Digital Filters

ELEC 3004: Digital Linear Systems Signals & Controls
Dr. Surya Singh

Lecture 8

elec3004@itee.uq.edu.au
http://robotics.itee.uq.edu.au/~elec3004/

Lecture Schedule:

<table>
<thead>
<tr>
<th>Week</th>
<th>Date</th>
<th>Lecture Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4-Mar</td>
<td>Introduction &amp; Systems Overview</td>
</tr>
<tr>
<td>1</td>
<td>6-Mar</td>
<td>Linear Dynamical Systems</td>
</tr>
<tr>
<td>2</td>
<td>11-Mar</td>
<td>Signals as Vectors &amp; Systems as Maps</td>
</tr>
<tr>
<td>2</td>
<td>13-Mar</td>
<td>Signals</td>
</tr>
<tr>
<td>3</td>
<td>18-Mar</td>
<td>Sampling &amp; Data Acquisition &amp; Anti-aliasing Filters</td>
</tr>
<tr>
<td>3</td>
<td>20-Mar</td>
<td>Sampling &amp; Data Acquisition &amp; Anti-aliasing Filters</td>
</tr>
<tr>
<td>4</td>
<td>25-Mar</td>
<td>System Analysis &amp; Convolution</td>
</tr>
<tr>
<td>4</td>
<td>27-Mar</td>
<td>Convolution &amp; FT</td>
</tr>
<tr>
<td>5</td>
<td>1-Apr</td>
<td>Frequency Response &amp; Filter Analysis</td>
</tr>
<tr>
<td>5</td>
<td>3-Apr</td>
<td>Filters</td>
</tr>
<tr>
<td>6</td>
<td>8-Apr</td>
<td>Discrete Systems &amp; Z-Transforms</td>
</tr>
<tr>
<td>6</td>
<td>10-Apr</td>
<td>Z-Transforms</td>
</tr>
<tr>
<td>7</td>
<td>15-Apr</td>
<td>Introduction to Digital Control</td>
</tr>
<tr>
<td>7</td>
<td>17-Apr</td>
<td>Feedback</td>
</tr>
<tr>
<td>8</td>
<td>29-Apr</td>
<td>Digital Filters</td>
</tr>
<tr>
<td>9</td>
<td>1-May</td>
<td>Digital Control</td>
</tr>
<tr>
<td>9</td>
<td>6-May</td>
<td>Digital Control</td>
</tr>
<tr>
<td>9</td>
<td>8-May</td>
<td>Digital Control</td>
</tr>
<tr>
<td>10</td>
<td>13-May</td>
<td>Stability of Digital Systems</td>
</tr>
<tr>
<td>10</td>
<td>15-May</td>
<td>Stability</td>
</tr>
<tr>
<td>11</td>
<td>20-May</td>
<td>State-Space</td>
</tr>
<tr>
<td>11</td>
<td>22-May</td>
<td>Controllability &amp; Observability</td>
</tr>
<tr>
<td>12</td>
<td>27-May</td>
<td>PID Control &amp; System Identification</td>
</tr>
<tr>
<td>12</td>
<td>29-May</td>
<td>Digital Control System Hardware</td>
</tr>
<tr>
<td>13</td>
<td>3-Jun</td>
<td>Applications in Industry &amp; Information Theory &amp; Communications</td>
</tr>
<tr>
<td>13</td>
<td>5-Jun</td>
<td>Summary and Course Review</td>
</tr>
</tbody>
</table>

April 29, 2014
Linear, Discrete Dynamical Systems

Linear Difference Equations

\[ u_k = f(e_0, \ldots, e_k, u_0, \ldots, u_{k-1}). \]
\[ u_k = -a_1 u_{k-1} - a_2 u_{k-2} - \cdots - a_n u_{k-n} + b_0 e_k + b_1 e_{k-1} + \cdots + b_m e_{k-m}. \]

\[ \nabla u_k = u_k - u_{k-1} \] (first difference),
\[ \nabla^2 u_k = \nabla u_k - \nabla u_{k-1} \] (second difference),
\[ \nabla^n u_k = \nabla^{n-1} u_k - \nabla^{n-1} u_{k-1} \] (nth difference).

\[ u_k = u_k, \]
\[ u_{k-1} = u_k - \nabla u_k, \]
\[ u_{k-2} = u_k - 2\nabla u_k + \nabla^2 u_k. \]

\[ a_2 \nabla^2 u_k - (a_1 + 2a_2) \nabla u_k + (a_2 + a_1 + 1) u_k = b_0 e_k. \]
Assume a form of the solution

\[ z^k : \]
- k: “order of difference”
- k: delay

\[ A_z^k = A_z^{k-1} + A_z^{k-2}. \]

\[ 1 = z^{-1} + z^{-2} \]

\[ z^2 = z + 1. \]

\[ 29 \text{ April 2014} - \text{ELEC 3004: Systems} 5 \]

