ECE 201: Introduction to Signal Analysis

Prof. Paris

Last updated: November 19, 2007

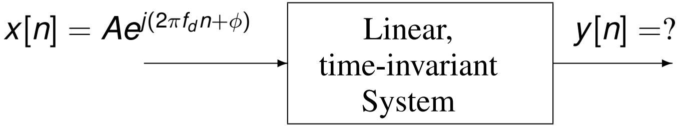
Part I

Frequency Response

Lecture: Introduction to Frequency Response

Introduction

- We have discussed:
  - Sinusoidal and complex exponential signals,
  - Spectrum representation of signals:
    - arbitrary signals can be expressed as the sum of sinusoidal (or complex exponential) signals.
  - Linear, time-invariant systems.
- Next: complex exponential signals as input to linear, time-invariant systems.



Example: 3-Point Averaging Filter

- Consider the 3-point averager:

$$y[n] = \frac{1}{3}\sum_{k=0}^{2} x[n-k] = \frac{1}{3} \cdot (x[n] + x[n-1] + x[n-2]).$$

- **Question:** What is the output $y[n]$ if the input is $x[n] = \exp(j2\pi f_d n)$?
  - Recall that f_d is the normalized frequency f/f_s; we are assuming the signal is oversampled, $|f_d| < \frac{1}{2}$
  - Initially, assume $A = 1$ and $\phi = 0$; generalization is easy.

Delayed Complex Exponentials

- The 3-point averager involves delayed versions of the input signal.
- We begin by assessing the impact the delay has on the complex exponential input signal.
- For

$$x[n] = \exp(j2\pi f_d n)$$

a delay by k samples leads to

$$\begin{aligned} x[n-k] &= \exp(j2\pi f_d(n-k)) \\ &= e^{j(2\pi f_d n - 2\pi f_d k)} = e^{j2\pi f_d n} \cdot e^{-j2\pi f_d k} \\ &= e^{j(2\pi f_d n + \phi_k)} = e^{j2\pi f_d n} \cdot e^{j\phi_k} \end{aligned}$$

where $\phi_k = -2\pi f_d k$ is the phase shift induced by the delay.

Average of Delayed Complex Exponentials

- Now, the output signal $y[n]$ is the average of three delayed sinusoids

$$\begin{aligned} y[n] &= \tfrac{1}{3}\textstyle\sum_{k=0}^{2} x[n-k] \\ &= \tfrac{1}{3}\textstyle\sum_{k=0}^{2} e^{j(2\pi f_d n - 2\pi f_d k)} \end{aligned}$$

- This expression involves the sum of sinusoids of the same frequency and the phasor addition rule applies:

$$y[n] = e^{j2\pi f_d n} \cdot \frac{1}{3} \sum_{k=0}^{2} e^{-j2\pi f_d k}.$$

- **Important Observation:** The output signal is a complex exponential of the same frequency as the input signal.
  - The amplitude and phase are different.

Frequency Response of the 3-Point Averager

- The output signal $y[n]$ can be rewritten as:

$$\begin{aligned} y[n] &= e^{j2\pi f_d n} \cdot \tfrac{1}{3} \sum_{k=0}^{2} e^{-j2\pi f_d k} \\ &= e^{j2\pi f_d n} \cdot H(f_d). \end{aligned}$$

where

$$\begin{aligned} H(f_d) &= \tfrac{1}{3} \sum_{k=0}^{2} e^{-j2\pi f_d k} \\ &= \tfrac{1}{3} \cdot (1 + e^{-j2\pi f_d} + e^{-j2\pi 2 f_d}) \\ &= \tfrac{1}{3} \cdot e^{-j2\pi f_d} (e^{j2\pi f_d} + 1 + e^{-j2\pi f_d}) \\ &= \frac{e^{-j2\pi f_d}}{3} (1 + 2\cos(2\pi f_d)). \end{aligned}$$

Examples

- Let us evaluate

$$H(f_d) = \frac{e^{-j2\pi f_d}}{3}(1 + 2\cos(2\pi f_d))$$

for different values of f_d.
- Let $x[n]$ be a complex exponential with $f_d = 0$.
  - Then, all samples of $x[n]$ equal to one.
- For $f_d = 0$, the frequency response $H(f_d) = 1$.
- Consequently, the output $y[n]$ is given by

$$y[n] = H(0) \cdot \exp(j2\pi 0n).$$

Thus, all output samples are equal to one.

