16.36: Communication Systems and Networks

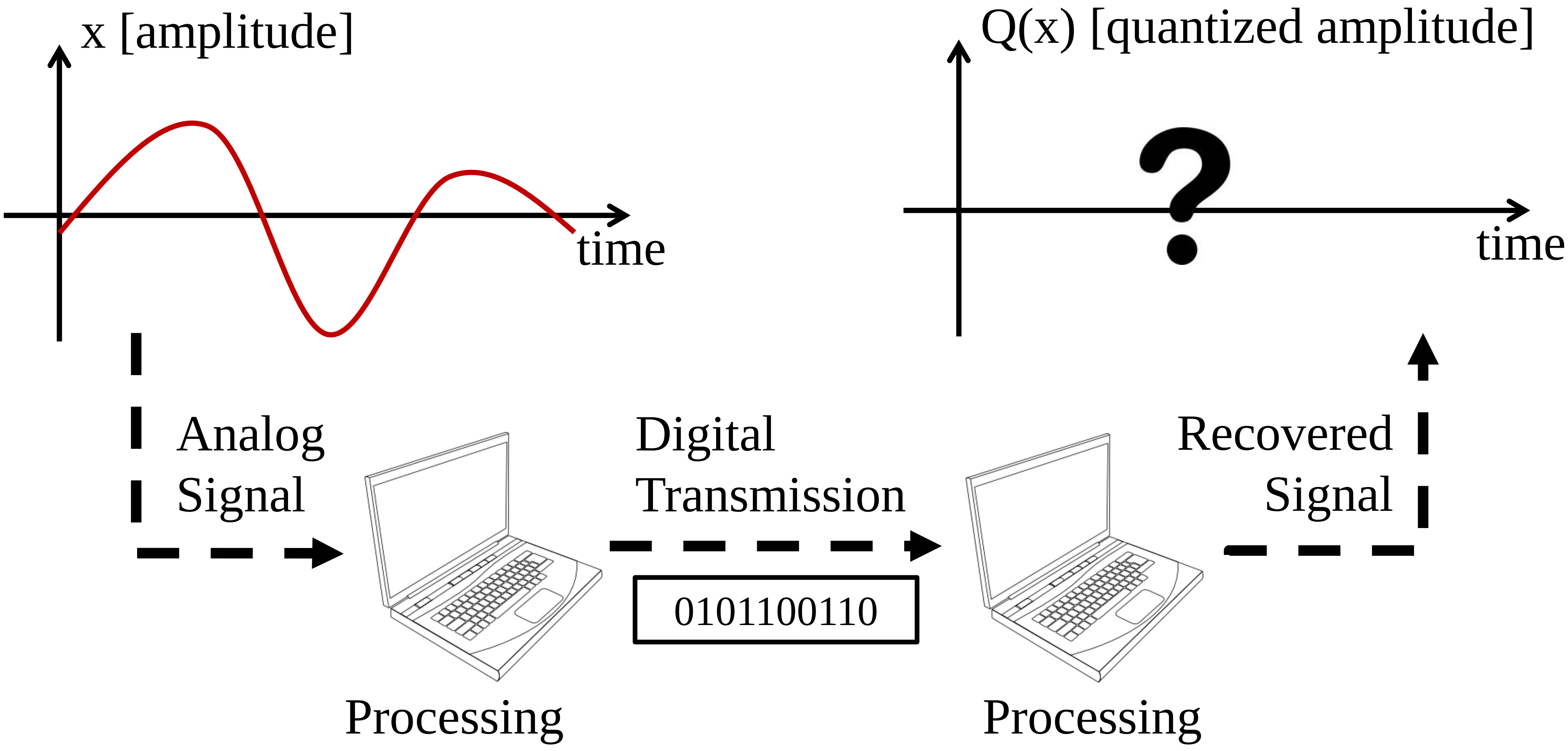
Lecture 5 - Quantization

Igor Kadota and Eytan Modiano
Laboratory for Information and Decision Systems
Massachusetts Institute of Technology

Cambridge, February 21, 2019

Motivation

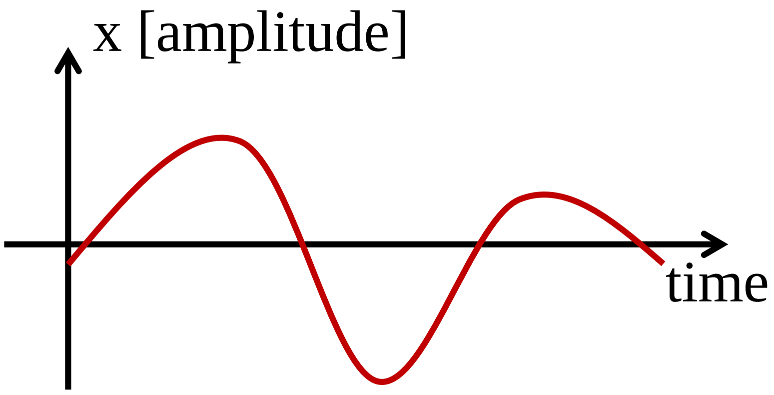
How would you transmit an analog signal through a digital medium?

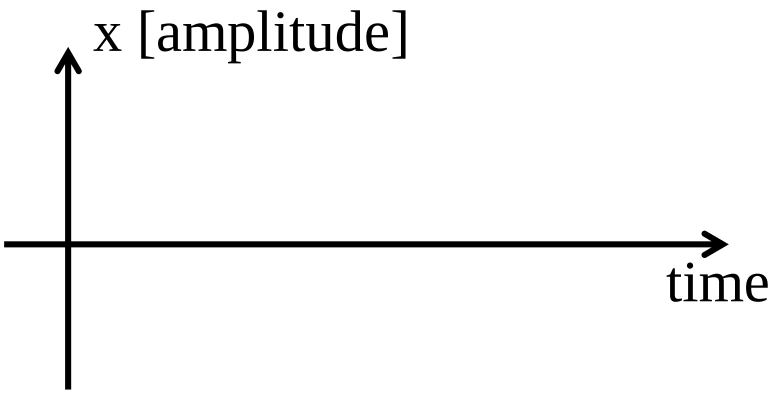


Motivation

Turning an **Analog** Signal into a **Digital** Signal.

What is the first step?

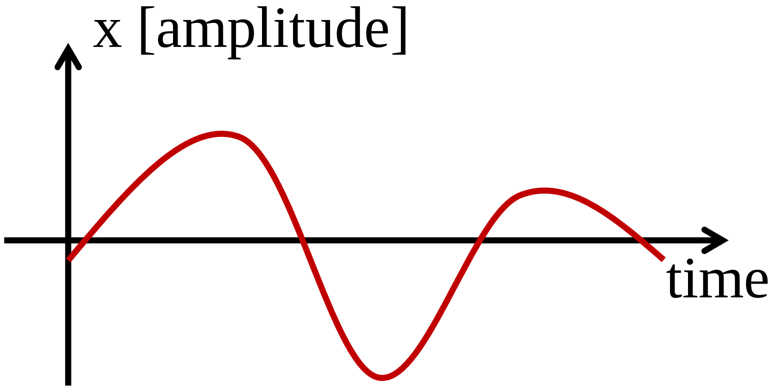


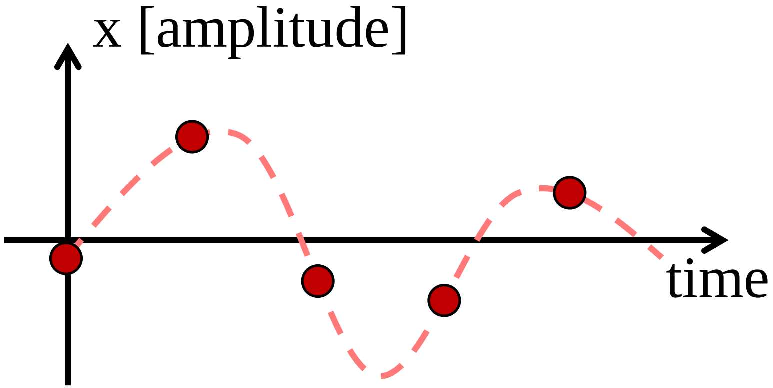


Motivation

Turning an **Analog** Signal into a **Digital** Signal.

