6.003: Signal Processing

Superposition and Convolution

\[ y[n] = (h \ast x)[n] = \sum_{m} h[m]x[n - m] \]
So far, we have focused our attention on signals and their mathematical representations. However, we are often also interested in *manipulating* signals. To this end, we introduce the notion of a **system** (or **filter**).

**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
Representations of Signals

Characterize systems by their input/output relationships.

**Difference Equation**: represent system by algebraic constraints on samples

**Convolution**: represent system by its unit sample response

**Filter**: represent system as amplification/attenuation of frequency components
**Linearity**

A system is linear if its response to a **weighted sum of inputs** is equal to the **weighted sum of its responses** to each of the inputs.

Given

\[
x_1[n] \rightarrow \text{system} \rightarrow y_1[n]
\]

and

\[
x_2[n] \rightarrow \text{system} \rightarrow y_2[n]
\]

the **system is linear** if

\[
\alpha x_1[n] + \beta x_2[n] \rightarrow \text{system} \rightarrow \alpha y_1[n] + \beta y_2[n]
\]

is true for all \( \alpha \) and \( \beta \) and all possible inputs.

A system is linear if it is both additive and homogeneous.
**Time-Invariance**

A system is time-invariant if delaying the input to the system simply delays the output by the same amount of time.

Given

\[ x[n] \rightarrow \text{system} \rightarrow y[n] \]

the **system is time invariant** if

\[ x[n-n_0] \rightarrow \text{system} \rightarrow y[n-n_0] \]

is true for all \( n_0 \) and for all possible inputs.
If a system is linear and time-invariant, its input-output relation is completely specified by the system’s **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 $\delta[n]$.

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

Consider the following signal:

\[ x[n] = \begin{cases} 
1 & \text{if } n = 0 \\
-1 & \text{if } n = 3 \\
-2 & \text{if } n = 4 \\
0 & \text{otherwise}
\end{cases} \]

This signal can be represented as:

\[ x[n] = \delta[n] - \delta[n - 3] - 2\delta[n - 4] \]

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

\[ x[n] = \sum_{m=-\infty}^{\infty} x[m] \delta[n - m] \]

\[ = \ldots + x[-1] \delta[n + 1] + x[0] \delta[n] + x[1] \delta[n - 1] + x[2] \delta[n - 2] + \ldots \]
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 \( x[\cdot] \) can be viewed as a sum of scaled, shifted unit sample responses:

\[ y[n] = \sum_{m=-\infty}^{\infty} x[m] h[n-m] \]

**Structure of Superposition**

If a system is linear and time-invariant (LTI) then its output is the sum of weighted and shifted unit sample responses.

\[
\delta[n] \rightarrow \text{system} \rightarrow h[n]
\]

\[
\delta[n-m] \rightarrow \text{system} \rightarrow h[n-m]
\]

\[
x[m]\delta[n-m] \rightarrow \text{system} \rightarrow x[m]h[n-m]
\]

\[
x[n] = \sum_{m=-\infty}^{\infty} x[m]\delta[n-m] \rightarrow \text{system} \rightarrow y[n] = \sum_{m=-\infty}^{\infty} x[m]h[n-m]
\]
Superposition by Example

Consider an LTI system described by \( y[n] = 3x[n] - x[n - 1] \). Compute the response of this system to an input \( x[\cdot] \) given by:

\[
x[n] = 2\delta[n - 1] + 5\delta[n - 2] + 3\delta[n - 4]
\]
Convolution

Response of an LTI system to an arbitrary input.

\[ y[n] = \sum_{m=-\infty}^{\infty} x[m] h[n-m] \equiv (x \ast h)[n] \]

This operation is called \textit{convolution} (verb form: \textit{convolve}).
Convolution

Convolution is represented with an asterisk.

\[
\sum_{m=-\infty}^{\infty} x[m]h[n - m] \equiv (x * h)[n]
\]

Convolution operates on **signals**, not samples.

The symbols \(x\) and \(h\) represent DT signals.
Convolving \(x\) with \(h\) generates a new DT signal \(x * h\).
DT Convolution: Summary

Unit sample response $h[n]$ is a complete description of an LTI system.

Given $h[\cdot]$, we can compute the response $y[\cdot]$ to any input $x[\cdot]$ by convolving $x[\cdot]$ and $h[\cdot]$: 

$$y[n] = (x \ast h)[n] \equiv \sum_{m=-\infty}^{\infty} x[m]h[n-m]$$
Discrete-Time Example: Solera Process

Aging and blending wines from different crops.

Start with 3 barrels of wine: newest at left, oldest at right.
Discrete-Time Example: Solera Process

Aging and blending wines from different crops.

Sell half of the oldest stock.

\[-1 - 2 - 3
\]

\[-3\]
sell
Discrete-Time Example: Solera Process

Aging and blending wines from different crops.

