Course: ECE 5210

Subject: DSP Course Cliff Notes

Date: February 19, 2025

Contents

1 Preliminary Mathematics

1.1 Summations

We can compute the following summations of the form

$$\sum_{n=0}^{N} r^n = \frac{1-r^{N+1}}{1-r} \quad \text{for } r \neq 1$$
$$\sum_{n=N_1}^{N_2} r^n = \frac{r^{N_1}-r^{N_2+1}}{1-r} \quad \text{for } r \neq 1$$
$$\sum_{n=0}^{\infty} r^n = \frac{1}{1-r} \quad \text{for } |r| < 1$$
$$\sum_{n=1}^{\infty} r^n = \frac{r}{1-r} \quad \text{for } |r| < 1.$$

1.2 Complex Numbers

A complex number $z \in \mathbb{C}$ has a real part a and an imaginary part b, and is written as $z = a + jb$. The magnitude of a complex number is $|z| = \sqrt{a^2 + b^2}$, and the angle of a complex number is $\angle z = \tan^{-1}\left(\frac{b}{a}\right)$. Euler's formula states that the complex exponential is defined as

$$e^{j\theta} = \cos(\theta) + j\sin(\theta).$$

Similarly, we can write a complex number in polar form as

$$z = |z|e^{j\angle z}.$$

We can expand Euler's formula to define a cosine function as

$$\cos(\theta) = \frac{e^{j\theta} + e^{-j\theta}}{2}$$

and a sine function as

$$\sin(\theta) = \frac{e^{j\theta} - e^{-j\theta}}{2j}.$$

The complex conjugate of a complex number $z = a + jb$ is denoted as $z^* = a - jb$, which essentially says we need to negate the imaginary part of z.

The product of a complex number and its conjugate is given by

$$zz^* = |z|^2.$$

The division of a complex number by its magnitude is given by

$$\frac{z}{|z|} = e^{j\angle z}.$$

The sum of two complex numbers is given by

$$(a + jb) + (c + jd) = (a + c) + j(b + d)$$

and the difference of two complex numbers is given by

$$(a + jb) - (c + jd) = (a - c) + j(b - d).$$

The product of two complex numbers is given by

$$(a + jb)(c + jd) = (ac - bd) + j(ad + bc)$$

and the division of two complex numbers is given by

$$\frac{a + jb}{c + jd} = \frac{ac + bd}{c^2 + d^2} + j\frac{bc - ad}{c^2 + d^2}.$$

Generally, the addition and subtraction of two complex numbers is done in rectangular form, and the multiplication and division of two complex numbers is done in polar form.

1.3 Continuous-Time Signals

There are a few types of continuous-time signals that we will encounter in this course. The unit impulse function is defined as

$$\delta(t) = \begin{cases} 0 & \text{for } t \neq 0, \\ \infty & \text{for } t = 0. \end{cases}$$

But this is ultimately best described by the integral definition

$$\int_{-\infty}^{\infty} \delta(t) dt = 1.$$

The unit step function is defined as

$$u(t) = \begin{cases} 0 & \text{for } t < 0, \\ 1 & \text{for } t \geq 0. \end{cases}$$

Lastly, the triangle function is defined as

$$\Delta(t) = \begin{cases} 0 & \text{for } |t| > 1, \\ 1 - |t| & \text{for } |t| \leq 1. \end{cases}$$

We can scale the time axis by a factor of τ to get the triangle function $\Delta(t/\tau)$. The amplitude would be the same, but the width will be τ.

The rectangle function is defined as

$$\text{rect}\left(\frac{t}{\tau}\right) = \begin{cases} 0 & \text{for } |t| > \tau/2, \\ 1 & \text{for } |t| < \tau/2. \end{cases}$$

1.4 Continuous-Time Fourier Transform Pairs

The Fourier transform of a signal $x(t)$ is defined as

$$X(j\Omega) = \int_{-\infty}^{\infty} x(t) e^{-j\Omega t} dt.$$

The inverse Fourier transform of a signal $X(j\Omega)$ is defined as

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(j\Omega) e^{j\Omega t} d\Omega.$$

A table of Fourier transform pairs is given in Table 1.

Signal $x(t)$	**Fourier Transform** $X(j\Omega)$
$e^{-at}u(t)$	$\frac{1}{a+j\Omega}$
$e^{at}u(-t)$	$\frac{1}{a-j\Omega}$
$e^{-a\|t\|}$	$\frac{2a}{a^2+\Omega^2}$
$t^n e^{-at}u(t)$	$\frac{n!}{(a-j\Omega)^{n+1}}$
$\delta(t)$	1
1	$2\pi\delta(\Omega)$
$e^{j\Omega_0 t}$	$2\pi\delta(\Omega-\Omega_0)$
$\cos(\Omega_0 t)$	$\pi\left(\delta(\Omega-\Omega_0)+\delta(\Omega+\Omega_0)\right)$
$\sin(\Omega_0 t)$	$j\pi\left(\delta(\Omega+\Omega_0)-\delta(\Omega-\Omega_0)\right)$
$u(t)$	$\pi\delta(\Omega)+\frac{1}{j\Omega}$
sgn(t)	$\frac{2}{j\Omega}$
$\cos(\Omega_0 t)u(t)$	$\frac{\pi}{2}\left(\delta(\Omega-\Omega_0)+\delta(\Omega+\Omega_0)\right)+\frac{j\Omega}{\Omega^2-\Omega_0^2}$
$\sin(\Omega_0 t)u(t)$	$\frac{\pi}{2j}\left(\delta(\Omega-\Omega_0)-\delta(\Omega+\Omega_0)\right)+\frac{\Omega_0}{\Omega^2-\Omega_0^2}$
$e^{-at}\sin(\Omega_0 t)u(t)$	$\frac{\Omega_0}{(a+j\Omega)^2+\Omega_0^2}$
$e^{-at}\cos(\Omega_0 t)u(t)$	$\frac{a+j\Omega}{(a+j\Omega)^2+\Omega_0^2}$
$\text{rect}\left(\frac{t}{\tau}\right)$	$\tau\text{sinc}\left(\frac{\omega t}{2}\right)$
$\frac{W}{\pi}\text{sinc}(Wt)$	$\text{rect}\left(\frac{\Omega}{2W}\right)$
$\Delta\left(\frac{t}{\tau}\right)$	$\frac{\tau}{2}\text{sinc}^2\left(\frac{\omega t}{4}\right)$
$\frac{W}{2\pi}\text{sinc}^2\left(\frac{Wt}{2}\right)$	$\Delta\left(\frac{\Omega}{2W}\right)$
$\sum_{n=-\infty}^{\infty}\delta(t-nT)$	$\Omega_0\sum_{n=-\infty}^{\infty}\delta(\Omega-n\Omega_0), \qquad \Omega_0=\frac{2\pi}{T}$
$e^{-t^2/2\sigma^2}$	$\sigma\sqrt{2\pi}e^{-\Omega^2\sigma^2/2}$

Table 1: Common Fourier Transform Pairs

2 Discrete-Time Signals

A discrete-time signal is a sequence of numbers that are defined at discrete points in time. A discrete-time signal can be represented as a function of an integer variable n, where n is the time index. A discrete-time signal can be classified as either finite-duration or infinite-duration.

