TECH 350: DSP

Class VI: More Ways to Characterize LTI Systems

Tools for Describing Filters (So Far)

How can we describe filters?

First, by their **frequency response**

(magnitude and phase response)

Second, by **how they are implemented**

(difference equations, underlying functions (Chebyshev, Butterworth, etc.), order)

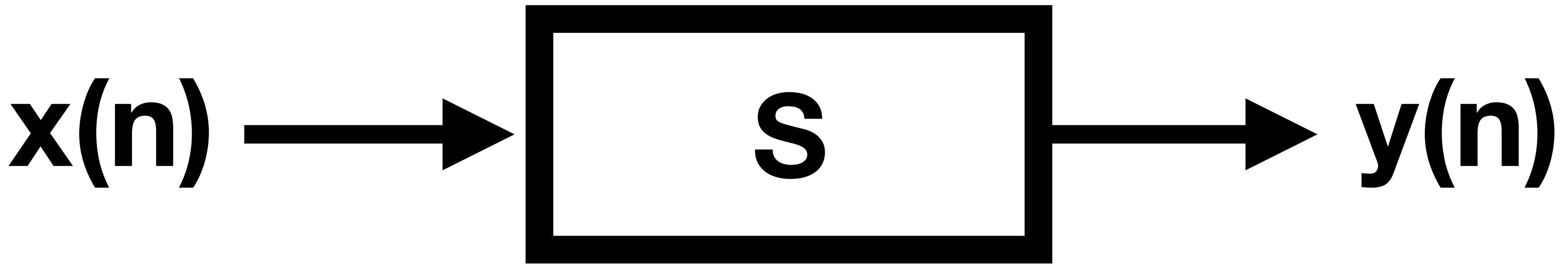
Third, by **contextualizing them vis-a-vis filter parameters**/characteristics

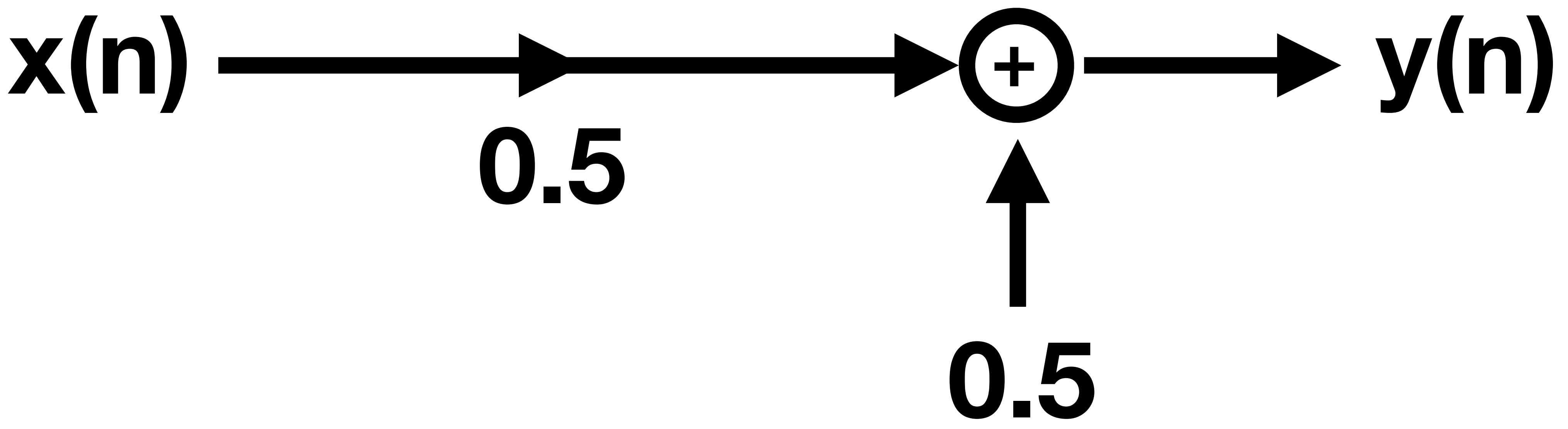
(cutoff frequency, resonance, etc.)

