Contents

1 Week 7 Module 5: Sampling and Quantization

Interpolation describes the process of building a continuous-time signal $\mathbf{x(t)}$ from a sequence of samples $\mathbf{x[n]}$. In other words, interpolation allows moving from the discrete-time world to the continuous-time world. Interpolation raises two interesting questions:

The first one is how to interpolate between samples?

- In the case, of two samples, this is simple enough and there is there is a straight line that goes between these two samples.
- In the case of three samples, similarly, you have a parabola that goes through these 3 samples.
- If you have many samples, you can try to do the same and go through all samples but you see this is a trickier issue compared to what we have done with two or three samples.

The second question is:

- is there a minimum set of values you need to measure the function at so that you can perfectly reconstruct it.

Later on in the module, we are going to study sampling, i.e. the process of moving from a continuous-time signal to a sequence of samples. In other words, sampling allows moving from the continuous-time world to the discrete-time world. Suppose we take equally-spaced samples of a function $\mathbf{x(t)}$. The question is when is there a one-to-one relationship between the continuous-time function and its samples, i.e. when do the samples form a unique representation of the continuous-time function? To answer this question, we are going to use all the tools in the toolbox that we have looked at so far:

- Hilbert spaces
- projections
- filtering
- sinc functions
- and so on.

Everything comes together in this module to develop a profound and very useful result, the **sampling theorem**.

Before moving to the heart of the topic, let us briefly review its history. The Shannon sampling theorem has a very interesting history which goes back well before Shannon. Numerical analysts were concerned about interpolating tables of functions and the first one to proove a version of the sampling theorem was Whittaker in England in 1915. Harry Nyquist at Bell Labs came up with the Nyqvist criterion, namely that a function that has a maximum frequency F0F_{0F0} could be sampled at 2F02F_{02F0}. In the Soviet Union, Kotelnikov proved a sampling theorem. The son of the first Whittaker further proved results on the sampling theorem. Then Herbert Raabe in Berlin wrote his PhD thesis about a sampling theorem that, wrong time wrong city, he got zero credit for. Denis Gabor worked on a version of the sampling theorem in the mid 1940s. Then Claude Shannon, the inventor of information theory, wrote a beautiful paper that is in the further reading for this class where the Shannon sampling theorem appears in the form that we use today. Last but not least, in 1949 Someya in Japan also proved the sampling theorem. You can see that it's a very varied history, it's a fundamental result where many people independently came up with this result.

1.1 The Continous-Time World

1.1.1 Introduction

The continuous-time world is the world we live in, the physical reality of the world, in contrast with the discrete-time world, the world inside a computer. We are first going to look at models of the world and compare digital with analog views of the world. Then we are going to study continuous-time signal processing in greater details. Furthermore, we will introduce the last form of Fourier transform we have not yet encountered in this class, the continuous-time Fourier transform.

1.1.2 The continous-time paradigm

Two views of the world

Sampling: $x(t) \to x[n]$ Interpolation: $x[n] \to x(t)$

1.2 Continuous-time signal processing

time real variable t

Table 1: Two views of the world 1

Digital World	**Analog World**
arithmetic	calculus
combinatorics	distributions
computer sience	system theory
DSP	electronics

Table 2: Two views of the world 2

Digital World	**Analog World**
countable integer index n	real-valued time t [sec]
sequences $x[n] \in \ell_2(\mathbb{Z})$	function $x(t) \in L_2(\mathbb{R})$
frequency $\omega \in [-\pi, \pi]$	frequency $\Omega \in \mathbb{R}(rad/sec)$
DTFT: $\ell_2(\mathbb{Z}) \to L_2[-\pi, \pi]$	FT: $L_2(\mathbb{R}) \to L_2(\mathbb{R})$

signal x(t) complex function of areal variable

finite energy $x(t) \in L_2(\mathbb{R})$

inner product in $L_2(\mathbb{R})$ $\langle x(t), y(t)\rangle = \int_{-\infty}^{\infty} x^*(t)\, y(t)$

energy $||x(t)||^2 = \langle x(t), x(t)\rangle$

1.2.1 Analog LTI filters

$$\begin{aligned} y(t) &= (x * h)t \\ &= \langle h^*(t-\tau), x(\tau)\rangle \\ &= \int_{-\infty}^{\infty} x(\tau)\, h(t-\tau)\, d\tau \end{aligned}$$

