Sampling

Alejandro Ribeiro

February 10, 2021

Signals exist in continuous time but it is not unusual for us to process them in discrete time. When we work in discrete time we say that we are doing *discrete signal processing,* something that is convenient due to the relative ease and lower cost of using computers to manipulate signals. When we use discrete time representations of continuous time signals we need to implement processes to move back and forth between continuous and discrete time. The process of obtaining a discrete time signal from a continuous time signal is called sampling. The process of recovering a continuous time signal from its discrete time samples is called signal reconstruction or interpolation.

Mathematically, the sampling process has an elementary description. Given a sampling time T_s and a continuous time signal x with values $x(t)$, the sampled signal x_s is one that take values

$$x_s(n) = x(nT_s), \quad n \in \mathbb{Z}. \tag{1}$$

As per (1), the sampled signal retains values at regular intervals spaced by T_s and discards the remaining values of $x(t)$—see Figure 1. The process by which this is done in, say, a sound card, is a problem of circuit design. For our purposes, let us just say that (1) is a reasonable model for the transformation of a continuous time signal into a discrete time signal.

A relevant question, perhaps the most relevant question, is what information is lost when discarding all the values of $x(t)$ except for those at times nT_s. To answer this question, we compare the spectral representations of x_s and x. In fact, since x_s is a discrete time signal and x is a continuous time signal it is convenient to introduce a continuous time representation of the sampled signal as we describe in the following section.

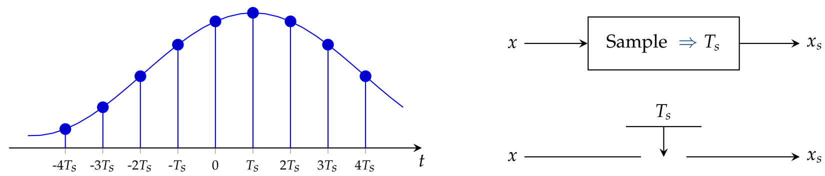

Figure 1. Sampling with sampling time T_s. Sampling the continuous time signal x to create the discrete time signal x_s entails retaining the values $x_s(n) = x(nT_s)$. A relevant question is in what respect the sampled signal $x_s(n)$ differs from the original signal $x(t)$.

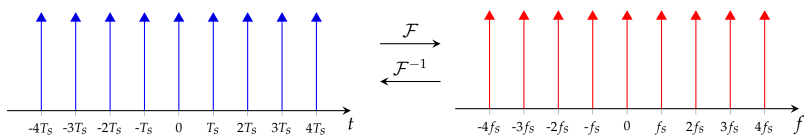

Figure 2. A Dirac train with spacing T_s (left). The Fourier transform of the Dirac train is another Dirac train with spacing $f_s = 1/T_s$.

1 Dirac train representation of sampled signals

A Dirac train, or Dirac comb, with spacing T_s is a signal x_c defined by a succession of delta functions located at positions nT_s (Figure 2):

$$x_c(t) = T_s \sum_{n=-\infty}^{\infty} \delta(t - nT_s). \tag{2}$$

A Dirac train is, in a sense, a trick to write down a discrete time signal in continuous time. The train is formally defined to be a continuous time signal but it becomes "relevant" only at the (discrete) set of times nT_s. In our forthcoming discussions of sampling, we use the Fourier transform of the Dirac comb. This transform can be seen to be another Dirac comb, but with spacing $f_s = 1/T_s$. I.e., if we denote the Fourier transform of x_c as $X_c = \mathcal{F}(x_c)$ we have that

$$X_c(f) = \sum_{k=-\infty}^{\infty} \delta(f - kf_s). \tag{3}$$

That $X_c(f)$ does represent the values of the Fourier transform of x_c is not difficult to show by identifying x_c with the discrete time constant signal $x(t) = 1$, but we don't show this derivation on these notes.

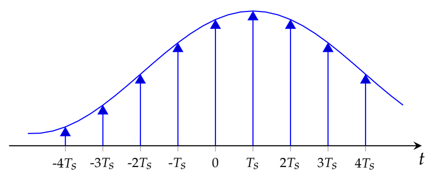

Figure 3. Representation of sampled signal with a modulated Dirac train. The representation is equivalent to the one in Figure 1 but makes comparisons with the original signal x easier.

