EE 123 Discussion Section 5

Jon Tamir

March 7, 2016

Some notes from Giulia Fanti, Frank Ong, Michael Lustig, Al Bovik, Kannan Ramchandran
Announcements

- HW 5 self grade – due Tuesday March 8
- Lab 3 – due Thursday March 10
- HW 6 – due Friday March 11

- Questions?
Plan

- Multi-rate DSP
Downsampling

- **Compresses** in the time domain
- **Expands** in the frequency domain

\[ y[n] = x[nN] \]

\[ Y(e^{j\omega}) = \frac{1}{N} \sum_{k=0}^{N-1} X(e^{j(\omega - 2\pi k)/N}) \]
Alternative derivation of downsampling DTFT

- **Recall:** A $N$-periodic sequence has a discrete Fourier series (DFS):

\[
\tilde{s}(m) = \frac{1}{N} \sum_{k=0}^{N-1} \tilde{S}_k e^{j 2\pi m k / N}
\]

DFS Representation

\[
\tilde{S}_k = \sum_{m=0}^{N-1} \tilde{s}(m) e^{-j 2\pi k m / N}
\]

DFS Coefficients
Alternative derivation of downsampling DTFT

• Impulse train: 
  \[ \tilde{s}(m) = \sum_{k \in \mathbb{Z}} \delta(m - kN) \]

• DFS Coefficients (check): 
  \[ \tilde{S}_k = 1 \]

• DFS Representation: 
  \[ \tilde{s}(m) = \frac{1}{N} \sum_{k=0}^{N-1} e^{j2\pi mk/N} \]
Alternative derivation of downsampling DTFT

\[ x[n] \downarrow N \rightarrow y[n] = x[nN] \]

\[ Y(e^{j\omega}) = \sum_{m=-\infty}^{\infty} y[m]e^{-j\omega n} = \sum_{m=-\infty}^{\infty} x[m]s[m]e^{-j\omega m/N}, \]

\[ s[m] = \begin{cases} 
1 & \text{if } m = kN \\
0 & \text{o.w.} 
\end{cases} = \sum_{k=-\infty}^{\infty} \delta(m - kN) = \frac{1}{N} \sum_{k=0}^{N-1} e^{j2\pi km/N} \]
Alternative derivation of downsampling DTFT

\[ x[n] \quad \Downarrow^N \quad y[n] = x[nN] \]

\[
Y(e^{j\omega}) = \frac{1}{N} \sum_{k=0}^{N-1} \sum_{m=-\infty}^{\infty} x[m] e^{-j(\omega - 2\pi k)m/N}
\]

\[
= \frac{1}{N} \sum_{k=0}^{N-1} X(e^{j(\omega - 2\pi k)/N})
\]
Downsampling

1. **Stretch** $X(e^{j\omega})$ to $X(e^{j\omega/N})$
2. Create $(N-1)$ copies of the stretched versions
3. Shift each copy by successive multiples of $2\pi$ and add
4. Divide by $N$

\[
Y(e^{j\omega}) = \frac{1}{N} \sum_{k=0}^{N-1} X(e^{j(\omega - 2\pi k)/N})
\]
Decimation (filtering and downsampling)

\[ Y(\omega) = \frac{1}{D} \sum_{p=0}^{D-1} H\left( e^{j\frac{\omega - 2\pi p}{D}} \right) X\left( e^{j\frac{\omega - 2\pi p}{D}} \right) \]
Decimation Demo
Subtleties in Time-Freq Tiling

- Assume $h_0, h_1$ are ideal low, high pass filters

![Diagram showing signal processing flows and frequency responses](image)
Upsampling

- Expands in the time domain
- Compresses in the frequency domain

\[ Y(e^{j\omega}) = X(e^{j\omega N}) \]
Interpolation (Upsampling and filtering)

\[ Y(e^{j\omega}) = G(e^{j\omega})X(e^{j\omega U}) \]
Interpolation

1. Smooth discrete-time signal
   • Low frequency content

2. Upsample by 3
   • No longer smooth! Contains high frequencies

3. LPF
   • Removes high frequencies
Interpolation Demo
Sampling Rate Conversion

• Link two systems with different sampling rates

\[ N = \frac{N_1}{N_2} = \frac{\text{System 1 Sampling Rate}}{\text{System 2 Sampling Rate}} \]
Sampling Rate Conversion

\[ N = \frac{N_1}{N_2} \]
Sampling Rate Conversion

- Chirp
Sampling Rate Conversion

- Re-sample by a factor $11 / 3$ (axis is rescaled)
2. A continuous time signal $x_c(t)$ with the spectrum $X_c(j\Omega)$ depicted below is sampled with period $T_1$, resulting in a discrete sequence $x_1[n]$ with the DTFT $X_1(e^{j\omega})$ below.

a) (15 points) Determine the largest sampling period $T_2$ that would avoid aliasing, and express it in terms of $T_1$. Sketch the DTFT of the sequence $x_2[n]$, sampled with period $T_2$.

