Overview of Some Signal Spectra

Conceptual

\[ x(n) = x_c(nT) \]

\[ y_c(t) = y_{sh}(t) \]

Typical Implementation

\[ x_c(t) \rightarrow \text{sample and hold} \rightarrow A/D converter \rightarrow \text{DSP} \rightarrow \text{D/A converter with hold} \rightarrow \text{analog low-pass filter} \rightarrow y_c(t) \]

\[ x_{sh}(t) = x(n) = x_c(nT) \]

\[ y_{sh}(t) \rightarrow y_c(t) \]
Example Signal Spectra

- $X_s(f)$ has same spectra as $X_c(f)$ but repeats every $f_s$ (assuming no aliasing occurs).

- $X(\omega)$ has same spectra as $X_s(f)$ freq axis normalized.

- Spectra for $X_{sh}(f)$ equals $X_s(f)$ multiplied by $\frac{\sin x}{x}$ response — in effect, filtering out high frequency images.
• \( x_s(t) \) scaled by \( \tau \) such that the area under the pulse at \( nT \) equals the value of \( x_c(nT) \).

• In other words, at \( t = nT \), we have

\[
x_s(nT) = \frac{x_c(nT)}{\tau}
\]  

(1)

Thus as \( \tau \to 0 \), height of \( x_s(t) \) at time \( nT \) goes to \( \infty \) and so we plot \( \tau x_s(t) \) instead.

• Define \( \vartheta(t) \) to be the step function,

\[
\vartheta(t) \equiv \begin{cases} 
1 & (t \geq 0) \\
0 & (t < 0)
\end{cases}
\]  

(2)

• then single-pulse signal, \( x_{sn}(t) \), can be written as

\[
x_{sn}(t) = \frac{x_c(nT)}{\tau} \left[ \vartheta(t - nT) - \vartheta(t - nT - \tau) \right]
\]  

(3)

and the entire signal \( x_s(t) \) as

\[
x_s(t) = \sum_{n = -\infty}^{\infty} x_{sn}(t)
\]  

(4)
Laplace Transform & Discrete-Time

• Above signals are defined for all time — we can find Laplace transforms of these signals.

• The Laplace transform $X_{sn}(s)$ for $x_{sn}(t)$ is

$$X_{sn}(s) = \frac{1}{\tau} \left(1 - \frac{e^{-s\tau}}{s}\right)x_c(nT)e^{-snT}$$  \hspace{1cm} (5)

and $X(s)$ is simply a linear combination of $x_{sn}(t)$, which results in

$$X_s(s) = \frac{1}{\tau} \left(1 - \frac{e^{-s\tau}}{s}\right) \sum_{n = -\infty}^{\infty} x_c(nT)e^{-snT}$$  \hspace{1cm} (6)

• Using the expansion $e^x = 1 + x + \frac{x^2}{2!} + \ldots$, when $\tau \to 0$, the term before the summation in (6) goes to unity.

• Therefore, as $\tau \to 0$,

$$X_s(s) = \sum_{n = -\infty}^{\infty} x_c(nT)e^{-snT}$$  \hspace{1cm} (7)

• This Laplace transform only depends on sample points, $x_c(nT)$ which in turn depends on the relative sampling-rate, $T$. 
Spectra of Discrete-Time Signals

• $x_s(t)$ spectra can be found by replacing $s = j\omega$ in (7)
• However, a more intuitive approach is ...
• Define a periodic pulse train, $s(t)$ as

$$s(t) = \sum_{n = -\infty}^{\infty} \delta(t - nT)$$

(8)

where $\delta(t)$ is the unit impulse function.
• Then $x_s(t)$ can be written as

$$x_s(t) = x_c(t)s(t)$$

(9)

$$X_s(j\omega) = \frac{1}{2\pi} X_c(j\omega) \otimes S(j\omega)$$

(10)

where $\otimes$ denotes convolution.

Spectra of Discrete-Time Signals

• Since the Fourier transform of a periodic impulse train is another periodic impulse train we have

$$S(j\omega) = \frac{2\pi}{T} \sum_{k = -\infty}^{\infty} \delta(\omega - k\frac{2\pi}{T})$$

(11)

• Thus, the spectra $X_s(j\omega)$ is found to be

$$X_s(j\omega) = \frac{1}{T} \sum_{k = -\infty}^{\infty} X_c(j\omega - \frac{jk2\pi}{T})$$

(12)

$$X_s(f) = \frac{1}{T} \sum_{k = -\infty}^{\infty} X_c(j2\pi f - jk2\pi f_s)$$

(13)
Spectra for Discrete-Time Signals

- The spectra for the sampled signal $x_s(t)$ equals a sum of shifted spectra of $x_c(t)$.

