Lecture 9: Sampling

So far we have considered working with signals in continuous time, this mean that for each value of *t*, there exist an associated value for the signal $x(t)$. However, if we are using a computer to process our signals, it is not feasible to acquire and store the information of the signal for each value of *t*, we will need a processor with a clock that has an infinite frequency, and we will also need infinite memory. However, it is not needed to store all the data points from the signal $x(t)$ in order to obtain the information that is relevant for our purposes.

For this section we used images extracted from wikipedia and the book Discrete Time signal Processing from Allan B. Oppenheim and Shaffer.

Sampling of Signals in Continuous Time

Let's suppouse we have a real life signal in continuous time, $x(t)$, and we would like to analyse its behaviour. In order to do so, we will need to measure its values. This measurement can be seen as taken a sample of the signal value at a specific time. Since we cannot measure for all the infinite time values in a particular window of measurement, then we are just gonna select some values within that time interval. The time of measurements can be arbitrary or can be periodic. We will consider in this course only periodic sampling. Mathematically, periodic sampling from a continuous time signal generates a sequence of values, such that:

$$x[n] = x(nT_s), \qquad -\infty < n < \infty,$$

where T_s is called the sampling period, and it is reciprocal to f_s, i.e. the sampling frequency, which have units of *samples/second*. In the same way we can represent this frequency in radians/second, in order to differienciate the frequency in continous time signals, from the one for discrete time signals, let's use the capital omega symbol, i.e. $\Omega_s = 2\pi f_s = 2\pi / T_s$.

Practically, the sampling of a signals is done by an Analog to Digital Converter, which is an electronic component that is in charge of digitalizing the signals. Digitalizing a signal involves the sampling and quantization processes. Quantization is the operation that converts the value of a signal in a digital value.

Sampling is not an invertable process, since the same sequence of numbers can produce and infinite number of signals in continious time. However, if we impose some conditions to the continuous time signals, and to the "sampler" we can remove the ambiguity.

Before defining especifically what is sampling, let´s talk a bit more in detail about the function Delta de Dirac. Mathematically this functions is defined as zero in all the values of time, except in zero, where its value is infinite, however, its area is 1. To understand where this signal is comming from, let´s define a pulse with amplitude $1/T$, and defined as zero everywhere except in the time interval between $[-T/2, T/2]$. Thearea for this pulse is one, no matter what is the value for *T*, if we apply th elimit when $T \to 0$, we see that the area is preserved, but the interval where the signal is defined tends to zero, and its amplitude tends to infinite. This is how the Delta de Diract function was built.

Periodic sampling can be considered a process that involves the modulation of a continuos time signal $x_c(t)$, by a periodic trains of impulses $s(t)$, such that we generate a sampled continuous signal $x_s(t) = x_c(t)s(t)$. This sampled continuous signal is just a train of impulses that is different from zero in some particular values of time, which are T_s seconds apart from each other. Once we get this signal, then we define a sequence which is equal to the values of the signal $x_s(t)$, evaluated for multiple values of T_s. This process is expresed in the figures below.

