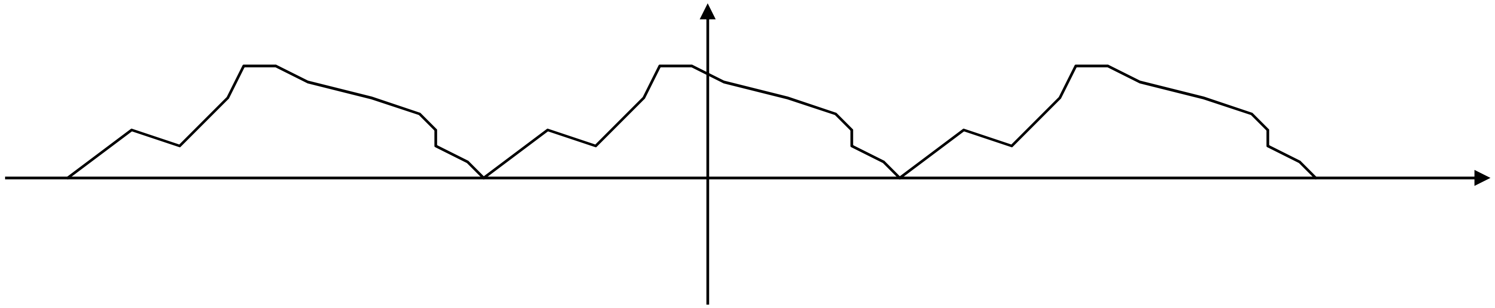
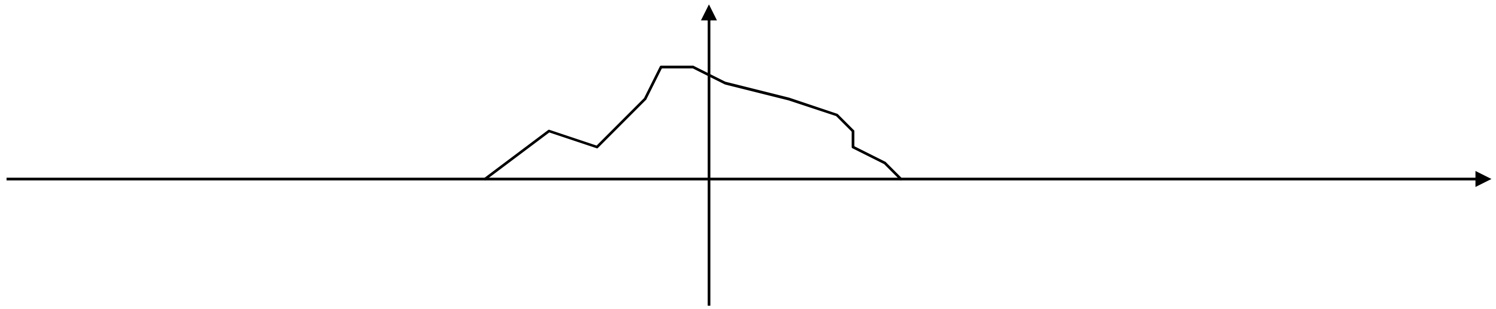
Summary