- **No aliasing** will occur if $X_c(j\omega)$ is bandlimited to $f_s/2$.

- Note that $x_s(t)$ can not exist in practice as it would require an infinite amount of power (seen by integrating $X_s(f)$ over all frequencies).

Spectra Example
Z-Transform

• The z-transform is merely a shorthand notation for (7).

• Specifically, defining

\[ z \equiv e^{sT} \]  

(14)

• we can write

\[ X(z) \equiv \sum_{n=-\infty}^{\infty} x_c(nT)z^{-n} \]  

(15)

• where \( X(z) \) is called the z-transform of the samples \( x_c(nT) \).

Z-Transform

• 2 properties of the z-transform are:

— If \( x(n) \leftrightarrow X(z) \), then \( x(n-k) \leftrightarrow z^{-k}X(z) \)

— Convolution in the time domain is equivalent to multiplication in the frequency domain.

\( X(z) \) is not a function of the sampling-rate!

• A 1Hz signal sampled at 10Hz has the same transform as a similar 1kHz signal sampled at 10kHz

• \( X(z) \) is only related to the numbers, \( x_c(nT) \) while \( X_s(s) \) is the Laplace transform of the signal \( x_s(t) \) as \( \tau \rightarrow 0 \).

• Think of the series of numbers as having a sample-rate normalized to \( T = 1 \) (i.e. \( f_s = 1 \text{Hz} \)).
Z-Transform

• Such a normalization results in
\[ X_s(f) = X \left( \frac{2\pi f}{f_s} \right) \]  \hspace{1cm} (16)

or equivalently, a frequency scaling of
\[ \omega = \frac{2\pi f}{f_s} \]  \hspace{1cm} (17)

• Thus, discrete-time signals have \( \omega \) in units of radians/sample.

• Continuous-time signals have frequency units of cycles/second (hertz) or radians/second.

---

Example Sinusoidal Signals

- \( x(n) \)
- \( x(n) \)
- \( x(n) \)
- \( x(n) \)

0 rad/sample = 0 cycles/sample

\( \pi / 8 \) rad/sample = \( 1 / 16 \) cycles/sample

\( \pi / 4 \) rad/sample = \( 1 / 8 \) cycles/sample

\( \pi / 2 \) rad/sample = \( 1 / 4 \) cycles/sample
Example Sinusoidal Signals

- A continuous-time sinusoidal signal of $1\text{kHz}$ when sampled at $4\text{kHz}$ will change by $\pi/2$ radians between each sample.
- Such a discrete-time signal is defined to have a frequency of $\pi/2 \text{rad/sample}$.
- Note that discrete-time signals are not unique since the addition of $2\pi$ will result in the same signal.
- For example, a discrete-time signal having a frequency of $\pi/4 \text{rad/sample}$ is identical to that of $9\pi/4 \text{rad/sample}$.
- Normally discrete-time signals are defined to have frequency components only between $-\pi$ and $\pi \text{rad/sample}$.

Downsampling

- Keep every $L$‘th sample and throw away $L - 1$ samples.
- It expands the original spectra by $L$.
- For aliasing not to occur, original signal must be bandlimited to $\pi/L$. 
Upsampling

• Insert \( L - 1 \) zero values between samples
• The frequency axis is scaled by \( L \) such that \( 2\pi \) now occurs where \( L2\pi \) occurred in the original signal.
• No worry about aliasing here.

Discrete-Time Filters

• An input series of numbers is applied to a discrete-time filter to create an output series of numbers.
• This filtering of discrete-time signals is most easily visualized with the shorthand notation of \( z \)-transforms.

Transfer-Functions
• Similar to those for continuous-time filters except instead of polynomials in \( s \), polynomials in \( z \) are obtained.
Cont-Time Transfer-Function

- Low-pass continuous-time filter, $H_c(s)$,

$$H_c(s) = \frac{4}{s^2 + 2s + 4} \quad (18)$$

- The poles are the roots of the denominator polynomial
- Poles: $-1.0 \pm 1.7321j$ for this example.
- Zeros: Defined to have two zeros at $\infty$ since the denominator poly is two orders higher than the numerator poly.

