Signal Processing EE2S31

Digital Signal Processing
Lecture 6: Quantization and round-off effects

Borbala Hunyadi

Delft University of Technology, The Netherlands

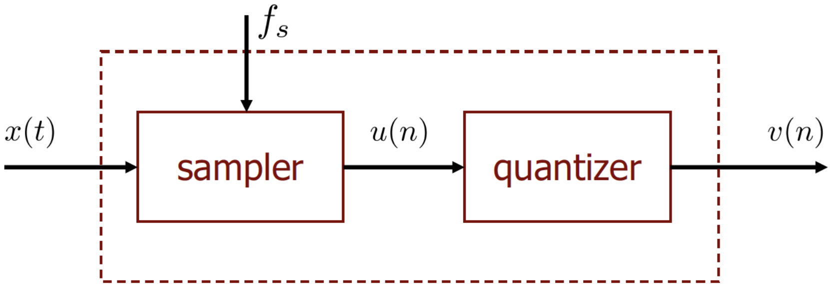
Outline

- Quantization
- Coding
- Its effect on digital filters

Outline

- Quantization
- Coding
- Its effect on digital filters

A/D converter



Basic task: convert a continuous range of input amplitudes to a discrete set of digital code words.

A/D converters

- quantization: a non-linear and non-invertible process that maps a given amplitude $x[n] = x_a(nT_s)$ at time $t = nT_s$ into an amplitude $\hat{x}_k$ taken from a finite set of values (*quantization level* or *alphabet*)

A/D converters

- quantization: a non-linear and non-invertible process that maps a given amplitude $x[n] = x_a(nT_s)$ at time $t = nT_s$ into an amplitude $\hat{x}_k$ taken from a finite set of values (*quantization level* or *alphabet*)

- coding: assigns a unique binary number (*code*) to each and every quantization level. This process is invertible (lossless).

Quantization

An L-level quantizer is characterized by

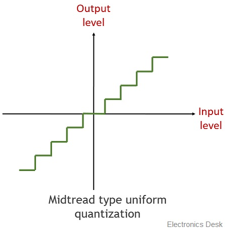
- a set of L+1 decision thresholds $x_1 < x_2 < ... < x_{L+1}$ and
- a set $\hat{X} = \{\hat{x}_k,\ k = 1, ..., L\}$ reconstruction values or quantization levels
- such that $\hat{x}[n] = \hat{x}_k$ if and only if $x_k \le x[n] < x_{k+1}$, where $x_1 = -\infty$ and $x_{L+1} = \infty$
- where the intervals $I_k = [x_k, x_{k+1})$ are called decision intervals or quantization cells

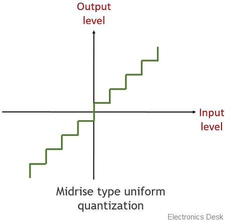
The map $Q : X \to \hat{X}$, which is a staircase function by definition, is given by:

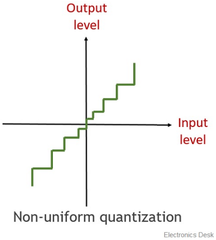
$$Q(x) = \hat{x}_k \text{ for } x \in I_k, \text{ k=1,...,L}$$

Quantization

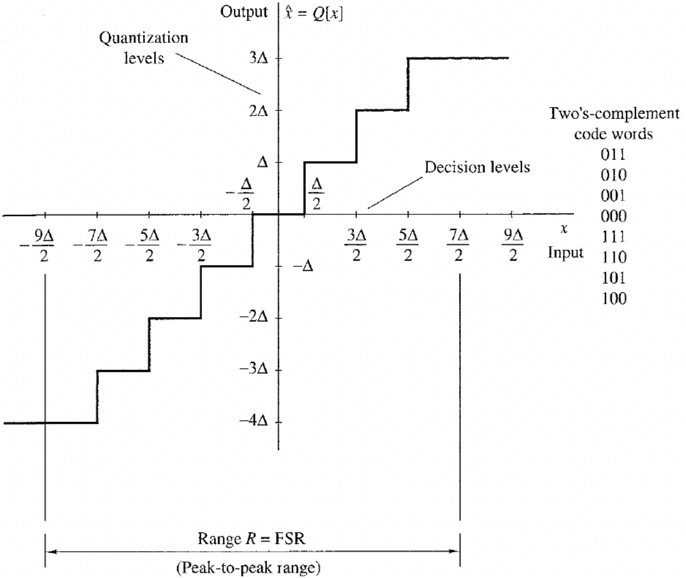
- uniform/non-uniform
- midtread/midrise







