Chapter 3

Sampling

3.1 Introduction

Most discrete-time signals come from sampling a continuous-time signal, such as speech and audio signals, radar and sonar data, and seismic and biological signals. The process of converting these signals into digital form is called analog-to-digital (A-D) conversion. The reverse process of reconstructing an analog signal from its samples is known as digital-to-analog (D-A) conversion. This chapter examines the issues related to A-D and D-A conversion. Fundamental to this discussion is the sampling theorem, which gives precise conditions under which an analog signal may be uniquely represented in terms of its samples.

3.2 Analog-to-Digital Conversion

An analog-to-digital converter (A-D) transforms an analog signal into a digital sequence. The input to the A-D converter, \( x_a(t) \), is a real-valued function of a continuous variable, \( t \). Thus, for each value of \( t \), the function \( x_a(t) \) may be any real number. The output of the A-D is a bit stream that corresponds to a discrete-time sequence, \( x(n) \), with an amplitude that is quantized, for each value of \( n \), to one of a finite number of possible values. The components of an A-D converter are shown in Fig. 3.1. The first is the sampler, which is sometimes referred to as a continuous-to-discrete (C-D) converter, or ideal A-D converter. The sampler converts the continuous-time signal \( x_a(t) \) into a discrete-time sequence \( x(n) \) by extracting the values of \( x_a(t) \) at integer multiples of the sampling period, \( T_s \),

\[
x(n) = x_a(nT_s)
\]

The components of an analog-to-digital converter are:

- Sampler
- Quantizer
- Encoder

Fig. 3.1: The components of an analog-to-digital converter.

3.2.1 Periodic Sampling

Typically, discrete-time signals are formed by periodically sampling a continuous-time signal

\[
x(n) = x_a(nT_s)
\]

(3.1)

The sample spacing \( T_s \) is the sampling period, and \( f_s = 1/T_s \) is the sampling frequency in samples per second. A convenient way to view this sampling process is illustrated in Fig. 3.2(a). First, the continuous-time signal is multiplied by a periodic sequence of impulses,

\[
s_a(t) = \sum_{n=-\infty}^{\infty} \delta(t-nT_s)
\]

to form the sampled signal

\[
x_s(t) = x_a(t)s_a(t) = \sum_{n=-\infty}^{\infty} x_a(nT_s)\delta(t-nT_s)
\]

Fig. 3.2: Continuous-to-discrete conversion. (a) A model that consists of multiplying \( x_a(t) \) by a sequence of impulses, followed by a system that converts impulses into samples. (b) An example that illustrates the conversion process.

Since the samples \( x_a(nT_s) \) have a continuous range of possible amplitudes, the second component of the A-D converter is the quantizer, which maps the continuous amplitude into a discrete set of amplitudes. For a \( m \)-bit quantizer, the number of quantization levels is \( 2^m \). The next component is the encoder, which takes the digital signal \( \hat{x}(n) \) and produces a sequence of binary codewords.

The sampling theorem provides a criterion for the minimum sampling rate to avoid aliasing.

\[
\text{Minimum Sampling Rate} = 2f_a
\]

where \( f_a \) is the highest frequency component in the signal. The process of sampling a signal is illustrated in Fig. 3.3.

Fig. 3.3: Sampling of a signal. (a) Continuous-time signal \( x(t) \) is sampled at a rate that is at least twice the highest frequency component. (b) The sampled signal \( x_s(t) \) is obtained by passing \( x(t) \) through a low-pass filter. (c) The output of the low-pass filter is the sampled signal \( x_s(t) \).

In practice, the sampling rate is often higher than the minimum rate to ensure that aliasing does not occur.
The Nyquist frequency is called the highest frequency in the analog signal that can be sampled without aliasing. The Nyquist rate is the minimum sampling frequency required to prevent aliasing.

Theorem 3.1.1: The periodicity of the Fourier transform with a period of 2\(\pi\) is a consequence of the time-scaling property.

Theorem 3.1.2: The Fourier transform of a frequency-scaled version of a signal is the frequency-scaled version of the Fourier transform of the original signal.

Example 3.2.1: Consider a signal that occupies the interval \((a, b)\) in the time domain. Its Fourier transform is as shown in the figure below.

\[
\mathcal{F}\{x(t)\} = \sum_{k=-\infty}^{\infty} X(k) e^{j2\pi nk/T}
\]

where \(X(k)\) are the Fourier coefficients of \(x(t)\) sampled at \(T\) seconds.

**CHAPTER 3. SAMPLING**
In some applications, such as speech coding, the quantizer levels are random variables, \( n \) is a sequence of independent random variables.

Quantization error is expressed as a quantization error model. In the context of quantization, the quantization error is described statistically. It is generally assumed that the quantization error is an additive noise source. Since the quantization error is typically not known, then the noise assumption is made.

The number of levels in a quantizer is generally of the order of one bit binary code word. A 3-bit uniform quantizer is a \( 3 \)-bit uniform quantizer. Figure 3.3 illustrates this process.