Sampling (which frequency?)

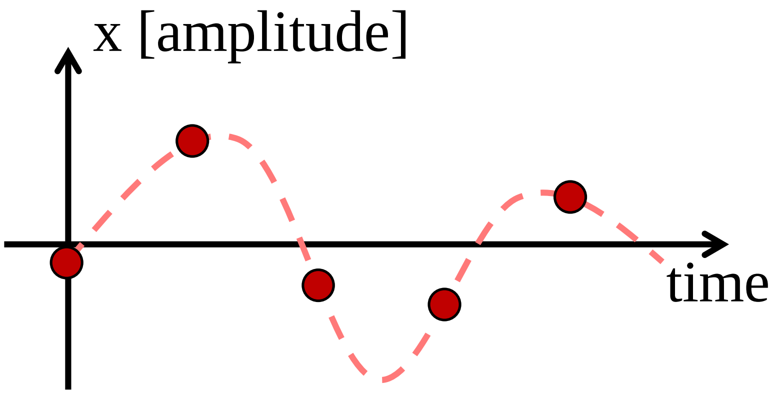


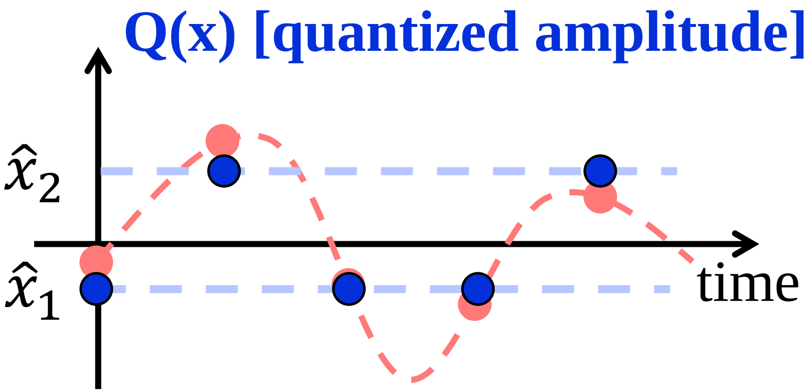


Motivation

Turning an **Analog** Signal into a **Digital** Signal.

How would you represent the samples using 2 amplitude levels?

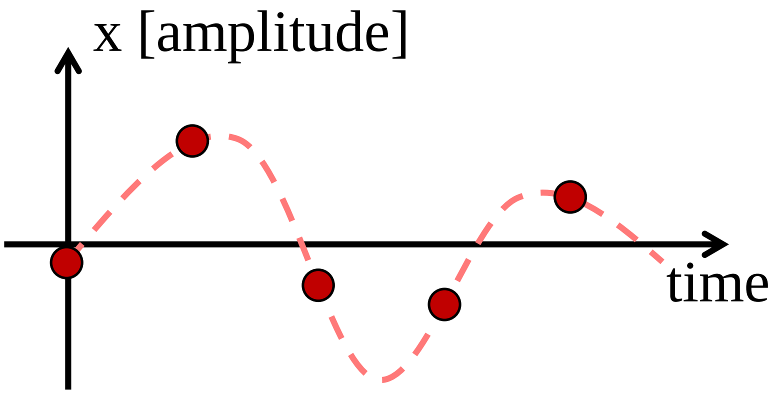


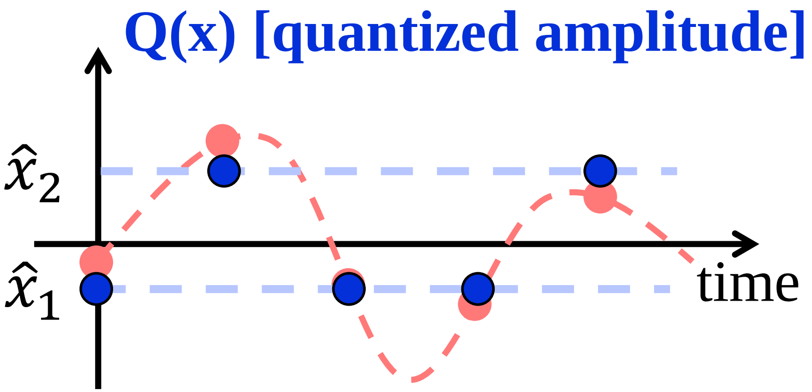


Motivation

Turning an **Analog** Signal into a **Digital** Signal.

Quantization





Choice of levels $\hat{x}_1, \hat{x}_2$ and of boundary $a_1 = \frac{\hat{x}_1 + \hat{x}_2}{2}$

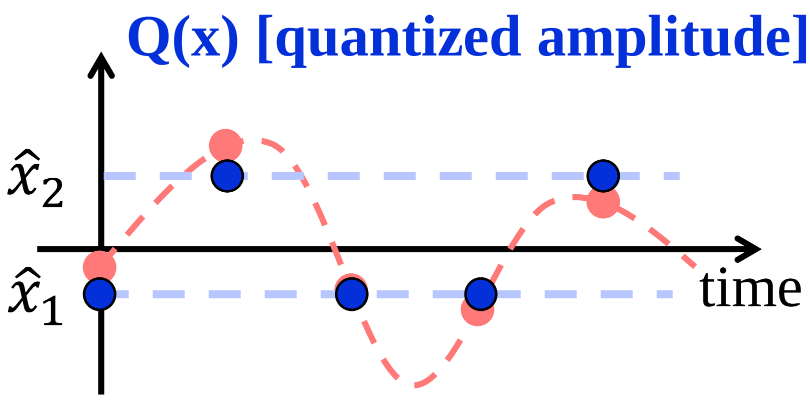
$$Q(x) \begin{cases} \hat{x}_2, & \text{if } x \geq \frac{\hat{x}_1 + \hat{x}_2}{2} \\ \hat{x}_1, & \text{if } x < \frac{\hat{x}_1 + \hat{x}_2}{2} \end{cases}$$

Motivation

Turning an **Analog** Signal into a **Digital** Signal.

How would you transform amplitude levels into **bits**?

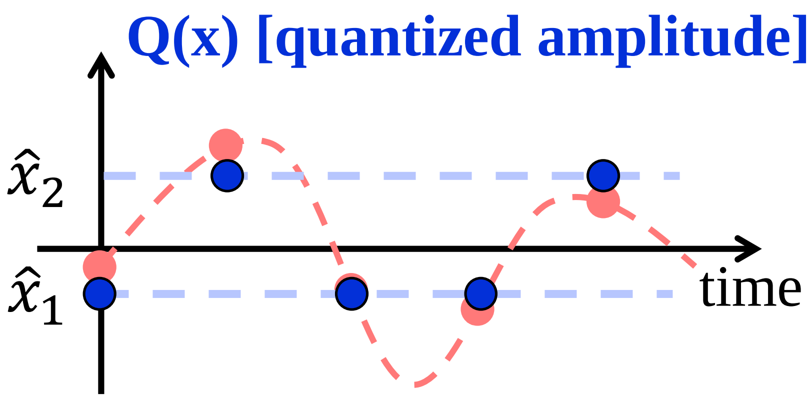
How many bits are necessary to represent two amplitude levels?



Motivation

Turning an **Analog** Signal into a **Digital** Signal.