DT Causality & BIBO Stability

- Causality:
  \[ h[n] = 0, \ n < 0 \]
  \[ \rightarrow y[n] = \sum_{k=0}^{\infty} h[k] x[n-k] \quad \text{or} \quad y[n] = \sum_{k=-\infty}^{n} x[k] h[n-k] \]

- Input is Causal if: \[ x[n] = 0, \ n < 0 \]

- Then output is Causal:
  \[ y[n] = \sum_{k=0}^{n} h[k] x[n-k] = \sum_{k=0}^{n} x[k] h[n-k] \]

- And, DT LTI is BIBO stable if:
  \[ \sum_{k=-\infty}^{\infty} |h[k]| < \infty \]

\[ 29 \text{ April 2014} - \text{ELEC 3004: Systems} 6 \]
Impulse Response (Graphically)

Let’s define the impulse response, \( h[n] \), as the result of applying an LTI system to the unit impulse:

\[
\begin{align*}
\delta[n] & \quad \text{LTI System} \quad h[n] \\
\delta[n-k] & \quad \text{LTI System} \quad h[n-k]
\end{align*}
\]

By time invariance, we know

\[
\begin{align*}
\delta[n-k] & \quad \text{LTI System} \quad h[n-k]
\end{align*}
\]

And by linearity, we know

\[
\alpha_1\delta[n-k_1] + \alpha_2\delta[n-k_2] \quad \text{LTI System} \quad \alpha_1h[n-k_1] + \alpha_2h[n-k_2]
\]

\[
u[n] = \sum_{k=-\infty}^{\infty} u[k]h[n-k]
\]

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

\[
\infty \text{ matrix } \times \infty \text{ vector}
\]

How do you multiply an infinite matrix?

- First let’s multiply circulant matrices...
  - A circulant matrix can be described completely by its first row or column
  
  \[
  A = \begin{bmatrix}
  a_0 & a_1 & a_2 & \cdots & a_{n-1} \\
  a_{n-1} & a_0 & a_1 & \cdots & a_{n-2} \\
  a_{n-2} & a_{n-1} & a_0 & \cdots & a_{n-3} \\
  \vdots & \vdots & \vdots & \ddots & \vdots \\
  a_1 & a_2 & a_3 & \cdots & a_0
  \end{bmatrix}
  \]
  
  \[
  \begin{bmatrix}
  h & Zh & Z^2h & \cdots & Z^{N-1}h
  \end{bmatrix}
  \]

  \[
  Z: \text{ Shift operator}
  \]

- Multiply by \( u[k] \)
  
  \[
  \begin{bmatrix}
  h & Zh & Z^2h & \cdots & Z^{N-1}h
  \end{bmatrix} \begin{bmatrix}
  u[0] \\
  u[1] \\
  u[2] \\
  \vdots \\
  u[N-1]
  \end{bmatrix} = \sum_{k=0}^{N-1} u[k]Z^k h
  \]

  \[
  \therefore \text{ For circulant matrices, matrix multiplication reduces to a weighted combination of shifted impulse responses}
  \]
## Two Types of Systems

- **Linear shift-invariant:**
  \[ y = \sum_{k=0}^{N-1} u[k] Z^k h \]
  
  **Z: Shift operator**
  
  \[ Z \cdot [u_0, u_1, u_2, \ldots, u_{n-1}]^T = [u_{n-1}, u_0, u_1, u_2, \ldots, u_{n-2}]^T \]

- **Linear time-invariant system**
  
  \[ y = \sum_{k=-\infty}^{\infty} u[k] R^k h \]
  
  **R: Unit delay operator**
  
  \[ R \cdot [\ldots, u_0, u_1, u_2, \ldots]^T = [\ldots, u_{-1}, u_0, u_1, \ldots]^T \]

## Impulse Response of Both Types

- **Linear shift-invariant:**
  \[ y[n] = \frac{1}{2} u[n - 1] + \frac{1}{2} u[n] \]
  
  \[ y[-1] = 0 \]
  \[ y[0] = \frac{1}{2} \]
  \[ y[1] = \frac{1}{2} \]
  \[ y[2] = 0 \]
  
  \[ \vdots \]

- **Linear time-invariant system**
  \[ y[n] = \frac{1}{2} y[n - 1] + u[n] \]
  
  \[ h[-1] = 0 \]
  \[ h[0] = 1 \]
  \[ h[1] = \frac{1}{2} \]
  \[ h[2] = \frac{1}{4} \]
  
  \[ \vdots \]
  
  \[ h[n] = \begin{cases} 
  0 & n < 0 \\
  \left(\frac{1}{2}\right)^n & n \geq 0 
  \end{cases} \]
Digital Filters

- Wikipedia Says:
  A digital filter is a system that performs mathematical operations on a sampled, discrete-time signal to reduce or enhance certain aspects of that signal.

