Lecture 22: Aliasing in Time: the DFT

Mark Hasegawa-Johnson

ECE 401: Signal and Image Analysis, Fall 2020

Outline

Transforms you know

- **Fourier Series:**

$$X_k = \frac{1}{T_0}\int_0^{T_0} x(t)e^{-j\frac{2\pi kt}{T_0}}dt \quad \leftrightarrow \quad x(t) = \sum_{k=-\infty}^{\infty} X_k e^{j\frac{2\pi kt}{T_0}}$$

- **Discrete Time Fourier Transform (DTFT):**

$$X(\omega) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n} \quad \leftrightarrow \quad x[n] = \frac{1}{2\pi}\int_{-\pi}^{\pi} X(\omega)e^{j\omega n}d\omega$$

- **Discrete Fourier Transform (DFT):**

$$X[k] = \sum_{n=0}^{N-1} x[n]e^{-j\frac{2\pi kn}{N}} \quad \leftrightarrow \quad x[n] = \frac{1}{N}\sum_{k=0}^{N-1} X[k]e^{j\frac{2\pi kn}{N}}$$

DFT = Frequency samples of the DTFT of a finite-length signal

Suppose $x[n]$ is nonzero only for $0 \leq n \leq N-1$. Then

$$
\begin{aligned}
X[k] &= \sum_{n=0}^{N-1} x[n] e^{-j\frac{2\pi kn}{N}} \\
&= \sum_{n=-\infty}^{\infty} x[n] e^{-j\frac{2\pi kn}{N}} \\
&= X(\omega_k), \quad \omega_k = \frac{2\pi k}{N}
\end{aligned}
$$

DFT = Discrete Fourier series of a periodic signal

Suppose $x[n]$ is periodic, with a period of N. If it were defined in continuous time, its Fourier series would be

$$X_k = \frac{1}{T_0}\int_0^{T_0} x(t)e^{-j\frac{2\pi kt}{T_0}}dt$$

The discrete-time Fourier series could be defined similarly, as

$$X_k = \frac{1}{N}\sum_{n=0}^{N-1} x[n]e^{-j\frac{2\pi kn}{N}}$$
$$= \frac{1}{N}X[k]$$

Outline

Frequency Response

- **Fourier Series:**

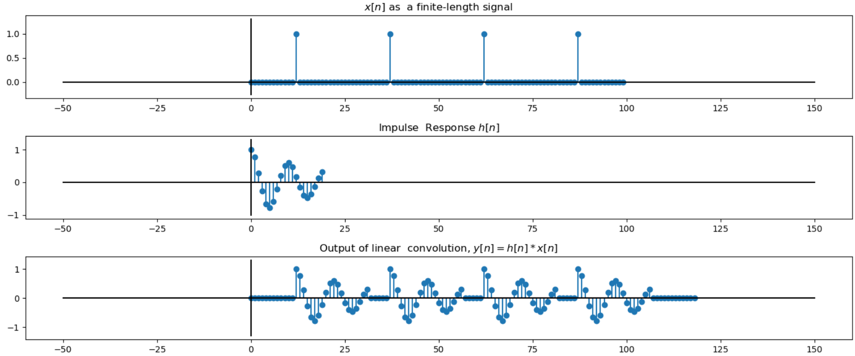
$$y(t) = x(t) * h(t) \quad \leftrightarrow \quad Y_k = H(\omega_k)X_k$$

- **DTFT:**

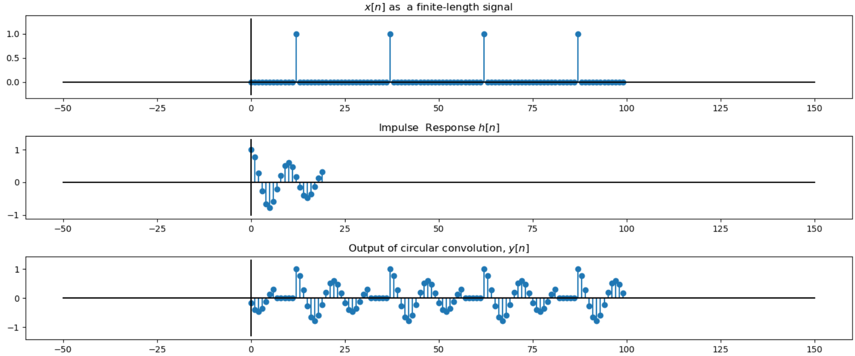
$$y[n] = x[n] * h[n] \quad \leftrightarrow \quad Y(\omega) = H(\omega)X(\omega)$$

- **DFT:**
  - If $y[n] = x[n] * h[n]$, does that mean $Y[k] = H[k]X[k]$?
  - **Only** if you assume $x[n]$ periodic. If you assume $x[n]$ is finite-length, then the formula fails.