1.2.2 Fourier analysis

- in discrete time max angular frequency is $\pm\pi$
- in continous time no max frequency: $\Omega \in \mathbb{R}$
- concept is the same:

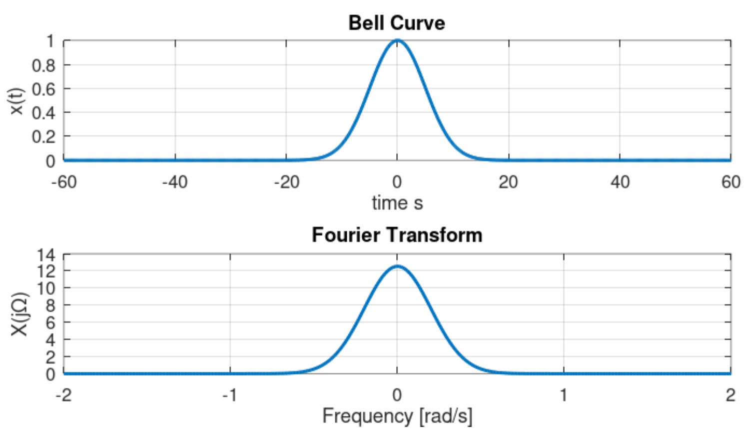
$$X(j\Omega) = \int_{-\infty}^{\infty} e^{-j\Omega\, t}\, dt$$

$$x(t) = \frac{1}{2\,\pi} \int_{-\infty}^{\infty} X(j\Omega)\, e^{j\Omega\, t} dt$$

1.2.3 Real-world frequency

- Ω expresse in rad/s
- $F = \dfrac{\Omega}{2\,\pi}$, expressed in Hertz (1/s)
- period $T = \dfrac{1}{F} = \dfrac{2\,\pi}{\Omega}$

1.2.4 Example



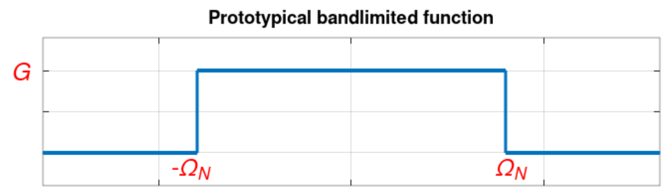
1.2.5 Convolution theorem

$$Y(j\ \Omega) = X(j\Omega)\ H(j\Omega)$$

1.2.6 Bandlimited Functions

$$X(j\ \Omega) = 0 \text{ for } |\Omega| > \Omega_N$$

1. The Fourier Transform of a bandlimited function



$$\Phi(j\ \Omega) = G\ rect(\frac{\Omega}{2\ \Omega_N})$$

2. The Inverse Fourier Transform of a bandlimited function

$$\begin{aligned}\phi(t) &= \frac{1}{2\ \pi}\int_{-\infty}^{\infty} \Phi(j\Omega)e^{j\Omega\ t}d\Omega \\ &= G\frac{\Omega_N}{\pi}sinc(\frac{\Omega_N}{\pi}\ t)\end{aligned}$$

normalization $G = \frac{\pi}{\Omega_n}$

totoal bandwith $\Omega_B = 2\ {}_N$

define $T_s = \frac{2\ \pi}{\Omega_B} = \frac{\pi}{\Omega_N}$

3. The prototypical bandlimited function

$$\begin{aligned}\Phi(j\ \Omega) &= \frac{\pi}{\Omega_N}\ rect(\frac{\Omega}{2\ \Omega_N}) \\ \phi(t) &= sinc(\frac{t}{T_s})\end{aligned}$$

1.2.7 TODO Plot Normalized prototypicale bandlimited function

1.3 Interpolation

Main Task $x[n] \Rightarrow x(t)$

Gaps fill the gaps between samples

1.3.1 Interpolation requirements

- decide on T_s
- make sure $x(nT_s) = x[n]$
- make sure x(t) is smooth

1.3.2 Why smoothness

- jumps (1st order discontinuities) wourl require the signal to move "faster than light"
- 2nd order discontinuities would require infinite acceleration
- the interpolation should be infinitely differentialbe
- "natural" solution: polynominal interpolation

