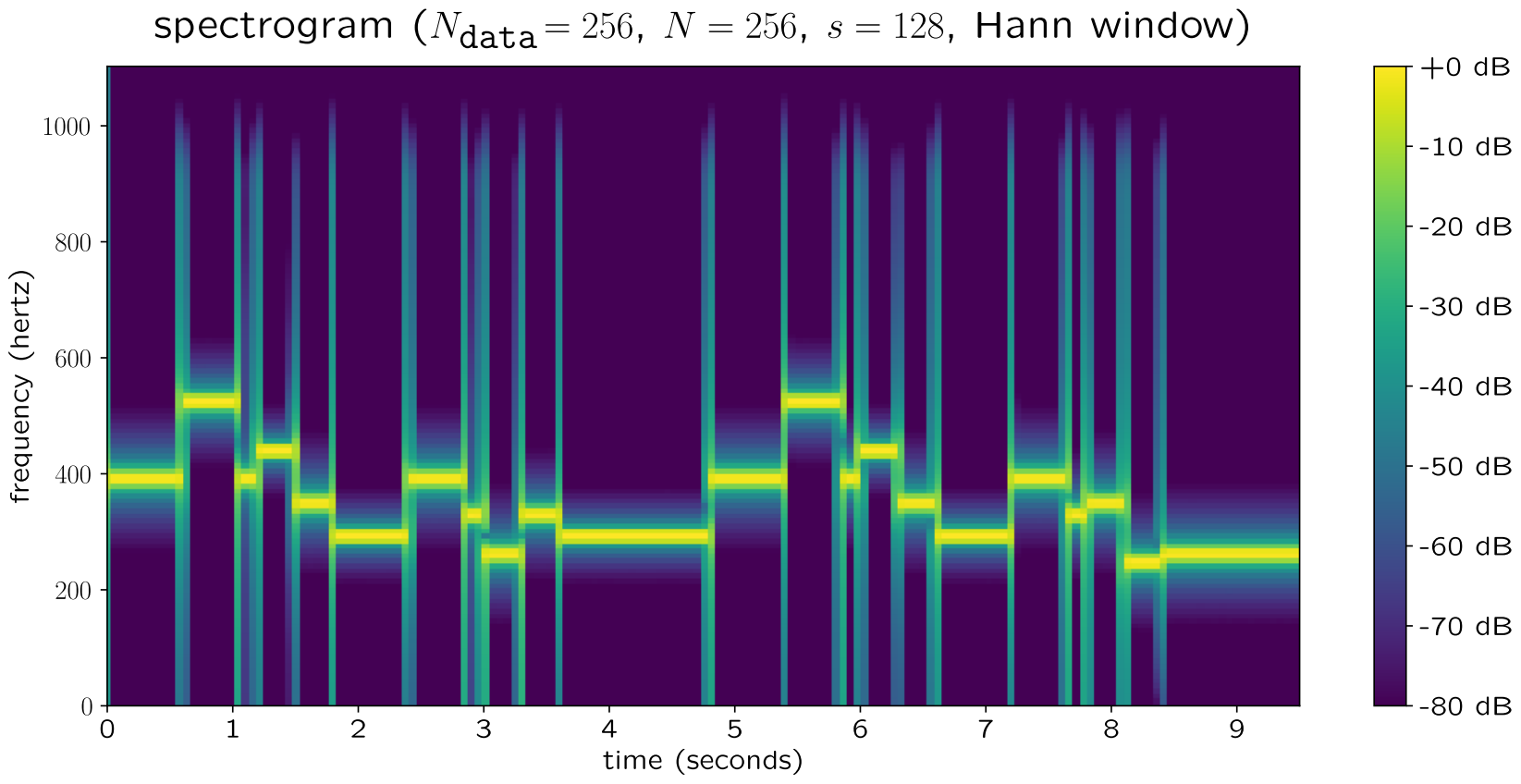
STFT: DFT Size



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

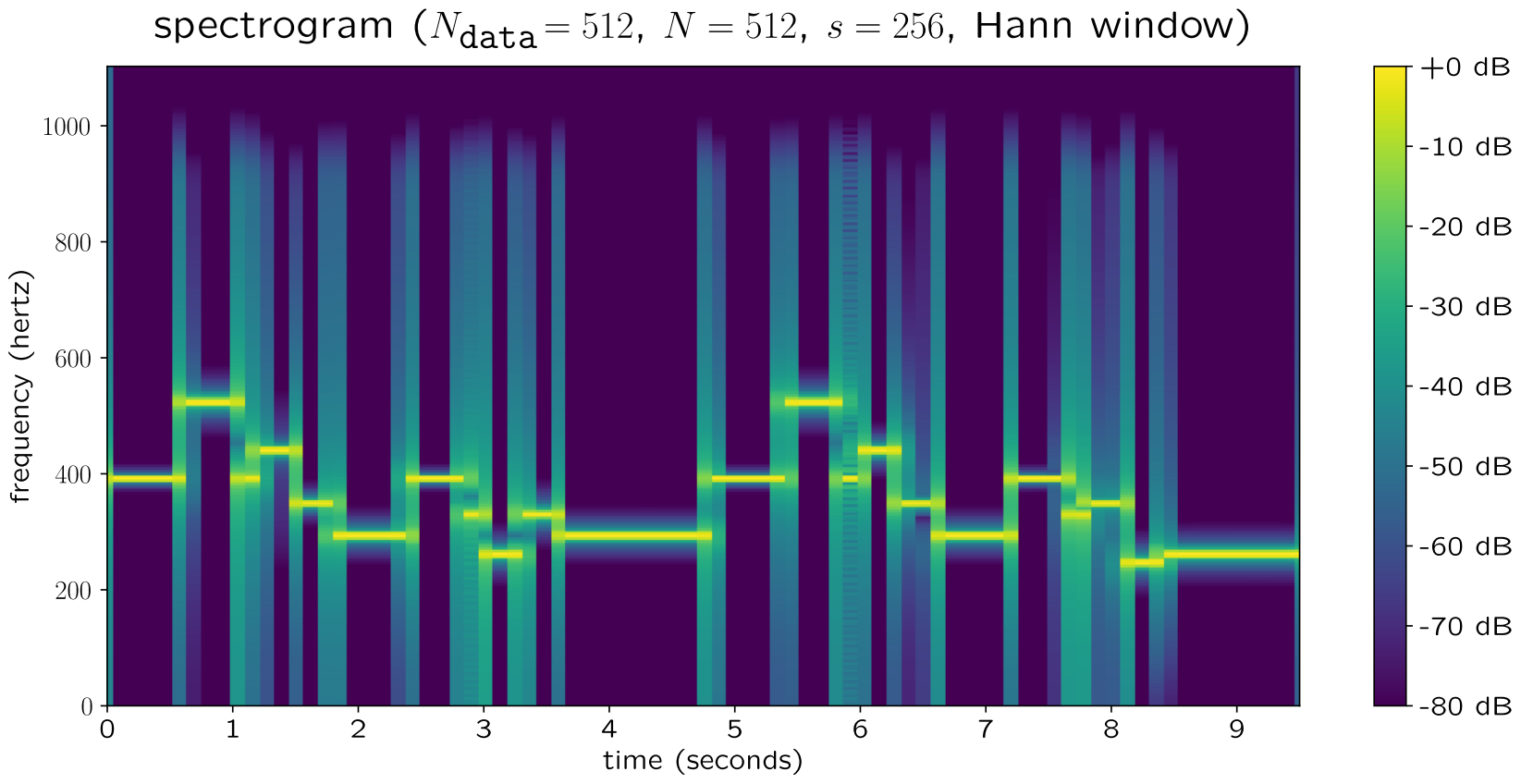
STFT: DFT Size



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

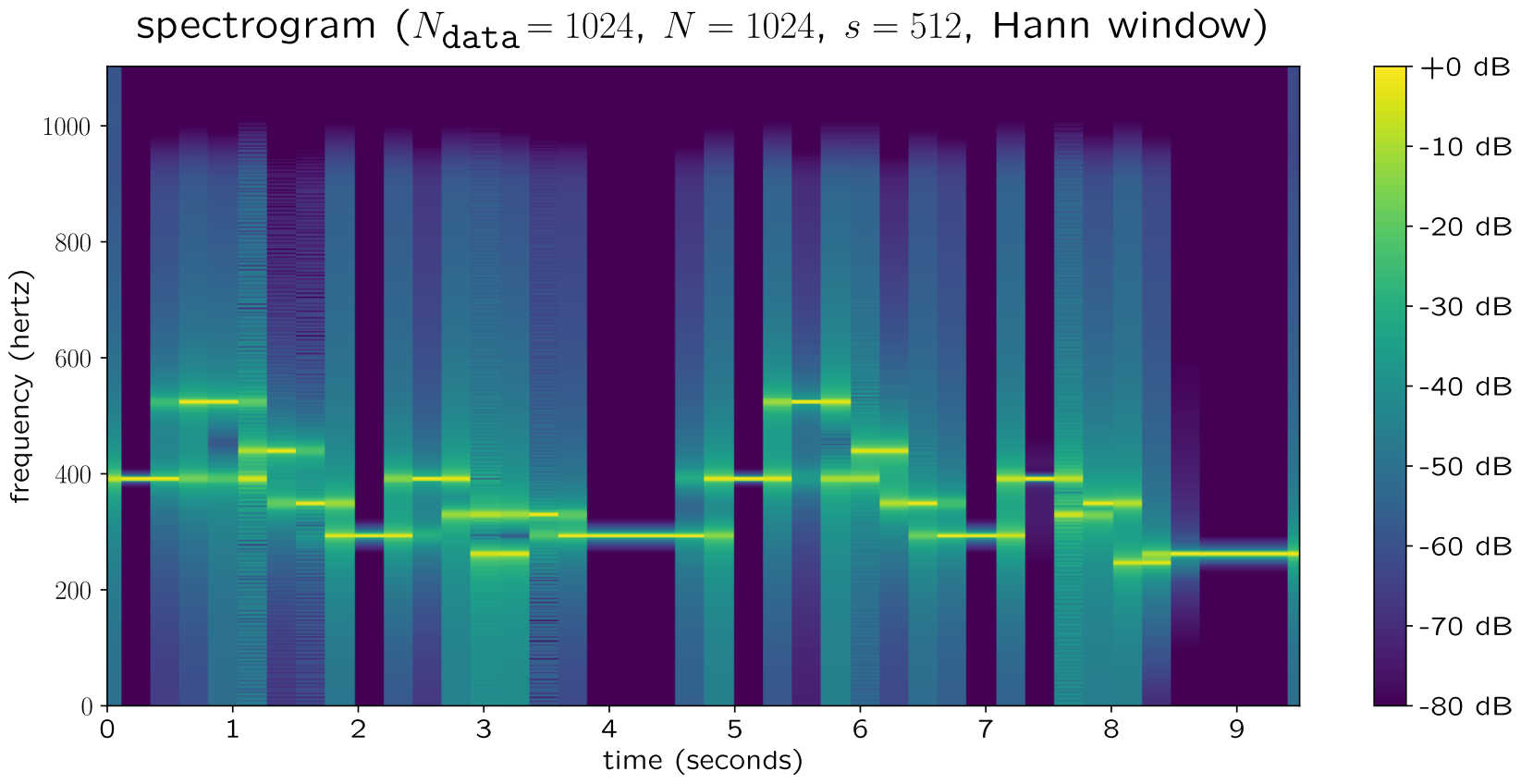
STFT: DFT Size



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

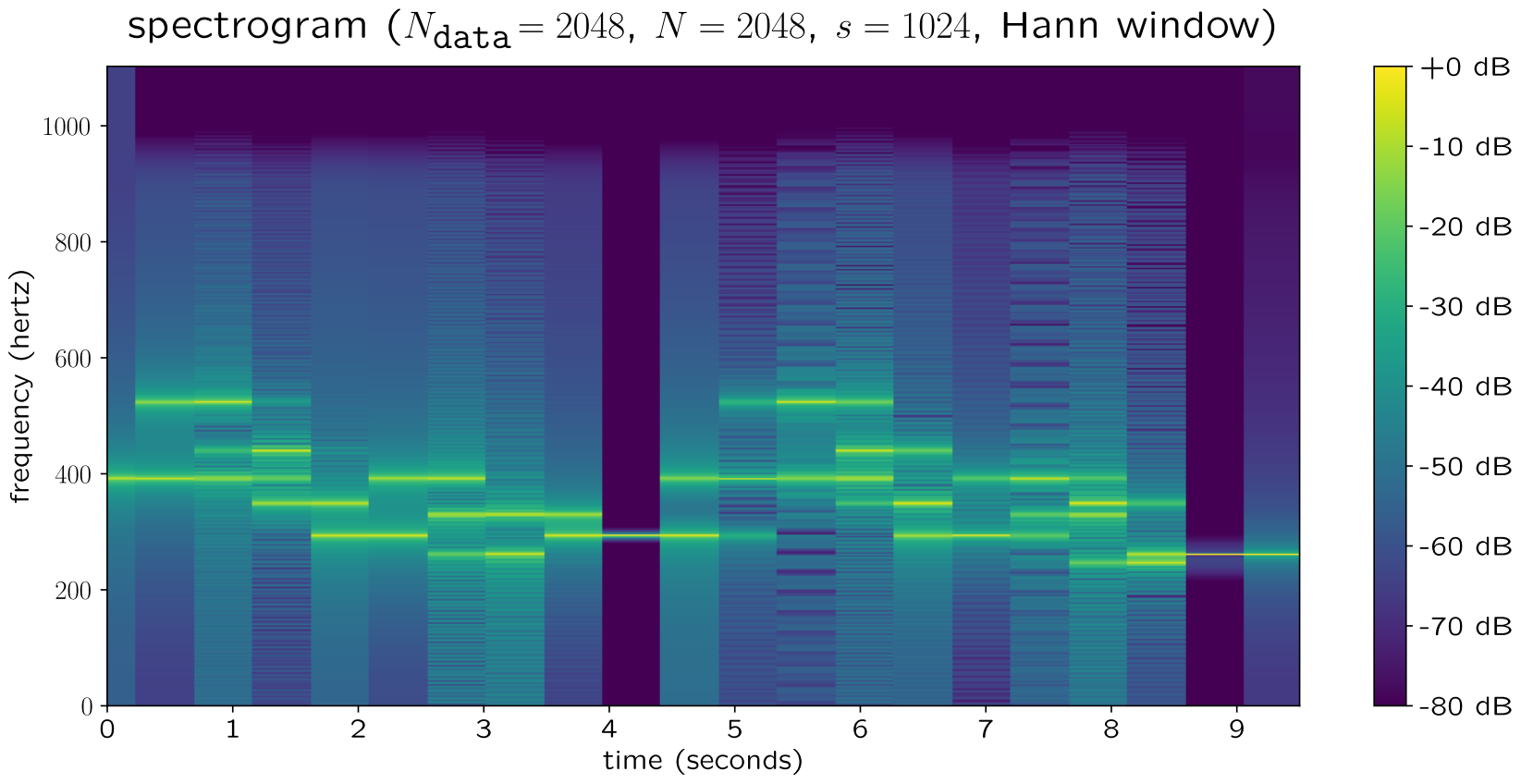
STFT: DFT Size



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

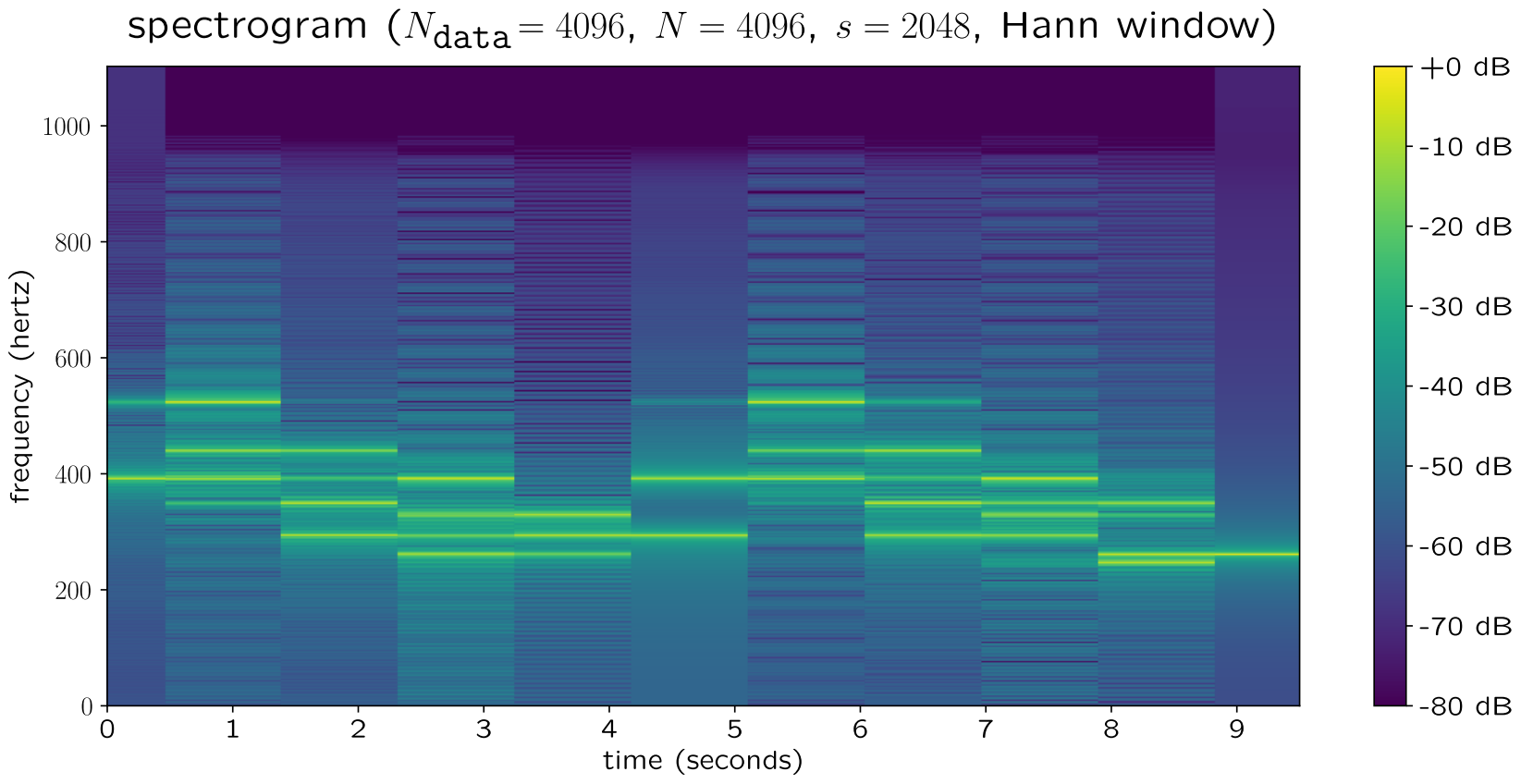
STFT: DFT Size



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

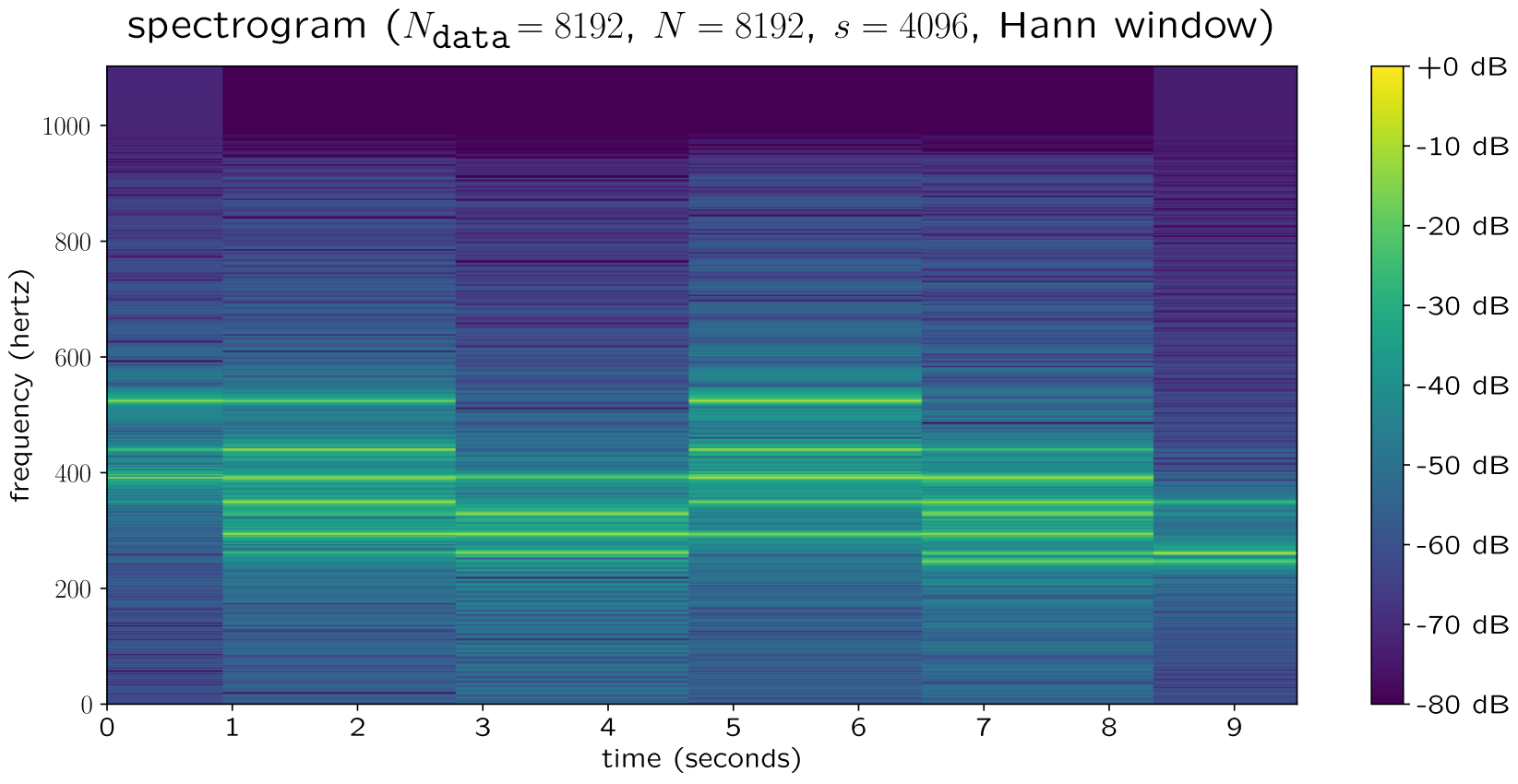
STFT: DFT Size



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

STFT: DFT Size