How the DFT sees its data

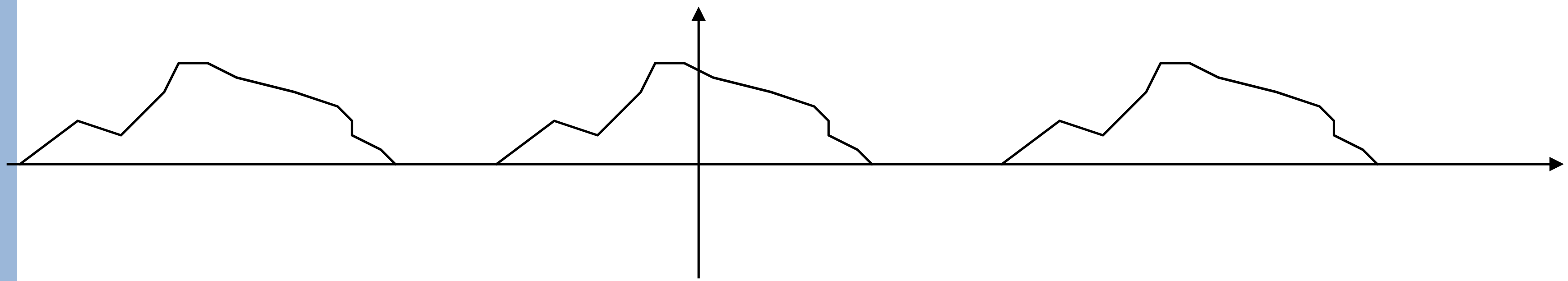


How the FT sees its data

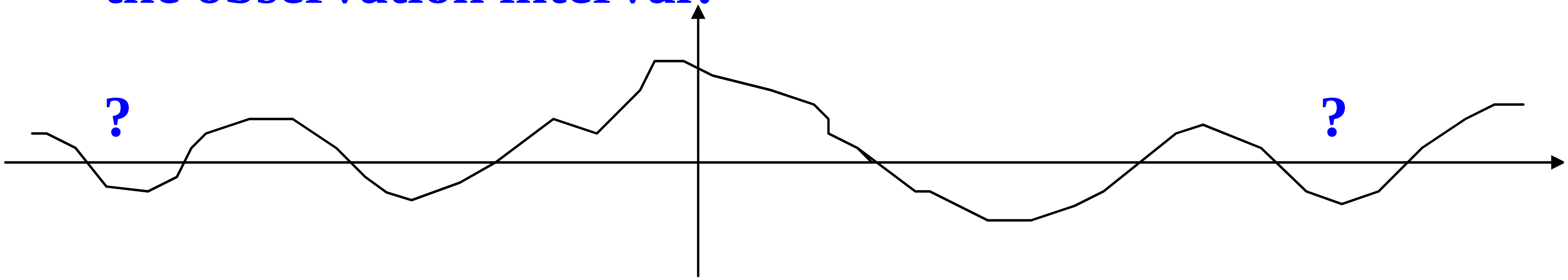


Summary

How the zero-padded DFT sees its data

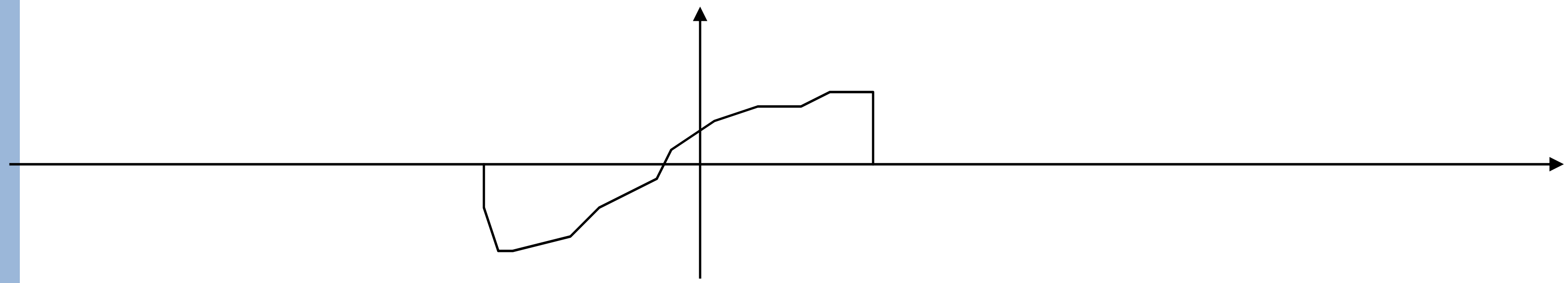


But what does the data really do outside the observation interval?



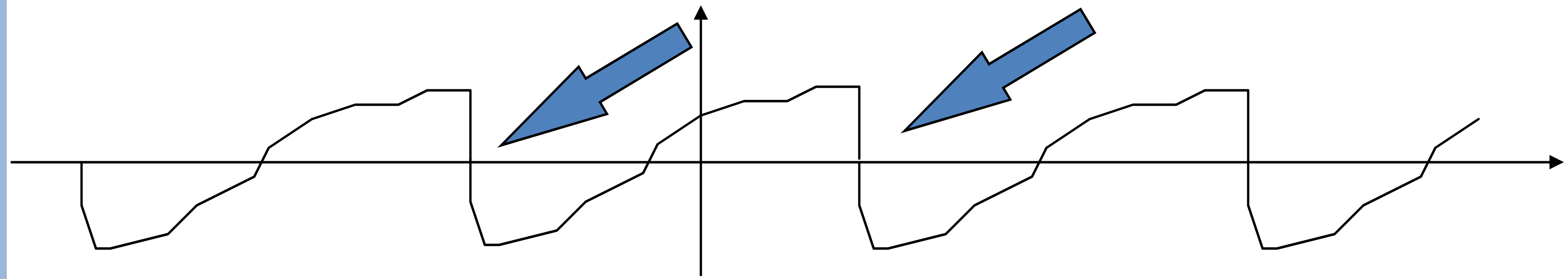
Need for Windows

What if these data look like this?