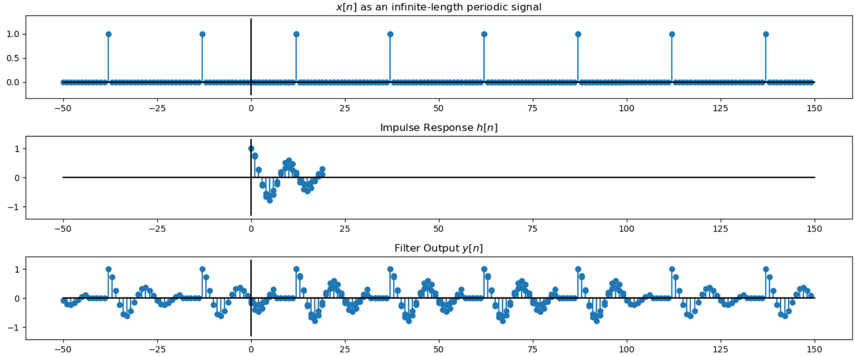
Example: $y[n] = x[n] * h[n]$



Example: $Y[k] = H[k]X[k]$



Example: $Y[k]$ and $X[k]$ as Fourier series coefficients



Circular Convolution: Motivation

- The inverse transform of $Y[k] = H[k]X[k]$ is the result of convolving a finite-length $h[n]$ with an **infinitely periodic** $x[n]$.
- Suppose $x[n]$ is defined to be **finite-length**, e.g., so you can say that $X[k] = X(\omega_k)$ (DTFT samples). Then $y[n] \neq h[n] * x[n]$. We need to define a new operator called **circular convolution**.

Circular Convolution: Definition

The inverse transform of $H[k]X[k]$ is a circular convolution:

$$Y[k] = H[k]X[k] \quad \leftrightarrow \quad y[n] = h[n] \circledast x[n],$$

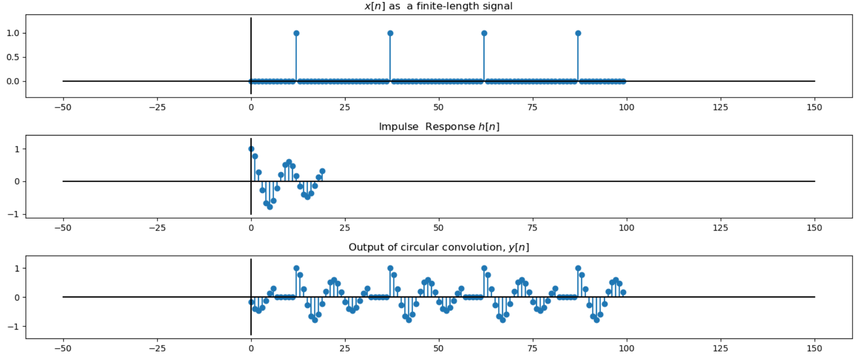
where circular convolution is defined to mean:

$$h[n] \circledast x[n] \equiv \sum_{m=0}^{N-1} h\left[m\right] x\left[\langle n - m \rangle_N\right]$$

in which the $\langle\rangle_N$ means "modulo N:"

$$\langle n \rangle_N = \begin{cases} n - N & N \le n < 2N \\ n & 0 \le n < N \\ n + N & -N \le n < 0 \\ & \vdots \end{cases}$$