Quantization



Quantization

Properties of the uniform (linear) quantizer:

- $x_{k+1} - x_k = \Delta$
- $\hat{x}_k = (x_{k+1} - x_k)/2 \Rightarrow \hat{x}_{k+1} - \hat{x}_k = \Delta$

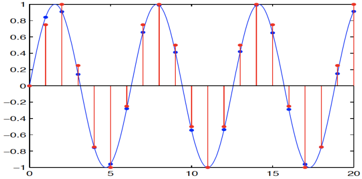
Δ is called the step size of the quantizer
The quantization error $z[n] = x[n] - \hat{x}[n]$ satisfies

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

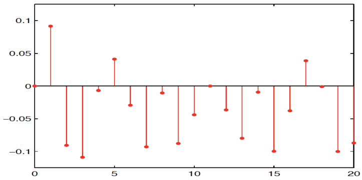
Analysis of quantization error

Example:

sampled signal (original (blue) and quantized (red))



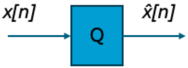
quantization error

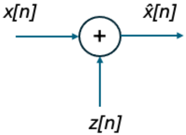


The quantization function is nonlinear (staircase function). The quantization error depends on the charateristics of the input function. For these reasons, deterministic analysis of the quantization error is intractable.

Statistical analysis of quantization error

Mathematical model of quantization:

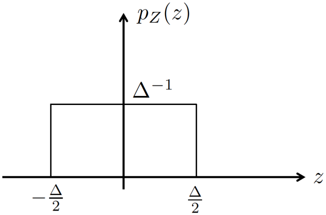




Assumptions:

- input signal $x[n]$ is the realization of a zero-mean WSS process
- quantization noise $z[n]$ is white (uncorrelated) and uniform
- quantization noise is uncorrelated to the input

Statistical analysis of quantization error



Then, the quantization noise power (= variance) of a quantizer with resolution (= step size) Δ is

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

- Proof? (Variance of a random variable with given PDF)

Signal to quantization noise ratio (SQNR)

Signal-to-quantization noise ratio (SQNR) quantifies the effect of additive noise on the desired signal:

- Let's denote the range of the quantizer with R
- Let's use $B+1$ bits to represent the quantized values
- Then

$$\Delta = \frac{R}{2^{B+1}}$$

- Therefore, the SQNR is:

$$SQNR = 10\log_{10}\left(\frac{\sigma^2(x)}{\sigma^2(z)}\right) = 10\log_{10}\frac{12\sigma^2(x)}{\Delta^2} =$$

$$= 6,02B + 16,81 + 20\log_{10}\left(\frac{\sigma(x)}{R}\right)$$

Every additional bit results in a 6dB increase in SQNR.

Outline

- Quantization
- **Coding**
- Its effect on digital filters

Coding

The *coding* process assigns a unique binary number to each quantization level.

Fixed-point representation

- String of digits with a decimal point, e.g.

$$3.14 \text{ or } 010.01$$

- Covers a fixed range of numbers with a fixed resolution, depending on the number of digits. For binary representation with B bits plus a sign bit:

$$\Delta = \frac{x_{max} - x_{min}}{m - 1}, \text{ with } m = 2^{B+1}$$

Fixed-point representation

$$X = (b_{-A}, \dots, b_{-1}, b_0, b_1, \dots b_B)_r = \sum_{i=-A}^{B} b_i r^{-i}$$

- For example:

$$(3.14)_{10} = 3 \cdot 10^0 + 1 \cdot 10^{-1} + 4 \cdot 10^{-2}$$
$$(010.01)_2 = 0 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 + 0 \cdot 2^{-1} + 0 \cdot 2^{-2}$$

