

A Digital Signal Processing Approach to Interpolation

RONALD W. SCHAFFER AND LAWRENCE R. RABINER

Abstract—In many digital signal processing systems, e.g., vocoders, modulation systems, and digital waveform coding systems, it is necessary to alter the sampling rate of a digital signal. Thus it is of considerable interest to examine the problem of interpolation of bandlimited signals from the viewpoint of digital signal processing. A frequency domain interpretation of the interpolation process, through which it is clear that interpolation is fundamentally a linear filtering process, is presented.

An examination of the relative merits of finite duration impulse response (FIR) and infinite duration impulse response (IIR) digital filters as interpolation filters indicates that FIR filters are generally to be preferred for interpolation. It is shown that linear interpolation and classical polynomial interpolation correspond to the use of the FIR interpolation filter. The use of classical interpolation methods in signal processing applications is illustrated by a discussion of FIR interpolation filters derived from the Lagrange interpolation formula. The limitations of these filters lead us to a consideration of optimum FIR filters for interpolation that can be designed using linear programming techniques. Examples are presented to illustrate the significant improvements that are obtained using the optimum filters

I. INTRODUCTION

THE PROCESS of interpolation is familiar to anyone who has had occasion to "read between the lines" in a table of mathematical functions. In digital signal processing, interpolation is required whenever it is necessary to change from one sampling rate to another. For example, in speech processing systems, estimates of speech parameters are often computed at a low sampling rate for low bit-rate storage or transmission; however, for constructing a synthetic speech signal from the low bit-rate representation, the speech parameters are normally required at much higher sampling rates [1], [2]. In such cases, the sampling rate must be increased by a digital interpolation process. As another example, an efficient digital realization of a frequency-multiplexed single-sideband system has been obtained [3] by performing complicated filtering functions at a low sampling rate and simpler functions at the high sampling rate required for grouping several channels into a frequency-multiplexed format. In this process, there is a need for both increasing and decreasing the sampling rate. Another example where sampling rate reduction is required is in converting a delta modulation representation of a waveform to a pulse-code modulation (PCM) representation [4].

In these and other examples, it is important to thoroughly understand the process of interpolation from the point of view of digital signal processing rather than from a numerical analysis viewpoint. For example, tables of mathematical functions are generally constructed so that linear interpolation produces sufficiently accurate results, and for cases where linear interpolation is inadequate, there exists a great variety of higher order polynomial interpolation formulas. In signal processing applications there is a great temptation to try to get along with linear interpolation because it is a simple technique. In this paper we present a frequency-domain interpretation of the interpolation process in which it is clear that interpolation is fundamentally a linear filtering process. This

discussion makes it abundantly clear that linear interpolation is generally not appropriate for digital signal processing applications. We discuss the advantages of finite duration impulse response (FIR) over infinite duration impulse response (IIR) digital filters for use as interpolation filters and we discuss the application of recently developed design techniques for FIR filters to the design of optimum interpolation filters. These filters are compared to filters derived from classical polynomial interpolation formulas to illustrate the improvement that can be achieved.

II. DIGITAL SAMPLING RATE ALTERATION

A. Sampling Continuous-Time Signals

Consider a continuous-time signal $x(t)$ with Fourier transform

$$\hat{X}(\omega) = \int_{-\infty}^{\infty} x(t)e^{-j\omega t} dt.$$

The signal $x(t)$ is sampled to produce the sequence

$$x(n) = x(nT), \quad -\infty < n < \infty$$

where T is the sampling period. The z transform of the sequence $x(n)$ is defined as

$$X(z) = \sum_{n=-\infty}^{\infty} x(n)z^{-n}.$$

The z transform evaluated on the unit circle $X(e^{j\omega T})$ will be called the *Fourier transform* of the sequence $x(n)$. It is well known that the Fourier transform of the sequence $x(n)$ is related to the Fourier transform of $x(t)$ by [5]

$$X(e^{j\omega T}) = \frac{1}{T} \sum_{k=-\infty}^{\infty} \hat{X}\left(\omega + k \frac{2\pi}{T}\right). \quad (1)$$

If $x(t)$ is bandlimited, i.e., $\hat{X}(\omega) = 0$ for $|\omega| \geq \Omega$, and if $T \leq \pi/\Omega$, then it can be seen from (1) that

$$X(e^{j\omega T}) = \frac{1}{T} \hat{X}(\omega), \quad -\frac{\pi}{T} \leq \omega \leq \frac{\pi}{T}$$

as depicted in Fig. 1 where $T = \pi/\Omega$.