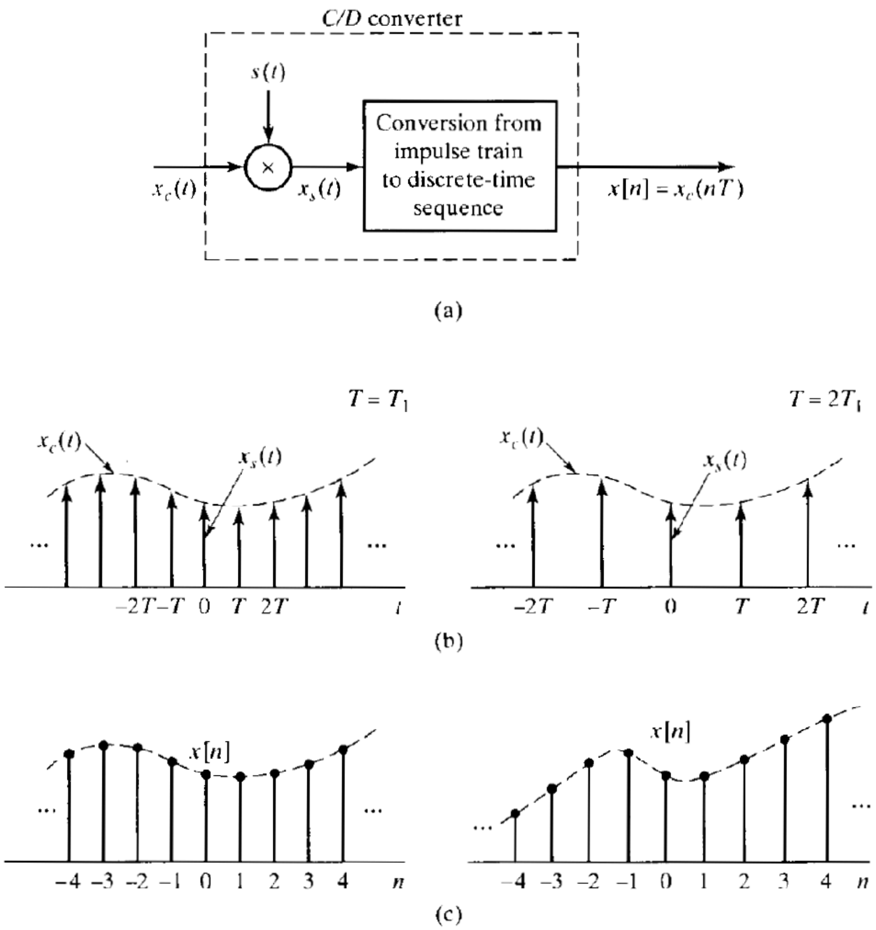


Figure 4.2 Sampling with a periodic impulse train followed by conversion to a discrete-time sequence. (a) Overall system. (b) $x_s(t)$ for two sampling rates. (c) The output sequence for the two different sampling rates.

Frequency Domain Representation of Sampling

To define what is the effect of sampling in the frequency domain, let's represent the sampling operator mathematically. Let's define the impulse train that modulates the continious time signal as follows:

$$s(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT_s)$$

where $n \in \mathbb{Z}$. Therefore the sampled continious signal will be:

$$x_s(t) = x_c(t)s(t) = x_c(t)\sum_{n=-\infty}^{\infty}\delta(t - nT_s),$$

which is equivalent to:

$$x_s(t) = \sum_{n=-\infty}^{\infty} x(nT_s)\delta(t - nT_s).$$

Now, lets consider the Fourier transform of the signal $x_s(t)$, since it is the product of two signals, its impulse response will be the convolution of the Fourier transform of $x_c(t)$ and $s(t)$. Let's start with the periodic train of impulses, but before this let´s talk about the Fourier transform for Periodic signals.

Fourier Transform for Periodic Signals

Fourier transorm is originally defined for aperiodic signals, however, we can also define it for periodic signals, giving a common framework to work with signals in general. Let's consider a signal $x(t)$ with a Fourier transform $X(j\omega) = 2\pi\delta(\omega - \omega_0)$, to find $x(t)$ let's apply the inverse transform:

$$x(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} 2\pi\delta(\omega - \omega_0)e^{j\omega t}d\omega = e^{j\omega_0 t}.$$

More generally, if $X(j\omega)$ has the shape of a train of impulses equally spaced, that is:

$$X(j\omega) = \sum_{k=-\infty}^{\infty} 2\pi a_k\delta(\omega - k\omega_0),$$

then

$$x(t) = \sum_{k=-\infty}^{\infty} a_k e^{jk\omega_0 t},$$

which is exactly the Fourier series for a periodic signal $x(t)$. Therefore, the Fourier transform for a periodic signal can be interpreated as the a train of impulses ocurring at frequency harmonically related to the fundamental frequency ω_0, for which the area of the impulse at the *k*th armonically related frequency $k\omega_0$ is 2π times the *k*th Fourier series coefficient a_k.

Therefore, in order to compute the Fourier transform of the periodic train of pulses, we compute first their Fourier series coefficients:

$$a_k = \frac{1}{T_s}\int_{T_s}\delta(t)e^{-jk\omega_0 t}dt = \frac{1}{T_s}.$$

Now we compute the Fourier transform:

$$\mathfrak{F}[s(t)] = S(j\omega) = \int_{-\infty}^{\infty}\left[\sum_{n=-\infty}^{\infty}\delta(t-nT_s)\right]e^{-j\omega t}dt = \frac{2\pi}{T_s}\sum_{k=-\infty}^{\infty}\delta\left(\omega-\frac{2\pi k}{T_s}\right).$$

The Fourier transform of a periodic impulse train in time domain, with period T_s, is a periodic impulse train in the frequency domain with period $2\pi/T_s$. Since we will work with signals in discrete time, let's use Ω, then:

$$S(j\Omega) = \frac{2\pi}{T_s}\sum_{k=-\infty}^{\infty}\delta(\Omega-k\Omega_s).$$

