Last time, we introduced the notion of a **system** (or **filter**). Many applications of signal processing can be thought of as systems that convert an input signal into an output signal:

![Signal system diagram]

**Examples:**
- Audio enhancement: equalization, noise reduction, reverberation, echo cancellation, pitch shift (auto-tune)
- Image enhancement: smoothing, edge enhancement, unsharp masking, feature detection
- Video enhancement: image stabilization, motion magnification

**LTI Systems**

In 6.003, we will focus our attention on **linear, time invariant** systems.

A system is linear and time invariant if it can be expressed in terms of a linear difference equation with constant coefficients. General form:

\[
\sum_{m} c_m y[n - m] = \sum_{k} d_k x[n - k]
\]

**Additivity:** output of sum is sum of outputs
**Homogeneity:** scaling an input scales its output
**Time invariance:** delaying an input delays its output
**Unit Sample Response**

If a system is linear and time-invariant, its input-output relation is completely specified by its unit sample response $h[n]$. The unit-sample response $h[n]$ is the output of the system when the input is the unit-sample signal $\delta[n]$:

$$
\delta[n] \xrightarrow{\text{system}} h[n]
$$

The output for more complicated inputs can be computed by summing scaled and shifted versions of the unit-sample response.

**Superposition**

In general, we can represent a signal as a sum of scaled, shifted deltas:

$$
x[n] = \sum_{m=-\infty}^{\infty} x[m] \delta[n-m] = \ldots + x[-1] \delta[n+1] + x[0] \delta[n] + x[1] \delta[n-1] + x[2] \delta[n-2] + \ldots
$$

If $h[\cdot]$ is the unit sample response of an LTI system, then the output of that system in response to this arbitrary input $x[n]$ can be viewed as a sum of scaled, shifted unit sample responses:

$$
y[n] = \sum_{m=-\infty}^{\infty} x[m] h[n-m] = \ldots + x[-1] h[n+1] + x[0] h[n] + x[1] h[n-1] + x[2] h[n-2] + \ldots
$$

**Convolution: Summary**

Unit-sample response $h[\cdot]$ is a complete description of an LTI system:

$$
x[n] \xrightarrow{h[\cdot]} y[n]
$$

Given $h[n]$ one can compute the response $y[n]$ to any input $x[n]$:

$$
y[n] = (x * h)[n] \equiv \sum_{m=-\infty}^{\infty} x[m] h[n-m]
$$
**Frequency Representation of Convolution**

Let $y[n] = (x * h)[n]$. Find $Y(\Omega)$.

\[
Y(\Omega) = \sum_{n=-\infty}^{\infty} (h * x)[n]e^{-j\Omega n}
\]
\[
= \sum_{n=-\infty}^{\infty} \left( \sum_{m=-\infty}^{\infty} h[m]x[n-m] \right) e^{-j\Omega n}
\]
\[
= \sum_{m=-\infty}^{\infty} h[m] \sum_{l=-\infty}^{\infty} x[l]e^{-j\Omega(l+m)}
\]
\[
= \sum_{m=-\infty}^{\infty} h[m]e^{-j\Omega m} \sum_{l=-\infty}^{\infty} x[l]e^{-j\Omega l}
\]
\[
= H(\Omega)X(\Omega)
\]

Convolution in time is multiplication in frequency!

---

**Filtering**

We can view filtering in both the time and frequency domains:

**Time Domain:**

\[
x[n] \rightarrow h[n] \rightarrow y[n] = (h * x)[n]
\]

**Frequency Domain:**

\[
X(\Omega) \rightarrow H(\Omega) \rightarrow Y(\Omega) = H(\Omega)X(\Omega)
\]

Each frequency component of input $X(\Omega)$ is scaled by a factor $H(\Omega)$, which can be possibly complex.

The system is completely described by the set of scale factors $H(\Omega)$, which we refer to as the **frequency response** of the system.

---

**Today**

Today: effects of filtering in the frequency domain.
Check Yourself!