- Basically we have a transfer function or … a difference equation

In the Z-domain:

\[
H(z) = \frac{B(z)}{A(z)} = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + \ldots + b_N z^{-N}}{1 + a_1 z^{-1} + a_2 z^{-2} + \ldots + a_M z^{-M}}
\]

- This is a recursive form with inputs (Numerator) and outputs (Denominator)
  ➔ “IIR infinite impulse response” behaviour
- If the denominator is made equal to unity (i.e. no feedback)
  ➔ then this becomes an FIR or finite impulse response filter.
Digital Filters Types

**FIR**

From $H(z)$:

$$H(\omega) = h_0 + h_1 e^{-j\omega} + \ldots + h_n e^{-j(n-1)\omega}$$

$$= \sum_{t=0}^{n-1} h_t \cos t\omega - i \sum_{t=0}^{n-1} h_t \sin t\omega$$

→ Filter becomes a “multiply, accumulate, and delay” system:

$$y(t) = \sum_{\tau=0}^{n-1} h_\tau u(t-\tau)$$

$$y[n] = h_0 x[n] + h_1 x[n-1] + \ldots + h_N x[n-N]$$

**IIR**

- **Impulse response** function that is non-zero over an infinite length of time.

FIR Properties

- Require no feedback.
- Are inherently stable.
- They can easily be designed to be linear phase by making the coefficient sequence symmetric.
- Flexibility in shaping their magnitude response.
- Very Fast Implementation (based around FFTs)

- The main disadvantage of FIR filters is that considerably more computation power in a general purpose processor is required compared to an IIR filter with similar sharpness or selectivity, especially when low frequency (relative to the sample rate) cutoffs are needed.
FIR as a class of LTI Filters

- Transfer function of the filter is

\[ H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{M} b_k z^{-k}}{1 + \sum_{k=1}^{N} a_k z^{-k}} \]

- Finite Impulse Response (FIR) Filters: \( N = 0 \), no feedback

⇒ From \( H(z) \):

\[ H(\omega) = h_0 + h_1 e^{-i\omega} + \cdots + h_{n-1} e^{-i(n-1)\omega} \]
\[ = \sum_{t=0}^{n-1} h_t \cos t\omega - i \sum_{t=0}^{n-1} h_t \sin t\omega \]

∴ \( H(\omega) \) is periodic and conjugate
∴ Consider \( \omega \in [0, \pi] \)

FIR Filters

- Let us consider an FIR filter of length \( M \)
- Order \( N = M - 1 \) (**watch out!**)
- Order ⇒ number of delays

\[ y(n) = \sum_{k=0}^{M-1} b_k x(n - k) = \sum_{k=0}^{M-1} h(k) x(n - k) \]

[Diagram of FIR filter]
FIR Impulse Response

Obtain the impulse response immediately with $x(n) = \delta(n)$:

$$h(n) = y(n) = \sum_{k=0}^{M-1} b_k \delta(n-k) = b_n$$

- The impulse response is of finite length $M$ (good!)

- FIR filters have only zeros (no poles) (as they must, $N=0$ !!)
  - Hence known also as all-zero filters

- FIR filters also known as feedforward or non-recursive, or transversal filters


FIR & Linear Phase

- The phase response of the filter is a linear function of frequency

- Linear phase has constant group delay, all frequency components have equal delay times. ∴ No distortion due to different time delays of different frequencies

- FIR Filters with:
  $$\sum_{n=-\infty}^{\infty} h[n] \cdot \sin(\omega \cdot (n - \alpha) + \beta) = 0$$
FIR & Linear Phase → Four Types

- Type 1: most versatile
- Type 2: frequency response is always 0 at $\omega=\pi$ (not suitable as a high-pass)
- Type 3 and 4: introduce a $\pi/2$ phase shift, 0 at $\omega=0$ (not suitable as a high-pass)

### Impulse response

<table>
<thead>
<tr>
<th>$h(n)$ = $h(M - 1 - n)$</th>
<th>Odd</th>
<th>$H(\omega)$</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>$h(n) = h(M - 1 - n)$</td>
<td>Even</td>
<td>$e^{-j\omega(M-1)/2} \sum_{k=1}^{M-3/2} h\left(\frac{M-1}{2} - k\right) \cos\left(\omega k\right)$</td>
<td>1</td>
</tr>
<tr>
<td>$h(n) = -h(M - 1 - n)$</td>
<td>Odd</td>
<td>$e^{-j\omega(M-1)/2} \sum_{k=1}^{M-1/2} h\left(\frac{M-1}{2} - k\right) \sin\left(\omega k\right)$</td>
<td>2</td>
</tr>
<tr>
<td>$h(n) = h(M - 1 - n)$</td>
<td>Even</td>
<td>$e^{-j\omega(M-1)/2} \sum_{k=1}^{M-1/2} h\left(\frac{M-1}{2} - k\right) \sin\left(\omega k - \frac{\pi}{2}\right)$</td>
<td>3</td>
</tr>
<tr>
<td>$h(n) = -h(M - 1 - n)$</td>
<td>Odd</td>
<td>$e^{-j\omega(M-1)/2} \sum_{k=1}^{M-1/2} h\left(\frac{M-1}{2} - k\right) \sin\left(\omega k + \frac{\pi}{2}\right)$</td>
<td>4</td>
</tr>
</tbody>
</table>