2.1 Finite-Duration Signals

A finite-duration signal is a signal that is nonzero for a finite number of time indices. A finite-duration signal can be represented as

$$x[n] = \begin{cases} x[n], & n_1 \leq n \leq n_2 \\ 0, & \text{otherwise} \end{cases}$$

where n_1 and n_2 are the start and end points of the signal, respectively. We can represent these signals in Python as a NumPy arrays.

2.2 Infinite-Duration Signals

An infinite-duration signal is a signal that is nonzero for an infinite number of time indices. An infinite-duration signal can be represented as

$$x[n] = \begin{cases} x[n], & n \in \mathbb{Z} \\ 0, & \text{otherwise} \end{cases}$$

There is no way to represent an infinite-duration signal in Python, but we can represent a portion of the signal. We can work with infinite-duration signals by real-time systems like we see in the lab.

2.3 Relationship Between Continuous-Time and Discrete-Time Signals

A continuous-time signal can be converted to a discrete-time signal by sampling the continuous-time signal. The continuous-time signal $x_c(t)$ can be sampled at a rate of f_s samples per second or with a sampling period T to get the discrete-time signal $x[n]$. The relationship between the continuous-time signal and the discrete-time signal is given by

$$x[n] = x_c(nT) = x_c\left(\frac{n}{f_s}\right).$$

2.4 Common Discrete-Time Signals

There are several common discrete-time signals that are used in signal processing. Some of these signals are listed in Table 2.

2.5 Periodic Discrete-Time Signals

A discrete-time signal is periodic if it satisfies the following relationship for some integer N and all time indices n

$$x[n] = x[n + N].$$

2.6 Discrete-time Sinusoids

A discrete-time sinusoid is a signal that is periodic in time. A discrete-time sinusoid can be represented as

$$x[n] = A\cos(\omega_0 n + \phi),$$

Signal $x[n]$	**Expression**
Unit impulse	$\delta[n] = \begin{cases} 1, & n = 0 \\ 0, & n \neq 0 \end{cases}$
Unit step	$u[n] = \begin{cases} 1, & n \geq 0 \\ 0, & n < 0 \end{cases}$
Exponential	$x[n] = a^n u[n]$
Sinusoidal	$x[n] = A\cos(\omega_0 n + \phi)$
Rectangular	$x[n] = \begin{cases} 1, & 0 \leq n \leq M \\ 0, & \text{otherwise} \end{cases}$

Table 2: Common Discrete-Time Signals

where A is the amplitude, ω_0 is the frequency, and ϕ is the phase of the signal. We can also represent discrete-sinusoids as complex exponentials

$$x[n] = Ae^{j(\omega_0 n + \phi)}.$$

Discrete-time sinusoids are not guaranteed to be periodic. A discrete-time sinusoid is periodic if the frequency ω_0 is a rational multiple of 2π, i.e., $\omega_0 = \frac{2\pi m}{N}$, where m and N are integers. Choose m and N to be integers where m is the smallest positive integer that satisfies the condition. The period of the discrete-time sinusoid is given by N.

2.7 Discrete-Time Signal Operations

There are several operations that can be performed on discrete-time signals. Some of these operations are listed in Table 3.

Operation	**Expression**
Time shifting	$x[n - n_0]$
Time scaling	$x[an]$
Time reversal	$x[-n]$
Addition	$x_1[n] + x_2[n]$
Multiplication	$x_1[n]x_2[n]$

Table 3: Discrete-Time Signal Operations

We can combine these operations to create more complex signals. For example, we can create a signal that is a time-scaled, time-reversed, and time-scaled discrete-time signal by performing the following operations on a signal $x[n]$

$$\begin{aligned} x_1[n] &= x[-2n-4] \\ &= x[-2(n+2)]. \end{aligned}$$

When performing these operations, it is important to remember that the order of the operations matters. For example, time scaling and time shifting do not commute. That is, $x[an-n_0] \neq x[n-n_0]$. To perform these operations, you should first perform the time scaling and then the time shifting. To do this, get the signal into the form $x[a(n-n_0)]$, time scale by a, then time shift by n_0.

2.8 Energy and Power of Discrete-Time Signals

The energy of a discrete-time signal $x[n]$ is defined as the sum of the squared magnitudes of the signal over all time indices

$$E = \sum_{n=-\infty}^{\infty} |x[n]|^2.$$

The power of a discrete-time signal is defined as the average energy per unit time

$$P = \lim_{N\to\infty} \frac{1}{2N+1} \sum_{n=-N}^{N} |x[n]|^2.$$

3 Checking Linear and Time Invariant (LTI) Systems

3.1 Linearity

To check if a system is linear, we need to verify the principle of superposition. This involves two steps:

1. **Additivity:** If the system is linear, then for any two input signals $x_1[n]$ and $x_2[n]$, and their corresponding outputs $y_1[n]$ and $y_2[n]$, the system should satisfy:

$$x_1[n] + x_2[n] \to y_1[n] + y_2[n]$$

2. **Homogeneity (Scaling):** If the system is linear, then for any input signal $x[n]$ and its corresponding output $y[n]$, and any scalar α, the system should satisfy:

$$\alpha x[n] \to \alpha y[n]$$

To check linearity, we can do it in a single step. For any two input signals $x_1[n]$ and $x_2[n]$, and their corresponding outputs $y_1[n]$ and $y_2[n]$, and any scalars a and b, the system should satisfy:

$$ax_1[n] + bx_2[n] \to ay_1[n] + by_2[n].$$

3.1.1 Example

Consider the system with the following input-output relationship

$$y[n] = 2x[n] + 3.$$

To check if this system is linear, we can use the following test

$$\begin{aligned}
\mathcal{H}\{ax_1[n] + bx[2]\} &= 2(ax_1[n] + b2x_2[n]) + 3 \\
&= a2x_1[n] + b2x_2[n] + 3 \\
&\neq y_1[n] + y_2[n].
\end{aligned}$$

3.2 Time-Invariance

To check if a system is time-invariant, we need to verify that a time shift in the input signal results in an identical time shift in the output signal. Specifically, for any input signal $x[n]$ and its corresponding output $y[n]$, and any integer k, the system should satisfy:

$$x[n-k] \to y[n-k]$$

To check time-invariance, first shift the input. This would amount to adding a $-N$ term to every instance of $x[n]$ in the system (in the square brackets). Then, shift the output by replacing every instance of n in the output with $n - N$. If the system is time-invariant, the two should be equal.

3.2.1 Example

Let's consider the system with the following input-output relationship

$$\begin{aligned}
y[n] &= \mathcal{H}\{x[n]\} \\
&= x[2n].
\end{aligned}$$

To shift the input, we add another $-N$ term to every $x[n]$ in the system. We will call this system response $\tilde{y}[n]$. We have

$$\tilde{y}[n] = x[2n - N]$$

To shift the output, we will replace every instance of n in the output with $n - N$

$$\begin{aligned} y[n-N] &= x[2(n-N)] \\ &= x[2n-2N]. \end{aligned}$$

We see that $\tilde{y}[n] \neq y[n - N]$, so the system is not time-invariant.

