DT LTI Sample Problems

[OW 2.1] Let $x[n] = \delta[n] + 2\,\delta[n-1] - \delta[n-3]$

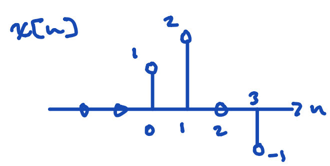
$h[n] = 2\,\delta[n+1] + 2\,\delta[n-1]$

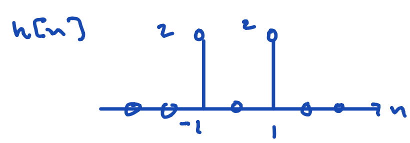
a) Compute and plot $y_1[n] = x[n] * h[n]$.

1. What is the solution supposed to be?

   A signal $y_1[n]$ that is the output of $h[n]$ with input $x[n]$.

2. Always draw a picture if you can!

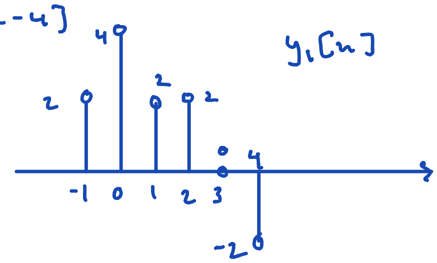




3. These are short signals so we could do it manually. Either use $\sum_{k=-\infty}^{\infty} h[k]\,x[n-k]$

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

$$
\begin{aligned}
y_1[n] &= 2\delta[n+1] + 2\,\delta[n-1] \\
&\quad + 4\,\delta[n] + 4\,\delta[n-2] \\
&\quad + 0\,(2\delta[n-1] + 2\delta[n-3]) \\
&\quad - 2\,\delta[n-2] - 2\,\delta[n-4] \\
&= 2\,\delta[n+1] + 4\,\delta[n] + 2\delta[n-1] + 2\delta[n-2] \\
&\quad - 2\,\delta[n-4]
\end{aligned}
$$



4. Do some sanity checks:

length of $y_1 = 6 = 4+3-1$

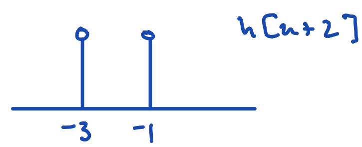
c) Compute and plot $y_2[n] = x[n] * h[n+2]$

$$h[n+2] = 2\delta[(n+2)+1] + 2\delta[(n+2)-1]$$

$$= 2\delta[n+3] + 2\delta[n+1]$$

$\Rightarrow$ just $h[n]$ shifted back 2 steps

Draw a picture!



It helps sometimes to just define some new notation to avoid confusion:

$$g[n] = h[n+2] = 2\delta[n+3] + 2\delta[n+1]$$

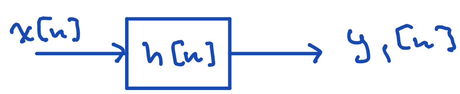
Now there are a couple of ways you can do this:

1) same way as above for part a)

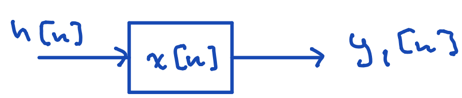
$\Rightarrow$ add up scaled & shifted copies of g

2) Use properties of LTI systems:

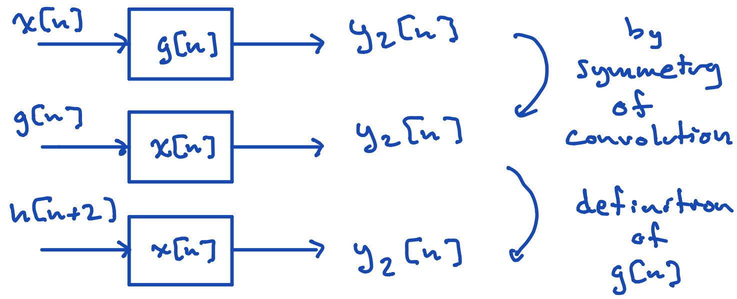
From part a) we have



Because convolution is commutative, we can also think of $y_1[n]$ as the output of an <u>LTI system</u> with impulse response $x[n]$ to an <u>input signal</u> $h[n]$:

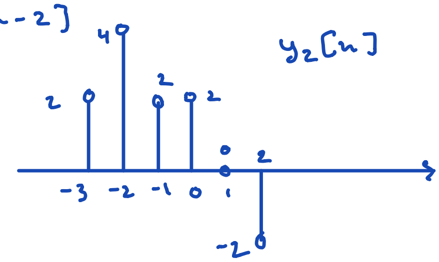


Now we want to figure out this system:



Since $x[n]$ is an <u>LTI system</u> it is time-invariant, which means shifting an input by k shifts the output by k. We know that the output of $x[n]$ with input $h[n]$ is $y_1[n]$. So if we shift the input back by $k=2$ time steps to get $h[n+2]$, the output is shifted back by 2:

$$\begin{aligned} &y_2[n] \\ &= y_1[n+2] \\ &= 2\delta[n+3] + 4\delta[n+2] + 2\delta[n+1] + 2\delta[n] \\ &\quad -2\delta[n-2] \end{aligned}$$



Key steps:

1) Defining $g[n]$ to avoid confusion

2) Use commutativity of convolution to make $x[n]$ the system and $h[n]$ the input.

3) Use LTI property to give $y_2[n]$ in terms of $y_1[n]$.

Part b) is easier — try it yourself!

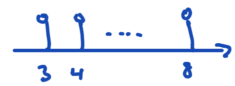
[OW 2.4] Compute and plot $y[n] = x[n] * h[n]$

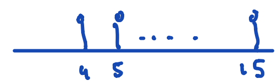
where

$$x[n] = \begin{cases} 1 & 3 \le n \le 8 \\ 0 & \text{otherwise} \end{cases}$$

$$h[n] = \begin{cases} 1 & 4 \le n \le 15 \\ 0 & \text{otherwise} \end{cases}$$

Draw a picture!





This is a case where the signals are a bit long so the direct approach we used above would get tedious. Instead, let's use the general formula for finite signals that we saw in class.

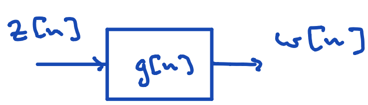
1) Try to get a sense of what the system is doing to the input signal: you can do this visually or by looking at the impulse response

$h[n]$ is taking the sum of 12 consecutive input values

2) Use LTI properties to make the problem simpler if you can:

$x[n+3]$ is 1 from 0 to 5
$h[n+4]$ is 1 from 0 to 11

So if we find the output of



where

$z[n] = x[n+3]$
$g[n] = h[n+4]$

we can use time invariance and commutativity

$z[n-3] \longrightarrow \boxed{g[n]} \longrightarrow w[n-3]$ by time invariance

$g[n] \longrightarrow \boxed{z[n-3]} \longrightarrow w[n-3]$ by commutativity

$g[n-4] \longrightarrow \boxed{z[n-3]} \longrightarrow w[n-7]$ by time-invariance

So we can just try to find $w[n]$ and delay it by 7 to get our final answer.

Finally, we had our general solution for $h[n]$ shorter than $x[n]$, so to make things match, use

$g[n] \longrightarrow \boxed{z[n]} \longrightarrow w[n]$

3) Figure out all the signal/impulse response start/stop times and the length of $w[n]$:

$M_{min} = 0$ $\qquad N_{min} = 0$

$M_{max} = 5$ $\qquad N_{max} = 11$

$M = 5 - 0 + 1 = 6$ $\qquad N = 11 - 0 + 1 = 12$

$\Rightarrow$ The length of $w[n]$ will be $N+M-1 = 17$

The length of $y[n]$ will also be 17, from $n = 7$ to $n = 23$

4) Use the 5-phase approach to find $(z * g)[n]$:

Phase 1: $w[n] = 0$ from $n = -\infty$ to $n = -1$

Phase 2:
$$w[n] = \sum_{k=M_{min}}^{n-N_{min}} z[k]\, g[n-k]$$
$$= \sum_{k=0}^{n} 1 \cdot 1$$
$$= n+1$$

for $0 \leq n < M_{min} + N_{max} = 5$

Phase 3: total overlap: all 6 values of $z[n]$ are $= 1$ and all values of $g[n] = 1$, so