Then, the Fourier transform for $X_s(j\Omega) = \frac{1}{2\pi}X_c(j\Omega) * S(j\Omega)$:

$$X_s(j\Omega) = \frac{1}{T_s}\int_{-\infty}^{\infty}X_c(j\Theta)\sum_{k=-\infty}^{\infty}\delta(\Omega-k\Omega_s-\Theta)d\Theta,$$

$$X_s(j\Omega) = \frac{1}{T_s}\sum_{k=-\infty}^{\infty}X_c(j(\Omega-k\Omega_s)).$$

This result can be interpreated as follows: the Fourier transform of a continious sampled signal consist on periodically repeated copies of the Fourier transform of the continious signal $x_c(t)$. These copies of $X_c(j\Omega)$ are shifted by integer multiples of the sampling frequency, and then they are summed in order to produce the Fourier transform of the impulse train of samples.

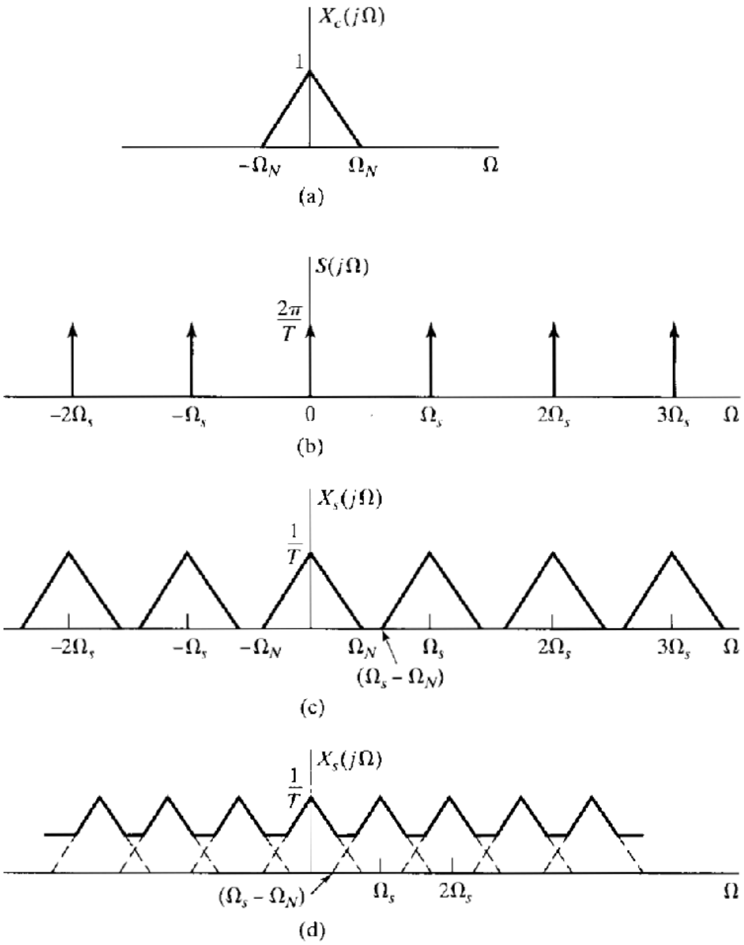


Figure 4.3 Effect in the frequency domain of sampling in the time domain. (a) Spectrum of the original signal. (b) Spectrum of the sampling function. (c) Spectrum of the sampled signal with $\Omega_s > 2\Omega_N$. (d) Spectrum of the sampled signal with $\Omega_s < 2\Omega_N$.

Now, let's consider that the signal $x_c(t)$ has a bandlimited spectrum up to a frequency Ω_N, if $\Omega_s - \Omega_N > \Omega_N$, or, $\Omega_s > 2\Omega_N$, then, when added, the spectrums are not overlap, and the replicas of $X_c(j\Omega)$ remain intact in the sampled signal (with a scale factor $1/T_s$). The maximum frequency in the signal that complies with this relation is called the *Nyquist Frequency*. The Nyquist Frequency can be seen as the maximum frequency that is present in the signal and does not produce aliasing. If the signal is sampled with sampling frequency lower or equal than twice the Nyquist Frequency, then the spectra is overlapped and the resulting spectrum does not resemble the spectrum for the original signal. This phenomena is called *aliasing,* when it occurs, the continious time signal cannot be recovered from the samples, since now high frequency components are mixed with low frequency components, i.e. the spectrum is distorted. To avoid this the signals should always be low-pass filtered before sampling, and the cut-off frequency of the filter should be at least 1/2 of the sampling frequency.