The DFT sees

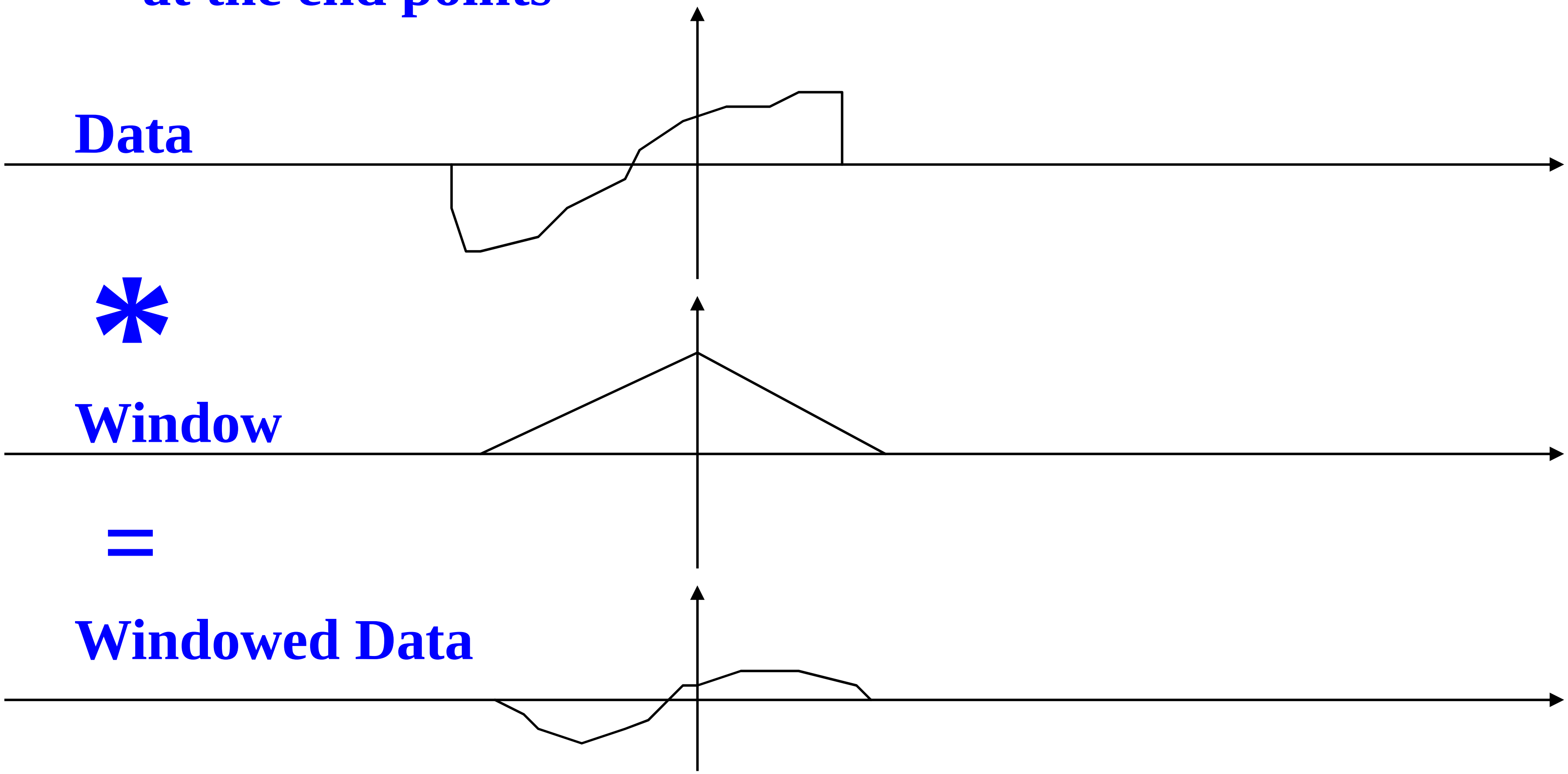
Discontinuities



Discontinuities lead to spurious components in DFT

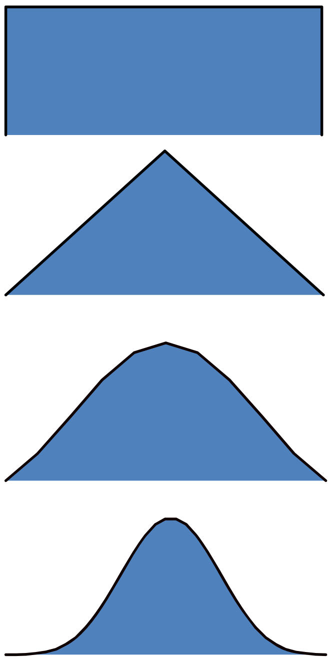
Solution: Data Windowing

Apply a weighting function that forces the data to zero at the end points



Window Functions

smpdemo
windodm



Rectangular (unwindowed)

Triangular or Bartlett

Hann (hanning) or raised cosine

Blackman-Harris

Windows will be discussed when we look at FIR filters

“On the use of windows for harmonic analysis with the discrete Fourier Transform,” Proc IEEE, V66, pp 51-83, 1978.

