6.300: Signal Processing

Review Material Story Sheet for Quiz #2
Author Titus K. Roesler (tkr@mit.edu)

The Story So Far (Quiz #1 to Quiz #2)

Systems

- **10/02:** LTI Systems — differential equations and difference equations
- **10/07:** Impulse Response — convolution and the superposition integral
- **10/09:** Frequency Response — convolution in time, multiplication in frequency
- **10/14:** Communication Systems — amplitude modulation

Discrete Fourier Transform

- **10/14:** Discrete Fourier Transform, Part #1 — a discrete-time, discrete-frequency Fourier transform
- **10/16:** Discrete Fourier Transform, Part #2 — frequency resolution and circular convolution
- **10/21:** Fast Fourier Transform — divide-and-conquer algorithm to compute the DFT
- **10/28:** Short-Time Fourier Transforms — moving-window Fourier transforms
- **10/30:** Speech Processing — source-filter model of speech production

Mathematics Review

Dimensional Analysis

$$T_0 \text{ (seconds)} \times f_s \text{ (samples / second)} = N_0 \text{ (samples)}$$

$$\omega_0 \text{ (radians / second)} \div f_s \text{ (samples / second)} = \Omega_0 \text{ (radians / sample)}$$

CT cyclical frequency $f_0 = 1/T_0$ — cycles per second or hertz (Hz)
CT angular frequency $\omega_0 = 2\pi/T_0 = 2\pi f_0$ — radians per second
DT angular frequency $\Omega_0 = \omega_0/f_s = 2\pi f_0/f_s = 2\pi f_0 T_s = 2\pi/N_0$ — radians per sample

Geometric Series

$$\sum_{n=0}^{N-1} z^n = \frac{1-z^N}{1-z} \qquad \sum_{n=0}^{\infty} z^n = \frac{1}{1-z} \text{ for } |z| < 1$$

Binomial Theorem

$$(\alpha+\beta)^n = \binom{n}{0}\alpha^n + \binom{n}{1}\alpha^{n-1}\beta + \binom{n}{2}\alpha^{n-2}\beta^2 + \cdots + \binom{n}{n-1}\alpha\beta^{n-1} + \binom{n}{n}\beta^n$$

$$\binom{n}{k} \equiv \frac{n!}{k!\,(n-k)!} \text{ where } n! \equiv (n)(n-1)(n-2)\cdots(3)(2)(1)$$

Continuous-Time Fourier Series

Continuous-Time Fourier Series

$f(t) = f(t+T)$ is a T-periodic function, and $\omega_0 = 2\pi/T$ denotes the fundamental angular frequency.

Continuous-Time Fourier Series in Trigonometric Form

$$f(t) = c_0 + \sum_{k=1}^{\infty} c_k \cos(k\omega_0 t) + \sum_{k=1}^{\infty} d_k \sin(k\omega_0 t)$$

$$\text{where } c_0 = \frac{1}{T}\int_T f(t)\,dt \text{ and } c_k = \frac{2}{T}\int_T f(t)\cos(k\omega_0 t)\,dt \text{ and } d_k = \frac{2}{T}\int_T f(t)\sin(k\omega_0 t)\,dt$$

c_0, the "direct current" (DC) term, represents the average value of $f(t)$ over a single period.

Continuous-Time Fourier Series in Complex Exponential Form

$$f(t) = \sum_{k=-\infty}^{\infty} a_k e^{jk\omega_0 t} \text{ where } a_k = \frac{1}{T}\int_T f(t) e^{-jk\omega_0 t} dt \qquad \left(\text{e.g., } a_0 = \frac{1}{T}\int_T f(t) dt\right)$$

Frequency

Time and frequency are inversely proportional.

cyclical: $f_0 = \dfrac{1}{T}$ (cycles per second, or hertz) **angular:** $\omega_0 = 2\pi f_0 = \dfrac{2\pi}{T}$ (radians per second)

Complex Variables

Think of complex variables geometrically — as points in the complex plane.

$$z = \underbrace{\mathrm{Re}\{z\} + j\,\mathrm{Im}\{z\}}_{\text{rectangular}} = \underbrace{re^{j\phi}}_{\text{polar}} \text{ where } \underbrace{r = \sqrt{\mathrm{Re}\{z\}^2 + \mathrm{Im}\{z\}^2}}_{\text{magnitude of } z} \text{ and } \underbrace{\tan(\phi) = \frac{\mathrm{Im}\{z\}}{\mathrm{Re}\{z\}}}_{\text{angle or phase of } z}$$

Euler's Formula

Euler's formula relates the rectangular-coordinate and polar-coordinate descriptions of complex variables.

$$e^{j\theta} = \cos(\theta) + j\sin(\theta) \qquad \cos(\theta) = \mathrm{Re}\{e^{j\theta}\} = \frac{e^{j\theta} + e^{-j\theta}}{2} \qquad \sin(\theta) = \mathrm{Im}\{e^{j\theta}\} = \frac{e^{j\theta} - e^{-j\theta}}{2j}$$

Sampling and Aliasing

From Continuous to Discrete

We refer to the process of discretizing time or space as *sampling.*

$$x[n] = x(n\Delta) \text{ where } n \in \mathbb{Z} \text{ and } \Delta \text{ denotes the sampling period, sampling interval, or time step}$$