Quantizers may have quantization levels that are either uniformly or nonuniformly spaced. How-
\[
\sum_{n=-\infty}^{\infty} (\delta(x - nT) - \delta(x)) = (1/T) \cdot \delta(x)
\]

The frequency domain representation of the impulse train function \((\delta(x - nT))\) is given by

\[
\mathcal{F}\{\delta(x - nT)\} = \int_{-\infty}^{\infty} \delta(x - nT) e^{-j2\pi f x} dx = \sum_{n=-\infty}^{\infty} \delta(f - n/T)
\]

and its Fourier transform is

\[
\mathcal{F}\{\delta(x)\} = \int_{-\infty}^{\infty} \delta(x) e^{-j2\pi f x} dx = \sum_{n=-\infty}^{\infty} \frac{1}{T} \cdot \delta(f - n/T)
\]

The impulse train function is reconstructed from its samples if

\[
t(x) = \sum_{n=-\infty}^{\infty} t(nT) \cdot \delta(x - nT)
\]

then the output of the filter is

\[
\sum_{n=-\infty}^{\infty} (\delta(x - nT) - \delta(x)) = (1/T) \cdot \delta(x)
\]

The frequency domain representation of the impulse train function \((\delta(x - nT))\) is given by

\[
\mathcal{F}\{\delta(x - nT)\} = \int_{-\infty}^{\infty} \delta(x - nT) e^{-j2\pi f x} dx = \sum_{n=-\infty}^{\infty} \delta(f - n/T)
\]

and its Fourier transform is

\[
\mathcal{F}\{\delta(x)\} = \int_{-\infty}^{\infty} \delta(x) e^{-j2\pi f x} dx = \sum_{n=-\infty}^{\infty} \frac{1}{T} \cdot \delta(f - n/T)
\]

The impulse train function is reconstructed from its samples if

\[
t(x) = \sum_{n=-\infty}^{\infty} t(nT) \cdot \delta(x - nT)
\]

then the output of the filter is

\[
\sum_{n=-\infty}^{\infty} (\delta(x - nT) - \delta(x)) = (1/T) \cdot \delta(x)
\]

The frequency domain representation of the impulse train function \((\delta(x - nT))\) is given by

\[
\mathcal{F}\{\delta(x - nT)\} = \int_{-\infty}^{\infty} \delta(x - nT) e^{-j2\pi f x} dx = \sum_{n=-\infty}^{\infty} \delta(f - n/T)
\]

and its Fourier transform is

\[
\mathcal{F}\{\delta(x)\} = \int_{-\infty}^{\infty} \delta(x) e^{-j2\pi f x} dx = \sum_{n=-\infty}^{\infty} \frac{1}{T} \cdot \delta(f - n/T)
\]

The impulse train function is reconstructed from its samples if

\[
t(x) = \sum_{n=-\infty}^{\infty} t(nT) \cdot \delta(x - nT)
\]

then the output of the filter is

\[
\sum_{n=-\infty}^{\infty} (\delta(x - nT) - \delta(x)) = (1/T) \cdot \delta(x)
\]
If the discrete-time system is linear and shift-invariant with a frequency response

\[
H(e^{j\omega}) = \left| H(e^{j\omega}) \right| \angle H(e^{j\omega})
\]

then the cascade of the periodic spectrum of the discrete-time signal \(H(e^{j\omega})\) approximates a lowpass filter with a gain of \(e^{-j\omega_T}\) over the passband. Figure 3.8 shows the magnitude of the frequency response of the zero-order hold, and the magnitude of the frequency response of the ideal reconstruction compensation filter. Note that the cascade of the reconstruction filter in the D-C converter. Since the input signal \(x(t)\) has been converted to impulses, the zero-order hold produces the output signal \(y(t)\) as shown in Fig. 3.7. With a zero-order hold, it is common to post-process the output with a reconstruction filter. The impulse response of a zero-order hold is given by

\[
(\frac{1}{T} \sum_{n=-\infty}^{\infty} x(nT)\delta(t-nT)) = \frac{1}{T} \sum_{n=-\infty}^{\infty} x(nT)\left(\frac{t-nT}{T}\right)
\]

where \(x(nT)\) is the sequence of samples and \(\delta(t)\) is the Dirac delta function. The frequency response of a zero-order hold is given by

\[
H_{\text{zero-order hold}}(\omega) = \frac{1}{\pi} \left[ \frac{\sin(\omega T/2)}{\omega T/2} \right] = \frac{1}{\pi} \left[ \frac{\sin(\pi \omega T)}{\pi \omega} \right]
\]

which approximates a lowpass filter with a gain of \(e^{-j\omega_T}\) over the passband. Figure 3.9: Processing an analog signal using a discrete-time system.