Assuming that $x(t)$ is bandlimited, the original continuous-time signal can be obtained uniquely from the samples $x(n)$ through the interpolation formula

$$x(t) = \sum_{k=-\infty}^{\infty} x(k) \frac{\sin \left[\frac{\pi}{T} (t - kT) \right]}{\frac{\pi}{T} (t - kT)}. \quad (2)$$

In many digital signal processing problems, we are given a sequence $x(n)$, corresponding to sampling period T , and we must obtain from the sequence $x(n)$ a sequence $y(n) = x(nT')$; i.e., the sequence $y(n)$ corresponds to sampling $x(t)$ at a different sampling rate. If we evaluate (2) for $t = nT'$, we obtain

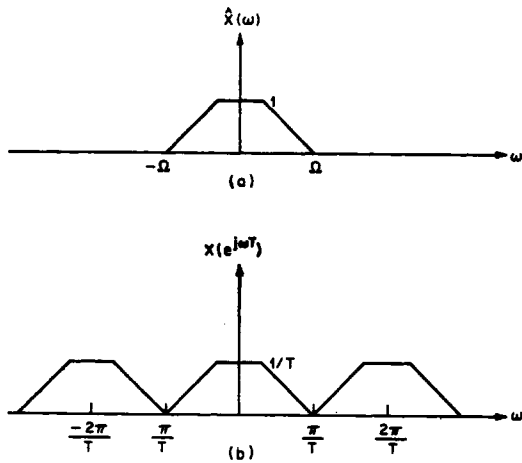


Fig. 1. Illustration of the relationship between the Fourier transform of a continuous-time signal and the Fourier transform of the sequence obtained by sampling with period T .

a direct relationship between $y(n)$ and $x(n)$, but it is clear that such an equation is impossible to evaluate because the functions $\sin [(\pi/T)(t-kT)]/[(\pi/T)(t-kT)]$ are of infinite duration. Rather than simply truncate these functions, it is more reasonable to design finite duration interpolators. To understand how such interpolators can be designed and to understand the limitations of classical interpolators, it is useful to consider the frequency-domain representation of the process of changing the sampling rate.

B. Sampling Rate Reduction—Integer Factors

Suppose that the desired sampling period is $T' = MT$. If M is an integer, this simply implies that the new sequence is

$$y(n) = \hat{x}(nT') = \hat{x}(nMT) = x(Mn).$$

That is, the sequence $x(n)$ is "sampled" by retaining only one out of each group of M consecutive samples. The values of the sequence y are samples of the original waveform $\hat{x}(t)$; however, these samples will uniquely determine $\hat{x}(t)$ if and only if $T' \leq \pi/\Omega$. This is clearly just a consequence of the sampling theorem as expressed by (1).

Since we are interested in direct relationships between the sequences $y(n)$ and $x(n)$, it is instructive to derive an equation similar to (1) that relates the Fourier transforms of the two sequences. The derivation of the equation is facilitated by the definition of a new sequence $w(n)$ which is nonzero only at integer multiples of M ; that is

$$w(n) = x(n), \quad n = 0, \pm M, \pm 2M, \dots \\ = 0, \quad \text{elsewhere}$$

where the sampling period is assumed to be T for both sequences. A convenient representation of $w(n)$ is

$$w(n) = x(n) \left\{ \frac{1}{M} \sum_{l=0}^{M-1} e^{j(2\pi/M)ln} \right\}, \quad -\infty < n < \infty$$

where the term in brackets may be recognized as a discrete Fourier series representation of a periodic sequence that is one at integer multiples of M and zero otherwise. The sequence $y(n)$, corresponding to sampling period $T' = MT$, is

$$y(n) = w(Mn), \quad -\infty < n < \infty.$$

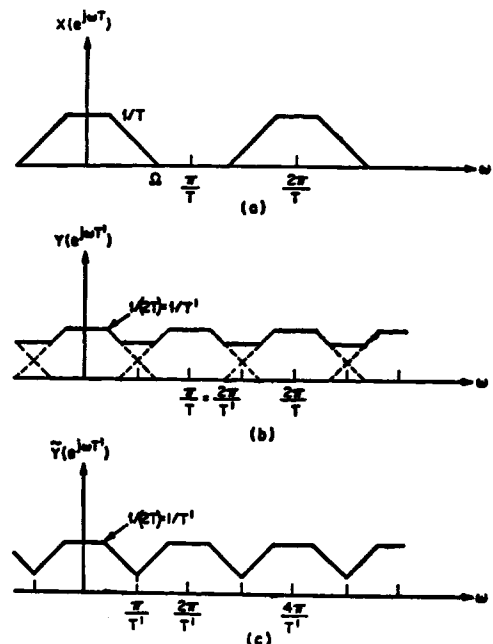


Fig. 2. Sampling rate reduction ($T' = 2T$). (a) Fourier transform of original sequence $x(n)$. (b) Fourier transform of sequence $y(n) = x(2n)$ showing aliasing. (c) Fourier transform obtained after sampling rate reduction of a low-pass filtered version of $x(n)$.

Therefore

$$Y(z) = \sum_{n=-\infty}^{\infty} w(Mn)z^{-n}.$$

Since $w(n)$ is zero except at integer multiples of M , we obtain

$$Y(z) = \sum_{n=-\infty}^{\infty} w(n)z^{-n/M} \\ = \sum_{n=-\infty}^{\infty} x(n) \frac{1}{M} \sum_{l=0}^{M-1} e^{j(2\pi/M)ln} z^{-n/M} \\ = \frac{1}{M} \sum_{l=0}^{M-1} \sum_{n=-\infty}^{\infty} x(n) e^{j(2\pi/M)ln} z^{-n/M} \\ Y(z) = \frac{1}{M} \sum_{l=0}^{M-1} X(e^{-j(2\pi/M)l} z^{1/M}). \quad (3)$$

If we evaluate $Y(z)$ on the unit circle, with normalization appropriate for the new sampling rate, we obtain

$$Y(e^{j\omega T'}) = \frac{1}{M} \sum_{l=0}^{M-1} X(e^{j(\omega T' - 2\pi l)/M}). \quad (4)$$

There is a clear similarity between (4) and (1).

