Bandlimited Interpolation — Introduction and Algorithm

Julius O. Smith III

Center for Computer Research in Music and Acoustics (CCRMA)
Department of Music, Stanford University, Stanford, California 94305

DRAFT 1/5/93

Abstract

A technique for bandlimited interpolation of digital signals is described which supports non-uniform and time-varying resampling. The method is based on interpolated look-up in a large table of filter coefficients. One filter table handles all conversion factors. Formulas are given for determining the look-up table size needed for a given precision requirement.

1. Introduction

Bandlimited interpolation of discrete-time signals is a basic tool having extensive application in digital signal processing. In general, the problem is to correctly compute signal values at arbitrary continuous times from a set of discrete-time samples of the signal amplitude. In other words, we must be able to interpolate the signal between samples. Since the original signal is always assumed to be *bandlimited* to half the sampling rate, (otherwise aliasing distortion would occur upon sampling), Shannon's sampling theorem tells us the signal can be exactly and uniquely reconstructed for all time from its samples by bandlimited interpolation.

Considerable research has been devoted to the problem of interpolating discrete points. A comparison between classical (e.g. Lagrange) and bandlimited interpolation is given in (Schafer and Rabiner 1973). The book **Multirate Digital Signal Processing** (Crochiere and Rabiner 1983) provides a comprehensive summary and review of previous techniques for sampling-rate conversion. In these techniques, the signal is first interpolated by an integer factor L and then decimated by an integer factor M. This provides sampling-rate conversion by any rational factor L/M. The conversion requires a digital lowpass filter whose cutoff frequency depends on $max\{L, M\}$. While sufficiently general, this formulation is less convenient when it is desired to resample the signal at arbitrary times or change the sampling-rate conversion factor smoothly over time.

In this paper, a resampling algorithm is described which will evaluate a signal at any time specifiable by a fixed-point number. In addition, one lowpass filter is used regardless of the sampling-rate conversion factor. The algorithm effectively implements the "analog interpretation" of rate conversion, as discussed in (Crochiere and Rabiner 1983), in which a certain lowpass-filter impulse response must be available as a continuous function. Continuity of the impulse response is simulated by linearly interpolating between samples of the impulse response stored in a table. Due to the relatively low cost of memory, the method is quite practical for hardware implementation.

In section 2, the basic theory is presented, section 3 addresses practical issues, and implementation details are discussed in section 4. Finally, section 5 discusses numerical requirements on the length, width, and interpolation accuracy of the filter coefficient table.

2. Theory

We review briefly the "analog interpretation" of sampling rate conversion (Crochiere and Rabiner 1983) on which the present method is based. Suppose we have samples $x(nT_s)$ of a continuous absolutely integrable signal $x(t)$, where t is time in seconds (real), n ranges over the integers, and T_s is the sampling period. We assume $x(t)$ is bandlimited to $\pm F_s/2$, where $F_s = 1/T_s$ is the sampling rate. If $X(\omega)$ denotes the Fourier transform of $x(t)$, i.e., $X(\omega) = \int_{-\infty}^{\infty} x(t)e^{-j\omega t}dt$, then we assume $X(\omega) = 0$ for $|\omega| \geq \pi F_s$. Consequently, Shannon's sampling theorem gives us that $x(t)$ can be uniquely reconstructed from the samples $x(nT_s)$ via

$$\hat{x}(t) \triangleq \sum_{n=-\infty}^{\infty} x(nT_s)h_s(t - nT_s) \equiv x(t)\,, \tag{1}$$

where

$$h_s(t) \triangleq \mathrm{sinc}(F_s t) \triangleq \frac{\sin(\pi F_s t)}{\pi F_s t}\,. \tag{2}$$

To resample $x(t)$ at a new sampling rate $F'_s = 1/T'_s$, we need only evaluate Eq. 1 at integer multiples of T'_s.

When the new sampling rate F'_s is less than the original rate F_s, the lowpass cutoff must be placed below half the new lower sampling rate. Thus, in the case of an ideal lowpass, $h_s(t) = min\{1, F'_s/F_s\}\mathrm{sinc}(min\{F_s, F'_s\}t)$, where the scale factor maintains unity gain in the passband.

A plot of the sinc function $\mathrm{sinc}(t)$ for the first seven zero-crossings to the left and right of the origin $t = 0$ is shown in Fig. 1.