To recover the original signal from the sampled signal, if $\Omega_s > 2\Omega_N$, we can use an ideal low pass filter as shown in the next figure:

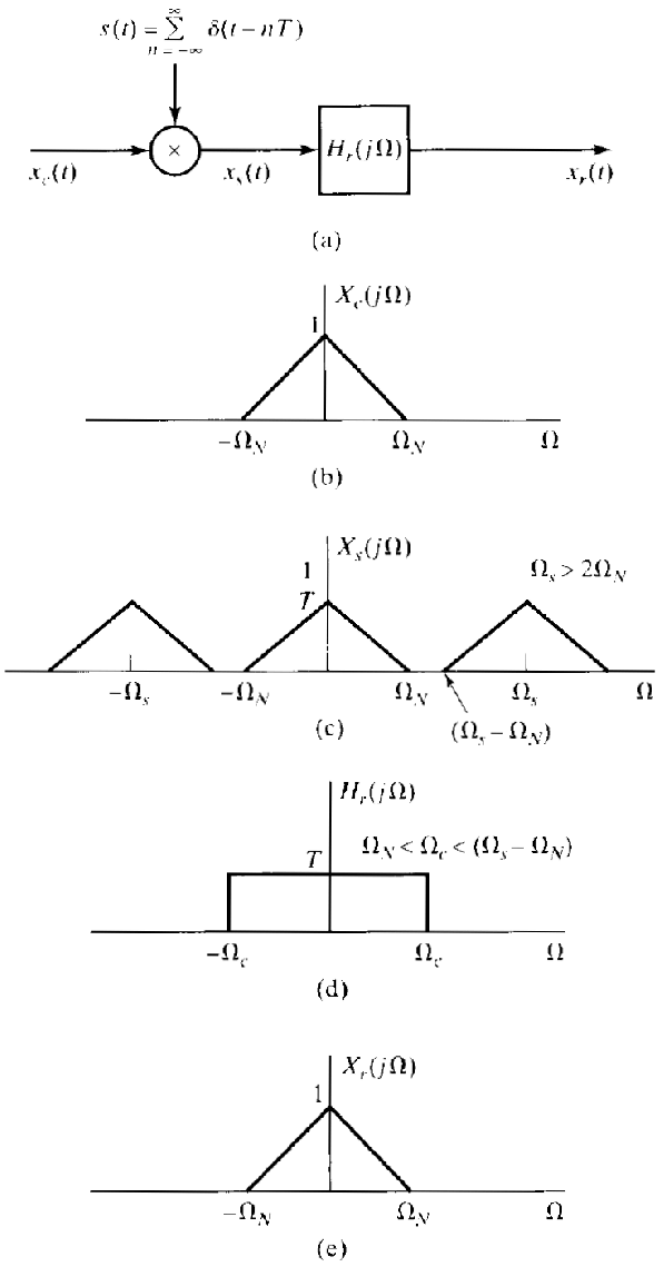
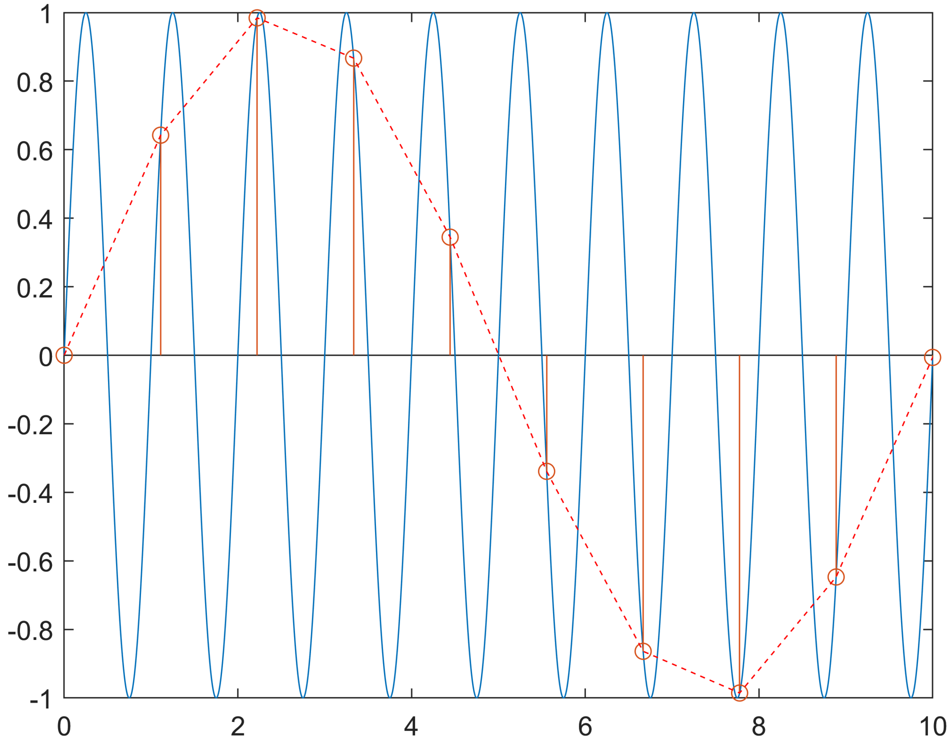