We refer to the process of discretizing amplitude as *quantization.* (e.g., rounding)

$$\hat{x}[n] = Q\{x[n]\} \text{ where } Q\{\cdot\} \text{ denotes a quantization operator}$$

$$\text{e.g., } Q_\Delta\{x[n]\} = \Delta \left\lfloor \frac{x[n]}{\Delta} + \frac{1}{2} \right\rfloor \text{ for some constant } \Delta > 0, \text{ where } \lfloor\cdot\rfloor \text{ denotes the floor function}$$

Digital signals are discrete in both time and amplitude. Digital systems such as laptops process digital signals. *Discrete-time* signals are discrete in time — but not necessarily in amplitude. In 6.300, we won't study quantization in great depth. We'll focus on *discrete-time signal processing.*

Sampling and Aliasing

We *sample* a continuous-time signal $x(t)$ every Δ seconds to obtain a discrete-time signal $x[n]$.

$$x[n] = x(n\Delta) \text{ where } n \in \mathbb{Z} \text{ and } \Delta \text{ denotes the sampling period, sampling interval, or time step}$$

Note that $x[n]$ is a function of the integer n, which is enclosed in square brackets. In contrast, $x(t)$ is a function of the real variable t, which is enclosed in parentheses. With this notation, $x[n] \neq x(n)$ in general — when you write $x(n)$, you're implicitly saying that $\Delta = 1$.

Aliasing

Sampling involves throwing away information. If we don't sample frequently enough, the information within our signal will be distorted: Frequencies will "fold in" on each other.

Nyquist-Shannon sampling theorem: Let $f_{\max}$ denote the highest frequency in $x(t)$. The minimum sampling rate that prevents aliasing is $2f_{\max}$ — twice the highest frequency in $x(t)$.

Frequencies

Always keep the dimensions of quantities in mind.

$$\textbf{CT cyclical: } f = \frac{1}{T} \qquad \textbf{CT angular: } \omega = 2\pi f = \frac{2\pi}{T} \qquad \textbf{DT angular: } \Omega = \frac{2\pi f}{f_s} = \frac{\omega}{f_s} = \frac{2\pi}{N}$$

The argument to a trigonometric or exponential function must be expressed in radians.

- units$\{2\pi ft\}$ = (radians/cycle) × (cycles/second) × (seconds) = radians
- units$\{\omega t\}$ = (radians/second) × (seconds) = radians
- units$\{\Omega n\}$ = (radians/sample) × (samples) = radians

Discrete-Time Fourier Series

Fourier Series Formulæ

Continuous-Time Fourier Series (CTFS)

$f(t)$ is a T-periodic function with fundamental angular frequency $\omega_0 = 2\pi/T$.

$$\textbf{Synthesis: } f(t) = \sum_{k=-\infty}^{\infty} a_k e^{jk\omega_0 t} \qquad \textbf{Analysis: } a_k = \frac{1}{T}\int_T f(t)e^{-jk\omega_0 t}$$

Discrete-Time Fourier Series (DTFS)

$f[n]$ is an N-periodic sequence with fundamental angular frequency $\Omega_0 = 2\pi/N$.

$$\textbf{Synthesis: } f[n] = \sum_{k=\langle N\rangle} a_k e^{jk\Omega_0 n} \qquad \textbf{Analysis: } a_k = \frac{1}{N}\sum_{n=\langle N\rangle} f[n]e^{-jk\Omega_0 n}$$

A Few Properties of Fourier Series

Here are a few properties of Fourier series that we'll use often in this class. We'll learn more over time.

Linearity

$$f_1(t) \iff F_1[k] \qquad f_2(t) \iff F_2[k] \qquad f(t) = \alpha f_1(t) + \beta f_2(t) \iff F[k] = \alpha F_1[k] + \beta F_2[k]$$

$$f_1[n] \iff F_1[k] \qquad f_2[n] \iff F_2[k] \qquad f[n] = \alpha f_1[n] + \beta f_2[n] \iff F[k] = \alpha F_1[k] + \beta F_2[k]$$

Time Shift

$$f(t) \iff F[k] \qquad f(t-t_0) \iff F[k]e^{-jk\omega_0 t_0} = |F[k]|e^{j(\angle F[k] - k\omega_0 t_0)}$$

$$f[n] \iff F[k] \qquad f[n-n_0] \iff F[k]e^{-jk\Omega_0 n_0} = |F[k]|e^{j(\angle F[k] - k\Omega_0 n_0)}$$

Time Flip

$$f(t) \iff F[k] \qquad f(-t) \iff F[-k]$$

$$f[n] \iff F[k] \qquad f[-n] \iff F[-k]$$

Conjugate Symmetry (Hermitian Symmetry)

Real-valued signals have conjugate-symmetric Fourier series coefficients.

real-valued $f(t) \iff F[k]$ such that $F^*[k] = F[-k]$ where * denotes complex conjugation

real-valued $f[n] \iff F[k]$ such that $F^*[k] = F[-k]$ where * denotes complex conjugation

Continuous-Time Fourier Transform

Continuous-Time Fourier Transform (CTFT)

The continuous-time Fourier transform may be conceptualized as the continuum limit of a continuous-time Fourier series. Infinitely-many discrete harmonics $k\omega_0$ cluster infinitely-close together to form a continuous frequency spectrum: $k\omega_0$ (function of integer k) $\mapsto \omega$ (function of real-valued ω).