Example 1

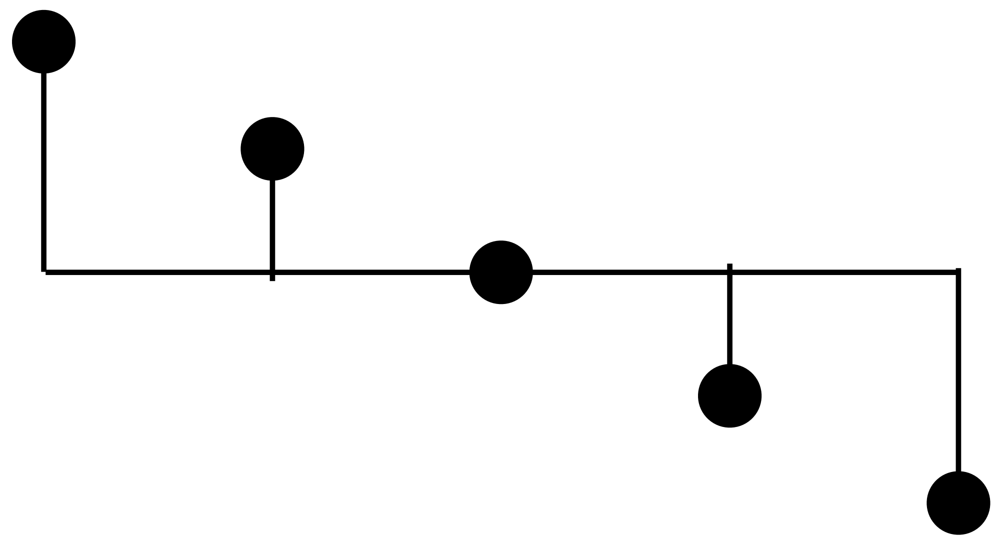
System: y(n) = S(x(n)), where S: output = input * 0.5 + 0.5