An example of sampling rate reduction by a factor of 2 is shown in Fig. 2. The Fourier transform of $x(n)$ is depicted in Fig. 2(a) for the case when $\pi/2\Omega < T < \pi/\Omega$ so that

$$X(e^{j\omega T}) = \frac{1}{T} \hat{X}(\omega), \quad -\frac{\pi}{T} < \omega \leq \frac{\pi}{T}.$$

Fig. 2(b) shows $Y(e^{j\omega T'})$ for $T' = 2T$. In this case, aliasing occurs and it is clear that, in general, aliasing will occur in the process of digital sampling rate reduction unless the original sampling period satisfies

$$T \leq \frac{\pi}{M\Omega} \quad (5)$$

where Ω is the Nyquist frequency of $\hat{x}(t)$. If this inequality is satisfied, however, then

$$\begin{aligned} Y(e^{j\omega T'}) &= \frac{1}{M} X(e^{j\omega T}) \\ &= \frac{1}{MT} \hat{X}(\omega) \\ &= \frac{1}{T'} \hat{X}(\omega), \quad -\frac{\pi}{T'} < \omega < \frac{\pi}{T'} \end{aligned}$$

If the original sampling period does not satisfy (5), aliasing distortion can be avoided only by passing the sequence $x(n)$ through an ideal low-pass digital filter with cutoff frequency π/T' . It can be seen that the filter must have unit gain, since (4) provides the factor $1/M$ needed to correct the amplitude for the new sampling rate. This, of course, results in a sequence $y(n)$ corresponding to a continuous time signal $\hat{y}(t)$ which is a low-pass filtered version of the original signal $\hat{x}(t)$.

C. Sampling Rate Increase—Integer Factors

If the sampling rate is increased by an integer factor L , then the new sampling period will be $T' = T/L$. Since the sequence $x(n)$ provides samples of the desired sequence only at intervals of L samples at the new sampling rate, the remaining samples must be filled in by interpolation. To see how this can be done using a digital filter, consider the sequence

$$\begin{aligned} v(n) &= x(n/L), \quad n = 0, \pm L, \pm 2L, \dots \\ &= 0, \quad \text{otherwise.} \end{aligned}$$

The z transform of $v(n)$ is

$$\begin{aligned} V(z) &= \sum_{n=-\infty}^{\infty} x(n/L)z^{-n} \\ &= \sum_{n=-\infty}^{\infty} x(n)z^{-Ln} = X(z^L). \end{aligned} \quad (6)$$

The Fourier transform of this sequence is

$$\begin{aligned} V(e^{j\omega T'}) &= X(e^{j\omega T' L}) \\ &= X(e^{j\omega T}). \end{aligned}$$

Thus $V(e^{j\omega T'})$ is periodic with period $2\pi/T = 2\pi/LT'$, rather than $2\pi/T'$ as is the case in general for sequences associated with a sampling period T' . Fig. 3(a) shows $V(e^{j\omega T'})$ for the case $T' = T/3$. If we wish to obtain a sequence $y(n)$ such that

$$y(n) = \hat{x}(nT')$$

then we must insure that

$$Y(e^{j\omega T'}) = \frac{1}{T'} \hat{X}(\omega), \quad -\frac{\pi}{T'} \leq \omega \leq \frac{\pi}{T'} \quad (7)$$

Assuming that

$$X(e^{j\omega T}) = \frac{1}{T} \hat{X}(\omega), \quad -\frac{\pi}{T} \leq \omega \leq \frac{\pi}{T}$$

then it is clear from Fig. 3(a), that the images of $(1/T)\hat{X}(\omega)$

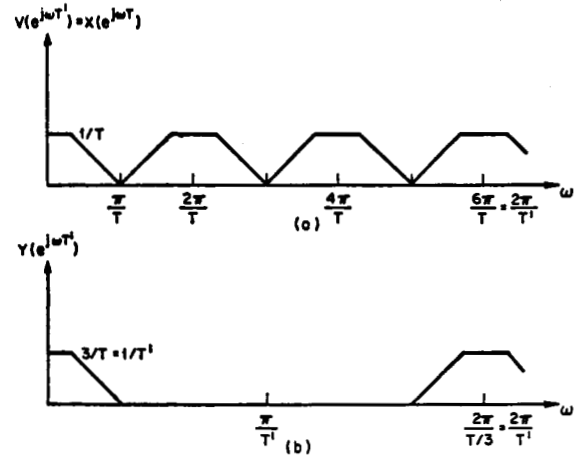


Fig. 3. Sampling rate increase ($T' = T/3$). (a) Fourier transform of sequences $x(n)$ and $v(n)$. (b) Fourier transform of desired output of interpolation process.