### FIR Filter Design

- How to get all these coefficients?

$$H(\omega) = h_0 + h_1 e^{-i\omega} + \cdots + h_{n-1} e^{-i(n-1)\omega}$$

**FIR Design Methods:**

1. Impulse Response Truncation
   - Simplest
   - Undesirable frequency domain-characteristics, not very useful
2. Windowing Design Method
   - Simple
   - Not optimal (not minimum order for a given performance level)
3. Optimal filter design methods
   - “More optimal”
   - Less simple…
FIR Filter Design & Operation
Ex: Lowpass FIR filter

- Set Impulse response (order \( n = 21 \))
- “Determine” \( h(t) \)
  - \( h(t) \) is a 20 element vector that we’ll use to as a weighted sum

- FFT (“Magic”) gives Frequency Response & Phase

Why is this “hard”? Looking at the Low-Pass Example

\[
H_d(\omega) = \begin{cases} 
1 & \text{if } |\omega| \leq \omega_c \\
0 & \text{if } \omega_c < |\omega| < \pi 
\end{cases}
\]

- Why is this hard?
  - Shouldn’t it be “easy” ??
    … just hit it with some FFT “magic” and then keep the bands we want and then hit it with some Inverse-FFT “supermagic”???

  - Remember we need a “system” that does this “rectangle function” in frequency

  - Let’s consider what that means…
    - It basically suggests we need an Inverse FFT of a “rectangle function”
Flashback: Fourier Series & Rectangular Functions

\[ \mathcal{F}^{-1}\left\{ \text{rect}\left(\frac{\omega}{2}\right) \right\} = \frac{\text{sinc}(t)}{\pi} \]

See:
- Table 7.1 (p. 702) Entry 17
- Table 9.1 (p. 852) Entry 7

Ref:
- http://cnx.org/content/m32899/1.8/

Ref:
- http://cnx.org/content/m26719/1.1/
- http://www.wolframalpha.com/input/?i=IFFT%28sinc%28f%29%29

29 April 2014 - ELEC 3004: Systems

Flashback: Fourier Series & Rectangular Functions [2]

- The sinc function might look familiar
  - This is the frequency content of a square wave (box)

- This also applies to signal reconstruction!
  ➔ Whittaker–Shannon interpolation formula
  - This says that the “better way” to go from Discrete to Continuous (i.e. D to A) is not ZOH, but rather via the sinc!

\[ x(t) = \sum_{n=-\infty}^{\infty} x[n] \cdot \text{sinc}\left(\frac{t-nT}{T}\right) \]
\[ H_d(\omega) = \begin{cases} 1 & \text{if } |\omega| \leq \omega_c \\ 0 & \text{if } \omega_c < |\omega| < \pi \end{cases} \]

Has impulse response:

\[ h_d(n) = \frac{\omega_c \sin \omega_c n}{\pi} \frac{1}{\omega_c n} \]

Thus, to filter an impulse train with an ideal low-pass filter use:

\[ x(t) = \left( \sum_{n=-\infty}^{\infty} x[n] \cdot \delta(t-nT) \right) * \text{sinc} \left( \frac{t}{T} \right) \]

\[ \text{However}!! \]

a sinc is non-causal and infinite in duration

And, this cannot be implemented in practice \( \otimes \)

\[ \because \text{we need to know all samples of the input, both in the past and in the future} \]

Plan 0: Impulse Response Truncation

Maybe we saw this coming...

\[ \hat{h}(n) = \frac{\sin (n\omega_c)}{\pi n} \text{ for } |n| \leq M \text{ and } 0 \text{ otherwise} \]

- Ripples in both passband/stopband and the transition not abrupt (i.e., a transition band).
- As \( M \to \infty \), transition band \( \to 0 \) (as expected!)
FIR Filters: Window Function Design Method

- Windowing: a generalization of the truncation idea

- There many, many “window” functions:
  - Rectangular
  - Triangular
  - Hanning
  - Hamming
  - Blackman
  - Kaiser
  - Lanczos
  - Many More… (see: http://en.wikipedia.org/wiki/Window_function)

Some Window Functions [1]

1. Rectangular

\[ w(n) = 1 \]
Windowing and its effects/terminology

Lathi, Fig. 7.45

29 April 2014 - ELEC 3004: Systems 29

Some More Window Functions …

2. Triangular window

\[ w(n) = 1 - \left| \frac{n - \frac{N-1}{2}}{\frac{N-1}{2}} \right| \]

- And Bartlett Windows
  - A slightly narrower variant with zero weight at both ends:

\[ w(n) = 1 - \left| \frac{n - \frac{N-1}{2}}{\frac{N-1}{2}} \right| \]
Some More Window Functions…

3. Generalized Hamming Windows

\[ w(n) = \alpha - \beta \cos \left( \frac{2\pi n}{N-1} \right) \]

\[ \rightarrow \text{Hanning Window} \]

\[ w(n) = 0.5 \left( 1 - \cos \left( \frac{2\pi n}{N-1} \right) \right) \]

\[ \rightarrow \text{Hamming’s Window} \]

\[ \rightarrow \alpha = 0.54, \beta = 1 - \alpha = 0.46 \]