Quantization



Bit Mapping

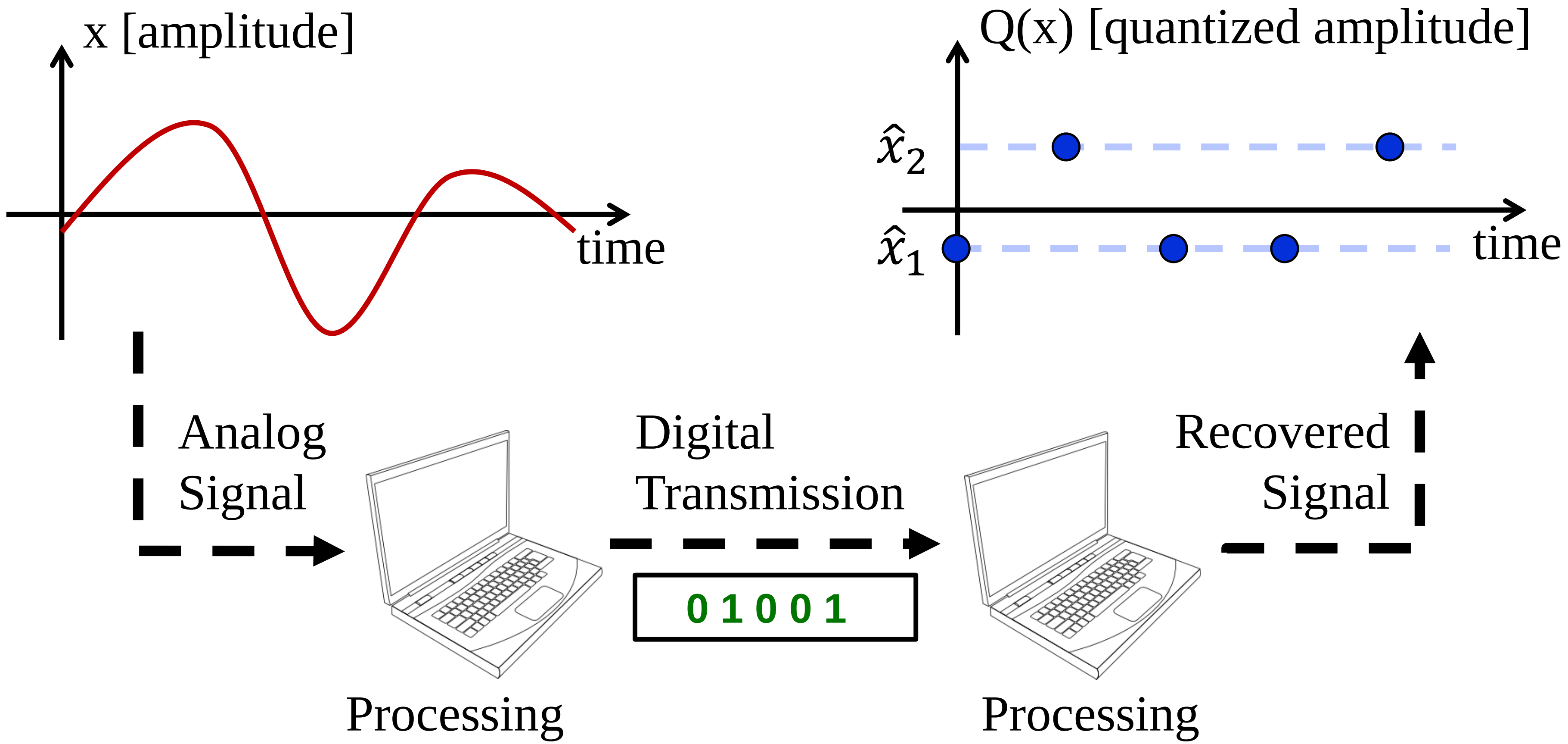
$$\text{bit} \begin{cases} 1, & \text{if } \hat{x}_2 \\ 0, & \text{if } \hat{x}_1 \end{cases}$$

Mapping may seem arbitrary.
After we discuss **Grey Coding**, come back to this lecture and think why Grey Coding should be used in Quantization!

Message = 0 1 0 0 1

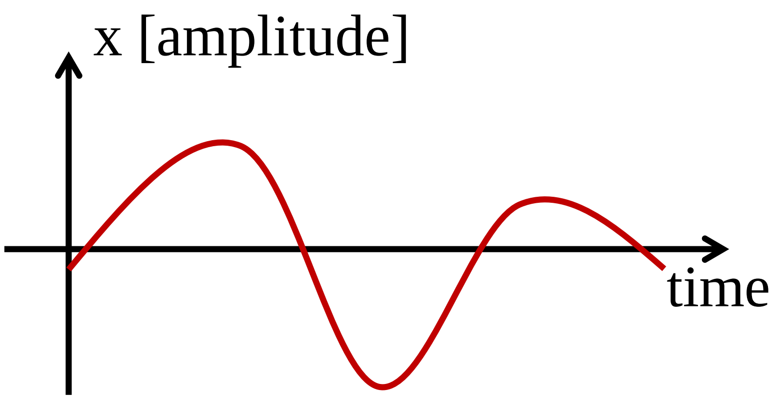
Motivation

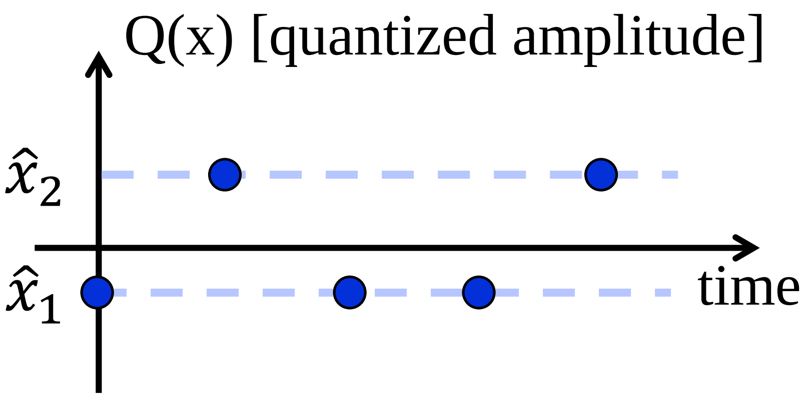
How would you transmit an analog signal through a digital medium?



Motivation

How would you transmit an analog signal through a digital medium?





Can we recover the original signal **(without errors)** at the destination?

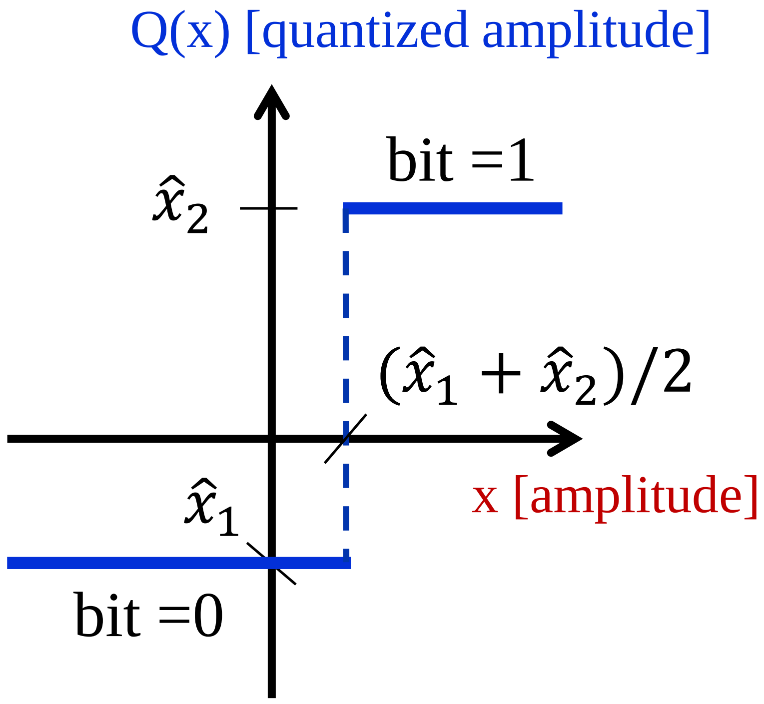
Objective: design quantizers with good performance (low error)

Quantization Scheme

In our motivation, we presented two mappings (on the LHS). They can be **summarized** using the scheme on the RHS

$$Q(x) \begin{cases} \hat{x}_2, & \text{if } x \geq \dfrac{\hat{x}_1 + \hat{x}_2}{2} \\ \hat{x}_1, & \text{if } x < \dfrac{\hat{x}_1 + \hat{x}_2}{2} \end{cases}$$

$$\text{bit} \begin{cases} 1, & \text{if } \hat{x}_2 \\ 0, & \text{if } \hat{x}_1 \end{cases}$$