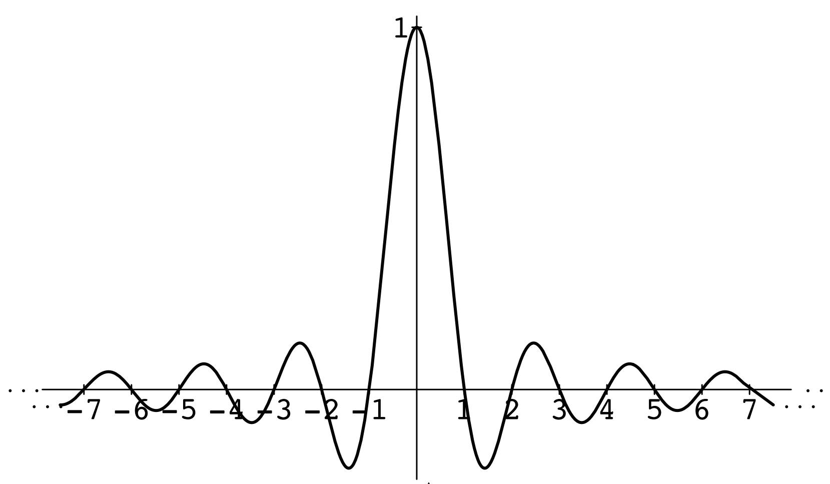

Figure 1. The sinc function $\mathrm{sinc}(t) \triangleq \sin(\pi t)/(\pi t)$ plotted for seven zero-crossings to the left and right. The peak is at amplitude 1 and zero-crossings

occur at all nonzero integers. The sinc function can be seen as a hyperbolically weighted sine function with its zero at the origin canceled out.

If "$*$" denotes the convolution operation for digital signals, then the summation in Eq. 1 can be written as $(x * h_s)(t)$.

Equation 1 can be interpreted as a superpositon of shifted and scaled sinc functions h_s. A sinc function instance is translated to each signal sample and scaled by that sample, and the instances are all added together. Note that zero-crossings of sinc(z) occur at all integers except $z = 0$. That means at time $t = nT_s$, (i.e., on a sample instant), the only contribution to the sum is the single sample $x(nT_s)$. All other samples contribute sinc functions which have a zero-crossing at time $t = nT_s$. Thus, the interpolation goes precisely through the existing samples, as it should.

A plot indicating how sinc functions sum together to reconstruct bandlimited signals is shown in Fig. 2. The figure shows a superposition of five sinc functions, each at unit amplitude, and displaced by one-sample intervals. These sinc functions would be used to reconstruct the bandlimited interpolation of the discrete-time signal $x = [\ldots, 0, 1, 1, 1, 1, 1, 0, \ldots]$. Note that at each sampling instant $t = nT_s$, the solid line passes exactly through the tip of the sinc function for that sample; this is just a restatement of the fact that the interpolation passes through the existing samples. Since the nonzero samples of the digital signal are all 1, we might expect the interpolated signal to be very close to 1 over the nonzero interval; however, this is far from the case. The deviation from unity between samples can be thought of as "overshoot" or "ringing" of the lowpass filter which cuts off at half the sampling rate, or it can be considered a "Gibbs phenomenon" associated with bandlimiting.

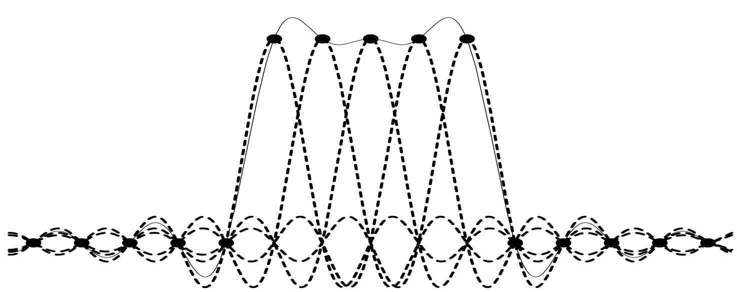

Figure 2. Bandlimited reconstruction of the signal $x(t)$ from its samples $x = [\ldots, 0, 1, 1, 1, 1, 1, 0, \ldots]$. The dots show the signal samples, the dashed lines show the component sinc functions, and the solid line shows the unique bandlimited reconstruction from the samples obtained by summing the component sinc functions.

A second interpretation of Eq. 1 is as follows: to obtain the interpolation at time t, shift the signal samples under *one* sinc function so that time t in the signal is translated under the peak of the sinc function, then create the output as a linear combination of signal samples where the coefficient of each signal sample is given by the value of the sinc function at the location of each sample. That this interpretation is equivalent to the first can be seen as a result of the fact that convolution is commutative; in the first interpretation, all signal samples are used to form a linear combination of shifted sinc functions, while in the second interpretation, samples from one sinc function are used to form a linear combination of samples of the shifted input signal. The practical bandlimited interpolation algorithm presented below is based on the second interpretation.

