ECE 317

Digital Signal Processing I (DSP I)

Lecture 6

Prof. Mojtaba Soltanalian

* Based on 2023 Slides from Prof. Rashid Ansari.

Topics of last and today's lectures

- **Last class:**
  - DT systems properties: Linearity, shift-invariance, memoryless, causality, stability
  - Examples
- **Today's class:**
  - Linear shift-invariant (LSI) systems
  - Convolution and examples
  - Short classwork exercise

DT systems: Implication of Linearity and Shift-Invariance

$$x[n] \longrightarrow \boxed{T} \longrightarrow y[n]$$

Linearity implies:

If $x_1[n] \longrightarrow \boxed{T} \longrightarrow y_1[n]$ and $x_2[n] \longrightarrow \boxed{T} \longrightarrow y_2[n]$

Then $x[n] = ax_1[n] + bx_2[n] \longrightarrow \boxed{T} \longrightarrow$

$$\begin{aligned} y[n] &= T(ax_1[n] + bx_2[n]) \\ &= aT(x_1[n]) + bT(x_2[n]) \\ &= ay_1[n] + by_2[n] \end{aligned}$$

Generalization to more sum of than two sequences applied to a linear system:

If the input is $x[n] = \sum_{k=-\infty}^{\infty} a_k x_k[n]$, and $T(x_k[n]) = y_k[n]$, then the output is

$$y[n] = T\left(\sum_{k=-\infty}^{\infty} a_k x_k[n]\right) = \sum_{k=-\infty}^{\infty} a_k T(x_k[n]) = \sum_{k=-\infty}^{\infty} a_k y_k[n]$$

Shift Invariance implies:

If $x_k[n] \longrightarrow \boxed{T} \longrightarrow y_k[n]$ then $x_k[n-n_0] \longrightarrow \boxed{T} \longrightarrow y_k[n-n_0]$

Recall: Signal representation as a sum of scaled and shifted unit impulse signals

Any signal $x[n]$ can be represented as a sum of scaled and shifted unit impulse signals

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

$x[k]$ is a scalar, not a signal/sequence

$\delta[n-k]$ is a signal – an impulse signal with a shift by an integer k

$x[n]$ is applied to a linear shift-invariant system:

$$x[n] \longrightarrow \boxed{T} \longrightarrow y[n]$$

In the special case where the input is an impulse signal:

$$\delta[n] \longrightarrow \boxed{T} \longrightarrow h[n]$$

we will denote the output as $h[n]$, and call $h[n]$ the system impulse response.

Linear Shift-Invariant (LSI) System

Linear Shift-Invariant (LSI) System

We will deal primarily with LSI systems in this course. That is, the system is both linear and shift-invariant. Consider a LSI system T characterized by impulse response $h[n]$. That is, $T(\delta[n]) = h[n]$.

Now consider an arbitrary input $x[n]$ to the system:

$$x[n] \longrightarrow \boxed{T} \longrightarrow y[n] = T(x[n])$$

Now $x[n] = \sum_{k=-\infty}^{\infty} x[k]\delta[n-k]$ where $x[k]$ are scalars and $\delta[n-k]$ are shifted imppulse sequences.

So $y[n] \ = T(x[n]) = T(\sum_{k=-\infty}^{\infty} x[k]\delta[n-k])$.

Since T is linear, $y[n] \ = \sum_{k=-\infty}^{\infty} x[k]T(\delta[n-k])$.

Since T is shift-invariant $T(\delta[n-k]) = h[n-k]$.

Therefore $y[n] \ = \sum_{k=-\infty}^{\infty} x[k]h[n-k]$.

System response is completely characterized by its impulse response $h[n]$.

Linear Shift-Invariant (LSI) System

LSI system is characterized by its impulse response $h[n]$

Linear Shift-Invariant (LSI) System

LSI system is characterized by its impulse response $h[n]$

- Output of LSI system $y[n]$ is the convolution sum of input $x[n]$ and $h[n]$

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

Once you know the impulse response $h[n]$, you can determine the output $y[n]$ for any input $x[n]$.

Stability and causality of an LSI system can be inferred from $h[n]$.

- Stable LSI system $\Leftrightarrow \sum_{n=-\infty}^{\infty} |h[n]| < \infty$

- Causal LSI system $\Leftrightarrow h[n] = 0, \quad n < 0$

Later, we will see why these two conditions are valid.

- Impulse response duration: FIR (finite-duration impulse response), IIR (infinite-duration impulse response)

Next, we will review how convolution is computed.