b) (15 points) Draw the block diagram of a post-processing unit that downsamples $x_2[n]$ by a factor of $T_2/T_1$. Sketch the DTFT of the output and compare it with $X_1(e^{j\omega})$ above.
Solution 1

\[ X_f(j \omega) \]

\[ \Omega_s = \frac{2\pi n}{\nu_1} \]

\[ \Omega_s = \frac{3}{4} \cdot \frac{2\pi}{\nu_1} \]
To avoid aliasing, we need
\[ \frac{2\pi}{T_2} = 2\pi f = \frac{2\pi}{T_1} \Rightarrow T_2 = \frac{2}{3} T_1 \]

b) \[ \frac{T_2}{T_1} = \frac{2}{3} \]

\[ x_1(u) \xrightarrow{\uparrow 2} x_{1(2u)} \xrightarrow{\text{LPF } \omega_{c} = \frac{\pi}{3}} x_{c(u)} \xrightarrow{\frac{2}{3}} y(u) \]

\[ y(c^i\omega) \neq x_c(c^i\omega) \]
4.46. Consider the system in Figure P4.46-1 with \( H_0(z) \), \( H_1(z) \), and \( H_2(z) \) as the system functions of LTI systems. Assume that \( x[n] \) is an arbitrary stable complex signal without any symmetry properties.

(a) Let \( H_0(z) = 1 \), \( H_1(z) = z^{-1} \), and \( H_2(z) = z^{-2} \). Can you reconstruct \( x[n] \) from \( y_0[n] \), \( y_1[n] \), and \( y_2[n] \)? If so, how? If not, justify your answer.

(b) Assume that \( H_0(e^{j\omega}) \), \( H_1(e^{j\omega}) \), and \( H_2(e^{j\omega}) \) are as follows:

\[
H_0(e^{j\omega}) = \begin{cases} 
1, & |\omega| \leq \pi/3, \\
0, & \text{otherwise,}
\end{cases}
\]

\[
H_1(e^{j\omega}) = \begin{cases} 
1, & \pi/3 < |\omega| \leq 2\pi/3, \\
0, & \text{otherwise,}
\end{cases}
\]

\[
H_2(e^{j\omega}) = \begin{cases} 
1, & 2\pi/3 < |\omega| \leq \pi, \\
0, & \text{otherwise.}
\end{cases}
\]

Can you reconstruct \( x[n] \) from \( y_0[n] \), \( y_1[n] \), and \( y_2[n] \)? If so, how? If not, justify your answer.
Solution 2 a,b

(a) Notice that

\[ y_0[n] = x[3n] \]
\[ y_1[n] = x[3n + 1] \]
\[ y_2[n] = x[3n + 2] \]

and therefore,

\[ x[n] = \begin{cases} 
  y_0[n/3], & n = 3k \\
  y_1[(n-1)/3], & n = 3k + 1 \\
  y_2[(n-2)/3], & n = 3k + 2 
\end{cases} \]

(b) Yes. Since the bandwidth of the filters are \( 2\pi/3 \), there is no aliasing introduced by downsampling. Hence to reconstruct \( x[n] \), we need the system shown in the following figure:

This solution has an error. What is it?
Question 2c

Now consider the system in Figure P4.46-2. Let $H_3(e^{j\omega})$ and $H_4(e^{j\omega})$ be the frequency responses of the LTI systems in this figure. Again, assume that $x[n]$ is an arbitrary stable complex signal with no symmetry properties.

\[ H_3(e^{j\omega}) \]
\[ x[n] \rightarrow \downarrow 2 \rightarrow y_3[n] \]
\[ H_4(e^{j\omega}) \]
\[ x[n] \rightarrow \downarrow 2 \rightarrow y_4[n] \]

\textbf{Figure P4.46-2}

\textbf{(c)} Suppose that $H_3(e^{j\omega}) = 1$ and

\[ H_4(e^{j\omega}) = \begin{cases} 1, & 0 \leq \omega < \pi, \\ -1, & -\pi \leq \omega < 0. \end{cases} \]

Can you reconstruct $x[n]$ from $y_3[n]$ and $y_4[n]$? If so, how? If not, justify your answer.
$H_3(e^{j\omega}) = \frac{1}{2\pi} \begin{array}{c} 1 \\ \pi \end{array}$

$H_4(e^{j\omega}) = \frac{1}{2\pi} \begin{array}{c} 1 \\ \pi \end{array}$

$H_3 + H_4 = \frac{2}{2\pi} \begin{array}{c} 2 \\ \pi \end{array}$

$H_3 - H_4 = \frac{2}{2\pi} \begin{array}{c} 1 \\ \pi \end{array}$

**Sampling is LINEAR?**

which we know how to reconstruct.
Multirate Noble Identities

\[ x[n] \xrightarrow{H(z)} \uparrow L \xrightarrow{y[n]} \equiv x[n] \xrightarrow{\uparrow L} H(z^L) \xrightarrow{y[n]} \]

\[ x[n] \xrightarrow{\downarrow M} H(z) \xrightarrow{y[n]} \equiv x[n] \xrightarrow{H(z^M)} \downarrow M \xrightarrow{y[n]} \]
Multirate Noble Identities

\[ x[n] \downarrow M \rightarrow H(z) \rightarrow y[n] \equiv x[n] \rightarrow H(z^M) \downarrow M \rightarrow \tilde{y}[n] \]

