Sampling and quantization

Digital Signal Processing with a focus on audio signals

Instructor: *Fabio Antonacci*

Feb. 16, 2026

Discrete-time signals

- A discrete-time signal is represented as a sequence of numbers:

$$\{x(n)\,, -\infty < n < \infty\}$$

  - n is an integer;
  - $x(n)$ is the nth **sample** of the sequence.

Discrete-time signals

- A discrete-time signal is represented as a sequence of numbers:

$$\{x(n)\,, -\infty < n < \infty\}$$

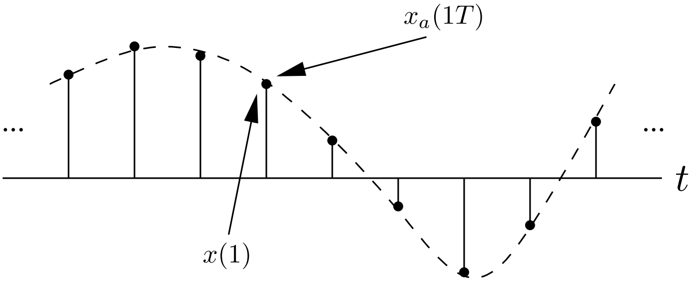
  - n is an integer;
  - $x(n)$ is the nth **sample** of the sequence.
- Discrete-time signals are often obtained by sampling continuous-time signals. In this case, the nth sample is obtained as

$$x(n) = x_a(nT)$$

  - x_a is a continuous-time signal;
  - T is the **sampling period**.

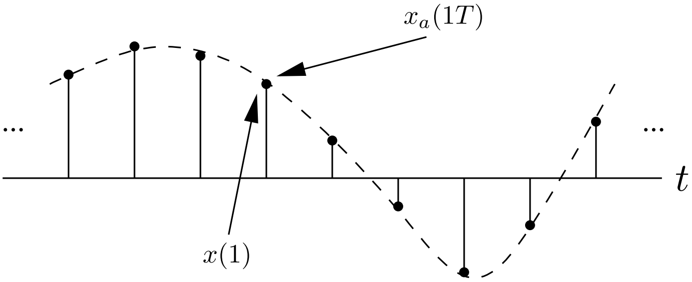
Discrete-time signals: sampling period

- The sampling operation can be illustrated as follows:



Discrete-time signals: sampling period

- The sampling operation can be illustrated as follows:

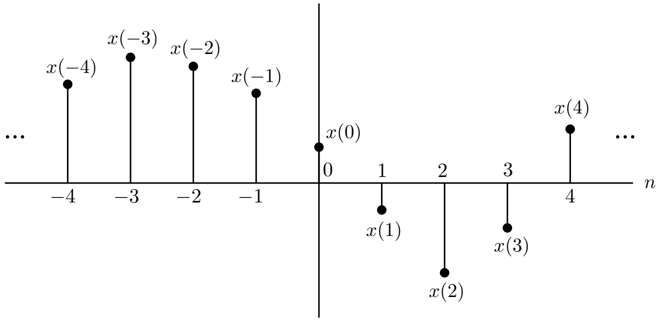


- Example: reading the temperature (continuous-time signal) every hour:
  $T = 3600\,\mathrm{s}$

$$x(1) = 20^\circ\mathrm{C}\ ,\ x(2) = 18^\circ\mathrm{C}\ ,\ x(3) = 21^\circ\mathrm{C}\ \ldots$$

Discrete-time signals: representation

- Discrete-time signals are often depicted graphically using stem plots:



- The value $x(n)$ is undefined for non-integer values of n: the discrete sequence does not contain any information between adjacent samples!