3.2.2 Example

Consider the system with the following input-output relationship

$$\begin{aligned} y[n] &= \mathcal{H}\{x[n]\} \\ &= (-1)^n x[n]. \end{aligned}$$

To shift the input, we add another $-N$ term to every $x[n]$ in the system. We will call this system response $\tilde{y}[n]$. We have

$$\tilde{y}[n] = (-1)^n x[n - N]$$

To shift the output, we will replace every instance of n in the output with $n - N$

$$\begin{aligned} y[n-N] &= (-1)^{n-N} x[n-N] \\ &= (-1)^{n-N} x[n-N]. \end{aligned}$$

We see that $\tilde{y}[n] = y[n - N]$, so the system is not time-invariant.

3.3 Memoryless Systems

A system is memoryless if the output at any time n depends only on the input at that time n. A memoryless system can be described by the following relationship:

$$y[n] = g(x[n]).$$

If the system description can be written in this form, then the system is memoryless.

3.4 Causality

A system is causal if the output at any time n depends only on the input at that time n and at times $n - 1$, $n - 2$, etc. A causal system can be described by the following relationship:

$$y[n] = g(x[n], x[n-1], x[n-2], \ldots).$$

If the system description can be written in this form, then the system is causal.

A system is not causal if the output at time n depends on the input at time $n + 1$ or later.

3.5 Stability

A system is stable if the output is bounded for any bounded input. A system is stable if the output is bounded for any input signal $x[n]$ that satisfies

$$|x[n]| < M$$

for all n, where M is a constant. If the output satisfies the condition that $|y[n]| < N$ for all n, where N is a constant, then the system is stable. If the output is unbounded for any bounded input, then the system is unstable.

4 Convolution

The convolution of two signals $x[n]$ and $h[n]$ is defined as

$$y[n] = x[n] * h[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k].$$

4.1 Performing Convolution

To perform convolution, we can use the following steps:

1. **Plot the signals:** Plot the input signal $x[k]$ and the impulse response $h[k]$ to get a sense of the signals. Remember, we are interested in plotting as functions of k not n.
2. **Flip one of the functions:** Flip either the input signal $x[k]$ or the impulse response $h[k]$ to get $x[-k]$ or $h[-k]$. (Typically choose the simpler of the two to flip. For the notation here we will use $h[-k]$).
3. **Shift the impulse response:** Shift the flipped term by n to get $h[-(k-n)] = h[n-k]$.
4. **Multiply and sum:** Multiply the input signal $x[k]$ with the flipped and shifted impulse response $h[n-k]$ and sum over all k to get the output signal $y[n]$ at point n, which is how much you shifted $h[-(k-n)]$.

4.2 Properties of Convolution

The convolution operation has several important properties:

1. **Commutativity:**
$$x[n] * h[n] = h[n] * x[n]$$
2. **Associativity:**
$$x[n] * (h[n] * g[n]) = (x[n] * h[n]) * g[n]$$
3. **Distributivity:**
$$x[n] * (h[n] + g[n]) = x[n] * h[n] + x[n] * g[n]$$
4. **Identity:**
$$x[n] * \delta[n] = x[n]$$
5. **Time Invariance:**
$$x[n-n_0] * h[n] = (x[n] * h[n])[n-n_0]$$

5 DTFT

The Discrete-Time Fourier Transform (DTFT) of a signal $x[n]$ is defined as

$$X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n}.$$

The DTFT is a continuous function of frequency ω, and is periodic with period 2π.

5.1 Inverse DTFT

The inverse DTFT is defined as

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

To compute the inverse DTFT, we can use the inverse DTFT formula, or you can use the DTFT pairs to find the inverse DTFT. The DTFT pairs are listed in Table 5. You will also might also need to use the properties of the DTFT, which are listed in Table 4. If you use the DTFT pairs, you also might need to use partial fraction expansion to simplify the inverse DTFT.

5.2 DTFT Symmetry

The DTFT of a real signal is conjugate symmetric, i.e., $X(e^{j\omega}) = X^*(e^{-j\omega})$. This can be shown by substituting $x[n] = x^*[n]$ into the definition of the DTFT.

Similarly, if the signal is imaginary, the DTFT is conjugate anti-symmetric, i.e., $X(e^{j\omega}) = -X^*(e^{-j\omega})$.

If the $x[n]$ is real and even, then the DTFT is real and even. If $x[n]$ is real and odd, then the DTFT is imaginary and odd.

5.3 DTFT Properties

The DTFT has several properties that are useful in signal processing. Some of these properties are listed in Table 4.

5.4 Common DTFT Pairs

A list of common DTFT pairs is shown in Table 5.

Property	**DTFT Relationship**
Linearity	$ax_1[n] + bx_2[n] \Longleftrightarrow aX_1(e^{j\omega}) + bX_2(e^{j\omega})$
Time Shifting	$x[n - n_0] \Longleftrightarrow e^{-j\omega n_0} X(e^{j\omega})$
Frequency Shifting	$e^{j\omega_0 n} x[n] \Longleftrightarrow X(e^{j(\omega-\omega_0)})$
Conjugation	$x^*[n] \Longleftrightarrow X^*(e^{-j\omega})$
Frequency Differentiation	$x[-n] \Longleftrightarrow X(e^{-j\omega})$
Time Scaling	$nx[n] \Longleftrightarrow j\frac{d}{d\omega} X(e^{j\omega})$
Convolution	$(x_1 * x_2)[n] \Longleftrightarrow X_1(e^{j\omega}) X_2(e^{j\omega})$
Multiplication	$x_1[n]x_2[n] \Longleftrightarrow \frac{1}{2\pi} \int_{-\pi}^{\pi} X_1(e^{j\theta}) X_2(e^{j(\omega-\theta)}) d\theta$
Parseval's Theorem	$\sum_{n=-\infty}^{\infty} \lvert x[n] \rvert^2 = \frac{1}{2\pi} \int_{-\pi}^{\pi} \lvert X(e^{j\omega}) \rvert^2 d\omega$
Parseval's Theorem for two signals	$\sum_{n=-\infty}^{\infty} x_1[n] x_2^*[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} X_1(e^{j\omega}) X_2^*(e^{j\omega}) d\omega$

Table 4: Properties of the DTFT

Signal $x[n]$	**DTFT** $X(e^{j\omega})$
$\delta[n]$	1
$\delta[n-n_0]$	$e^{-j\omega n_0}$
$a^n u[n]$	$\frac{1}{1-ae^{-j\omega}}, \quad \lvert a\rvert < 1$
$u[n]$	$\frac{1}{1-e^{-j\omega}} + \pi \sum_{k=-\infty}^{\infty} \delta(\omega - 2\pi k)$
$(n+1)a^n u[n], \quad \lvert a\rvert < 1$	$\frac{1}{(1-ae^{-j\omega})^2}$
$\frac{r^n \sin(\omega_p(n+1))}{\sin(\omega_p)} u[n], \quad \lvert r\rvert < 1$	$\frac{1}{1-2r\cos(\omega_p)e^{-j\omega}+r^2 e^{-j\omega}}$
$\frac{\sin(\omega_c n)}{\pi n}$	$X(e^{j\omega}) = \begin{cases} 1, & \lvert\omega\rvert \leq \omega_c \\ 0, & \text{otherwise} \end{cases}$
$x[n] = \begin{cases} 1, & 0 \leq n \leq M \\ 0, & \text{otherwise} \end{cases}$	$\frac{\sin(\omega(M+1)/2)}{\sin\left(\frac{\omega}{2}\right)} e^{-j\omega M/2}$
$e^{j\omega_0 n}$	$2\pi\delta(\omega - \omega_0)$
$\cos(\omega_0 n)$	$\pi\left(\delta(\omega-\omega_0) + \delta(\omega+\omega_0)\right)$
$\sin(\omega_0 n)$	$\frac{\pi}{j}\left(\delta(\omega-\omega_0) - \delta(\omega+\omega_0)\right)$