Figure 4.4 Exact recovery of a continuous-time signal from its samples using an ideal lowpass filter.

Let´s simulate some aliasing effects. Lets consider the continious signal to be a discrete signal with a large sampling rate.

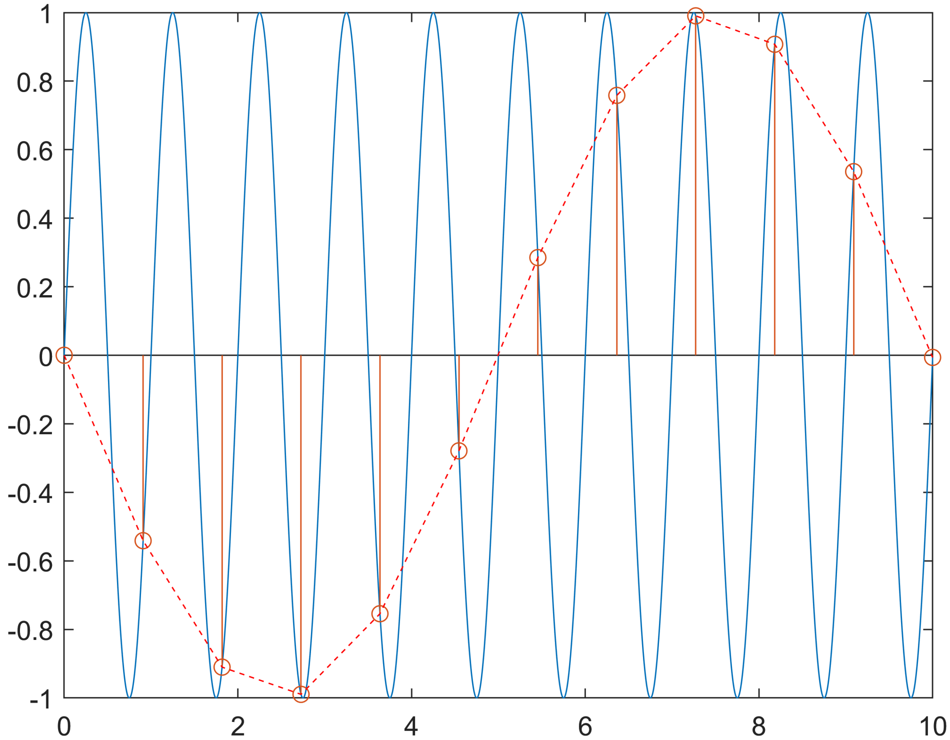
```
F_s = 0.9;
F_si=1000;
t = 0:1/F_si:10;
x = sin(2*pi*1*t);

figure, plot(t,x)
hold on
stem(t(1:fix(F_si/F_s):end),x(1:fix(F_si/F_s):end),'o')
hold on
plot(t(1:fix(F_si/F_s):end),x(1:fix(F_si/F_s):end),'r--')
```




```
F_s = 1.1;
F_si=1000;
t = 0:1/F_si:10;
x = sin(2*pi*1*t);

figure, plot(t,x)
hold on
stem(t(1:fix(F_si/F_s):end),x(1:fix(F_si/F_s):end),'o')
hold on
plot(t(1:fix(F_si/F_s):end),x(1:fix(F_si/F_s):end),'r--')
```



Examples of Aliasing in Real Life

The aliasing effect in real life.

- Wagon-Wheel I
- Wagon-Wheel Effect II
- Wagon-Wheel Effect III
- Stroboscopic Effect I
- Stroboscopic Effect II
- A Little Related

In images

The Nyquist-Shannon Sampling Theorem

The Nyquist-Shannon Theorem for sampling states that:

If a function $x(t)$ *contains no frequencies higher than* $\Omega_N = 2\pi f_N$, *it is completely determined by giving its ordinates at a series of points spaced* $T_s = \frac{\pi}{\Omega_N}$ *seconds apart.*

Or in other words, the sampling frequency of the signal should be $f_s > \frac{\Omega_N}{\pi}$.