in $V(e^{j\omega T'})$ that are centered at $\omega = 2\pi/T$ and $4\pi/T$ must be removed by a digital low-pass filter that rejects all frequencies in the range $\pi/T < |\omega| < \pi/T'$. Furthermore, to insure that the amplitude is correct for sampling interval T' , the gain of the filter must be L . That is,

$$\begin{aligned} Y(e^{j\omega T'}) &= H(e^{j\omega T'})V(e^{j\omega T'}) = H(e^{j\omega T'})X(e^{j\omega T}) \\ &= \frac{1}{T} H(e^{j\omega T'}) \hat{X}(\omega) \end{aligned}$$

where $H(e^{j\omega T'})$ is periodic with period $2\pi/T'$ and

$$\begin{aligned} H(e^{j\omega T'}) &= L, \quad |\omega| \leq \frac{\pi}{T} \\ &= 0, \quad \frac{\pi}{T} < |\omega| < \frac{\pi}{T'}. \end{aligned} \quad (8)$$

Thus the ideal interpolation scheme for increasing the sampling rate requires the creation of a sequence of $L-1$ zero-valued samples between each value of the original sequence, which is then filtered with an ideal low-pass filter as in (8).

D. Changing by Noninteger Factors

In the previous two subsections we have discussed methods for increasing or decreasing the sampling rate using a linear time-invariant digital filter. However, because of the fact that we required that the process be entirely a discrete-time process, the new sampling period T' was restricted to be either an integer multiple or submultiple of the original sampling period T . This restriction can be eased somewhat by a two-step process involving a sampling rate increase followed by a decrease.

Suppose the desired sampling period is $T' = (M/L)T$, where M and L are integers. This does not seem to be a significant limitation in practice since any factor can be approximated as closely as desired by proper choice of M and L . The two-step process for interpolating to a sampling period $T' = (M/L)T$ is as follows:

- 1) Increase the sampling rate by a factor of L by interpolation as in Section II-C. Let the resulting sequence be denoted $y_1(n)$ with sampling period $T_1 = T/L$.
- 2) Decrease the sampling rate by a factor of M as in Section II-B to obtain the desired sequence $y(n)$ with sampling period $T' = MT_1 = (M/L)T$.

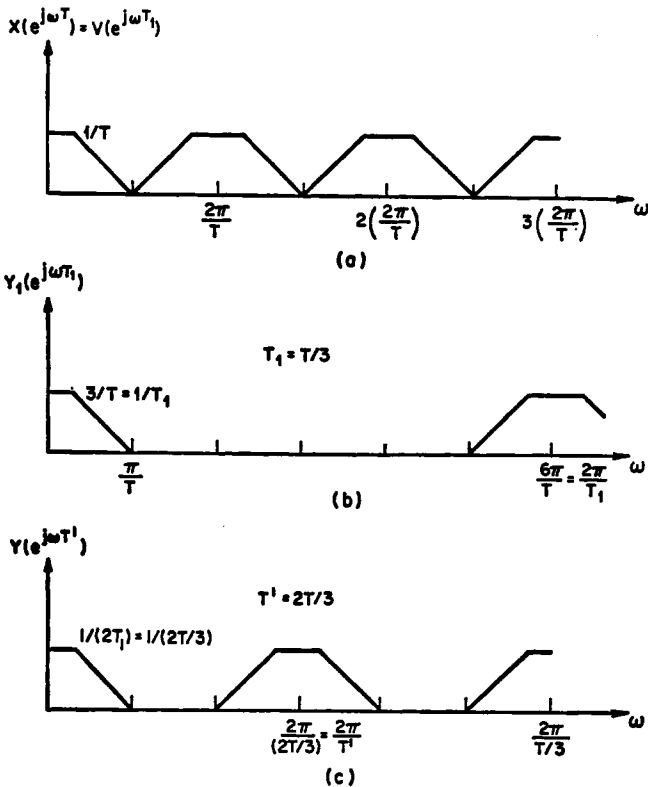


Fig. 4. Sampling rate change by noninteger factor ($T' = \frac{2}{3}T$). (a) Fourier transform of $x(n)$ and $v(n)$. (b) Fourier transform for increase of sampling rate by a factor of 3. (c) Fourier transform after sampling rate reduction by factor of 2.

This process is illustrated in Fig. 4 for $T' = \frac{2}{3}T$; i.e., for a net increase in sampling rate. Fig. 4(a) shows the Fourier transform of the original sequence $x(n)$. Fig. 4(b) shows the Fourier transform of the intermediate sequence $y_1(n)$ which results from filtering the sequence

$$v(n) = x(n), \quad n = 0, \pm 3, \pm 6, \dots$$

$$= 0, \quad \text{elsewhere}$$

with an ideal low-pass filter having gain 3 and cutoff frequency π/T . The result of reducing the sampling rate of $y_1(n)$ by a factor of 2 is shown in Fig. 4(c).

If $M < L$, i.e., there is a net increase in sampling rate as in Fig. 4, no aliasing can occur, and the interpolation filter can have a cutoff frequency π/T . However if $M > L$, there is a possibility of aliasing. In this case aliasing can be avoided by making the cutoff frequency of the interpolation filter equal to π/T' . As in Section II-B, the resulting output sequence $y(n)$ will correspond to samples of a low-pass filtered version of the original continuous time signal.