Discrete-time signals: basic operations

Discrete sequences can be manipulated in several ways. The most basic operations are the following:

Discrete-time signals: basic operations

Discrete sequences can be manipulated in several ways. The most basic operations are the following:

- Scaling:

$$y(n) = Ax(n) \; , \quad A \in \mathbb{R}$$

Discrete-time signals: basic operations

Discrete sequences can be manipulated in several ways. The most basic operations are the following:

- Scaling:

$$y(n) = Ax(n)\,, \quad A \in \mathbb{R}$$

- Sum of two sequences:

$$y(n) = x_1(n) + x_2(n)$$

Discrete-time signals: basic operations

Discrete sequences can be manipulated in several ways. The most basic operations are the following:

- Scaling:

$$y(n) = Ax(n)\ , \quad A \in \mathbb{R}$$

- Sum of two sequences:

$$y(n) = x_1(n) + x_2(n)$$

- Product of two sequences:

$$y(n) = x_1(n) \cdot x_2(n)$$

Discrete-time signals: basic operations

Discrete sequences can be manipulated in several ways. The most basic operations are the following:

- Scaling:

$$y(n) = Ax(n) \; , \quad A \in \mathbb{R}$$

- Sum of two sequences:

$$y(n) = x_1(n) + x_2(n)$$

- Product of two sequences:

$$y(n) = x_1(n) \cdot x_2(n)$$

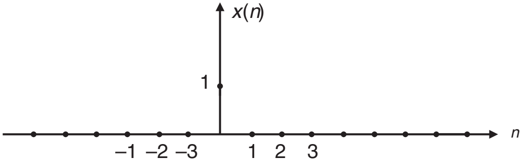
- Time-shifting:

$$y(n) = x(n - n_0)$$

Discrete-time impulse

Unit impulse

$$\delta(n) = \begin{cases} 0 & n \neq 0 \\ 1 & n = 0 \end{cases}$$

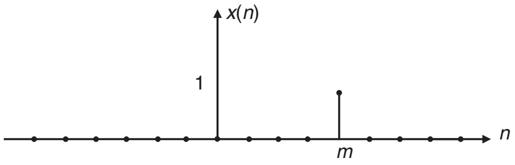


- The unit pulse (Kronecker delta function) plays the same role for discrete-time signals as the Dirac delta function does for continuous-time signals
- No mathematical complications in its definition!

Other important discrete sequences

Delayed unit impulse

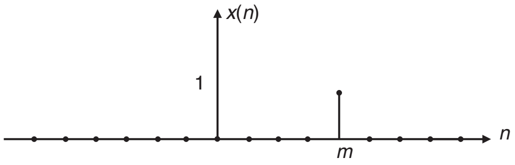
$$\delta(n-m) = \begin{cases} 0 & n \neq m \\ 1 & n = m \end{cases}$$



Other important discrete sequences

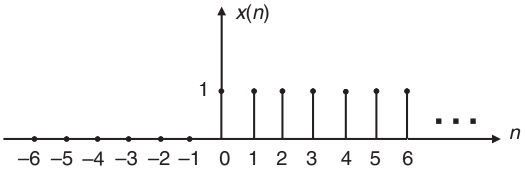
Delayed unit impulse

$$\delta(n-m) = \begin{cases} 0 & n \neq m \\ 1 & n = m \end{cases}$$



Unit step

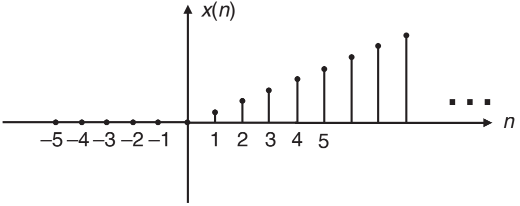
$$u(n) = \begin{cases} 0 & n < 0 \\ 1 & n \geq 0 \end{cases}$$



Other important discrete sequences

Unit ramp

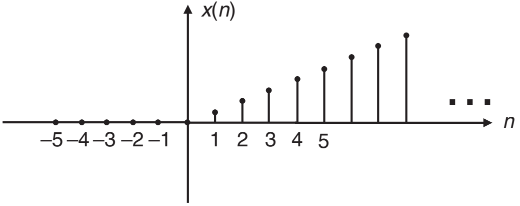
$$r(n) = \begin{cases} n & n \geq 0 \\ 0 & n < 0 \end{cases}$$