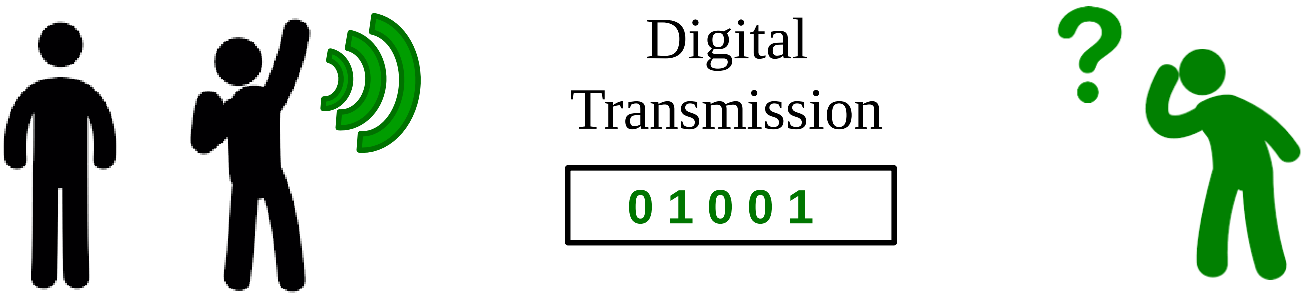


Example

- Student 1: generates samples at every 5 seconds. Each sample is a real number between -10 and 10 chosen at random.
- Student 2: maps each sample to a bit and transmits that bit to the receiver over the communication channel.
- Receiver recovers the original sample and writes on the board.

Setup:

- Student 2 and Receiver come up with a Quantization Scheme before starting the example.



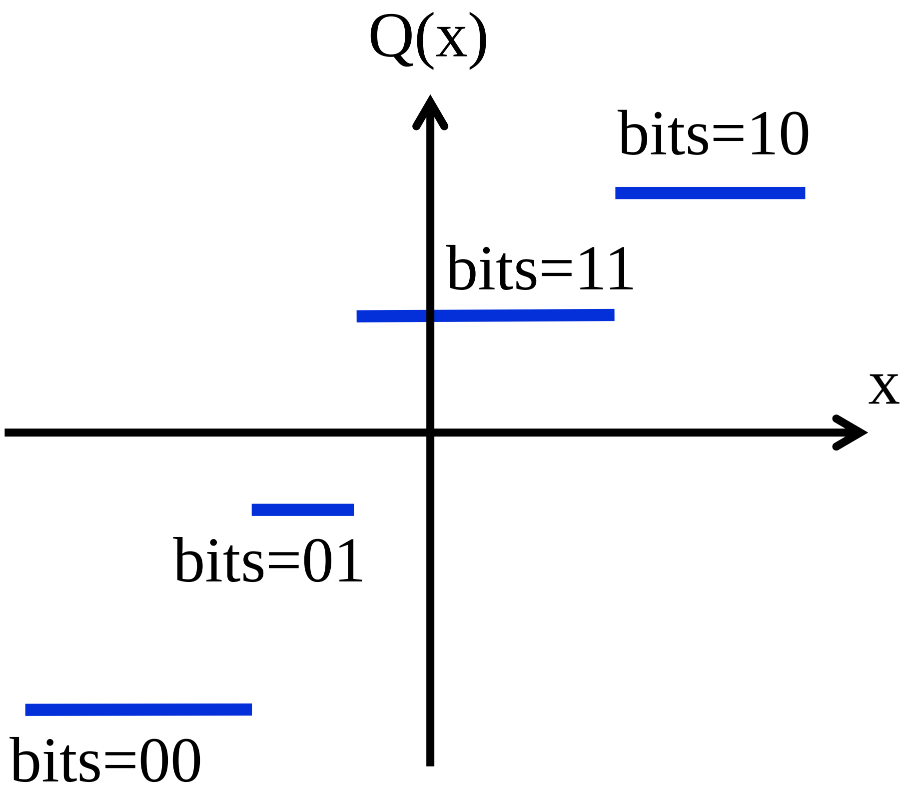
Question: how would you improve this system?

Quantizer

The **Quantization Scheme** provides all necessary information for the **quantization process.**

It maps all possible sample amplitudes to the quantized value (and the associated bit representation).

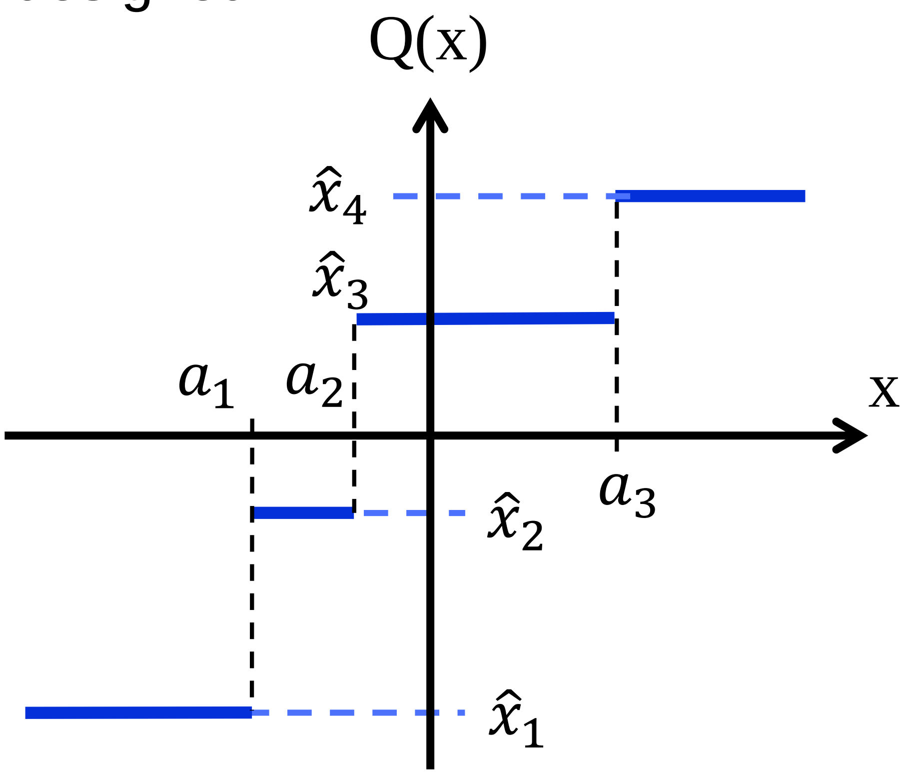
Notice that N levels imply in $\log_2(N)$ bits per sample.



General Quantizer

For a given number of levels N, there is a total of 2N-1 parameters to be designed:

- N-1 region boundaries a_i
- N quantization levels $\hat{x}_j$

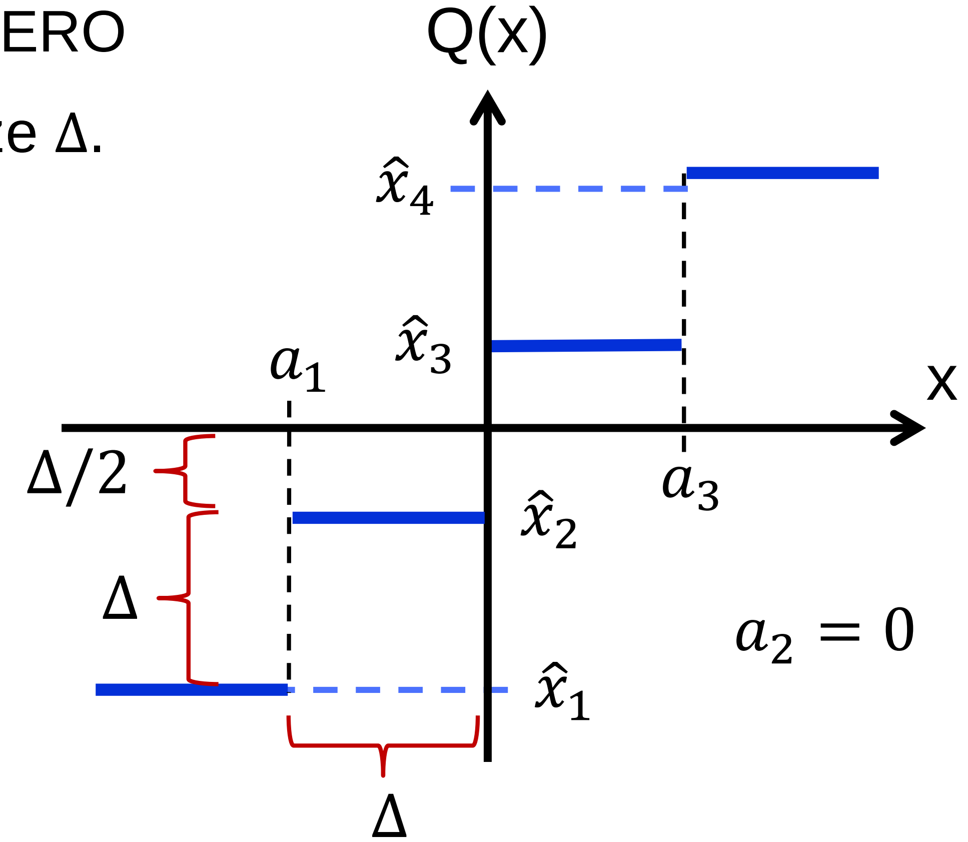


Simplified Uniform Quantizer

Single design parameter Δ.

The quantization scheme is as follow:

- Middle region boundary at ZERO
- All regions with the same size Δ.
- Quantization levels are the mid-points of the regions.



If samples are NOT finite valued, the first and last regions have infinite size.

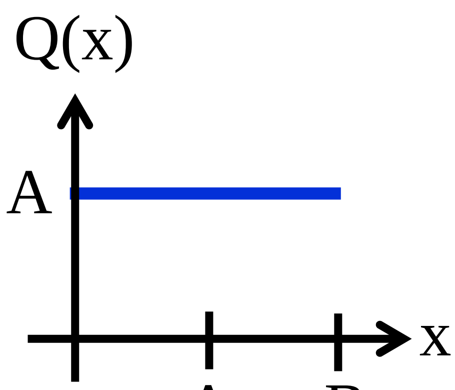
Quantization Error

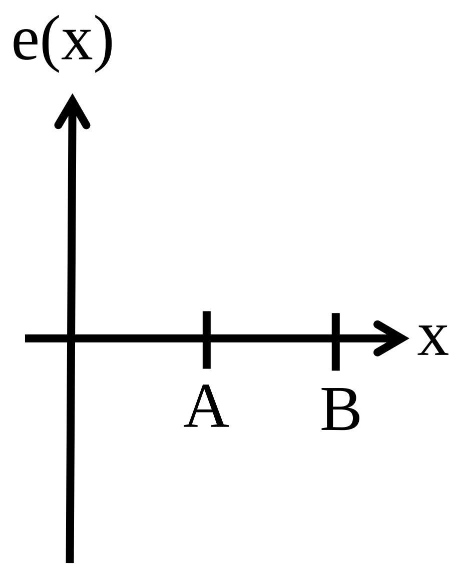
Information is lost in the process of quantization.

Performance metric:

$$e(x) = Q(x) - x \qquad \text{(Quantization Error)}$$

Quantization error:





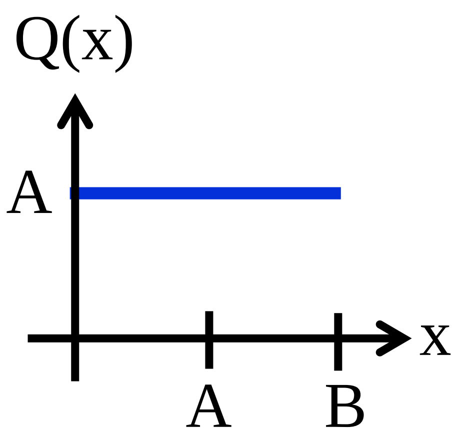
Quantization Error

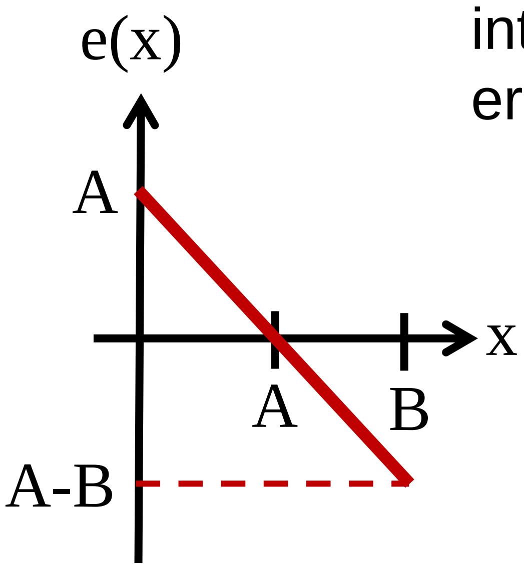
Information is lost in the process of quantization.

Performance metric:

$$e(x) = Q(x) - x \qquad \text{(Quantization Error)}$$

Quantization error:





If B = 2A, what is the integral of the quant. error in this interval?

Quantization Error

Information is lost in the process of quantization.

Performance metric:

$$e(x) = Q(x) - x \qquad \text{(Quantization Error)}$$

Squared-error distortion measure:

$$d\big(x, Q(x)\big) = e(x)^2 = (Q(x) - x)^2$$

Since X (amplitude of the sample) is a random variable:

$$D = \mathbb{E}[(Q(x) - x)^2] \qquad \text{(Quantization Noise)}$$

Signal-to-Quantization Noise Ratio: $SQNR = \dfrac{\mathbb{E}[X^2]}{\mathbb{E}[(Q(X) - X)^2]}$

Example for 2 levels

Sample amplitude (X) is uniformly distributed between –A and A.

$$f(x) = \frac{1}{2A}, x \in [-A, A]$$ and zero otherwise.

Simplified uniform quantizer with 2 levels → $\Delta = ?$

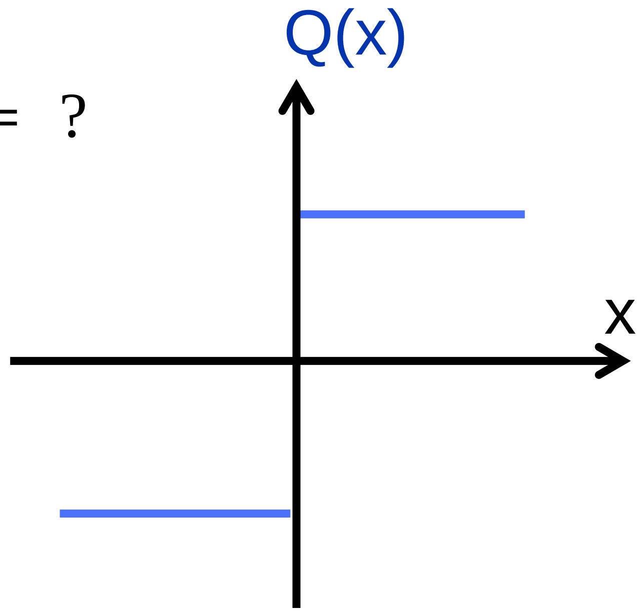
Noise:

$$\mathbb{E}[(Q(X) - X)^2] = \int_{x=?}^{?} (Q(x) - x)^2 f(x) dx = \ ?$$

Power:

$$\mathbb{E}[X^2] = \int_{x=?}^{?} x^2 f(x) dx = \ ?$$

$$\mathbf{SQNR} = \ ?$$



Solution for N levels

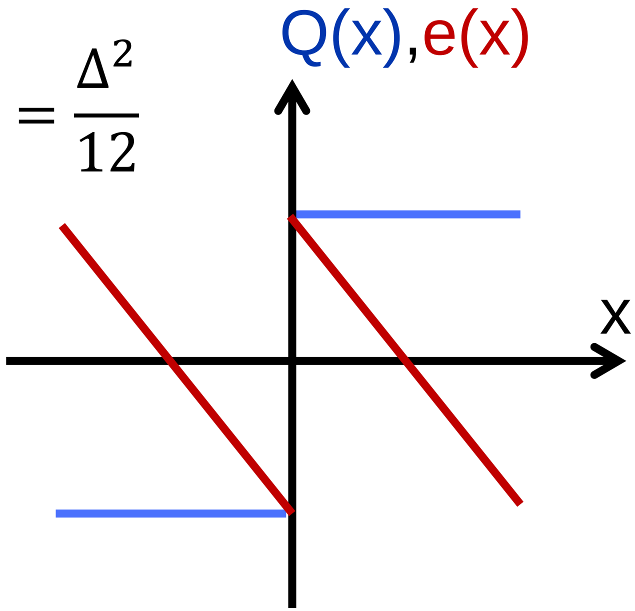
Sample amplitude (X) is uniformly distributed between –A and A.