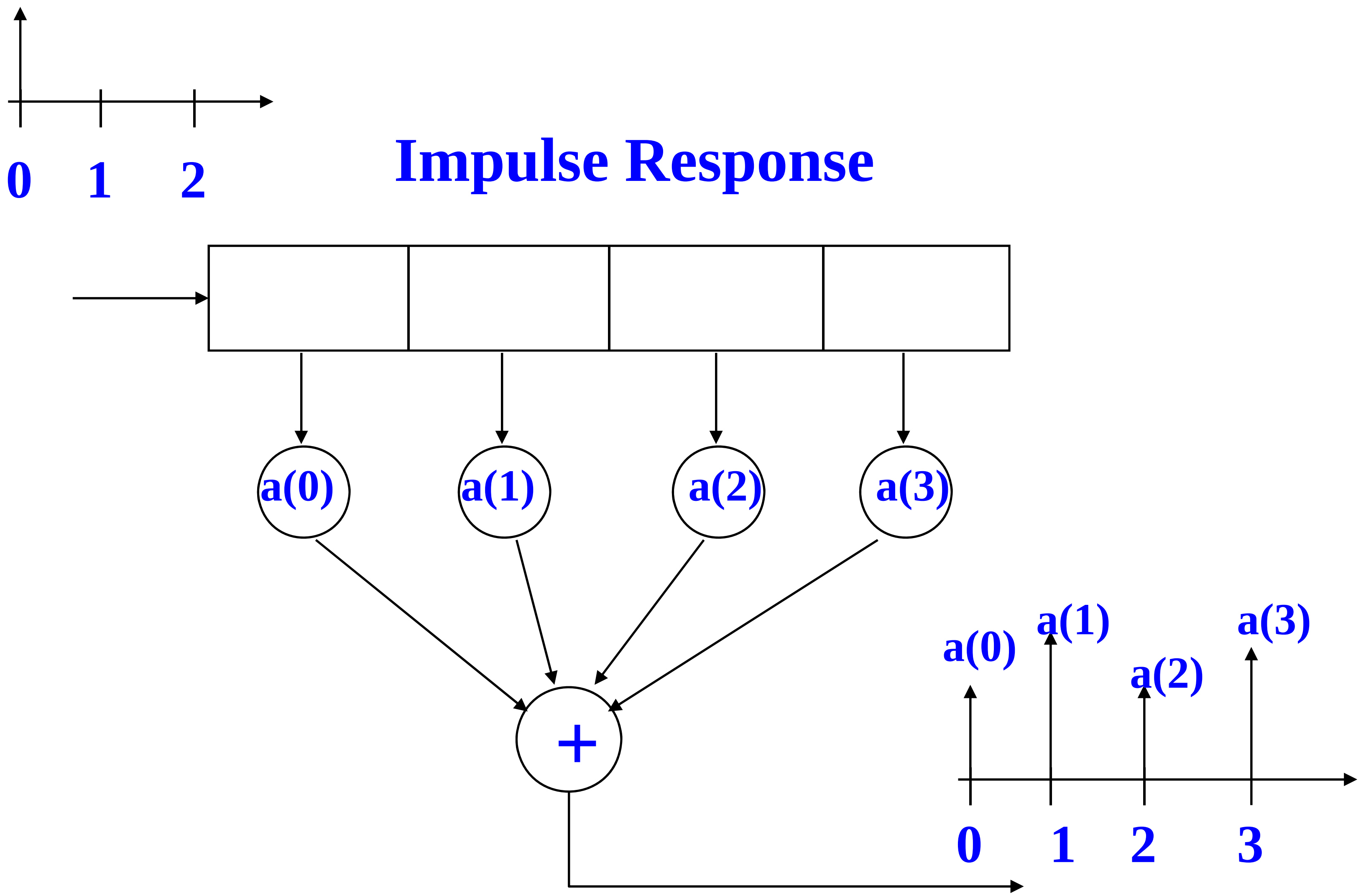
Vectors as Polynomials

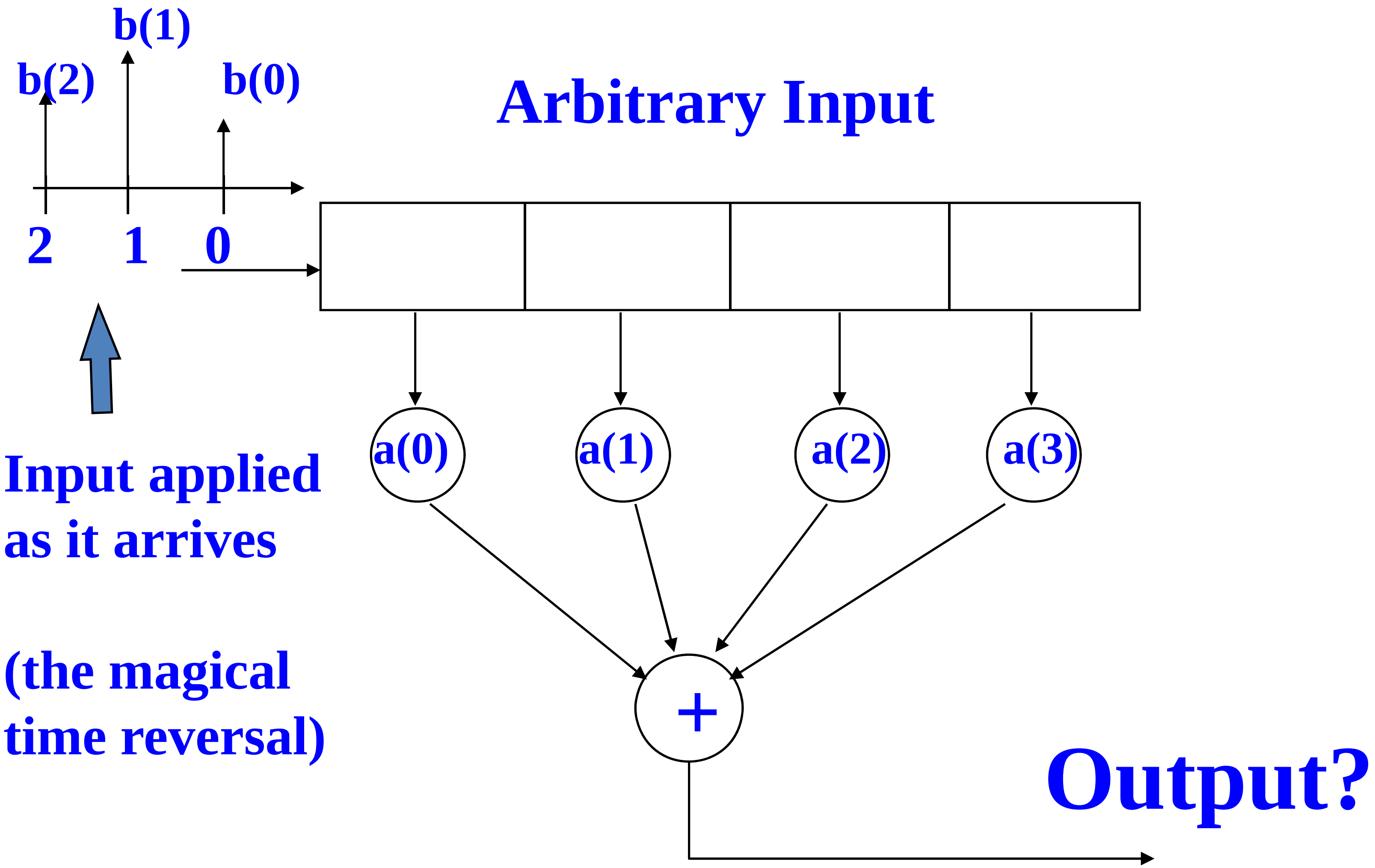
- There is an isomorphism between a vector and the coefficients of a polynomial.
- Thus it is often convenient to replace a vector *v* with its corresponding polynomial in *z*, say. This is called the Z-transform of *v*, but it is just a clever bookkeeping trick.

$$v = [a_0, a_1, \cdots, a_{n-1}]$$

$$V(z) = a_0 + a_1 z^{-1} + \cdots + a_{n-1} z^{n-1}$$

Convolution





b(1)
b(2)
b(0)
2
1
0
Arbitrary Input
a(0)
a(1)
a(2)
a(3)
Input applied
as it arrives
(the magical
time reversal)
+
Output?

N^2 multiplications

Conventional integer product

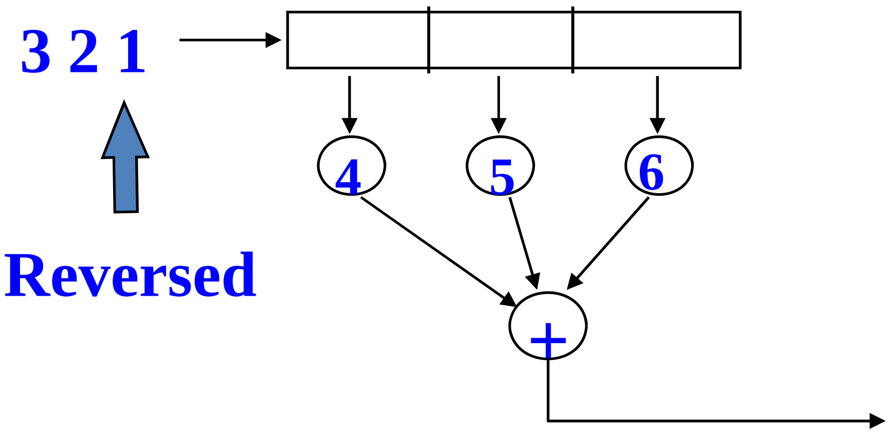
		1	2	3
		4	5	6
		7	3	8
	6	1	5	
4	9	2		
5	6	0	8	8

Convolution ⇨

No carry integer product

		1	2	3
		4	5	6
		6	12	18
	5	10	15	
4	8	12		
4	13	28	27	18
5	6	0	8	8

- We realise that convolution is just integer multiplication without the carry.



4	5	6		
	8	10	12	
		12	15	18
4	13	28	27	18
5	6	0	8	8

We can extend this trick and show that polynomial multiplication is equivalent to convolution of the corresponding sequences.

Convolve 1, 2, 3 with 4, 5, 6

1. Convert to polynomials in Z

$$X(z) = 1 + 2z^{-1} + 3z^{-2}$$

$$Y(z) = 4 + 5z^{-1} + 6z^{-2}$$

2. Multiply polynomials and gather terms in powers of *z*

$$Z(z) = X(z)Y(z)$$

$$= (1 + 2z^{-1} + 3z^{-2})(4 + 5z^{-1} + 6z^{-2})$$

$$= 4 + 13z^{-1} + 28z^{-2} + 27z^{-3} + 18z^{-4}$$

3. Convert polynomial back to sequence

4 13 28 27 18

Comments

- Clever bookkeeping trick
  - Recognise the one-to-one and onto correspondence between sequences and polynomials
- Note that multiplication of two polynomials is equivalent to convolution of the corresponding sequences. The power of z^{-1} in the polynomial corresponds to the output index, and the coefficient corresponds to the output value. The shifting and lining up of powers of z^{-1} in the product produces the equivalent convolution.