Other important discrete sequences

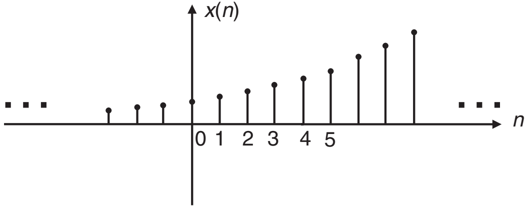
Unit ramp

$$r(n) = \begin{cases} n & n \geq 0 \\ 0 & n < 0 \end{cases}$$



Real exponential function

$$x(n) = e^{an}$$



Periodic sequences

- A sequence is periodic iff there exists an integer $N \neq 0$ such that

$$x(n) = x(n+N) \quad \forall n \; ,$$

  where N is the **period** (in samples) of the sequence.
- important periodic sequences are **sinusoids** and **complex exponentials**;
- however, we will see that some criteria must be satisfied for guaranteeing their periodicity!

Sinusoidal sequences

- A discrete sinusoid is described by the sequence

$$x(n) = A\cos(\omega n + \phi)$$

  - $A \in \mathbb{R}$ is the **amplitude**, $\phi \in \mathbb{R}$ is the **phase angle** (radians)
  - $\omega \in \mathbb{R}$ is the **angular frequency** (radians/sample)
  - $f_0 \triangleq \omega/2\pi$ is the **frequency**, (cycles/sample)

- Angular frequency is usually limited in the range $[0, 2\pi)$, as a digital sinusoid of frequency ω is not distinguishable from one at frequency $\omega + 2k\pi$, k integer:

$$\cos[(\omega + 2k\pi)n + \phi] = \cos[\omega n + \phi + 2k\pi n] = \cos[\omega n + \phi] \;\; \forall k \in \mathcal{Z}$$

- Often, complex exponentials are used in place of real sinusoids:

$$x(n) = Ae^{j(\omega n + \phi)} = A\cos(\omega n + \phi) + jA\sin(\omega n + \phi)$$

Discrete vs continuous sinusoids

- In the continuous-time case, sinusoidal and complex exponential sequences are always periodic
- In the discrete-time domain, we must pay attention! Let's try to find the period N of the signal, so that

$$\cos(\omega n) = \cos\left[\omega(n+N)\right]$$

  - N must be an integer
  - therefore we must have $\omega N = k2\pi$, where k is some integer
  - noting that $\omega = 2\pi f_0$, this corresponds to requiring

$$f_0 = \frac{k}{N} \ ,$$

  i.e. that f_0 is a rational number

- It turns that a discrete sinusoid is periodic if and only if f_0 is rational (or, analogously, iff $N \cdot f_0$ is an integer)

Discrete vs continuous sinusoids (cont')

- The above condition implies that

$$\cos(\omega n + \phi) = \cos(\omega(n + kN) + \phi) \ , \quad \forall k \in \mathbb{Z}$$

and

$$Ae^{j(\omega n+\phi)} = Ae^{j[\omega(n+kN)+\phi]} \ , \quad \forall k \in \mathbb{Z}$$