Cont-Time Frequency Response

- Poles and zeros plotted in the $s$-plane.
- Substitution $s = j\omega$ is equivalent to finding the magnitude and phase of vectors from a point along the $j\omega$ axis to all the poles and zeros.
Discrete-Time Transfer-Function

\[ H(z) = \frac{0.05}{z^2 - 1.6z + 0.65} \]  \hspace{1cm} (19)

- Poles: \( 0.8 \pm 0.1j \) in the \( z \)-plane and two zeros are again at \( \infty \).
- To find the frequency response of \( H(z) \), the poles and zeros can be plotted in the \( z \)-plane, and the unit circle contour is used, \( z = e^{j\omega} \).

Discrete-Time Frequency Response

- Note that poles or zeros occurring at \( z = 0 \) do not affect the magnitude response of \( H(z) \) since a vector from the origin to the unit circle always has a length of unity. However, they would affect the phase response.
Discrete-Time Frequency Response

- $z = 1$ corresponds to the frequency response at both dc (i.e. $\omega = 0$) and for $\omega = 2\pi$.
- The time normalization of setting $T = 1$ implies that $\omega = 2\pi$ is equivalent to the sampling-rate speed (i.e. $f = f_s$) for $X_s(f)$.
- As with cont-time filter, if filter coefficients are real, poles and zeros occur in complex-conjugate pairs — magnitude is symmetric, phase is anti-symmetric.
- Going around the circle again would give the same result as the first time implying that the frequency response repeats every $2\pi$.

Stability of Discrete-Time Filters

- To realize rational polynomials in “$z$”, discrete-time filters use delay elements (i.e. “$z^{-1}$” building blocks) much the same way that analog filters can be formed using integrators (i.e. “$s^{-1}$” building blocks).
- The result is finite difference equations describing discrete-time filters.
Stability of Discrete-Time Filters

- A finite difference equation can be written for above system
  \[
  y(n + 1) = bx(n) + ay(n)
  \]  (20)

- In the \( z \)-domain, this equation is written as
  \[
  zY(z) = bX(z) + aY(z)
  \]  (21)

- We find \( H(z) \) given by
  \[
  H(z) \equiv \frac{Y(z)}{X(z)} = \frac{b}{z - a}
  \]  (22)

which has a pole on the real axis at \( z = a \).

---

Stability of Discrete-Time Filters

- To test for stability, let the input \( x(n) \) be an impulse
  \[
  y(0) = k
  \]

where \( k \) is some arbitrary initial state value for \( y \).

\[
\begin{align*}
y(1) &= b + ak \\
y(2) &= ab + a^2k \\
y(3) &= a^2b + a^3k \\
y(4) &= a^3b + a^4k \\
&\vdots
\end{align*}
\]
Stability of Discrete-Time Filters

- The response, $h(n)$, is seen to be given by

$$h(n) = \begin{cases} 
0 & (n < 1) \\
(a^n - b + a^n k) & (n \geq 1)
\end{cases}$$  \hspace{1cm} (23)

- This response remains bounded only when $|a| \leq 1$ for this first-order filter and is unbounded otherwise.

- In general, a linear time-invariant discrete-time filter, $H(z)$, is stable if and only if all its poles are located within the unit circle.

IIR Filters

- Infinite-Impulse-Response (IIR) filters are those discrete-time filters that when excited by an impulse, their outputs remain non-zero assuming infinite precision arithmetic.

- The above example is IIR when $a \neq 0$

- IIR filters can be more efficient when long impulse responses are needed.

- They have some unusual behaviors due to finite-precision effects such as limit-cycles.
FIR Filters

- Finite-Impulse-Response (FIR) filters are those discrete-time filters that when excited by an impulse, their outputs go precisely to zero (and remain zero) after a finite value of $n$.
- Example — running average of 3

$$y(n) = \frac{1}{3}(x(n) + x(n-1) + x(n-2))$$ \hfill (24)

$$H(z) = \frac{2}{3} \sum_{i=0}^{2} z^{-i}$$ \hfill (25)

- Has poles but they all occur at $z = 0$.
- FIR filters are always stable and exact linear phase filters can be realized.