where n = 0, 1, 2, 3, . . . are indices to input X and output Y

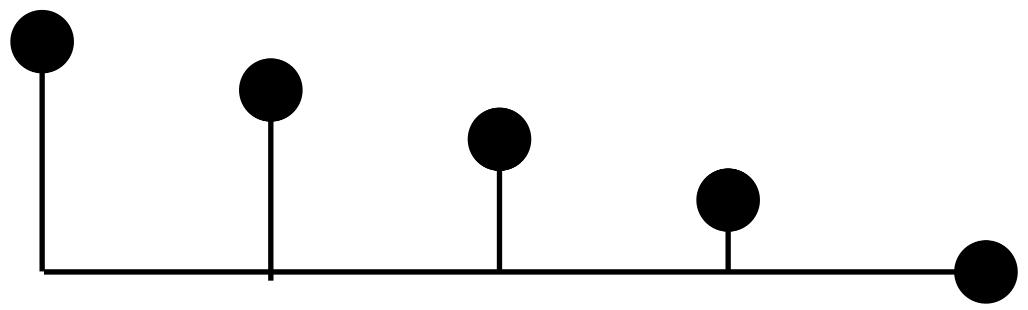


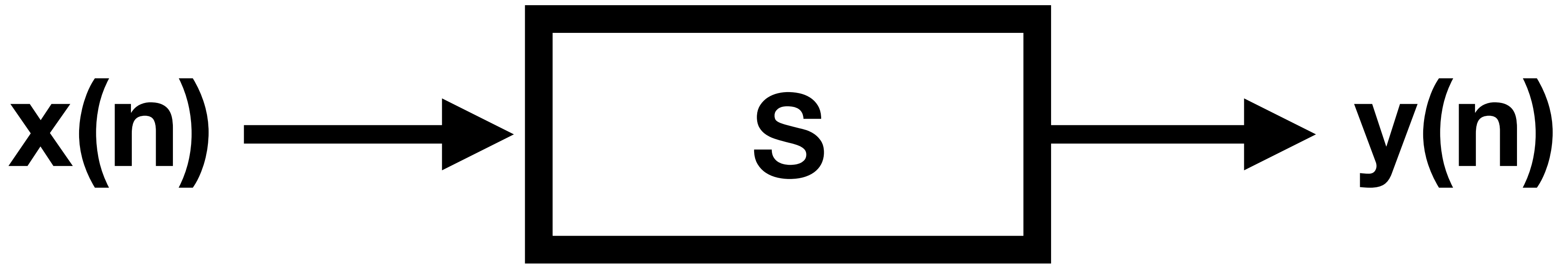


Input: 1 0.5 0 -0.5 -1



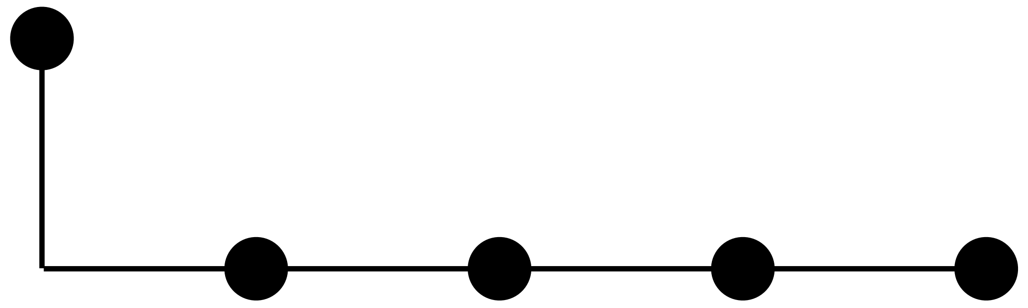
Output: 1 0.75 0.5 0.25 0

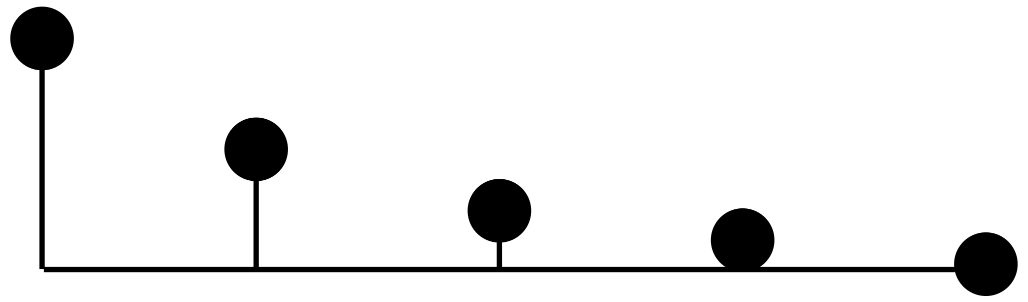




Input: 1 0 0 0 0 …

Output: 1 0.5 0.25 0.125 0.0625…





The *Impulse*

or *Kronecker delta*

$\partial_{t,k}$, with k = 0 in this case

The *Impulse Response* (IR) (AKA h(n)) of the system S…

but only for an impulse at t = 0.

What about at other times?

The Impulse Response

Impulse Response (IR): the output signal of a system S when an impulse is applied to the system input, often written as *h[n]*

What information does the IR give us?

What about for impulses at times other than k=0?

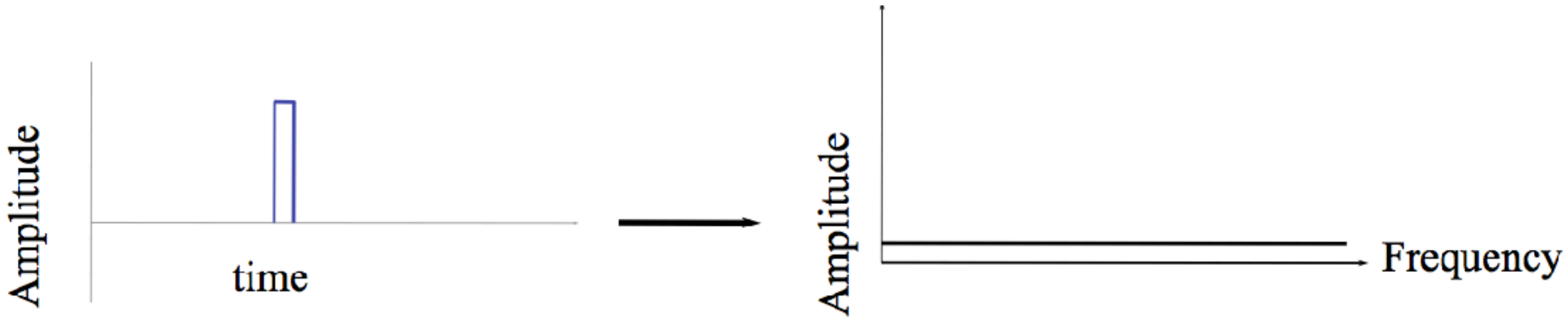
Answer: S is LTI, so we're all set! More specifically, we can define any input as time-shifted and scaled impulses, and because S is LTI, the output is equal to the sum of impulses, time-shifted and scales in an identical configuration (this is exceptionally important later).

Does the IR give us information about the full bandwidth of the digital system encapsulating S?