Examples

- Let $x[n]$ be a complex exponential with $f_d = \frac{1}{3}$.
  - Then, the samples of $x[n]$ are the periodic repetition of $\{1, -\frac{1}{2}, -\frac{1}{2}\}$.
- For $f_d = \frac{1}{3}$, the frequency response $H(f_d) = 0$.
- Consequently, the output $y[n]$ is given by

$$y[n] = H(\frac{1}{3}) \cdot \exp(j2\pi\frac{1}{3}n) = 0.$$

Thus, all output samples are equal to zero.

General Complex Exponential

- Let $x[n]$ be a complex exponential of the from $Ae^{j(2\pi f_d n + \phi)}$.
    - This signal can be written as

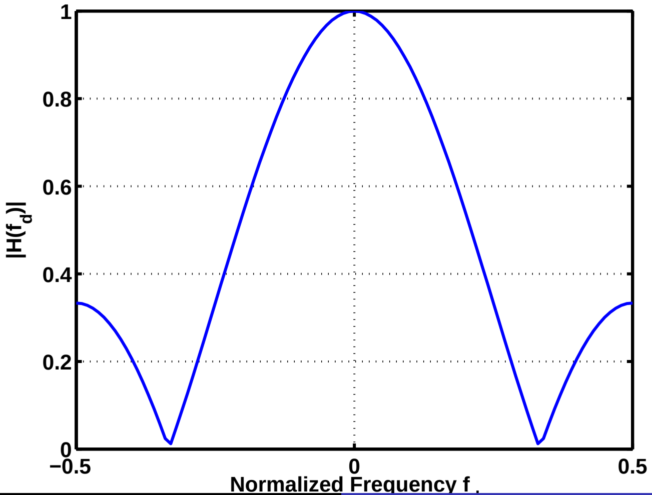
$$x[n] = X \cdot e^{j2\pi f_d n},$$

      where $X = Ae^{j\phi}$ is the *phasor* of the signal.
- Then, the output $y[n]$ is given by

$$y[n] = H(f_d) \cdot X \cdot \exp(j2\pi f_d n).$$

- **Interpretation:** The output is a complex exponential of the same frequency f_d
- The phasor for the output signal is the product $H(f_d) \cdot X$.

Plot of Frequency Response



Lecture: The Frequency Response of LTI Systems

Introduction

- We have demonstrated that for linear, time-invariant systems
  - the output signal $y[n]$
  - is the convolution of the input signal $x[n]$ and the impulse response $h[n]$.

$$\begin{aligned} y[n] &= x[n] * h[n] \\ &= \sum_{k=0}^{M} h[k] \cdot x[n-k] \end{aligned}$$

- **Question:** Find the output signal $y[n]$ when the input signal is $x[n] = A\exp(j(2\pi fn + \phi)$.

Response to a Complex Exponential

- **Problem:** Find the output signal $y[n]$ when the input signal is $x[n] = A\exp(j(2\pi fn + \phi)$.
- Output $y[n]$ is convolution of input and impulse response

$$\begin{aligned} y[n] &= x[n] * h[n] \\ &= \sum_{k=0}^{M} h[k] \cdot x[n-k] \\ &= \sum_{k=0}^{M} h[k] \cdot A\exp(j(2\pi f(n-k) + \phi) \\ &= A\exp(j(2\pi fn + \phi) \cdot \sum_{k=0}^{M} h[k] \cdot \exp(-j2\pi fk) \\ &= A\exp(j(2\pi fn + \phi) \cdot H(f) \end{aligned}$$

- The term

$$H(f) = \sum_{k=0}^{M} h[k] \cdot \exp(-j2\pi fk)$$

is called the Frequency Response of the system.

Interpreting the Frequency Response

The Frequency Response of an LTI system with impulse response $h[n]$ is

$$H(f) = \sum_{k=0}^{M} h[k] \cdot \exp(-j2\pi fk)$$

- **Observations:**
  - The response of a LTI system to a complex exponential signal is a complex exponential signal of the same frequency.
    - Complex exponentials are eigenfunctions of LTI systems.
  - When $x[n] = A\exp(j(2\pi fn + \phi))$, then $y[n] = x[n] \cdot H(f)$.
    - This is true only for exponential input signals, including complex exponentials!