III. INTERPOLATION USING FIR DIGITAL FILTERS

The previous section makes it abundantly clear that the process of changing the sampling rate requires a low-pass filter. Since it is impossible to realize the ideal low-pass filter that is required for exact results, we must consider digital filters that approximate this ideal behavior. As in all filtering problems, there are many important considerations. A basic consideration is the choice between filters from the class of FIR filters and from the class of IIR filters. Given the type of filter to be used, the problem of approximating the ideal low-pass filter must be solved. Finally, there are important consid-

erations in how the filter is realized as software or hardware. All of these facets of the problem are interrelated—resulting in arbitrary tradeoffs between accuracy of interpolation and efficiency of realization. In this section we present some observations on filter design and realization that seem to imply that FIR filters are the proper choice for most interpolation problems.

A. Phase Distortion

The ideal interpolator has zero phase or at most a linear phase corresponding to an integer number of samples of delay. IIR filters cannot have precisely linear phase [6]. In contrast, there currently exist several techniques for designing optimal FIR digital filters with precisely linear phase. These filters are optimal in the sense that the width of transition band between passband and stopband is minimum for given values of passband and stopband ripple and specified passband and stopband cutoff frequencies [7]–[11]. These filters can be designed with arbitrarily small values for passband ripple, stopband ripple, and transition bandwidth, at the cost of increased impulse response duration. Thus with FIR filters, the interpolation error due to phase nonlinearity can be zero and the error due to amplitude distortion can be made arbitrarily small. In the case of IIR filters, although extremely good amplitude characteristics can be achieved, there will always be an interpolation error due to phase nonlinearity.

B. Filter Realization

IIR filters have recursive realizations that are very economical in terms of computational complexity. Leaving phase considerations aside, FIR filters in general require more computation to achieve a given accuracy of approximation to the desired amplitude response than do IIR filters. However, the particular nature of the interpolation problem makes FIR filters computationally competitive with IIR filters.

Consider for convenience a zero-phase FIR filter with impulse response $h(n)$ which is nonzero in the interval

$$-\left(\frac{N-1}{2}\right) \leq n \leq \left(\frac{N-1}{2}\right)$$

where N is an odd integer.¹ In reducing the sampling rate by an integer factor, i.e., $T' = MT$, we may need a unity gain low-pass filter to insure that no aliasing occurs in retaining only every M th sample of the sequence $x(n)$. In this case computation is reduced because of the nature of the desired output sequence. The filtered output sequence at the original sampling rate, defined as $y(n)$, is

$$y(n) = \sum_{k=-((N-1)/2)}^{((N-1)/2)} h(k)x(n-k) \quad (9)$$

where all sequences in (9) are associated with sampling period T . Clearly, all values of the sequence $y(n)$ need not be computed since the desired output is

$$y(n) = y(nM) = \sum_{k=-((N-1)/2)}^{((N-1)/2)} h(k)x(nM-k) \quad (10a)$$

where $y(n)$ is associated with a sampling period $T' = MT$. This is in contrast to a comparable IIR filter where the computations required to realize the poles of the system function would have to be carried out at the original sampling rate even

¹ See Section III-C for a comment on why N should be odd.

though $M-1$ out of M samples of $y(n)$ would be discarded. A further simplification results from the fact that zero-phase FIR filters have the property

$$h(n) = h(-n).$$

Thus (10a) becomes

$$y(n) = \sum_{k=1}^{((N-1)/2)} h(k)[x(nM-k) + x(nM+k)] + h(0)x(nM). \quad (10b)$$

In the case of sampling rate increase, the interpolated output is obtained with sampling period $T' = T/L$ by filtering the sequence

$$v(n) = x(n/L), \quad n = 0, \pm L, \pm 2L, \dots \\ = 0, \quad \text{otherwise.}$$

In this case it is convenient to write

$$y(n) = \sum_{k=n-((N-1)/2)}^{n+((N-1)/2)} v(k)h(n-k). \quad (11)$$

Substituting for $v(k)$ results in

$$y(n) = \sum_{k=n-((N-1)/2)}^{n+((N-1)/2)} x(k/L)h(n-k), \quad k/L \text{ an integer} \\ = \sum_{k=\lfloor (n/L) - ((N-1)/2L) \rfloor}^{\lfloor (n/L) + ((N-1)/2L) \rfloor} x(k)h(n-kL) \quad (12)$$

where $\lfloor a \rfloor$ means "the largest integer contained in a ." Although (11) suggests that computation required for each output sample is proportional to N , we note that only one out of every L samples of $v(n)$ is nonzero. Thus we see from (12) that the actual computation required is proportional to N/L . Note that in this case the symmetry of the impulse response cannot be exploited to reduce computation.

If an IIR filter were used, relatively little saving could be achieved. In fact, in a cascade realization, almost no computational saving could be attained by considering the zeroes in the input sequence.

Changing the sampling period according to $T' = (M/L)T$, requires that we first increase the sampling rate by a factor L and then reduce it by a factor M . Clearly, the savings previously discussed could be incorporated into both steps of this process.