Z Transform of Finite Sequences

The finite sequence $h_0, h_1, \cdots, h_{N-1}$ denoted $\{h_n\}$ has an associated polynomial denoted $H(z)$ defined by

$$H(z) = \sum_{n=0}^{N-1} h_n z^{-n}$$

This polynomial is defined for all values of z.

We say that the Region of Convergence (ROC) is the finite plane.

Region of Convergence

- In DSP we generally deal with finite sequences. This means that we do not have to concern ourselves with the region of convergence (ROC) since we can always find the Z transform of a finite sequence and it is well-defined.
- However, sometimes we have to deal with infinite sequences and some infinite sequences do not converge. This is where the ROC is important and the Z transform of such an infinite sequence may only be valid for certain values of z.

Z Transform of Infinite Sequences

The infinite sequence $h_0, h_1, \cdots, h_N, \cdots$ denoted $\{h_n\}$ has an associated series denoted $H(z)$ defined by

$$H(z) = \sum_{n=0}^{\infty} h_n z^{-n}$$

This polynomial is not defined for all values of z. We say that the Region of Convergence (ROC) is the annulus containing values of z for which the above summation is finite.

Example: Infinite Sum

$$H(z) = \sum_{n=0}^{\infty} a^n z^{-n} = \sum_{n=0}^{\infty} (az^{-1})^n$$

$$= (az^{-1})^0 + (az^{-1})^1 + \cdots + (az^{-1})^{N-1} + \cdots$$

Geometric Series

$$G = g^0 + g^1 + \cdots$$

$$gG = \qquad g^1 + g^2 + \cdots$$

If G converges:

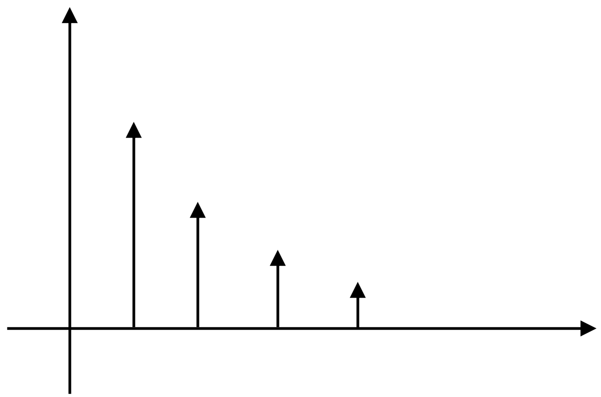
$$G - gG = 1$$

$$G(1 - g) = 1$$

$$G = \frac{1}{1-g}$$

Hence

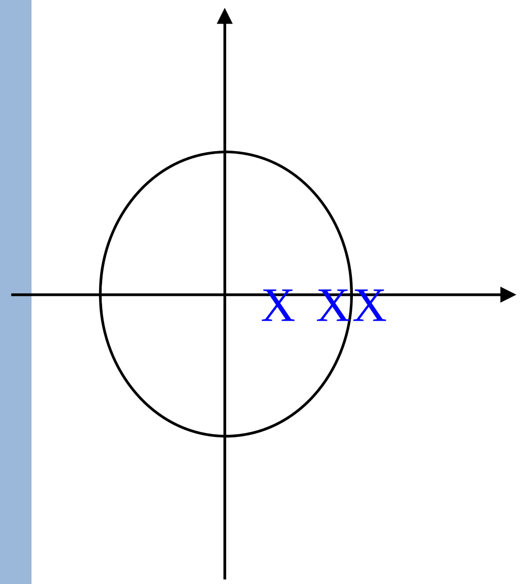
$$\mathrm{H}(z) = \frac{1}{1 - az^{-1}} = \frac{z}{z-a}$$