Synthesis: $x(t) = \dfrac{1}{2\pi}\displaystyle\int_{-\infty}^{\infty} X(\omega)e^{j\omega t}d\omega$ **Analysis:** $X(\omega) = \displaystyle\int_{-\infty}^{\infty} x(t)e^{-j\omega t}dt$

Fourier Transform Pairs

$$\delta(t) \iff 1 \text{ (for all } \omega) \qquad 1 \text{ (for all } t) \iff 2\pi\delta(\omega)$$

$$\delta(t-t_0) \iff e^{-j\omega t_0} \qquad e^{j\omega_0 t} \iff 2\pi\delta(\omega-\omega_0)$$

Fourier Transform Properties

You fill this in! Many properties that we've seen in the context of Fourier series still hold true.

Fourier Series vs. Fourier Transform for Periodic Signals

Series: $f(t) \iff F[k]$ **Transform:** $f(t) \iff \displaystyle\sum_k \underbrace{2\pi F[k]\delta(\omega-k\omega_0)}_{\text{impulses at harmonics}}$

Discrete-Time Fourier Transform

Discrete-Time Fourier Transform (DTFT)

The discrete-time Fourier transform is the discrete-time analogue of the continuous-time Fourier transform — a Fourier transform for discrete-time signals. Infinitely-many discrete harmonics $k\Omega_0$ cluster infinitely-close together to form a continuous frequency spectrum: $k\Omega_0 \mapsto \Omega$.

Synthesis: $x[n] = \frac{1}{2\pi} \int_{2\pi} X(\Omega) e^{j\Omega n} d\Omega$ **Analysis:** $X(\Omega) = \sum_{n=-\infty}^{\infty} x[n] e^{-j\Omega n}$

Discrete-Time Fourier Transform Pairs

$$\delta[n] \iff 1 \text{ (for all } \Omega) \qquad 1 \text{ (for all } n) \iff 2\pi\delta(\Omega \bmod 2\pi)$$

$$\delta[n-n_0] \iff e^{-j\Omega n_0} \qquad e^{j\Omega_0 n} \iff 2\pi\delta(\Omega - \Omega_0 \bmod 2\pi)$$

Discrete-Time Fourier Transform Properties

You fill this in! Many properties that we've seen in the context of Fourier series still hold true.
After you solve the problems on the other side of this sheet, write down the properties you just derived.

Fourier Series vs. Fourier Transform for Periodic Signals

Series: $f[n] \iff F[k]$ **Transform:** $f[n] \iff \sum_k \underbrace{2\pi F[k]\delta(\Omega - k\Omega_0)}_{\text{impulses at harmonics}}$

LTI Systems

Linear Time-Invariant Systems

A linear time-invariant (LTI) system is a system which is both linear and time-invariant.

$$x(t) \to \boxed{\textbf{LTI}} \to y(t) \qquad\qquad x[n] \to \boxed{\textbf{LTI}} \to y[n]$$

We will look at three equivalent representations of LTI systems in this class.

- differential equation (CT) or difference equation (DT)
- impulse response (CT) or unit-sample response (DT)
- frequency response

Linearity

A system is linear if and only if it is both additive and homogeneous.

Additivity

The response to a sum of inputs is a sum of the respective outputs.

$$\text{If } x_1[n] \to \boxed{\text{additive system}} \to y_1[n] \text{ and } x_2[n] \to \boxed{\text{additive system}} \to y_2[n],$$

$$\text{then } x[n] = x_1[n] + x_2[n] \to \boxed{\text{additive system}} \to y[n] = y_1[n] + y_2[n].$$

Homogeneity

Scaling the input correspondingly scales the output.

$$\text{If } x[n] \to \boxed{\text{homogeneous system}} \to y[n], \text{ then } \alpha \cdot x[n] \to \boxed{\text{homogeneous system}} \to \alpha \cdot y[n].$$

Linearity

The response to a sum of scaled inputs is a sum of the respective scaled outputs.

$$\text{If } x_1[n] \to \boxed{\text{linear system}} \to y_1[n] \text{ and } x_2[n] \to \boxed{\text{linear system}} \to y_2[n],$$

$$\text{then } x[n] = \alpha \cdot x_1[n] + \beta \cdot x_2[n] \to \boxed{\text{linear system}} \to y[n] = \alpha \cdot y_1[n] + \beta \cdot y_2[n].$$

Time-Invariance

Delaying (or advancing) the input delays (or advances) the output by the exact same duration of time.

$$\text{If } x[n] \to \boxed{\text{time-invariant system}} \to y[n], \text{ then } x[n-n_0] \to \boxed{\text{time-invariant system}} \to y[n-n_0].$$

Convolution and Unit-Sample Response

Convolution

Convolution is a mathematical operation. It is associative, commutative, and distributive.
In discrete time, convolution takes the form of a summation.

$$\textbf{Convolution Sum: } (x * h)[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k] = (h * x)[n] = \sum_{k=-\infty}^{\infty} h[k]x[n-k]$$

Compute the result using superposition. Use a table or sketch a plot to keep track of the values.