3. From Theory to Practice

The summation in Eq. 1 cannot be implemented in practice because the "ideal lowpass filter" impulse response $h_s(t)$ actually extends from minus infinity to infinity. It is necessary in practice to *window* the ideal impulse response so as to make it finite. This is the basis of the *window method* for digital filter design (DSP Committee 1979, Rabiner and Gold 1975). While many other filter design techniques exist, the window method is simple and robust, especially for very long impulse responses. In the case of the algorithm presented below, the filter impulse response is very long because it is heavily oversampled.

Figure 3 shows the frequency response of the ideal lowpass filter. This is just the Fourier transform of $h_s(t)$.

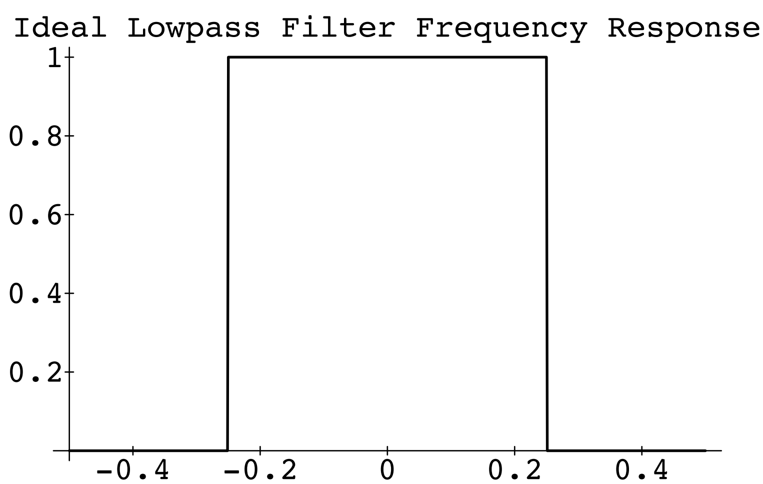

Figure 3. Frequency response of the ideal lowpass filter.

If we truncate $h_s(t)$ at the fifth zero-crossing to the left and the right of the origin, we obtain the frequency response shown in Figure 4. Note that the stopband exhibits only slightly more than 20 dB rejection.

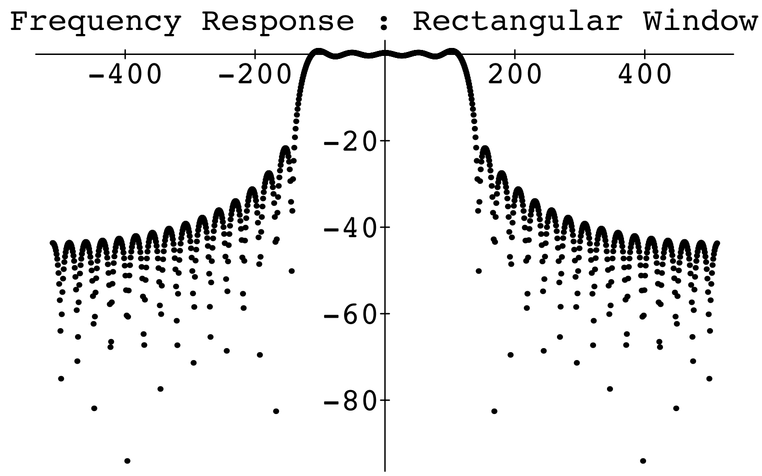

Figure 4. Frequency response of the ideal lowpass filter after rectangularly windowing the ideal (sinc) impulse response at the fifth zero crossing to the left and right of the time origin. The vertical axis is in units of decibels (dB), and the horizontal axis is labeled in units of spectral samples between plus and minus half the sampling rate.

If we instead use the Kaiser window to taper $h_s(t)$ to zero by the fifth zero-crossing to the left and the right of the origin, we obtain the frequency response shown in Figure 5. Note that now the stopband starts out close to -80 dB. The Kaiser window has a single parameter which can be used to modify the stop-band attenuation, trading it against the transition width from pass-band to stop-band.

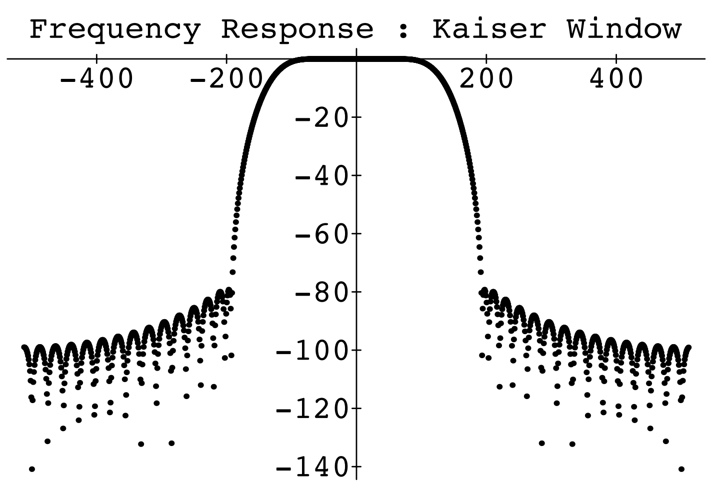

Figure 5. Frequency response of the ideal lowpass filter Kaiser windowed at the fifth zero crossing to the left and right.

