Discrete-Time LTI Systems

- DT Convolution

- Define impulse response of DT System that is both Linear and Time-Invariant (LTI)

\[ \delta[n] \xrightarrow{\text{LTI}} h[n] \]

- To easily derive DT convolution formula, we view \( x[n] \) (input) as a sum of amplitude-scaled and time-shifted (Kronecker) delta functions \( \Rightarrow \) See Fig. 2.1 on pg. 76

\[ x[n] = \sum_{k=-\infty}^{\infty} x[k] \delta[n-k] \]
Time-Invariance dictates:

\[ \delta[n-k] \xrightarrow{\text{LTI}} h[n-k] \]

Homogeneity aspect of linearity dictates:

\[ x[k] \delta[n-k] \xrightarrow{\text{LTI}} x[k] h[n-k] \]

Superposition aspect of linearity dictates:

\[ \sum_{k=-\infty}^{\infty} x[k] \delta[n-k] \xrightarrow{\text{LTI}} \sum_{k=-\infty}^{\infty} x[k] h[n-k] \]

\[ = x[n] \]

\[ = x[n] * h[n] \]

See Fig. 2.2 on pg. 79
Summarizing:

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

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

- There are at least 3 ways to compute DT convolution:
- Method 1: collectively sum

Example: \( y[n] = x[n] + x[n-1] + x[n-2] \)

Find output when: \( x[n] = \delta(n) + 2\delta(n-1) + 3\delta(n-2) \)

Impulse response of system?

\( y[n] = h[n] \) when \( x[n] = \delta(n) \)

\( h[n] = \delta(n) + \delta(n-1) + \delta(n-2) \)
Since \( x[n] = f[n] + 2 \delta[n-1] + 3 \delta[n-2] \)

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

\[
= x[0] h[n] = h[n]
\]

\[
+ x[1] h[n-1] = 2 h[n-1]
\]

\[
+ x[2] h[n-2] = 3 h[n-2]
\]
Answer is sum: $y(n)$

$x(n)$ of "length" 3
$h(n)$ of "length" 3

$X(n) * H(n)$ of length $3 + 3 - 1 = 5$

$X(n) * H(n)$ of "length" $N_1 

h[n]$ of "length" $N_2$

Generally: $x(n)$ of "length" $N_1$

$h[n]$ of "length" $N_2$

$N_1 + N_2 - 1$
Note: not concerned with initial conditions in this course => unless stated otherwise assume system is initially at rest => all initial conditions = 0

Method 2: "run" input signal thru difference equation (Note: all DT LTI systems may be expressed as a difference equation)

• In the previous example: \(x[n] = 0\) for \(n < 0\)
  \(x[0] = 1, x[1] = 2, x[2] = 3, x[n] = 0\) for \(n > 2\)

- \(n = 0\)
  \(y[0] = x[0] + x[1] + x[-2] = 1 + 0 + 0 = 1\)
- \(y[1] = x[1] + x[0] + x[-1] = 2 + 1 + 0 = 3\)
- \(y[n] = 0\) for \(n > 4\)
**Method 3**

Graphical Method similar to that for CT convolution:

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

1. View \( x[k] \) and \( h[-(k-n)] \) as functions of \( k \)
2. Flip \( h[k] \) about \( k=0 \) to form \( h[-k] \)
3. Time-shift \( h[-k] \) to the right by \( n \) to form \( h[-(k-n)] \)
4. Pointwise-multiply to form product \( x[k]h[-(k-n)] \)
5. Sum the values of the product \( x[k]h[-(k-n)] \) over all \( k \)
6. Ostensibly repeat for each value of \( n \)

See Example 2.3 in text on pg. 83
More generally:

\[ x[n] = \alpha^n u[n] \]

\[ h[n] = \beta^n u[n] \]

Problem \( \alpha + \beta \)

\( y[n] = 0 \) for \( n < 0 \).