Refill oldest barrel from next-to-oldest barrel.

\[\begin{align*}
-1 & \quad -2 \\
\rightarrow & \\
-2 & \quad -3 \\
\end{align*}\]

-3 sell
Discrete-Time Example: Solera Process

Aging and blending wines from different crops.

Refill next-to-oldest barrel from youngest barrel.
Discrete-Time Example: Solera Process

Aging and blending wines from different crops.

Refill youngest barrel with this year’s harvest.
Discrete-Time Example: Solera Process

Aging and blending wines from different crops.

Old and new contents mix; ready for next year.

\[
\begin{array}{ccc}
0 & -1 & -2 \\
-1 & -2 & -3 \\
\end{array}
\]
Discrete-Time Example: Solera Process

Aging and blending wines from different crops.

Old and new contents mix; ready for next year.

Properties of solera process:

- Mixing produces a more uniform product.
- Mitigates worst-case results of one bad year.
- Blends wines from MANY previous years.
We can analyze these effects with a tracer experiment.

Add 1 unit of tracer to new crop; track tracer through the system.

How much tracer will be in each barrel at the end of year 3?
Solera Analysis

Add 1 unit of tracer to new crop; track tracer through the system.

<table>
<thead>
<tr>
<th>Year ( n )</th>
<th>Tracer in ( x[n] )</th>
<th>Barrel #1</th>
<th>Barrel #2</th>
<th>Barrel #3</th>
<th>Tracer out ( y[n] )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1/2</td>
<td>1/2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>1/4</td>
<td>2/4</td>
<td>1/4</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>1/8</td>
<td>3/8</td>
<td>3/8</td>
<td>1/8</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>1/16</td>
<td>4/16</td>
<td>6/16</td>
<td>3/16</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>1/32</td>
<td>5/32</td>
<td>10/32</td>
<td>6/32</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>1/64</td>
<td>6/64</td>
<td>15/64</td>
<td>10/64</td>
</tr>
</tbody>
</table>

\[
x[n] = 1
\]

\[
y[n] = \begin{cases} 3/16 & n = 4 \\ \cdots & \text{else} \end{cases}
\]
How would results change if tracer were added in year 1 (not 0)?

Original response:

Delayed input $\rightarrow$ delayed output:

Delaying the input by a year simply delays the outputs by one year.
A system is time-invariant if delaying the input to the system simply delays the output by the same amount of time.

Given

\[ x[n] \rightarrow \text{system} \rightarrow y[n] \]

the system is **time invariant** if

\[ x[n - n_0] \rightarrow \text{system} \rightarrow y[n - n_0] \]

is true for all \( n_0 \).
Scaling the input amplitudes:

$$0.5x[n]$$

$$0.5y[n]$$

Adding two inputs:

$$x[n] + x[n - 6]$$

$$y[n] + y[n - 6]$$

Linearly combining two inputs:

$$x[n] + 0.5x[n - 6]$$

$$y[n] + 0.5y[n - 6]$$
A system is linear if its response to a weighted sum of inputs is equal to the weighted sum of its responses to each of the inputs.

Given

\[ x_1[n] \rightarrow \text{system} \rightarrow y_1[n] \]

and

\[ x_2[n] \rightarrow \text{system} \rightarrow y_2[n] \]

the system is linear if

\[ \alpha x_1[n] + \beta x_2[n] \rightarrow \text{system} \rightarrow \alpha y_1[n] + \beta y_2[n] \]

is true for all \( \alpha \) and \( \beta \).
Convolution

If a system is linear and time invariant, the response to an arbitrary input is the convolution of that input with the unit-sample response.

\[ x[n] = \sum_{m=-\infty}^{\infty} x[k] \delta[n - m] \]

\[ y[n] = \sum_{m=-\infty}^{\infty} x[k] h[n - m] \]
## Check Yourself

For solera process ...

<table>
<thead>
<tr>
<th>Year $n$</th>
<th>Tracer in $x[n]$</th>
<th>Barrel #1</th>
<th>Barrel #2</th>
<th>Barrel #3</th>
<th>Tracer out $y[n]$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1/2</td>
<td>1/2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>1/4</td>
<td>2/4</td>
<td>1/4</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>1/8</td>
<td>3/8</td>
<td>3/8</td>
<td>1/8</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>1/16</td>
<td>4/16</td>
<td>6/16</td>
<td>3/16</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>1/32</td>
<td>5/32</td>
<td>10/32</td>
<td>6/32</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>1/64</td>
<td>6/64</td>
<td>15/64</td>
<td>10/64</td>
</tr>
</tbody>
</table>

How much tracer would be found in output from year 5, if one unit of tracer is added in years 0, 1, and 2?

1. $\frac{21}{32}$  
2. $\frac{1}{2}$  
3. $\frac{3}{16}$  
4. $\frac{9}{16}$  
5. none of above