Table 5: Common DTFT Pairs

Signal $x[n]$	**Transform** $X(z)$	ROC
$\delta[n]$	1	$\lvert z\rvert > 0$
$u[n]$	$\frac{1}{1-z^{-1}}$	$\lvert z\rvert > 1$
$-u[n-1]$	$\frac{1}{1-z^{-1}}$	$\lvert z\rvert < 1$
$\delta[n-m]$	z^{-m}	all z except 0 $(m > 0)$ or ∞ $(m < 0)$
$a^n u[n]$	$\frac{1}{1-az^{-1}}$	$\lvert z\rvert > \lvert a\rvert$
$-a^n u[-n-1]$	$\frac{1}{1-az^{-1}}$	$\lvert z\rvert < \lvert a\rvert$
$na^n u[n]$	$\frac{az^{-1}}{(1-az^{-1})^2}$	$\lvert z\rvert > \lvert a\rvert$
$-na^n u[-n-1]$	$\frac{az^{-1}}{(1-az^{-1})^2}$	$\lvert z\rvert < \lvert a\rvert$
$\cos(\omega_0 n)u[n]$	$\frac{1-z^{-1}\cos(\omega_0)}{1-2z^{-1}\cos(\omega_0)+z^{-2}}$	$\lvert z\rvert > 1$
$\sin(\omega_0 n)u[n]$	$\frac{z^{-1}\sin(\omega_0)}{1-2z^{-1}\cos(\omega_0)+z^{-2}}$	$\lvert z\rvert > 1$
$r^n\cos(\omega_0 n)u[n]$	$\frac{1-rz^{-1}\cos(\omega_0)}{1-2rz^{-1}\cos(\omega_0)+r^2z^{-2}}$	$\lvert z\rvert > r$
$r^n\sin(\omega_0 n)u[n]$	$\frac{rz^{-1}\sin(\omega_0)}{1-2rz^{-1}\cos(\omega_0)+r^2z^{-2}}$	$\lvert z\rvert > r$
$\begin{cases} a^n & 0 \le n \le N-1 \\ 0 & \text{otherwise} \end{cases}$	$\frac{1-a^N z^{-N}}{1-az^{-1}}$	$\lvert z\rvert > 0$

Table 6: Common Z-transform pairs.

6 Z-Transform

The Z-transform of some signal $x[n]$ is defined as

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

where z is a complex variable $z = re^{j\omega}$. This is a generalization of the DTFT, which is the Z-transform evaluated on the unit circle $z = e^{j\omega}$.

When we compute the Z-transform, we need to consider the set of values of z for which the sum converges. This is called the region of convergence (ROC). The ROC is a ring in the complex plane, defined by the inner and outer radii of the ring. The Z-transform is a function of z that is defined in the ROC.

A table of common Z-transform pairs is given in Table 6.

6.1 Regions of Convergence

The following rules apply to the ROC of the Z-transform:

- The ROC will be of the form $0 \leq r_R < |z|$, or $|z| < r_L < \infty$ or the annulus (ring) $0 \leq r_R < |z| < r_L \infty$.
- The DTFT of a signal $x[n]$ exists if the ROC of the Z-transform includes the unit circle $|z| = 1$.
- The ROC cannot contain any poles.
- If $x[n]$ is finite-duration, the ROC will be the entire z-plane except for (possibly) $z = 0$ and $z = \infty$.
- If $x[n]$ is right-sided, the ROC will be outside a circle in the z-plane, extending from the outermost pole.
- If $x[n]$ is left-sided, the ROC will be inside a circle in the z-plane, extending from the innermost pole.
- If $x[n]$ is two-sided, the ROC will be an annulus in the z-plane, bounded on the interior and exterior by a pole and not containing any poles.
- The ROC must be a connected region.

6.2 Properties of the Z-Transform

The Z-transform has many properties that are similar to the DTFT. Some of the most important properties are given in Table 7.

Property	**Transform**
Linearity	$ax_1[n] + bx_2[n] \Longleftrightarrow aX_1(z) + bX_2(z)$
Time Shifting	$x[n - n_0] \Longleftrightarrow z^{-n_0} X(z)$
Convolution	$x_1[n] * x_2[n] \Longleftrightarrow X_1(z)X_2(z)$
Time Reversal	$x[-n] \Longleftrightarrow X(z^{-1})$
Differentiation	$nx[n] \Longleftrightarrow -z\frac{d}{dz}X(z)$

Table 7: Properties of the Z-transform.

6.3 Inverse Z-Transform

The inverse Z-transform is given by the contour integral

$$x[n] = \frac{1}{2\pi j} \oint_C X(z) z^{n-1} dz.$$

Those who have taken a class in complex variables might be able to compute this integral, but for the most part, we are going to use tables of Z-transform pairs to find the inverse Z-transform.

6.3.1 Inspection Method

The inspection method is a way to find the inverse Z-transform by inspection of the Z-transform. This method is useful when the Z-transform is a rational function of z. Here, we will just match the Z-transform with the corresponding pair on the table.

6.3.2 Partial Fraction Expansion

The partial fraction expansion is a method to find the inverse Z-transform when the Z-transform is a rational function of z. The Z-transform is written as a sum of simpler fractions, and the inverse Z-transform is found by inspection of the table. Partial fraction expansion is most commonly performed by the Heaviside cover-up method. The Heaviside cover-up method is a way to find the coefficients of the partial fraction expansion by evaluating the Z-transform at the poles of the Z-transform.

A typical Z-transform will have the form

$$X(z) = \frac{\sum_{k=0}^{M} b_k z^{-k}}{\sum_{k=0}^{N} a_k z^{-k}} = \frac{b_0}{a_0} \frac{\prod_{k=1}^{M}(1 - c_k z^{-1})}{\prod_{k=1}^{N}(1 - d_k z^{-1})}.$$

If $N < M$ and there are no repeating poles, the Z-transform takes the form

$$X(z) = \sum_{k=1}^{N} \frac{A_k}{1 - d_k z^{-1}}$$

and the coefficients A_k can be found by the Heaviside cover-up method

$$A_k = (1 - d_k z^{-1}) X(z) \Big|_{z=d_k}.$$

If $M \geq N$, the Z-transform can be written as

$$X(z) = \sum_{r=0}^{M-N} B_r z^{-1} + \sum_{k=1}^{N} \frac{A_k}{1 - d_k z^{-1}}$$

where the coefficients A_k can be found by the Heaviside cover-up method, and the coefficients B_r can be found by polynomial division.

If there are repeating poles, the Z-transform can be written as

$$X(z) = \sum_{r=0}^{M-N} B_r z^{-1} + \sum_{k=1, k \neq i}^{N} \frac{A_k}{1 - d_k z^{-1}} + \sum_{m=1}^{s} \frac{C_m}{(1 - d_i z^{-1})^m}$$

where C_m can be found by the Heaviside cover-up method or other methods.

