Downsampling

Section 6, Nick Antipa, 3/9/2018
Based on slides by Jon Tamir

Some notes from Giulia Fanti, Frank Ong, Michael Lustig,
Midterm 1

• $X[k] = \sum_{n=0}^{3N-1} x[n]e^{-j\pi nk/N}$
  
  $= \sum_{n=0}^{N-1} x[n]e^{-j\pi nk/N} + \sum_{n=N}^{2N-1} x[n]e^{-j\pi nk/N} + \sum_{n=2N}^{3N-1} x[n]e^{-j\pi nk/N}$

• What is the problem with this?
My favorite example of aliasing
And

- http://i.imgur.com/8X8Fcoy.gifv
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):

$$\tilde{s}(m) = \frac{1}{N} \sum_{k=0}^{N-1} \tilde{S}_k e^{j2\pi mk/N}$$

**DFS Representation**

$$\tilde{S}_k = \sum_{m=0}^{N-1} \tilde{s}(m) e^{-j2\pi km/N}$$

**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

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

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

\[ = \frac{1}{N} \sum_{k=0}^{N-1} X(e^{j(\omega-2\pi k)/N}) \]
1. **Stretch** $X(e^{j\omega})$ to $X(e^{\frac{j\omega}{N}})$
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)

\[
Y(e^{j\omega}) = \frac{1}{D} \sum_{p=0}^{D-1} H\left(e^{j\left(\frac{\omega - 2\pi p}{D}\right)}\right)X\left(e^{j\left(\frac{\omega - 2\pi p}{D}\right)}\right)
\]
Q: for a chirp of length \( L \) seconds, ranging from \( f_0 \) to \( f_m \), what is the expression for \( x_c(t) \), the continuous-time signal?

A: 

\[ x_c(t) = \sin \left( 2\pi \left[ f_0 + \frac{f_m - f_0}{\tau} t \right] t \right) \]

Q: How fast do we need to sample this signal?

A: \( f_s > 2f_m \)

Q: What is \( |X_d(j\Omega)| \)? (ignore effects of windowing)

A: \( \text{rect} \left( \frac{\Omega}{2\Omega_0} \right) \)
Chirp

\[ x_c(t) = \sin \left( 2\pi \left[ f_0 + \frac{f_m - f_0}{\tau} t \right] t \right) \]

Q: What is \( x[n] \) if we sample exactly at Nyquist?

\[ x[n] = \sin \left( 2\pi \left[ f_0 + \frac{f_m - f_0}{\tau} Tn \right] Tn \right) \]
\[ = \sin \left( n\pi \left[ \frac{\Omega_0}{\Omega_m} \right] \left( 1 - \frac{n}{N} \right) + \frac{n}{N} \right) \]

Q: Plot the frequency vs \( n \)
Downsample by 2 (without AA filter)

Recall: \( x[n] = \sin \left( n\pi \left( \frac{\Omega_0}{\Omega_m} \left( 1 - \frac{n}{N} \right) + \frac{n}{N} \right) \right) \)

A: \( x[2n] = \sin \left( 2n\pi \left( \frac{\Omega_0}{\Omega_m} \left( 1 - \frac{2n}{N} \right) + \frac{2n}{N} \right) \right) \)

Q: What is the frequency vs \( n \) now?

\[
2\pi \left[ \frac{\Omega_0}{\Omega_m} \left( 1 - \frac{2n}{N} \right) + \frac{2n}{N} \right]
\]
Chirp demo
Imaging example
Decimated by 2
Decimated by 2 again (4 total)
Lowpass filter first
Resampling looks fine!

properly resampled image
Upsampling

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

\[ X(e^{j\omega}) \uparrow N \rightarrow Y(e^{j\omega}) = X(e^{j\omega N}) \]
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