Professor Brendan Morris, SEB 3216, brendan.morris@unlv.edu

EE482: Digital Signal Processing Applications

Spring 2014
TTh 14:30-15:45 CBC C222

Lecture 02
DSP Fundamentals
14/01/21

http://www.ee.unlv.edu/~b1morris/ee482/

Outline

- Elementary Signals
- System Concepts
- Z-Transform
- Frequency Response

Elementary Digital Signals

- Digital signal
  - $x(n) \quad n \in \mathbb{Z}$
  - Deterministic – expressed mathematically (e.g. sinusoid)
  - Random – cannot be described exactly by equations (e.g. noise, speech)
- Unit impulse (Kronecker delta)
  - $\delta(n) = \begin{cases} 1, & n = 0 \\ 0, & n \neq 0 \end{cases}$
  - Basic building block of all digital signals
- Unit step
  - $u(n) = \begin{cases} 1, & n \geq 0 \\ 0, & n < 0 \end{cases} = \sum_{k=-\infty}^{n} \delta(k)$

Sinusoidal Signals

- Continuous
  - $x(t) = A\sin(\Omega t + \phi) = A\sin(2\pi f t + \phi)$
- Sampled
  - $x(n) = A\sin(\Omega nT + \phi) = A\sin(2\pi f nT + \phi)$
    - $\Omega = 2\pi f$
  - $x(n) = A\sin(\omega n + \phi) = A\sin(\mathrm{F}\pi n + \phi)$
    - $\omega = \Omega T$