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

STFT: Zero-Padding

Remember:

- N_{data} is the length of the data.
- N is the DFT size.

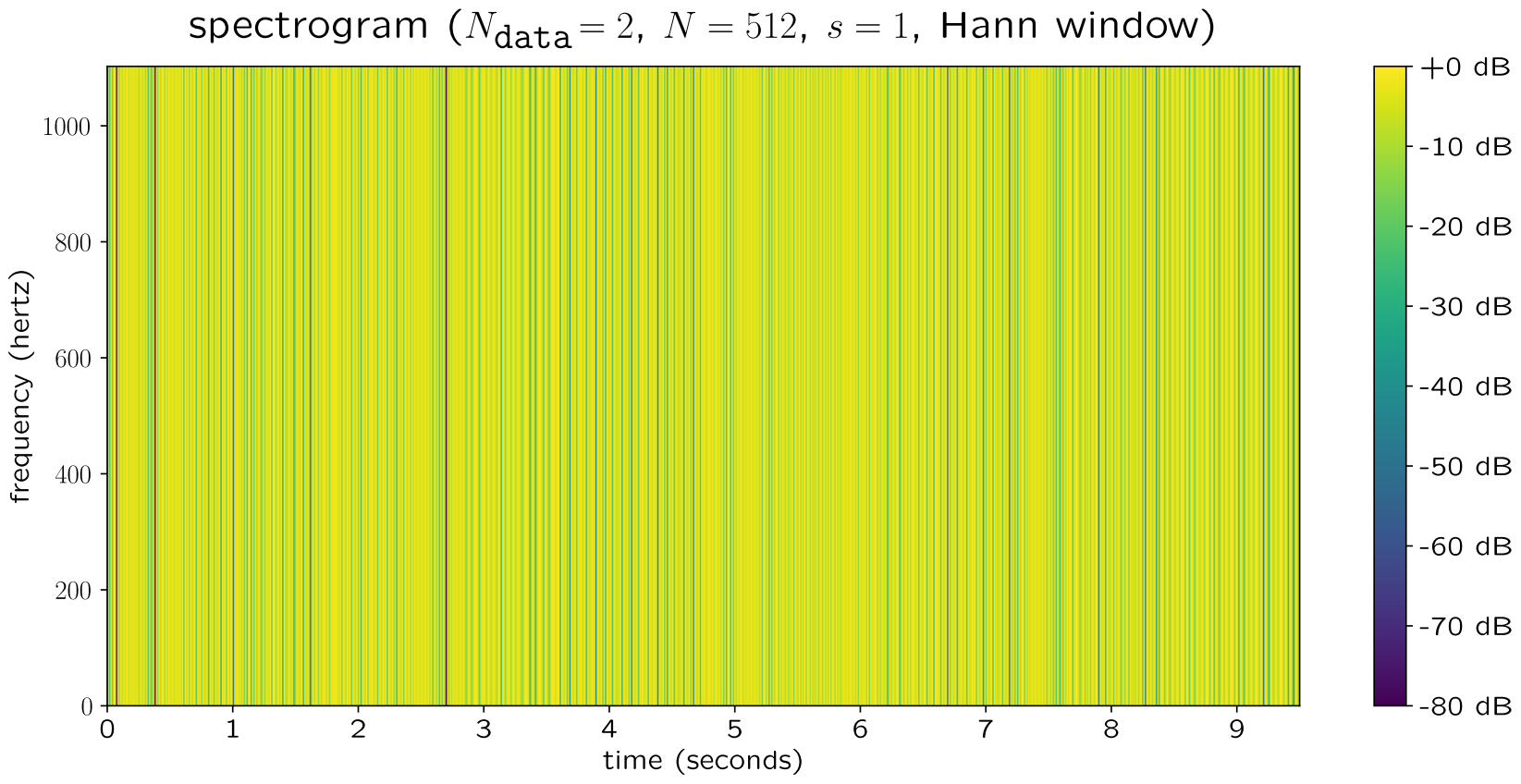
When $N > N_{\text{data}}$, we assume that we are **zero-padding**.

As N increases, we sample the underlying DTFT at more points — but this does not alter $W(\Omega)$, the DTFT of the window function. The width of $W(\Omega)$'s mainlobe ($\sim 1/N_{\text{data}}$) determines frequency resolution.

The more data we have, the larger N_{data} can be. As N_{data} increases, $W(\Omega)$ gets narrower — and the frequency resolution improves.

> You cannot improve frequency resolution by adding "zero information" — only by analyzing more data.