4. Implementation

Our implementation provides signal evaluation at an arbitrary time, where time is specified as an unsigned binary fixed-point number in units of the input sampling period (assumed constant). Figure 6 shows the time register t, and Figure 7 shows an example configuration of the input signal and lowpass filter at a given time. The time register is divided into three fields: The leftmost field gives the number n of samples into the input signal buffer, the middle field is an initial index l into the filter coefficient table $h(l)$, and the rightmost field is interpreted as a number η between 0 and 1 for doing linear interpolation between samples l and $l+1$ (initially) of the filter table. The concatenation of l and η are called $P \in [0,1)$ which is interpreted as the position of the current time between samples n and $n+1$ of the input signal.

Let the three fields have n_n, n_l, and n_η bits, respectively. Then the input signal buffer contains $N = 2^{n_n}$ samples, and the filter table contains $L = 2^{n_l}$ "samples per zero-crossing." (The term "zero-crossing" is precise only for the case of the ideal lowpass; to cover practical cases we generalize "zero-crossing" to mean a multiple of time $t_c = 1/f_c$, where f_c is the lowpass cutoff frequency.) For example, to use the ideal lowpass filter, the table would contain $h(l) = \mathrm{sinc}(l/L)$.

Our implementation stores only the "right wing" of a symmetric finite-impulse-response (FIR) filter (designed by the window method based on a Kaiser window (Rabiner and Gold 1975)). It also stores a table of differences $\overline{h}(l) = h(l+1) - h(l)$ between successive FIR sample values in order to speed up the linear interpolation. The length of each table is then $N_h = L(N_z + 1)$.

Consider a sampling-rate conversion by the factor $\rho = F_s'/F_s$. For each output sample, the basic interpolation Eq. 1 is performed. The filter table is traversed twice—first to apply the left wing of the FIR filter, and second to apply the right wing. After each output sample is computed, the time register is incremented by $2^{n_l+n_\eta}/\rho$ (i.e., time is incremented by $1/\rho$ in fixed-point format). Suppose the time register t has just been updated, and an interpolated output $y(t)$ is desired. For $\rho \geq 1$, output is computed via

$$\begin{aligned} v &\leftarrow \sum_{i=0}^{h\ \mathrm{end}} x(n-i)\left[h(l+iL) + \eta\overline{h}(l+iL)\right] \\ P &\leftarrow 1 - P \\ y(t) &\leftarrow v + \sum_{i=0}^{h\ \mathrm{end}} x(n+1+i)\left[h(l+iL) + \eta\overline{h}(l+iL)\right], \end{aligned} \tag{3}$$

where $x(n)$ is the current input sample, and $\eta \in [0,1)$ is the interpolation factor. When $\rho < 1$, the initial P is replaced by $P' = \rho P$, $1 - P$ becomes $\rho - P' = \rho(1-P)$, and the step-size through the filter table is reduced to ρL instead of L; this lowers the filter cutoff to avoid aliasing. Note that η is fixed throughout the computation of an output sample when $\rho \geq 1$ but changes when $\rho < 1$.

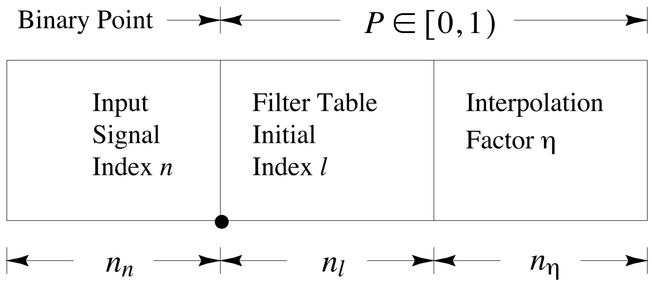

Figure 6. Time register format.