$w[n] = 6$ for $5 \leq n \leq M_{min} + N_{max} = 11$

Phase 4:
$$w[n] = \sum_{k=n-N_{max}}^{M_{max}} z[k]\, g[n-k]$$
$$= \sum_{n-11}^{5} 1 \cdot 1$$
$$= 5 - (n-11) + 1$$
$$= 17 - n$$

for $11 < n \leq M_{max} + N_{max} = 16$

Phase 5: $w[n] = 0$ for $n > 16$

Putting it together:

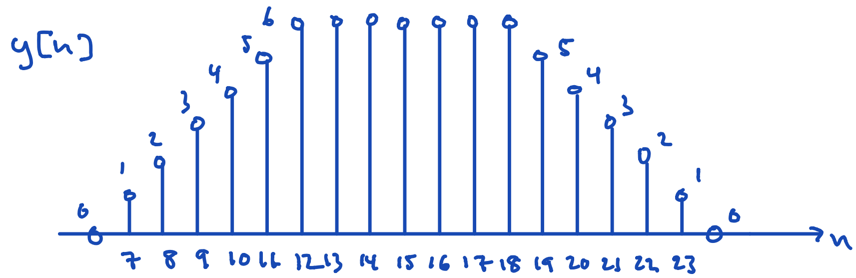
$$w[n] = \begin{cases} 0 & n < 0 \\ n+1 & n = 0, \cdots, 4 \\ 6 & n = 5, \cdots 11 \\ 17 - n & n = 12, \cdots 16 \\ 0 & n > 16 \end{cases}$$

So

$$y[n] = w[n-7] = \begin{cases} 0 & n < 7 \\ n-6 & n = 7, \cdots, 11 \\ 6 & n = 12, \cdots, 18 \\ 24 - n & n = 19, \cdots, 23 \\ 0 & n > 23 \end{cases}$$

Key tricks: 1) Use LTI properties to shift things to 0

2) Use commutativity of convolution to get things into a form where you can apply the general result.



Note: this is generally what you get when you convolve 2 boxes: a linear ramp up, then flat where one box covers the other, and then a linear ramp down.

[OW 2.21] a) Compute the convolution $y[n] = x[n] * h[n]$

of $\quad x[n] = \alpha^n u[n]$

$\quad h[n] = \beta^n u[n] \qquad \alpha \neq \beta$

Note: If general parameters α, β throw you for a loop, try just setting them to some value, solving the problem, and then go back to generalize. Let's try that here.

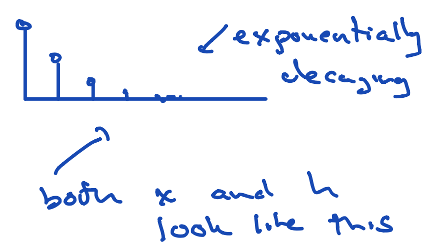
This is an infinite signal through an IIR filter. Since $h[n]$ and $x[n]$ are 0 for $n < 0$, we know

$y[n] = 0$ for $n < 0$ (can you see why?)

Since α and β are confusing, let's try $\alpha = 1/2$ and $\beta = 1/3$ for starters.

$$x[n] = \frac{1}{2^n} u[n]$$

$$h[n] = \frac{1}{3^n} u[n]$$



Now just plug into the convolution formula:

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

$$= \sum_{k=-\infty}^{\infty} \frac{1}{2^k} \frac{1}{3^{n-k}} \underbrace{u[k]}_{=0 \text{ for } k<0} \underbrace{u[n-k]}_{=0 \text{ for } k>n}$$

$$= \sum_{k=0}^{n} \frac{1}{2^k} \frac{1}{3^{n-k}}$$

$$= \frac{1}{3^n} \sum_{k=0}^{n} \left(\frac{3}{2}\right)^k$$

bad! $3/2 > 1$ so we can't use geometric series!?