Interpreting the Frequency Response

- **Observations:**
  - $H(f)$ is best interpreted in polar coordinates:
    $$H(f) = |H(f)| \cdot e^{j\angle H(f)}.$$
    - Then, for $x[n] = A\exp(j(2\pi fn + \phi)$
      $$\begin{aligned} y[n] &= x[n] \cdot H(f) \\ &= A\exp(j(2\pi fn + \phi) \cdot |H(f)| \cdot e^{j\angle H(f)} \\ &= (A \cdot |H(f)|) \cdot \exp(j(2\pi fn + \phi + \angle H(f)) \end{aligned}$$
  - The amplitude of the resulting complex exponential is the product $A \cdot |H(f)|$.
    - Therefore, $|H(f)|$ is called the gain of the system.
  - The phase of the resulting complex exponential is the sum $\phi + \angle H(f)$.
    - $\angle H(f)$ is called the phase of the system.

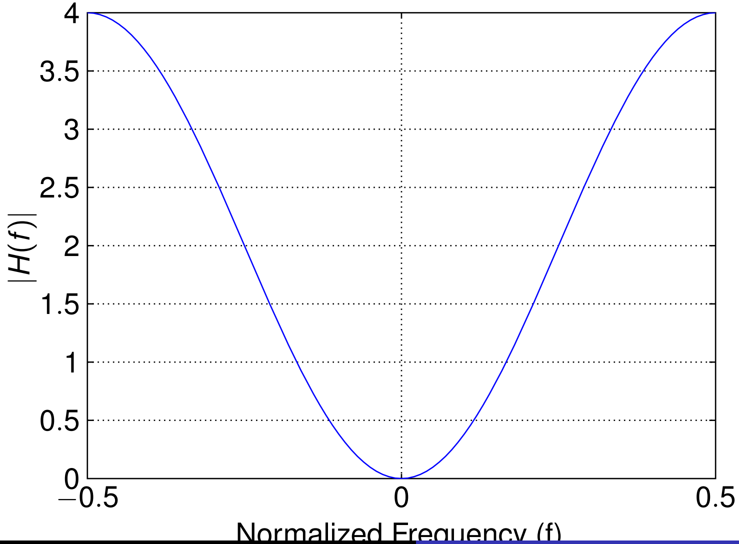
Example

- Let $h[n] = \{1, -2, 1\}$.
- Then,

$$\begin{aligned} H(f) &= \sum_{k=0}^{2} h[k] \cdot \exp(-j2\pi fk) \\ &= 1 - 2 \cdot \exp(-j2\pi f) + 1 \cdot \exp(-j2\pi f2) \\ &= \exp(-j2\pi f) \cdot (\exp(j2\pi f) - 2 + \exp(-j2\pi f)) \\ &= \exp(-j2\pi f) \cdot (2\cos(2\pi f) - 2). \end{aligned}$$

- Gain: $|H(f)| = |(2\cos(2\pi f) - 2|$

Example



Example

- The filter with impulse response $h[n] = \{1, -2, 1\}$ is a high-pass filter.
  - It rejects sinusoids with frequencies near $f = 0$,
  - and passes sinusoids with frequencies near $f = \frac{1}{2}$
- Note how the function of this system is much easier to describe in terms of the frequency response $H(f)$ than in terms of the impulse response $h[n]$.
- **Question:** Find the output signal when input equals $x[n] = 2\exp(j2\pi n/4 - \pi/2)$.
- **Solution:**

$$H(\frac{1}{4}) = \exp(-j2\pi\frac{1}{4}) \cdot (2\cos(2\pi\frac{1}{4}) - 2) = -2e^{-j\pi/2} = 2e^{j\pi/2}.$$

Thus,

$$y[n] = 2e^{j\pi/2} \cdot x[n] = 4\exp(j2\pi n/4).$$

Exercise

1. Find the Frequency Response $H(f)$ for the LTI system with impulse response $h[n] = \{1, -1, -1, 1\}$.
2. Find the output for the input signal $x[n] = 2\exp(j(2\pi n/3 - \pi/4))$.