Convolution of finite-duration sequences $x[n]$ and $h[n]$

Convolution of finite-duration sequences $x[n]$ and $h[n]$

$x[n] = [\ldots 0\ \underline{2}\ 2\ 1\ 0\ 0\ 0 \ldots]$ $\quad$ $h[n] = [\ldots 0\ \underline{1}\ 2\ 3\ 0\ 0\ 0 \ldots]$

$$x[n] \longrightarrow \boxed{h[n]} \longrightarrow y[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k]$$

$$\delta[n] \longrightarrow \boxed{h[n]} \longrightarrow h[n] \qquad K\delta[n-k] \longrightarrow \boxed{h[n]} \longrightarrow Kh[n-k]$$

$$x[n] = \sum_{k=-\infty}^{\infty} x[k]\delta[n-k] \longrightarrow \boxed{h[n]} \longrightarrow y[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k]$$

$x[n] = [\ldots 0\ \underline{2}\ 2\ 1\ 0\ 0\ 0 \ldots]$

$$x[0]\delta[n] = [\ldots 0\ \underline{2}\ 0\ 0\ 0\ 0\ 0 \ldots] \longrightarrow \boxed{h[n]} \longrightarrow x[0]h[n] = [\ldots 0\ \underline{2}\ 4\ 6\ 0\ 0\ 0 \ldots]$$

$$x[1]\delta[n-1] = [\ldots 0\ \underline{0}\ 2\ 0\ 0\ 0\ 0 \ldots] \longrightarrow \boxed{h[n]} \longrightarrow x[1]h[n-1] = [\ldots 0\ \underline{0}\ 2\ 4\ 6\ 0\ 0 \ldots]$$

$$x[2]\delta[n-1] = [\ldots 0\ \underline{0}\ 0\ 1\ 0\ 0\ 0 \ldots] \longrightarrow \boxed{h[n]} \longrightarrow x[2]h[n-2] = [\ldots 0\ \underline{0}\ 0\ 1\ 2\ 3\ 0 \ldots]$$

$x[n] = [\ldots 0\ \underline{2}\ 2\ 1\ 0\ 0\ 0 \ldots]$ $\qquad$ $y[n] = [\ldots 0\ \underline{2}\ 6\ 11\ 8\ 3\ 0 \ldots]$

We will now consider another method of computing convolution

Another method of computing convolution of $x[n]$ and $h[n]$ (each of finite duration)

Another method of computing convolution of $x[n]$ and $h[n]$ (each of finite duration)

$x[n] \quad = [\ldots 0\ \underline{2}\ 2\ 1\ 0\ 0\ 0 \ldots]$ $\qquad h[n] \quad = [\ldots 0\ \underline{1}\ 2\ 3\ 0\ 0\ 0 \ldots]$

$x[n] = [\underline{2} \quad 2 \quad 1]$

$h[n] = [\underline{1} \quad 2 \quad 3]$

$y[n] = (x * h)[n] = x[n] * h[n]$, the last expression in common engineering notation.

$$Duration\ of\ y = Duration\ of\ x + Duration\ of\ h - 1$$
$$= 3 + 3 - 1 = 5.$$

$h[n$

$x[n]$

×	1	2	3
2	2	4	6
2	2	4	6
1	1	2	3

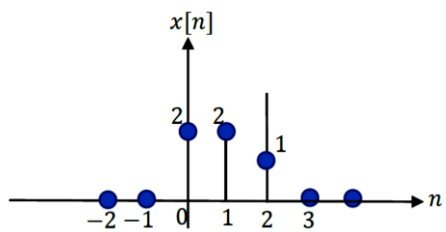
$y[n] = [\underline{2} \quad 6 \quad 11 \quad 8 \quad 3].$

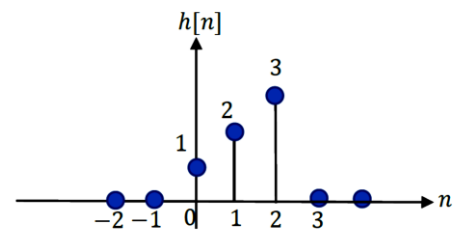
Method works well for short-duration signals and short-duration impulse response

A different look at convolution of $x[n]$ and $h[n]$ (of finite duration)

A different look at convolution of $x[n]$ and $h[n]$ (of finite duration)

$x[n] \quad = [\ldots 0\ \underline{2}\ 2\ 1\ 0\ 0\ 0 \ldots]$ $\qquad$ $h[n] \quad = [\ldots 0\ \underline{1}\ 2\ 3\ 0\ 0\ 0 \ldots]$





