A Flexible Sampling-Rate Conversion Method

Julius O. Smith* and Phil Gossett**

*Systems Control Technology
1801 Page Mill Road, Palo Alto, CA 94303

**Center for Computer Research in Music and Acoustics
Department of Music, Stanford University, Stanford, California 94305
or: Symmetry Group, 1169 Burgoyne Ave., Mountain View, CA 94043

Abstract

A digital resampling method is proposed which allows 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 size of look-up table needed for a given precision requirement.

1. Introduction

Digital sampling-rate conversion is a basic tool having extensive application in digital signal processing. In its general form, the problem is to compute signal values at arbitrary times from a set of discrete samples. The problem is therefore one of *interpolation* between samples of the original signal. It is usually assumed that the original signal is bandlimited to half the sampling rate which implies it can be *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 [2]. The recent book by Crochiere and Rabiner [5] is a comprehensive summary and review of the prior art in sampling-rate conversion.

In previous approaches to sampling-rate conversion, 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 can be inconvenient when it is desired to resample the signal at arbitrary times or change the conversion factor smoothly over time.

In this paper, a resampling algorithm is proposed which allows signal evaluation 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 [5], 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, and implementation details are discussed in section 3. Finally, section 4 discusses numerical requirements on the length, width, and interpolation accuracy of the filter coefficient table.

2. Theory of Operation

We review briefly the "analog interpretation" of sampling rate conversion [5] on which our 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, $x(t)$ 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

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

To resample $x(t)$ at a new sampling rate $F'_s = 1/T'_s$, we need only evaluate (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, $h_s(t) = \min\{1, F'_s/F_s\} \operatorname{sinc}(\pi \min\{F_s, F'_s\}t)$, where the scale factor maintains unity gain in the passband.

3. 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 1 shows the time register t,

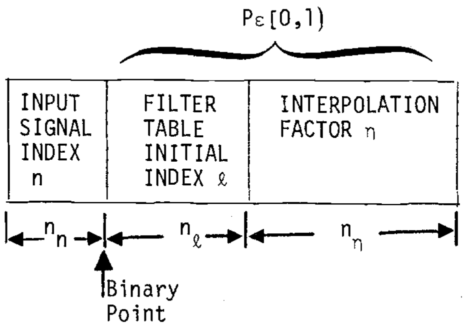

Figure 1. Time register format.

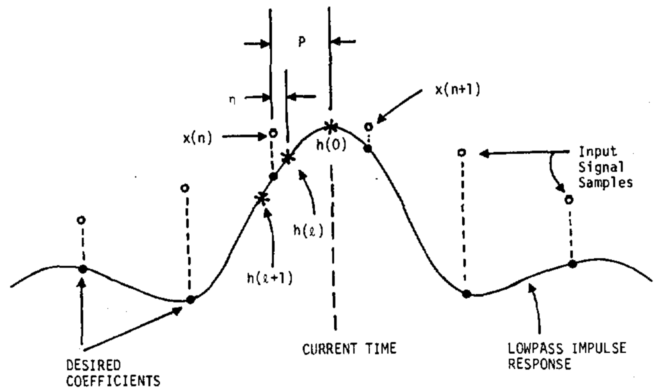

Figure 2. Illustration of waveforms and parameters in the interpolator.

and Fig. 2 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 is 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 $h(l) = \text{sinc}(\pi l/L)$. 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.

Our implementation stores only the "right wing" of a symmetric finite-impulse-response (FIR) filter (designed by the window method using a Kaiser window [4]). 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 $N_h = L\,N_z$.

Consider a sampling-rate conversion by the factor $\rho = F'_s/F_s$. For each output sample, the basic interpolation equation (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

$$
\begin{aligned}
v &\leftarrow \sum_{i=0}^{h\text{ end}} x(n-i)\big[h(l+iL) + \eta\overline{h}(l+iL)\big] \\
P &\leftarrow \text{One's-Complement}(P) \qquad\qquad (3) \\
y(t) &\leftarrow v + \sum_{i=0}^{h\text{ end}} x(n+1+i)\big[h(l+iL) + \eta\overline{h}(l+iL)\big],
\end{aligned}
$$

where $x(n)$ is the current input sample, and $\eta \in [0,1)$ is the interpolation factor. When $\rho < 1$, 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 $\rho < 1$.

When $\rho < 1$, more input samples are required to reach the end of the filter table, thus preserving the filtering quality. The number of computations (2 multiplies, 2 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 $\rho \rightarrow 0$, 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., $\rho_{\max} = 2^{n_l+n_\eta}$.

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, need not use the linear interpolation feature (because we may take $\eta \equiv 0$). In this case our method reduces to the normal type of bandlimited interpolator [5]. With the availability of interpolated lookup, however, the range of conversion factors is boosted to the order of $2^{n_l+n_\eta}/M$.