n	$=$	$\boxed{\mathbf{0}}$	$\boxed{\mathbf{1}}$	$\boxed{\mathbf{2}}$	$\boxed{\mathbf{3}}$	$\boxed{\mathbf{4}}$	$\boxed{\mathbf{5}}$	$\boxed{\mathbf{6}}$	$\boxed{\mathbf{7}}$	$\boxed{\mathbf{8}}$	$\boxed{\mathbf{9}}$
$x[n]$	$=$	1	1	1	1	0	0	0	0	0	0
$h[n]$	$=$	1	2	3	0	0	0	0	0	0	0

n	$=$	$\boxed{\mathbf{0}}$	$\boxed{\mathbf{1}}$	$\boxed{\mathbf{2}}$	$\boxed{\mathbf{3}}$	$\boxed{\mathbf{4}}$	$\boxed{\mathbf{5}}$	$\boxed{\mathbf{6}}$	$\boxed{\mathbf{7}}$	$\boxed{\mathbf{8}}$	$\boxed{\mathbf{9}}$
$h[0]\,x[n-0]$	$=$	1	1	1	1	0	0	0	0	0	0
$h[1]\,x[n-1]$	$=$	0	2	2	2	2	0	0	0	0	0
$h[2]\,x[n-2]$	$=$	0	0	3	3	3	3	0	0	0	0
$(x * h)[n]$	$=$	$\boxed{\mathbf{1}}$	$\boxed{\mathbf{3}}$	$\boxed{\mathbf{6}}$	$\boxed{\mathbf{6}}$	$\boxed{\mathbf{5}}$	$\boxed{\mathbf{3}}$	$\boxed{\mathbf{0}}$	$\boxed{\mathbf{0}}$	$\boxed{\mathbf{0}}$	$\boxed{\mathbf{0}}$

In continuous time, convolution takes the form of an integral.

$$\textbf{Convolution Integral: } (x * h)(t) = \int_{-\infty}^{\infty} x(\tau)h(t-\tau)d\tau = (h * x)(t) = \int_{-\infty}^{\infty} h(\tau)x(t-\tau)d\tau$$

It's probably simplest to evaluate the integral piecewise over several intervals.

Unit-Sample Response

$$\text{unit-sample signal } \delta[n] \rightarrow \boxed{\text{DT LTI System}} \rightarrow \text{unit-sample response } h[n] = \sum_{k=-\infty}^{\infty} h[k]\delta[n-k]$$

$$\text{e.g., } y[n] = \tfrac{1}{2}x[n] + \tfrac{1}{2}x[n-1] \text{ with initial rest conditions } \Longleftrightarrow h[n] = \tfrac{1}{2}\delta[n] + \tfrac{1}{2}\delta[n-1]$$

$$\text{e.g., } y[n] = \tfrac{1}{2}y[n-1] + x[n] \text{ with initial rest conditions } \Longleftrightarrow h[n] = \left(\tfrac{1}{2}\right)^n u[n]$$

Impulse Response

$$\text{unit-impulse signal } \delta(t) \rightarrow \boxed{\text{CT LTI System}} \rightarrow \text{impulse response } h(t) = \int_{-\infty}^{\infty} h(\tau)\delta(t-\tau)d\tau$$

Frequency Response and Filtering

Frequency Response

Complex exponentials are eigenfunctions of LTI systems.

$$e^{j\omega t} \to \boxed{\text{LTI}} \to H(\omega)e^{j\omega t} = |H(\omega)|e^{j(\omega t + \angle H(\omega))} \qquad X(\omega) \to \boxed{\text{LTI}} \to Y(\omega) = H(\omega)X(\omega)$$

$$e^{j\Omega n} \to \boxed{\text{LTI}} \to H(\Omega)e^{j\Omega n} = |H(\Omega)|e^{j(\Omega n + \angle H(\Omega))} \qquad X(\Omega) \to \boxed{\text{LTI}} \to Y(\Omega) = H(\Omega)X(\Omega)$$

We may characterize an LTI system as a *filter* that shapes a signal's spectrum.

Here is the physical interpretation: If the input to an LTI system is a sinusoid, the output of the LTI system is a scaled (i.e., amplified or attenuated) and phase-shifted sinusoid.

$$\cos(\omega t) \to \boxed{\text{LTI}} \to |H(\omega)|\cos(\omega t + \angle H(\omega))$$

$$\cos(\Omega n) \to \boxed{\text{LTI}} \to |H(\Omega)|\cos(\Omega n + \angle H(\Omega))$$

Convolution Theorem

Convolution in time corresponds to multiplication in frequency.
Convolution in frequency corresponds to multiplication in time.

$$(x * h)(t) \iff X(\omega)H(\omega) \qquad x(t)h(t) \iff \tfrac{1}{2\pi}(X * H)(\omega)$$

$$(x * h)[n] \iff X(\Omega)H(\Omega) \qquad x[n]h[n] \iff \tfrac{1}{2\pi}(X * H)(\Omega)$$

Difference Equation vs. Impulse Response vs. Frequency Response

Consider the LTI system represented by the following difference equation.

$$\sum_{k=-\infty}^{\infty} a_k y[n-k] = \sum_{k=-\infty}^{\infty} b_k x[n-k]$$

Computing the DTFT yields the frequency response.