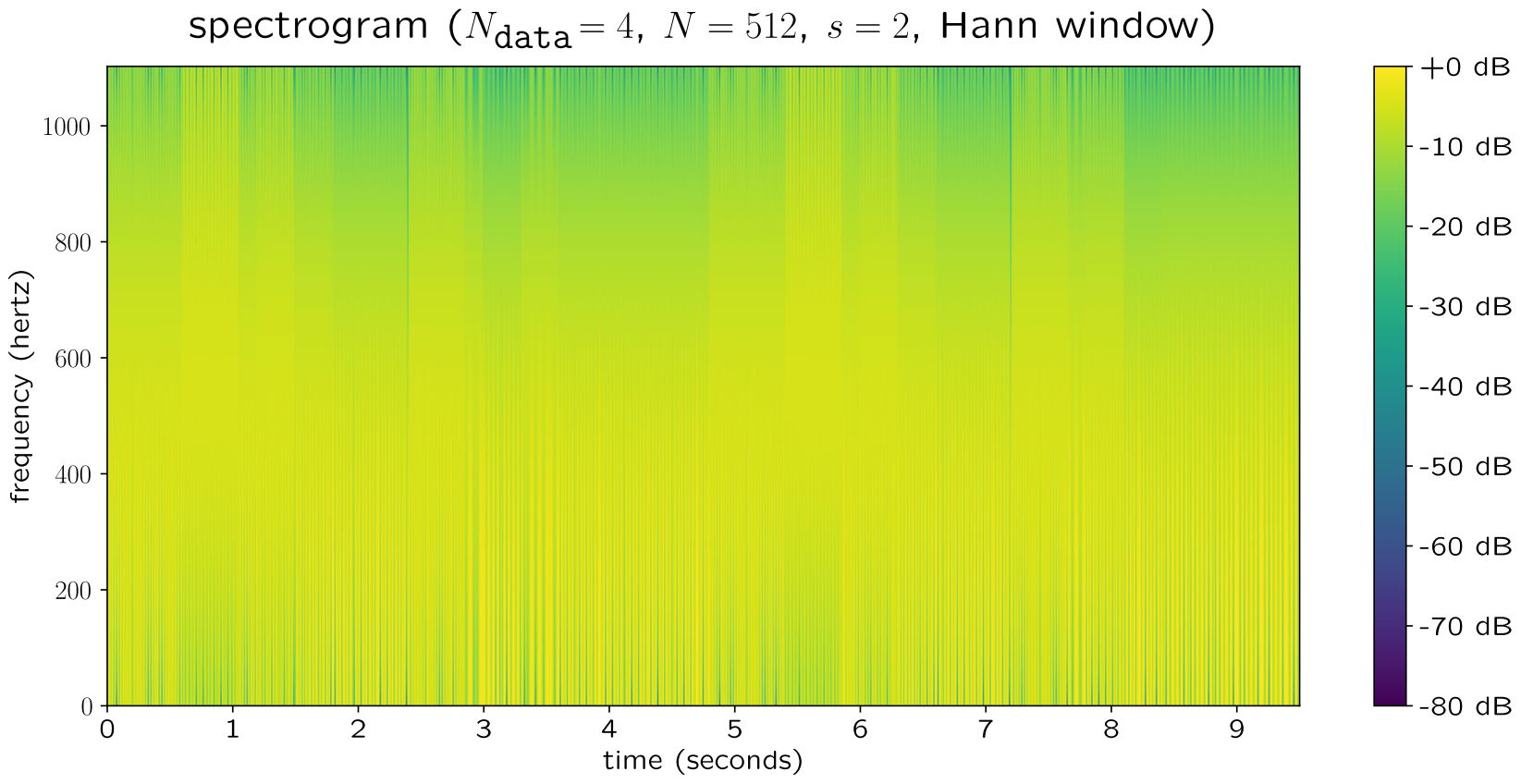
STFT: Zero-Padding



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

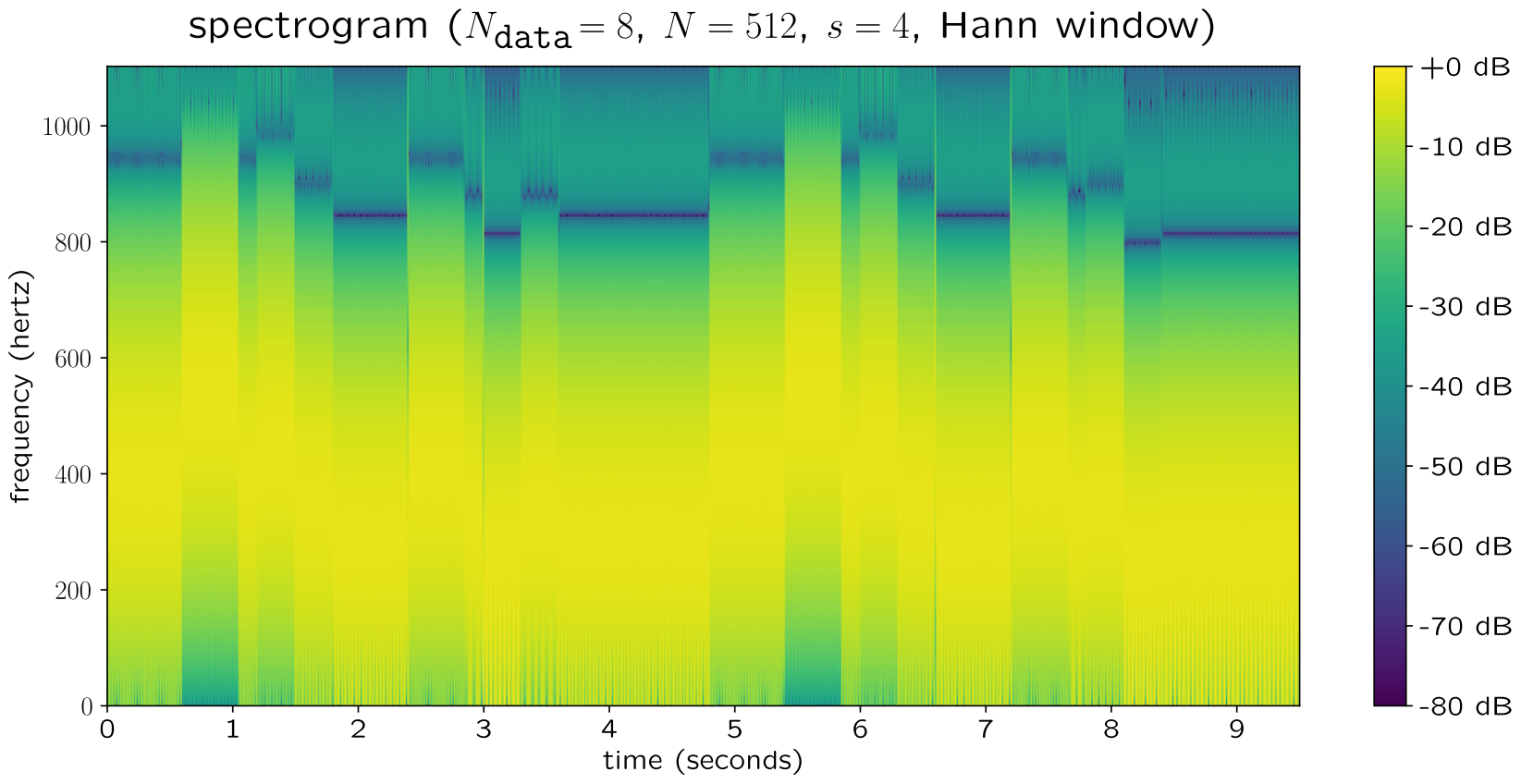
STFT: Zero-Padding



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

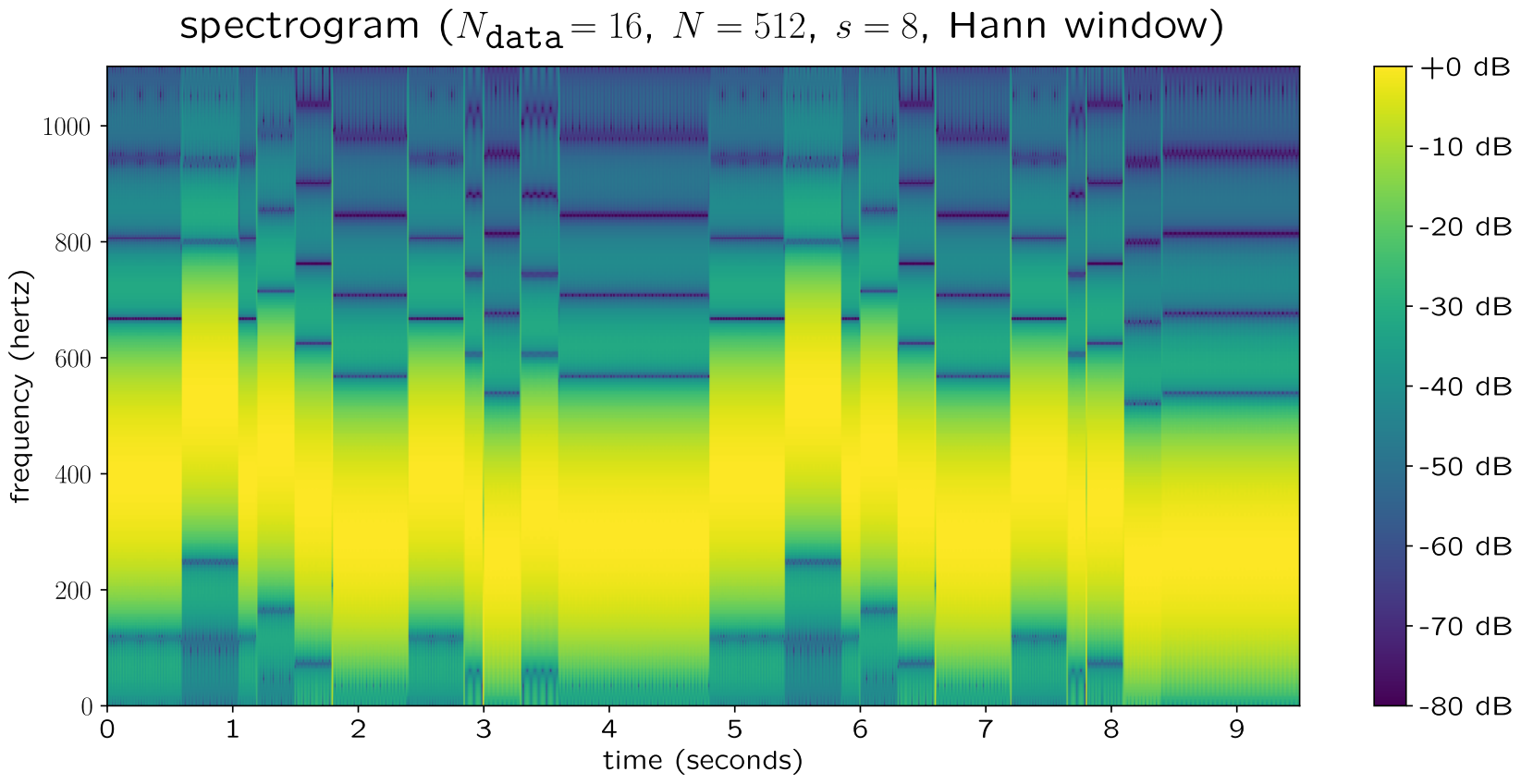
STFT: Zero-Padding



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

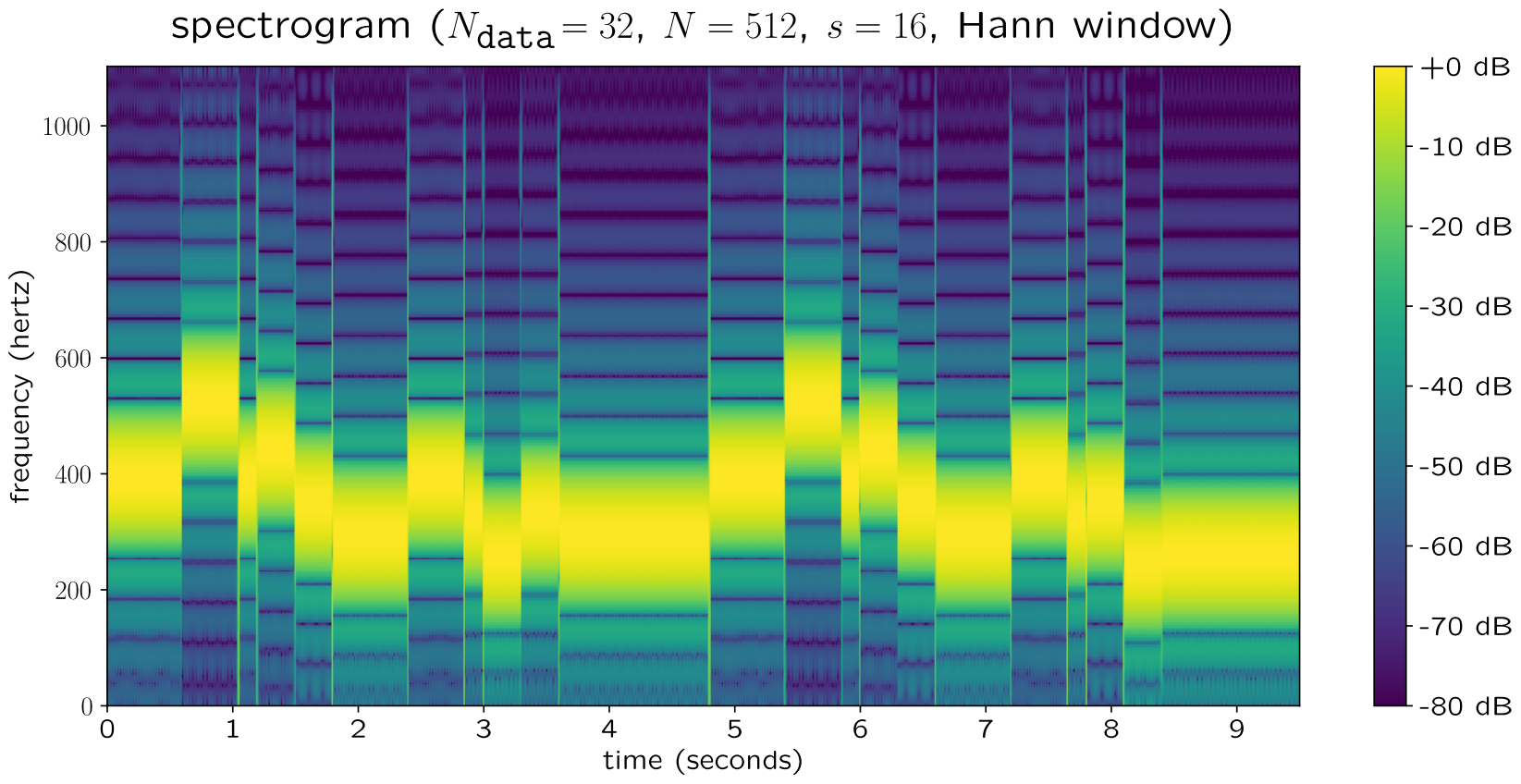
STFT: Zero-Padding



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

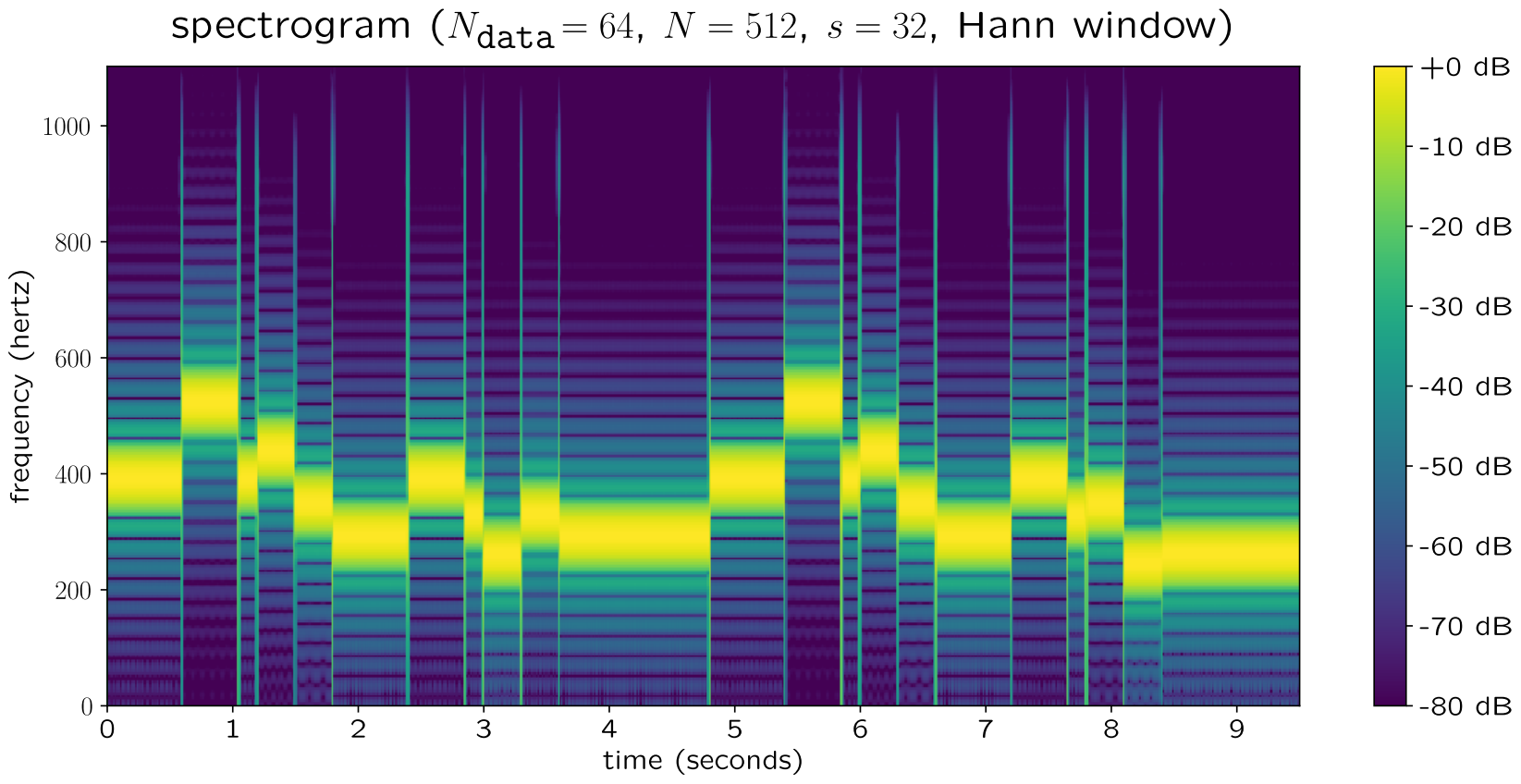
STFT: Zero-Padding