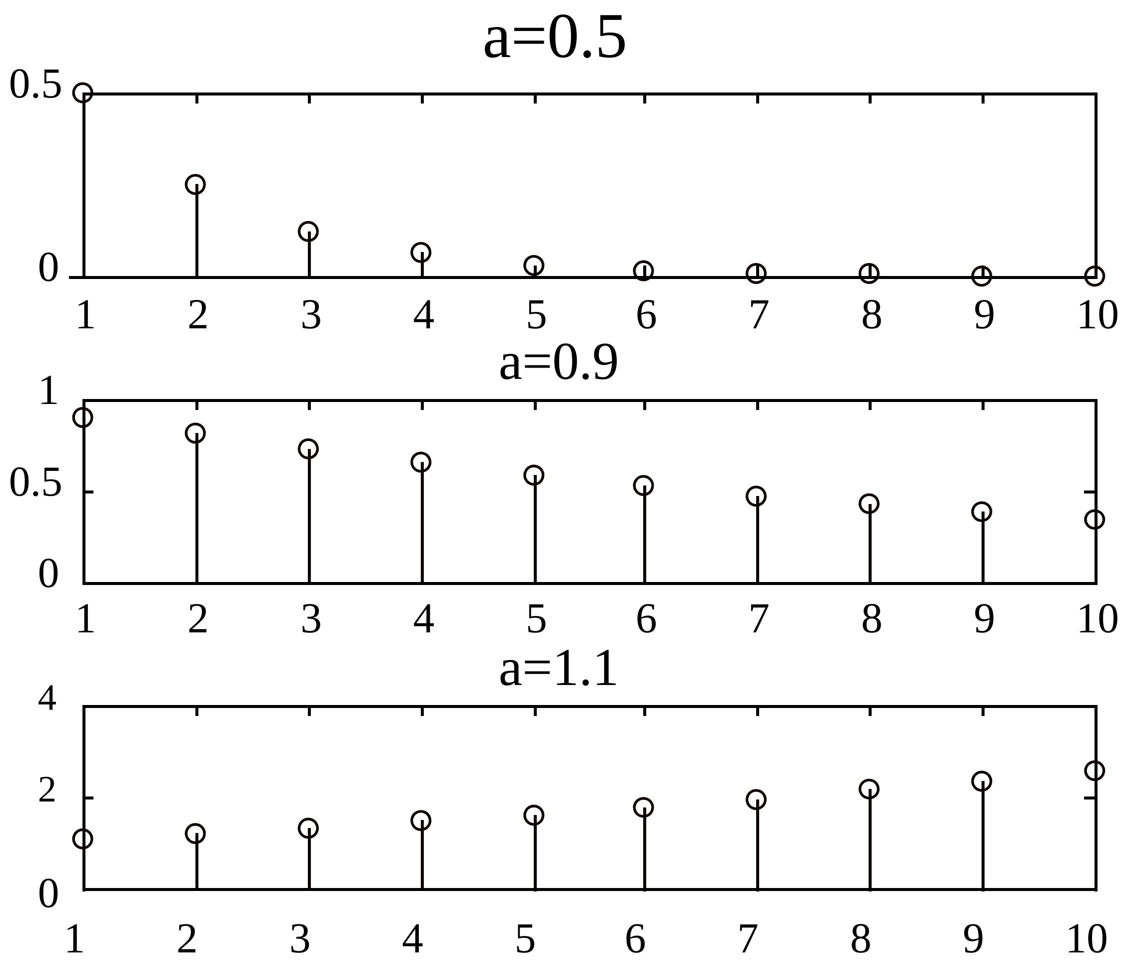
Pole at z=a

Convergence

Region of Convergence



$|z|<1$



Example: Finite Sum

$$H(z) = \sum_{n=0}^{N-1} a^n z^{-n} = \sum_{n=0}^{N-1} (az^{-1})^n$$

$$= (az^{-1})^0 + (az^{-1})^1 + \cdots + (az^{-1})^{N-1}$$

Geometric Series

$$S = g^0 + g^1 + \cdots + g^{N-1}$$

$$gS = \qquad g^1 + g^2 + \cdots + g^{N-1} + g^N$$

Hence

$$S - gS = 1 - g^N$$

$$S(1-g) = 1 - g^N$$

$$S = \frac{1-g^N}{1-g}$$

Hence

$$\mathrm{H(z)} = \frac{1 - (\mathrm{az}^{-1})^N}{1 - a\mathrm{z}^{-1}} = \frac{z^N - a^N}{z^{N-1}(z-a)}$$

If we let N go to infinity, we get the infinite series result as long as a^N goes to zero. That is, |a|<1

Direct Convolution

$$y(n)=\sum_{k=0}^{N-1}x(n-k)h(k)$$

$$Y(z)=X(z)H(z)=X(z)\sum_{n=0}^{N-1}h(n)z^{-n}$$

Sinusoidal Steady State Response

$$\mathrm{x(n)}=e^{j\phi n}$$

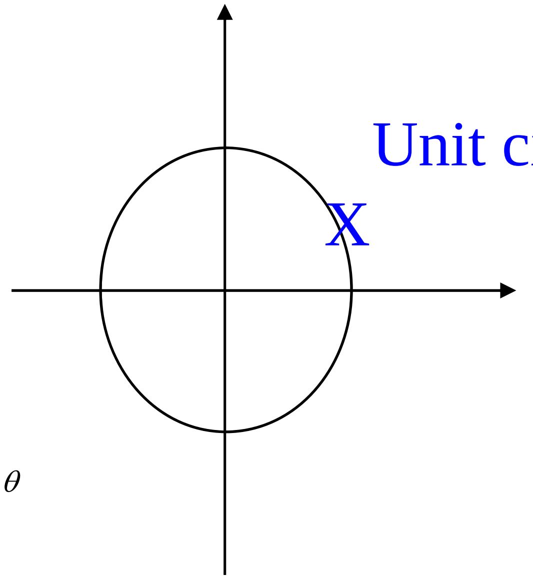
$$y(n)=\sum_{k=0}^{N-1}e^{j(n-k)\phi}h(k)$$

$$y(n)=e^{j\phi n}\sum_{k=0}^{N-1}h(k)e^{-jk\phi}$$

For large index N, the summation is the steady state gain H(θ)

$$H(\theta)=\sum_{n=0}^{\infty}h(n)e^{-j\theta n}=H(z)\text{ evaluated at }z=e^{j\theta}$$

Evaluate Z transform at points on the unit circle to determine the steady state sinusoidal gain. This is effectively what is done in the DFT.



Poles and Zeros

- How does all this relate to the poles and zeros in control theory? Are they the same?
- Yes.
- What we are really interested in is representing or approximating systems by polynomial functions.
- The same ideas arise in a number of fields. e.g., mathematics, control, time series forecasting, DSP, electronics. For this reason, the names associated with each technique may vary.

Polynomial Approximations to Functions

- Simple polynomial approximation to a function
  - Taylor series, Power series, Laurent Series, FIR Filter, Moving Average Filter, Non-Recursive Filter, Transversal Filter, Feedforward filter, Linear Phase Filter, All Zero Filter, Tapped Delay Line Filter
- Ratio of two polynomials
  - (Chebyshev-)Pade approximation, IIR Filter, ARMA Filter, Recursive Filter, Feedback Filter, Pole-Zero Filter, General Filter, Rational Transfer Function Model
- Special case where numerator polynomial is a constant
  - AR model, Autoregressive Filter, All Pole, Analog Filter

Different Models

All Zero

$$X(z) = a_0 + a_1 z^{-1} + \cdots + a_{N-1} z^{N-1}$$

Pole Zero

$$Y(z) = \frac{a_0 + a_1 z^{-1} + \cdots + a_{N-1} z^{N-1}}{b_0 + b_1 z^{-1} + \cdots + b_{M-1} z^{M-1}}$$

All Pole

$$Z(z) = \frac{A}{b_0 + b_1 z^{-1} + \cdots + b_{M-1} z^{M-1}}$$

Difference Equations

$$y(n) = [b_0 x(n) + b_1 x(n-1) + \cdots + b_M (n-M)] - [a_1 y(n-1) + a_2 y(n-2) + \cdots + a_M y(n-M)]$$

$$\sum_{k=0}^{M} a_k y(n-k) = \sum_{k=0}^{M} b_k x(n-k)$$

$$Y(z)\sum_{k=0}^{M} a_k z^{-k} = X(z)\sum_{k=0}^{M} b_k z^{-k}$$

This step is a bit confusing. It is helpful to remember that this is effectively a system of equations for all n, rather a single equation

$$T(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{M} b_k z^{-k}}{\sum_{k=0}^{M} a_k z^{-k}} = \frac{\sum_{k=0}^{M} b_k z^{M-k}}{\sum_{k=0}^{M} a_k z^{M-k}}$$