C. Impulse Response Constraints

The previous discussion has presented compelling reasons for the use of linear-phase FIR filters in interpolation. To conclude this section we discuss some constraints on the impulse response that are specific to the problem of interpolation. Recall that the output $y(n)$ is given by (12). A reasonable requirement on the interpolation filter is that the values of the output at the original sampling times be the same as the original samples.³ That is, for r an integer,

$$y(rL) = x(rL/L) = x(r), \quad -\infty < r < \infty.$$

Equation (12) implies that

$$y(rL) = x(r) = \sum_{k=\lfloor (r-L) - ((N-1)/2L) \rfloor}^{\lfloor (r-L) + ((N-1)/2L) \rfloor} x(k)h(rL-kL) \quad (13)$$

³ This is also a property of (2), as can be easily verified.

for all integer values of r . From this equation, we see that

$$h(0) = 1$$

$$h(n) = 0, \quad n = \pm L, \pm 2L, \dots, \left\lfloor \frac{N-1}{2L} \right\rfloor L. \quad (14)$$

Constraints of this type are rather difficult to impose on an IIR filter design procedure.

Some final comments on the choice of the length of the impulse response N are in order. First, we have assumed that N is an odd integer; however, in general N can be either even or odd for the FIR filter. It can be shown [6], [12] that for N even, a linear-phase FIR filter must have a delay of at least one-half sample. This one-half sample delay itself corresponds to interpolation between samples; thus such a filter could not preserve the samples of the original sequence. Although there may be instances where the half-sample delay may not be objectionable, or may even be desirable, odd values of N appear to be appropriate for most applications.

A second comment regarding the choice of N concerns the fact that we have asserted that by increasing N , we can achieve a better approximation in the frequency domain to the ideal interpolating filter. In the time domain, increasing N implies that more values of the original sequence are involved in the computation of a given interpolated sample and thus we should expect increasingly better results as N increases. We have observed that because the sequence $v(n)$ has mostly zero samples, the computation is proportional to N/L , since the impulse response always spans approximately N/L nonzero samples. Suppose that we wish to always have Q samples of the original input sequence involved in the computation of each interpolated sample. Using (12) it is easily shown that the length of the impulse response should be

$$N = QL, \quad \text{if } Q \text{ and } L \text{ are odd} \\ = QL - 1, \quad \text{if either } Q \text{ or } L \text{ are even.} \quad (15)$$

If N is slightly larger than this value, the computation of some interpolated samples will involve $Q+1$ original samples while the rest will involve only Q . If N is slightly smaller than the value given by (15), then the interpolation will involve either Q or $Q-1$ samples. Thus to insure that Q samples of the original sequence are always involved, N should satisfy (15).

This constraint is satisfied, as we will see, for filters derived from classical interpolation formulas. It is also important in hardware and software realizations of interpolation filters where it allows the computations to be structured so that one does not have to check for zero samples in order to eliminate multiplications by zero.

IV. CLASSICAL POLYNOMIAL INTERPOLATION

In this section we apply the previous discussion of the interpolation process to a study of classical polynomial interpolation methods. Our aim is to give a frequency domain interpretation of these formulas that will shed some light on their applicability in interpolation problems arising in digital signal processing. In the course of this discussion, we shall indicate how to derive linear phase interpolation filters from tables of Lagrange coefficients. We begin with a discussion of linear interpolation.

A. Linear Interpolation

Linear interpolation involves only two consecutive samples of the original sequence $x(n)$ in the computation of an

interpolated sample. Specifically, the values interpolated between two samples $x(0)$ and $x(1)$ lie on a straight line connecting the two original samples,³ with the original samples of course being preserved. Thus the equation relating the output $y(n)$, having sampling period $T' = T/L$, to the input sequence $x(n)$, having sampling period T , is

$$y(n) = x(0) + \frac{x(1) - x(0)}{LT' - 0} (nT' - 0)$$

$$= x(0) \left(1 - \frac{n}{L}\right) + x(1) \left(\frac{n}{L}\right), \quad 0 \leq n < L. \quad (16)$$

In order to interpret linear interpolation as a linear filtering process, we must derive an impulse response for the linear interpolation filter. This can be done by comparing (16) to (12). To begin, it is clear that the length of the impulse response must be

$$N = 2L - 1$$

as discussed in Section III-C. If N were larger, more than 2 samples of $x(n)$ would be involved in the computation of some of the interpolated values. Likewise, if N were smaller, only one sample of the input would enter into the computation of some of the interpolated values. Using this information, we can write (12) as

$$y(n) = x(0)h(n) + x(1)h(n - L), \quad 0 \leq n < L. \quad (17)$$

Comparing (16) and (17) we see that

$$h(n) = 1 - \frac{n}{L}, \quad 0 \leq n < L$$

$$h(n - L) = \frac{n}{L}, \quad 0 \leq n < L.$$

Thus $h(n)$ is seen to be

$$h(n) = 1 - |n|/L, \quad |n| < L$$

$$= 0, \quad \text{otherwise.} \quad (18)$$

Clearly $h(n)$ satisfies the requirements of (14) since $h(0) = 1$ and $h(n) = 0$ for $|n| > L$. The length of the impulse response is

$$N = 2L - 1$$

consistent with (15) for $Q = 2$.

