Digital Signal Processing: Exam #2 Review

Soo Min Kwon

April 12, 2021

Preamble

This guide serves as a general review of some of the topics for the second exam for the course Digital Signal Processing (ECE 346). Note that there may be typos so use it at your own risk. If you spot a mistake, please let me know so that I can edit it.

1 CTFT & DTFT Conversion

Consider the following block diagram of a DSP system:

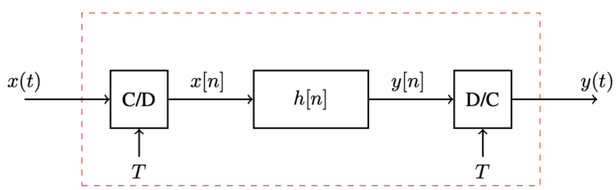

Figure 1: Block diagram of a DSP system

We want to define what happens in the frequency domain when going directly from the CTFT to the DTFT (and vice versa):

- When going from $x(t)$ to $x[n]$:

$$X(e^{j\omega}) = \frac{1}{T}\sum_{k=-\infty}^{\infty} X\left(j\left(\frac{\omega - 2\pi k}{T}\right)\right), \tag{1}$$

  where $X(e^{j\omega})$ is the DTFT of $x[n]$ and T is the sampling period. The important thing to note here is the scaling of $1/T$.

- When going from $y[n]$ to $y(t)$:

$$Y(j\Omega) = H_r(j\Omega)Y(e^{j\Omega T}), \tag{2}$$

where $Y(j\Omega)$ is the CTFT of $y(t)$.

- In the case where $H_r(j\Omega)$ above is an **ideal reconstruction filter**:

$$Y(j\Omega) = \begin{cases} TY(e^{j\Omega T}), & |\Omega| \leq \pi/T \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

One more important mistake that many students do is scale the delta function when doing this direct conversion. For example, let's say you wanted to convert the signal

$$X(j\Omega) = 2\pi\delta(\Omega - 100\pi), \tag{4}$$

into $X(e^{j\omega})$ for some arbitrary T. Note that we do NOT need to scale the amplitude due to the properties of the delta function:

$$\begin{aligned} X(e^{j\omega}) &= \frac{2\pi}{T}\delta(\Omega - 100\pi)|_{\Omega=\omega/T} && (5)\\ &= \frac{2\pi}{T}\delta\left(\frac{\omega}{T} - 100\pi\right) && (6)\\ &= \frac{2\pi}{T}\delta\left(\frac{1}{T}(\omega - 100\pi T)\right) && (7)\\ &= \frac{2\pi}{T} \cdot T\delta\left(\omega - 100\pi T\right) && (8)\\ &= 2\pi\delta(\omega - 100\pi T). && (9) \end{aligned}$$

2 Discrete-Time Processing of Continuous-Time Signals

Recall the following block diagram of a DSP system:

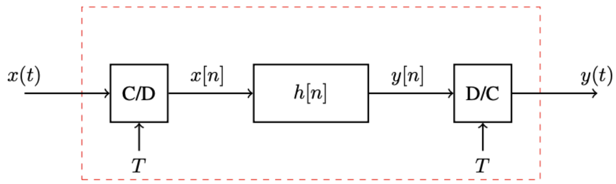

Figure 2: Block diagram of a DSP system

If there is no aliasing occuring in the sampling of $x(t)$, then the dashed block acts as an LTI system which can be expressed as $H_{eff}(j\Omega)$:

$$H_{eff}(j\Omega) = \begin{cases} H(e^{j\Omega T}), & |\Omega| \leq \pi/T \\ 0, & \text{otherwise,} \end{cases} \tag{10}$$

where $H(e^{j\Omega T})$ is the DTFT of $h[n]$ located inside the dashed block.

If we can stop aliasing with an anti-aliasing filter, then the dashed block can act as an LTI system for the subset of signals that were not killed in the anti-aliasing filter.

3 Review of Z-Transforms

The concept of z-transforms are simple as long as you know the analysis equation and how to deal with the region of convergence (ROC). The analysis equation is given by the following:

$$X(z) = \sum_{n=\infty}^{\infty} x[n]z^{-n}, \tag{11}$$

where $x[n]$ is an arbitrary signal. We generally use the geometric formula to simplify this into a closed-form expression. If you are stuck on understanding the region of convergence, refer to the notes on Canvas (under Recitation Notes).

One important concept that we saw with z-transforms (not only limited to z-transforms) is the expression of a constant coefficient difference equation (CCDE). That is, we can express the transfer function (z-transform of the impulse response) as a rational function of two polynomials in z^{-1}:

$$H(z) = \frac{Y(z)}{X(z)} = \frac{p_0 + p_1 z^{-1} + \ldots + p_M z^{-M}}{q_0 + q_1 z^{-1} + \ldots + q_M z^{-M}}. \tag{12}$$

Note that if we are given that the DTFT exists (i.e. all poles are inside the unit circle), then we can compute the DTFT of the impulse response given the transfer function as

$$H(e^{j\omega}) = H(z)|_{z=e^{j\omega}} \tag{13}$$

4 Introduction to Digital Filters

There are two broad categorizations of digital filters:

- Finite impulse response (FIR) filter
- Infinite impulse response (IIR) filter

In the real world, we cannot always implement ideal filters. Instead, we can approximate an ideal filter by an infinite impulse response (IIR) filter that can be described by the **difference equation**. If we can express the filter into a difference equation (and hence, a ratio of two polynomials), then the filter is an IIR filter. The z-transform of an FIR filter is generally a sum of finite terms. Since this implicitly implies that we have no denominator (compared to IIR filters), FIR filters only have trivial poles at $z = 0$ and $z = \infty$. If you are wondering how the poles and zeros affect a filter:

- The zeroes help us interpret the frequency response of the filter. The closer to zero is to the unit-circle, the more attenuated our frequencies are in the direction of the zero.
- The poles help us identify the region of convergence.