- r: radix or base; e.g. $r = 2$ for binary
- A: number of integer digits, B: number of fractional digits

Often used n–bit integer format:

- $A = 0$ and $B = n - 1$, with a binary point between b_0 and b_1

Fixed-point signed binary format

There are various possible formats:

- signed-magnitude (SM)
- one's complement (1C)
- two's complement (2C)

Positive numbers are the same in all formats. Example:

- $X = (0.101)_2 = 2^{-1} + 2^{-3} = 1/2 + 1/8 = 5/8$

Negatige numbers:

- $X_{SM} = (1.101)_2 = -(2^{-1} + 2^{-3}) = -(1/2 + 1/8) = -5/8$
- $X_{1C} = (1.010)_2 = -5/8$
- $X_{2C} = (1.011)_2 = -5/8$

Fixed-point signed binary format

There are various possible formats:

- signed-magnitude (SM)
- one's complement (1C)
- two's complement (2C)

Positive numbers are the same in all formats. Example:

- $X = (0.101)_2 = 2^{-1} + 2^{-3} = 1/2 + 1/8 = 5/8$

Negatige numbers:

- $X_{SM} = (1.101)_2 = -(2^{-1} + 2^{-3}) = -(1/2 + 1/8) = -5/8$
- $X_{1C} = (1.010)_2 = -5/8$ $\quad \downarrow\ \overline{b_i} = 1 - b_i$
- $X_{2C} = (1.011)_2 = -5/8$

Fixed-point signed binary format

There are various possible formats:

- signed-magnitude (SM)
- one's complement (1C)
- two's complement (2C)

Positive numbers are the same in all formats. Example:

- $X = (0.101)_2 = 2^{-1} + 2^{-3} = 1/2 + 1/8 = 5/8$

Negatige numbers:

- $X_{SM} = (1.101)_2 = -(2^{-1} + 2^{-3}) = -(1/2 + 1/8) = -5/8$
- $X_{1C} = (1.010)_2 = -5/8$
- $X_{2C} = (1.011)_2 = -5/8$ $\downarrow X_{2C} = X_{1C} + 00\ldots01$

Fixed-point signed binary format

There are various possible formats:

- signed-magnitude (SM) easy multiplication
- one's complement (1C) easy addition
- two's complement (2C) easy addition, larger range

Positive numbers are the same in all formats. Example:

- $X = (0.101)_2 = 2^{-1} + 2^{-3} = 1/2 + 1/8 = 5/8$

Negatige numbers:

- $X_{SM} = (1.101)_2 = -(2^{-1} + 2^{-3}) = -(1/2 + 1/8) = -5/8$
- $X_{1C} = (1.010)_2 = -5/8$
- $X_{2C} = (1.011)_2 = -5/8$

Floating-point format

- consists of 2 parts: mantissa and exponent

$$X = M \cdot 2^E$$

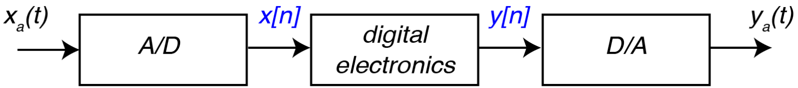
- It can cover a large dynamic range with varying resolution
- both mantissa and exponent are signed
- mantissa is a signed fraction, we can use SM, 1C or 2C
- multiplications: multiply mantissas, add exponents
- addition: equal exponents are needed → shifting

Quantization effects in digital filters

- Quantization of filter coefficients (9.5)
- Round-off effects in filter arithmetics (9.6.1)
- Statistical analysis of quantization effects (9.6.3)

Quantization effects in digital filters

- Quantization of filter coefficients (9.5)
- Round-off effects in filter arithmetics (9.6.1)
- Statistical analysis of quantization effects (9.6.3)



Quantization of filter coefficients

System function of a general IIR filter:

$$H(z) = \frac{B(z)}{A(z)} = \frac{\sum_{k=0}^{M} b_k z^{-k}}{1 - \sum_{k=0}^{N} a_k z^{-k}}$$