Fig. 5 depicts linear interpolation as a convolution process. The sequence $v(k)$ and the triangular envelope of the impulse response $h(n-k)$ are shown for the case of $T' = T/5$. It is clear that because the impulse response has duration $N = 5(2) - 1 = 9$, only two nonzero samples of $v(k)$ are ever coincident with $h(n-k)$. Also it is clear that

$$y(n) = x(n/L), \quad n = 0, \pm L, \pm 2L, \dots$$

The system function corresponding to linear interpolation is

$$H(e^{j\omega T'}) = \frac{1}{L} \left\{ \frac{\sin[\omega L T'/2]}{\sin[\omega T'/2]} \right\}^2. \quad (19)$$

This system function is plotted in Fig. 6 for $L = 5$. (Curve

³ We can choose samples $x(0)$ and $x(1)$ without loss of generality since interpolation is a time-invariant process.

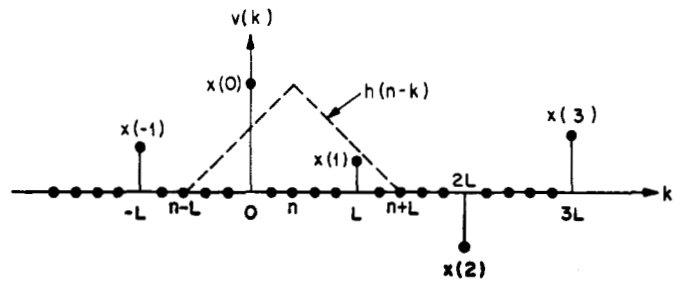


Fig. 5. Linear filtering interpretation of linear interpolation.

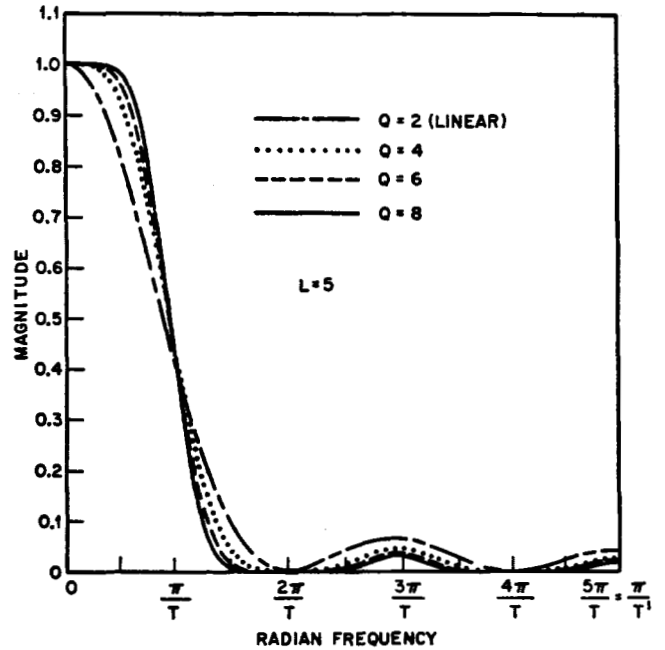


Fig. 6. Comparison of Lagrange interpolation frequency responses for $L = 5$.

labelled $Q = 2$.) We recall that the purpose of the interpolation filter is to remove the images of the signal spectrum that are centered at integer multiples of $2\pi/T$, while leaving the frequencies below π/T unaltered. It can be seen from Fig. 6, that linear interpolation achieves significant attenuation in only a very small region around each integer multiple of $2\pi/T$. Specifically, attenuation is 40 dB or greater in a band of width $0.35\pi/T$ centered at $2\pi/T, 4\pi/T$, etc. Thus it seems reasonable to note that linear interpolation is appropriate only if the original sampling rate is many times the Nyquist rate.

B. Lagrange Interpolation

Clearly linear interpolation will not be satisfactory in many digital signal processing applications. In classical numerical analysis, the inadequacies of linear interpolation lead to the use of higher order polynomials; i.e., in contrast to connecting two points by a straight line, one finds a polynomial of degree $Q - 1$ that passes through Q original samples. The interpolated values then, are samples of this polynomial. A variety of formulas have been derived for obtaining samples of the polynomial directly from the samples $x(n)$, however since we are only interested in the interpolation filter corresponding to polynomial interpolation, we shall use the most convenient form; namely, the Lagrange interpolation formula [13], [14]. The form corresponding to interpolation with equal spacing from sampling period T to $T' = T/L$ is

$$y(n) = \sum_{k=-((Q-2)/2)}^{Q/2} A_k^Q(n/L)x(k), \quad Q \text{ even} \quad (20a)$$

$$y(n) = \sum_{k=-((Q-1)/2)}^{((Q-1)/2)} A_k^Q(n/L)x(k), \quad Q \text{ odd} \quad (20b)$$

where we have again chosen to consider interpolation around some arbitrary time labelled O . In (20), the quantities $A_k^Q(\cdot)$ are called the Lagrange coefficients and are given by the equations

$$A_k^Q(t) = \frac{(-1)^{k+Q/2}}{\left(\left(\frac{Q-2}{2} + k\right)!\left(\frac{Q-2}{2} - k\right)!(t-k)\right)} \cdot \prod_{i=1}^Q \left(t + \frac{Q}{2} - i\right), \quad Q \text{ even} \quad (21a)$$

$$A_k^Q(t) = \frac{(-1)^{k+(Q-1)/2}}{\left(\left(\frac{Q-1}{2} + k\right)!\left(\frac{Q-1}{2} - k\right)!(t-k)\right)} \cdot \prod_{i=0}^{Q-1} \left(t + \left(\frac{Q-1}{2} - i\right)\right), \quad Q \text{ odd.} \quad (21b)$$