$$f(x) = \frac{1}{2A}, x \in [-A, A] \text{ and zero otherwise.}$$

Simplified uniform quantizer with N levels → $\Delta = 2A/N$

Noise:

$$\mathbb{E}[(Q(X) - X)^2] = \int_{x=-A}^{A} (Q(x) - x)^2 f(x) dx = \frac{\Delta^2}{12}$$



Power:

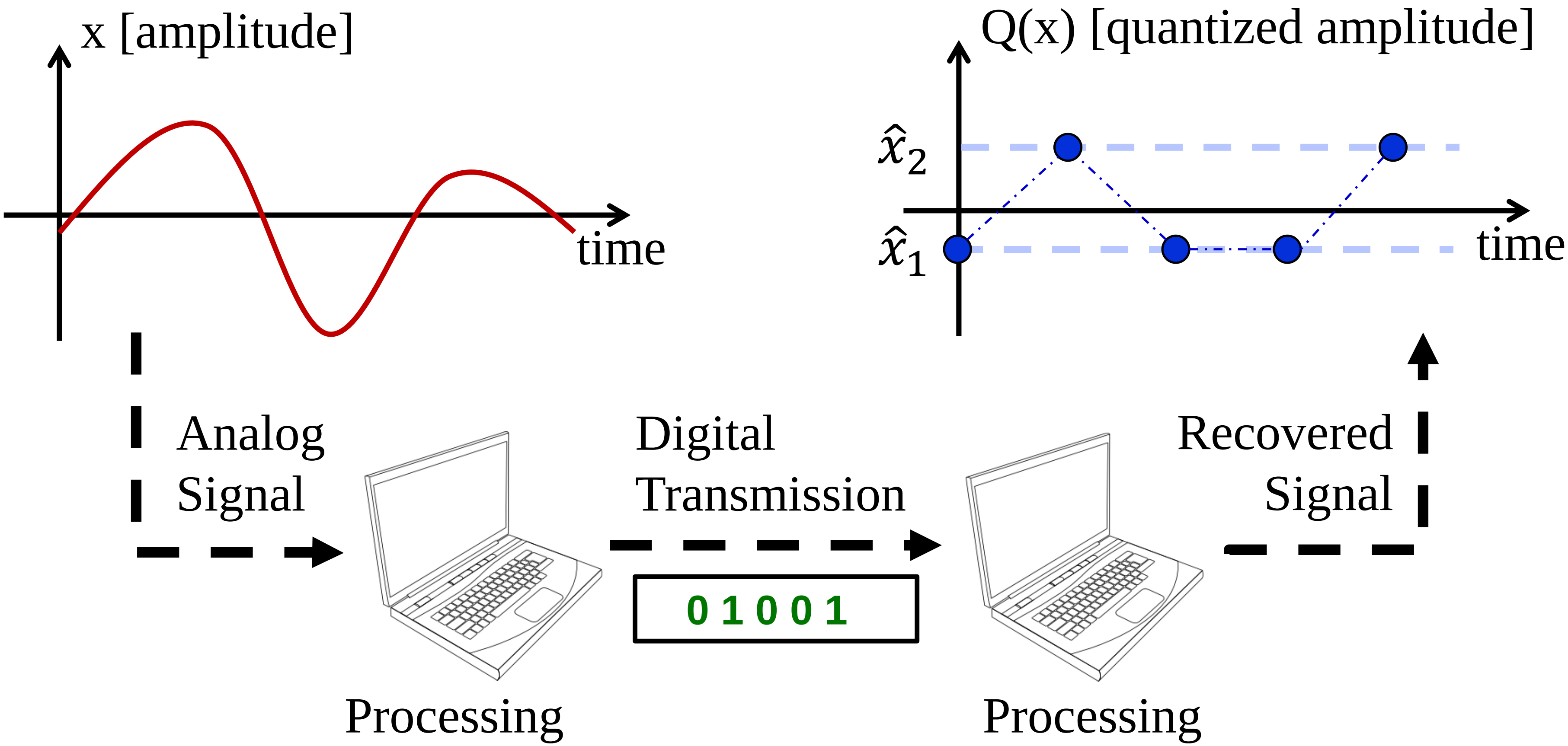
$$\mathbb{E}[X^2] = \int_{x=-A}^{A} x^2 f(x) dx = \frac{A^2}{3}$$

$$\textbf{SQNR} = N^2$$

what happens to SQNR when we add one bit to the quantizer?

MATLAB Demonstration

Sampling of audio signal with N=64,16, 4 and 2 levels.



Quantizer design

Uniform quantizer is good when input is uniformly distributed

When input is **not uniformly** distributed:

- Non-uniform quantization regions
- Finer regions around more likely values
- Optimal quantization values not necessarily the region midpoints

Possible Approaches:

- Use uniform quantizer anyway. Optimize the choice of Δ
- Use non-uniform quantizer. Choice of quantization regions and values
- Transform signal into one that looks uniform and use uniform quantizer

Optimal Uniform Quantizer

Given the number of regions, N

- Find the optimal value of Δ and **one** region boundary
- Find the optimal quantization levels $\hat{x}_j$ within each region
- Optimization over N+2 variables

Simplification: Let quantization levels be the midpoints of the quantization regions (except first and last regions, when input is not finite valued). Also, let the middle region boundary be at zero. This is the **Simplified Uniform Quantizer** discussed earlier.

Solve for Δ to minimize distortion:

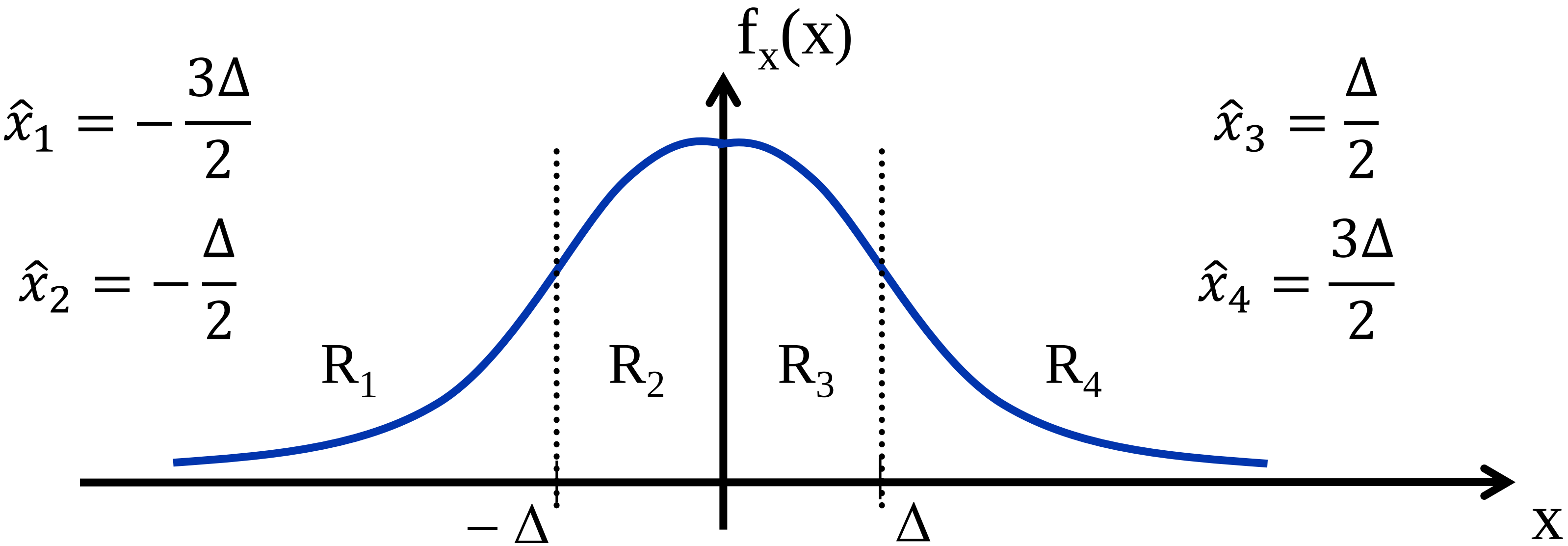
- Solution depends on input pdf and can be done numerically for commonly used pdfs (e.g., Gaussian, table 6.2, p. 296 of text)

