

Sampling, Quantization, and Digital Representation of Signals

Ayfer Ozgur and Siddharth Chandak

A continuous-time signal cannot be stored or processed directly by a digital system. To represent such a signal digitally, we must sample it in time, quantize its amplitude, and finally represent the resulting symbols using bits. These notes study this pipeline in stages. We first focus on sampling, which addresses the conditions under which a continuous-time signal can be fully represented by its samples. We then discuss aliasing; specifically, what goes wrong when these sampling assumptions are violated. Finally, we introduce quantization, which maps continuous amplitude values to a finite set of levels, introducing an unavoidable loss of precision.

1 Sampling

In many applications, signals of interest are defined in continuous time. For example, an audio signal corresponds to air pressure as a function of time, and an electrical signal corresponds to voltage varying continuously over time. However, digital systems cannot store or process continuous-time signals directly. Sampling is the first step in converting a continuous-time signal into a form that can be processed digitally.

1.1 What is sampling?

Definition 1 (Sampling). *Given a continuous-time signal $x(t)$ and a sampling period $T_s > 0$, sampling consists of observing the signal at uniform time instants:*

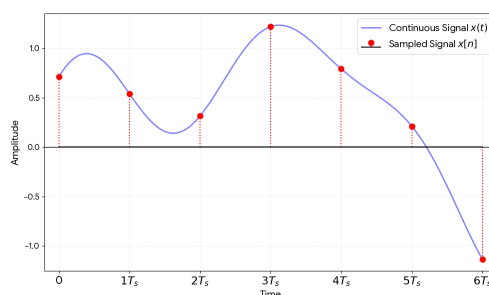
$$\{x(0), x(T_s), x(2T_s), x(3T_s), \dots\}.$$

The resulting sequence is called a discrete-time signal.

The quantity T_s is called the *sampling period* or *sampling interval*. Equivalently, we define the *sampling frequency*

$$f_s = \frac{1}{T_s},$$

which represents the number of samples taken per second. Importantly, sampling does *not* restrict the range of values the signal can take: the samples $x(kT_s)$ are still real numbers.



Sampling alone does not guarantee that the original signal can be recovered. Since we only observe the signal at isolated instants, it could, in principle, behave arbitrarily between those points. This raises two fundamental questions:

- Is there a specific class of signals for which the samples uniquely determine the original waveform?
- What condition must the sampling frequency f_s satisfy for this to hold?

At first glance, the idea of perfect reconstruction feels non-intuitive, even unreasonable. If we only have a handful of dots, how can we possibly know what happened in the gaps? Since infinitely many curves could pass through the same set of samples, it seems impossible to know the right one. The surprising fact is that, under the right conditions, this is not just possible, it is mathematically guaranteed. The key lies in the frequency-domain representation of signals. By restricting our attention to “smooth” signals with limited frequency content, we will see that a finite set of samples can uniquely “lock” the signal into place everywhere in time.

1.2 Bandlimited signals

Suppose $x(t)$ is a periodic signal with period T . Recall that it admits a Fourier series representation:

$$x(t) = b_0 + \sum_{j=1}^{\infty} \left(a_j \sin\left(2\pi \frac{j}{T}t\right) + b_j \cos\left(2\pi \frac{j}{T}t\right) \right).$$

Each term corresponds to a sinusoidal component with frequency j/T . It is often convenient to combine the sine and cosine coefficients at a given frequency into a single amplitude

$$A_j = \sqrt{a_j^2 + b_j^2}.$$

The collection $\{A_j\}$ describes how the signal’s energy is distributed across frequencies.

Definition 2 (Bandlimited Signal). *A signal is called bandlimited if its Fourier series representation contains only finitely many nonzero frequency components. Equivalently, $A_j \neq 0$ for only finitely many values of j .*

Bandlimited signals cannot oscillate arbitrarily fast. Rapid variations in time require high-frequency components; restricting the signal to finitely many frequencies enforces smoothness.