Relationships Between Frequency Variables

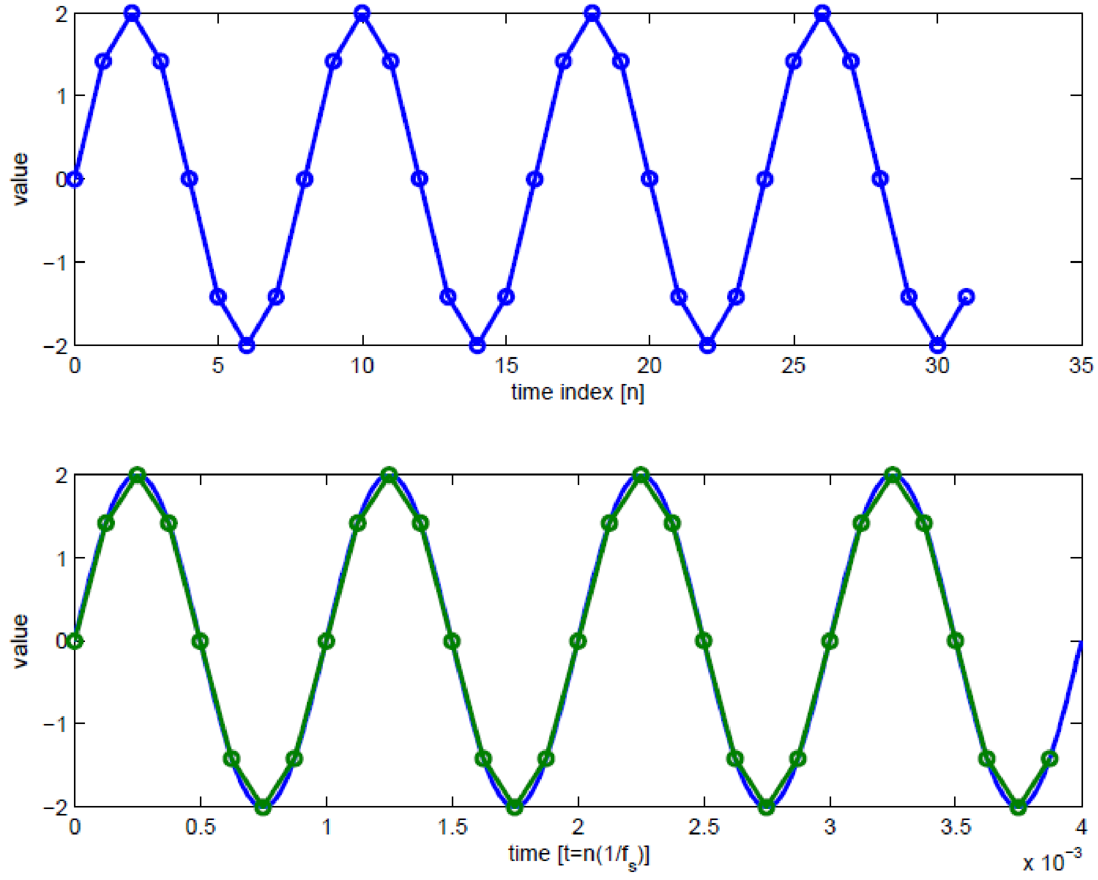
Table 2.1

Variable	Units	Relationships	Ranges
Ω	rads/sec	$\Omega = 2\pi f$	$-\infty < \Omega < \infty$
f	cycles/sec (Hz)	$f = \frac{\Omega}{2\pi} = \frac{\omega f_s}{2\pi}$	$-\infty < f < \infty$
ω	rads/sample	$\omega = \Omega T = \frac{2\pi f}{f_s}$	$-\pi \leq \omega \leq \pi$
F	cycles/sample	$F = \frac{f}{f_s/2} = \frac{\omega}{2}$	$-1 \leq F \leq 1$

- Normalized frequency measures
  - Note: max frequency for π or definition over a 2π interval
    - Consider $e^{j(\omega+2\pi k)}$

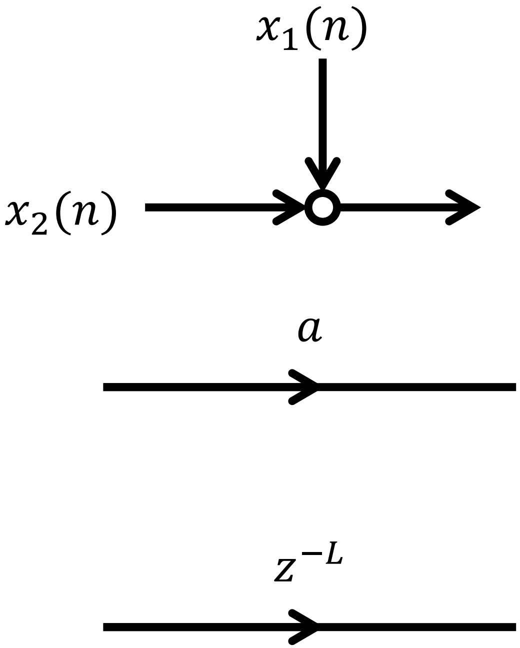
Example 2.1

```
A=2;
f=1000;
fs = 8000;

n=0:31;
w = 2*pi*f/fs;
x = A*sin(w*n);

h=figure;
%plot sampled sine
subplot(2,1,1)
plot(n,x,'-o','linewidth',2);
xlabel('time index [n]')
ylabel('value')
%plot analog sine
subplot(2,1,2)
t=0:1e-5:4e-3;
plot(t, A*sin(2*pi*f*t),
'linewidth',2);
hold all;
plot(n*(1/fs),x,'-
o','linewidth',2);
xlabel('time [t=n(1/f_s)]')
ylabel('value')
```



Block Diagram Representation

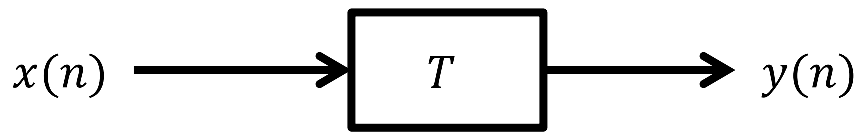
- Processing accomplished with 3 basic operations
- Addition
  - $y(n) = x_1(n) + x_2(n)$
- Multiplication
  - $y(n) = ax(n)$
- Time shift (delay)
  - $y(n) = x(n - L)$
  - Multiple delays can be implemented with a shift register (first-in, first-out buffer)(tapped delay line)