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

Let us examine $h[n-k]$ as a function of k

To compute the convolution, we define $\boxed{g[k] = h[-k].}$

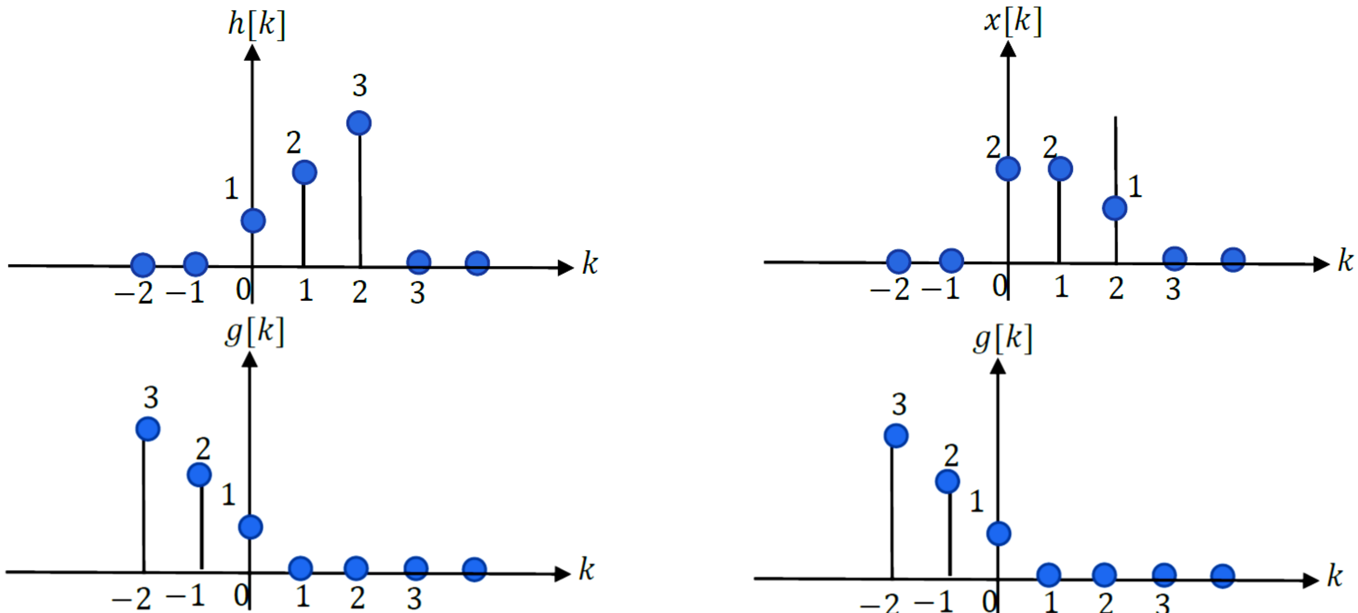
$\Rightarrow g[k-n] = h[-(k-n)] = h[n-k].$

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k] \Rightarrow y[n] = \sum_{k=-\infty}^{\infty} x[k]g[k-n].$$

A different look at convolution of $x[n]$ and $h[n]$ (of finite duration)

$x[n] \quad = [\ldots 0\ \underline{2}\ 2\ 1\ 0\ 0\ 0 \ldots]$ $\qquad h[n] \quad = [\ldots 0\ \underline{1}\ 2\ 3\ 0\ 0\ 0 \ldots]$

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]g[k-n].$$

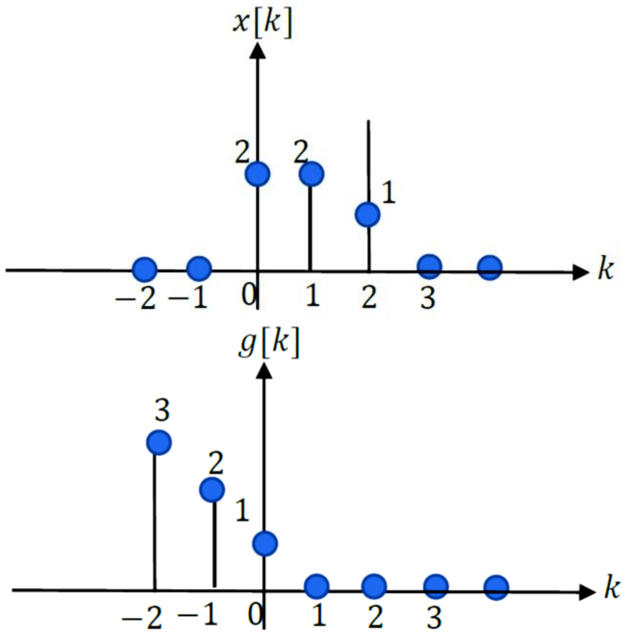


$$y[0] = \sum_{k=-\infty}^{\infty} x[k]g[k]$$

A different look at convolution: Computing steps

$x[n] \quad = [\ldots 0\ \underline{2}\ 2\ 1\ 0\ 0\ 0 \ldots]$ $\qquad h[n] \quad = [\ldots 0\ \underline{1}\ 2\ 3\ 0\ 0\ 0 \ldots]$

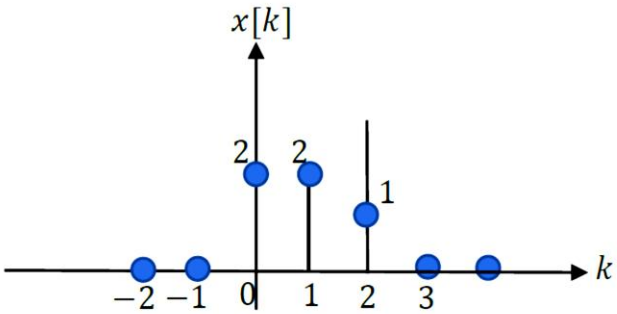
$$y[n] = \sum_{k=-\infty}^{\infty} x[k]g[k-n].$$

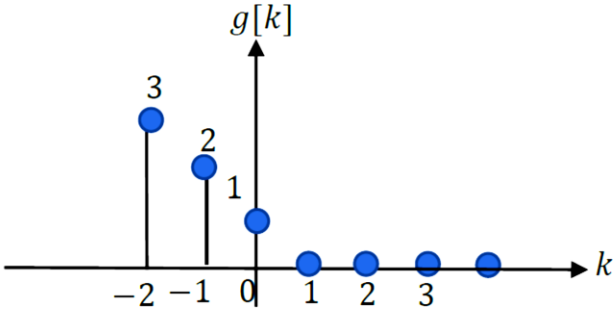


A different look at convolution: Computing steps

$x[n] \quad = [\ldots 0\ \underline{2}\ 2\ 1\ 0\ 0\ 0 \ldots]$ $\qquad h[n] \quad = [\ldots 0\ \underline{1}\ 2\ 3\ 0\ 0\ 0 \ldots]$

$$y[n] = \sum_{k=-\infty}^{\infty} x[k] g[k-n].$$





Steps in computing $y[n]$:

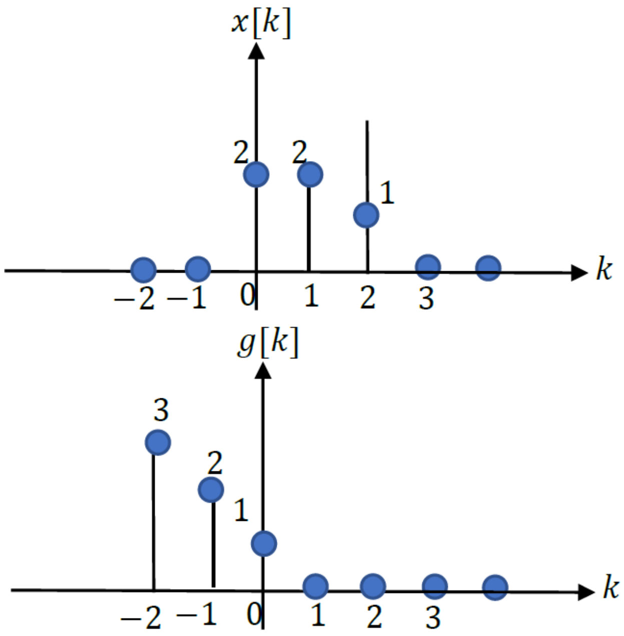
A different look at convolution: Computing steps

$x[n] \quad = [\ldots 0\ \underline{2}\ 2\ 1\ 0\ 0\ 0 \ldots]$ $\qquad h[n] \quad = [\ldots 0\ \underline{1}\ 2\ 3\ 0\ 0\ 0 \ldots]$

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]g[k-n].$$

Steps in computing $y[n]$:

- shift $g[k]$ by n, that is, obtain $g[k-n]$



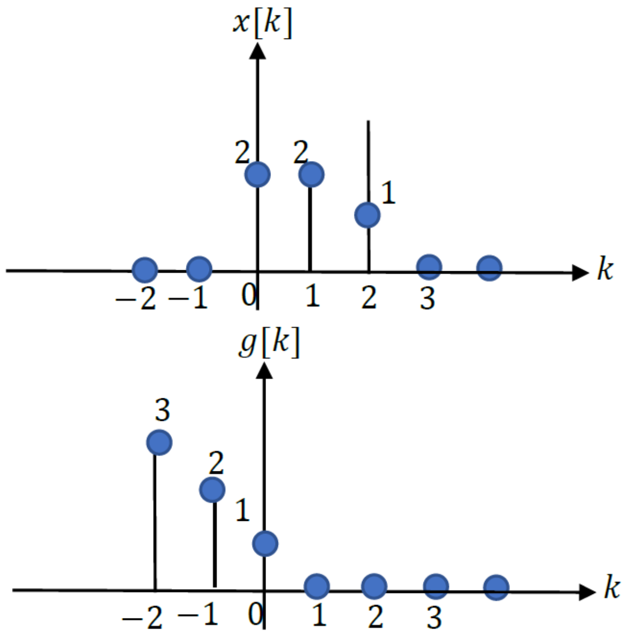
A different look at convolution: Computing steps

$x[n] \quad = [\ldots 0\ \underline{2}\ 2\ 1\ 0\ 0\ 0 \ldots]$ $\qquad h[n] \quad = [\ldots 0\ \underline{1}\ 2\ 3\ 0\ 0\ 0 \ldots]$

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]g[k-n].$$

Steps in computing $y[n]$:

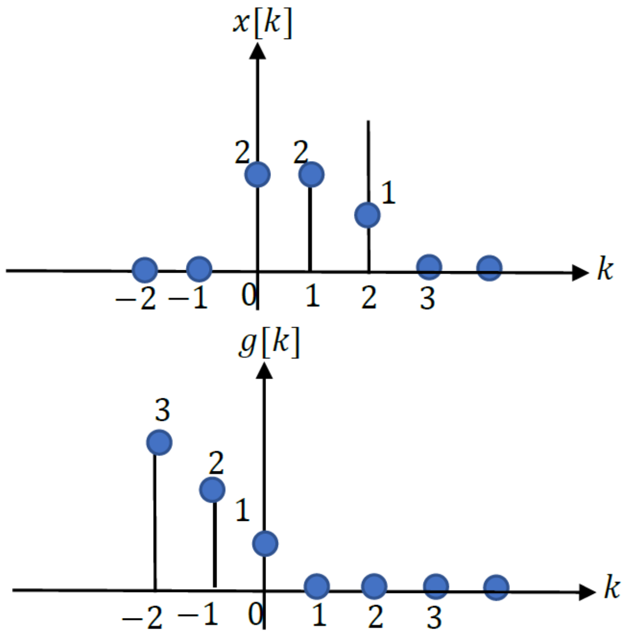
- shift $g[k]$ by n, that is, obtain $g[k-n]$
- multiply $x[k]$ point-wise with$g[k-n]$.