After quantization:

$$\hat{a}_k = a_k + \Delta a_k, \; \hat{b}_k = b_k + \Delta b_k$$

As a result, the practically implemented transfer function changes as follows:

$$\hat{H}(z) = \frac{\hat{B}(z)}{\hat{A}(z)} = \frac{\sum_{k=0}^{M} \hat{b}_k z^{-k}}{1 - \sum_{k=0}^{N} \hat{a}_k z^{-k}}$$

Quantization of filter coefficients

As a consequence, the position of the poles and zeros change as well:

$$\hat{p}_k = p_k + \Delta p_k$$
$$\hat{z}_k = z_k + \Delta z_k$$

It can be shown that:

$$\Delta p_k = \sum_{l=1}^{N} \frac{p_k^{N-l}}{\prod_{k=1, m\neq k}^{N} (p_k - p_m)} \Delta a_l$$

Closely spaced poles give rise to large errors!

Quantization of filter coefficients

Strategies to minimize the error Δp_k, i.e. $|p_k - p_l|$:

- Realize higher order filters with one or two-pole filter sections
- note: one-pole filter sections require complex arithmetic
- solution: use second order sections with complex-conjugated poles
- complex-conjugated poles are sufficiently far, i.e. perturbation error will be under control

Quantization of filter coefficients

Even in two-pole filter sections, the structure used to implement the section plays an important role in the error caused by coefficient quantization.

Consider the following filter:

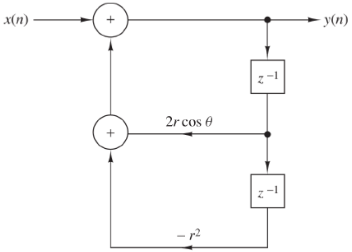
$$H(z) = \frac{1}{1 - 2rcos\theta z^{-1} + r^2 z^{-2}}$$

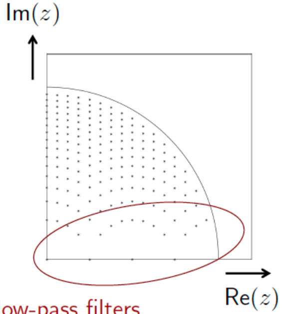
The filter has two poles, at $z = re^{\pm j\theta}$

Quantization of filter coefficients

Assuming 4-bit quantization:

- We need to quantize $2r\cos\theta$ and r^2, each will have 16 possible values
- What are the possible pole positions?
  - quantized values of $2r\cos\theta$ give uniformly placed positions along the real axis
  - quantized values of r^2 give positions at non-uniformly distributed distances from the origin



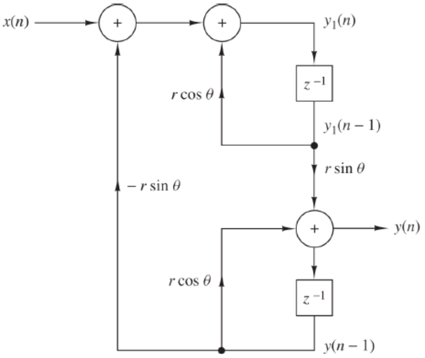


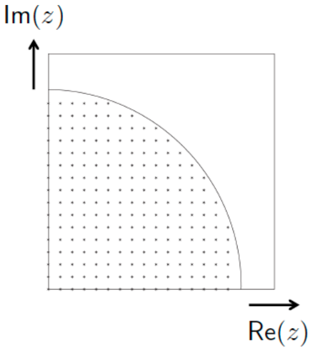
problems with low-pass filters

Quantization of filter coefficients

Alternative realization:

- We need to quantize $r\cos\theta$ and $r\sin\theta$
- What are the possible pole positions?
- Both linear in r!





Quantization of filter coefficients

General strategy:

- choose a realization which yields uniform pole positions
- unfortunately there is no systematic design method
- for higher order structures, cascade is preferred over parallel form
- floating point arithmetic is preferred over fixed-point

Practice:

- Exercise 9.33

Quantization effects in digital filters

- Quantization of filter coefficients (9.5)
- **Round-off effects in filter arithmetics** (9.6.1)
- Statistical analysis of quantization effects (9.6.3)

Round-off effects in filters arithmetics

- In recursive systems, non-linearities due to finite-precision arithmetic operations cause periodic oscillations, called **limit cycles**.
- Let's consider the followig single-pole system:

$$y(n) = ay(n-1) + x(n) \tag{1}$$

- The actual system, however, quantizes the result of the multiplication:

$$v(n) = Q[av(n-1)] + x(n) \tag{2}$$

With $a < 1$ the ideal system (1) decays towards zero exponentially (i.e. $y(n) = a^n \to 0$ as $n \to \infty$). What about the actual system (2)?

