Timeseries Translation

L. S. Finn

Center for Gravitational Wave Physics, Penn State University

(Dated: January 20, 2002)

Abstract

Brief notes describing how a properly bandlimited continuous function can be derived from reconstructed from its samples, both ideally and in an approximation whose errors are quantifiable and controlled. From the reconstruction the function can be resampled, at the same sample rate, but translated with respect to the original sampling by a non-integer number of samples. This facility is required by the LDAS datacondAPI to support joint analysis of data from different detectors, or data from different channels of the same detector when one or both have been resampled by (different) non-integer factors.

I. INTRODUCTION

Resampling and other operations may introduce non-integer delays into a timeseries: i.e., the sample times of the samples at the output of the operation may be advanced or retarded relative to the sample times of the input by a non-integer number of samples. Additionally, we have the need jointly analyze several different channels, which may arise from different detectors, whose sample times do not correspond. In both cases it is necessary that we translate one of the sampled timeseries so that, in its translated representation, the sample times correspond with the other channels. Since translation by an integer delay is trivial, we can restrict attention to translation by a fraction of a sample period. In these notes we describe how to reconstruct a sampled function from its samples and then re-sample the function with an arbitrary delay.

II. IDEAL RECONSTRUCTION

A bandwidth limited signal with $f < f_N$ is fully represented by its samples if the sampling rate is greater than the Nyquist sampling rate $f_S = 2f_N$. To understand the reconstruction it is useful to take a Green function approach: i.e., if we understand the reconstruction of the bandlimited signal whose sampled representation is the Kronecker delta, then the reconstruction of any sampled signal is the superposition of these, with the appropriate weights.

In the continuous limit the Kronecker delta becomes a Dirac delta function. Thus the continuous, bandlimited function whose sampled representation is a Kronecker delta is the bandlimited Dirac delta function, or the sinc function:

$$
\begin{aligned}
\operatorname{sinc} f_S(t'-t_0) &= \int_{-f_N}^{f_N} \exp(2\pi i f t') \frac{df}{2f_N} \int_{-\infty}^{\infty} dt\, \delta(t-t_0) \exp(-2\pi i f t) \\
&= \int_{-f_N}^{f_N} \exp\left[2\pi i f (t'-t_0)\right] \frac{df}{2f_N} \\
&= \frac{\sin \pi f_S (t'-t_0)}{\pi f_S (t'-t_0)}. \qquad (1)
\end{aligned}
$$

The conclusion, that the bandlimited function $x(t)$ whose sampled representation is the timeseries $x(t_k) = \delta_{jk}$ is $\operatorname{sinc} f_S(t-t_j)$, is easily verified.

We can thus write the reconstruction of the bandlimited function x from its timeseries $x[k]$ ($x[k] = x(kT_S)$) as a convolution of $x[k]$ with a sinc function, as in

$$
x(t) = \sum_{k=-\infty}^{\infty} x[k] \operatorname{sinc} \pi f_S (t - kT_S) \qquad (2)
$$

where T_S is the sample period $1/f_S$. The most important result here is not that the kernel for reconstruction is the sinc function, but that reconstruction at any particular point is a linear operation that can be written as a convolution of the sampled function with another sampled function.

III. APPROXIMATION

The sinc function has infinite support; consequently, ideal reconstruction by convolution with the sinc function, as in equation 2, is not practical. We require an approximation to ideal reconstruction, preferably with errors that we can control, or at least quantify.

Begin by writing our approximate reconstruction of the timeseries $g[k]$ in the form

$$g(t) = \sum_{k=-\infty}^{\infty} g[k] b(t - t_k) \tag{3}$$

where $b(t)$ is a function that acts as the kernel for the reconstruction. If $b(t)$ is $\mathrm{sinc} f_S t$ then we have ideal reconstruction.

The sample $g[k]$ is the value of $g(t)$ at t_k, which we may express in terms of a Taylor series of g about t, as in

$$g[k] = g(t_k) = \sum_{n=0}^{\infty} \frac{(t_k - t)^n}{n!} g^{(n)}(t). \tag{4}$$

Substituting this Taylor expansion into our expression for approximate reconstruction we obtain

$$\begin{aligned} g(t) &= \sum_{n=0}^{\infty} g^(n)(t) \frac{1}{n!} \sum_{k=-\infty}^{\infty} (t_k - t)^n b(t - t_k) \\ &= \sum_{n=0}^{\infty} g^{(n)}(t) B_n(t) \end{aligned} \tag{5}$$

where

$$B_n(t) = \frac{1}{n!} \sum_{k=-\infty}^{\infty} (t_k - t)^n b(t - t_k) \tag{6}$$

For perfect reconstruction we want to choose $b(t)$ such that $B_0(t)$ is unity and $B_n(t)$ for $n \geq 1$ vanishes. If, on the other hand, we choose a $b(t)$ such that $B_0(t)$ is unity and $B_n(t)$ vanishes for $1 \leq n \leq N$, then we will have found an approximation to perfect reconstruction that is *exact* for polynomials of order less than or equal to N.

This last suggests that we look to Lagrange interpolation, which provides the unique polynomial of of order $N - 1$ that fits N samples of a function. Lagrange interpolation can be written in the form

$$g(t) = \sum_{k=0}^{N-1} a_k(t) g[k] \tag{7}$$

where

$$a_k(t) = \prod_{\substack{j=0 \\ j \neq k}}^{N-1} \frac{t - t_j}{t_k - t_j}. \tag{8}$$

This is exactly of the form of equation 5; so, we identify the a_k as the order N approximation, where we understand that the order N approximation exactly reconstructs polynomials of order less than or equal to N. To put this result another way, reconstruction using this approximation makes errors only in the order $N + 1$ and higher derivatives of the reconstructed function.

IV. TIMESERIES TRANSLATION

We have now all the machinery necessary to translate a timeseries. We begin with the sampled timeseries $g[j]$, where

$$g[j] = g(jT_S), \tag{9}$$

where T_S is the sample period and $g(t)$ is an appropriately bandlimited function. We desire the timeseries

$$g'[j] = g\left((j+\alpha)T_S\right) \tag{10}$$

where without loss of generality $0 < \alpha < 1$. For an approximation exact in the first N derivatives of g(t), equations 5 and 8 can be combined to yield

$$g'[j] = g\left((j+\alpha)T_S\right) \tag{11}$$

$$= \sum_{k=m-N+1}^{m} g[j-k] b_N[k] \tag{12}$$

where

$$b_N[k] = \prod_{\substack{l=0 \\ l\neq m-k}}^{N-1} \frac{\alpha+m-l}{m-k-l} \tag{13}$$

$$m = \begin{cases} N/2 & \text{if } N \text{ even} \\ (N-1)/2 & \text{if } N \text{ odd} \end{cases} \tag{14}$$

(There is a freedom in our choice of the samples $g[j]$ from which we interpolate $g'[k]$. Here we have chosen to interpolate using an equal number of samples to the right and left of $g'[k]$ or, for odd-order interpolation, one more sample to the right than to the left.) Equation 12 is just a non-causal FIR filter.

V. IMPLEMENTATION NOTES

Once α is specified, the FIR filter coefficients $b_N[k]$ are fully determined. The datacondAPI linear filter class can be used to apply this filter to the input time series. The linear filter class assumes the filter is causal: i.e., the linear filter output will be translated exactly $m - N + 1$ samples with respect to the desired output. The final translation by an integer number of samples is readily accomplished by resetting the time of the first sample in the output timeseries metadata.