After a sequence of samples \(x(nT)\) has been converted to impulses, the zero-order hold produces the output signal \(y(t)\). The impulse response of a zero-order hold is given by

\[
(\frac{1}{T} \sum_{n=-\infty}^{\infty} x(nT)\delta(t-nT)) = \frac{1}{T} \sum_{n=-\infty}^{\infty} x(nT)\left(\frac{t-nT}{T}\right)
\]

which approximates a lowpass filter with a gain of \(e^{-j\omega_T}\) over the passband. Figure 3.9: Processing an analog signal using a discrete-time system.
where the summation index $i$ is the expression for the DTFT of $x[n]$, and

\[
\sum_{i=0}^{\infty} = \sum_{i=-\infty}^{\infty}
\]

Therefore, the overall system behaves as a linear time-invariant continuous-time system with an

d-c converter produces the continuous-time signal

\[
\frac{\sin \pi \omega}{\pi \omega}x(T \omega)
\]

Finally, the D-C converter produces the continuous-time signal

\[
\frac{\sin \pi \omega}{\pi \omega}x(T \omega)
\]

Therefore, the overall system behaves as a linear time-invariant continuous-time system with an

d-c converter produces the continuous-time signal

\[
\frac{\sin \pi \omega}{\pi \omega}x(T \omega)
\]

Finally, the D-C converter produces the continuous-time signal

\[
\frac{\sin \pi \omega}{\pi \omega}x(T \omega)
\]

Therefore, the overall system behaves as a linear time-invariant continuous-time system with an

d-c converter produces the continuous-time signal

\[
\frac{\sin \pi \omega}{\pi \omega}x(T \omega)
\]

Finally, the D-C converter produces the continuous-time signal

\[
\frac{\sin \pi \omega}{\pi \omega}x(T \omega)
\]

Therefore, the overall system behaves as a linear time-invariant continuous-time system with an

d-c converter produces the continuous-time signal

\[
\frac{\sin \pi \omega}{\pi \omega}x(T \omega)
\]

Finally, the D-C converter produces the continuous-time signal

\[
\frac{\sin \pi \omega}{\pi \omega}x(T \omega)
\]

Therefore, the overall system behaves as a linear time-invariant continuous-time system with an

d-c converter produces the continuous-time signal

\[
\frac{\sin \pi \omega}{\pi \omega}x(T \omega)
\]

Finally, the D-C converter produces the continuous-time signal

\[
\frac{\sin \pi \omega}{\pi \omega}x(T \omega)
\]
3.5. Sample Rate Increase by an Integer Factor

If we would like to increase the sampling rate by an integer factor $L$, we can use a cascade of an up-sampler with a lowpass filter shown in Fig. 3.12(b) to achieve this. The samples of $x(nT)$ are simply scaled in frequency after up-sampling. After up-sampling, it is necessary to remove the frequency components that are integer multiples of $\frac{2\pi}{M}$ from $X(j\omega)$. These components are not part of $X(j\omega)$ and should be filtered out.

Suppose that we would like to increase the sampling rate by an integer factor $L$. This can be achieved by using an up-sampler with a gain of $\frac{1}{L}$, followed by a lowpass filter. The output of the up-sampler is given by $\hat{x}(nT) = x(nT/L)$.

Therefore, in order to prevent aliasing, the frequency components at integer multiples of $\frac{2\pi}{L}$ in $X(j\omega)$ should be filtered prior to downsampling with a lowpass filter.

The interpolation process in the frequency domain is illustrated in Fig. 3.14.
3.5 SAMPLE RATE CONVERSION

Figure 3.13: (a) The output of the up-sampler. (b) The interpolation between the samples $	ilde{x}_i(n)$ that is performed by the lowpass filter. The sample rate converter may be simplified as illustrated in Fig. 3.15(b).

Example 3.5.1
Suppose that a signal $x_a(t)$ has been sampled with a sampling frequency of 8 kHz, and that we would like to derive the discrete-time signal that would have been obtained if $x_a(t)$ had been sampled with a sampling frequency of 10 kHz. Thus, we would like to change the sampling rate by a factor of $L = 10/8 = 5/4$.

This may be accomplished by up-sampling $x(n)$ by a factor of five, filtering the up-sampled signal with a lowpass filter that has a cutoff frequency $\omega_c = \pi/5$ and a gain of five, and then down-sampling the filtered signal by a factor of four.

Figure 3.14: Frequency domain illustration of the process of interpolation. (a) The continuous-time signal. (b) The DTFT of the sampled signal $x(n) = x_a(nT_s)$. (c) The DTFT of the up-sampler output. (d) The ideal lowpass filter to perform the interpolation. (e) The DTFT of the interpolated signal.
3.5. SAMPLE RATE CONVERSION

- Lowpass Filter
  - Cutoff = $\pi/L$
  - Gain = $L$

- Lowpass Filter
  - Gain = 1
  - Cutoff = $\pi/M$

- $x(n_0) \times (n)$
  - $L \times i(n)$  \[ \sim \]
  - $x(n) \times d(n)$

Figure 3.15: (a) Cascade of an interpolator and a decimator for changing the sampling rate by a rational factor $L/M$. (b) A simplified structure that results when the two lowpass filters are combined.