What to do? Use commutativity:

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

$$= \sum_{k=0}^{n} \frac{1}{3^k} \cdot \frac{1}{2^{n-k}}$$

$$= \frac{1}{2^n} \sum_{k=0}^{n} \left(\frac{2}{3}\right)^k$$

better!

$$= \frac{1}{2^n} \frac{1-(2/3)^{n+1}}{1-2/3}$$

from our general geometric series formulas

$$= \frac{1/2^n - 2/3^{n+1}}{1/3}$$

$$= \frac{3}{2^n} - \frac{2}{3^{n+1}}$$

Oh how can we generalize to α, β?

It seems to matter whether $\alpha > \beta$ (e.g above we had $1/2 > 1/3$)

So suppose $\alpha > \beta$:

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

$$= \sum_{k=-\infty}^{\infty} \beta^k \alpha^{n-k}\, u[k]\, u[n-k]$$

$$= \sum_{k=0}^{n} \beta^k \alpha^{n-k}$$

$$= \alpha^n \sum_{k=0}^{n} \left(\frac{\beta}{\alpha}\right)^k$$

$\beta/\alpha < 1$ so we're ok

$$= \alpha^n \frac{1-(\beta/\alpha)^{n+1}}{1-\beta/\alpha}$$

$$= \frac{\alpha^n - \alpha^{-1}\beta^{n+1}}{1-\beta/\alpha}$$

$$= \frac{\alpha^{n+1} - \beta^{n+1}}{\alpha - \beta}$$

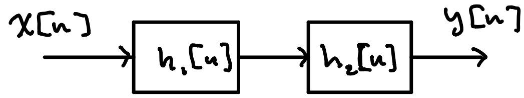
If $\beta > \alpha$ we can do $x * h$ and then

$$y[n] = \frac{\beta^{n+1} - \alpha^{n+1}}{\beta - \alpha}$$

(check it yourself!)

What the general α, β solution shows us is the <u>structure</u> of the solution.

[OW 2.43] c) Suppose we have a cascade of two systems as shown:



where $h_1[n] = \sin(8n)$

$h_2[n] = a^n u[n] \qquad |a| < 1$

$x[n] = \delta[n] - a\,\delta[n-1]$

Find $y[n]$.

The goal of this problem is to use LTI system properties & convolution properties to make life a lot easier.

The <u>bad way</u> to do this is to try to compute

$$(x * h_1)[n] = \sum_{k=-\infty}^{\infty} x[k]\, h_1[n-k]$$

$$= \sum_{k=-\infty}^{\infty} (\delta[k] - a\,\delta[k-1])$$

$$= \sin(8n) - a \sin(8n)$$

and then convolve this with h_2. Go ahead and try it — it's doable but messy.

Instead, use commutativity and associativity:

$$(x * h_1) * h_2 = x * (h_1 * h_2)$$

$$= x * (h_2 * h_1)$$

$$= (x * h_2) * h_1$$

To see why in this case this might be better:

$$(x * h_2)[n] = \sum_{k=-\infty}^{\infty} \left(\delta[k] - a\,\delta[k-1] \right) a^{n-k} u[n-k]$$

$$= a^n u[n] - a \cdot a^{n-1} \underbrace{u[n-1]}_{\substack{= 1 \text{ for } n \geq 1 \\ \text{and } 0 \text{ for } n < 1}}$$

$$= a^0 \delta[n]$$

$$= \delta[n]$$

So $h_2[n]$ outputs $\delta[n]$ for input $x[n]$. If we think of $x[n]$ as the impulse response of some system, then $h_2[n]$ is the <u>inverse</u> of that system. If the next to last line is confusing to you, just write the unit step functions as sums of δ-functions:

$$a^n u[n] = a^0 \delta[n] + a^1 \delta[n-1] + a^2 \delta[n-2] \ldots$$

$$a^n u[n-1] = a^1 \delta[n-1] + a^2 \delta[n-2] + \ldots$$

So when you subtract you just get $\delta[n]$.

Now we convolve $(x * h_2)[n]$ with $h_1[n]$. But since $(x * h_2)[n] = \delta[n]$, the output is just the impulse response $h_1[n]$.