Bilinear Transform

- Consider $H_c(p)$ as a continuous-time transfer-function (where “$p$” is the complex variable equal to $\sigma_p + j\Omega$), the bilinear transform is defined to be given by,

$$p = \frac{z - 1}{z + 1}$$ \hfill (26)

- The inverse transformation is given by,

$$z = \frac{1 + p}{1 - p}$$ \hfill (27)

- The $z$-plane locations of 1 and -1 (i.e. dc and $f_s/2$) are mapped to $p$-plane locations of 0 and $\infty$, respectively.
Bilinear Transform

• The unit circle, \( z = e^{j\omega} \), in the \( z \)-plane is mapped to the entire \( j\Omega \) axis in the \( p \)-plane.

\[
p = \frac{e^{j\omega} - 1}{e^{j\omega} + 1} = \frac{e^{j(\omega/2)}(e^{j(\omega/2)} - e^{-j(\omega/2)})}{e^{j(\omega/2)}(e^{j(\omega/2)} + e^{-j(\omega/2)})}
\]

\[
= \frac{2j\sin(\omega/2)}{2\cos(\omega/2)} = j\tan(\omega/2)
\]

(28) (29)

• Results in the following frequency “warping”.

\[ \Omega = \tan(\omega/2) \]

(30)

Bilinear Transform Filter Design

• Design a continuous-time transfer-function, \( H_c(p) \), and choose the discrete-time transfer-function, \( H(z) \), such that

\[ H(z) \equiv H_c((z - 1)/(z + 1)) \]

(31)

so that

\[ H(e^{j\omega}) = H_c(j\tan(\omega/2)) \]

(32)

• The response of \( H(z) \) is seen to be equal to the response of \( H_c(p) \) except with a frequency “warping”

• Order of the cont-time and discrete-time also same.
Bilinear Design Example

- Find a first-order $H(z)$ that has a 3db frequency at $f_s/20$, a zero at -1 and a dc gain of one.

- Using (30), the frequency value, $f_s/20$, or equivalently, $\omega = (2\pi)/20 = 0.314159$ is mapped to $\Omega = 0.1584$.

- Thus, $H_c(p)$ should have a 3dB frequency value of 0.1584 rad/s.

- Such a 3db frequency value is obtained by having a $p$-plane zero equal to $\infty$ and pole equal to -0.1584.

Bilinear Design Example

- Transforming these continuous-time pole and zero back using (27) results in a $z$-plane zero at -1 and a pole at 0.7265.

- Therefore, $H(z)$ appears as

$$H(z) = \frac{k(z + 1)}{z - 0.7265} \quad (33)$$

- The constant $k$ can be determined by setting the dc gain to one, or equivalently, $|H(1)| = 1$ which results in $k = 0.1368$. 
Sample-and-Hold Response

- A sample-and-held signal, $x_{sh}(t)$, is related to its sampled signal by the mathematical relationship,

$$
x_{sh}(t) = \sum_{n = -\infty}^{\infty} x_c(nT)[\delta(t - nT) - \delta(t - nT - T)]
$$

(34)

- $x_{sh}(t)$ is well-defined for all time and thus the Laplace transform can be found to be equal to

$$
X_{sh}(s) = \frac{1 - e^{-sT}}{s} \sum_{n = -\infty}^{\infty} x_c(nT)e^{-snT}
$$

$$
= \frac{1 - e^{-sT}}{s} X_s(s)
$$

(35)

Sample-and-Hold Response

- The hold transfer-function, $H_{sh}(s)$, is equal to

$$
H_{sh}(s) = \frac{1 - e^{-sT}}{s}
$$

(36)

- The spectra for $H_{sh}(s)$ is found by substituting $s = j\omega$

$$
H_{sh}(j\omega) = \frac{1 - e^{-j\omega T}}{j\omega} = T \times e^{-\frac{j\omega T}{2}} \times \frac{\sin\left(\frac{\omega T}{2}\right)}{\left(\frac{\omega T}{2}\right)}
$$

(37)
Sample-and-Hold Response

- The magnitude of this response is given by

\[ |H_{sh}(j\omega)| = T \frac{\sin\left(\frac{\omega T}{2}\right)}{\omega T/2} \]

or

\[ |H_{sh}(f)| = T \frac{\sin\left(\frac{\pi f}{f_s}\right)}{\pi f/f_s} \] (38)

- referred to as the \(\frac{\sin x}{x}\) or “sinc” response.

Sample-and-Hold Response

- This frequency shaping of a sample-and-hold only occurs for a continuous-time signal.
- Specifically, a sample-and-hold before an A/D converter does not aid in any anti-aliasing requirement since the A/D converter has a true discrete-time output.