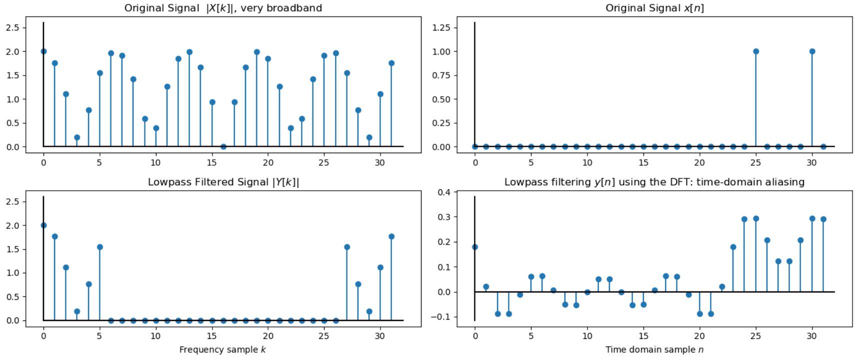
$Y[k] = H[k]X[k] \quad \leftrightarrow \quad y[n] = h[n] \circledast x[n]$



Practical Issues: Can I use DFT to filter a signal?

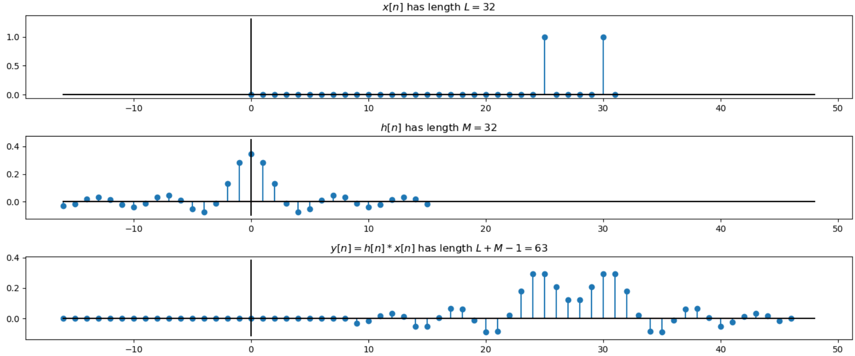
- Sometimes, it's easier to design a filter in the frequency domain than in the time domain.
- ...but if you multiply $Y[k] = H[k]X[k]$, that gives $y[n] = h[n] \circledast x[n]$, which is not the same thing as $y[n] = h[n] * x[n]$.
- Is there any way to use DFT to do filtering?

Practical Issues: Filtering in DFT domain causes circular convolution



The goal: Linear convolution

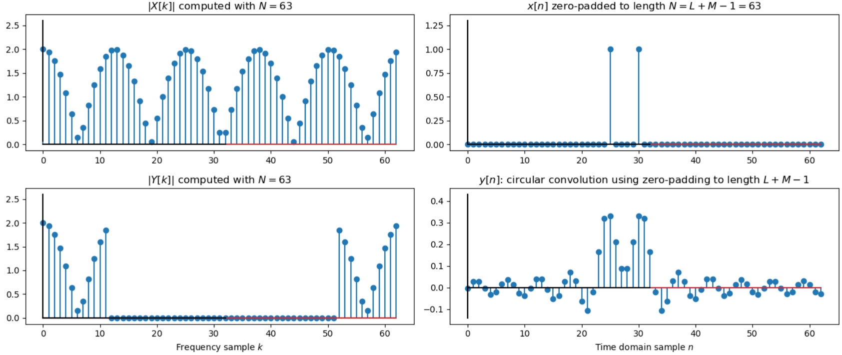
When you convolve a length-L signal, $x[n]$, with a length-M filter $h[n]$, you get a signal $y[n]$ that has length $M+L-1$:



In this example, $x[n]$ has length $L=32$, and $h[n]$ has length $M=32$, so $y[n]$ has length $L+M-1=63$.

How to make circular convolution = linear convolution

So in order to make circular convolution equivalent to linear convolution, you need to use a DFT length that is at least $N \geq M + L - 1$:



Zero-padding

This is done by just zero-padding the signals:

$$x_{ZP}[n] = \begin{cases} x[n] & 0 \le n \le L-1 \\ 0 & L \le n \le N-1 \end{cases}$$

$$h_{ZP}[n] = \begin{cases} h[n] & 0 \le n \le M-1 \\ 0 & M \le n \le N-1 \end{cases}$$

Then we find the N-point DFT, $X[k]$ and $H[k]$, multiply them together, and inverse transform to get $y[n]$.