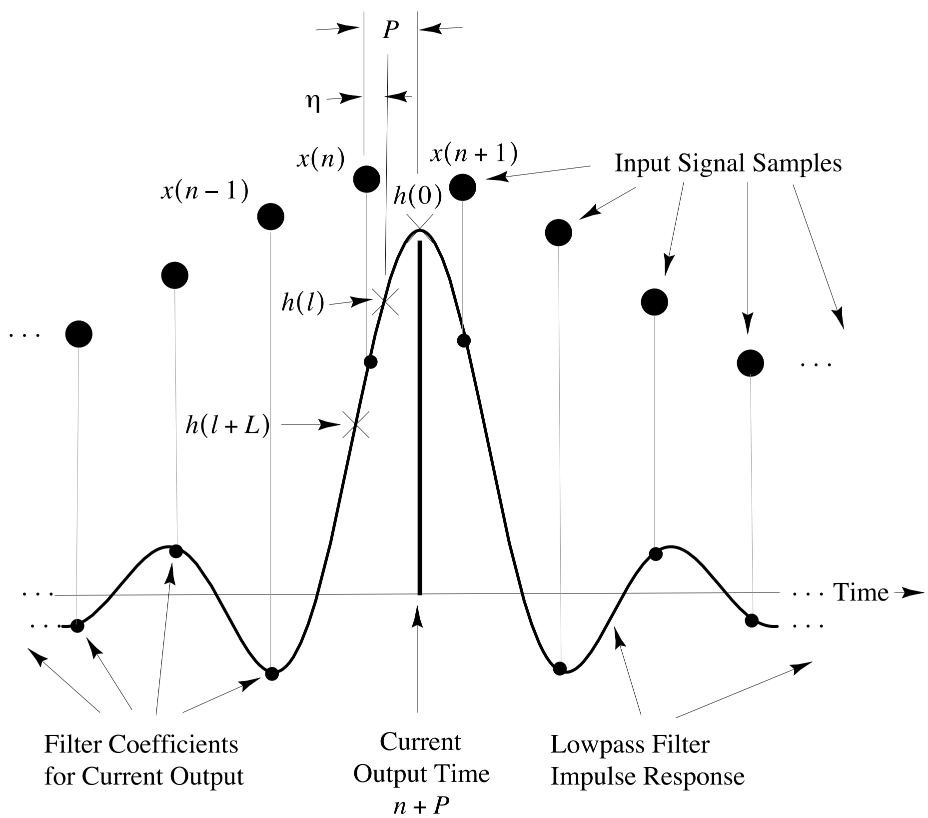

Figure 7. Illustration of waveforms and parameters in the interpolator.

When $\rho < 1$, more input samples are required to reach the end of the filter table, thus preserving the filtering quality. The number of multiply-adds per second is approximately $(2N_z + 1)\, max\{F_s, F_s'\}$. Thus the higher sampling rate determines the work rate. Note that for $\rho < 1$ there must be $\lceil N_z F_s / F_s' \rceil$ extra input samples available before the initial conversion time and after the final conversion time in the input buffer. As $\rho t_1 0$, the required extra input data becomes infinite, and some limit must be chosen, thus setting a minimum supported ρ. For $\rho \geq 1$, only N_z extra input samples are required on the left and right of the data to be resampled, and the upper bound for ρ is determined only by the fixed-point number format, viz., $\rho\text{max} = 2^{n_l+n_\eta}$.

As shown below, if n_c denotes the word-length of the stored impulse-response samples, then one may choose $n_l = 1 + n_c/2$, and $n_\eta = n_c/2$ to obtain $n_c - 1$ effective bits of precision in the interpolated impulse response.

Note that rational conversion factors of the form $\rho = L/M$, where $L = 2^{n_l}$ and M is an arbitrary positive integer, do not use the linear interpolation feature (because $\eta \equiv 0$). In this case our method reduces to the normal type of bandlimited interpolator (Crochiere and Rabiner 1983). With the availability of interpolated lookup, however, the range of conversion factors is boosted to the order of $2^{n_l+n_\eta}/M$. E.g., for $\rho \approx 1$, $n_l = 9, n_\eta = 8$, this is about 5.1 decimal digits of accuracy in the conversion factor ρ. Without interpolation, the number of significant figures in ρ is only about 2.7.

The number N_z of zero-crossings stored in the table is an independent design parameter. For example, we use $N_z = 13$ in a system designed for audio quality with 20% oversampling.

For a given quality specification in terms of aliasing rejection, a trade-off exists between N_z and sacrificed bandwidth. The lost bandwidth is due to the so-called "transition band" of the lowpass filter (Rabiner and Gold 1975). In general, for a given stop-band specification (such as "80 dB attenuation"), lowpass filters need approximately twice as many multiply-adds per sample for each halving of the transition band width.