4. 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 [1, p. 119], we have

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

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

$$\hat{h}(t) \triangleq \overline{\eta} h(t_0) + \eta h(t_1), \qquad (5)$$

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 $h(t_0 + \eta) = \overline{\eta} h(t_0 + \eta) + \eta h(t_1 - \overline{\eta})$, applying (4) to both terms on the right-hand side, and subtracting (5) gives

$$\tilde{h}(t_0 + \eta) = \eta\overline{\eta}\left[h'(t_0) - h'(t_1) + \frac{\eta h''(\xi_0) + \overline{\eta} h''(\xi_1)}{2}\right], \qquad (6)$$

where $\xi_0, \xi_1 \in [t_0, t_1]$. Defining $M_2 \triangleq \max_t |h''(t)|$ and noting that $h'(t_1) = h'(t_0) + h''(t_0 + \lambda)$ for some $\lambda \in [0, 1]$ which implies $|h'(t_0) - h'(t_1)| \le M_2$, we obtain the upper bound

$$\left|\tilde{h}(t_0 + \eta)\right| \le \eta\overline{\eta}\left[M_2 + \frac{M_2}{2}\right] \le \frac{3}{8} M_2. \qquad (7)$$

For the ideal lowpass filter, we have

$$h(t) = \text{sinc}(\omega_L t) \triangleq \frac{\sin(\omega_L t)}{\omega_L t} = \frac{1}{\omega_L}\int_0^{\omega_L} \cos(\omega t) d\omega, \qquad (8)$$

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 (8) is simply the inverse Fourier transform of the ideal lowpass-filter frequency response. Twice differentiating with respect to t, we obtain

$$h''(t) = -\frac{1}{\omega_L}\int_0^{\omega_L} \omega^2 \cos(\omega t) d\omega, \qquad (9)$$

from which it follows that the maximum magnitude is

$$M_2 = \frac{\omega_L^2}{3} = \frac{\pi^2}{3L^2}. \qquad (10)$$

Note that this bound is attained at $t = 0$. Substituting (10) into (7), we obtain the error bound

$$\left|\tilde{h}(t_0 + \eta)\right| \le \frac{\pi^2}{8L^2} < \frac{1.234}{L^2} = 1.234 \cdot 2^{-2n_l}. \qquad (11)$$

Thus for the ideal lowpass filter $h(t) = \text{sinc}(\pi t/L)$, 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. 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 $L \sim 2^{16} = 65536$. Thus 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 maximum second derivative M_2 is determined primarily by the bandwidth of the filter, as equations (8) and (9) indicate.

Table-Entry 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

$$h_q(t_0) = -b_0 + \sum_{i=1}^{n_c - 1} b_i 2^{-i}, \qquad (12)$$

where $b_i \in \{0, 1\}$ is the ith bit, and the worst-case rounding error is $|h(t) - h_q(t)| \le 2^{-n_c}$. Letting $h_q(t_i) = h(t_i) + \epsilon_i$, where $|\epsilon_i| \le 2^{-n_c}$, the interpolated look-up becomes

$$\hat{h}_q(t_0 + \eta) = \overline{\eta} h_q(t_0) + \eta h_q(t_1) = \hat{h}(t_0 + \eta) + \overline{\eta}\epsilon_0 + \eta\epsilon_1. \qquad (13)$$

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

$$\left|\tilde{h}_q(t)\right| \le \frac{3}{8} M_2 + 2^{-n_c}, \qquad (14)$$

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

$$\left|\tilde{h}_q(t)\right| < \frac{1.234}{L^2} + 2^{-n_c} = 1.234 \cdot 2^{-2n_l} + 2^{-n_c}. \qquad (15)$$

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

$$\begin{aligned} t &= t_0 + \eta, \qquad |\eta| \le \frac{1}{2} \\ \hat{h}(t) &= h(t_0) \\ \tilde{h}(t) &= h(t) - h(t_0) \\ &= \eta h'(t_0) + \frac{1}{2}\eta^2 h''(t_0 + \lambda\eta) \\ \left|\tilde{h}(t)\right| &\le \frac{M_1}{2} + \frac{M_2}{8}, \end{aligned} \tag{16}$$

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

$$h'(t) = -\frac{1}{\omega_L}\int_0^{\omega_L} \omega \sin(\omega t) d\omega = \frac{\omega_L t \cos(\omega_L t) - \sin(\omega_L t)}{\omega_L t^2}. \tag{17}$$

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

$$|h'(t)| \le \frac{a}{2L} + \frac{\pi^2}{24L^2} < \frac{0.7854}{L} + \frac{0.4113}{L^2}. \tag{18}$$

Choice of Interpolation Resolution

The quantized interpolation factor and its complement are representable as $\eta_q = \eta + \nu$, $\overline{\eta}_q = \overline{\eta} - \nu$, where, since $\eta, \overline{\eta}$ are unsigned, $|\nu| \le 2^{-(n_\eta+1)}$. The interpolated coefficient look-up then gives

$$\begin{aligned} \hat{h}_{qq}(t) &= (\overline{\eta} - \nu)[h(t_0) + \epsilon_0] + (\eta + \nu)[h(t_1) + \epsilon_1] \\ &= \hat{h}(t) + \overline{\eta}\epsilon_0 + \eta\epsilon_1 + \nu[h(t_1) - h(t_0)], \end{aligned} \tag{19}$$

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

$$\left|\tilde{h}_{qq}(t)\right| \le 2^{-n_c} + 2^{-(n_\eta+1)}M_1 + \frac{3}{8}M_2. \tag{20}$$

The three terms in (20) are caused by coefficient quantization, interpolation quantization, and linear-approximation error, respectively.

For the ideal lowpass, the error bound is

$$\left|\tilde{h}_{qq}(t)\right| \le 2^{-n_c} + a2^{-(n_l+n_\eta+1)} + \frac{\pi^2}{8}2^{-n_l}.$$

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 $n_\eta \ge n_c/2$.

5. 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

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

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

[3] J. F. Kaiser, "Nonrecursive Digital Filter Design Using the I_0-sinh Window Function," *Proc. IEEE Int. Symp. on Circuits and Syst.*, April 22–25, pp. 20–23, 1974.

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

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