Zero-padding doesn't change the spectrum

Suppose $x[n]$ is of length $L < N$. Suppose we define

$$x_{ZP}[n] = \begin{cases} x[n] & 0 \le n \le L-1 \\ 0 & L \le n \le N-1 \end{cases}$$

Then

$$X_{ZP}(\omega) = X(\omega)$$

...so zero-padding is the right thing to do!

Outline

Truncating changes the spectrum

On the other hand, suppose $s[n]$ is of length $M > L$. Suppose we define

$$x[n] = \begin{cases} s[n] & 0 \le n \le L-1 \\ 0 & L \le n \le N-1 \end{cases}$$

Then

$$X(\omega) \ne S(\omega)$$

and

$$X[k] \ne S[k]$$

How does truncating change the spectrum?

Truncating, as it turns out, is just a special case of windowing:

$$x[n] = s[n]w_R[n]$$

where the "rectangular window," $w_R[n]$, is defined to be:

$$w_R[n] = \begin{cases} 1 & 0 \le n \le L-1 \\ 0 & \text{otherwise} \end{cases}$$

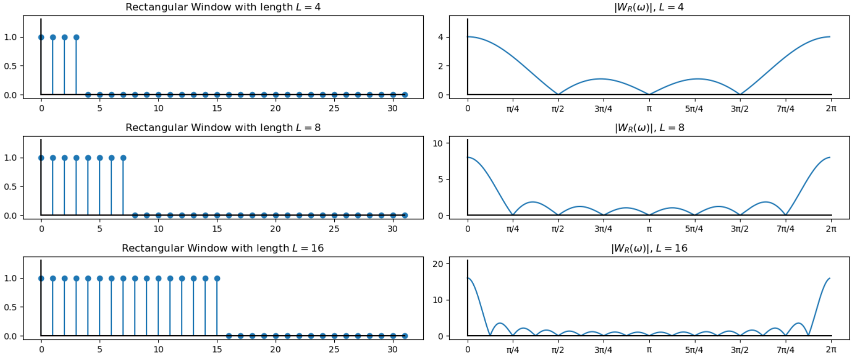
Spectrum of the rectangular window

$$w_R[n] = \begin{cases} 1 & 0 \le n \le L-1 \\ 0 & \text{otherwise} \end{cases}$$

The spectrum of the rectangular window is

$$\begin{aligned} W_R(\omega) &= \sum_{n=-\infty}^{\infty} w[n]e^{-j\omega n} \\ &= \sum_{n=0}^{L-1} e^{-j\omega n} \\ &= e^{-j\omega\left(\frac{L-1}{2}\right)} \frac{\sin(\omega L/2)}{\sin(\omega/2)} \end{aligned}$$

Spectrum of the rectangular window

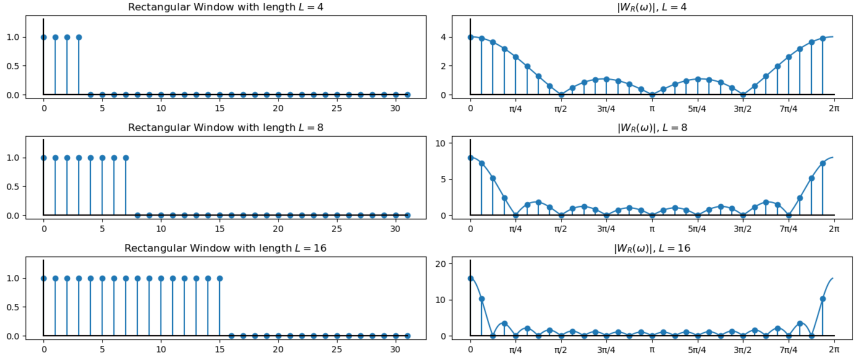


DFT of the rectangular window

The DFT of a rectangular window is just samples from the DTFT:

$$W_R[k] = W_R\left(\frac{2\pi k}{N}\right)$$

DFT of the rectangular window

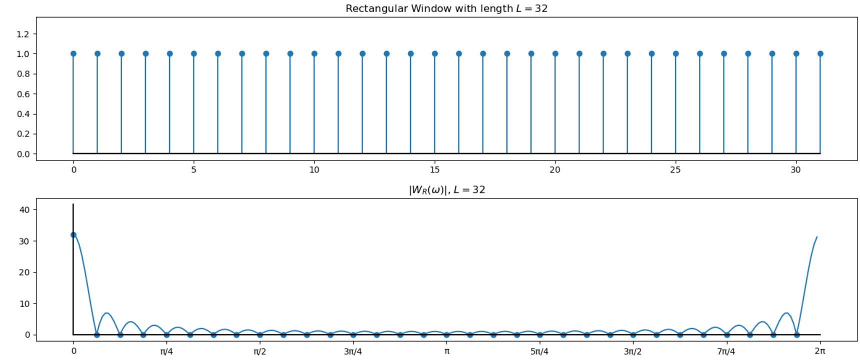


DFT of a length-N rectangular window

There is an interesting special case of the rectangular window. When $L = N$:

$$
\begin{aligned}
W_R[k] &= W_R\left(\frac{2\pi k}{N}\right)\\
&= e^{-j\frac{2\pi k}{N}\left(\frac{N-1}{2}\right)}\frac{\sin\left(\frac{2\pi k}{N}\left(\frac{N}{2}\right)\right)}{\sin\left(\frac{2\pi k}{N}\left(\frac{1}{2}\right)\right)}\\
&= \begin{cases} 1 & k = 0\\ 0 & \text{otherwise}\end{cases}
\end{aligned}
$$

DFT of a length-N rectangular window



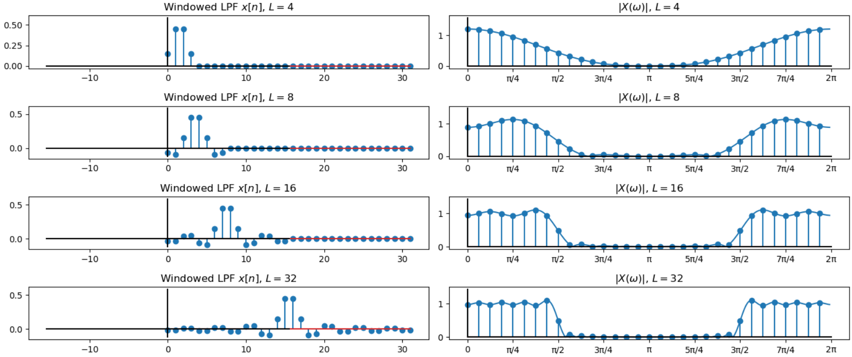
How does truncating change the spectrum?

When we window in the time domain:

$$x[n] = s[n]w_R[n]$$

that corresponds to $X(\omega)$ being a kind of smoothed, rippled version of $S(\omega)$, with smoothing kernel of $W_R(\omega)$.

How does truncating change the spectrum?

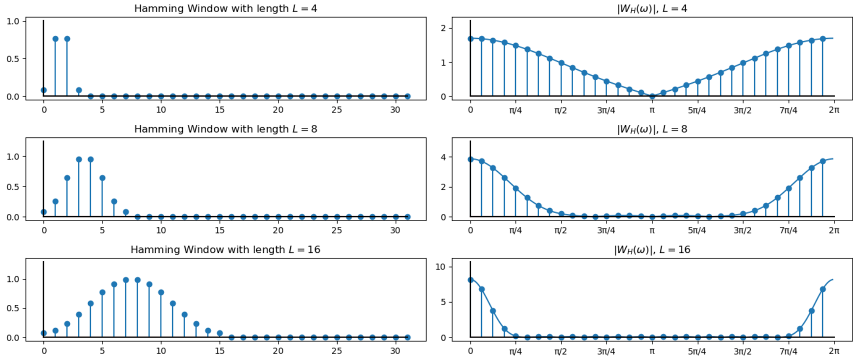


Hamming window

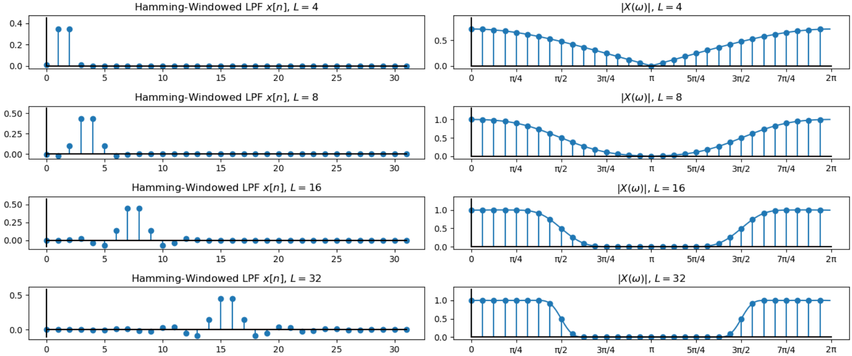
In order to reduce out-of-band ripple, we can use a Hamming window, Hann window, or triangular window. The one with the best spectral results is the Hamming window:

$$w_H[n] = w_R[n]\left(0.54 - 0.46\cos\left(\frac{2\pi n}{L-1}\right)\right)$$

Hamming window



Hamming window



Outline

What is the DFT of a Pure Tone?

What is the DFT of a pure tone? Say, a cosine:

$$x[n] = 2\cos(\omega_0 n) = e^{j\omega_0 n} + e^{-j\omega_0 n}$$

Actually, it's a lot easier to compute the DFT of a complex exponential, so let's say "complex exponential" is a pure tone:

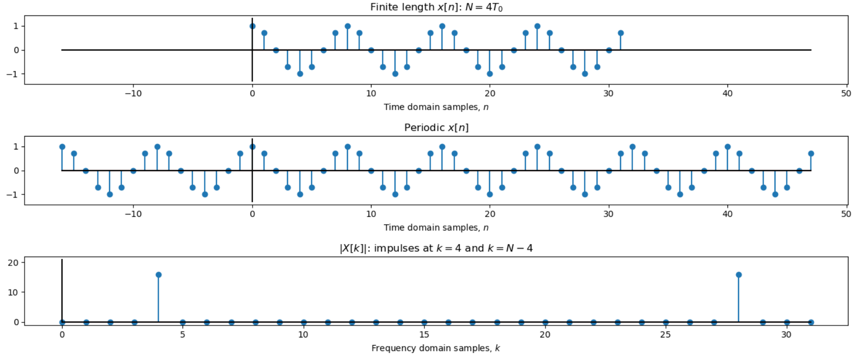
$$x[n] = e^{j\omega_0 n}$$

where $\omega_0 = \frac{2\pi}{T_0}$ is the fundamental frequency, and T_0 is the period.

What is the DFT of a Pure Tone?

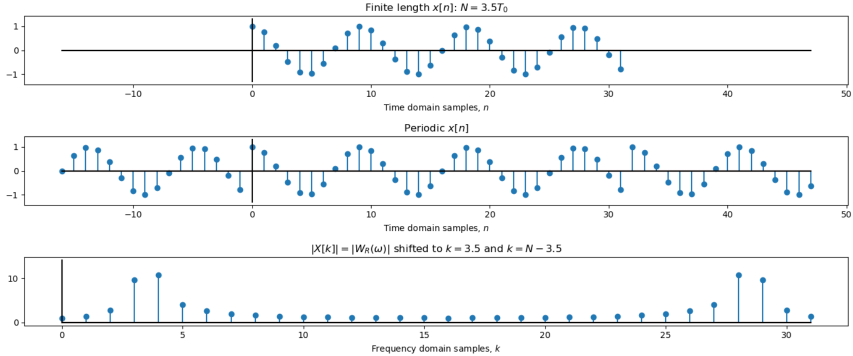
The DFT is a scaled version of the Fourier series. So if the cosine has a period of $T_0 = \frac{N}{k_0}$ for some integer k_0, then the DFT is

$$X[k] = \begin{cases} 1 & k = k_0, N - k_0 \\ 0 & \text{otherwise} \end{cases}:$$



What is the DFT of a Pure Tone?

If N is not an integer multiple of T_0, though, then $|X[k]|$ gets messy:



What is the DFT of a Pure Tone?