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

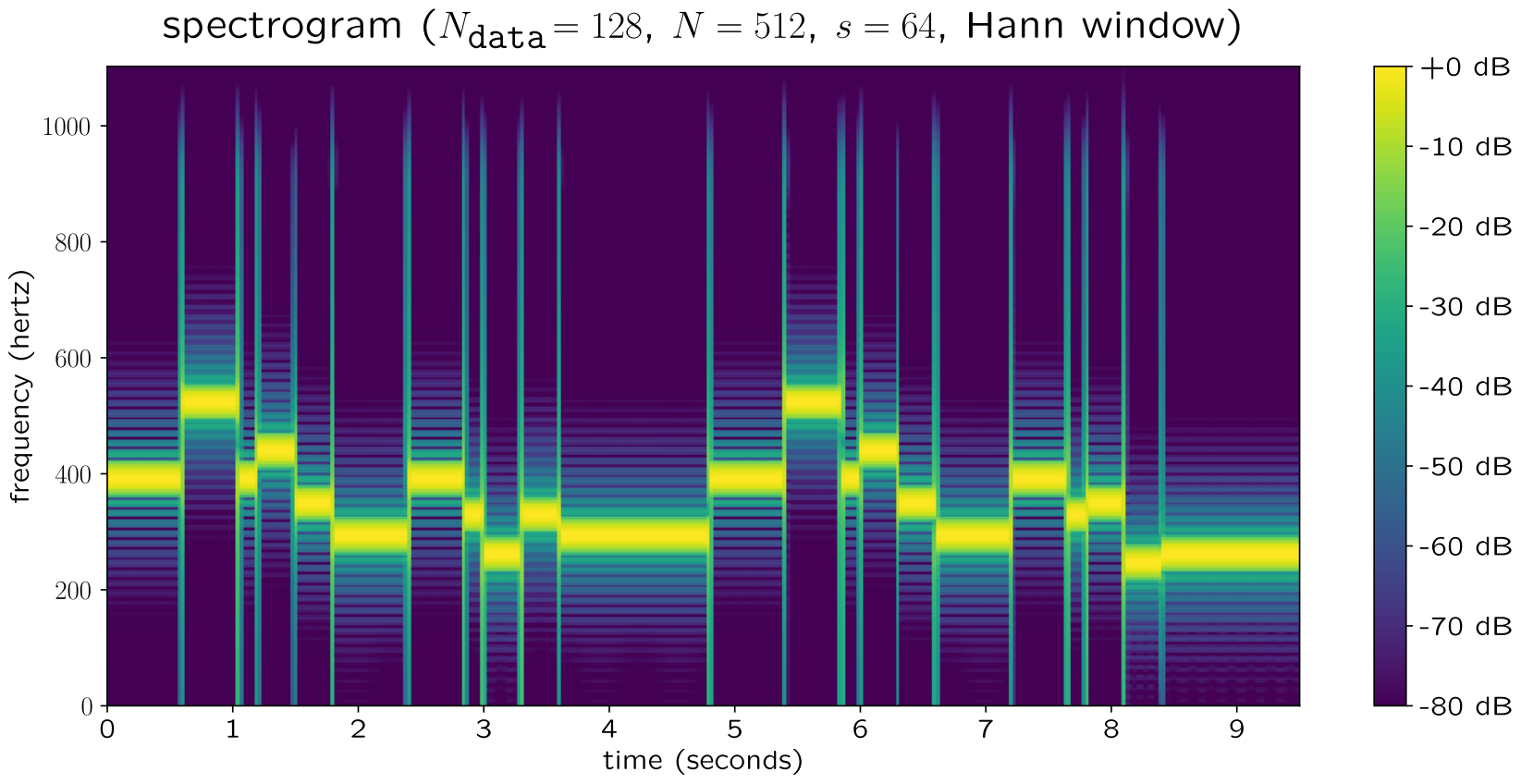
STFT: Zero-Padding



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

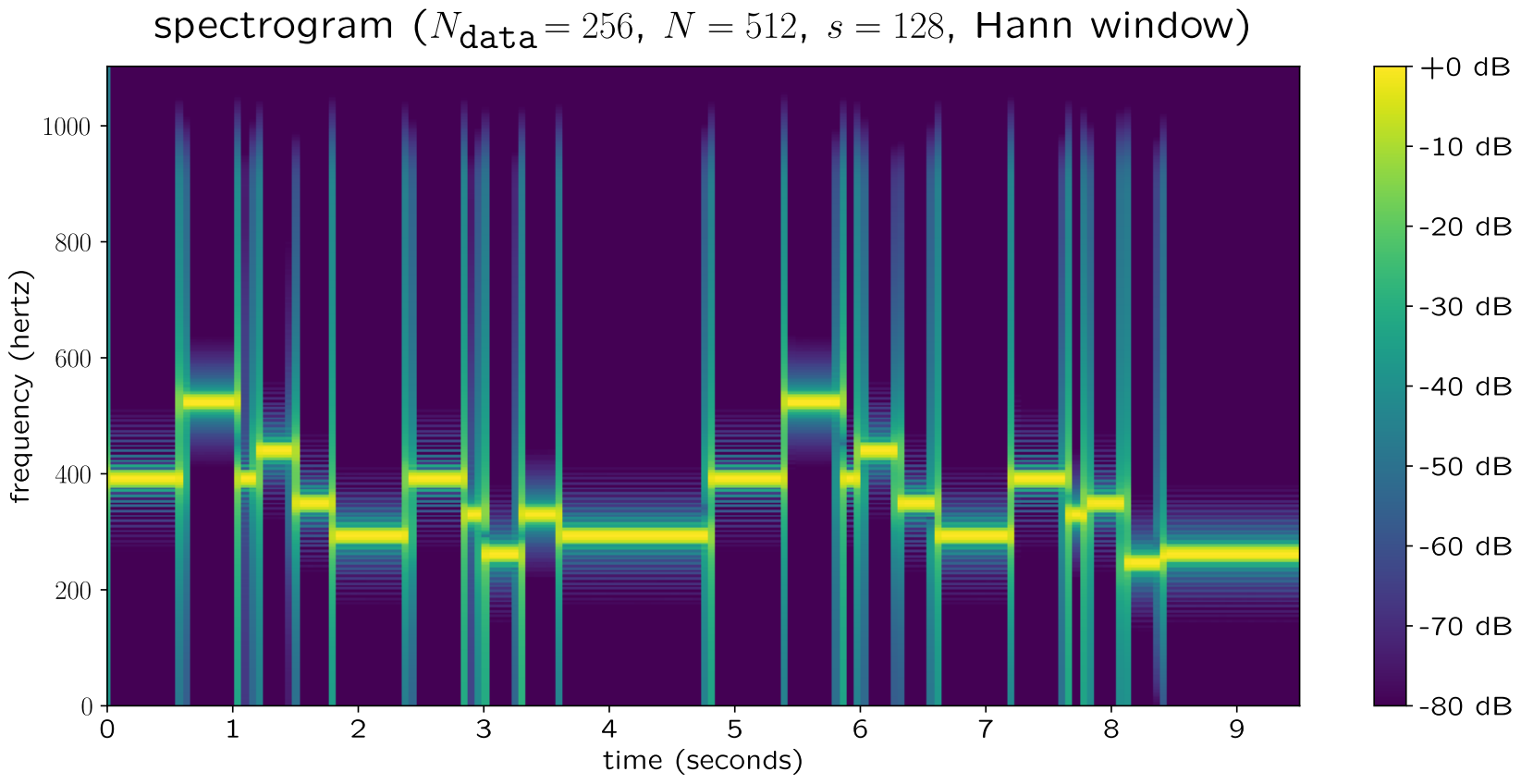
STFT: Zero-Padding



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

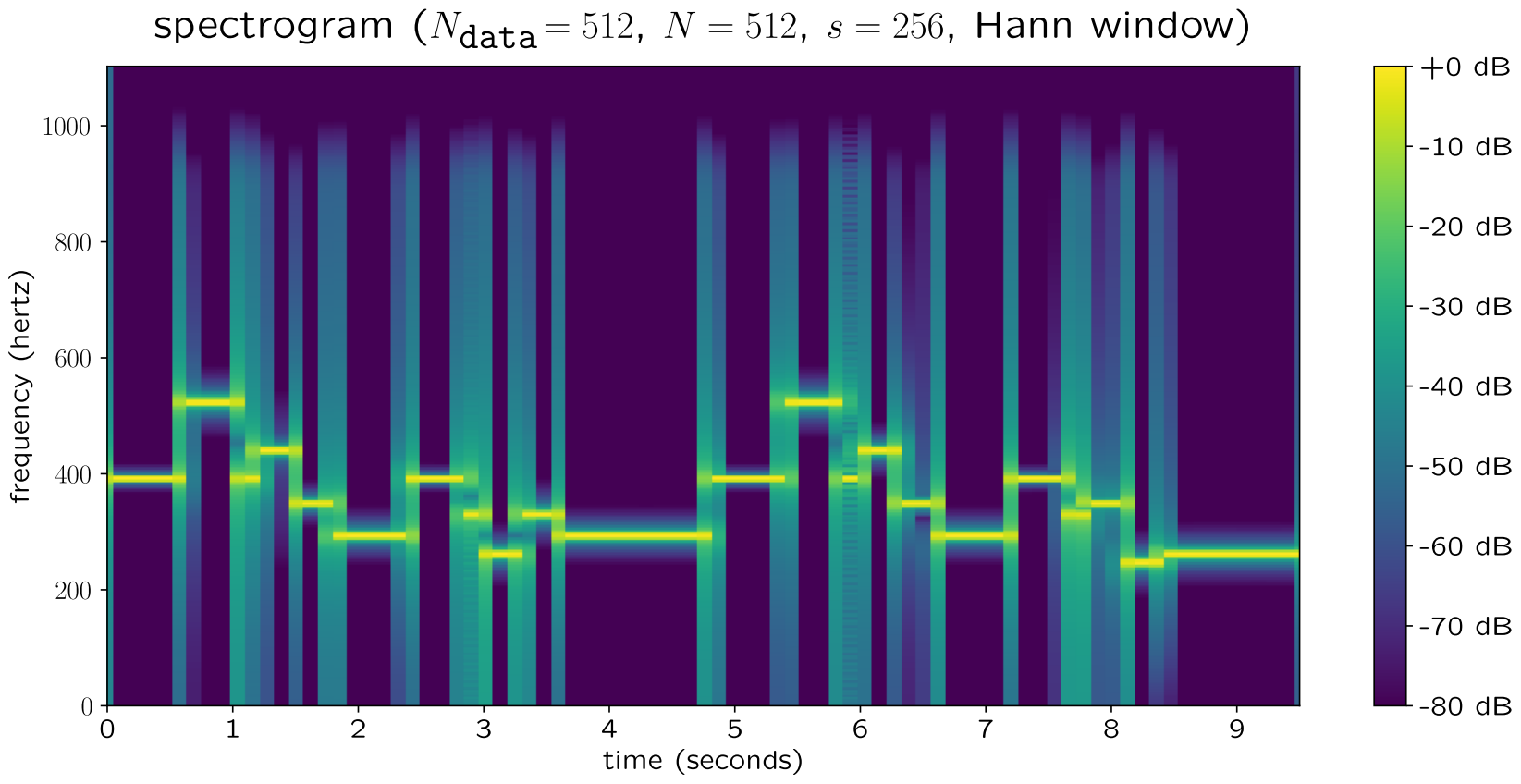
STFT: Zero-Padding



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

STFT: Zero-Padding



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

Short-Time Fourier Transform

Parameters:

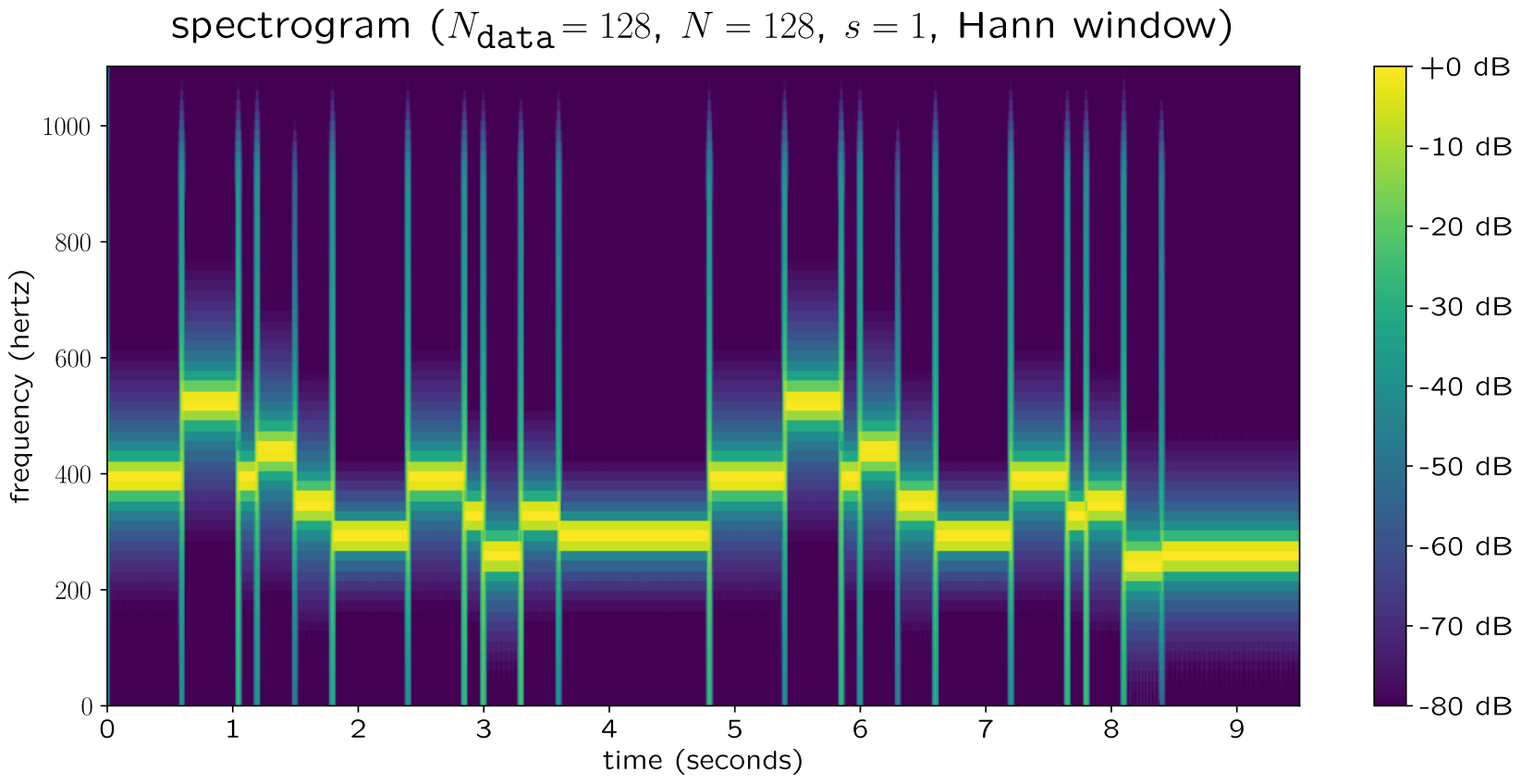
- type of window: $w[n]$
- DFT size: N
- step size: s

Short-Time Fourier Transform

Parameters:

- type of window: $w[n]$
- DFT size: N
- step size: s

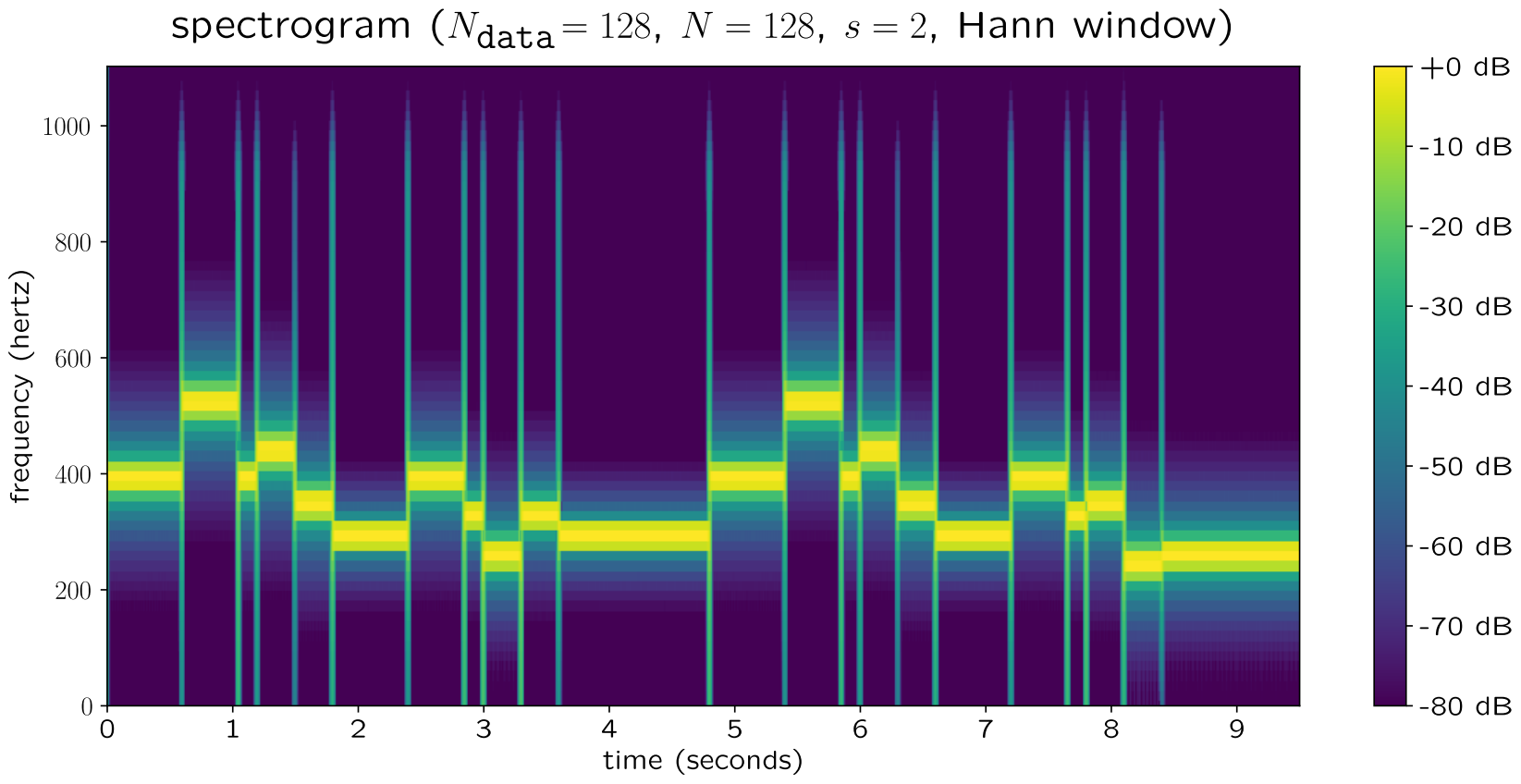
STFT: Step Size



Decibels: $x_{\mathsf{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

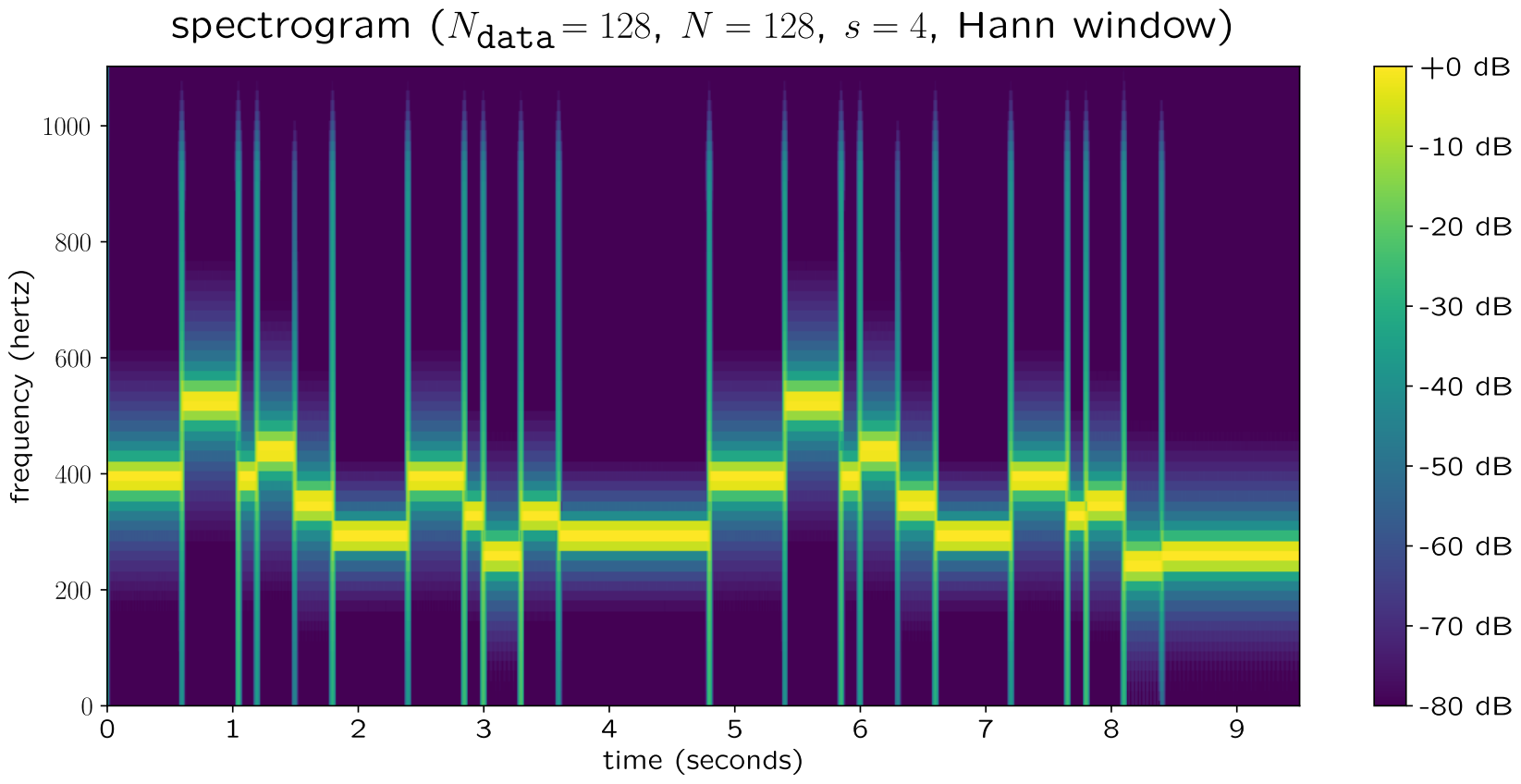
STFT: Step Size



Decibels: $x_{\mathsf{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

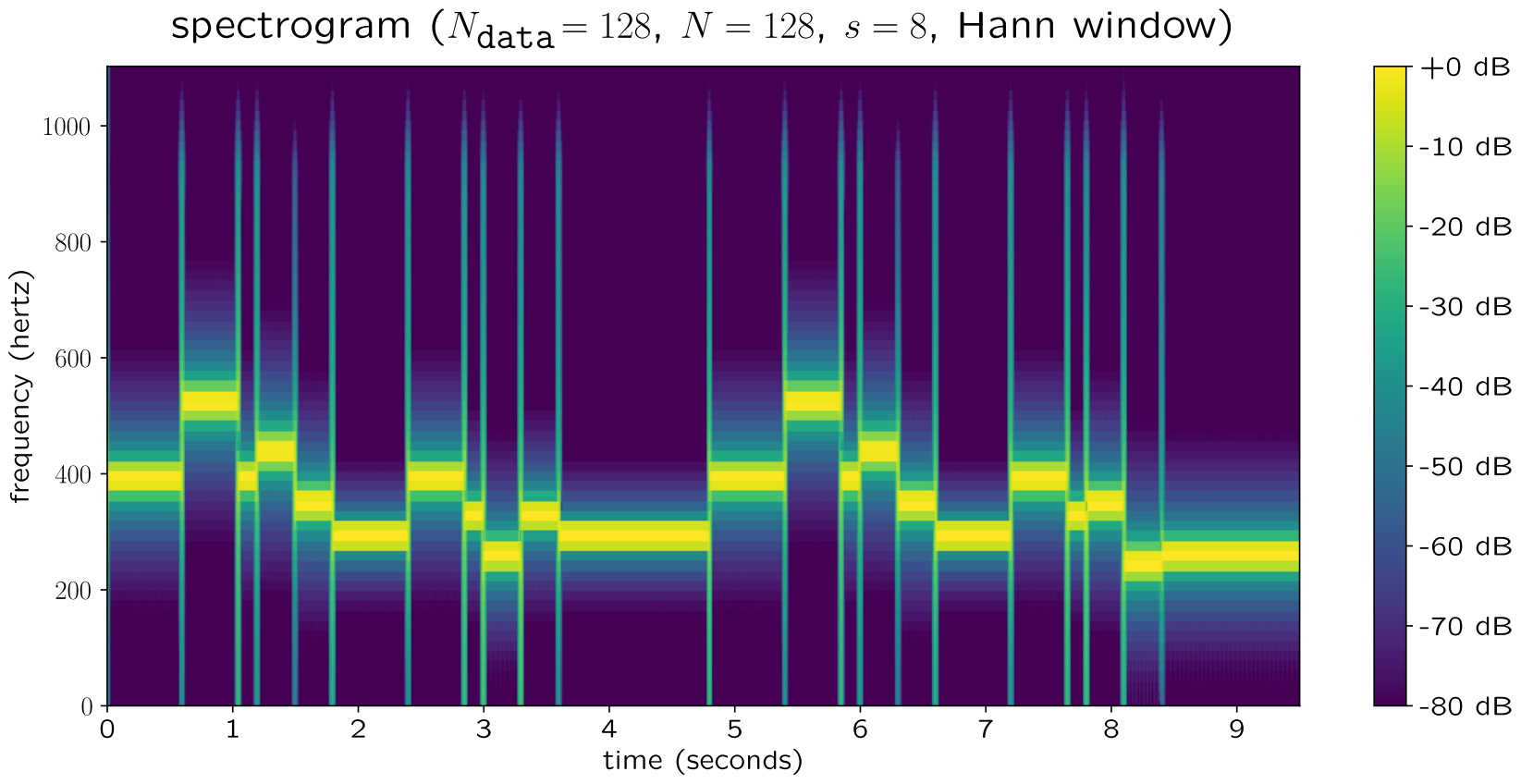
STFT: Step Size



Decibels: $x_{\mathsf{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

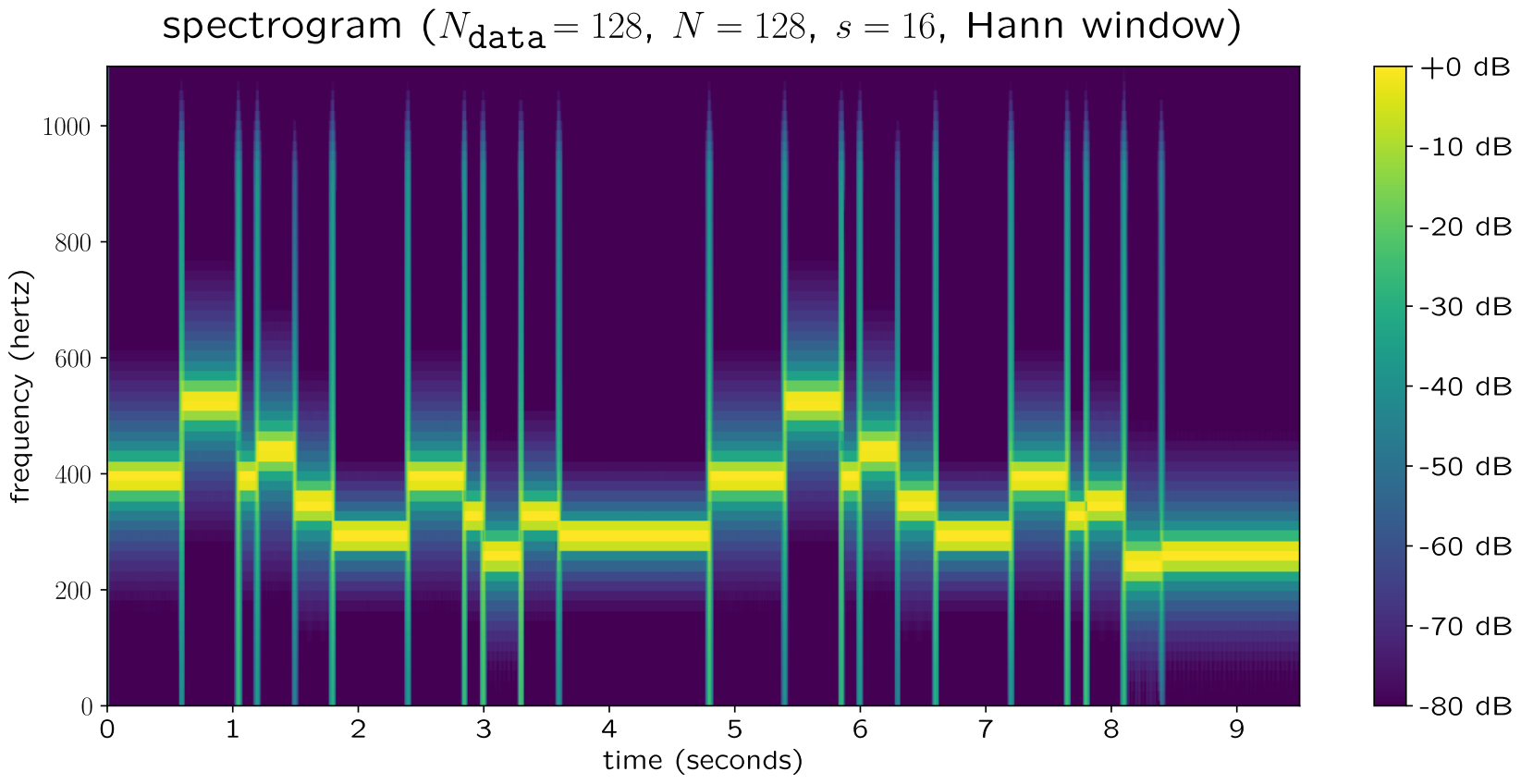
STFT: Step Size



Decibels: $x_{\mathsf{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