Computing Frequency Response in MATLAB

```
function HH = FreqResp( hh, ff )
% FreqResp - compute frequency response of LTI system
%
% inputs:
%     hh - vector of impulse repsonse coefficients
%     ff - vector of frequencies at which to evaluate frequency response
%
% output:
%     HH - frequency response at frequencies in ff.
%
% Syntax:
%     HH = FreqResp( hh, ff )

HH = zeros( size(ff) );
for kk = 1:length(hh)
    HH = HH + hh(kk)*exp(j*2*pi*(kk-1)*ff);
end
```

Lecture: Comprehensive Example

Introduction

- **Objective:** Apply many of the things we covered to the solution of a “real-world” problem.
- **Problem:** Design and implement a decoder for “touch-tone” dialing.
- When dialing a digit on a telphone touch-pad a two-tone signal is emitted. These are called dual tone multifrequency (DTMF) signals.

Frequencies (Hz)	1209	1336	1477
697	1	2	3
770	4	5	6
852	7	8	9
941	*	0	#

Generating DTMF Signals

- Generating DTMF signals for a given digit is straightforward.
  - Determine the frequencies that the signal contains,
  - Generate two sinusoids of these frequencies,
  - Add sinusoids.
- Repeat for each digit to be dialed.
- The following MATLAB code extracts digits to be dialed from a string and forms the signal.
- Function signature:

```
function tones = dtmfdial( string , fs , tonedur , pausedur)
```

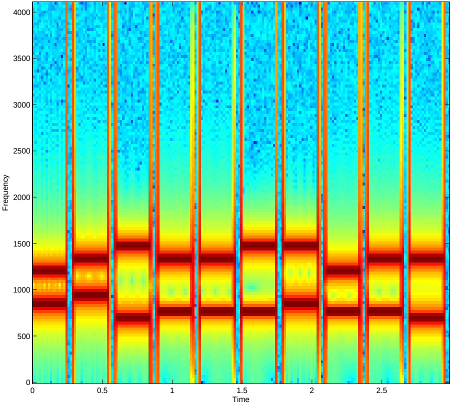
Parsing the Dial-String

```
% lookup table to translate numbers string into numbers
Digits = double('123456789*0#');
for kk=1:12
    ReverseDigits( Digits(kk) ) = kk;
end

RawNumbers = double( string );
numbers = ReverseDigits( RawNumbers );

% ensure numbers are integers between 1 and 12
numbers = round( numbers ); % silently discard fractional part
if ( min( numbers ) < 1 || max( numbers ) > 12 )
    error( 'input numbers must be integers between 1 and 12' );
end
```

Generating the DTMF Signal

```
% convert durations to number of samples
Ntone  = round( fs*tonedur );
Npause = round( fs*pausedur);

% figure out how long the output signal will be
Nnumbers = length( numbers );
Nsamples = Nnumbers*(Ntone + Npause);

tones = zeros(1, Nsamples );
pause = zeros(1, Npause);

% associate numbers with DTMF pairs, record normalized frequencies!
dtmfpairs = ...
    [ 697 697 697 770 770 770 852 852 852 941 941 941;
      1209 1336 1477 1209 1336 1477 1209 1336 1477 1209 1336 1477 ]/fs;

% loop over all numbers
for kk = 1:length(numbers)
    Start = (kk-1)*(Ntone + Npause) + 1;
    End   = kk*(Ntone + Npause);

    freqs = dtmfpairs( :, numbers(kk) );
    currtone = 0.5* cos( 2*pi*freqs(1)*(0:Ntone-1) ) + ...
        0.5*cos( 2*pi*freqs(2)*(0:Ntone-1) );
    tones(Start:End) = [ currtone pause ];
end
```

Spectrogram of Signal



Plan for Recovering the Dial String

- Use bandpass-filters for each of the possible frequencies
  - **Intent:** Isolate the different tones.
- Detect the strongest two tones in each dialing period.
- Map tones to digits (decoding)

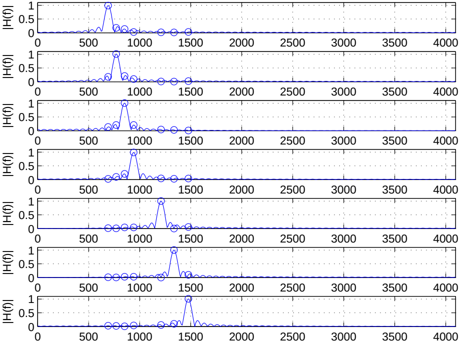
A simple bandpass filter

- We discussed the M-point average and showed that it has low-pass filter characteristics.
- Analogously, a simple bandpass filter centered at frequency f_0 has impulse response equal to
  - M samples of $2/M \cos(2\pi f_0 n)$.
- The following MATLAB function implements this design strategy.