Let's solve it. If $x[n] = e^{j\omega_0 n}$, then

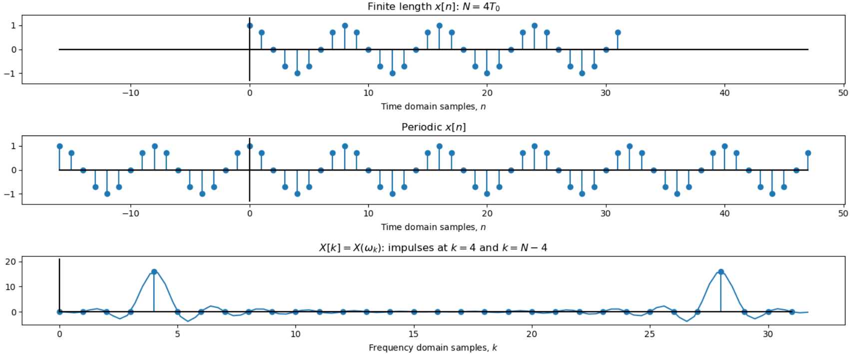
$$\begin{aligned} X[k] &= \sum_{n=0}^{N-1} x[n] e^{-j\frac{2\pi kn}{N}} \\ &= \sum_{n=0}^{N-1} e^{j\left(\omega_0 - \frac{2\pi k}{N}\right)n} \\ &= W_R\left(\frac{2\pi k}{N} - \omega_0\right) \end{aligned}$$

So the DFT of a pure tone is just a frequency-shifted version of the rectangular window spectrum!

What is the DFT of a Pure Tone?

$$X[k] = W_R\left(\frac{2\pi k}{N} - \omega_0\right)$$

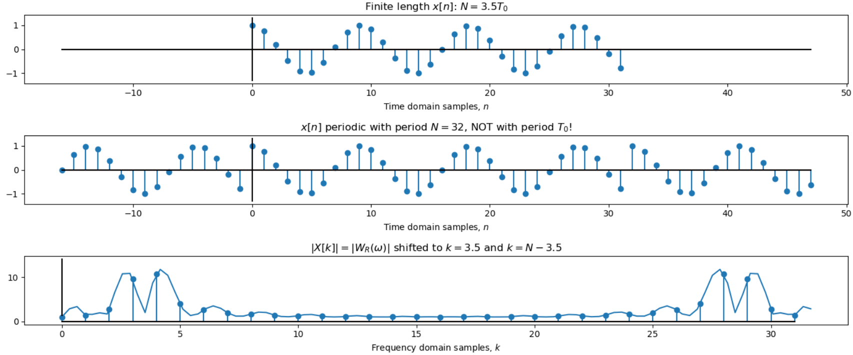
If N is a multiple of T_0, then the numerator is always zero, and $X[k]$ samples the sinc right at its zero-crossings:



What is the DFT of a Pure Tone?

$$X[k] = W_R\left(\frac{2\pi k}{N} - \omega_0\right)$$

If N is NOT a multiple of T_0, then $X[k]$ samples the sinc in more complicated places:



Outline

Summary: Circular Convolution

- If you try to compute convolution by multiplying DFTs, you get **circular convolution** instead of linear convolution. This effect is sometimes called "time domain aliasing," because the output signal shows up at an unexpected time:

$$h[n] \circledast x[n] \equiv \sum_{m=0}^{N-1} h\left[m\right] x\left[\langle n-m\rangle_N\right]$$

- The way to avoid this is to **zero-pad** your signals prior to taking the DFT:

$$x_{ZP}[n] = \begin{cases} x[n] & 0 \le n \le L-1 \\ 0 & L \le n \le N-1 \end{cases} \quad h_{ZP}[n] = \begin{cases} h[n] & 0 \le n \le M-1 \\ 0 & M \le n \le N-1 \end{cases}$$

Then you can compute $y[n] = h[n] * x[n]$ by using a length-N DFT, as long as $N \ge L + M - 1$.

Summary: Windowing

- If you truncate a signal in order to get it to fit into a DFT, then you get windowing effects:

$$x[n] = s[n]w_R[n]$$

where

$$w_R[n] = \begin{cases} 1 & 0 \le n \le L-1 \\ 0 & \text{otherwise} \end{cases} \quad \leftrightarrow \quad W_R(\omega) = e^{-j\omega\left(\frac{L-1}{2}\right)} \frac{\sin\left(\frac{\omega L}{2}\right)}{\sin\left(\frac{\omega}{2}\right)}$$

- The DFT of a pure tone is a frequency-shifted window spectrum:

$$x[n] = e^{j\omega_0 n} \quad \leftrightarrow \quad X[k] = W_R\left(\frac{2\pi k}{N} - \omega_0\right)$$