Example 1. *Consider the signal $x(t) = 5 + 2 \sin\left(2\pi \frac{10}{T}t\right) - 9 \sin\left(2\pi \frac{35}{T}t\right)$. This signal contains only three frequency components and is therefore bandlimited.*

Example 2. *The signal*

$$x(t) = \sum_{j=1}^{\infty} \frac{1}{j} \sin\left(2\pi \frac{j}{T}t\right)$$

contains infinitely many frequency components and is not bandlimited.

Example 3 (Square wave and triangle wave). Recall the square and triangle waves studied in the previous miniPset. These are not bandlimited. A square wave has jump discontinuities. Representing a jump discontinuity requires infinitely many frequency components in its Fourier series. As a result, a square wave is not bandlimited. A triangle wave is continuous but has sharp corners (points where the derivative is discontinuous). These corners also require infinitely many frequency components to represent exactly. Hence, a triangle wave is not bandlimited.

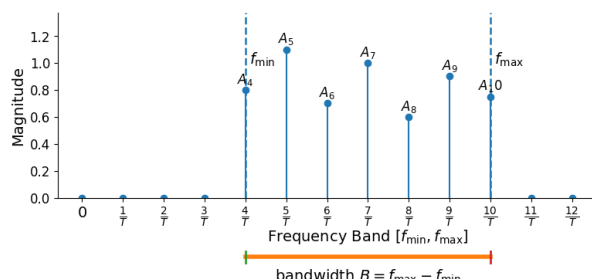
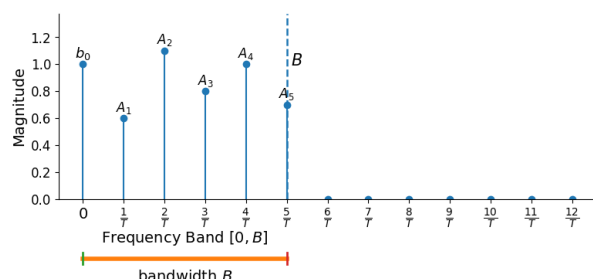
To make this precise, we introduce terminology that describes the range of frequencies present in a signal.

Definition 3 (Frequency band and bandwidth). The frequency band of a signal is the smallest frequency interval that contains all frequencies present in the signal. The bandwidth is the width of this interval.

We now define two important subclasses of bandlimited signals.

Definition 4 (Baseband Signal). A signal is called baseband with bandwidth B if all its frequency components lie in the interval $[0, B]$.

Definition 5 (Passband Signal). A signal is called passband if all its frequency components lie in an interval $[f_{\min}, f_{\max}]$.



Example 4. Consider the signal $x(t) = 15 \cos(2\pi 25t) + \sin(2\pi 100t)$. Its frequency components are 25 Hz and 100 Hz. The frequency band is $[25, 100]$ Hz, so the signal is passband with bandwidth 75 Hz.

Example 5. Consider $x(t) = 12 - \cos(2\pi t) + 18 \sin(2\pi 14t)$. The frequency components are 0, 1, and 14 Hz. The frequency band is $[0, 14]$ Hz, so the signal is baseband with bandwidth 14 Hz.

Note that whether a signal is baseband or passband depends only on the frequencies present in the signal, not on their amplitudes. Changing the coefficients in front of the sine or cosine terms does not change the frequency components, and hence does not change the frequency band or the bandwidth. For example, the signal $x(t) = 6 + \cos(2\pi t) - 9 \sin(2\pi 14t)$ has the same frequency components as the previous example, and therefore the same frequency band and bandwidth, even though the amplitudes are different.

1.3 Representing bandlimited signals over a finite duration

We now show that bandlimited signals have only finitely many degrees of freedom over a finite time interval. Suppose $x(t)$ is a baseband signal with bandwidth B and is modeled as a periodic signal with period T . Since all frequency components satisfy

$$0 \leq \frac{j}{T} \leq B,$$

we must have

$$0 \leq j \leq BT.$$

Thus, the Fourier series truncates:

$$x(t) = b_0 + \sum_{j=1}^{BT} \left(a_j \sin\left(2\pi \frac{j}{T}t\right) + b_j \cos\left(2\pi \frac{j}{T}t\right) \right).$$

The signal is completely determined by the coefficients

$$\{b_0, a_1, b_1, \dots, a_{BT}, b_{BT}\},$$

which is a total of $2BT + 1$ real numbers.