1.3.3 Polynomial interpolation

- N points $\Rightarrow$ polynomial of degree (N-1)
- $p(t) = a_0 + a_1 t + a_2 t^2 a ... + a_{N-1} t^{(N-1)}$
- "naive" approach

$$\begin{cases} p(0) & = x[0] \\ p(T_s) & = x[1] \\ p(2T_s) & = x[2] \\ ...... & \\ p((N-1)T_s) & = x[N-1] \end{cases}$$

Without loss of gnerality:

- consider a symmetric interval $I_N = [-N...N]$
- set $T_s = 1$

$$\begin{cases} p(-N) & = x[-N] \\ p(-N+1) & = x[-N+1] \\ ...... & \\ p(0) & = x[0] \\ p(N) & = x[N] \end{cases}$$

1.3.4 Lagrange interpolation

The natural solution to this interpolation problem is given by Lagrange interpolation

- P_N : space of degree-2N polynominals over I_N
- a basis for P_N is the family of 2N + 1 Lagrange polynominals

$$L_n^{(N)}(t) = \prod_{k=-N}^{N} \frac{t-k}{n-k} \text{ for } n = -N,\ldots,N$$

The formula:

$$p(t) = \sum_{n=-N}^{N} x[n] L_n^{(N)}(t)$$

The Lagrange interpolation is the sought-after polynominal interpolation:

- polynominal of degree 2N through 2N+1 points is unique
- the Lagrangian interpolator satisfies

$$p(N) = x[N] \text{ for } -N \leq n \leq N$$

since

$$L_n^{(N)}(m) = \begin{cases} 1 \text{ if } n = m \\ 0 \text{ if } n \neq m \end{cases} \quad -N \leq n, m \leq N$$

key property maximmally smooth (infinitely many continuous derivatives)

drawback interpolation ”bricks” depend on N

1.3.5 Local interpolation schemes

$$x(t) \sum_{n=-N}^{N} x[n] i_c(t-n)$$

Interpolator’s requirements:

- i_c: interpolation kernel
- $i_c(0) = 1$

- $\mathrm{i_c(t) = 0}$

Key property same interpolating function independently of N.

drawback lack of smoothness

1.3.6 Sinc interpolation formula

$$x(t) \sum_{n=-N}^{N} x[n] sinc(\frac{t - nT_s}{T_s})$$

1.4 Sampling of bandlimited functions

1.4.1 The spectrum of interpolated signals

$$X(j) = \begin{cases} (\pi/\Omega_N) X(e^{j\pi(\Omega/\Omega_N)}) \text{ for } |\Omega| \leq \Omega_N \\ 0 \text{ otherwise} \end{cases}$$

Pich interpolation period $\mathrm{T_s}$:

- $X(j\Omega)$ is $\mathrm{\$_N\$}$-bandlimited, with $\Omega_N = \pi/T_s$
- fast interpolation ($_s$ small) ⇒ wider spectrum
- slow interpolation ($_s$ large) ⇒ narrower spectrum

1.4.2 The space of bandlimited functions

The space $\pi - BL$

- is a a vector space because $\pi - BL \subset L_2(\mathbb{R})$
- inner product is standard inner product in $L_2(\mathbb{R})$
- completeness… that's more delicate

Inner product:

$$\langle x(t), y(t) \rangle = \int_{-\infty}^{\infty} x^*(t) y(t) dt$$

Convolution:

$$(x * y)(t) = \langle x^*(\tau), y(t - \tau) \rangle$$

A basis for the $\pi - BL$ space

$$\phi^{(n)}(t) = sinc(t - n), \text{for} n \in \mathbb{Z}$$

$$FT sinc(t) = rect(\frac{\Omega}{2\pi})$$

$$(sinc * sinc)(m-n) = \begin{cases} 1 \text{ for } m = n \\ 0 \text{ otherwise} \end{cases}$$

1.4.3 The sampling Theorem

Analysis formula:

$$x[n] = \langle sinc(\frac{t - nT_s}{T_s}), x(t) \rangle = T_s x(nT_s)$$

Synthesis formula:

$$x(t) = \frac{1}{T_s} \sum_{n=-\infty}^{\infty} x[n] sinc(\frac{t - nT_s}{T_s})$$

- the space of $\Omega_n - bandlimited$ functions is a Hilbert space
- set $T_s = \pi/\Omega_N$
- the functions $\phi^{(n)}(t) = sinc((t - nT_s)/T_s)$ form a bais for the space
- for any $x(t) \in \Omega_N - BL$ the coefficients in the sinc basis are the (scaled) samples $T_s x(nT_s)$

for any $x(t) \in \Omega_N - BL$, a sufficient representation is the sequence $x[n] = x(nT_s)$