For \( n > 0 \):

\[ x[k] = \alpha^k u[k] \]

\[ h[-(k-n)] = \beta^{-(k-n)} u[-(k-n)] \]

\[ y[n] = \sum_{k=0}^{n} \alpha^k \beta^{n-k} = \beta^n \sum_{k=0}^{n} \left( \frac{\alpha}{\beta} \right)^k \]
\[ y[n] = \beta^n \frac{1 - (\frac{\alpha}{\beta})^{n+1}}{1 - \frac{\alpha}{\beta}} = \beta^n \frac{\beta - \frac{\alpha^n}{\beta^n}}{\beta - \alpha} \]

\[ = \left\{ \begin{array}{ll}
\frac{\beta}{\beta - \alpha} \beta^n - \frac{\alpha}{\beta - \alpha} \alpha^n \\
\end{array} \right\} u[n] \]

since starts at \( n = 0 \)

---

**Example 2.4 in text on pg. 85**

\[ x[n] = u[n] - u[n-5] = \begin{cases} 
1 & 0 \leq n \leq 4 \\
0 & \text{otherwise} 
\end{cases} \]

\[ h[n] = \alpha^n \{ u[n] - u[n-7] \} \]

In contrast to text approach (Method 2), this is short enough to do by Method 1
\[ x(0) = x(1) = x(2) = x(3) = x(4) = 1 \]

<table>
<thead>
<tr>
<th>( n )</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x(0) h(n) )</td>
<td>1</td>
<td>2</td>
<td>2^2</td>
<td>2^3</td>
<td>2^4</td>
<td>2^5</td>
<td>2^6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( x(1) h(n-1) )</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>2^2</td>
<td>2^3</td>
<td>2^4</td>
<td>2^5</td>
<td>2^6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( x(2) h(n-2) )</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>2^2</td>
<td>2^3</td>
<td>2^4</td>
<td>2^5</td>
<td>2^6</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( x(3) h(n-3) )</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>2^2</td>
<td>2^3</td>
<td>2^4</td>
<td>2^5</td>
<td>2^6</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( x(4) h(n-4) )</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>2^2</td>
<td>2^3</td>
<td>2^4</td>
<td>2^5</td>
<td>2^6</td>
<td>0</td>
</tr>
<tr>
<td>( y(n) )</td>
<td>1</td>
<td>1+2x</td>
<td>1+2x^2</td>
<td>( \frac{1-x^4}{1-x} )</td>
<td>( \frac{1-x^5}{1-x} )</td>
<td>( \frac{2^4(1-x^5)}{1-x} )</td>
<td>( \frac{2^3(1-x^4)}{1-x} )</td>
<td>( \frac{2^2(1-x^5)}{1-x} )</td>
<td>( \frac{2^2(1-x^3)}{1-x} )</td>
<td>( \frac{2^x}{1-x} )</td>
<td>( \frac{2^x}{1-x} )</td>
<td></td>
</tr>
</tbody>
</table>
DT convolution satisfies:

1. Commutativity: \( x_1[n] * x_2[n] = x_2[n] * x_1[n] \)

2. Associativity:
\[
(x_1[n] * x_2[n]) * x_3[n] = x_1[n] * (x_2[n] * x_3[n])
\]

3. Distributive Property:
\[
x_1[n] * (x_2[n] + x_3[n]) = x_1[n] * x_2[n] + x_1[n] * x_3[n]
\]
Supplemental Notes on Properties of Convolution which follow from the fact that the Fourier Transform converts convolution in time to multiplication in the frequency domain. (Fourier Transform is 1 to 1 linear operator)
Convolution satisfies distributive property:

\[ x[n] * (h_1[n] + h_2[n]) \]

\[ = x[n] * h_1[n] + x[n] * h_2[n] \]

\[ \xrightarrow{DFT} X(\omega) \left\{ H_1(\omega) + H_2(\omega) \right\} \]

\[ = X(\omega) H_1(\omega) + X(\omega) H_2(\omega) \]

\[ \rightarrow \text{because multiplication satisfies distributive property} \]
Convolution is a commutative operator because multiplication is!

\[ x[n] * h[n] = h[n] * x[n] \]

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

\[ X(w) \leftrightarrow \text{DTFT} \quad x[n] \]

\[ H(w) \leftrightarrow \text{DTFT} \quad h[n] \]

\[ h[n] * x[n] \leftrightarrow \text{DTFT} \quad H(w) X(w) \]
Convolution satisfies associativity:

\[ x[n] * h_1[n] * h_2[n] \]

\[ = (x[n] * h_1[n]) * h_2[n] \]

\[ = x[n] * (h_1[n] * h_2[n]) \]

\[ \xrightarrow{\text{DFT}} X(\omega) H_1(\omega) H_2(\omega) \]

\[ = X(\omega) H_2(\omega) H_1(\omega) \]

\[ = (x[n] * h_2[n]) * h_1[n] \]