We can compute the zeroes and poles by setting the numerator and denominator equal to zero and solving for z, respectively.

5 Linear-Phase, Real-Valued FIR Filters

Let's first go over some notes on linear phase, real-valued FIR filters:

- Given frequency response $H(e^{j\omega}) = |H(e^{j\omega})|e^{j\angle H(e^{j\omega})}$, a filter has linear phase if its phase is defined as

$$\angle H(e^{j\omega}) = -\alpha\omega \pm \beta, \tag{14}$$

  where α and β are constants. This is because linear phase is often described as by a constant group delay, where the group delay is defined as

$$\tau(\omega) = -\frac{d\angle H(e^{j\omega})}{d\omega}. \tag{15}$$

- If a filter is a linear-phase, causal FIR filter, then if $z = re^{j\phi}$ is a zero, then

$$z = (re^{j\phi})^{-1} = \frac{1}{r}e^{-j\phi} \tag{16}$$

  must also be a zero.
- Note that a filter can have zeroes on the unit circle (e.g. $z = -1, 1$) or at zero and still be a filter with linear phase, as the inverse of a zero on the unit circle is itself.
- If a filter is a real-valued FIR filter, then if $z = re^{j\phi}$ is a zero, then its complex conjugate

$$z = re^{-j\phi}, \tag{17}$$

  must also be a zero.

- Putting these facts together, if a filter is a real-valued, linear phase FIR filter and a zero is NOT real and NOT on the unit circle, then the zero must come in multiples of 4.
- An even length real-valued, linear phase FIR filter must have a zero at $z = 1$ and/or at $z = -1$.
- We conclude these notes by talking about the **four types of real-valued, linear-phase FIR filters:**

  1. Type 1: Odd-length filter with a symmetric impulse response (used for either LPF or HPF)
  2. Type 2: Even-length filter with a symmetric impulse response (not good for HPF)
  3. Type 3: Odd-length filter with a anti-symmetric impulse response (not good for either LPF or HPF)
  4. Type 4: Even-length filter with a anti-symmetric impulse response (not good for LPF)

I think there are enough problems on Canvas to practice if you haven't done so already on these types of filters.

6 Window-based FIR Filter Design

I will conclude these notes with window-based filtering. Since the notion of *desired* impulse responses are important, here are some general filters shown mathematically:

1. Low-pass filter with cutoff ω_c:

$$H_d(e^{j\omega}) = \begin{cases} 1, & |\omega| \leq \omega_c \\ 0, & \omega_c < |\omega| \leq \pi \end{cases} \tag{18}$$

   with impulse response

$$h_d[n] = \frac{\sin(\omega_c n)}{\pi n}. \tag{19}$$

2. High-pass filter with cutoff ω_c:

$$H_d(e^{j\omega}) = 1 - H_{LP}^{\omega_c}(e^{j\omega}) \tag{20}$$

$$= \begin{cases} 0, & |\omega| < \omega_c \\ 1, & \omega_c \leq |\omega| \leq \pi \end{cases} \tag{21}$$

with impulse response

$$h_d[n] = \delta[n] - \frac{\sin(\omega_c n)}{\pi n}. \tag{22}$$

3. Band-pass filter with cutoff ω_{c1} and ω_{c2}:

$$H_d(e^{j\omega}) = H_{LP}^{\omega_{c2}}(e^{j\omega}) - H_{LP}^{\omega_{c1}}(e^{j\omega}) \tag{23}$$

$$= \begin{cases} 0, & |\omega| < \omega_{c1} \\ 1, & \omega_{c1} \le |\omega| \le \omega_{c2} \\ 0, & \omega_{c2} < |\omega| < \pi. \end{cases} \tag{24}$$

with impulse response

$$h_d[n] = \frac{\sin(\omega_{c2} n)}{\pi n} - \frac{\sin(\omega_{c1} n)}{\pi n}. \tag{25}$$

4. Multi-band filter (total of c bands):

$$H_d(e^{j\omega}) = A_k, \quad \omega_{k-1} \le |\omega| \le \omega_k, \quad k = 1, \ldots c, \tag{26}$$

where $\omega_0 = 0$ and $\omega_c = \pi$. The impulse response is a sum of sinc functions:

$$h_d[n] = \sum_{i=1}^{c} (A_i - A_{i+1}) \frac{\sin(\omega_i n)}{\pi n}. \tag{27}$$

The general formula for designing window-based filters is the following:

1. State the desired (or ideal) impulse response of the filter, $h_d[n]$. This would be something like one of the filters described above.

2. Shift your filter to make the filter causal:

$$\tilde{h}[n] = h_d[n - M]. \tag{28}$$

3. Window the filter using some windowing method to get the designed filter $h[n]$:

$$h[n] = \tilde{h}[n] \cdot w[n], \tag{29}$$

where $w[n]$ is the windowing function.

Lastly, there are questions that ask to find the minimum filter length. You can easily compute this by finding the transition width and using the transition width equation. For example, for a rectangular window, this equation is

$$\Delta\omega = \frac{0.92\pi}{M}. \tag{30}$$

Upon computing M, we can compute the final length of the filter,

$$N = 2M + 1. \tag{31}$$

Note that sometimes there are more than one transition width and/or ripple value. You want to take the minimum of them, as you want the stricter condition to hold for this filter.