Multiplication in z domain

System Concepts

- Generic system



- Linearity
  - Additive and homogeneity (scaling) properties
  - $T\{ax_1(n) + bx_2(n)\} = ay_1(n) + by_2(n)$
- Time invariance
  - Shift in input causes corresponding shift in output
  - $y(n - n_0) = T\{x(n - n_0)\}$
    - To test
      - Find $y_1(n) = y(n - n_0)$ — replace n by n_0
      - Find $y_2(n) = T\{x(n - n_0)\}$ — response of system to shifted input

LTI Systems

- Impulse response

  - Output of LTI system $y(n) = h(n)$ to input $x(n) = \delta(n)$
- Convolution
  - Input-output relationship of LTI system

$$y(n) = x(n) * h(n) = \sum_{k=-\infty}^{\infty} x(k)h(n-k) = \sum_{k=-\infty}^{\infty} h(k)x(n-k)$$

General Difference Equation Systems

$$y(n) = \sum_{k=0}^{L-1} b_k x(n-k) - \sum_{k=0}^{M} a_k y(n-k)$$

- Infinite impulse response (IIR)
  - $h(n)$ non-zero as $n \to \infty$
- Finite impulse response (FIR)
  - $h(n)$ defined over finite set of n
  - Special case of above with $a_k = 0$
  - This system only has zeroes and poles at $z = 0$

- Causality
  - Output only depends on previous input
  - $h(n) = 0, \qquad n < 0$

Z-Transform

- Very useful computational tool for studying digital systems
- Definition

$$X(z) = \sum_{n=-\infty}^{\infty} x(n) z^{-n} \qquad z = re^{j\theta}$$

Complex variable

  - Has associated region of convergence (ROC)
    - Values of *z* where summation converges
- Useful summation formulas

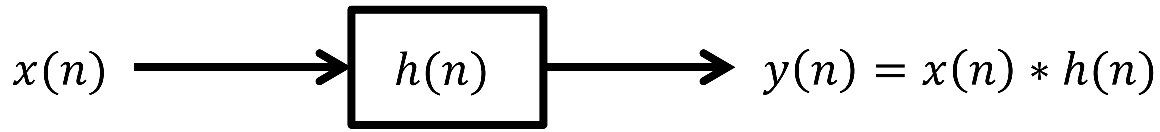
$$\sum_{k=0}^{N} \alpha^n = \frac{1-\alpha^{N+1}}{1-\alpha} \qquad \sum_{k=0}^{\infty} \alpha^n = \frac{1}{1-\alpha}$$

$|\alpha| < 1$

Z-Transform Properties

- Linearity
  - $\mathcal{Z}\{ax_1(n) + bx_2(n)\} = aX_1(z) + bX_2(z)$
- Time shift
  - $\mathcal{Z}\{x(n-k)\} = z^{-k}X(z)$
- Convolution
  - $x(n) = x_1(n) * x_2(n) \rightarrow X(z) = X_1(z)X_2(z)$
    - ROC = $R_{x1} \cap R_{x2}$

Transfer Functions



- $Y(z) = X(z)H(z)$
- $H(z) = \frac{Y(z)}{X(z)}$

- General polynomial form from difference equation

$$H(z) = \frac{\sum_{k=0}^{L-1} b_k z^{-k}}{1 + \sum_{k=1}^{M} a_k z^{-k}}$$

Poles and Zeros

$$H(z) = b_0 \frac{\prod_{k=1}^{L-1} (z - z_k)}{\prod_{k=1}^{M} (z - p_k)} = b_0 \frac{(z - z_1)(z - z_2) \ldots}{(z - p_1)(z - p_2) \ldots}$$

- Zeros
  - Roots of the numerator polynomial
  - Locations in z-plane that make output zero
- Poles
  - Roots of the denominator polynomial
  - Locations in z-plane that make output infinity (unstable)
    - System is considered unstable if the ROC doesn't contain the unit circle (no DTFT exists)
    - Causal system → poles should be inside unit circle