A different look at convolution: Computing steps

$x[n] \quad = [\ldots 0\ \underline{2}\ 2\ 1\ 0\ 0\ 0 \ldots]$ $\qquad h[n] \quad = [\ldots 0\ \underline{1}\ 2\ 3\ 0\ 0\ 0 \ldots]$

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]g[k-n].$$



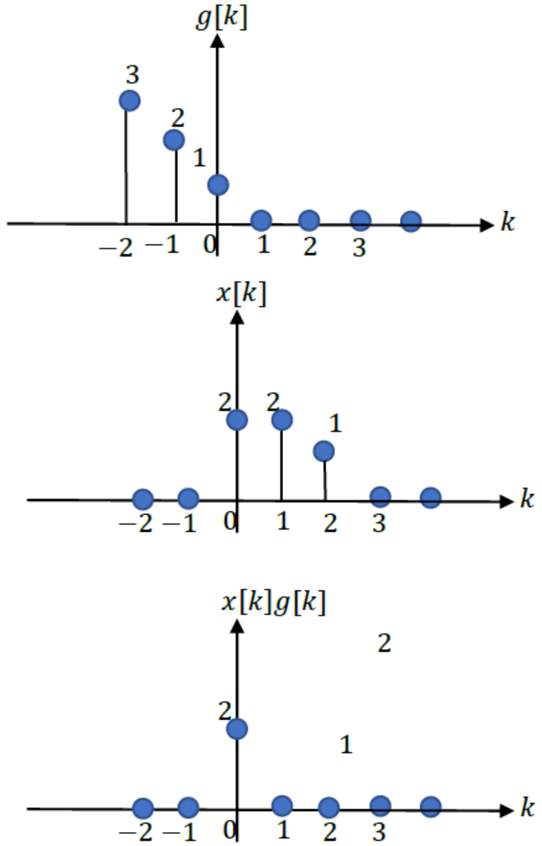
Steps in computing $y[n]$:

- shift $g[k]$ by n, that is, obtain $g[k-n]$
- multiply $x[k]$ point-wise with $g[k-n]$.
- add the terms $x[k]g[k-n]$ for all k.

A different look at convolution: Computing steps

$x[n] \quad = [\ldots 0\ \underline{2}\ 2\ 1\ 0\ 0\ 0\ \ldots]$ $\qquad h[n] \quad = [\ldots 0\ \underline{1}\ 2\ 3\ 0\ 0\ 0\ \ldots]$

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



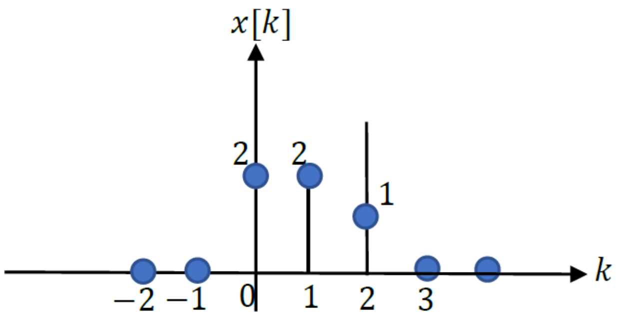
$y[n]$ for $n = 0$

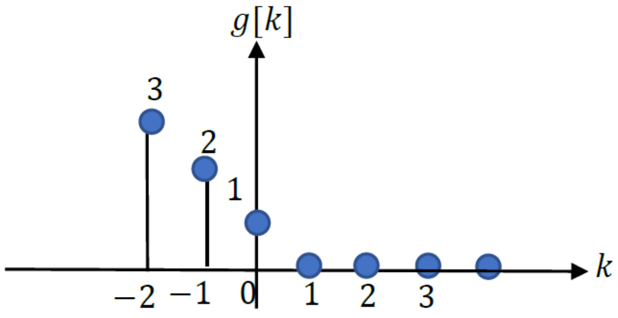
$y[0] = 2 \times 1 = 2$

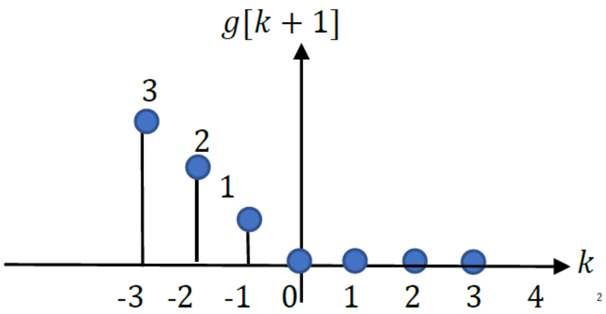
A different look at convolution: Computing steps

$x[n] \quad = [\ldots 0\ \underline{2}\ 2\ 1\ 0\ 0\ 0 \ldots]$ $\qquad h[n] \quad = [\ldots 0\ \underline{1}\ 2\ 3\ 0\ 0\ 0 \ldots]$