It is worth noting that a given percentage increase in the original sampling rate ("oversampling") gives a larger percentage savings in filter computation time, for a given quality specification, because the added bandwidth is a larger percentage of the filter transition bandwidth than it is of the original sampling rate. For example, given a cut-off frequency of 20 kHz, (ideal for audio work), the transition band available with a sampling rate of 44 kHz is about 2 kHz, while a 48 kHz sampling rate provides a 4 kHz transition band. Thus, a 10% increase in sampling rate *halves* the work per sample in the digital lowpass filter.

5. Quantization Issues

In this section, we investigate the requirements on the sampling density $L = 2^{n_l}$ of the lowpass-filter impulse response, and the number of bits n_η required in the interpolation factor η. These quantities are determined by computing the worst-case error and comparing it to the filter coefficient quantization error.

Choice of Table Size

It is desirable that the stored filter impulse response be sampled sufficiently densely so that interpolating linearly between samples does not introduce error greater than the quantization error. We will show that this condition is satisfied whenever the filter table contains at least $L = 2^{1+n_c/2}$ entries per zero-crossing, where n_c is the number of bits allocated to each table entry.

Linear Interpolation Error Bound

Let $h(t)$ denote the lowpass filter impulse response, and assume it is twice continuously differentiable for all t. By Taylor's theorem (Goldstein 1967, p. 119), we have

$$h(t_0 + \eta) = h(t_0) + \eta h'(t_0) + \frac{1}{2}\eta^2 h''(t_0 + \lambda\eta)\,, \tag{4}$$

for some $\lambda \in [0, 1]$, where $h'(t_0)$ denotes the time derivative of $h(t)$ evaluated at $t = t_0$, and $h''(t_0)$ is the second derivative at t_0.

The linear interpolation error is defined as

$$\tilde{h}(t) \triangleq h(t) - \hat{h}(t)\,, \tag{5}$$

where $t = t_0 + \eta$, $t_0 = \lfloor t \rfloor$, $\eta = t - t_0$, and $\hat{h}(t)$ is the interpolated value given by

$$\hat{h}(t) \triangleq \overline{\eta} h(t_0) + \eta h(t_1)\,, \tag{6}$$

where $\overline{\eta} \triangleq 1 - \eta$ and $t_1 \triangleq t_0 + 1$. Thus t_0 and t_1 are successive time instants for which samples of $h(t)$ are available, and $\eta \in [0, 1)$ is the linear interpolation factor. (We ignore errors in the linear interpolation itself at this point.)

Expressing $h(t)$ as

$$h(t_0 + \eta) = \overline{\eta} h(t_0 + \eta) + \eta h(t_1 - \overline{\eta}) \tag{7}$$

applying (4) to both terms on the right-hand side, and subtracting (6) gives

$$\tilde{h}(t_0 + \eta) = \eta\overline{\eta} \left[h'(t_0) - h'(t_1) + \frac{\eta h''(\xi_0) + \overline{\eta} h''(\xi_1)}{2} \right]\,, \tag{8}$$

where both ξ_0 and ξ_1 are in $[t_0, t_1]$. Defining

$$M_2 \triangleq \max_t |h''(t)| \tag{9}$$

and noting that $h'(t_1) = h'(t_0) + h''(t_0 + \lambda)$ for some $\lambda \in [0, 1]$ which implies

$$|h'(t_0) - h'(t_1)| \le M_2\,, \tag{10}$$

we obtain the upper bound

$$\left| \tilde{h}(t_0 + \eta) \right| \le \eta\overline{\eta} \left[M_2 + \frac{M_2}{2} \right] \le \frac{3}{8} M_2\,. \tag{11}$$

Application to the Ideal Lowpass Filter

For the ideal lowpass filter, we have

$$h(t) = \mathrm{sinc}(\omega_L t/\pi) \triangleq \frac{\sin(\omega_L t)}{\omega_L t} = \frac{1}{\omega_L}\int_0^{\omega_L} \cos(\omega t) d\omega\,, \tag{12}$$

where $\omega_L = \pi/L$, and $L = 2^{n_l}$ is the number of table entries per zero-crossing. Note that the rightmost form in (12) is simply the inverse Fourier transform of the ideal lowpass-filter frequency response. Twice differentiating with respect to t, we obtain

$$h''(t) = -\frac{1}{\omega_L}\int_0^{\omega_L} \omega^2 \cos(\omega t) d\omega\,, \tag{13}$$

from which it follows that the maximum magnitude is

$$M_2 = \frac{\omega_L^2}{3} = \frac{\pi^2}{3L^2}\,. \tag{14}$$

Note that this bound is attained at $t = 0$. Substituting (14) into (11), we obtain the error bound

$$\left|\tilde{h}(t_0+\eta)\right| \leq \frac{\pi^2}{8L^2} < \frac{1.234}{L^2} = 1.234 \cdot 2^{-2n_l}\,. \tag{15}$$