6.3.3 Power Series Expansion

This method is essentially coefficient matching. The Z-transform is expanded as a power series, and the coefficients with each z term are matched with the corresponding value of $x[n]$.

6.4 Transfer Functions

Given some difference equation

$$\sum_{k=0}^{N} a_k y[n-k] = \sum_{m=0}^{M} b_m x[n-m]$$

we take the Z-transform of both sides to get

$$\sum_{k=0}^{N} a_k z^{-k} Y(z) = \sum_{m=0}^{M} b_m z^{-m} X(z)$$

and the transfer function is defined as

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{m=0}^{M} b_m z^{-m}}{\sum_{k=0}^{N} a_k z^{-k}}.$$

We also notice that the relationship between the transfer function and the impulse response is given by the Z-transform pair

$$h[n] \Longleftrightarrow H(z).$$

If an LTI system $\mathcal{H}$ has the input/output relationship

$$y[n] = \mathcal{H}\{x[n]\}$$

the output is represented as the convolution of the input with the impulse response in the time-domain

$$y[n] = h[n] * x[n].$$

In the Z-domain, the output is represented as the product of the input with the transfer function

$$Y(z) = H(z)X(z).$$

7 Sampling

Suppose we have a continuous-time signal $x_c(t)$. In undergraduate signals and systems we learned that we can model this process by multiplying this signal with a train of impulse functions, which we will call $s(t)$. Mathematically, this is represented as

$$\begin{aligned}
x_s(t) &= x_c(t)s(t) \\
&= x_c(t) \sum_{n=-\infty}^{\infty} \delta(t - nT) \\
&= \sum_{n=-\infty}^{\infty} x_c(t)\delta(t - nT) \\
&= \sum_{n=-\infty}^{\infty} x_c(nT)\delta(t - nT).
\end{aligned}$$

where T is the sampling period. The signal $x_s(t)$ is called the sampled signal, but *it is still a continuous-time signal* because it is defined for all values of a continuous variable t (the points where $t \neq nT$ this function is zero).

In your first semester of signals you also learned that this sampled signal can be represented in the frequency domain as

$$X_s(j\Omega) = \frac{1}{T} \sum_{n=-\infty}^{\infty} X_c(j(\Omega - n\Omega_s)),$$

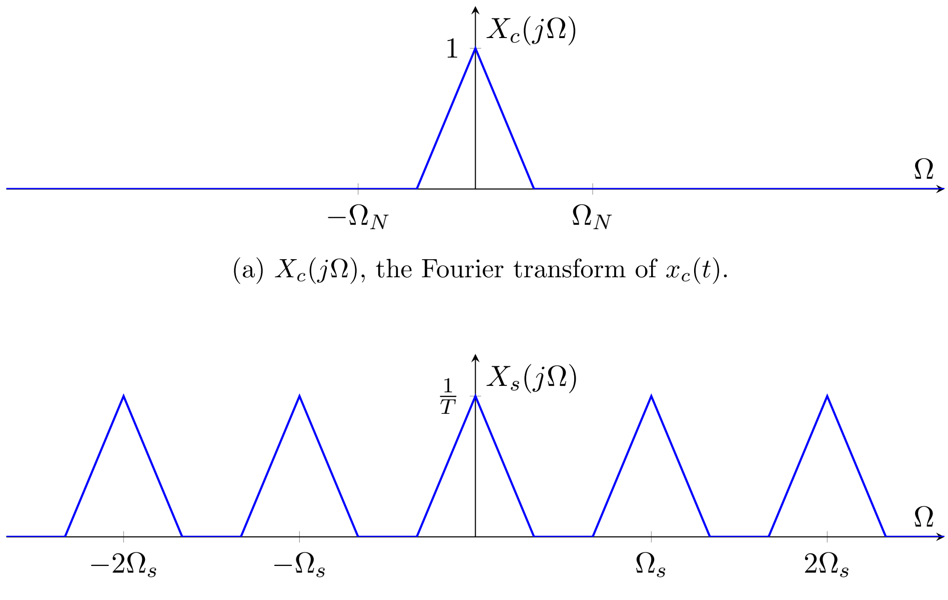
where $\Omega_s = 2\pi/T$ is the sampling frequency. Please refer to your previous textbooks for the details of the derivation if you are curious. The Fourier transform $X_s(j\Omega)$ is a continuous-time Fourier transform, with the same frequency axis as the continuous-time Fourier transform $X_c(j\Omega)$ (the frequency variable Ω is the continuous-time frequency variable). This is an important result with two major implications:

1. The Fourier transform of a sampled signal $X_s(j\Omega)$ is similar to the Fourier transform of the original signal $X_c(j\Omega)$, but the spectrum is repeated at integer multiples of the sampling frequency Ω_s.
2. The amplitude of the spectrum of the sampled signal is scaled by $1/T$.

The Fourier transform of a sampled signal is shown in Fig. 1. The top plot shows the Fourier transform of the continuous-time signal $X_c(j\Omega)$, and the bottom plot shows the Fourier transform of the sampled signal $X_s(j\Omega)$. The spectrum of the sampled signal is a series of copies of the spectrum of the continuous-time signal, with each copy scaled by $1/T$.

7.1 Nyquist Sampling Theorem

The Nyquist sampling theorem states that a continuous-time signal $x_c(t)$ can be perfectly reconstructed from its samples $x[n] = x_c(nT)$ if the signal is bandlimited to Ω_N, where Ω_N is the highest frequency component in the signal. The Nyquist sampling theorem states that the sampling frequency Ω_s must be greater than $2\Omega_N$ to perfectly reconstruct the signal. In other words, if we sample fast enough, we can space the extra spectral copies far enough apart that they do not overlap with the original spectrum. With this, it is possible to perfectly reconstruct the signal from its samples by using an ideal low-pass filter to remove the extra spectral copies. See Fig. 2 as a visualization.



(a) $X_c(j\Omega)$, the Fourier transform of $x_c(t)$.

(b) $X_s(j\Omega)$, the Fourier transform of $x_s(t)$.

Figure 1: The Fourier transform of a sampled signal.

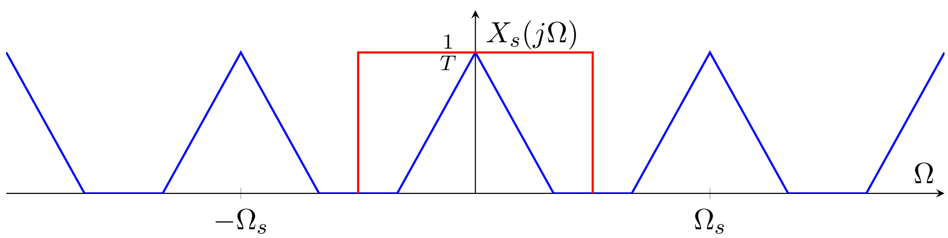


Figure 2: Reconstruction of a signal from its samples. Ideal low-pass filter is indicated in red.

However, if do not sample below the Nyquist rate, the extra spectral copies will overlap with the original spectrum, and the signal cannot be perfectly reconstructed. This is shown in Fig. 3. If the sampling frequency is less than $2\Omega_N$, then the signal cannot be perfectly reconstructed.