Consider the system described by:

\[ y[n] = \frac{x[n-1] + x[n] + x[n+1]}{3} \]

Sketch this system's response to the following input:

\[ x[n] = \delta[n-1] + 2\delta[n-2] + 3\delta[n-3] \]

Surprise!
Eigenfunctions and Eigenvalues

If the output signal is a scalar multiple of the input signal, we refer to the signal as an **eigenfunction**, and the multiplier as the **eigenvalue**.

$$x[n] \xrightarrow{\text{system}} \lambda x[n]$$

Complex Exponentials

Complex exponentials are eigenfunctions of LTI systems. If $h[n]$ is a system’s unit sample response, and $x[n] = e^{j\Omega n}$, then the system’s output is:

$$y[n] = \sum_{m=-\infty}^{\infty} h[m]x[n-m]$$
$$= \sum_{m=-\infty}^{\infty} h[m]e^{j\Omega(n-m)}$$
$$= \sum_{m=-\infty}^{\infty} h[m]e^{j\Omega n}e^{-j\Omega m}$$
$$= e^{j\Omega n}\sum_{m=-\infty}^{\infty} h[m]e^{-j\Omega m}$$
$$= e^{j\Omega n}H(\Omega)$$

The eigenvalues $H(\Omega)$ are generally complex-valued, and so affect both the amplitude and phase of the output:

$$e^{j\Omega n} \xrightarrow{\text{system}} |H(\Omega)|e^{j\Omega n}e^{-j\Omega n}H(\Omega)$$

Notes
Response to Eternal Sinusoids

If \( h[n] \) is purely real, we have \( H(\Omega) = H^*(-\Omega) \).
Consider \( x[n] = \cos(\Omega n) \) (for all \( n \)), which can be written as:
\[
x[n] = \frac{1}{2} (e^{j\Omega n} + e^{-j\Omega n})
\]
Then:
\[
y[n] = \frac{1}{2} (H(\Omega)e^{j\Omega n} + H(-\Omega)e^{-j\Omega n})
\]
\[
= \text{Re} \left( H(\Omega)e^{j\Omega n} \right)
\]
\[
= \text{Re} \left( H(\Omega)e^{j(\Omega n + \angle H(\Omega))} \right)
\]
\[
y[n] = |H(\Omega)| \cos(\Omega n + \angle H(\Omega))
\]
Output in response to a pure cosine is a (scaled, shifted) pure cosine at the same frequency!

Check Yourself!

Consider the moving average filter from before:
\[
y[n] = \frac{x[n-1] + x[n] + x[n+1]}{3}
\]
What is this system’s frequency response?
Does this agree with the input/output relationship we see?

Example: Bass Boost

Let’s take a look at taking a song and boosting its low frequencies.

First subproblem: isolate the low frequencies (by attenuating high frequencies).
**Check Yourself!**

Let \( x[\cdot] \) be our original input, and \( (x * h_L)[\cdot] \) represent a low-passed version of \( x[\cdot] \).

We want to use a single convolution to produce a new signal with the high frequencies still present, but the low frequencies amplified:

\[
(x * h_B)[n] = x[n] + c(x * h_L)[n]
\]

Find an expression for \( h_B[n] \).

---

**“Ideal” Filters**

So far, we have seen one example of designing a filter to accomplish a task, based on a desired frequency response. For the rest of the day, we will consider some examples of idealized filters, and think about their representations in the time and frequency domains.

---

**The “Ideal” Low-Pass Filter**

Consider a system characterized by the following purely real frequency response:

\[
H(\Omega) = \begin{cases} 1 & \text{for } |\Omega| < \Omega_c \\ 0 & \text{otherwise} \end{cases}
\]

Such a system is called a **low-pass filter**, because it allows low frequencies to pass through unmodified, while attenuating high frequencies.

We could apply this filter to a signal by multiplying the DTFT of that signal by the values above. But we could also apply the filter by operating in the time domain.
An “Ideal” Band-Pass Filter

Consider a different filter, which is designed to pass frequencies in a band centered around $\Omega_b$, which is not necessarily zero. Its frequency response is shown below:

If we wanted to apply this filter by operating in the time domain, what $h[n]$ would we convolve our input with?

Summary

Today, focused on frequency response.
- characterize system by responses to complex exponentials
- designed a filter to have a particular frequency response
- discussed examples of idealized filters, time and frequency representations