$$\sum_{k=-\infty}^{\infty} a_k e^{-jk\Omega} Y(\Omega) = \sum_{k=-\infty}^{\infty} b_k e^{-jk\Omega} X(\Omega) \iff H(\Omega) = \frac{Y(\Omega)}{X(\Omega)} = \frac{\sum_{k=-\infty}^{\infty} b_k e^{-jk\Omega}}{\sum_{k=-\infty}^{\infty} a_k e^{-jk\Omega}}$$

The impulse response is given by the inverse DTFT of the frequency response. $H(\Omega)$ is a rational polynomial that one can decompose into a sum of simpler rational polynomials (e.g., via partial fractions) with simpler inverse DTFTs.

$$h[n] = \frac{1}{2\pi}\int_{2\pi} H(\Omega)e^{j\Omega n}d\Omega$$

$$\text{e.g., } h[n] = \frac{1}{2\pi}\int_{2\pi}\left(\frac{1}{1-\frac{1}{2}e^{-j\Omega}} + \frac{1}{1+\frac{1}{3}e^{-j\Omega}}\right)e^{j\Omega n}d\Omega = \left(\tfrac{1}{2}\right)^n u[n] + \left(-\tfrac{1}{3}\right)^n u[n]$$

Communications Systems

Communications Systems

A key problem in the design of any communications system is matching characteristics of the signal to those of the channel medium. Much of the current research in communications focuses on modifying signals to better accommodate constraints imposed by the channel medium. In lecture, we looked at simple channel-medium-matching strategies based on modulation. Modulation underlies virtually all matching schemes.

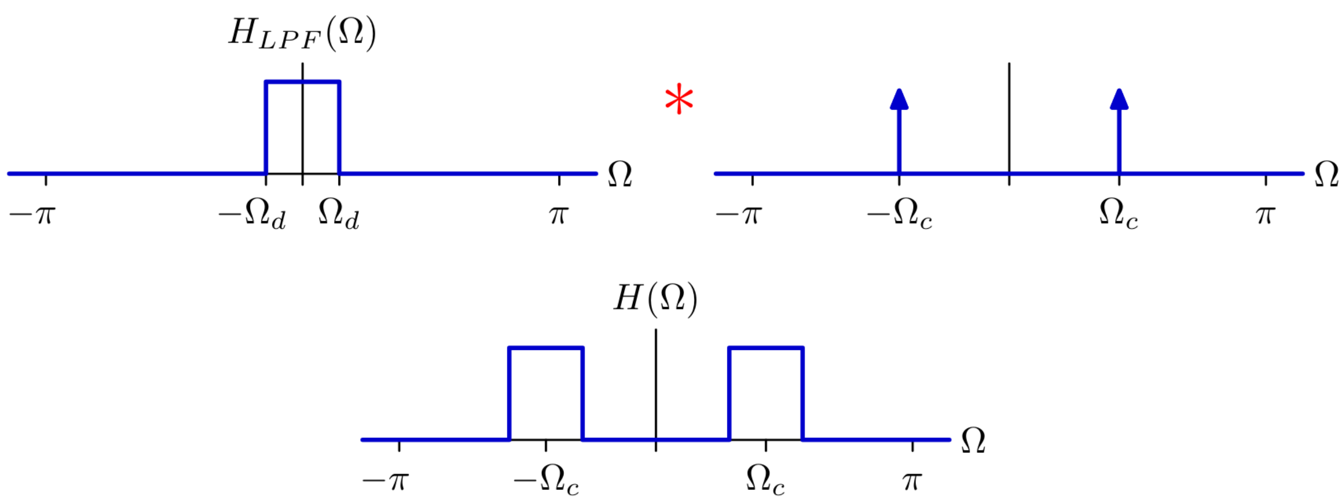
Modulation Property (Frequency Shift Property)

Multiplying a signal $x(t)$ by a time-varying complex exponential shifts the transform $X(\omega)$ in frequency.

$$x(t)e^{j\omega_c t} \iff \tfrac{1}{2\pi}\left(X(\omega) * 2\pi\delta(\omega-\omega_c)\right) = X(\omega-\omega_c)$$

Just as convolution in time corresponds to multiplication in frequency, multiplication in time corresponds to convolution in frequency.

Convolution with Impulses



Sinusoidal Modulation

We examined amplitude modulation (AM) in class. Perhaps you've heard of frequency modulation (FM) or phase modulation (PM) — but you don't need to know these for this class, per se.

$$y(t) = A\cos(\omega t + \phi)$$

- amplitude modulation (AM) — time-varying amplitude $A = A(t)$
- frequency modulation (FM) — time-varying frequency $\omega = \omega(t)$
- phase modulation (PM) — time-varying phase $\phi = \phi(t)$

Discrete Fourier Transform

Discrete Fourier Transform

We may represent a periodic discrete-time signal $x[n] = x[n+N]$ as a finite-length sum of harmonically-related complex exponentials with a discrete-time Fourier series. Periodicity is rather restrictive, as many real-world signals of interest are not periodic. To be periodic, a signal must repeat forever!

The discrete-time Fourier transform is a Fourier representation for aperiodic discrete-time signals $x[n]$, but the analysis formula entails an infinite-length sum, and the spectrum $X(\Omega)$ — a function of the continuous variable Ω — is continuous. While a useful theoretical tool, the DTFT is also somewhat impractical.