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]g[k-n].$$





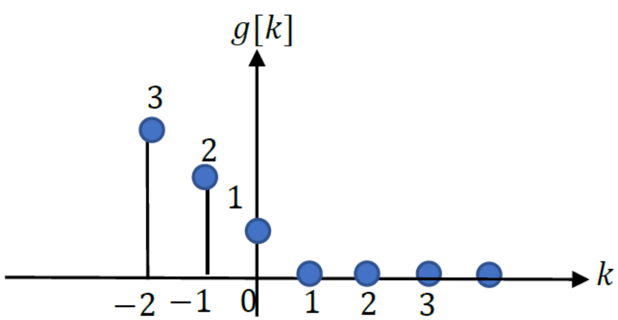


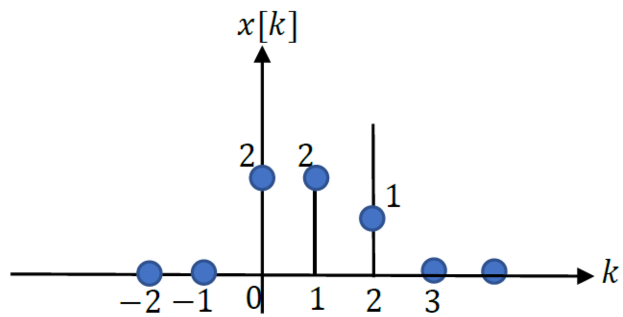
$y[n]$ for n =-1
y[-1]= 0

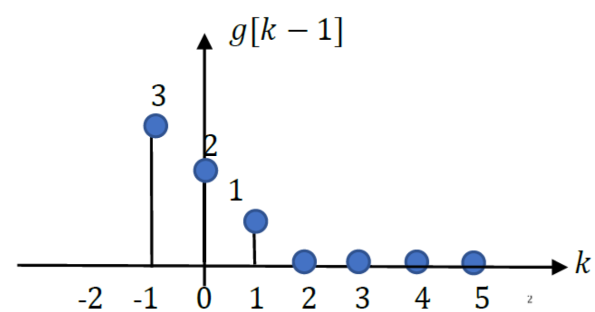
A different look at convolution: Computing steps

$x[n] \quad = [\ldots 0\ \underline{2}\ 2\ 1\ 0\ 0\ 0 \ldots]$ $\qquad$ $h[n] \quad = [\ldots 0\ \underline{1}\ 2\ 3\ 0\ 0\ 0 \ldots]$

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]g[k-n].$$







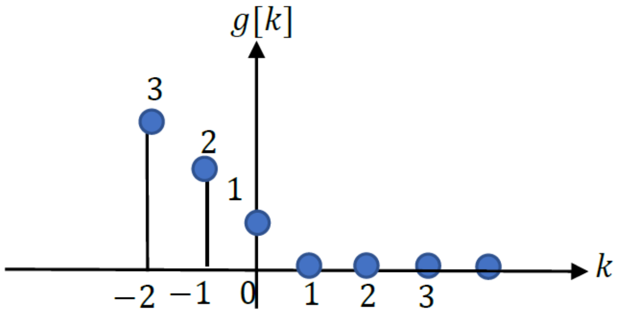
$y[n]$ for $n = 1$

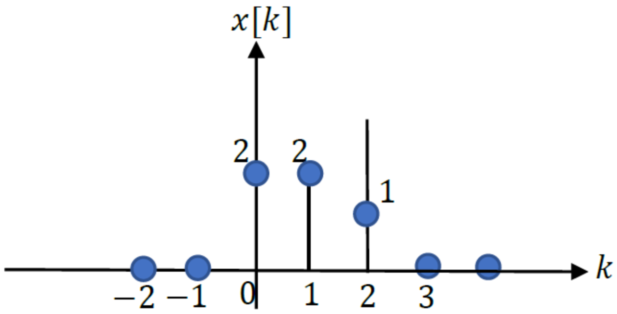
$y[1] = 2 \times 2 + 2 \times 1 = 6$

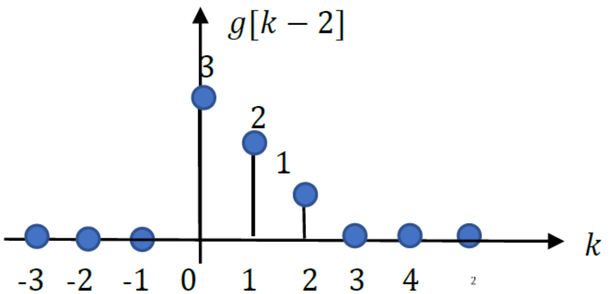
A different look at convolution: Computing steps

$x[n] \quad = [\ldots 0\ \underline{2}\ 2\ 1\ 0\ 0\ 0 \ldots]$ $\qquad h[n] \quad = [\ldots 0\ \underline{1}\ 2\ 3\ 0\ 0\ 0 \ldots]$