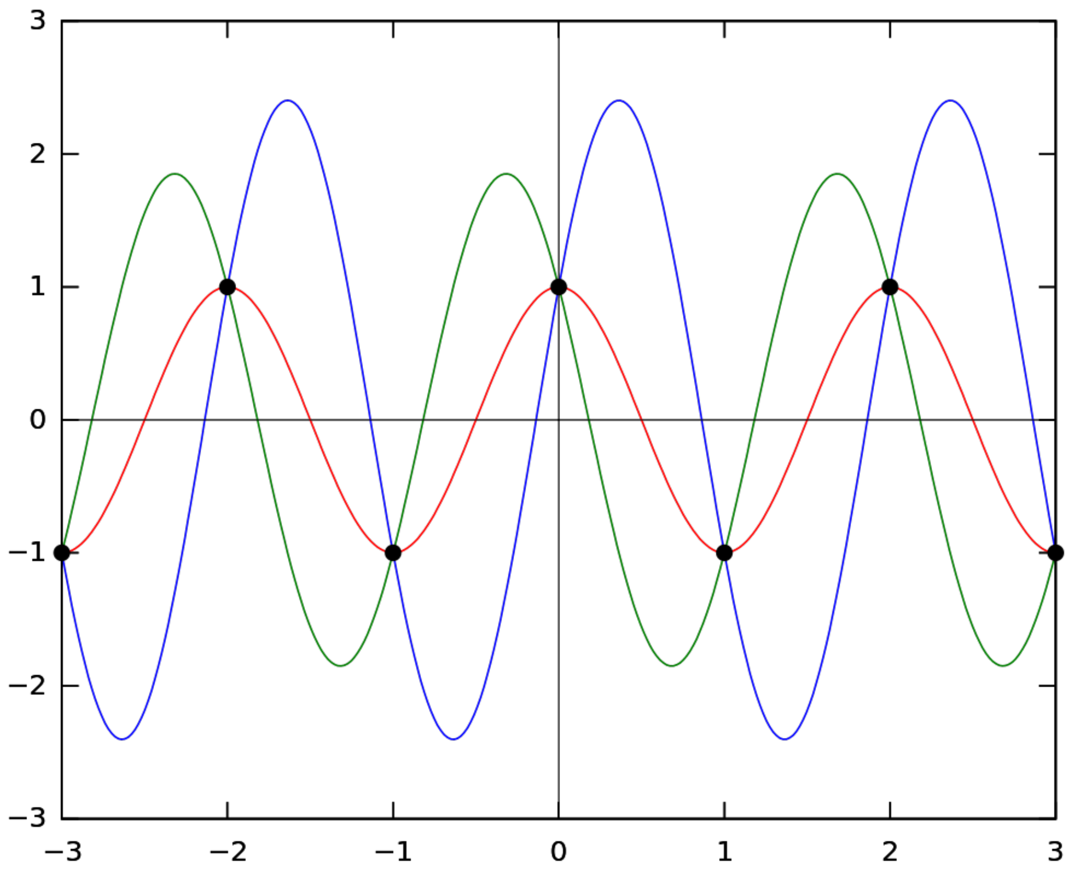
What happens if we are just in the limit? Let's consider the signal

$$x(t) = \frac{\cos(\Omega_N t + \theta)}{cos(\theta)} = \cos(\Omega_N t) - \sin(\Omega_N t)\tan(\theta),$$

if $f_s = \frac{\Omega_N}{\pi}$, or equivalent to a sampling Period of $T_s = \frac{\pi}{\Omega_N}$. Then we get:

$$x(nT) = \cos(n\pi) - \sin(n\pi)\tan(\theta) = (-1)^n,$$

regardless the value of θ we get the same samples. Due to this ambiguity, it is needed that $f_s > 2f_N$, with emphasis in the strictly larger than. For an example of this effect check the figure below.