Example 2.10

- $H(z) = \frac{1}{L}\left[\frac{1-z^{-L}}{1-z^{-1}}\right]$
  - Notice this is a polynomial in z^{-1}
- Convert to polynomial in z to get all poles and zeros
- $H(z) = \frac{1}{L}\left[\frac{z^{L}-1}{z^{L}-z^{-L-1}}\right] = \frac{1}{L}\left[\frac{z^{L}-1}{z^{L-1}(z-1)}\right]$
  - Poles
    - $(z-1) = 0 \rightarrow z = 1$
    - $z^{L-1} = 0 \rightarrow$ L-1 poles at $z = 0$
  - Zeros
    - $z^{L} - 1 = 0 \rightarrow z_l = e^{j\left(\frac{2\pi}{L}\right)l}$
    - L zeros even spaced around unit circle

- Matlab
  - `fvtool([1 0 0 0 0 0 0 0 -1], [1 -1]);`

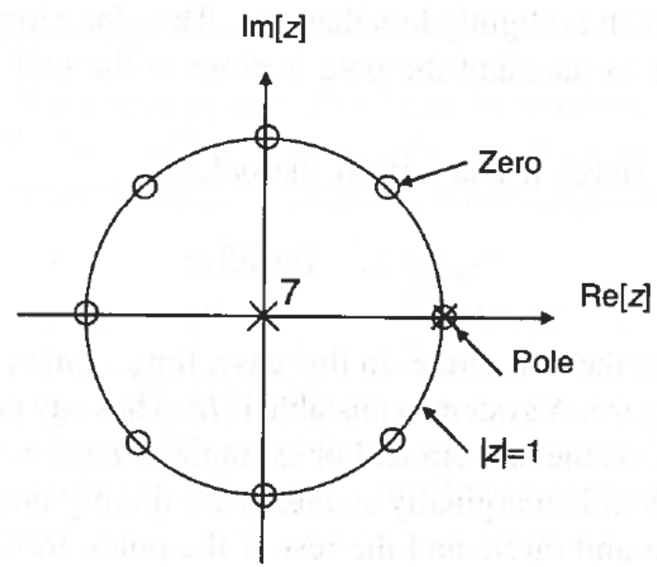


Figure 2.12 Pole–zero diagram of the moving-averaging filter, $L = 8$

Frequency Response

- Discrete-time Fourier transform (DTFT)

$$H(\omega) = H(z)|_{z=e^{j\omega}} = \sum_{n=-\infty}^{\infty} h(n)e^{-j\omega n}$$

  - Evaluate transfer function along the unit circle $|z| = |e^{j\omega}|$

$$H(\omega) = |H(\omega)|e^{\angle H(\omega)}$$

$$|H(\omega)| = \sqrt{H(\omega)H^*(\omega)} \quad \angle H(\omega) = \arctan\left(\frac{\operatorname{Im} H(\omega)}{\operatorname{Re} H(\omega)}\right)$$