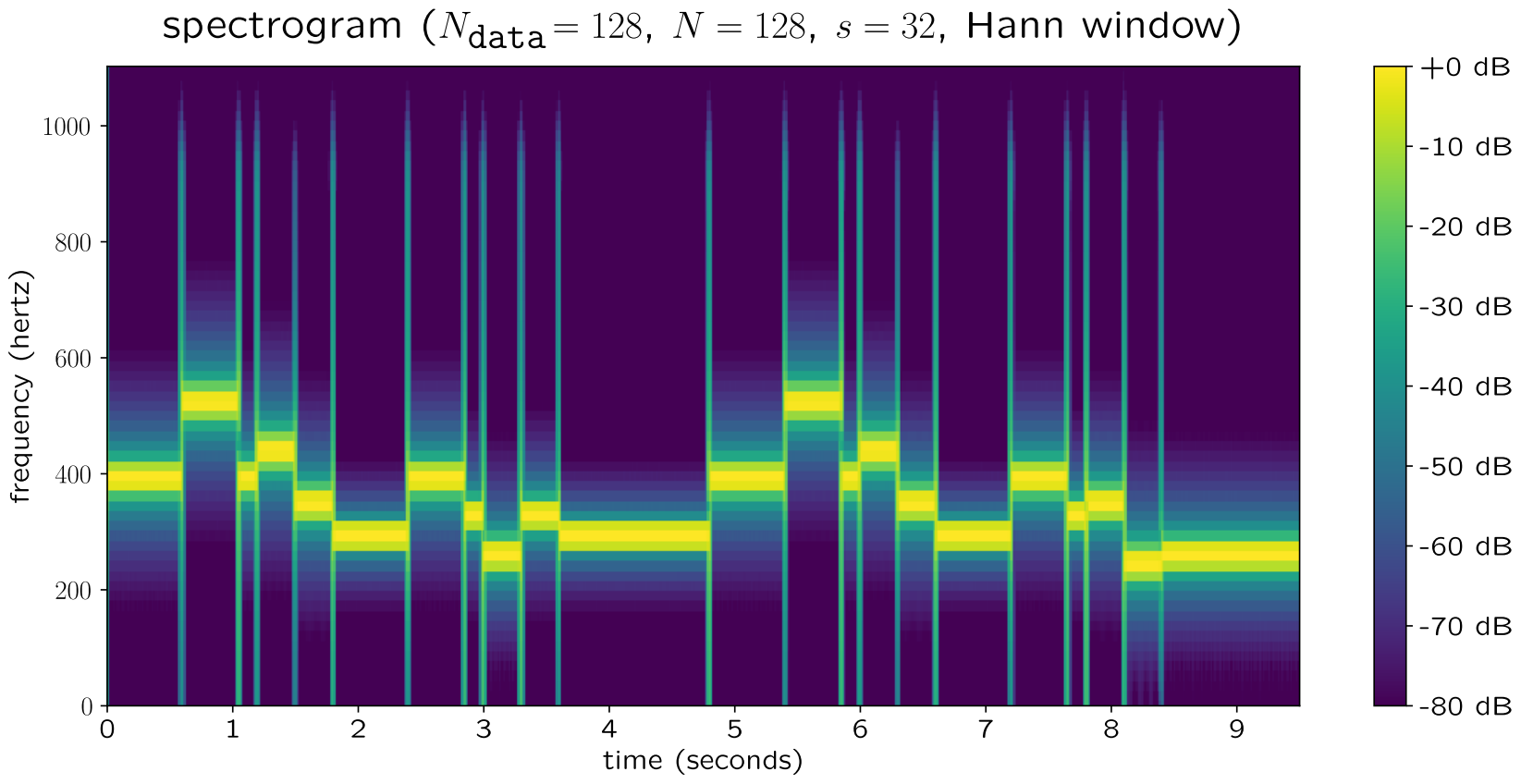
STFT: Step Size



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

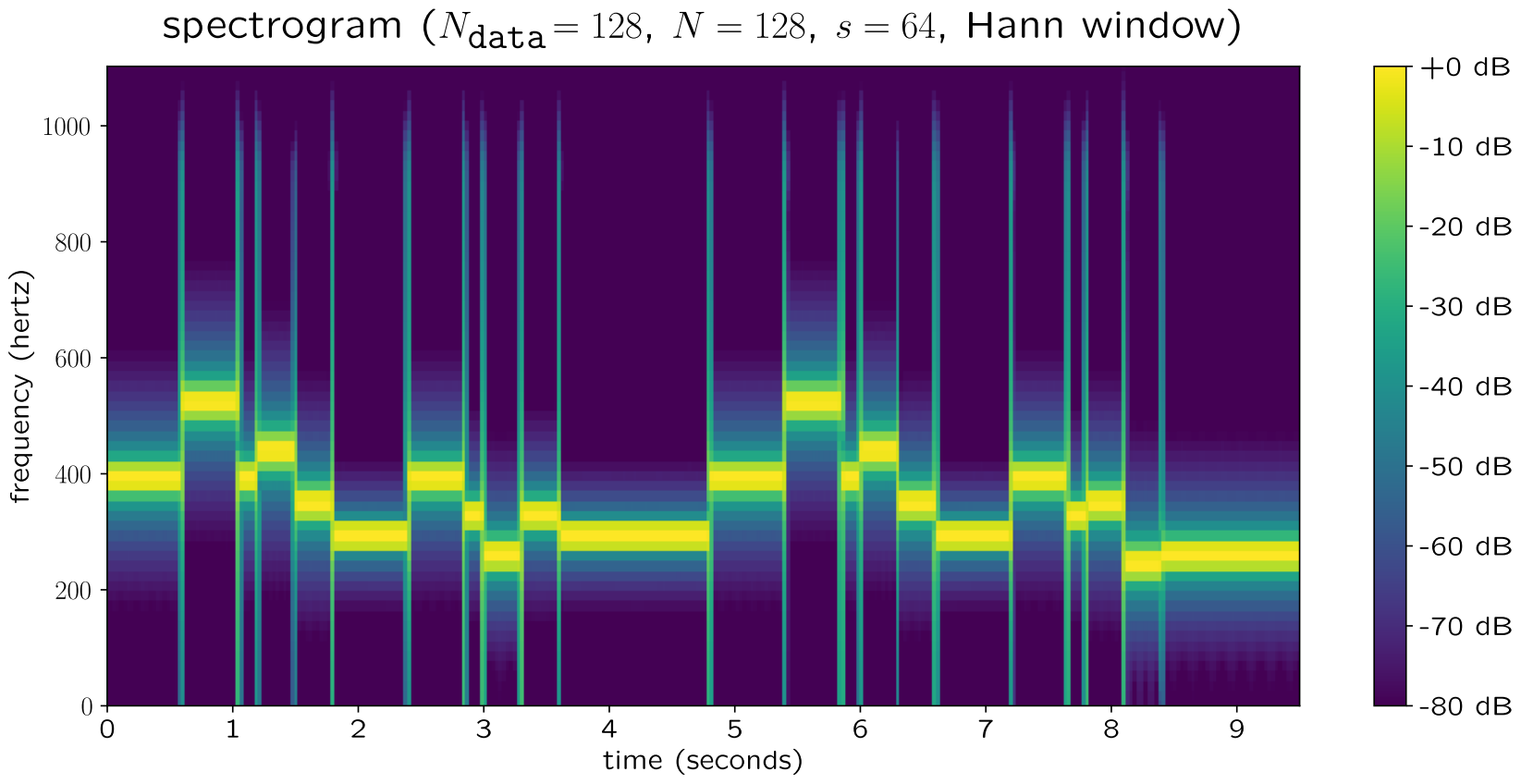
STFT: Step Size



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

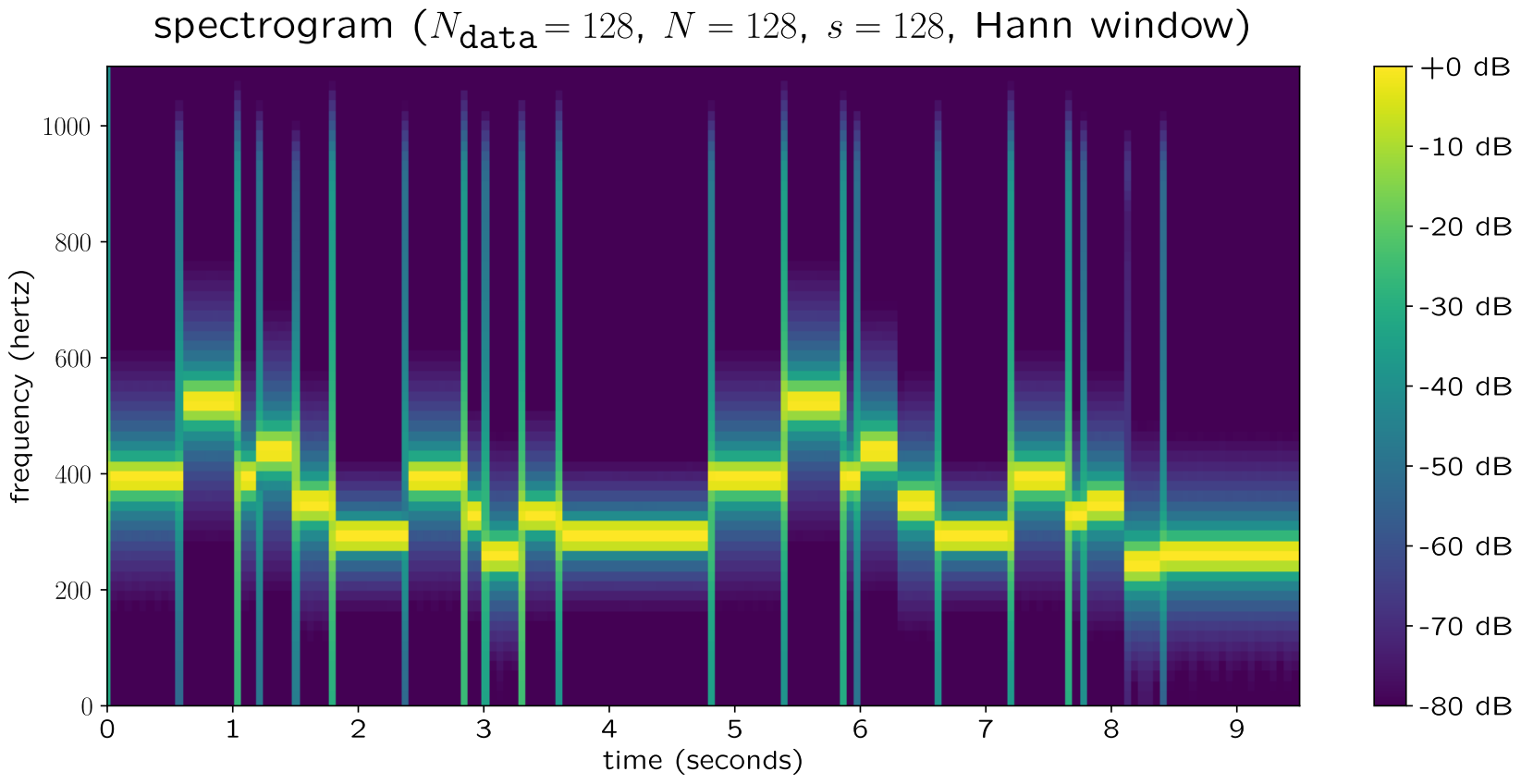
STFT: Step Size



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

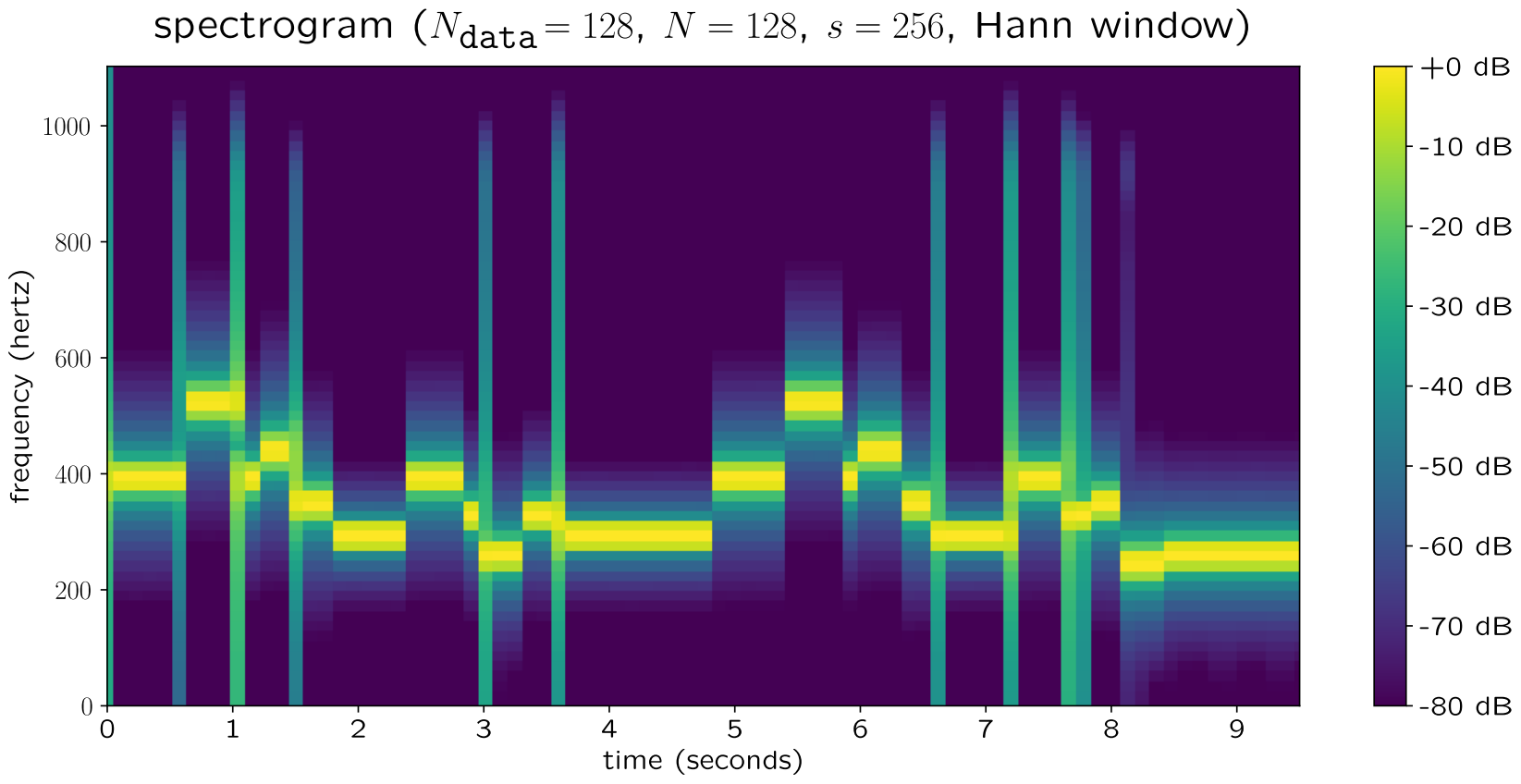
STFT: Step Size



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

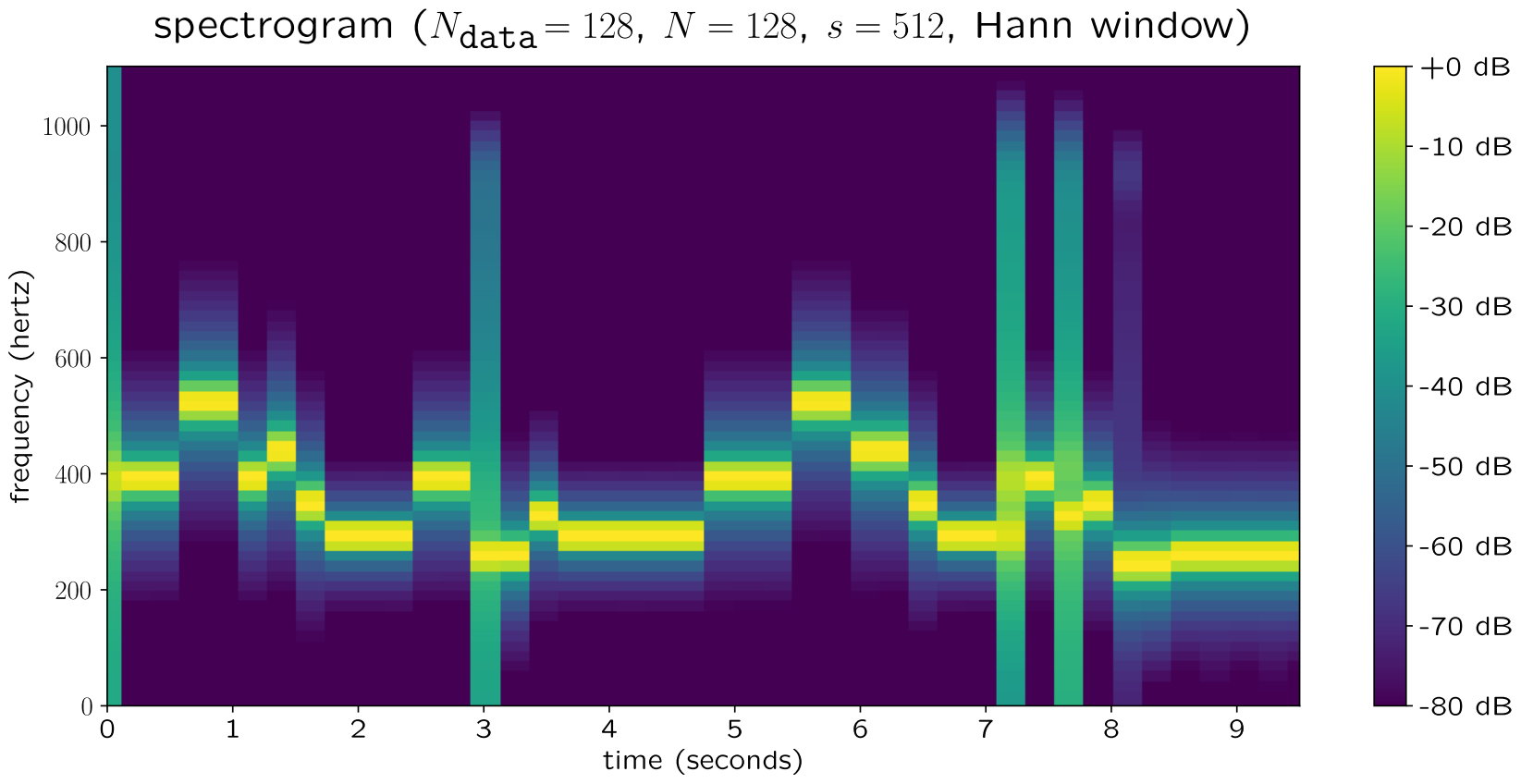
STFT: Step Size



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