How to Avoid Aliasing

In order to avoid aliasing the signal in continuous time, or in discrete time with a large f_s, should be low-pass filtered, such that the larger frequency after filtering complies with the Nyquist-Shannong sampling theorem. The low -pass filter used for this purposes is called an *antialiasing filter.*

Quantization

We will not focus much on quantization, however it is a concept that we should be aware of. When representing signals in a computer, we cannot asing a continious of values to each point in the signal, instead, since they are represented by a binary number, we have to assign a binary code that is closest to a given number. Since these codes are also finite, this means that several numbers in the signal will be assign the same binary code. An example of a quantizied signal is given in the figure below

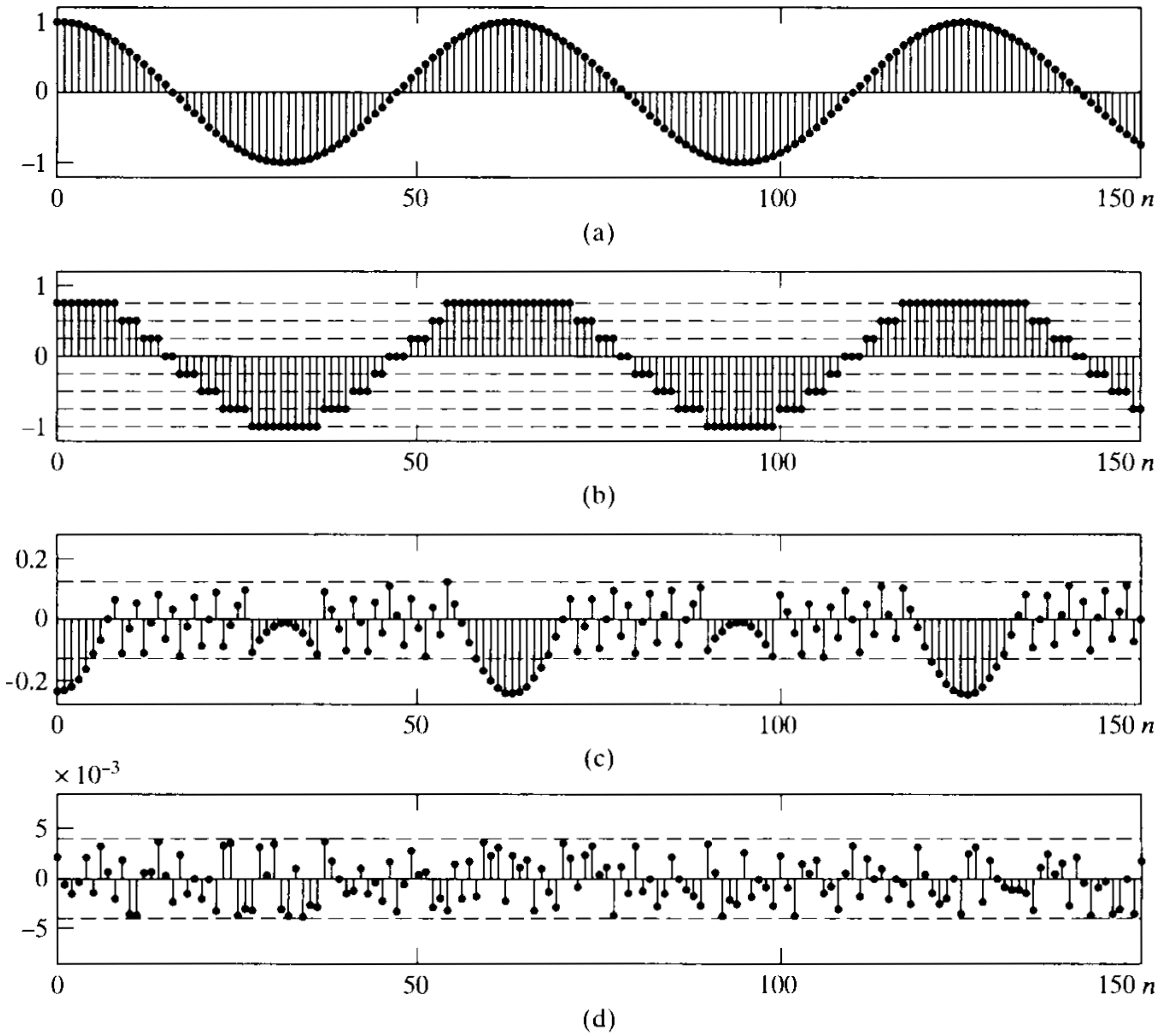


Figure 4.51 Example of quantization noise. (a) Unquantized samples of the signal $x[n] = 0.99\cos(n/10)$. (b) Quantized samples of the cosine waveform in part (a) with a 3-bit quantizer. (c) Quantization error sequence for 3-bit quantization of the signal in (a). (d) Quantization error sequence for 8-bit quantization of the signal in (a).