Some More Window Functions…

4. Blackman–Harris Windows

- A generalization of the Hamming family,
- Adds more shifted sinc functions for less side-lobe levels

\[ w(n) = a_0 - a_1 \cos \left( \frac{2\pi n}{N-1} \right) + a_2 \cos \left( \frac{4\pi n}{N-1} \right) - a_3 \cos \left( \frac{6\pi n}{N-1} \right) \]
5. Kaiser window
   - A DPSS (discrete prolate spheroidal sequence)
   - Maximize the energy concentration in the main lobe

\[ w(n) = \frac{I_0(\pi \alpha \sqrt{1 - \frac{4n}{N-1} - 1})}{I_0(\pi \alpha)} \]

   - Where: \( I_0 \) is the zero-th order modified Bessel function of the first kind, and usually \( \alpha = 3 \).

Some More Window Functions…

Comparison of Alternative Windows – Time Domain
Adding Order
+ Transition and Smoothness
  – Increased Size

Comparison of Alternative Windows
**Frequency Domain**

Punukaya, Slide 94

Punukaya, Slide 91
Summary Characteristics of Common Window Functions

<table>
<thead>
<tr>
<th>No.</th>
<th>Window ( w(t) )</th>
<th>Mainlobe Width</th>
<th>Rolloff Rate (dB/oct)</th>
<th>Peak Sidelobe Level (dB)</th>
<th>Peak ( 20 \log_{10} \delta )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Rectangular: ( \text{rect} \left( \frac{t}{T} \right) )</td>
<td>( \frac{4\pi}{T} )</td>
<td>-6</td>
<td>-13.3</td>
<td>-21 dB</td>
</tr>
<tr>
<td>2</td>
<td>Bartlett: ( \Delta \left( \frac{t}{2T} \right) )</td>
<td>( \frac{8\pi}{T} )</td>
<td>-12</td>
<td>-26.5</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Hanning: ( 0.5 \left[ 1 + \cos \left( \frac{2\pi t}{T} \right) \right] )</td>
<td>( \frac{8\pi}{T} )</td>
<td>-18</td>
<td>-31.5</td>
<td>-44 dB</td>
</tr>
<tr>
<td>4</td>
<td>Hanning: ( 0.54 + 0.46 \cos \left( \frac{2\pi t}{T} \right) )</td>
<td>( \frac{8\pi}{T} )</td>
<td>-6</td>
<td>-42.7</td>
<td>-53 dB</td>
</tr>
<tr>
<td>5</td>
<td>Blackman: ( 0.42 + 0.5 \cos \left( \frac{2\pi t}{T} \right) + 0.08 \cos \left( \frac{4\pi t}{T} \right) )</td>
<td>( \frac{12\pi}{T} )</td>
<td>-18</td>
<td>-58.1</td>
<td>-74 dB</td>
</tr>
<tr>
<td>6</td>
<td>Kaiser: ( \frac{l_0 \left( \alpha \sqrt{1 - \left( \frac{t}{T} \right)^2} \right)}{l_0(\alpha)} ), ( 0 \leq \alpha \leq 10 )</td>
<td>( \frac{11.2\pi}{T} )</td>
<td>-6</td>
<td>-59.9 (( \alpha = 8.160 ))</td>
<td></td>
</tr>
</tbody>
</table>

FIR: Rectangular & Hanning Windows

- **Rectangular**

- **Hanning**

  - Hanning: Less ripples, but wider transition band

Punskaya, Slide 93

Lathi, Table 7.3
Windowed FIR Property 1:
Equal transition bandwidth

- Equal transition bandwidth on both sides of the ideal cutoff frequency

Windowed FIR Property 2:
Peak Errors same in Passband & Stopband

- Peak approximation error in the passband ($1+\delta \rightarrow 1-\delta$) is equal to that in the stopband ($\delta \rightarrow -\delta$)
Windowed FIR Property 3: Mainlobe Width

- The distance between approximation error peaks is approximately equal to the width of the mainlobe $\Delta w_m$.

Windowed FIR Property 4: Mainlobe Width [2]

- The width of the mainlobe is wider than the transition bandwidth.

Punskaya, Slide 99

Punskaya, Slide 96
Windowed FIR Property 5:
Peak $\Delta \delta$ is determined by the window shape

- peak approximation error is determined by the window shape, independent of the filter order

FIR Filter Design
• How to get all these coefficients?

$$H(\omega) = h_0 + h_1 e^{-i\omega} + \cdots + h_{N-1} e^{-i(n-1)\omega}$$

FIR Design Methods:
1. Impulse Response Truncation
   + Simplest
   - Undesirable frequency domain-characteristics, not very useful
2. Windowing Design Method
   + Simple
   - Not optimal (not minimum order for a given performance level)
3. Optimal filter design methods
   + “More optimal”
   - Less simple…
FIR Filter Design & Operation
Ex: Lowpass FIR filter

- Set Impulse response (order n = 21)
- “Determine” h(t)
  - h(t) is a 20 element vector that we’ll use to as a weighted sum

- FFT (“Magic”) gives Frequency Response & Phase

Why is this “hard”? Looking at the Low-Pass Example

\[
H_d(\omega) = \begin{cases} 
1 & \text{if } |\omega| \leq \omega_c \\
0 & \text{if } \omega_c < |\omega| < \pi 
\end{cases}
\]

- Why is this hard?
  - Shouldn’t it be “easy” ??
    … just hit it with some FFT “magic” and then keep the bands we want and then hit it with some Inverse-FFT “supermagic”???