Key Observation: A baseband signal of bandwidth B and duration T has only $2BT + 1$ degrees of freedom, meaning it can be fully described by $2BT + 1$ real numbers.

1.4 Sampling as a system of linear equations

Each sample of the signal provides one linear equation in the unknown Fourier coefficients. For example, let $T = 1$ and $B = 2$. Then

$$x(t) = b_0 + a_1 \sin(2\pi t) + b_1 \cos(2\pi t) + a_2 \sin(4\pi t) + b_2 \cos(4\pi t).$$

Sampling at $t = 0$ gives

$$x(0) = b_0 + b_1 + b_2.$$

Sampling at $t = T_s$ gives

$$x(T_s) = b_0 + a_1 \sin(2\pi T_s) + b_1 \cos(2\pi T_s) + a_2 \sin(4\pi T_s) + b_2 \cos(4\pi T_s).$$

Each additional sample produces another linear equation. To uniquely determine all $2BT + 1$ unknown coefficients, we need at least $2BT + 1$ independent equations. If we sample a signal of duration T every T_s seconds, the total number of samples is

$$\frac{T}{T_s} + 1.$$

To recover all $2BT + 1$ coefficients, we require

$$\frac{T}{T_s} + 1 > 2BT + 1,$$

which simplifies to

$$T_s < \frac{1}{2B} \iff f_s > 2B.$$

We avoid the case of equality since it corresponds to a critical situation where the system of equations becomes just-determined and may fail to have a unique solution depending on the sampling instants. Requiring a strict inequality ensures that we have strictly more equations than unknowns, and hence guarantees unique recovery of the signal.

Fact 1. *Let $x(t)$ be a baseband signal with bandwidth B . If the sampling frequency satisfies $f_s > 2B$, then the samples $\{x(mT_s)\}$ uniquely determine the signal.*

This fact states that, under the condition $f_s > 2B$, the samples contain all the information about the original signal. In other words, among all bandlimited signals with bandwidth B , there is exactly one signal that is consistent with the given set of samples. Sampling fast enough therefore does not lose information for this class of signals. The importance of this fact lies in the observation that many natural signals of interest are either baseband to begin with or can be made baseband by appropriate preprocessing. For example, the range of frequencies audible to humans is approximately limited to 20 kHz. Audio signals are therefore modeled as baseband signals with bandwidth $B \approx 20$ kHz. Sampling such signals at a rate exceeding $2B$, for instance at 44.1 kHz, guarantees that no information is lost in the sampling process.

This result is surprising for several reasons. A continuous-time signal has infinitely many values, whereas sampling keeps only values at discrete time instants. Between samples, the signal could, in principle, behave arbitrarily. A priori, it is not obvious that a finite sampling rate could capture all the information in a signal defined for all real times. What makes this possible is the structure imposed by bandlimitedness. Bandlimited signals are inherently smooth in time and cannot exhibit arbitrarily rapid changes between samples. Sampling at a rate greater than $2B$ captures all the degrees of freedom of such signals, which is why the samples uniquely determine the signal everywhere in time.

2 Undersampling and Aliasing

We have established that if we sample a bandlimited signal fast enough ($f_s > 2B$), the samples contain all the information necessary to uniquely identify the original signal. However, in practice, we may encounter situations where this condition is violated. When the sampling frequency is too low relative to the signal’s bandwidth—a condition known as *undersampling*—the samples no longer uniquely describe the source.

In this regime, high-frequency components “impersonate” lower-frequency ones. Because the sampling rate is insufficient to capture the rapid oscillations of the original signal, the resulting data points appear consistent with a much slower, “alias” signal. This phenomenon is called *aliasing*.

2.1 The Stroboscopic Effect

A classic real-world example of aliasing is the stroboscopic effect, which occurs when a continuous motion is observed through a series of short bursts of light or a discrete shutter. Imagine a fan rotating at 60 revolutions per second (60 Hz). If we illuminate the fan with a strobe light flashing at exactly 60 Hz, each flash will catch the blades in the exact same position. To our eyes, the fan will appear to be standing perfectly still.