In search of a Fourier representation that is amenable for digital computation, we turn to the DFT. You might call the DFT a discrete-time, discrete-frequency Fourier transform.

- finite-length signals — windowing $x_w[n] = x[n]w[n]$
- discrete in time — time indexed by integer n
- discrete in frequency — frequency indexed by integer k

Analysis: $$X[k] = \frac{1}{N}\sum_{n=0}^{N-1} x[n]e^{-jk\frac{2\pi}{N}n}$$

Synthesis: $$x[n] = \sum_{k=0}^{N-1} X[k]e^{jk\frac{2\pi}{N}n}$$

Note: There is no general consensus on where to place the $1/N$ factor in the equations that define the DFT. In 6.300, we place the $1/N$ factor in the analysis equation, so that the DFT and DTFS analysis equations match. In both cases, then, the DC term $X[0]$ is the average value of $x[n]$ over a length-N interval. Some authors and numerical packages (e.g., MATLAB) place the $1/N$ factor in the synthesis equation instead, however. Other authors and numerical packages even put a factor of $1/\sqrt{N}$ in both the analysis and synthesis equations to make the DFT a unitary transformation.

Relation to Discrete-Time Fourier Series (DTFS)

The length-N DFT is equivalent to the DTFS of an N-periodic extension of the windowed signal $x_w[n] = x[n]w[n]$. From this perspective, sharp discontinuities in the periodic extension of $x_w[n]$ lead to spurious high-frequency content spread across the spectrum of $x_w[n]$.

$$X[k] = \frac{1}{N}\sum_{n=0}^{N-1} x_w[(n \bmod N)]e^{-j2\pi kn/N}$$

Relation to Discrete-Time Fourier Transform (DTFT)

The DFT returns N scaled, equally-spaced samples of the DTFT over the interval $[0, 2\pi)$. Increasing the DFT length N yields more samples of the DTFT, which are spaced more closely together.

$$X[k] = \tfrac{1}{N}X_w\left(\tfrac{2\pi}{N}k\right)$$

cyclical frequency resolution: $\dfrac{f_s}{N}$ hertz

angular frequency resolution: $\dfrac{2\pi}{N}$ radians

Circular Convolution

Circular Convolution

Multiplication of N-point DFTs corresponds to time-domain circular convolution.

$$\tfrac{1}{N}(x \circledast h)[n] \iff X_N[k]H_N[k] \qquad\qquad (x \circledast h)[n] = N\,\mathrm{DFT}_N^{-1}\{X_N[k]H_N[k]\}$$

$$(x \circledast h)[n] = \sum_{m=0}^{N-1} x[m]h\Big[((n-m) \bmod N)\Big] = (h \circledast x)[n] = \sum_{m=0}^{N-1} h[m]x\Big[((n-m) \bmod N)\Big]$$

Computing a circular convolution may seem complicated, but it is really simple.

- Perform linear convolution — the convolution you're already familiar with.
- Wrap the result of linear convolution into a length-N interval.
  This is *time-aliasing*, which is the time-domain analogue of the more familiar frequency-aliasing.
- Periodically extend every N samples.

Compute the result using superposition. Use a table or sketch a plot to keep track of the values.

n	=	0	1	2	3	4	5	6	7	8	9
$x[n]$	=	1	1	1	1	0	0	0	0	0	0
$h[n]$	=	1	2	3	0	0	0	0	0	0	0

n	=	0	1	2	3	4	5	6	7	8	9
$h[0]\,x[n-0]$	=	1	1	1	1	0	0	0	0	0	0
$h[1]\,x[n-1]$	=	0	2	2	2	2	0	0	0	0	0
$h[2]\,x[n-2]$	=	0	0	3	3	3	3	0	0	0	0
$(x * h)[n]$	=	1	3	6	6	5	3	0	0	0	0
$(x \circledast h)_6[n]$	=	1	3	6	6	5	3	1	3	6	6
$(x \circledast h)_5[n]$	=	4	3	6	6	5	4	3	6	6	5
$(x \circledast h)_4[n]$	=	6	6	6	6	6	6	6	6	6	6
$(x \circledast h)_3[n]$	=	7	8	9	7	8	9	7	8	9	7
$(x \circledast h)_2[n]$	=	12	12	12	12	12	12	12	12	12	12
$(x \circledast h)_1[n]$	=	24	24	24	24	24	24	24	24	24	24

Here's the take-home message for today: You need to get comfortable with performing convolution from this moment on. If you can't perform convolution, you can't perform circular convolution. We will perform convolution for the rest of the class. There will be many convolution problems in the homework and on the quizzes. Going forward, you cannot succeed in this class if you cannot perform convolution.

On the bright side, once you can perform convolution, circular convolution is easy!