The sampling theorem in Hertz

ans signal x(t) bandlimited to F_N Hz can be sampled with no loss of information using a sampling frequency $F_s \geq 2F_N$ (i.e. sampling period $T_s \leq 1/2\ F_N$

1.5 Sampling of nonbandlimited functions

1.5.1 Raw Sampling

Raw sampling is when we don't care about first taking the inner product with the sinc function. So we just take x(t) and every T_s seconds, we take a sample.

x(n): The continous-time complex exponential

1.5.2 Aliasing

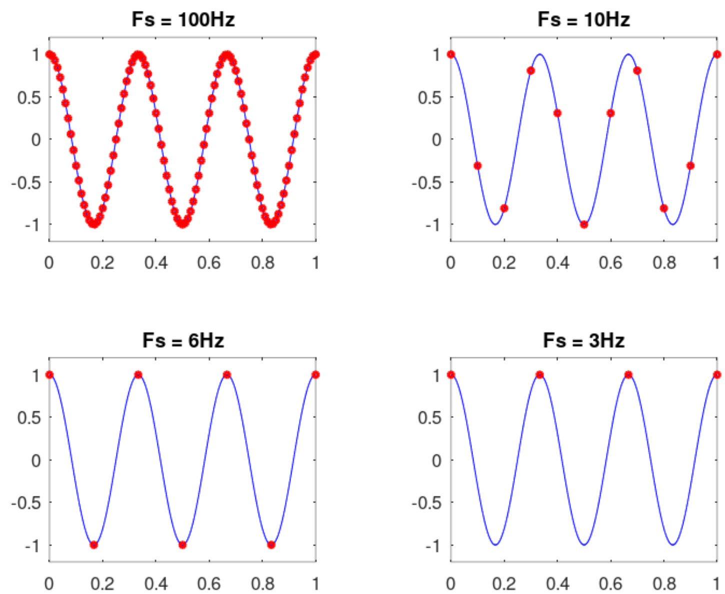
$$x(t) = cos(2_o t)$$
$$x[n] = x(nT_s) = cos(_0 n)$$

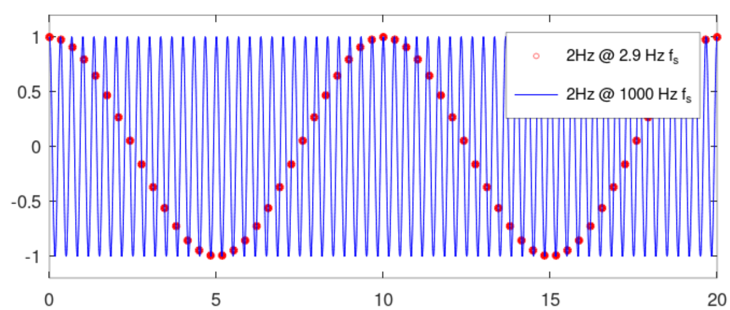
with

$$F_s = \frac{1}{T_s}$$
$$\omega_o = 2\pi(\frac{F_0}{F_s})$$

Table 3: Aliasing

sampling period	digital frequency	$\{\hat{x}\}$
$T_s < \pi/\Omega_0$	$0 < \omega_o < \pi$	e^{j_0}
$\pi/\Omega_0 < T_s < 2\pi/Omega_0$	$\pi < _0 < 2\pi$	e^{j_1}: $\Omega_1 = \Omega_0 - 2\pi/T_s$
$T_s > 2\pi/\Omega_0$	$_0 > 2\pi$	e^{j_2}: $\Omega_2 = \Omega_0 mod(2\pi/T_s)$





1.5.3 Aliasing for arbitrary spectra

A contiuous time signal x_c sampled every T_s seconds gives a sequence x[n]. Which is equal to the contious time signals at multiples of the sampling intervals T_s.

- $x_c(t) \Rightarrow x[n] = x_c(nT_s)$

In Fourier Transform domain we have a spectra of the continuous time signal $X_c(j\Omega)$. And at the output we have a discrete time Fourier Transform of the sequence $X(j\omega)$. What is that going to be in genaral? And how is it going to be related to the input spectrum?