Thus for the ideal lowpass filter $h(t) = \mathrm{sinc}(t/L)$, the pointwise error in the interpolated lookup of $h(t)$ is bounded by $1.234/L^2$. This means that n_l must be about half the coefficient word-length n_c used for the filter coefficients. For example, if $h(t)$ is quantized to 16 bits, L must be of the order of $2^{16/2} = 256$. In contrast, we will show that without linear interpolation, n_l must increase proportional to n_c for n_c-bit samples of $h(t)$. In the 16-bit case, this gives $L \sim 2^{16} = 65536$. The use of linear interpolation of the filter coefficients reduces the memory requirements considerably.

The error bounds obtained for the ideal lowpass filter are typically accurate also for lowpass filters used in practice. This is because the error bound is a function of M_2, the maximum curvature of the impulse response $h(t)$, and most lowpass designs will have a value of M_2 very close to that of the ideal case. The maximum curvature is determined primarily by the bandwidth of the filter since, generalizing equations (12) and (13),

$$h''(0) = -\frac{1}{\pi}\int_0^{\pi} \omega^2 H(\omega) d\omega\,,$$

which is just the second moment of the lowpass-filter frequency response $H(\omega)$ (which is real for symmetric FIR filters obtained by symmetrically windowing the ideal sinc function (Rabiner and Gold 1975)). A lowpass-filter design will move the cut-off frequency slightly below that of the ideal lowpass filter in order to provide a "transition band" which allows the filter response to give sufficient rejection at the ideal cut-off frequency which is where aliasing begins. Therefore, in a well designed practical lowpass filter, the error bound M_2 should be lower than in the ideal case.

Relation of Interpolation Error to Quantization Error

If $h(t) \in [-1, 1-2^{-n_c}]$ is approximated by $h_q(t)$ which is represented in two's complement fixed-point arithmetic, then

$$h_q(t_0) = -b_0 + \sum_{i=1}^{n_c-1} b_i 2^{-i}\,, \tag{16}$$

where $b_i \in \{0,1\}$ is the ith bit, and the worst-case rounding error is

$$|h(t) - h_q(t)| \le 2^{-n_c}\,. \tag{17}$$

Letting $h_q(t_i) = h(t_i) + \epsilon_i$, where $|\epsilon_i| \le 2^{-n_c}$, the interpolated look-up becomes

$$\hat{h}_q(t_0+\eta) = \overline{\eta} h_q(t_0) + \eta h_q(t_1) = \hat{h}(t_0+\eta) + \overline{\eta}\epsilon_0 + \eta\epsilon_1\,. \tag{18}$$

Thus the error in the interpolated lookup between quantized filter coefficients is bounded by

$$\left|\tilde{h}_q(t)\right| \le \frac{3}{8} M_2 + 2^{-n_c}\,, \tag{19}$$

which, in the case of $h(t) = \mathrm{sinc}(t/L)$, can be written

$$\left|\tilde{h}_q(t)\right| < \frac{1.234}{L^2} + 2^{-n_c} = 1.234 \cdot 2^{-2n_l} + 2^{-n_c}\,. \tag{20}$$

If $L = 2^{1+n_c/2}$, then $|\tilde{h}_q(t)| < 1.5 \cdot 2^{-n_c}$, and the interpolation error is less than the quantization error by more than a factor of 2.

Error in the Absence of Interpolation

For comparison purposes, we derive the error incurred when no interpolation of the filter table is performed. In this case, assuming rounding to the nearest table entry, we have

$$\begin{aligned} t &= t_0 + \eta, \qquad |\eta| \le \frac{1}{2} \\ \hat{h}(t) &= h(t_0) \\ \tilde{h}(t) &= h(t) - h(t_0) \\ &= \eta h'(t_0) + \frac{1}{2}\eta^2 h''(t_0 + \lambda\eta) \\ \left|\tilde{h}(t)\right| &\le \frac{M_1}{2} + \frac{M_2}{8}\,, \end{aligned} \tag{21}$$

where $M_1 \triangleq max_t\,|h'(t)|$. For the ideal lowpass, we have

$$h'(t) = -\frac{1}{\omega_L}\int_0^{\omega_L} \omega \sin(\omega t)d\omega = \frac{\omega_L t\cos(\omega_L t) - \sin(\omega_L t)}{\omega_L t^2}\,. \tag{22}$$

Note that $h'(L) = 1/L$ and $|h'(t)| < \omega_L/2 = \pi/2L$. Thus $M_1 = a/L$ where $1 \le a < \pi/2$. The no-interpolation error bound is then

$$|h'(t)| \le \frac{a}{2L} + \frac{\pi^2}{24L^2} < \frac{0.7854}{L} + \frac{0.4113}{L^2}\,. \tag{23}$$

Choice of Interpolation Resolution

We now consider the error due to finite precision in the linear interpolation between stored filter coefficients. We will find that the number of bits n_η in the interpolation factor should be about half the filter coefficient word-length n_c.