$$= k_0 \frac{(z-z_1)(z-z_2)\cdots(z-z_M)}{(z-p_1)(z-p_2)\cdots(z-p_M)}$$

zeros

poles

Residues

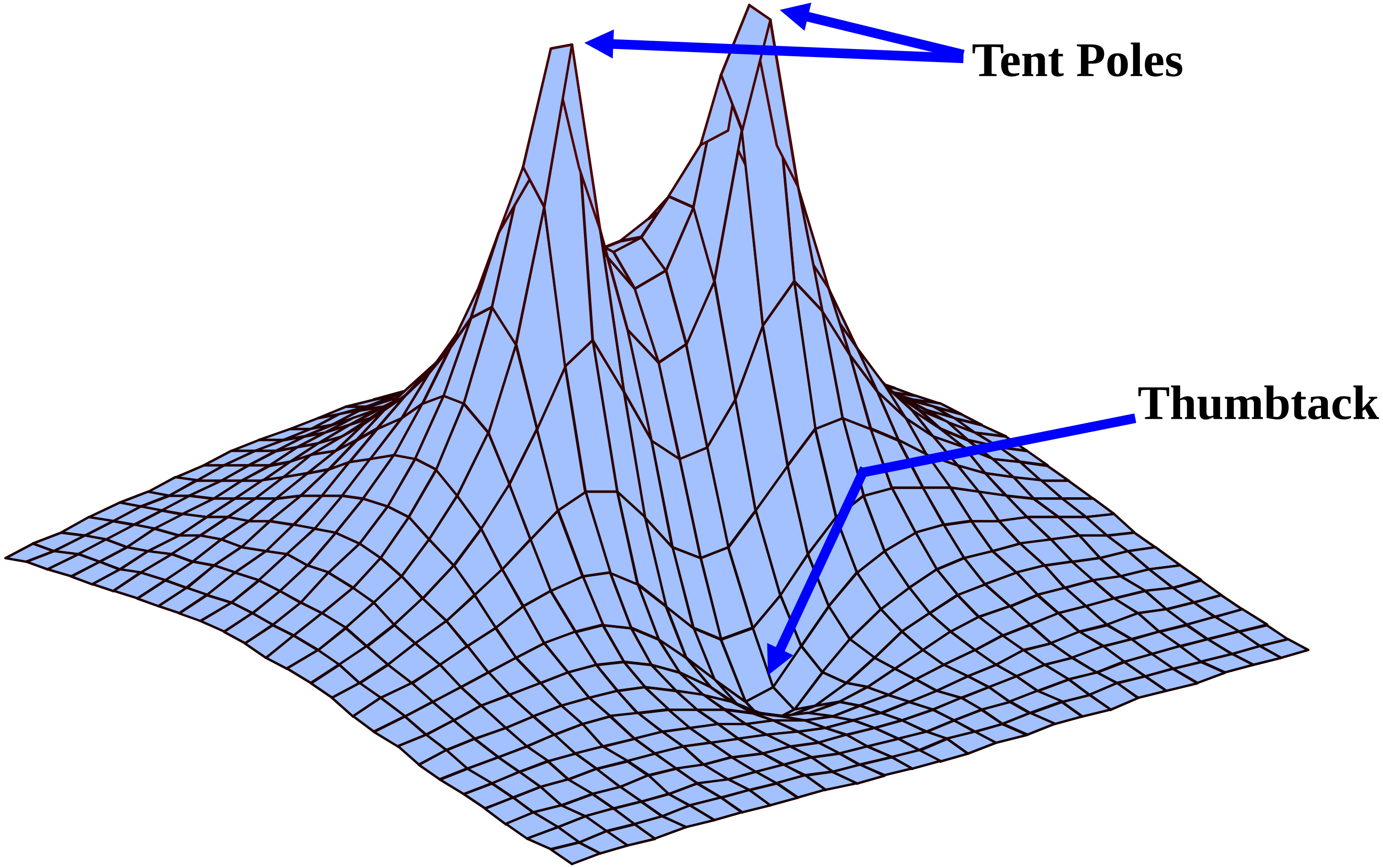
Any rational transfer function can be rewritten in the following form. This is called partial fraction expansion.

$$k_0 \frac{(z-z_1)(z-z_2)\cdots(z-z_M)}{(z-p_1)(z-p_2)\cdots(z-p_M)}$$

$$= \sum_{k=1}^{M} c_k \frac{z}{z-z_k}$$

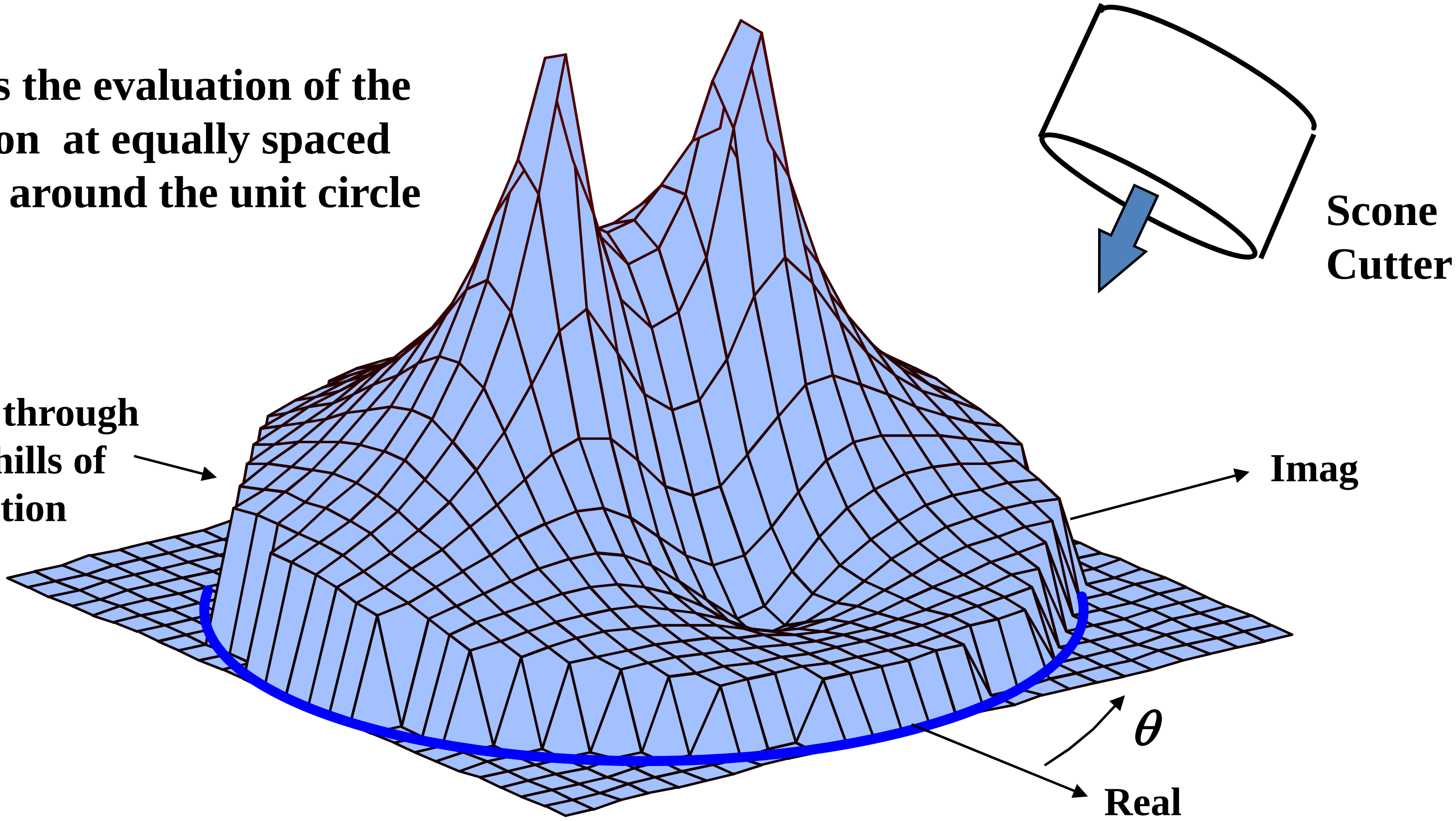
The c_k are called the residues of the system and are in fact the amplitudes of a given exponential response in the total impulse response. The z_k are the poles of the system.