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]g[k-n].$$







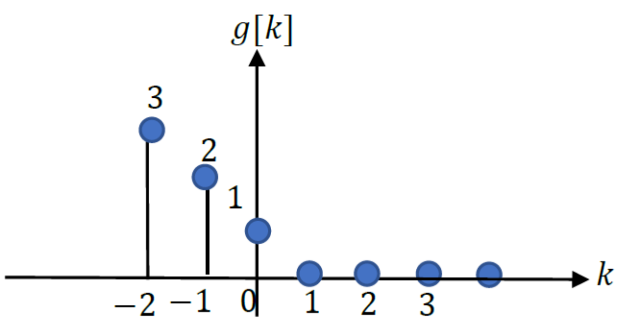
$y[n]$ for $n = 2$

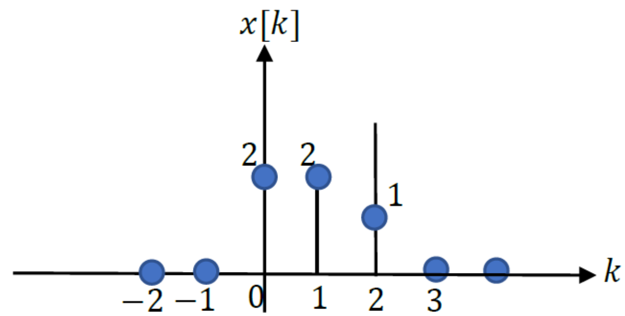
$y[2] = 2 \times 3 + 2 \times 2 + 1 \times 1 = 11$

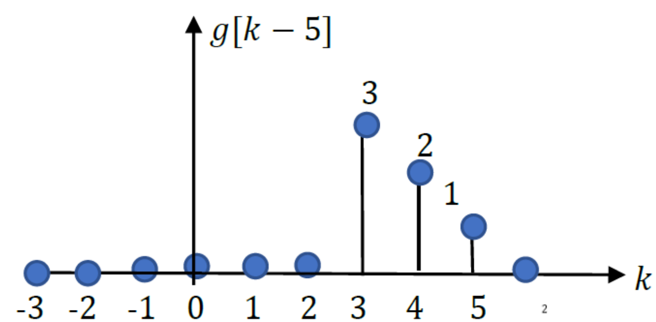
A different look at convolution: Computing steps

$x[n] \quad = [\ldots 0\ \underline{2}\ 2\ 1\ 0\ 0\ 0 \ldots]$ $\qquad$ $h[n] \quad = [\ldots 0\ \underline{1}\ 2\ 3\ 0\ 0\ 0 \ldots]$

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]g[k-n].$$







$y[n]$ for $n = 5$

$y[0] = 0$

$y[n] \quad = [\ldots 0\ \underline{2}\ 6\ 11\ 8\ 3\ 0\ 0 \ldots]$

A different look at convolution: Computing steps (without plots)

$x[k] \quad = [\ldots 0\ \underline{2}\ 2\ 1\ 0\ 0\ 0 \ldots]$

$h[k] \quad = [\ldots 0\ \underline{1}\ 2\ 3\ 0\ 0\ 0 \ldots]$

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

$g[k] = [\ldots 0\ 3\ 2\ \underline{1}\ 0\ 0 \ldots]$

$y[n]$ for n =-1

y[-1]= 0

$g[k+1] = [\ldots 0\ 3\ 2\ 1\ \underline{0}\ 0\ 0 \ldots]$

$x[k] \quad = [\ldots 0\ \underline{2}\ 2\ 1\ 0\ 0\ 0 \ldots]$

$y[n]$ for $n = 0$

$y[0] = 2 \times 1 = 2$

$g[k] = h[\text{-}k] = [\ldots 0\ 3\ 2\ \underline{1}\ 0\ 0 \ldots]$

$x[k] \quad = [\ldots 0\ \underline{2}\ 2\ 1\ 0\ 0\ 0 \ldots]$

$y[n]$ for $n = 1$

$y[1] = 2 \times 2 + 2 \times 1 = 6$

$g[k-1] = [\ldots 0\ 0\ 0\ 3\ \underline{2}\ 1\ 0 \ldots]$

$x[k] \quad = [\ldots 0\ \underline{2}\ 2\ 1\ 0\ 0\ 0 \ldots]$

$y[n] \quad = [\ldots 0\ \underline{2}\ 6\ 11\ 8\ 3\ 0\ 0 \ldots]$

A different look at convolution: Computing steps

$x[k] \quad = [\ldots 0\ \underline{2}\ 2\ 1\ 0\ 0\ 0 \ldots]$

$h[k] \quad = [\ldots 0\ \underline{1}\ 2\ 3\ 0\ 0\ 0 \ldots]$

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

$g[k] = [\ldots 0\ 3\ 2\ \underline{1}\ 0\ 0 \ldots]$

$y[n]$ for $n = 2$

$y[2] = 2 \times 3 + 2 \times 2 + 1 \times 1 = 11$

$g[k-2] = [\ldots 0\ 0\ 0\ \underline{3}\ 2\ 1 0\ 0\ \ldots]$
$x[k] \quad = [\ldots 0\ \underline{2}\ 2\ 1\ 0\ 0\ 0 \ldots]$

$y[n]$ for $n = 3$

$y[3] = 2 \times 3 + 1 \times 2 = 8$

$g[k-3] = [\ldots 0\ 0\ 0\ \underline{0}\ 3\ 2\ 1\ 0 \ldots]$
$x[k] \quad = [\ldots 0\ \underline{2}\ 2\ 1\ 0\ 0\ 0 \ldots]$