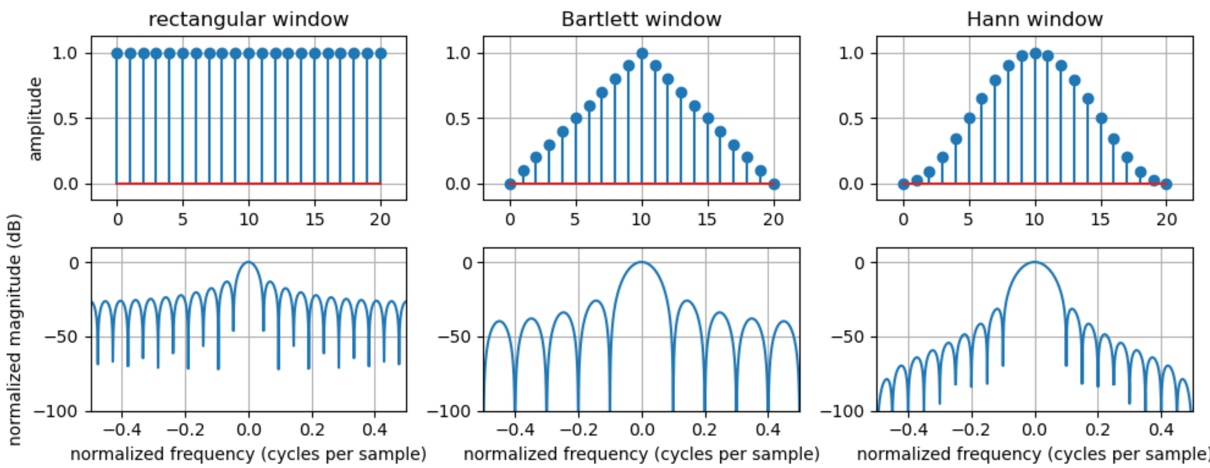
Window Functions

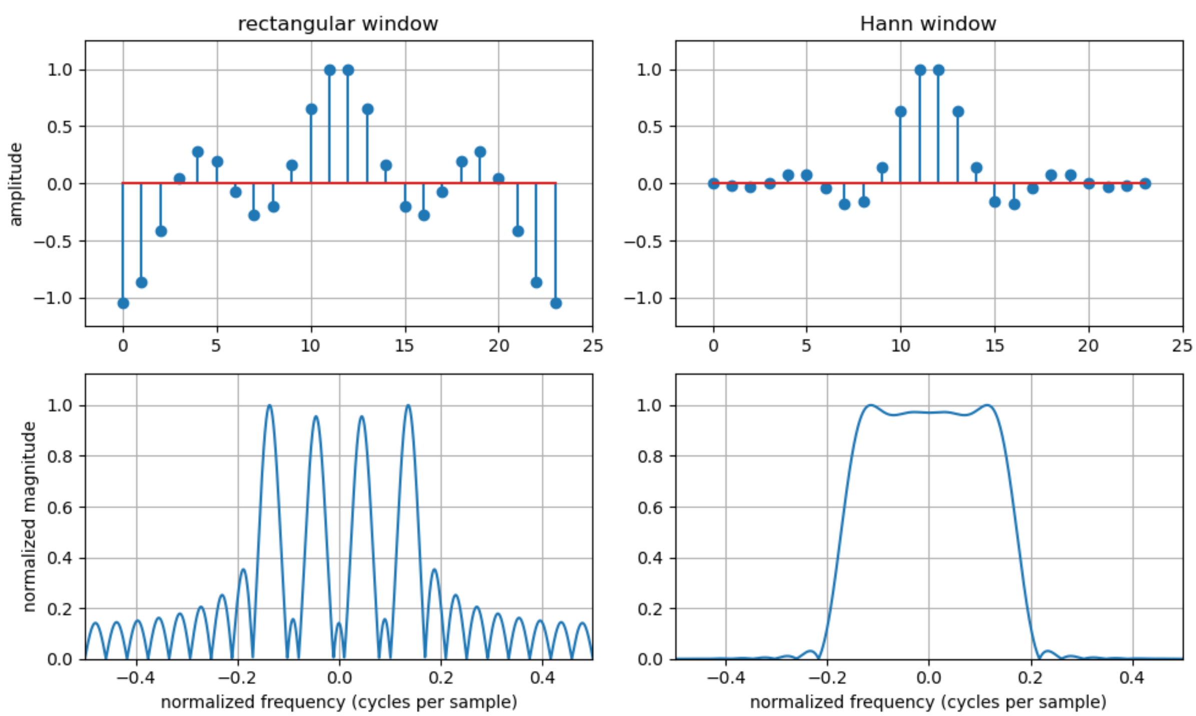
A defining feature of the DFT is that signals we analyze have a finite length: N samples. The number of samples plays a key role in determining the trade-off between time resolution and frequency resolution. We can restrict a signal $x[n]$ to have length N by multiplying pointwise by a window function $w[n]$.