2 Poles and a Zero



The DFT as a Scone Cutter

DFT is the evaluation of the function at equally spaced points around the unit circle



Summary

Notice the gain

- Roots of numerator are called zeros
  - These are the thumbtacks that hold the response down
- Roots of denominator are called poles
  - These are the tent “poles” that push the response up
- Equivalent information
  - Numerator and denominator polynomials
  - Zeros and poles
  - Residues and poles

poleplt2
poleplt3
poleplt4
poleplt5

FIR Filter Structures

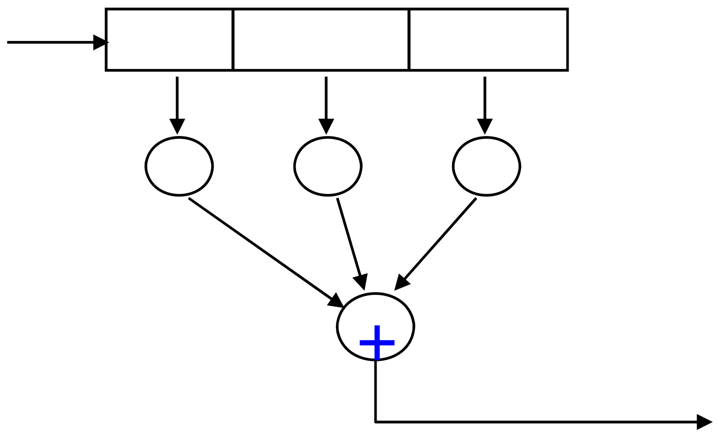
- If the Z transform has a finite number of weighting terms (numerator polynomial only), they can be stored explicitly within the mechanism which produces the running weighted sum.

Non-recursive

Feedforward

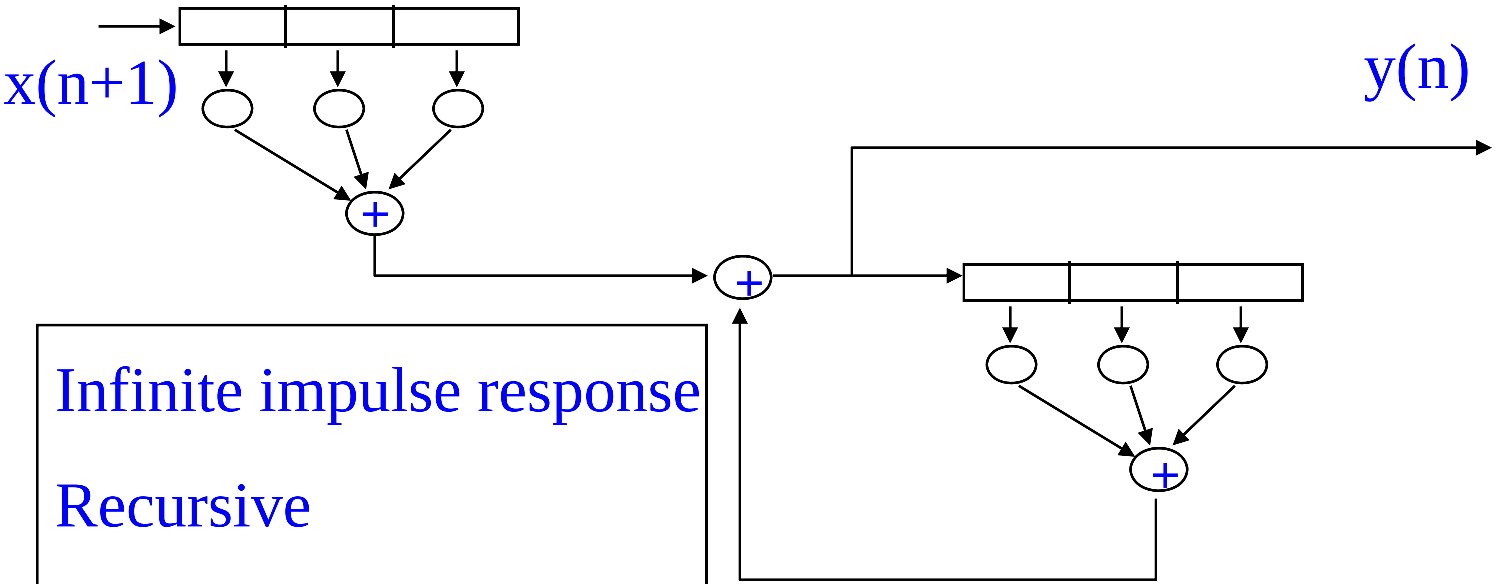
Finite Impulse Response

All Zero



IIR Filter Structures

- If the Z transform has an infinite number of terms (denominator polynomial), we need to have some feedback in the filter.



Infinite impulse response

Recursive

Pole-Zero