Round-off effects in filter arithmetics

The actual system's response $v(n)$ reaches a steady-state periodic output sequence, i.e. the limit cycle!

- Let us assume 4-bit fixed-point arithmetic (plus sign bit)
- Let us also assume that the product is "rounded upward" at the decision interval limits, i.e. $\hat{x}[n] = \hat{x}_k$ if and only if $x_k \leq x[n] < x_{k+1}$
- Let us assume that $x(n) = \frac{11}{16}\delta(n)$ and $a = \frac{3}{4}$ (see 3^{rd} column)

TABLE 9.2 Limit Cycles for Lowpass Single-Pole Filter

n	$a = 0.1000 = \frac{1}{2}$		$a = 1.1000 = -\frac{1}{2}$		$a = 0.1100 = \frac{3}{4}$		$a = 1.1100 = -\frac{3}{4}$	
0	0.1111	$(\frac{15}{16})$	0.1111	$(\frac{15}{16})$	0.1011	$(\frac{11}{16})$	0.1011	$(\frac{11}{16})$
1	0.1000	$(\frac{8}{16})$	1.1000	$(-\frac{8}{16})$	0.1000	$(\frac{8}{16})$	1.1000	$(-\frac{8}{16})$
2	0.0100	$(\frac{4}{16})$	0.0100	$(\frac{4}{16})$	0.0110	$(\frac{6}{16})$	0.0110	$(\frac{6}{16})$
3	0.0010	$(\frac{2}{16})$	1.0010	$(-\frac{2}{16})$	0.0101	$(\frac{5}{16})$	1.0101	$(-\frac{5}{16})$
4	0.0001	$(\frac{1}{16})$	0.0001	$(\frac{1}{16})$	0.0100	$(\frac{4}{16})$	0.0100	$(\frac{4}{16})$
5	0.0001	$(\frac{1}{16})$	1.0001	$(-\frac{1}{16})$	0.0011	$(\frac{3}{16})$	1.0011	$(-\frac{3}{16})$
6	0.0001	$(\frac{1}{16})$	0.0001	$(\frac{1}{16})$	0.0010	$(\frac{2}{16})$	0.0010	$(\frac{2}{16})$
7	0.0001	$(\frac{1}{16})$	1.0001	$(-\frac{1}{16})$	0.0010	$(\frac{2}{16})$	1.0010	$(-\frac{2}{16})$
8	0.0001	$(\frac{1}{16})$	0.0001	$(\frac{1}{16})$	0.0010	$(\frac{2}{16})$	0.0010	$(\frac{2}{16})$

Round-off effects in filter arithmetics

- The amplitude of the output during a limit cycle, $v_d(n)$, is confined to a certain range called the *dead band* of the filter.
- For a single-pole filter the dead band is determined by:

$$|v_d(n)| \leq \frac{\frac{1}{2}2^{-b}}{1-|a|}$$

Round-off effects in filter arithmetics

Practice

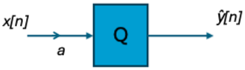
- Exercise 9.31
- Exercise 9.35

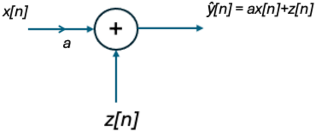
Outline

- Quantization of filter coefficients (9.5)
- Round-off effects in filter arithmetics (9.6.1)
- **Statistical analysis of quantization effects on digital filters** (9.6.3)

Model of noise in digital filters

The quantization error in multipliers can be modeled as additive, uniformly distributed white noise:





Superposition principle:

- The output of the system is equal to its response to the input plus its response to the quantization noise.
- In case of multiple noise sources, their effect is also additive.