- Proof for M=2:

"Pf": Ex.

\[ W(z) = \frac{1}{2} X(z^{1/2}) + \frac{1}{2} X(-z^{1/2}) \]
\[ Y(z) = \frac{1}{2} X(z^{1/2}) H(z) + \frac{1}{2} X(-z^{1/2}) H(z) \]

\[ T(z) = X(z) H(z^2) \]
\[ Y'(z) = \frac{1}{2} \left[ T(z^{1/2}) + T(-z^{1/2}) \right] \]
\[ = \frac{1}{2} X(z^{1/2}) H(z) + \frac{1}{2} X(-z^{1/2}) H(z) \]
\[ = Y(z) \]
Ex. Cascade of subsampling and filtering

\[ H_0(2) \xrightarrow{2\downarrow} H_1(2) \xrightarrow{2\downarrow} H_2(2) \xrightarrow{2\downarrow} \]

\[ = H_0(2) H_1(2^2) H_2(2^4) \xrightarrow{8\downarrow} \]

\[ H_0(2) \xrightarrow{2\downarrow} H_0(2) \xrightarrow{2\downarrow} H_1(2) \xrightarrow{2\downarrow} \]

\[ = H_0(2) H_0(2^2) \xrightarrow{4\downarrow} \]

\[ H_1(2) \xrightarrow{2\downarrow} \]

\[ = H_0(2) H_1(2^2) \xrightarrow{4\downarrow} \]

\[ H_1(2) \xrightarrow{2\downarrow} \]

\[ = H_0(2) H_1(2^2) \xrightarrow{4\downarrow} \]
Ex.: Cascade of upsampling and filters
Polyphase Implementation of Decimation

\[ x[n] \xrightarrow{H(z)} y[n] \xrightarrow{\downarrow M} w[n] = y[nM] \]

\[ H(z) = \sum_{k=0}^{M-1} E_k(z^M)z^{-k} \]
Polyphase Implementation of Decimation

\[ x[n] \xrightarrow{H(z)} y[n] \xrightarrow{\downarrow M} w[n] = y[nM] \]

Interchange filter with decimation

\[ x[n] \xrightarrow{z^{-1}} E_0(z^M) \xrightarrow{\downarrow M} \xrightarrow{z^{-1}} E_1(z^M) \xrightarrow{\downarrow M} \ldots \xrightarrow{z^{-1}} E_{M-1}(z^M) \xrightarrow{\downarrow M} w[n] \]

now, what can we do?
Polyphase Implementation of Decimation

\[ x[n] \rightarrow H(z) \rightarrow y[n] \rightarrow \downarrow M \rightarrow w[n] = y[nM] \]

Interchange filter with decimation

Computation:
Each Filter: \( N/M \times (1/M) \) mult/unit time
Total: \( N/M \) mult/unit time

what about interpolation?
\[ e_{00} = h_0[2n] \]
\[ e_{01} = h_0[2n + 1] \]
\[ e_{10} = h_1[2n] = e^{j2\pi n} h_0[2n] = e_{00}[n] \]
\[ e_{11} = h_1[2n + 1] = e^{j2\pi n} e^{j\pi} h_0[2n + 1] = -e_{01}[n] \]
Polyphase Filter-Bank

\[ x[n] \rightarrow z^{-1} \rightarrow \downarrow 2 \rightarrow e_{00}[n] \rightarrow + \]
\[ \downarrow 2 \rightarrow e_{01}[n] \rightarrow + \]
\[ \downarrow 2 \rightarrow e_{10}[n] \rightarrow + \]
\[ \downarrow 2 \rightarrow e_{11}[n] \]

\[ e_{00} = h_0[2n] \]
\[ e_{01} = h_0[2n + 1] \]
\[ e_{10} = e_{00}[n] \]
\[ e_{11} = -e_{01}[n] \]

M. Lustig, EECS UC Berkeley
Polyphase Filter-Bank

\[ x[n] \xrightarrow{\downarrow 2 \quad z^{-1}} e_{00}[n] \]
\[ \xrightarrow{\downarrow 2 \quad z^{-1}} e_{01}[n] \]
\[ \xrightarrow{\downarrow 2 \quad z^{-1}} e \]

\[ e_{00} = h_0[2n] \]
\[ e_{01} = h_0[2n + 1] \]
\[ e_{10} = e_{00}[n] \]
\[ e_{11} = -e_{01}[n] \]