- $X(j\Omega) \Rightarrow X(j\omega) = ?$

The key idea:

- pick T_s and set $\Omega_N = \pi/T_s$
- pick $\Omega_= < \Omega_N$

$$
\begin{aligned}
e^{j\Omega_0 t} &\to e^{j\Omega_0 T_s n} \\
e^{j(\Omega_0+2\Omega_N)t} &\to e^{j(\Omega_0+2\Omega_N)T_s n}, \text{ add } 2\Omega_N \\
e^{j(\Omega_0+2\Omega_N)t} &\to e^{j(\Omega_0 T_s n+2\Omega_N T_s n)}, \text{ expand this product} \\
e^{j(\Omega_0+2\Omega_N)t} &\to e^{j(\Omega_0 T_s n+\frac{2\pi}{T_s} T_s n)} \\
e^{j(\Omega_0+2\Omega_N)t} &\to e^{j(\Omega_0 T_s n+2)}, \; e^{j2} \text{ is equal to one} \\
e^{j(\Omega_0+2\Omega_N)t} &\to e^{j\Omega_0 T_s n}, \text{ the same discrete time sequence as before}
\end{aligned}
$$

So we do not see the higher frequency complex exponential, it simply looks like the lower frequency exponential Ω_0.

> So in general, if we have two frequencies sampled, the higher frequency is aliased back onto the lower frequency and we simply see the sum of these two.

1. Spectrum of raw-sampled signals

   - start with the inverse Fourier Transform

   $$x[n] = x_c(nT_s) = \frac{1}{2\pi}\int_{-\infty}^{\infty} X_c(j\Omega)e^{j\Omega n T_s}d\Omega$$

   - frequencies $2\Omega_N$ apart will be aliased, so split the integration interval

   $$x[n] = \frac{1}{2\pi}\sum_{k=-\infty}^{\infty}\int_{(2k-1)\Omega_N}^{(2k+1)\Omega_N} X_c(j\Omega)e^{j\Omega n T_s}d\Omega$$

   - with a change of variable and using $e^{j(\Omega+2k\Omega_N)T_s n} = e^{j\Omega T_s n}$

$$x[n] = \frac{1}{2\pi} \sum_{k=-\infty}^{\infty} \int_{-\Omega T_s n}^{\Omega T_s n} X_c(j(\Omega - 2k\Omega_N) e^{j\Omega n T_s} d\Omega$$

- interchange summation and integral

$$x[n] = \frac{1}{2\pi} \int_{-\Omega T_s n}^{\Omega T_s n} \left[\sum_{k=-\infty}^{\infty} X_c(j(\Omega - 2k\Omega_N) \right] e^{j\Omega n T_s} d\Omega$$

- periodization of the spectrum; define

$$X_c(j\Omega) = \sum_{k=-\infty}^{\infty} X_c(j(\Omega - 2k\Omega_N)$$

- so that

$$x[n] = \frac{1}{2\pi} \int_{-\Omega T_s n}^{\Omega T_s n} X_x(j\Omega) e^{j\Omega n T_s} d\Omega$$

- set $\omega = \Omega T_s$

$$\begin{aligned} x[n] &= \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{1}{T_s} X_c(j\frac{\omega}{T_s}) e^{j\omega n} d\omega \\ &= IDTFT \ \frac{1}{T_s} X_c(j\frac{\omega}{T_s}) \\ X(e^{j\omega}) &= \frac{1}{T_s} \sum_{k=-\infty}^{\infty} X_c\left(j\frac{\omega}{T_s} - j\frac{2\pi k}{T_s}\right) \end{aligned}$$

$$X(e^{j\omega}) = \frac{1}{T_s} \sum_{k=-\infty}^{\infty} X_c\left(j\frac{\omega}{T_s} - j\frac{2\pi k}{T_s}\right)$$

2. Sampling strategies

   given a sampling period T_s

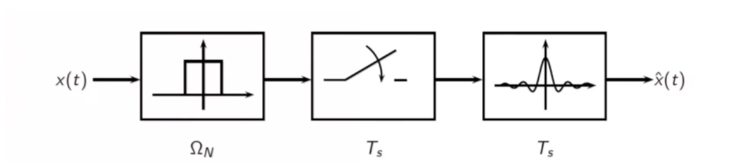
   - if the signal is bandlimited to π/T_s or less, raw sampling is fine i.e. equivalent to sinc sampling up to scaling factor T_s.
   - if the signal is not bandlimited, two choices:

  - bandlimit via lowpass filter in the *continuous-time domain* before sampling i.e. sinc sampling
  - or raw sample the signal an incur aliasing
- aliasing sounds horrible, so usualle we choose to bandlimit in continuous time

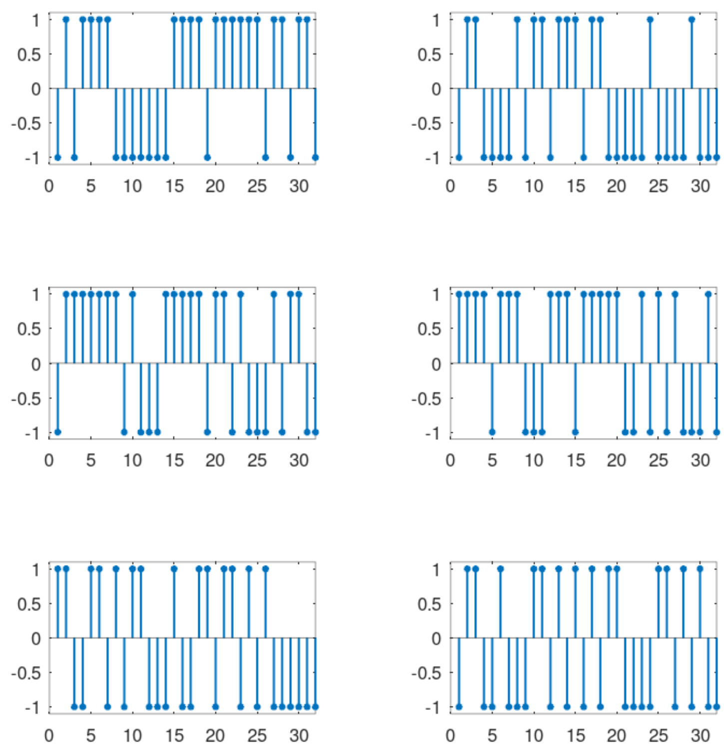
(a) Sinc Sampling and Interpolation

$$\hat{X}[n] = \langle sinc\left(\frac{t - nT_s}{T_s}\right), x(t)\rangle = (sinc_{Ts} * x)(nT_s)$$

$$\hat{X}[n] = \sum_n x[n]\ sinc\left(\frac{t - nT_s}{T_s}\right)$$



1.6 Stochastic signal processing



1.6.1 Averaging

- when faced with random data an intuitive response is to take "averages"
- in probability theory the average is across realizations and it's called

  for the coin-toss signal $E[x[n]] = -1 \times P[n - thtossistail] + 1 \times P[n - thtossishead] = 0$
- so the average value for each sample is zero....

1.6.2 Energy and power

- the coin-toss signal has infinite energy

$$E_x = \sum_{k=-\infty}^{\infty} |x[n]|^2 = \lim_{N\to\infty} = \infty$$

- however it has finite power over any interval:

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

1.6.3 Power spectral density

$$P[k] = E\left[|X_N[k]|^2 / N\right]$$

- it looks very much as if è[k] = 1
- if $|X_N[k]|^2$ tends to the *energy* distribution in frequnec....
- ... $|X_N[k]|^2 / N$ tends to the *power* distributi8on (aka density) in frequency
- the frequency-domain representation for stochastic processes is the power spectral density

1.6.4 Summary

- a stochastic process is characterized by its power spectral density (PSD)
- it can be shown (see text book) that the PSD is

  $$P_x(e^{j\omega}) = DTFT r_x[n]$$

  where $r_x[n] = E\left[x[k]x[n+k]\right]$ is the autocorrelation of the process
- for a filtered stochastic process $y[n] = \mathfrak{H}x[n]$, it is:

  $$P_y(e^{j\omega}) = |H(e^{j\omega}|^2 P_x(e^{j\omega})$$

1.6.5 White noise

- ”white” indicates uncorrelated samples
- $r_w[n] = \sigma^2\delta[n]$
- $\mathrm{P_w}(\mathrm{e}^{\mathrm{j}\omega}) = \sigma^2$
- the PSD is independent of the probability distribution of the signal samples (depnds only on the variance)
- distribution is important to estimate bounds for the signal
- very often a Gaussian distribution models the expermiental data the best
- AWGN: additive white Gaussion noise

1.6.6 Quantization