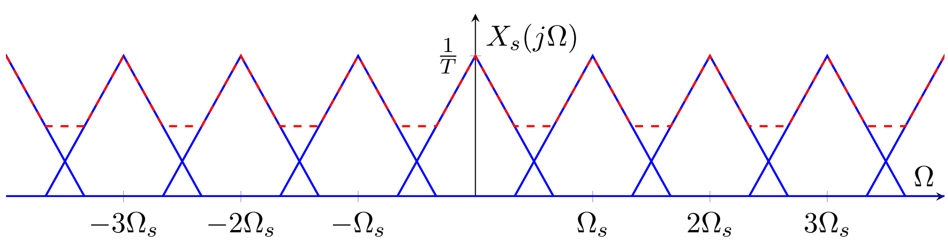


Figure 3: Example of aliasing. The spectral copies are shown individual in blue. When added together, the higher frequency copies overlap with the original spectrum. This is shown by the dashed red line.

7.2 Relationship With the DTFT

We can relate $X_c(j\Omega)$ with the DTFT $X(e^{j\omega})$ of the discrete-time signal $x[n] = x_c(nT)$ by noting the relationship between the digital frequency ω and the continuous-time frequency Ω, which is $\omega = \Omega T$. The relationship between the DTFT and the continuous-time Fourier transform is

$$X(e^{j\omega}) = X_s(j\Omega)\Big|_{\Omega=\omega/T}.$$

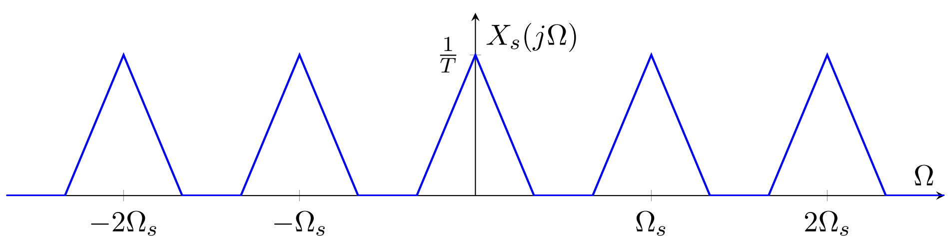
Similarly, the converse is also true

$$X_s(j\Omega) = X(e^{j\omega})\Big|_{\omega=T\Omega}.$$

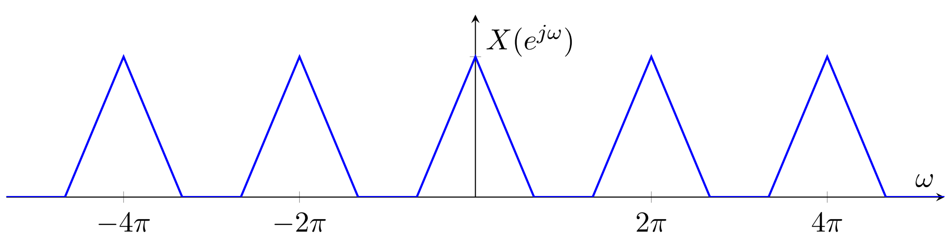
Essentially, we need to adjust the x-axis of the frequency domain by a factor of T to relate the DTFT to the continuous-time Fourier transform. This is shown in Fig. 4. We notice that if $\omega = \Omega T$, then the sampling frequency turns into

$$\begin{aligned}\omega_s &= \Omega_s T\\ &= \frac{2\pi}{T}T\\ &= 2\pi.\end{aligned}$$

Thus, we see that the spectral copies appear at integer multiples of 2π. (And why the DTFT is periodic with period 2π!)



(a) $X_s(j\Omega)$, the Fourier transform of $x_s(t)$.



(b) $X(e^{j\omega})$, the DTFT of $x[n]$.

Figure 4: The Fourier transform of a sampled signal.

7.3 Discrete-Time Processing of Continuous-Time Signals

Consider the following signal processing chain in Fig. 5. We have a continuous-time signal $x_c(t)$ that is sampled to produce $x[n] = x_c(nT)$. This signal is then processed by a discrete-time system with impulse response $h[n]$. The output of the system is $y[n] = x[n] * h[n]$. The output is then reconstructed to produce the continuous-time signal $y_c(t)$. The reconstruction is assumed to be perfect—using an ideal low-pass filter.

Looking at the $x[n] \longrightarrow y[n]$ portion of the signal processing chain, we can write the relationship as

$$Y(e^{j\omega}) = H(e^{j\omega})X(e^{j\omega}),$$

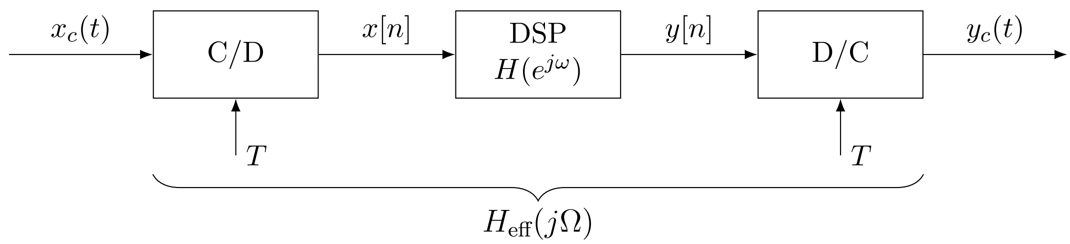


Figure 5: Discrete-time processing of continuous-time signals.

where $H(e^{j\omega})$ is the frequency response of the discrete-time system. We can relate this to the continuous-time signals by noting that the continuous-time frequency Ω is related to the digital frequency ω by $\Omega = \omega/T$. Thus, we can write this relationship as

$$Y(e^{j\Omega T}) = H(e^{j\Omega T})X(e^{j\Omega T}).$$

The output of the reconstruction filter is then

$$\begin{aligned} Y_r(j\Omega) &= H_r(j\Omega)Y(e^{j\Omega T}) \\ &= H_r(j\Omega)H(e^{j\Omega T})\frac{1}{T}\sum_{k=-\infty}^{\infty} X_c\left(j\left(\Omega - \frac{2\pi k}{T}\right)\right) \end{aligned}$$

where $H_r(j\Omega)$ is the frequency response of the reconstruction filter. Assuming that the original signal $x_c(t)$ is bandlimited to π/T and that the frequency response of the reconstruction filter is ideal such that $H_r(j\Omega) = T$ for $|\Omega| < \pi/T$, we can write the output of the reconstruction filter as

$$Y_r(j\Omega) = \begin{cases} H(e^{j\Omega T})X_c(j\Omega) & |\Omega| < \pi/T \\ 0 & \text{otherwise.} \end{cases}$$

We can describe the total system response as

$$Y_c(j\Omega) = H_{\text{eff}}(j\Omega)X_c(j\Omega),$$

where (assuming ideal reconstruction and sampling above the Nyquist rate)

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

There are a few conditions you need to meet for this to work:

- The original signal $x_c(t)$ must be bandlimited to π/T.
- The sampling frequency must be greater than Nyquist.
- The DSP system is LTI.
- The reconstruction filter must be ideal.