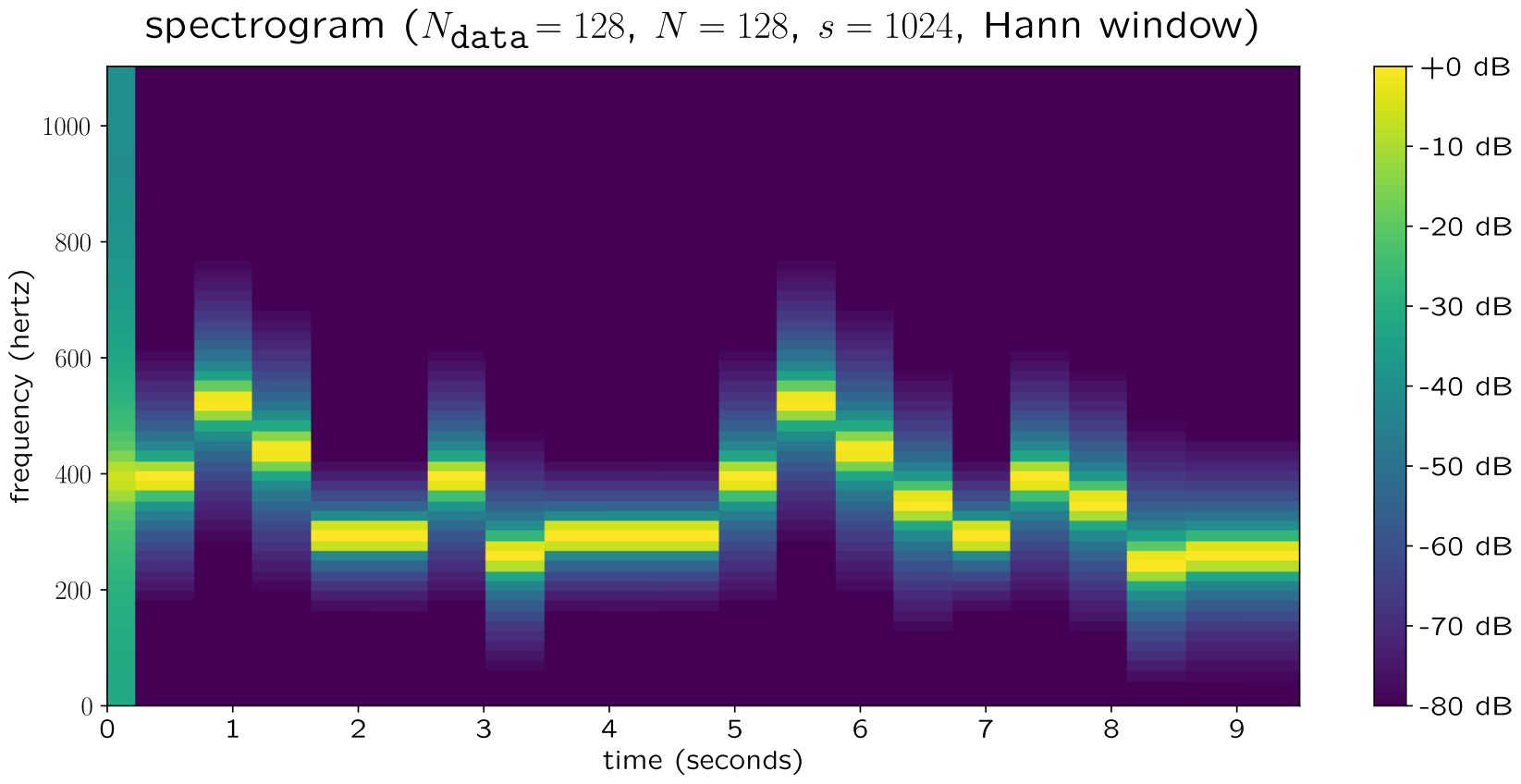
STFT: Step Size



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

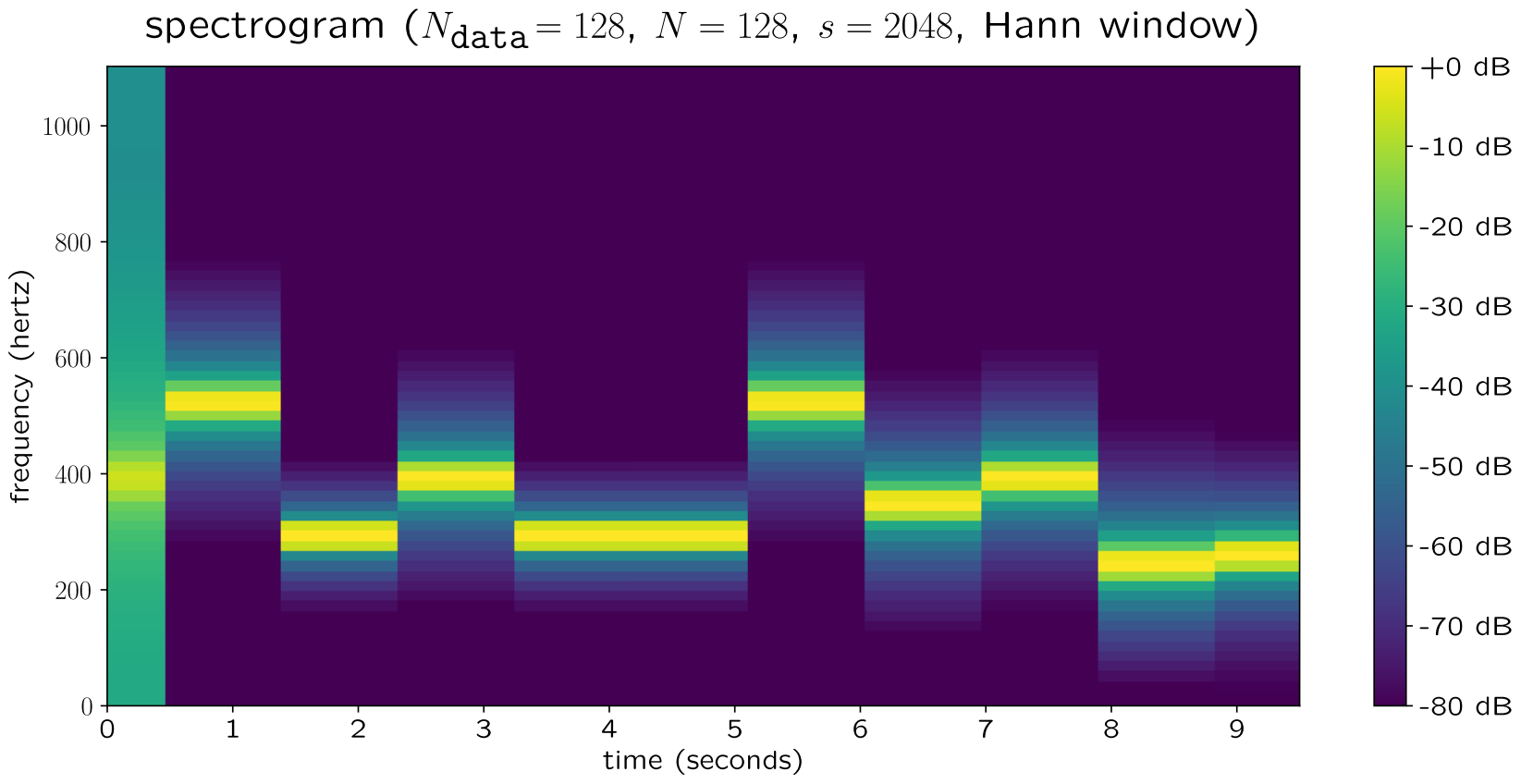
STFT: Step Size



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

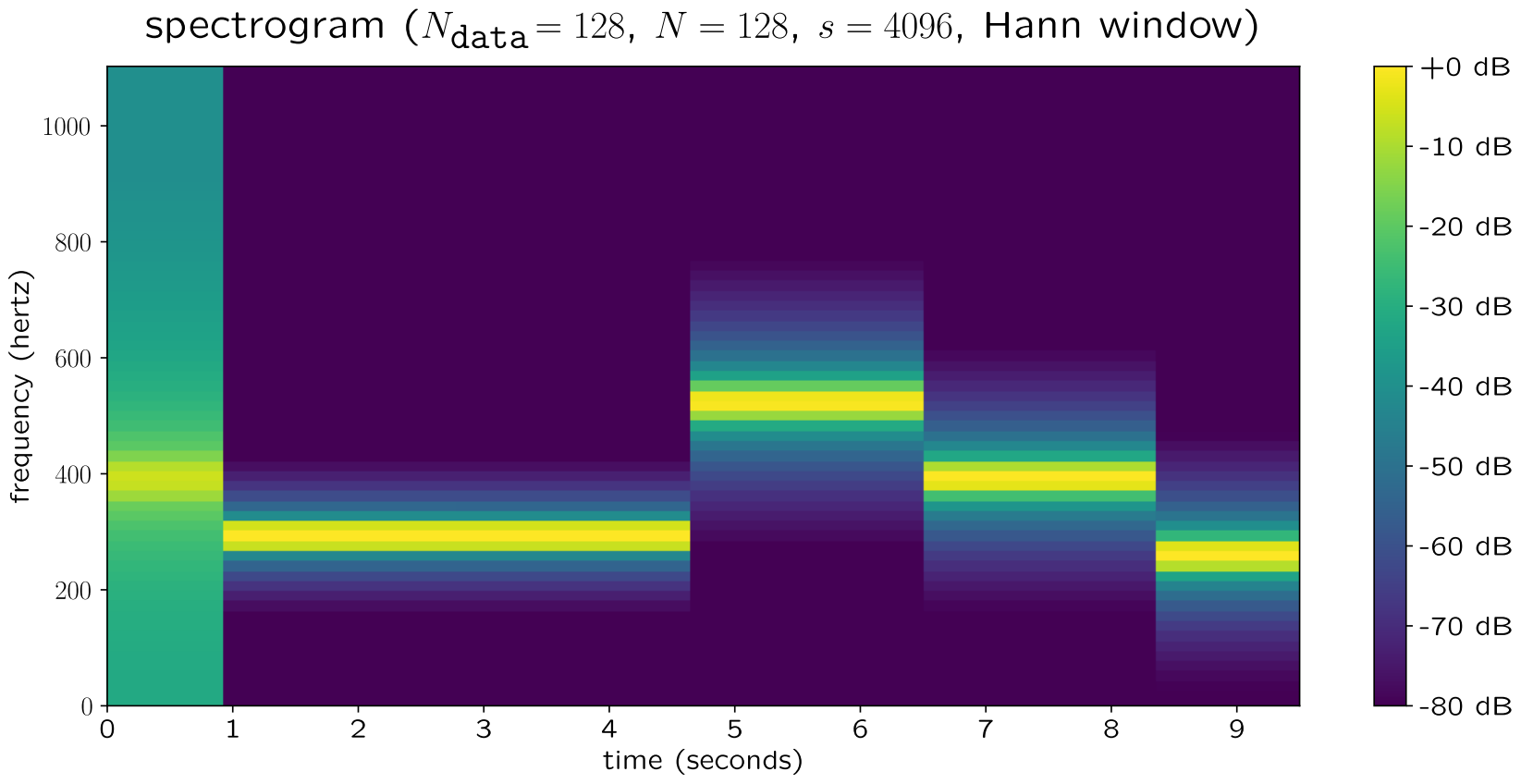
STFT: Step Size



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

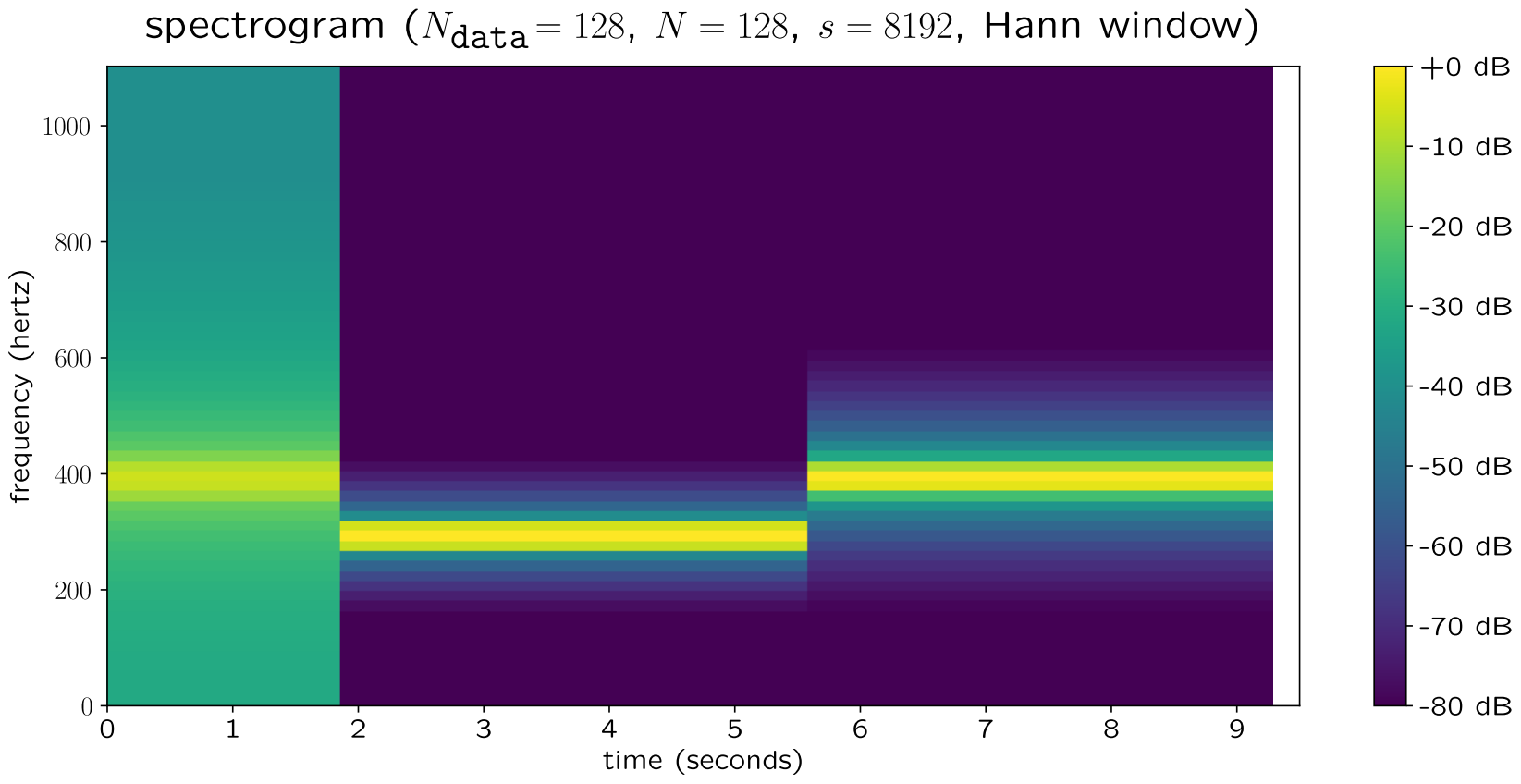
STFT: Step Size



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$

STFT: Step Size



Decibels: $x_{\text{dB}} = 10 \log_{10}(x)$

e.g., -10 dB $\iff$ $x = 10^{-1}$ and -20 dB $\iff$ $x = 10^{-2}$