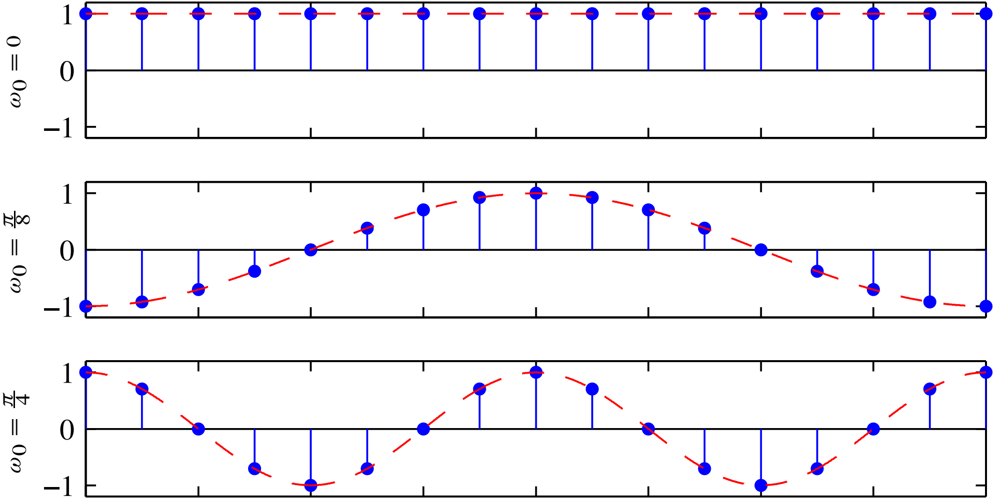
- We reach the conclusion that there are only N distinguishable frequencies for which the corresponding sinusoidal sequences are periodic, with period N:

$$\omega_k = \frac{2\pi k}{N} \ , \quad k = 0, 1, \dots N - 1$$

- Additionally, for discrete-time sequences the interpretation of high and low frequencies is different from that of countinous-time signals:
  - The sinusoid $x(n) = A\cos(\omega n + \phi)$ oscillates more rapidly as ω increases in the range $(0, \pi)$;
  - Oscillations become slower as ω increases in the range $(\pi, 2\pi)$.

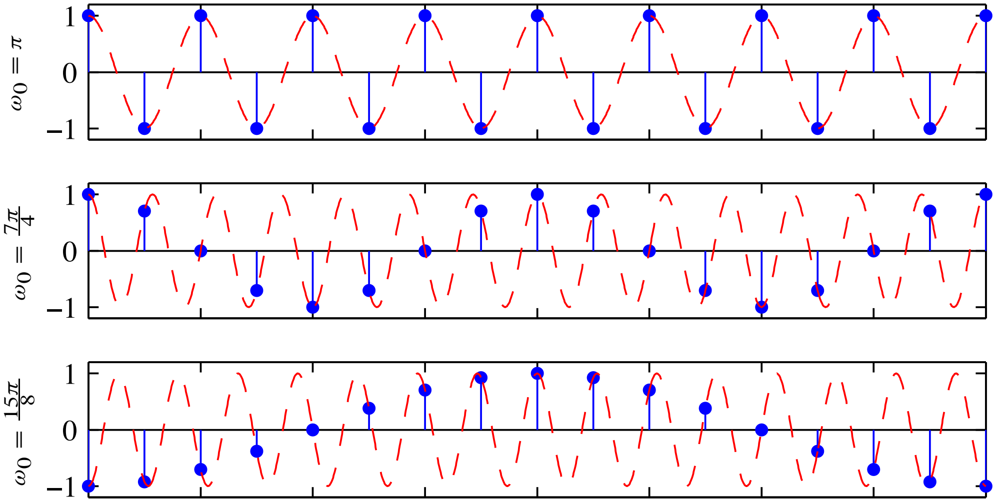
Discrete vs continuous sinusoids

Example: sinusoids with period $N = 16$



Discrete vs continuous sinusoids

Example: sinusoids with period $N = 16$



The discrete-time exponential function

- An important class of functions is that of discrete exponentials

$$x(n) = \gamma^n\ , \quad \gamma \in \mathcal{C}$$

  - We'll see they play the same role of $e^{\lambda t}$ in the continuous time domain!
- To analyze its behavior, let's rewrite

$$\gamma = |\gamma| e^{\angle\gamma}$$

  so that

$$x(n) = |\gamma|^n e^{nj\angle\gamma}$$

- It follows that

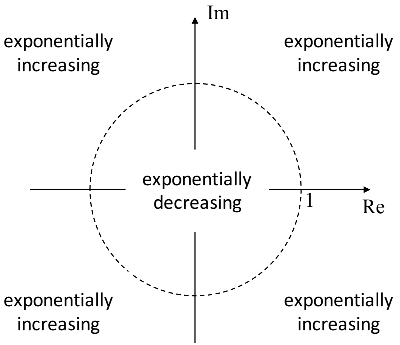
$$\lim_{n\to\infty} \gamma^n = \begin{cases} 0 & |\gamma| < 1 \\ \infty & |\gamma| > 1 \\ \nexists & |\gamma| = 1 \end{cases}$$