Optimal Uniform Quantizer - Example

$N = 4, \quad X \sim N(0,1)$ with $f_x(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-x^2/2\sigma^2}$ and $\sigma^2 = 1$

From table 6.2, we have Δ=0.9957, D=0.1188, H(Q)= 1.904

- Two bits can be used to represent 4 quantization levels.
- Obs.: H(Q) bits are necessary to encode the quantized source



Optimal Non-uniform Quantizer

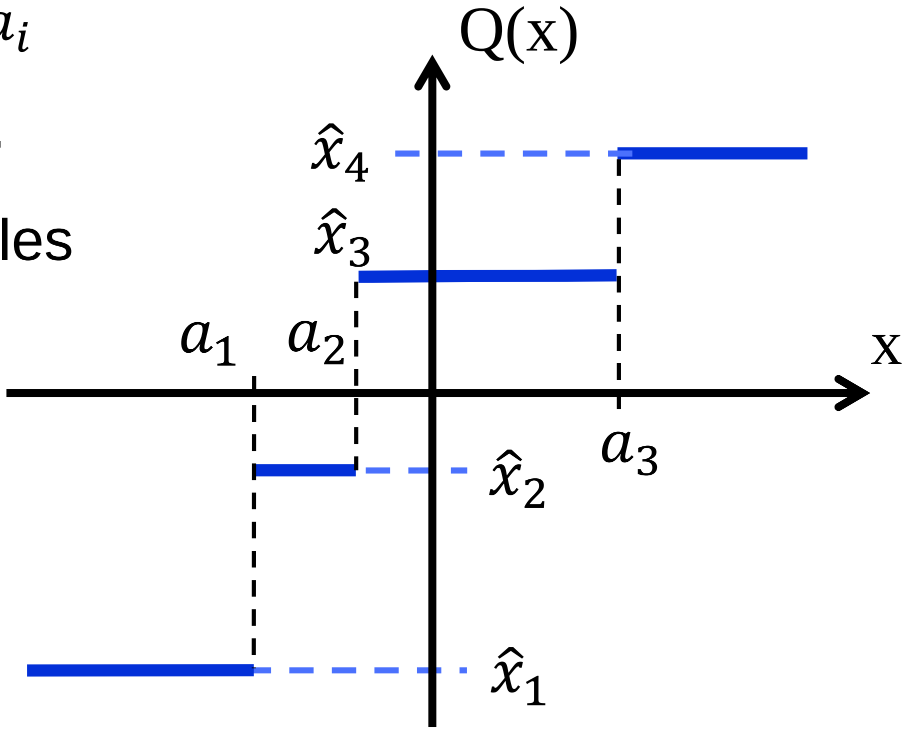
Quantization regions need not be of same length

Quantization levels need not be at midpoints

To minimize distortion, we need to determine:

- optimal quantization regions a_i
- optimal quantization levels $\hat{x}_j$.
- optimization over 2N-1 variables

  (high complexity)



Optimal Non-uniform Quantizer

Key questions:

Given quantization regions, what should the quantization levels be?

Given quantization levels, what should the quantization regions be?

Iterative approach for minimizing distortion:

Given regions, solve for quantization levels

Then, with the new levels, solve for quantization regions.

Loop until distortion stops improving.

Optimal Quantization Levels

- Goal is to minimize **distortion**, D
  - Optimal level affects distortion only *within its region.*

$$D_i = \int_{x=a_{i-1}}^{a_i} (x - \hat{x}_i)^2 f_x(x) dx$$

$$\frac{dD_i}{d\hat{x}_i} = \int_{x=a_{i-1}}^{a_i} 2(x - \hat{x}_i) f_x(x) dx = 0$$

$$\hat{x}_i = \int_{x=a_{i-1}}^{a_i} x f_x(x | a_{i-1} \le x \le a_i) dx = \mathbb{E}[X | a_{i-1} \le x \le a_i]$$

- Quantization values should be the “centroid” of their regions
  - The conditional expected value of that region

Optimal Quantization Regions

- Goal is to minimize **distortion**, D
  - Take derivative with respect to integral boundaries

$$D_i = \int_{x=a_{i-1}}^{a_i} (x - \hat{x}_i)^2 f_x(x) dx \qquad D_{i+1} = \int_{x=a_i}^{a_{i+1}} (x - \hat{x}_{i+1})^2 f_x(x) dx$$

$$\frac{dD}{da_i} = [(a_i - \hat{x}_i)^2 - (a_i - \hat{x}_{i+1})^2] f_x(a_i) = 0$$

$$a_i = \frac{\hat{x}_i + \hat{x}_{i+1}}{2}$$

- Boundaries of the quantization regions are the midpoint of the quantization values

Optimal Non-uniform Quantizer

Necessary conditions for optimality:

1. Quantization levels are the "centroid" of their region
2. Boundaries of the quantization regions are the midpoint of the quantization values

Clearly 1 depends on 2 and vice versa. The two can be solved iteratively to obtain an optimal quantizer.

Lloyd-Max algorithm:

Start with arbitrary regions (e.g., uniform Δ)

A) Find optimal quantization values ("centroids")

B) Use quantization values to get new regions ("midpoints")

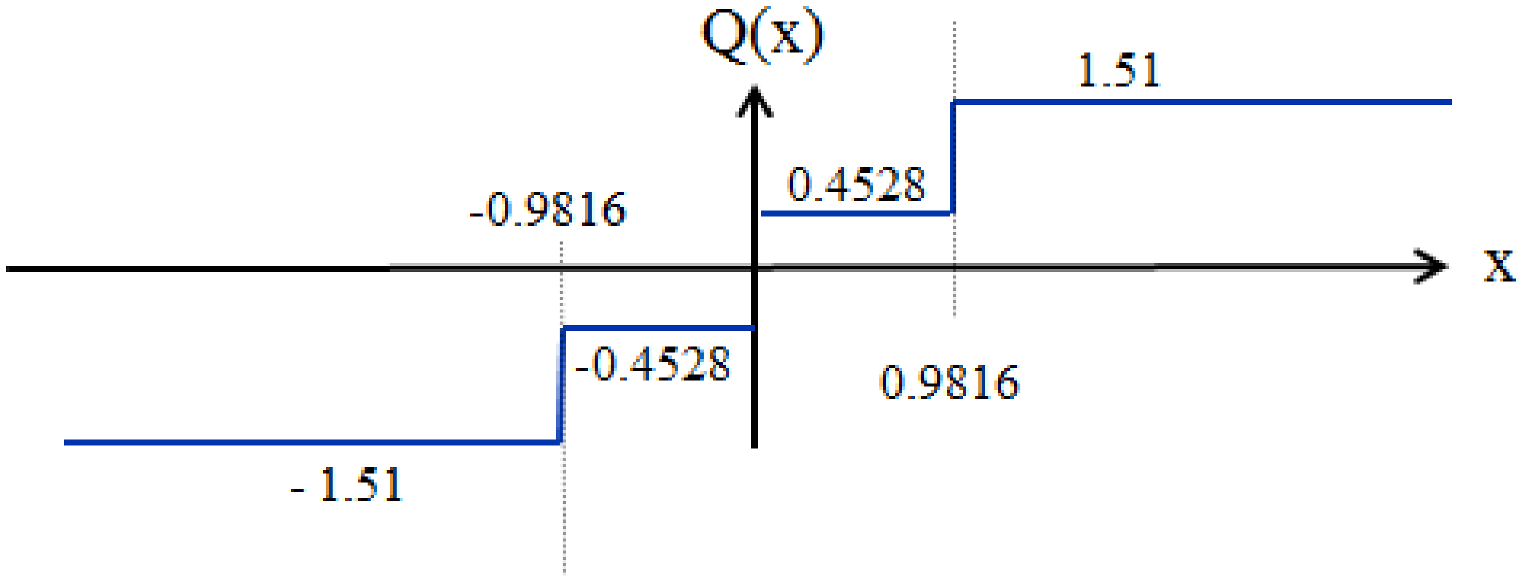
– Repeat A & B until further improvement in D is negligible.