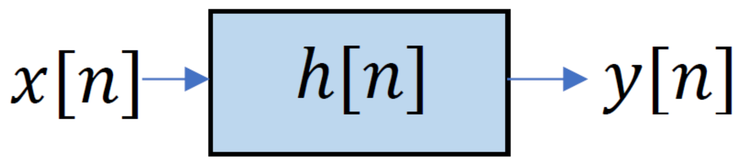
$y[n]$ for $n = 4$

$y[4] = 1 \times 3 = 3$

$g[k-4] = [\ldots 0\ 0\ 0\ \underline{0}\ 0\ 3\ 2\ 1\ 0 \ldots]$
$x[k] \quad = [\ldots 0\ \underline{2}\ 2\ 1\ 0\ 0\ 0 \ldots]$

$y[n] \quad = [\ldots 0\ \underline{2}\ 6\ 11\ 8\ 3\ 0\ 0 \ldots]$

Linear Shift-Invariant (LSI) System



Output: $y[n] = \sum_{k=-\infty}^{\infty} h[k]x[n-k] = \sum_{k=-\infty}^{\infty} x[k]h[n-k]$

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k]$$
$$= \cdots + h[-2]x[n+2] + h[-1]x[n+1] + h[0]x[n] + h[1]x[n-1]$$
$$+ h[2]x[n-2] + h[3]h[n-3] + \cdots$$

LSI system is completely characterized by its impulse response $h[n]$

Some observations about Linear Shift-Invariant (LSI) Systems

LSI system is characterized by its impulse response $h[n]$

- Output of LSI system $y[n]$ is the convolution sum of input $x[n]$ and $h[n]$

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

  Once you know the impulse response $h[n]$, you can determine the output $y[n]$ for any input $x[n]$.

- Stable LSI system $\Leftrightarrow \sum_{n=-\infty}^{\infty} |h[n]| < \infty$

- Causal LSI system $\Leftrightarrow h[n] = 0, \quad n < 0$

- Impulse response duration: FIR (finite-duration impulse response), IIR (infinite-duration impulse response)

Causality of LSI System

Output: $y[n] = \sum_{k=-\infty}^{\infty} h[k]x[n-k]$

Causality: What is the requirement on the system impulse response?

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k]$$
$$= \cdots + h[-2]x[n+2] + h[-1]x[n+1] + h[0]x[n] + h[1]x[n-1]$$
$$+ h[2]x[n-2] + h[3]h[n-3] + \cdots$$

What is the requirement on the impulse response so that the output $y[n]$ does not depend on future input samples $x[n+1], x[n+2]$, …?

Requirement on the impulse response: $h[n] = 0 \text{ for } n < 0$

Stability of LSI System

Output: $y[n] = \sum_{k=-\infty}^{\infty} h[k]x[n-k]$

Stability: What is the requirement on the system impulse response?

Assume input $x[n]$ is bounded. That is $|x[n]| \leq B_I$. Now

$$|y[n]| = |\sum_{k=-\infty}^{\infty} h[k]x[n-k]| \leq \sum_{k=-\infty}^{\infty} |h[k]||x[n-k]| \leq B_I \sum_{k=-\infty}^{\infty} |h[k]|$$

Let $S_h = \sum_{k=-\infty}^{\infty} |h[k]|$ be finite. Then $|y[n]| \leq S_h B_I$.

That is, the output is bounded $|y[n]|$ for all n by a positive number $B_O = S_h B_I$.

So the system is stable if $\sum_{k=-\infty}^{\infty} |h[k]| < \infty$.

We can also show that if the system is stable then $\sum_{k=-\infty}^{\infty} |h[k]| < \infty$.

Therefore: LSI system stable $\Leftrightarrow \sum_{k=-\infty}^{\infty} |h[k]| < \infty$.

Convolution and LSI systems: Some simple notions

Consider an LSI system with a special simple impulse response:

$$h[n] = \delta[n - n_0]$$

If the input is any signal $x[k]$, the output is:

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k] = \sum_{k=-\infty}^{\infty} x[k]\delta[n - n_0 - k] =$$

$$y[n] = x[n - n_0]$$

The system just shifts the input by n_0. Let us examine two simple inputs to the system.

- Now, let the input to this system be $x[k] = \delta[n]$. What is the output?

  $y[n] = \delta[n] * \delta[n - n_0] = \delta[n - n_0]$. That is impulse response by definition.

- If $x[n] = \delta[n - n_1]$, then $y[n] = x[n - n_0] = \delta[n - n_1 - n_0]$

  $$y[n] = x[n] * h[n] = \delta[n - n_1] * \delta[n - n_0] = \delta[n - n_1 - n_0]$$

Convolution and impulse signals: Some simple notions

What is $\delta[n-2] * \delta[n-3]$?

$$= \delta[n-5]$$

What is $\delta[n+1] * \delta[n+5]$?

$$= \delta[n+6]$$

What is $\delta[n-3] * \delta[n+2]$?

$$= \delta[n-1]$$

Note that the operation of convolution is distributive over the operation of addition.

$$y[n] = (x_1[n] + x_2[n]) * h[n] = x_1[n] * h[n] + x_2[n] * h[n]$$

What is $(\delta[n-3] + \delta[n+2]) * \delta[n-1]$?

$$= \delta[n-4] + \delta[n+1]$$