Quantized Interpolation Error Bound

The quantized interpolation factor and its complement are representable as

$$\begin{aligned} \eta_q &= \eta + \nu \\ \overline{\eta}_q &= \overline{\eta} - \nu \end{aligned} \tag{24}$$

where, since $\eta, \overline{\eta}$ are unsigned, $|\nu| \leq 2^{-(n_\eta+1)}$. The interpolated coefficient look-up then gives

$$\begin{aligned} \hat{h}_{qq}(t) &= (\overline{\eta} - \nu)[h(t_0) + \epsilon_0] + (\eta + \nu)[h(t_1) + \epsilon_1] \\ &= \hat{h}(t) + \overline{\eta}\epsilon_0 + \eta\epsilon_1 + \nu[h(t_1) - h(t_0)], \end{aligned} \tag{25}$$

where second-order errors $\nu\epsilon_0$ and $\nu\epsilon_1$ are dropped. Since $|h(t_1) - h(t_0)| \leq M_1$, we obtain the error bound

$$\left|\tilde{h}_{qq}(t)\right| \leq 2^{-n_c} + 2^{-(n_\eta+1)} M_1 + \frac{3}{8} M_2 \,. \tag{26}$$

The three terms in (26) are caused by coefficient quantization, interpolation quantization, and linear-approximation error, respectively.

Ideal Lowpass Filter

For the ideal lowpass, the error bound is

$$\left|\tilde{h}_{qq}(t)\right| \leq 2^{-n_c} + a 2^{-(n_l+n_\eta+1)} + \frac{\pi^2}{8} 2^{-n_l} \,.$$

Let $n_l = 1 + n_c/2$ and require that the added error is at most $\frac{1}{2}2^{-n_c}$. Then we arrive at the requirement

$$n_\eta \geq \frac{n_c}{2} \,. \tag{27}$$

6. Conclusions

A digital resampling method has been described which is convenient for non-uniform or time-varying resampling, and which is attractive for hardware implementation. We have presented the case which assumes uniform sampling of the input signal; however, extension to non-uniformly sampled input signals is straightforward.

A quantization error analysis led to the conclusion that for n_c-bit filter coefficients, the number of impulse-response samples stored in the filter lookup table should be on the order of $2^{n_c/2}$ times the number of "zero-crossings" in the impulse response, and the number of bits in the interpolation between impulse-response samples should be about $n_c/2$. With these choices, the linear interpolation error and the error due to quantized interpolation factors are each about equal to the coefficient quantization error. A signal resampler designed according to these rules will typically be limited primarily by the lowpass filter *design*, rather than by quantization effects.

We note that the error analysis presented here is pessimistic in the sense that it assumes worst-case input signal conditions (e.g., a sinusoid at half the sampling rate or white noise). A different type of error analysis is possible by treating the filter coefficients as exact but subject to time jitter. In this approach, the error can be expressed in terms of the input signal Taylor series expansion, and consequently in terms of the input signal bandwidth (or maximum slope). Such an analysis reveals that for most practical signals, the quantization error is considerably less than the levels derived here.

References

R. E. Crochiere and L. R. Rabiner, *Multirate Digital Signal Processing,* Prentice-Hall Inc., Englewood Cliffs, NJ, 1983.

Digital Signal Processing Committee, ed., *Programs for Digital Signal Processing,* IEEE Press, New York, 1979.

A. A. Goldstein, *Constructive Real Analysis,* Harper and Row, New York, 1967.

L. R. Rabiner and B. Gold, *Theory and Application of Digital Signal Processing,* Prentice-Hall Inc., Englewood Cliffs, NJ, 1975.

R. W. Schafer and L. R. Rabiner, "A Digital Signal Processing Approach to Interpolation," *Proc. IEEE,* vol. 61, pp. 692–702, June 1973.

J. O. Smith and P. Gossett, "A Flexible Sampling-Rate Conversion Method," *Proc. IEEE Conf. Acoust. Sp. and Sig. Proc., vol. 2, pp. 19.4.1-19.4.2, San Diego, March 1984.* (Original version of this paper.)

Note: A UNIX-compressed, Postscript-format version of this paper and a related Mathematica notebook (covering use of the Kaiser window) can be obtained via anonymous FTP to ccrma-ftp.stanford.edu, subdirectory pub/DSP/.