In the Dirac train representation of sampling we use the samples $x_s(n) = x(nT_s)$ to modulate the deltas of a Dirac train. Specifically, we define the signal x_δ as (Figure 3)

$$x_\delta(t) = T_s \sum_{n=-\infty}^{\infty} x(nT_s)\delta(t - nT_s). \tag{4}$$

That (1) and (4) are equivalent representations of sampling follows from the simple observation that when given the value $x_s(n)$ we can determine $x_\delta(nT_s)$ and vice versa. The representation in (1) is simpler, but the representation in (4) allows comparisons with the original signal x.

Indeed, the sampling representation in (4) can be written as the product between $x(t)$ and the Dirac train in (2):

$$x_\delta(t) = x(t) \times \left[T_s \sum_{n=-\infty}^{\infty} \delta(t - nT_s) \right]. \tag{5}$$

Note that the expressions in (4) and (5) are equivalent since in the multiplication of the function $x(t)$ with the shifted delta function $\delta(t - nT_s)$, only the value $x(nT_s)$ is relevant. It is therefore equivalent to simply multiply $\delta(t - nT_s)$ by $x(nT_s)$.

Straightforward as it is, rewriting (4) as (5) allows us to rapidly characterize the spectrum of the sampled signal $x_\delta(t)$. Since we know that multiplication in time is equivalent to convolution in frequency we have that the Fourier transform $X_\delta = \mathcal{F}(x_\delta)$ can be written in terms of the Fourier transforms $X = \mathcal{F}(x)$ of x and the Dirac train as

$$X_\delta = X * \mathcal{F}\left[T_s \sum_{n=-\infty}^{\infty} \delta(t - nT_s) \right]. \tag{6}$$

The Fourier transform of the Dirac train $x_c(t) = T_s \sum_{n=-\infty}^{\infty} \delta(t - nT_s)$ we have seen is given by the Dirac train in (3). Using this result in (6) and

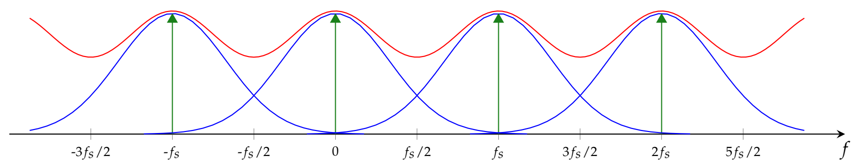

Figure 4. Spectrum $X_\delta = \mathcal{F}(x_\delta)$ of the sampled signal x_δ. The spectrum of the original signal (in blue) is copied and shifted to all the frequencies that are integer multiples of f_s (in green). The spectrum X_δ (in red) is the sum of all these shifted copies.

the linearity of the convolution operation we further conclude that

$$X_\delta = \sum_{k=-\infty}^{\infty} X * \delta(f - kf_s). \tag{7}$$

A final simplification come from observing that the convolution of X with the shifted delta function $\delta(f - kf_s)$ is just a shifting of the spectrum of X so that it is re-centered at $f = f_s$. We can therefore write

$$X_\delta(f) = \sum_{k=-\infty}^{\infty} X(f - kf_s). \tag{8}$$

The result in (8) is sufficiently important so as to deserve a summary in the form of a Theorem that we formally state next.

Theorem 1 *Consider a signal x with Fourier transform $X = \mathcal{F}(x)$, a sampling time T_s, and the corresponding sampled signal x_δ as defined in (4). The spectrum $X_\delta = \mathcal{F}(x_\delta)$ of the sampled signal x_δ is a sum of shifted versions of the original spectrum*

$$X_\delta(f) = \sum_{k=-\infty}^{\infty} X(f - kf_s).$$

The result in Theorem 1 is explained in terms of what we call spectrum periodization. We start from the spectrum X of the continuous time signal that we replicate and shift to each of the frequencies that are multiples of the sampling frequency f_s. The spectrum X_δ of the sampled signal is given by the sum of all these shifted copies—see Figure 4.

The result of spectrum periodization provides a very clear answer to the question of what information is lost when we sample a signal at frequency f_s. The answer is that whatever information is contained by frequency components $X(f)$ outside of the set $f \in [-f_s/2, f_s/2]$ is completely lost. Information contained at frequencies f close to the borders of this set are not completely lost but rather distorted by their mixing with the frequency components outside of the set $f \in [-f_s/2, f_s/2]$. We refer to this distortion phenomenon as aliasing.

The result in Theorem 1 points out to a particularly interesting result for the case of bandlimited signals that you are asked to analyze.