Statistical analysis of quantization effects on digital filters

The effect of the quantization noise depends on the transfer function of the noise source to the output of the filter.

Recap: filtering stochastic processes

Let $g[n]$ denote the impulse reponse of an LTI system and $q[n]$ denote the response of this LTI system to a white stochastic input $z[n]$. Then,

$$\sigma_q^2 = \sigma_z^2 \sum_{n=-\infty}^{\infty} g(n)^2 = \frac{\sigma_z^2}{2\pi} \int_0^{2\pi} |G(e^{j\omega})|^2 d\omega$$

Recall related lectures from SP track!

Statistical analysis of quantization effects on digital filters

Let us consider a single-pole IIR filter:

$$h(n) = a^n u(n),\ |a| < 1$$

Therefore

$$\sum_{n=-\infty}^{\infty} h(n)^2 = \sum_{n=-\infty}^{\infty} a^{2n} = \frac{1}{1-a^2}$$

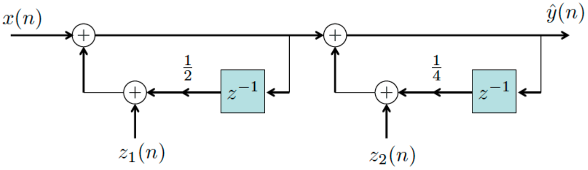
The noise power is enhanced relative to the input noise, depending on a:

$$\sigma_q^2 = \sigma_z^2 \frac{1}{1-a^2}$$

Statistical analysis of quantization effects

Example:

$$H(z) = \frac{z^2}{(z-\frac{1}{2})(z-\frac{1}{4})} = \frac{z}{(z-\frac{1}{2})} \cdot \frac{z}{(z-\frac{1}{4})}$$



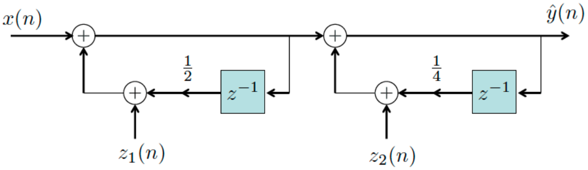
Let us consider a second-order filter $H(z)$, which is a cascade of two first-order filter sections $H_1(z)$ and $H_2(z)$.

- Due to superposition, the total noise power at the output is the sum of the output noise powers of $z_1(n)$ and $z_2(n)$.
- The transfer function of $z_1(n)$ to the output is $H(z)$, while the transfer function of $z_2(n)$ is $H_2(z)$.

Statistical analysis of quantization effects

Example:

$$H(z) = \frac{z^2}{(z-\frac{1}{2})(z-\frac{1}{4})} = \frac{z}{(z-\frac{1}{2})} \cdot \frac{z}{(z-\frac{1}{4})}$$



The impulse responses are as follows:

- $h(n) = (2(\frac{1}{2})^n - (\frac{1}{4})^n)u(n)$
- $h_2(n) = (\frac{1}{4})^n u(n)$

The output quantization noise power is:

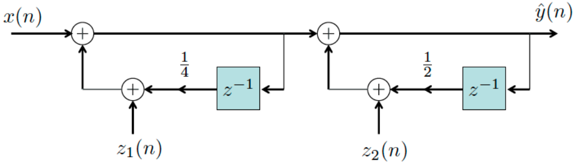
- $\sigma_{q_1}^2 = \frac{\Delta^2}{12} \sum (2(\frac{1}{2})^n - (\frac{1}{4})^n)^2 \approx 1.83 \frac{\Delta^2}{12}$
- $\sigma_{q_2}^2 = \frac{\Delta^2}{12} \sum (\frac{1}{4})^{2n} \approx 1.07 \frac{\Delta^2}{12}$

Total $2.90 \frac{\Delta^2}{12}$

Statistical analysis of quantization effects

What if we interchange the 2 sections? Is the output quantization noise power A: larger? B: smaller? C: equal?

$$H(z) = H_1(z)H_2(z) = H_2(z)H_1(z)$$



Statistical analysis of quantization effects

Practice:

- Exercise 9.32
- Exercise 9.34
- Exercise 9.38