  - Remember we need a “system” that does this “rectangle function” in frequency

  - Let’s consider what that means…
    - It basically suggests we need an Inverse FFT of a “rectangle function”
Flashback: Fourier Series & Rectangular Functions

\[ \tilde{f} \{ \text{rect} \left( \frac{\omega}{2} \right) \} = \frac{sinc(t)}{\pi} \]

Ref: http://cnx.org/content/m32899/1.8/
http://www.wolframalpha.com/input/?i=IFFT%28sinc%28f%29%29

\[ \tilde{f} \{ \text{rect} (t) \} = sinc \left( \frac{\omega}{2} \right) \]

Ref: http://cnx.org/content/m26719/1.1/
http://www.wolframalpha.com/input/?i=FFT%28rect%28t%29%29

See:
- Table 7.1 (p. 702) Entry 17
- Table 9.1 (p. 852) Entry 7

- The sinc function might look familiar
  - This is the frequency content of a square wave (box)

- This also applies to signal reconstruction!

  ➔ Whittaker–Shannon interpolation formula
  - This says that the “better way” to go from Discrete to Continuous (i.e. D to A) is not ZOH, but rather via the sinc!

\[ x(t) = \sum_{n=-\infty}^{\infty} x[n] \cdot sinc \left( \frac{t-nT}{T} \right) \]
∴ FIR and Low Pass Filters...

\[ H_d(\omega) = \begin{cases} 
1 & \text{if } |\omega| \leq \omega_c \\
0 & \text{if } \omega_c < |\omega| < \pi 
\end{cases} \]

Has impulse response:

\[ h_d(n) = \frac{\omega_c}{\pi} \frac{\sin(\omega_c n)}{\omega_c n} \]

Thus, to filter an impulse train with an ideal low-pass filter use:

\[ x(t) = \left( \sum_{n=-\infty}^{\infty} x[n] \cdot \delta(t - nT) \right) * \text{sinc} \left( \frac{t}{T} \right) \]

- **However**!!
  - a sinc is non-causal and infinite in duration
  And, this **cannot** be implemented in practice ⊗
  - we need to know all samples of the input, both in the past and in the future

Plan 0: Impulse Response Truncation

Maybe we saw this coming…

- Clip off the sinc at some large \( n \)

\[ \tilde{h}(n) = \frac{\sin(n\omega_c)}{\pi n} \text{ for } |n| \leq M \text{ and } 0 \text{ otherwise} \]

- **Ripples** in both passband/stopband
  and the transition not abrupt (i.e., a transition band).
- As \( M \to \infty \), transition band \( \to 0 \) (as expected!)
FIR Filters: Window Function Design Method

- Windowing: a generalization of the truncation idea

- There many, many “window” functions:
  - Rectangular
  - Triangular
  - Hanning
  - Hamming
  - Blackman
  - Kaiser
  - Lanczos
  - Many More … (see: http://en.wikipedia.org/wiki/Window_function)

Some Window Functions [1]

1. Rectangular

\[ w(n) = 1 \]
Windowing and its effects/terminology

Lathi, Fig. 7.45

Some More Window Functions …

2. Triangular window

\[ w(n) = 1 - \frac{n - \frac{N-1}{2}}{\frac{N+1}{2}} \]

- And Bartlett Windows
  - A slightly narrower variant with zero weight at both ends:

\[ w(n) = 1 - \left| \frac{n - \frac{N-1}{2}}{\frac{N-1}{2}} \right| \]
3. Generalized Hamming Windows

\[ w(n) = \alpha - \beta \cos\left(\frac{2\pi n}{N-1}\right) \]

→ Hanning Window

\[ \rightarrow w(n) = 0.5 \left(1 - \cos\left(\frac{2\pi n}{N-1}\right)\right) \]

→ Hamming’s Window

\[ \rightarrow \alpha = 0.54, \beta = 1 - \alpha = 0.46 \]

4. Blackman–Harris Windows

- A generalization of the Hamming family,
- Adds more shifted sinc functions for less side-lobe levels

\[ w(n) = a_0 - a_1 \cos\left(\frac{2\pi n}{N-1}\right) + a_2 \cos\left(\frac{4\pi n}{N-1}\right) - a_3 \cos\left(\frac{6\pi n}{N-1}\right) \]
Some More Window Functions…

5. Kaiser window
   - A DPSS (discrete prolate spheroidal sequence)
   - Maximize the energy concentration in the main lobe

\[ w(n) = \frac{I_0(\pi \alpha \sqrt{1 - \left(\frac{2n}{N-1} - 1\right)^2})}{I_0(\pi \alpha)} \]

   Where: \( I_0 \) is the zero-th order modified Bessel function of the first kind, and usually \( \alpha = 3 \).