1.1 Sampling of bandlimited signals. Suppose the signal X has bandwidth W, i.e., that $X(f) = 0$ for all $f \notin [-W/2, W/2]$. In this case, sampling entails no loss of information in that it is possible to recover $x(t)$ perfectly if only the samples $x_s(n)$ are given. Explain why this is true and describe a method to recover the continuous time signal x from the modulated Dirac train x_δ.

1.2 Avoiding aliasing. When we sample a signal that is *not* bandlimited, there is an unavoidable loss of the information contained in frequencies larger than $f_s/2$—-and the equivalent information contained in frequencies smaller than $-f_s/2$. However, it is possible to avoid aliasing through judicious use of a low pass filter. Explain how this is done.

1.3 Reconstruction with arbitrary pulse trains. While it is mathematically possible to reconstruct $x(t)$ from $x_\delta(t)$, it is physically implausible to generate a Dirac train because delta functions are not physical entities. We can, however, approximate $\delta(t)$ by a narrow pulse $p(t)$ and attempt to reconstruct $x(t)$ from the modulated train pulse

$$x_p(t) = \sum_{n=-\infty}^{\infty} x(nT_s)p(t - nT_s). \tag{9}$$

As long as the pulse $p(t - nT_s)$ is sufficiently tall and narrow, $x_p(t)$ is not too far from $x_\delta(t)$ and the reconstruction method described in 1.1 should yield acceptable results with $x_p(t)$ used in lieu of $x_\delta(t)$. Work in the frequency domain to explain what distortion is introduced by the use of $x_p(t)$ in lieu of $x_\delta(t)$. In the course of this analysis you will realize that there is a condition on $p(t)$ that guarantees no distortion, i.e., perfect reconstruction of $x(t)$ without using a Dirac train. Derive this condition and propose a particular pulse with this property. Do notice that the pulse you are proposing is not that narrow after all.

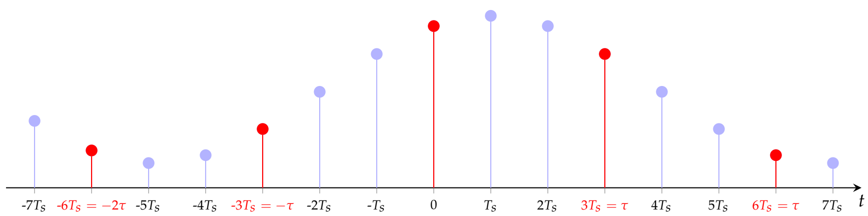

Figure 5. Subsampling (top). When subsampling a discrete time signal we retain a subset of the values of the given discrete time signal. In the figure, the sampling time of the given signal x is T_s and the sampling time of the subsampled signal x_s is $\tau = 3T_s$. We therefore keep one out of every three values of x to form x_s.

.

2 Subsampling

Most often, sampling is understood as a technique to generate a discrete time signal from a continuous time signal. However, we can also use sampling to generate a smaller number of samples from an already sampled signal. Consider then a discrete time signal x_d with sampling time T_s and values $x_d(n)$. We want to generate a (sub)sampled signal x_s with sampling time τ and values $x_s(m)$ given by

$$x_s(m) = x_d\left(m\frac{\tau}{T_s}\right). \tag{10}$$

For the expression in (10) to make sense we need to have the subsampling time τ to be an integer multiple of T_s. Under that assumption, (10) means that we retain one value of $x_d(n)$ out of every τ/T_s values. E.g., of $\tau/T_s = 2$, we keep every other sample of x_d into x_s. If $\tau/T_s = 3$, we make $x_s(0) = x_d(0)$, $x_s(1) = x_d(3)$, and, in general $x_s(m) = x_d(3m)$, so that we keep all the values in x that correspond to time indexes that are multiples of 3—see Figure 5.

As in the case of sampling, we want to understand what information, if any, is lost when we subsample x_d into x_s. And again as in the case of sampling, the difficulty in answering this question is that the support of the signals x_d and x_s are different. In (1), the continuous time signal x is a function of the continuous time parameter t and the sampled signal x_s is a function of the discrete time parameter n. In (10), the original signal x_d is defined for times nT_s, whereas the subsampled signal is defined for times $m\tau$.