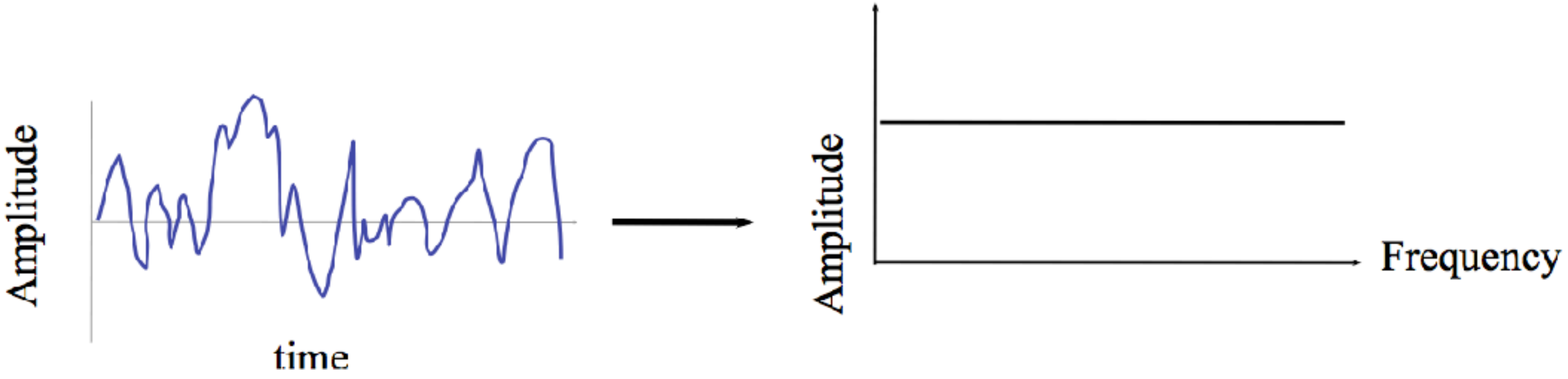
Answer: The impulse contains ***all frequencies***, so the IR informs us how the system responds at all frequencies.

Sound Typology (5)

- Non-periodic sounds have no pitch and tend to have continuous spectra, e.g. a short pulse (narrow in time, wide in frequency)



- The most complex sound is white noise (completely random)



Finite vs. Infinite Impulse Response

Finite Impulse Response (FIR)

A FIR filter has a fixed length IR (more explicitly, it becomes zero at some time t and stays that way!),

A FIR filter of order N as an IR of length N+1 samples. FIR filters may have feedforward, but no feedback.

Pros: Can have a flat (or variable) phase response, is more stable than IIR

Cons: is more computationally expensive

Example:

$y(n) = 1/2(x(n) + x(n-1))$ h(n) (as vector) = [0.5 0.5]

Finite vs. Infinite Impulse Response

Infinite Impulse Response (IIR)

An IIR filter has an IR that continues indefinitely. Many digital filters (any with feedback) are IIR.

Pros: Less computationally expensive

Cons: Less stability, nonlinear phase response

Example:

y(n) = 1/2(x(n) + y(n-1))   h(n) (as vector) = [0.5 0.25 0.125 0.0625 ….]

The biquadratic filter is also another IIR filter

z-Transform

We define ***Z*** such that powers of Z correspond to the number of delays or advances in a difference equation.

One sample of delay = Z. Two samples of delay = Z^2.

One sample of advance = Z^{-1}. Two samples of advance = Z^{-2}.

Ex.:

If we're given a sequence x(t) = [1 2 3 4], its z-Transform, $X(Z) = 1 + 2Z + 3Z^2 + 4Z^3$

Ex.:

If we're given a sequence x(t) = [1 3 0 -1 -5], its z-Transform, $X(Z) = 1 + 3Z - Z^3 - 5Z^4$

The Convolution Theorem

The convolution of f and g is written f*g.

An intuitive way to compute convolution by hand on 1-dimensional vectors is to think of it as a sliding window of multiplication and summation by a flipped 'kernel'. For example:

$x_1 = (1, 5)$

$x_2 = (1, 2, 3)$

$x_1 * x_2 = (1, 7, 13, 15)$

```
Convolution by hand (metaphor of sliding window):

5 1
  1 2 3
= 1 x 1 = 1

5 1
1 2 3
= 5 x 1 + 1 x 2 = 7

  5 1
1 2 3
= 5 x 2 + 1 x 3 = 13

    5 1
1 2 3
= 5 x 3 = 15
```

The Convolution Theorem

$x_1 = (1, 5)$

$x_2 = (1, 2, 3)$

z-Transforms of these:

$x_1(Z) = 1 + 5Z$

$x_2(Z) = 1 + 2Z + 3Z^2$

Multiplying their z-Transforms together:

$x_1(Z)x_2(Z) = 1 + 7Z + 13Z^2 + 15Z^3$

… which is the z-Transform of the convolution of x_1 and X_2!!

The Convolution Theorem

The Convolution Theorem: convolution in the time domain is equivalent to multiplication in the Z domain.

How does this help us?

Well, remember impulse responses of LTI systems?

A very important application of the convolution theorem: any signal x(n) that is input to an LTI system, the system's output y(x) is equal to the discrete convolution of the input signal x(n) and the system's impulse response (IR) h(n).

So, if x_1 = input signal and x_2 = a systems's impulse response, we can figure out its output!

For HW: Mini-Assignment II

Calculating impulse responses

z-Domain transforms

Practicing convolution of matrices

Some Octave commands + getting more knowledge of Complex Sinusoids