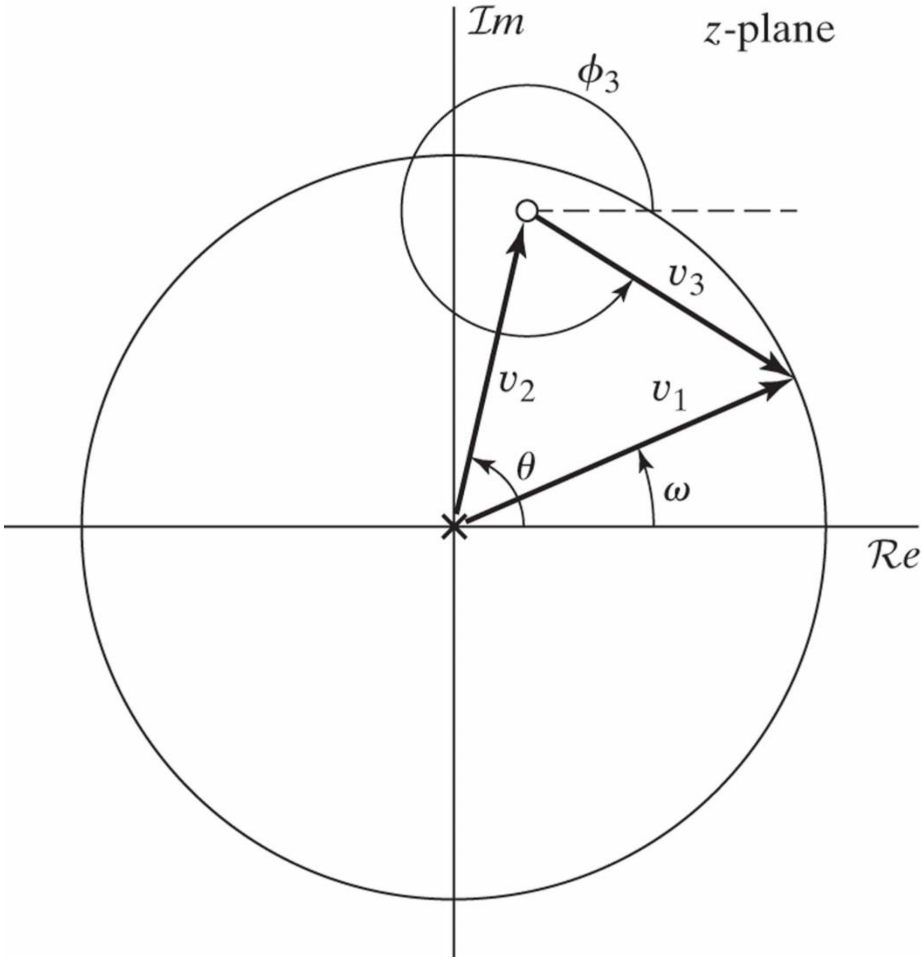
- Frequency response is periodic on 2π interval and symmetric
  - Only $[0, \pi]$ interval is required for evaluation

Graphical DTFT Interpretation

- Poles
  - $|H(\omega)|$ gets larger closer to θ
- Zeros
  - $|H(\omega)|$ gets smaller closer to θ

- What does a highpass filter look like?

- What does a lowpass filter look like?



Discrete Fourier Transform

- Notice the DTFT is a continuous function of ω
  - Requires an infinite number of samples to compute (infinite sum)
- DFT is a sampled version of the DTFT
  - Samples are taken at N equally spaced frequencies along unit circle
    - $\omega_k = \frac{2\pi k}{N}, k = 0,1,\ldots,N-1$

$$X(k) = X(\omega)\big|_{\omega=\frac{2\pi k}{N}} = \sum_{n=0}^{N-1} x(n) e^{-j\frac{2\pi k}{N}n}$$

  - n – time index
  - k – frequency index

DFT

$$X(k) = X(\omega)|_{\omega=\frac{2\pi k}{N}} = \sum_{n=0}^{N-1} x(n)e^{-j\frac{2\pi k}{N}n}$$

- DFT can be computed very efficiently with the fast Fourier transform (FFT)
- Frequency resolution of DFT
  - $\Delta_\omega = \frac{2\pi}{N}, \quad \Delta_f = \frac{f_s}{N}$
- Analog frequency mapping
  - $f_k = k\Delta_f = \frac{kf_s}{N}, \quad k = 0,1,\ldots,N-1$
  - Nyquist frequency $\frac{f_s}{2}$ corresponds to $k = \frac{N}{2}$

Example 2.16

```
N = 100;
A = 1;
f=1000;
fs = 10000;
n=0:N-1;
w = 2*pi*f/fs;

x = sin(w*n);
X = fft(x);
K = length(X);

h=figure;
subplot(2,1,1)
plot(0:K-1, 20*log10(abs(X)),
'linewidth', 2);
xlabel('freq index [k]');
ylabel('magnitude [dB]');
subplot(2,1,2)
%convert index to freq
f = (0:K-1) * fs/N;
plot(f, 20*log10(abs(X)),
'linewidth', 2);
xlabel('freq [Hz]');
ylabel('magnitude [dB]');
```