We can overcome this problem by introducing the analogous of the

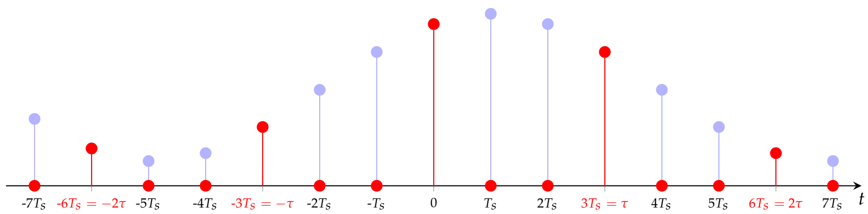

Figure 6. Delta train representation of subsampling. The difference with the subsampled signal in Figure 5 is that here we pad with zeros so that the support of this signal is the same support of the original signal.

.

modulated Dirac train in (4). To do so, consider a train of discrete time delta functions centered at discrete time indexes $m\tau/T_s$ and define the delta train representation of the subsampled signal as

$$x_\delta(n) = \sum_{m=-\infty}^{\infty} x_d\left(m\frac{\tau}{T_s}\right)\ \delta\left(n - m\frac{\tau}{T_s}\right). \tag{11}$$

A schematic representation of (11) is shown in Figure 6. The difference between x_δ and x_s is that x_δ is padded with zeros so that its support is the same support of the original signal x_d. Do notice that it is pointless to utilize x_δ for signal processing when we can use the equivalent signal x_s. However, the delta train representation x_δ is more convenient for analysis. In particular, it is ready to repeat the steps in (5)–(8) to conclude that a result equivalent to the periodization statement of Theorem 1 holds.

2.1 Subsampling theorem. Derive the equivalent of Theorem 1 relating the spectra of the discrete time signal x_d and its subsampled version x_δ. To solve this part you need to compute the DTFT of the delta train $\sum_{m=-\infty}^{\infty} \delta(n - m\tau/T_s)$. This DTFT is a Dirac train with spikes that are spaced by the subsampling frequency $\nu = 1/\tau$. If you have problems with this derivation, which you will most likely have, talk with one of your teaching assistants. If you don't want to talk with them, ponder the fact that the train $\sum_{m=-\infty}^{\infty} \delta(n - m\tau/T_s)$ is akin to a constant function when we use the sampling time τ.

2.2 Subsampling function. Create a class that takes as input a signal x_d, a sampling time T_s, and a subsampling time τ to return the subsampled signal x_s and its delta train representation x_δ. The latter signal would not be returned in practice, but we will use it here to perform some analyses. Test it with a Gaussian pulse of standard deviation $\sigma = 100$ms and mean $\mu = 1$s. Set the original sample frequency to $f_s = 40$kHz, the subsampling frequency to $f_{ss} = 4$kHz and the total observation period to $T = 2$s.

2.3 Spectrum periodization. Take the DFT of the functions x_d and x_δ of Part 2.2 and check that the periodization result of Part 2.1 holds. Keep all parameters unchanged and vary the standard deviation of the Gaussian pulse to observe cases with and without aliasing.

2.4 Prefiltering. The class you wrote in Part 2.2 results in aliasing when the spectrum of the signal x_d has a bandwidth W that exceeds ν. We can avoid aliasing by implementing a low pass filter to eliminate frequencies above ν before subsampling. Modify the class of Part 2.2 to add this feature.

2.5 Spectrum periodization with prefiltering . Repeat Part 2.3 using the class in Part 2.4. For the cases without aliasing the result should be the same. Observe and comment the differences for the cases in which you had observed aliasing.

2.6 Reconstruction function. Create a class that takes as input a subsampled signal x_s, a sampling time T_s, and a subsampling time τ to return the signal x_d. Depending on context this process may also be called interpolation—because we interpolate the values between subsequent samples in x_s—or upsampling—because we increase the sampling frequency from ν to f_s. In implementing this class you can assume that the signal x_s is bandlimited and was generated without aliasing. Test this for a Gaussian pulse. Choose parameters of Part 2.3 that did *not* result in aliasing. Choose parameters for which you observed aliasing and check that, indeed, the reconstructed pulse is not a faithful representation of the original pulse. For this latter experiment, test both the subsampling class in Part 2.2 and the subsampling class in Part 2.4.

3 Time management

This lab returns to the mean and is more involved than Lab 4. Part 1 includes results that we will derive on class, so it shouldn't be too onerous to finish. The teaching assistants will work on these problems during the first meeting. It should be an additional hour or so to wrap it up.

Part 2.1 is an odd man as it is asking that you do a somewhat involved derivation. Work on it during Wednesday and, if you can't solve it before the end of the day, go talk with one of your teaching assistants. A 2 hour investment should do.

We will work on the remaining parts on the last lab session. Completing the rest should take about 5 hours, 1 hour for each of the parts.