In this case, the “true” signal (60 Hz) is being undersampled. Because we only see the blades at intervals matching their period, our brain “reconstructs” a 0 Hz signal (stationary). If we slightly decrease the strobe frequency to 59 Hz, the blades will appear to rotate very slowly forward; if we increase it to 61 Hz, they will appear to rotate slowly backward. In all these cases, the high-frequency physical motion is aliased into a low-frequency visual perception.

A similar phenomenon, often called the wagon-wheel effect or cart-wheel effect, is frequently seen in movies or videos of moving vehicles. Video is a sampled representation of reality, typically recorded at 24 or 30 frames per second (f_s). Consider a spoked wheel spinning rapidly. Between two consecutive frames, a spoke might rotate almost a full 360 degrees, landing just behind its original position. When the video is played back, our visual system connects the spokes in the closest possible path, making the wheel appear to be spinning backward at a much slower rate. An interactive visualization of the stroboscopic (wagon-wheel) effect can be found [here](#).

2.2 Information Loss and Ambiguity

The fundamental problem with aliasing is ambiguity. Once a signal is undersampled, the resulting samples could have come from infinitely many different continuous-time signals. A high-frequency sine wave sampled too slowly becomes indistinguishable from a lower-frequency sine wave. See Figure 1.

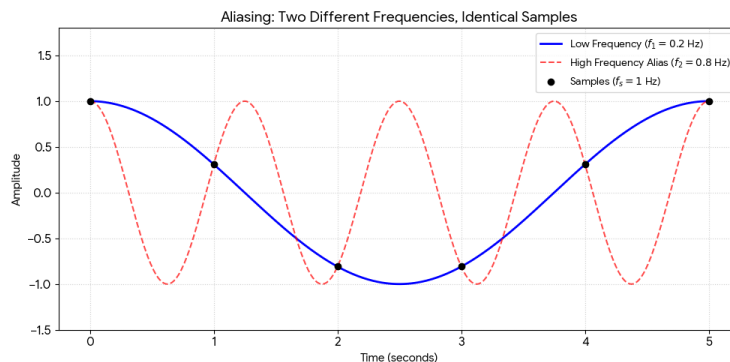


Figure 1: Aliasing in action: A high-frequency signal (red dashed) becomes indistinguishable from a low-frequency signal (blue) when the sampling rate is insufficient.

Unlike other parts of the digital pipeline (such as quantization, where the resulting distortion is a known quantity controlled by the engineer’s choice of resolution, or source coding, which is mathematically reversible) aliasing fundamentally alters the frequency identity of the signal. It creates a mathematical ‘impostor’ that is indistinguishable from a valid low-frequency source, making the error impossible to detect or undo once the samples are recorded. Once the samples are recorded, there is no mathematical way to “un-alias” the data and determine if the original signal was the slow one we see or a much faster one we missed.

This is why, in practical engineering, we often use an “anti-aliasing filter” before sampling. This filter removes all frequency components above $f_s/2$, ensuring that the sampling condition is strictly met and that no high-frequency “imposters” can corrupt our digital representation. This is why, in audio engineering, if you don’t filter out frequencies above 22 kHz before sampling at 44 kHz, those high-pitched noises “fold back” into the audible range as low-frequency distortion that wasn’t in

the original room.

3 Quantization and Dequantization

Sampling converts a continuous-time signal into a discrete-time sequence, but the sampled values $x(kT_s)$ are still real numbers. Digital systems cannot store real numbers with infinite precision. To represent these values using a finite number of bits, we must further restrict the set of values they are allowed to take. This process is known as quantization.

Definition 6 (Quantization). *Quantization is the process of mapping a real-valued number to one of finitely many discrete levels.*

Quantization operates on the amplitude of the signal, not on time. While sampling determines *when* the signal is observed, quantization determines *how precisely* each observed value is represented.

Uniform quantization. The simplest and most commonly used form of quantization is *uniform quantization*. Suppose the signal values are known to lie in an interval $[a, b]$. In uniform quantization, this interval is divided into N equal-width bins, each of width $\Delta = (b - a)/N$.