The discrete-time exponential function (cont')

- It means that the causal sequence

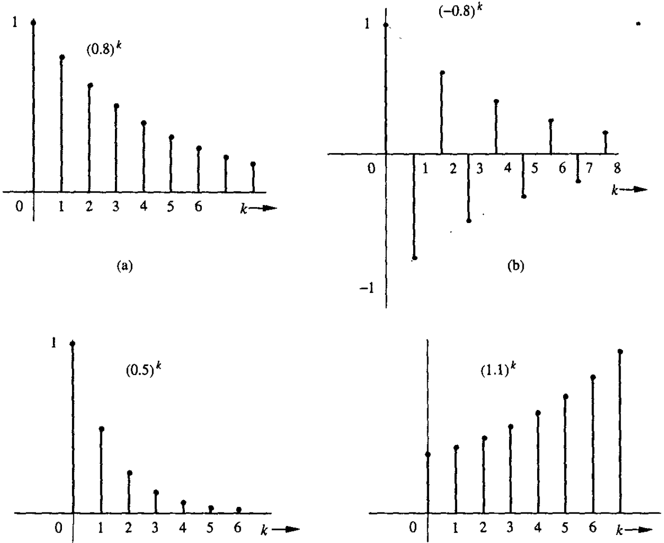
$$x(n) = \gamma^n u(n)$$

  - converges if γ is located within the unit circle in the complex plane
  - diverges if γ is outside the unit circle
  - oscillates if γ lies on the unit circle



The discrete-time exponential function (cont')

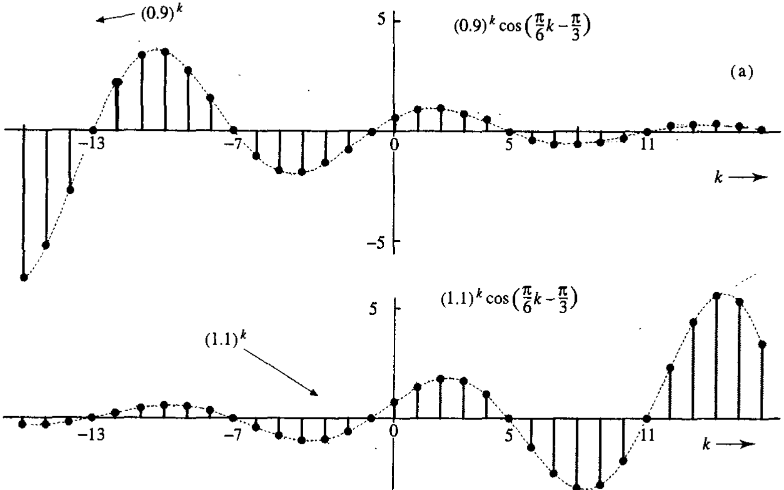
Examples:



Exponentially varying discrete sinusoids

- Sinusoid modulated by an exponential sequence:

$$x(n) = \gamma^n \cos(\omega n + \phi)$$



Size of discrete-time signals

- Energy:

$$E_x = \sum_{n=-\infty}^{\infty} |x(n)|^2$$

  - if E_x is finite, $x(n)$ is called an energy signal

- Power

$$P_x = \lim_{N\to\infty} \frac{1}{2N+1} \sum_{n=-N}^{N} |x(n)|^2$$

  - If P_x is finite and non-zero, $x(n)$ is called a power signal
  - For periodic signals, it is sufficient to evaluate the expression on a single period and leave out the limit from the definition

Quantization of discrete-time signals

Setup

We start from a discrete-time, continuous-amplitude signal:

$$x[n] \in \mathbb{R}, \quad n \in \mathbb{Z}$$

Quantization produces a discrete-amplitude signal:

$$x_q[n] \in \mathcal{Q} \quad \text{(finite set of levels)}$$

Quantization error

$$e[n] = x[n] - x_q[n]$$

Goal: represent $x[n]$ with a limited number of bits while keeping $e[n]$ small.