Windowing

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

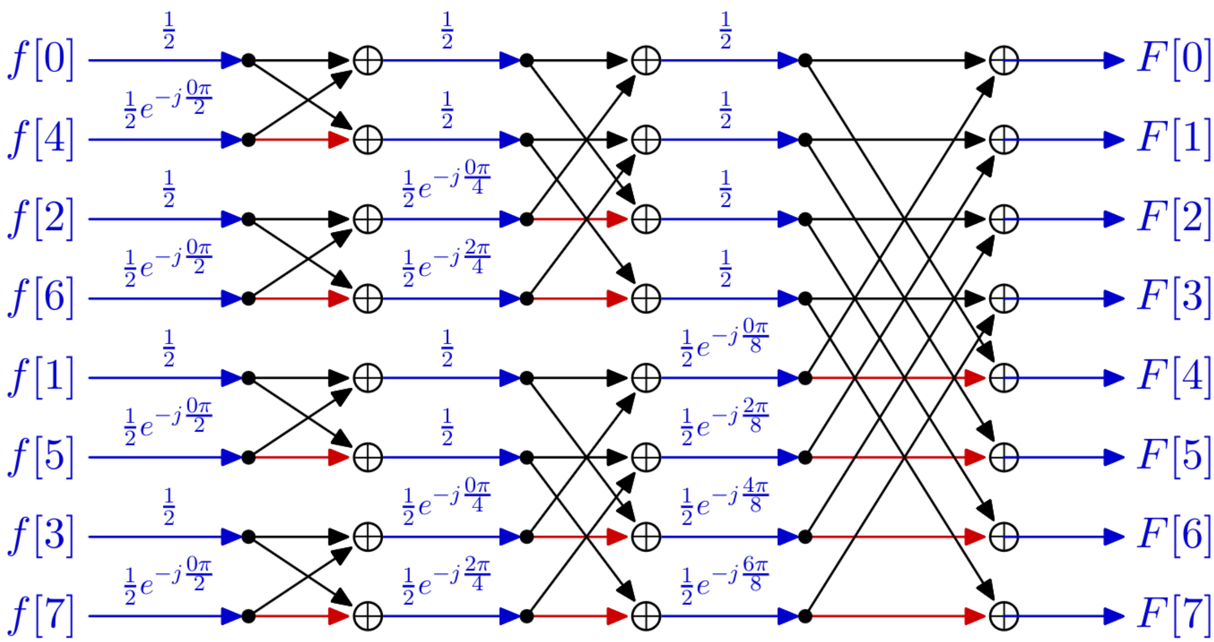
$$x_w[n] = x[n]w[n] \iff X_w(\Omega) = \frac{1}{2\pi}\big(X * W\big)(\Omega)$$





Fast Fourier Transform

Schematic: Decimation-in-Time FFT Algorithm



Inverse FFT

In lecture, we discussed FFT algorithms for computing the DFT. Because the analysis and synthesis equations for the DFT are so similar, we need only make a few tweaks to derive inverse FFT algorithms for the computing the inverse DFT. How would you change the code below to compute the inverse DFT?

```
from math import e, pi
def FFT(x):
    N = len(x)
    if N == 1:
        return x
    if N != 2 * N//2:
        print('N must be a power of 2')
        exit(1)
    xe = x[::2]
    xo = x[1::2]
    Xe = FFT(xe)
    Xo = FFT(xo)
    X = []
    for k in range(N//2):
        X.append((Xe[k] + e**(-2j*pi*k/N)*Xo[k]) / 2)
    for k in range(N//2):
        X.append((Xe[k] - e**(-2j*pi*k/N)*Xo[k]) / 2)
    return X
```

Reference

Transform Pairs: Continuous-Time Fourier Transform

$\delta(t) \Longleftrightarrow 1$ $\quad$ $\delta(t-t_0) \Longleftrightarrow e^{-j\omega t_0}$ $\quad$ $1 \Longleftrightarrow 2\pi\delta(\omega)$ $\quad$ $e^{j\omega_0 t} \Longleftrightarrow 2\pi\delta(\omega-\omega_0)$

Transform Pairs: Discrete-Time Fourier Transform

$\delta[n] \Longleftrightarrow 1$ $\quad$ $\delta[n-n_0] \Longleftrightarrow e^{-j\Omega n_0}$ $\quad$ $1 \Longleftrightarrow 2\pi\delta(\Omega \bmod 2\pi)$ $\quad$ $e^{j\Omega_0 n} \Longleftrightarrow 2\pi\delta((\Omega-\Omega_0) \bmod 2\pi)$

Transform Pairs: Discrete Fourier Transform

$\delta[n] \Longleftrightarrow \frac{1}{N}$ $\quad$ $\delta[n-n_0] \Longleftrightarrow \frac{1}{N}e^{-jk\frac{2\pi}{N}n_0}$ $\quad$ $1 \Longleftrightarrow \delta[k \bmod N]$ $\quad$ $e^{jk_0\frac{2\pi}{N}n} \Longleftrightarrow \delta\big[(k-k_0) \bmod N\big]$

Fourier Properties

$c_1x_1[n]+c_2x_2[n] \Longleftrightarrow c_1X_1(\Omega)+c_2X_2(\Omega)$	$(x_1 * x_2)[n] \Longleftrightarrow X_1(\Omega)X_2(\Omega)$
$\frac{1}{N}(x_1 \circledast x_2)[n] \Longleftrightarrow X_1[k]X_2[k]$	$x_1[n]x_2[n] \Longleftrightarrow \frac{1}{2\pi}(X_1 * X_2)(\Omega)$
$x[-n] \Longleftrightarrow X(-\Omega)$	$x[nM] \Longleftrightarrow X(\frac{\Omega}{M})$
$x[n-n_0] \Longleftrightarrow e^{-j\Omega n_0}X(\Omega)$	$e^{j\Omega_0 n}x[n] \Longleftrightarrow X(\Omega-\Omega_0)$
$n\,x[n] \Longleftrightarrow j\frac{d}{d\Omega}X(\Omega)$	$\frac{d}{dt}x(t) \Longleftrightarrow j\omega X(\omega)$