Comparison of Alternative Windows – Time Domain
Comparison of Alternative Windows

Frequency Domain

Summary Characteristics of Common Window Functions

<table>
<thead>
<tr>
<th>No.</th>
<th>Window $w(t)$</th>
<th>Mainlobe Width</th>
<th>Rollloff Rate (dB/dec)</th>
<th>Peak Sidelobe Level (dB)</th>
<th>$20\log_{10}\delta$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Rectangular: $\text{rect}\left(\frac{t}{T}\right)$</td>
<td>$\frac{2\pi}{T}$</td>
<td>-6</td>
<td>-13.3</td>
<td>-21dB</td>
</tr>
<tr>
<td>2</td>
<td>Bartlet: $\Delta\left(\frac{t}{2T}\right)$</td>
<td>$\frac{8\pi}{T}$</td>
<td>-12</td>
<td>-26.5</td>
<td>-44dB</td>
</tr>
<tr>
<td>3</td>
<td>Hanning: $0.5\left[1 + \cos\left(\frac{2\pi t}{T}\right)\right]$</td>
<td>$\frac{8\pi}{T}$</td>
<td>-18</td>
<td>-31.5</td>
<td>-53dB</td>
</tr>
<tr>
<td>4</td>
<td>Hamming: $0.54 + 0.46 \cos\left(\frac{2\pi t}{T}\right)$</td>
<td>$\frac{8\pi}{T}$</td>
<td>-6</td>
<td>-42.7</td>
<td>-74dB</td>
</tr>
<tr>
<td>5</td>
<td>Blackman: $0.42 + 0.5 \cos\left(\frac{2\pi t}{T}\right) + 0.08 \cos\left(\frac{4\pi t}{T}\right)$</td>
<td>$\frac{12\pi}{T}$</td>
<td>-18</td>
<td>-58.1</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Kaiser: $k_0 \left[\sqrt{1 - \left(\frac{t}{T}\right)^2}\right]$</td>
<td>$\frac{11.2\pi}{T}$</td>
<td>-6</td>
<td>-59.9 ($\alpha = 8.188$)</td>
<td></td>
</tr>
</tbody>
</table>

Lathi, Table 7.3
Punskaya, Slide 92
FIR: Rectangular & Hanning Windows

- Rectangular

- Hanning

⇒ Hanning: Less ripples, but wider transition band

Adding Order

+ Transition and Smoothness
  – Increased Size
Windowed FIR Property 1:
Equal transition bandwidth

- Equal transition bandwidth on both sides of the ideal cutoff frequency

Windowed FIR Property 2:
Peak Errors same in Passband & Stopband

- Peak approximation error in the passband (1+δ → 1-δ) is equal to that in the stopband (δ → -δ)
Windowed FIR Property 3: Mainlobe Width

- The distance between approximation error peaks is approximately equal to the width of the mainlobe $\Delta w_m$

Windowed FIR Property 4: Mainlobe Width [2]

- The width of the mainlobe is wider than the transition bandwidth
Windowed FIR Property 5:
Peak $\Delta \delta$ is determined by the window shape

- peak approximation error is determined by the window shape, independent of the filter order

Window Design Method Design Terminology

Where:
- $\omega_c$: cutoff frequency
- $\delta$: maximum passband ripple
- $\Delta \omega$: transition bandwidth
- $\Delta \omega_m$: width of the window mainlobe
Passband / stopband ripples

\( \omega_s \) and \( \omega_p \): Corner Frequencies

Passband / stopband ripples are often expressed in dB:

- passband ripple = \( 20 \log_{10} (1+\delta_p) \) dB
- peak-to-peak passband ripple \( \cong 20 \log_{10} (1+2\delta_p) \) dB
- minimum stopband attenuation = \(-20 \log_{10} (\delta_s) \) dB
Summary of Design Procedure

1. Select a suitable window function

2. Specify an ideal response $H_d(\omega)$

3. Compute the coefficients of the ideal filter $h_d(n)$

4. Multiply the ideal coefficients by the window function to give the filter coefficients

5. Evaluate the frequency response of the resulting filter and iterate if necessary (e.g. by increasing $M$ if the specified constraints have not been satisfied).