Uniform quantization (mid-tread) for audio

Assume amplitudes are normalized/clipped to:

$$x[n] \in [-A, A]$$

With B bits, the number of quantization levels is:

$$L = 2^B$$

Uniform step size:

$$\Delta = \frac{2A}{L-1} = \frac{2A}{2^B - 1}$$

Quantizer levels

A common choice is the set of equally spaced levels:

$$\mathcal{Q} = \{-A,\ -A + \Delta,\ \dots,\ A - \Delta,\ A\}$$

Quantization rule (rounding to nearest level)

Define the index of the closest level:

$$k[n] = \mathrm{round}\left(\frac{x[n] + A}{\Delta}\right)$$

Then the quantized signal is:

$$x_q[n] = -A + k[n]\Delta$$

With saturation (clipping to range):

$$k[n] \leftarrow \min\big(\max(k[n], 0),\ L - 1\big)$$

Interpretation

Quantization = *rounding* amplitudes to a grid spaced by Δ.

Quantization error bounds

For a uniform rounding quantizer (no overload/clipping), the error is bounded:

$$-\frac{\Delta}{2} \leq e[n] < \frac{\Delta}{2}$$

Key idea

Smaller Δ (more bits) $\Rightarrow$ smaller maximum error.

Overload distortion (clipping)

If $|x[n]| > A$, quantizer saturates and error can be large:

$$x_q[n] = \text{clip}(x[n], -A, A)$$

This is often more audible than quantization noise.

Quantization noise model (approximation)

A common model (when signal is not too small and not highly correlated with steps):

$$e[n] \sim \text{uniform}\left(-\frac{\Delta}{2}, \frac{\Delta}{2}\right), \quad \mathbb{E}[e[n]] = 0$$

Then the mean squared quantization error (noise power) is:

$$\sigma_e^2 = \mathbb{E}[e[n]^2] = \frac{\Delta^2}{12}$$

Why it matters

This lets us predict SNR and compare bit depths.

Signal-to-Quantization-Noise Ratio (SQNR)

Let signal power be:

$$P_x = \mathbb{E}[x[n]^2]$$

Quantization noise power (model):

$$P_e = \sigma_e^2 = \frac{\Delta^2}{12}$$

So:

$$\mathrm{SQNR} = \frac{P_x}{P_e} \quad \Rightarrow \quad \mathrm{SQNR}_{\mathrm{dB}} = 10\log_{10}\left(\frac{P_x}{P_e}\right)$$

Special case: full-scale sine wave

For a sine of amplitude A (peak), $P_x = A^2/2$ and (approximately):

$$\mathrm{SQNR}_{\mathrm{dB}} \approx 6.02B + 1.76 \text{ dB}$$

Audio interpretation: bit depth and dynamic range

A useful rule of thumb for ideal PCM:

$$\text{Dynamic range} \approx 6B \text{ dB}$$

Examples:

- $B = 8$ bits $\Rightarrow \approx 48$ dB (noticeable noise)
- $B = 16$ bits $\Rightarrow \approx 96$ dB (typical WAV audio)

Takeaway

More bits $\Rightarrow$ smaller Δ $\Rightarrow$ lower quantization noise, but clipping must still be avoided.

(Optional) Dither: making quantization noise less audible

Add small noise $d[n]$ before quantization:

$$x_d[n] = x[n] + d[n], \qquad x_q[n] = Q(x_d[n])$$

Purpose: reduce distortion and make error more noise-like.

Idea

Dither trades a small increase in noise for fewer unpleasant artifacts (especially at low signal levels).