Lloyd-Max algorithm - Example

Output of this algorithm is well-known for some common distributions.

- Table 6.3 (p. 299) gives optimal quantizer for Gaussian source.

For N = 4, D = 0.1175, H(Q) = 1.911



Optimal UNIFORM was: D= 0.1188, H(Q) = 1.904 (slight improvement)

(from previous slide)

Companders

Non-uniform quantizer can be difficult to design

- Requires knowledge of source statistics
- Different quantizers for different input types

Solution: Transfer input signal into one that looks uniform and then use uniform quantizer

Speech signal: high probabilities for low amplitudes

- **Compress the large amplitudes** before performing uniform quantization

μ -law compander

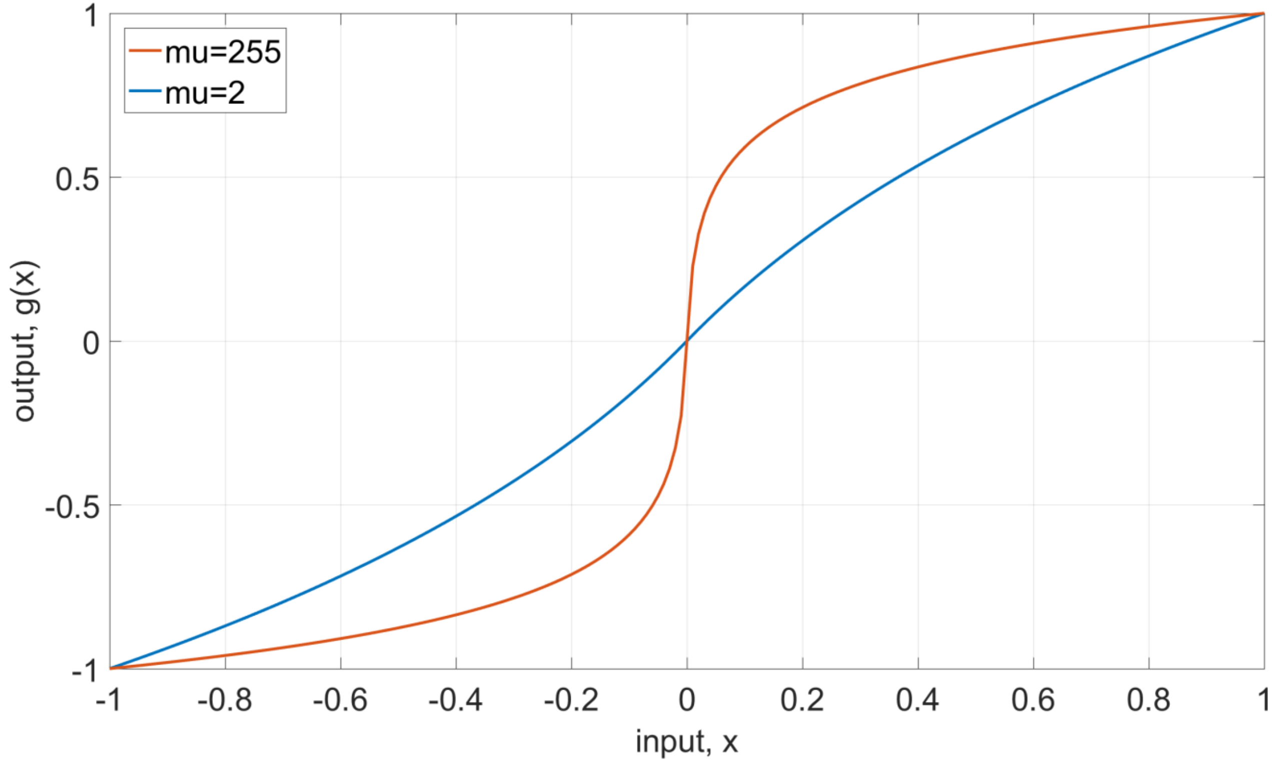
$$g(x) = \frac{\log(1+\mu|x|)}{\log(1+\mu)}\,\mathrm{sgn}(x), \quad x \in [-1,1]$$

- μ controls the level of compression
- μ = 255 typically used for voice

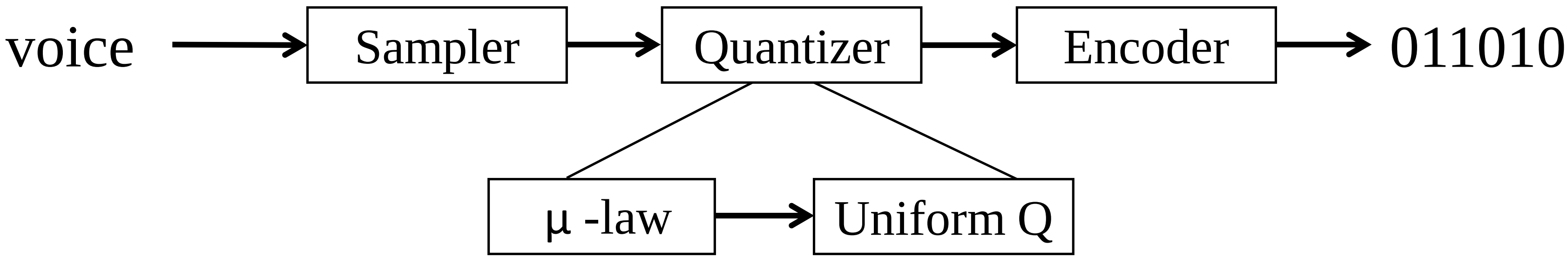
Companders

μ -law compander $$g(x) = \frac{\log(1+\mu|x|)}{\log(1+\mu)}\mathrm{sgn}(x), \quad x \in [-1,1]$$

- μ controls the level of compression
- μ = 255 typically used for voice



Pulse Code Modulation



Uniform PCM: x(t) ∈ $[-X_{max}, X_{max}]$

- N = 2^v quantization levels, each level encoded using v bits
- Uses a simplified uniform quantizer with **no compander**.
- SQNR: same as uniform quantizer

$$SQNR = \frac{\mathbb{E}[X^2] \times 3 \times 4^v}{X_{MAX}^2}, \text{since } \Delta = \frac{X_{max}}{2^{\nu-1}}$$

- Notice that increasing the number of bits by 1 increases SQNR by a factor of 4 (6 dB)

Speech coding

PCM with compander and μ = 255 (Non-uniform PCM)

Uniform quantizer with 128 levels, $N = 2^7$, 7 bits per sample

Speech typically limited to 4kHz

- Sample at 8kHz ⇒ Ts = 1/8000 = 125 μs

  8000 samples per second at 7 bits per sample => 56 kbps

Differential PCM:

- Speech samples are typically correlated
- Instead of coding samples independently, code the difference between samples
- Result: improved performance, lower bit rate speech