Windowed Filter Design Example

- Design a type I low-pass filter with:
  - $\omega_p = 0.2\pi$
  - $\omega_s = 0.3\pi$
  - $\delta = 0.01$
Windowed Filter Design Example:
Step 1: Select a suitable Window Function

<table>
<thead>
<tr>
<th>No.</th>
<th>Window set(1)</th>
<th>Matchbox Width</th>
<th>Rolloff Rate (dB)</th>
<th>Peak Side-lobe Level (dB)</th>
<th>Peak 25dBFreq</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Rectangular</td>
<td>$\frac{\pi}{2}$</td>
<td>$-6$</td>
<td>$-13.3$</td>
<td>$-21$ dB</td>
</tr>
<tr>
<td>2</td>
<td>Bartlett</td>
<td>$\frac{\pi}{2}$</td>
<td>$-12$</td>
<td>$-16.5$</td>
<td>$-24$ dB</td>
</tr>
<tr>
<td>3</td>
<td>Hanning $1/2 [1 + \cos (\pi x)]$</td>
<td>$\frac{\pi}{2}$</td>
<td>$-18$</td>
<td>$-16.5$</td>
<td>$-44$ dB</td>
</tr>
<tr>
<td>4</td>
<td>Hanning $\frac{3}{4} + \frac{1}{4} \cos \frac{\pi x}{2}$</td>
<td>$\frac{\pi}{2}$</td>
<td>$-4$</td>
<td>$-12$</td>
<td>$-53$ dB</td>
</tr>
<tr>
<td>5</td>
<td>Blackman $0.42 + 0.56 \cos \frac{4 \pi x}{5} + 0.08 \cos \frac{8 \pi x}{5}$</td>
<td>$\frac{\pi}{2}$</td>
<td>$-16$</td>
<td>$-16.1$</td>
<td>$-74$ dB</td>
</tr>
</tbody>
</table>

- LP with: $\omega_p = 0.2 \pi$, $\omega_s = 0.3 \pi$, $\delta = 0.01$
- $\delta = 0.01$: The required peak error spec: $-20 \log_{10}(\delta) = -40$ dB
- Main-lobe width:
  $\omega_s - \omega_p = 0.3\pi - 0.2\pi = 0.1\pi \rightarrow 0.1\pi = 8\pi / M$
  $\rightarrow$ Filter length $M \geq 80$ & Filter order $N \geq 79$
- BUT, Type-I filters have even order so $N = 80$

Windowed Filter Design Example:
Step 2: Specify the Ideal Response

- From Property 1 (Midpoint rule)

  $\omega_c = (\omega_s + \omega_p)/2 = (0.2\pi + 0.3\pi)/2 = 0.25\pi$

  $\therefore$ An ideal response will be:

  $$H_d(\omega) = \begin{cases} 
  1 & \text{if } |\omega| \leq 0.25\pi \\
  0 & \text{if } 0.25\pi < |\omega| < \pi
  \end{cases}$$
Windowed Filter Design Example:
Step 3: Compute the coefficients of the ideal filter

- The ideal filter coefficients $h_d$ are given by the Inverse Discrete time Fourier transform of $H_d(\omega)$

$$x(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(\omega) e^{j\omega n} d\omega = \frac{1}{2\pi} \int_{-\omega_c}^{\omega_c} e^{j\omega n} d\omega$$

$$= \frac{\omega_c \sin \omega_c n}{\pi} \frac{\omega_c}{\omega_c n}.$$

+ Delayed impulse response (to make it causal)

$$\hat{h}(n) = \hat{h} \left( n - \frac{N-1}{2} \right)$$

- Coefficients of the ideal filter (via equation or IFFT):

$$h(n) = \frac{\sin (0.5\pi(n-40))}{\pi(n-40)}$$

Windowed Filter Design Example:
Step 4: Multiply to obtain the filter coefficients

- Multiply by a Hamming window function for the passband:

$$w(n) = 0.54 - 0.46 \cos \left( \frac{2\pi n}{M} \right)$$
Windowed Filter Design Example:
Step 5: Evaluate the Frequency Response and Iterate

- The frequency response is computed as the DFT of the filter coefficient vector

- **If** the resulting filter does not meet the specifications, **then:**
  - Adjust the ideal filter frequency response (for example, move the band edge) and repeat (step 2)
  - Adjust the filter length and repeat (step 4)
  - change the window (& filter length) (step 4)

- **And/Or** consult with Matlab:
  - `FIR1` and `FIR2`
  - `B=FIR2(N,F,M)` : Designs a Nth order FIR digital filter with

Windowed Filter Design Example:
Consulting Matlab:

- **FIR1** and **FIR2**
  - `B=FIR2(N,F,M)` : Designs a Nth order FIR digital filter

  - `F` and `M` specify frequency and magnitude breakpoints for the filter such that `plot(N,F,M)` shows a plot of desired frequency

  - Frequencies `F` must be in increasing order between 0 and Fs/2, with Fs corresponding to the sample rate.

  - `B` is the vector of length N+1, it is real, has linear phase and symmetric coefficients

  - Default window is Hamming – others can be specified
In Conclusion

• FIR Filters are digital (can not be implemented in analog) and exploit the difference and delay operators

• A window based design builds on the notion of a truncation of the “ideal” box-car or rectangular low-pass filter in the Frequency domain (which is a sinc function in the time domain)

• Other Design Methods exist:
  – Least-Square Design
  – Equiripple Design
  – Remez method
  – The Parks-McClellan Remez algorithm
  – Optimisation routines …