7.3.1 Impulse Invariance

The impulse invariance method is a method to design a discrete-time system that approximates a continuous-time system $y(t) = h_c(t) * x_c(t)$. The idea is to sample the continuous-time impulse response $h_c(t)$ to produce

$$h[n] = Th_c(nT).$$

The discrete-time system is then designed to have impulse response $h[n]$.

7.3.2 Continuous-Time Processing of Discrete-Time Signals

The continuous-time processing of discrete-time signals is similar to the discrete-time processing of continuous-time signals. The signal processing chain is shown in Fig. 6. The discrete-time signal $x[n]$ is converted to a continuous-time signal $x_c(t)$ using an ideal reconstruction. The continuous-time signal is then processed by a continuous-time system with impulse response $h_c(t)$. The output of the system is $y_c(t) = x_c(t) * h_c(t)$. The output is then sampled to produce $y[n] = y_c(nT)$. The reconstruction is assumed to be perfect—using an ideal low-pass filter. We see that overall system response is

$$H(e^{j\omega}) = H_c\left(j\frac{\omega}{T}\right).$$

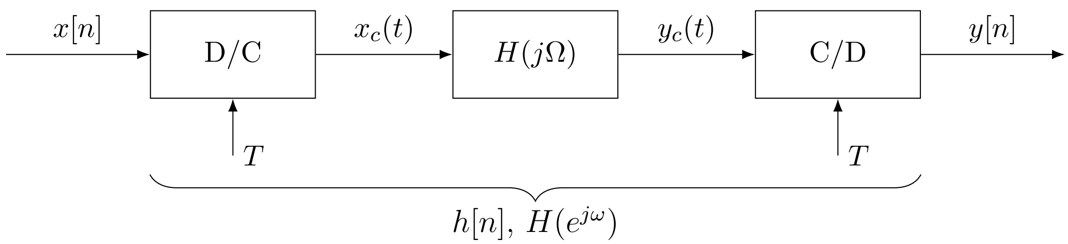


Figure 6: Continuous-time processing of discrete-time signals.

7.4 Downsampling

Oftentimes we need to reduce the sampling rate of a signal. This is called downsampling or decimation. Downsampling is accomplished by keeping every Mth sample of the signal. Mathematically, this is represented as

$$x_d[n] = x[nM].$$

This is similar to continuous-to-discrete conversion because we worry about aliasing, so we will need to use an anti-aliasing filter to mitigate these effects.

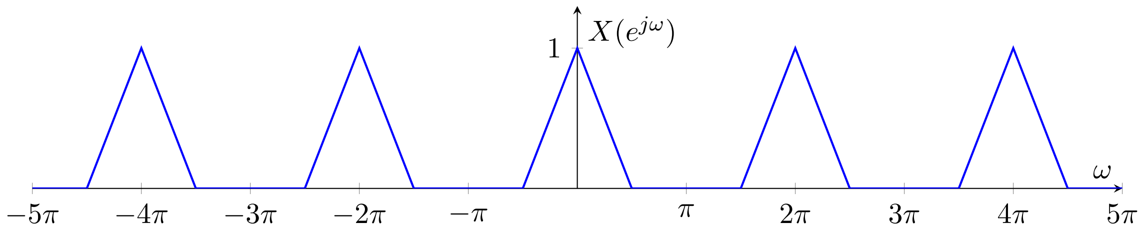
In the frequency domain, downsampling is represented as

$$X_d(e^{j\omega}) = \frac{1}{M}\sum_{i=0}^{M-1} X\left(e^{j\left(\frac{\omega - 2\pi i}{M}\right)}\right)$$

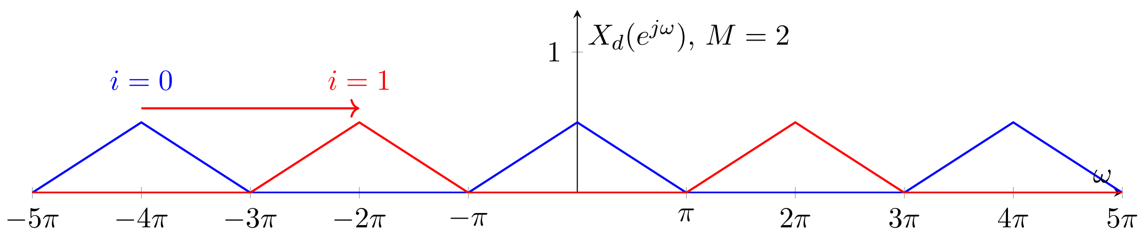
where $X(e^{j\omega})$ is the original DTFT of $x[n]$. This has the effect of scaling down the vertical axis by $1/M$. It also expands the frequency and adds copies of the spectrum at integer multiples of $2\pi/M$.

We see this depicted in Fig. 7. The top plot (Fig. 7a) shows the original signal $X(e^{j\omega})$, the middle plot (Fig. 7b shows the downsampled signal $X_d(e^{j\omega})$ with $M = 2$, and the bottom plot (Fig. 7c) shows the downsampled signal $X_d(e^{j\omega})$ with $M = 3$. We see that the spectrum is scaled by $1/M$ and that the copies of the spectrum are spaced by $2\pi/M$. Further, when $M = 3$, there is aliasing because the copies of the spectrum overlap with the original spectrum.

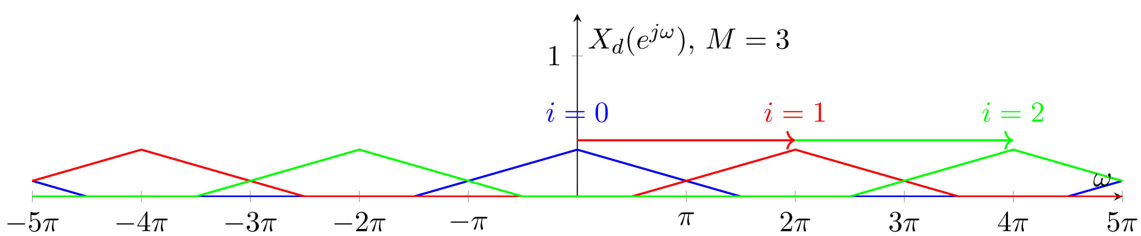
To avoid aliasing, we hope that the signal is bandlimited to π/M. This is the Nyquist rate for the downsampled signal. If the signal is not bandlimited to π/M, then we need to use an anti-aliasing filter to remove the extra spectral copies. The anti-aliasing filter should have a cutoff frequency of π/M. This is shown in Fig. 8.



(a) Original signal $X(e^{j\omega})$.



(b) Downsampled signal $X_d(e^{j\omega})$ with $M = 2$.



(c) Downsampled signal $X_d(e^{j\omega})$ with $M = 3$.

Figure 7: Downsampling of a signal (represented in the frequency-domain).

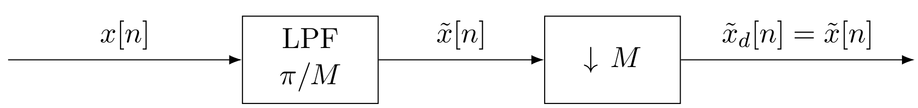


Figure 8: Anti-aliasing filter for downsampling.