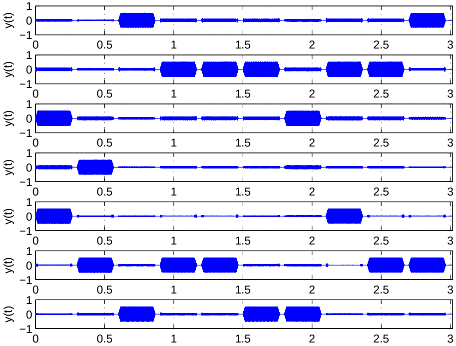
MATLAB function makeBPF.m

```
function hh = makeBPF( fd , N )
% makeBPF - design simple bandpass filter
%
% usage:
%   hh = makeBPF( fd , N )
%
% inputs:
%   fd - center frequency of pass band
%   N  - number of filter coefficients
%
% output:
%   hh - vector of filter coefficients

nn = -(N-1)/2:1:(N-1)/2;

hh = 2/N*cos(2*pi*fd*nn);
```

Frequency Response of Bandpass Filters



Output of Bandpass Filters



Detecting Tones

- The presence or absence is fairly easy to see in the output of the bandpass filters.
- However a single metric is needed to determine the presence or absence of each tone.
- Good strategy: for each filter output $k = 1, \ldots, 7$ and each dialing-period $m = 1, \ldots, 10$, compute the following score s

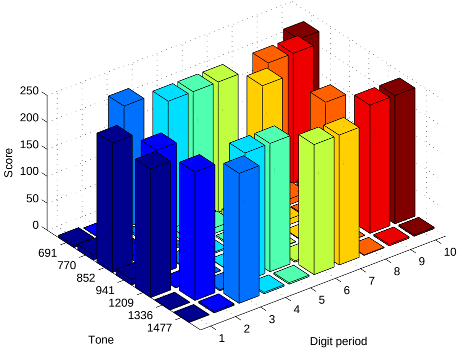
$$s(k,m) = \sum_{n \text{ in } m\text{-th dialing period}} (y_k[n])^2,$$

where y_k denotes the output of the k-th bandpass filter.
- MATLAB code for computing scores follows.

MATLAB code for Computing Scores

```
% decompose samples into periods for each number
Nnumbers = floor( length(xx)/(fs*(tonedur+pausedur)) );
NTonePlusPause = round(fs*(tonedur+pausedur));
NPause = round(fs*pausedur);

% score for each tone period: sum of squares in period
for nn=1:Nnumbers
    Startnn = (nn-1)*NTonePlusPause + 1 + floor(LBPF/2);
    Endnn = nn*NTonePlusPause - NPause - floor(LBPF/2);
    for kk = 1:length(DTMFFreqs)
        score(nn,kk) = sum( yy(kk,Startnn:Endnn).^2 );
    end
end
```

Scores


Score
250
200
150
100
50
0
691
770
852
941
1209
1336
1477
Tone
1
2
3
4
5
6
7
8
9
10
Digit period

Decoding

- It remains to find the two highest scores in each dialing period.
  - More specifically, the highest score among the lower four frequencies and the highest score among the higher three frequencies.
- The combination of frequencies yielding the highest score indicates which digit was dialed in that dialing period.
- MATLAB code follows

MATLAB code for Decoding Scores

```
% Decisions
% in each row of the score matrix find the biggest entry among the first
% four and final three columns
for nn=1:Nnumbers
    [ smax, imax_low(nn)] = max( score(nn, 1:4) );
    [ smax, imax_high(nn)] = max( score(nn, 5:7) );
end

% decode
% lookup table to translate numbers string into numbers
Digits = double('123456789*0#');      % table of ASCII codes for dial-pad

numbers = (imax_low-1)*3 + imax_high; % between 1 and 12
rawnumbers = Digits( numbers );       % ASCII codes for dialed numbers
decodedNumber = char( rawnumbers )    % convert to string
```