The Discrete Fourier Transform (DFT)

So far, we have used the Fourier transform in order to define the spectrum ogf Periodics, or a periodic signals. However, this transform is a continious function of the frequency variable ω. However, just in the same way we did with signals in the time domain, we can sample the Frequnecy domain and obtained a sequence of values that approximate the spectrum, such that we do not need to store all the values for the Fourier transform. These values will be equally spaced in frequency from each other, and their amplitude will correspond to samples from the Fourier Transform. This sampled version of the Fourier Transform is called Discrete Fourier

Transform (DFT). This sequence is important cause there exist several efficient algorithms that can be used for its computation, which will allow teh computational implementation of the theory for Signal Processing.

Before addressing the DFT let´s talk about the Discrete Fourier Series.

Representation of Periodic Sequences: The Discrete Fourier Series

Considere a sequence that is periodic $\tilde{x}[n]$, with period *N*, i.e. $\tilde{x}[n] = \tilde{x}[n + rN]$, for $r, n \in \mathbb{Z}$. Similarly to signals in continuous time, this periodic sequence can be represented as a sum of harminically related complex exponentials. These periodic exponentials willbe of the form: $e_k[n] = e^{j(2\pi/N)kn} = e_k[n + rN]$, whith *k* and integer, therefore the Fourier representation will be given by:

$$\tilde{x}[n] = \frac{1}{N}\sum_{k} \tilde{X}[k]e^{j(2\pi/N)kn}.$$

We can see that for a continious time signal we require infinitely harmonically related complex exponentials to represent the signal, whereas for discrete signals with period *N* we only require *N* harmonically related complex exponentials. This is because in general:

$$e_{k+\ell N}[n] = e^{j(2\pi/N)(k+\ell N)n} = e^{j(2\pi/N)kn}e^{j2\pi\ell N} = e^{j(2\pi/N)kn} = e_k[n],$$

where $\ell \in \mathbb{Z}$. Therefore, we can redefine the Fourier series for a periodic sequence with period *N* as:

$$\tilde{x}[n] = \frac{1}{N}\sum_{k=0}^{N-1} \tilde{X}[k]e^{j(2\pi/N)kn}.$$

To obtain the Fourier coefficients we wxploit orthogonality of complex exponentials, so we multiply by $e^{-j(2\pi/N)rn}$, and sum between zero and $N - 1$, so we get

$$\sum_{n=0}^{N-1} \tilde{x}[n]e^{-j(2\pi/N)rn} = \sum_{n=0}^{N-1}\frac{1}{N}\sum_{k=0}^{N-1} \tilde{X}[k]e^{j(2\pi/N)(k-r)n},$$

which leads to

$$\sum_{n=0}^{N-1} \tilde{x}[n]e^{-j(2\pi/N)rn} = \sum_{k=0}^{N-1} \tilde{X}[k]\left[\frac{1}{N}\sum_{n=0}^{N-1} e^{j(2\pi/N)(k-r)n}\right].$$

Since,

$$\frac{1}{N}\sum_{n=0}^{N-1} e^{j(2\pi/N)(k-r)n} = \begin{cases} 1, & k - r = mN, \quad m \text{ an integer}, \\ 0, & \text{otherwise}. \end{cases}$$

then:

$$\sum_{n=0}^{N-1} \tilde{x}[n]e^{-j(2\pi/N)rn} = \tilde{X}[r].$$

Then the Fouriere series coefficients $\tilde{X}[k]$, cna be found by:

$$\widetilde{X}[k] = \sum_{n=0}^{N-1} \widetilde{x}[n] e^{-j(2\pi/N)kn}.$$

Note que la sequencia obtenida $\widetilde{X}[k]$ es periodica, con periodo *N*

$$\widetilde{X}[k+N] = \sum_{n=0}^{N-1} \widetilde{x}[n] e^{-j(2\pi/N)(k+N)n} = \left(\sum_{n=0}^{N-1} \widetilde{x}[n] e^{-j(2\pi/N)kn} \right) e^{-j2\pi n} = \widetilde{X}[k]$$, for any *k*.

For convinience, let´s define the complex exponentials as $W_N = e^{-j(2\pi/N)}$, with this notation the Discrete fourier Series Equations become:

Analysis Equation: $\widetilde{X}[k] = \sum_{n=0}^{N-1} \widetilde{x}[n] W_n^{kn}$,

Synthesis equation: $\widetilde{x}[n] = \frac{1}{N} \sum_{k=0}^{N-1} \widetilde{X}[k] W_N^{-kn}$.

In both cases $\widetilde{x}[n]$ y $\widetilde{X}[k]$ son secuencias periodicas.