7.5 Upsampling

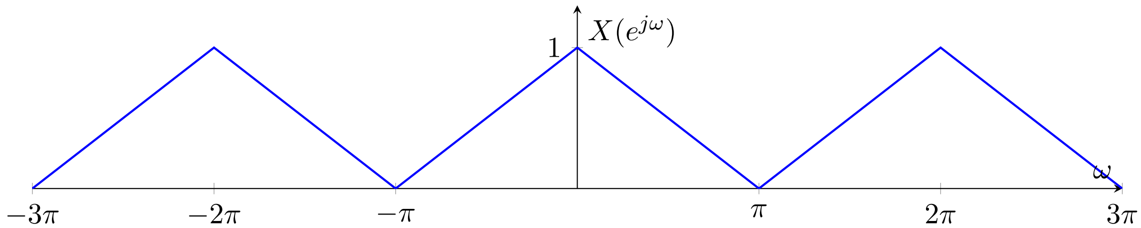
Upsampling is the opposite of downsampling. It is the process of increasing the sampling rate of a signal. This is accomplished by inserting $M - 1$ zeros between each sample of the signal. Mathematically, this is represented as

$$\begin{aligned} x_e[n] &= \begin{cases} x[n/L] & n = 0, \pm L, \pm 2L, \ldots \\ 0 & \text{otherwise.} \end{cases} \\ &= \sum_{k=-\infty}^{\infty} x_e[n]\delta[n - kL]. \end{aligned}$$

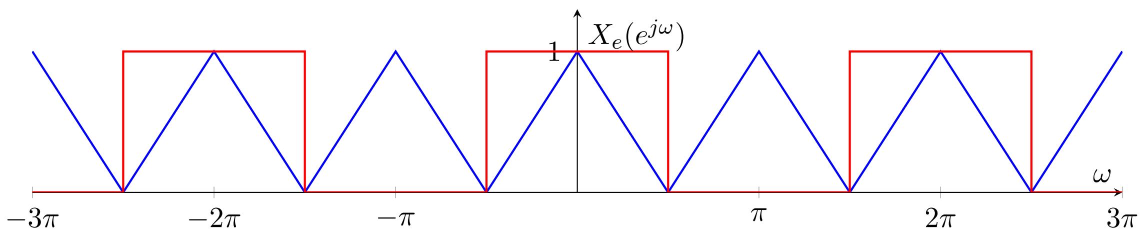
We can extend this to the frequency-domain by noting that the DTFT of the upsampled signal is

$$X_e(e^{j\omega}) = X(e^{j\omega L}).$$

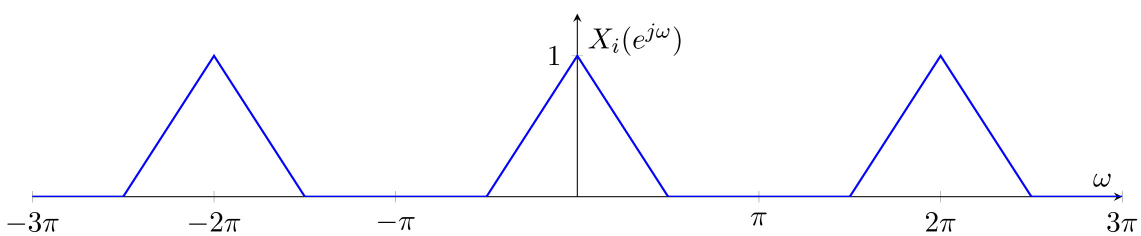
This corresponds to simply scaling in the frequency domain. This is shown in Fig. 9 where $L = 2$. The top plot (Fig. 9a) shows the original signal $X(e^{j\omega})$, the middle plot (Fig. 9b shows the upsampled signal $X_e(e^{j\omega})$ with $L = 2$ with the anti-imaging low-pass filter, and the bottom plot (Fig. 9c) shows the upsampled signal $X_e(e^{j\omega})$ with $L = 3$. We see that the spectrum is scaled in frequency by a factor of L.



(a) Original signal $X(e^{j\omega})$.



(b) Upsampled signal $X_e(e^{j\omega})$ with $L = 2$. Red line indicates the anti-imaging filter.



(c) Upsampled and interpolated signal $X_i(e^{j\omega})$ with $L = 2$.

Figure 9: Upsampling of a signal (represented in the frequency-domain).

7.5.1 Combining Decimation to Interpolation to Achieve Non-Integer Sampling Rates

Suppose we have a signal $x[n]$ that we want to upsample by a factor of L. We can do this by first upsampling by a factor of L and then downsampling by a factor of M, which adjusts the sampling rate by a factor L/M. This is shown in Fig. 10. When we do this it is advantageous to sample only once, so we will use a low-pass filter with a cutoff frequency of $\omega_c = \pi/\max(L, M)$. If the original $x[n]$ had a sampling period of T, the sampling period of $\tilde{x}_d[n]$ would be $\frac{TM}{L}$.

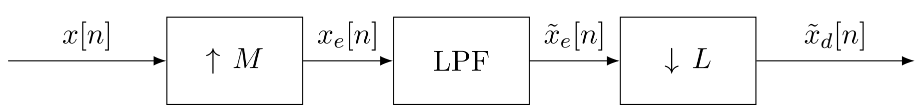


Figure 10: Combining decimation and interpolation to achieve non-integer sampling rates.

7.6 Multirate Signal Processing

Because decimators and expanders are not LTI systems, they do follow the commutative property. This means, we cannot rearrange them at will in the signal processing chain. However, we can apply the noble identity to rearrange them. This is seen in Fig. 11. This says we can switch the decimator and the filter if we expand the filter by a factor of M and then decimate by a factor of M. Suppose we have some $H(z)$ that is FIR and has some transfer function

$$H(z) = 1 + a_1 z^{-1} + a_2 z^{-2} + \ldots + a_{N-1} z^{-(N-1)}.$$

In the time-domain, this is

$$h[n] = \delta[n] + a_1 \delta[n-1] + a_2 \delta[n-2] + \ldots + a_{N-1} \delta[n-(N-1)].$$

If we modify $H(z) \to H(z^M)$ we would have

$$H(z^M) = 1 + a_1 z^{-M} + a_2 z^{-2M} + \ldots + a_{N-1} z^{-(N-1)M}.$$

and then in the time domain the altered impulse response $\tilde{h}[n]$ would be

$$\tilde{h}[n] = \delta[n] + a_1 \delta[n-M] + a_2 \delta[n-2M] + \ldots + a_{N-1} \delta[n-(N-1)M].$$

This is equivalent to inserting $M-1$ zeros between each sample of the original impulse response.

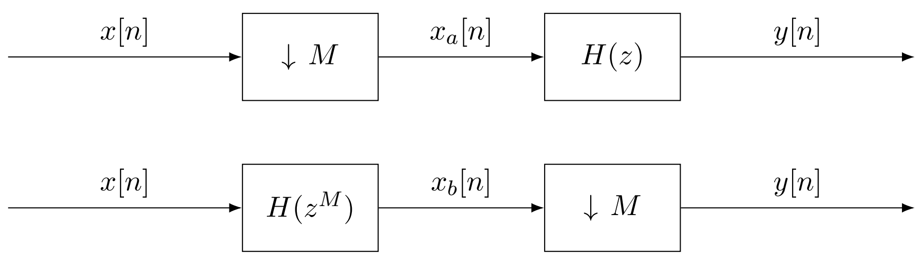


Figure 11: Multirate signal processing system.