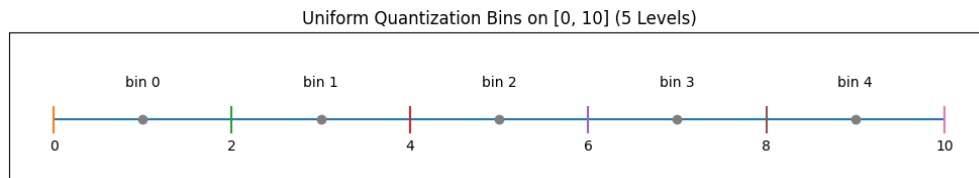
Quantization index representation. After quantization, each sample is represented by a discrete symbol (the bin index). These indices can then be encoded using bits. If there are N quantization levels, then each index can be represented using $\log_2 N$ bits. This makes explicit the tradeoff introduced by quantization: increasing N improves precision but increases the number of bits required per sample, while decreasing N reduces the number of bits but increases distortion.

3.1 Dequantization.

Quantized values are not directly useful for reconstruction, since they only identify bins, not actual signal values. Dequantization assigns each quantization index a representative real value.

Definition 7 (Dequantization). *Dequantization maps a quantized symbol back to a representative real value, typically chosen as the midpoint of the corresponding quantization bin.*

For uniform quantization, if a sample is mapped to a bin spanning $[a + i\Delta, a + (i + 1)\Delta)$, a common choice of reconstruction value is $a + (i + \frac{1}{2})\Delta$.



Example 6 (Uniform quantization with $N = 5$ bins). *Suppose a real-valued sample X lies in the interval $[0, 10]$ and we apply uniform quantization with $N = 5$ bins. The bin width is $\Delta = (10 - 0)/5 = 2$. The five bins are $[0, 2)$, $[2, 4)$, $[4, 6)$, $[6, 8)$, and $[8, 10]$. Consider the sample value $X = 6.3$. Since $6.3 \in [6, 8)$, the quantizer outputs the bin index 3. Dequantization maps this index to the midpoint of the bin, so the reconstructed value is $\hat{X} = 7$.*

Quantization error. The difference between the original sample value and its dequantized value is called the *quantization error*. For uniform quantization, this error is bounded in magnitude by $\Delta/2$. Unlike sampling (under the Nyquist condition), quantization is inherently lossy. Once a value has been quantized, the exact original value cannot be recovered. Reducing the number of quantization levels further increases the loss.

4 From Continuous-Time Signals to Bits and Back

We now summarize the complete digital signal representation pipeline of a continuous-time signal.

Forward direction (digitization).

1. **Sampling:** Observe a continuous-time signal $x(t)$ at discrete-time instants $\{x(kT_s)\}$.
2. **Quantization:** Map each real-valued sample to one of finitely many discrete levels.
3. **Coding:** Represent the resulting discrete symbols using bits, exploiting statistical structure when possible (e.g., Huffman coding).

Backward direction (reconstruction).

1. **Decoding:** Recover discrete symbols from the bitstream.
2. **Dequantization:** Map symbols back to representative real values.
3. **Reconstruction:** Utilize the dequantized samples to produce a continuous-time signal consistent with the data.

Sampling determines whether the discrete-time sequence preserves the identity of the original signal. As long as the sampling frequency satisfies $f_s > 2B$, the information contained in the samples is sufficient to uniquely identify the original continuous-time signal. In this regime, sampling is essentially *lossless*. However, if $f_s < 2B$, the signal's identity is corrupted by aliasing, creating an irreversible ambiguity.

Quantization plays a fundamentally different role: it limits the precision with which each sample value is represented. Unlike sampling (under the Nyquist condition), quantization is inherently *lossy*. It maps a continuum of real-valued amplitudes to a finite set of levels, introducing an unavoidable rounding error. Once a value has been quantized, the exact original amplitude cannot be recovered, regardless of the reconstruction method used.

The final step, representing symbols as bits, is purely a matter of efficient representation. Source coding is *lossless*, meaning the quantized symbols can be recovered exactly from the bitstream.

Thus, in a well-designed digital pipeline where the Nyquist condition is met, the only irreversible loss of information comes from quantization. If the Nyquist condition is violated, information is lost at the very first step through aliasing, which fundamentally alters the signal before it even reaches the quantizer.