Extensive tabulations of these coefficients are available in tables of mathematical functions [15]. It is interesting to note that

$$A_k^Q(t) = 0, \quad t \text{ an integer, and } t \neq k \\ = 1, \quad t = k.$$

This is a result of the fact that the interpolation polynomial passes through the Q original data samples. That is,

$$y(kL) = x(k), \quad \text{for } k \text{ an integer.}$$

Thus the condition of (14) is satisfied for Lagrange interpolation filters.

In general, the formulas in (20) may be evaluated for any value of n such that

$$-\left(\frac{Q-2}{2}\right)L < n < \frac{Q}{2}L, \quad Q \text{ even} \\ -\left(\frac{Q-1}{2}\right)L < n < \left(\frac{Q-1}{2}\right)L, \quad Q \text{ odd.}$$

In this case the *same* Q original samples are involved in the computation of all the interpolated samples in the time interval spanned by the original samples. If we perform the interpolation with a linear filter, we can choose the length of the filter impulse response so that Q original samples are always involved in the computation; however, a given set of Q original samples is used only to compute $L-1$ interpolated samples. Thus we can interpret the Lagrange formulas in (20) in terms of $Q-1$ different impulse responses—corresponding to the $Q-1$ different interpolation intervals between the Q original samples. As an example, consider the case for $Q=3$. In comparing (12) and (20b), we note that the three samples $x(-1)$, $x(0)$, and $x(1)$ can enter into the interpolation in two

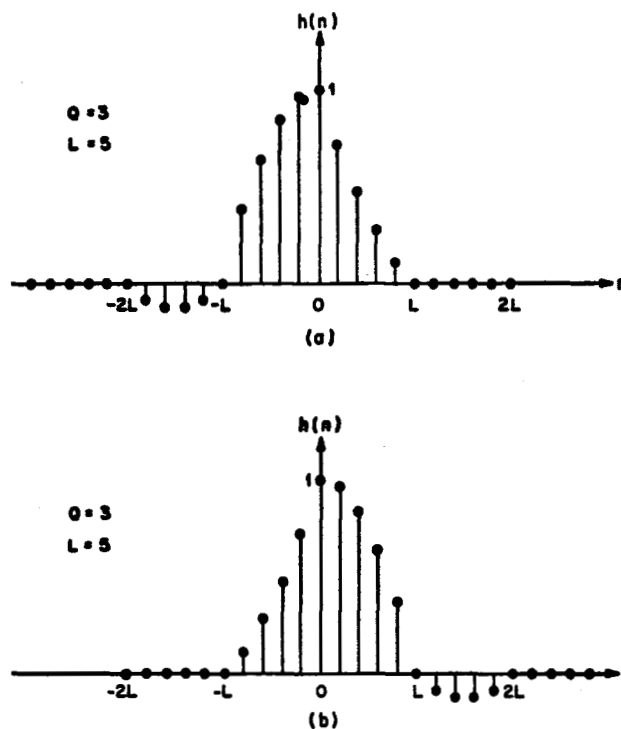


Fig. 7. Impulse responses for three-point ($Q=3$) Lagrange interpolation. (a) Interpolation in the interval $-L < n \leq 0$. (b) Interpolation in the interval $0 \leq n < L$.

intervals. Thus

$$y(n) = \sum_{k=-1}^1 A_k^Q(n/L)x(k) \\ = \sum_{k=-1}^1 x(k)h(n - kL)$$

for values of n

$$-L < n \leq 0 \\ 0 \leq n < L.$$

If we compare the above two equations, we obtain

$$h(n+L) = A_{-1}^3(n/L) \\ h(n) = A_0^3(n/L) \\ h(n-L) = A_1^3(n/L).$$

If these equations are evaluated for the first interval we obtain the impulse response of Fig. 7(a) where $L=5$. Likewise, Fig. 7(b) shows the impulse response corresponding to Lagrange interpolation in the second interval. Clearly neither of these impulse responses has linear phase, since they do not satisfy the symmetry condition $h(n) = h(-n)$. Indeed, it is easy to show that whenever Q is odd, none of the impulse responses corresponding to Lagrange interpolation can have linear phase. However, if Q is even, one of the $Q-1$ impulse responses does have linear phase.

As a second example suppose $Q=4$. Using (20a) we obtain

$$y(n) = \sum_{k=-1}^2 A_k^4(n/L)x(k). \quad (22)$$

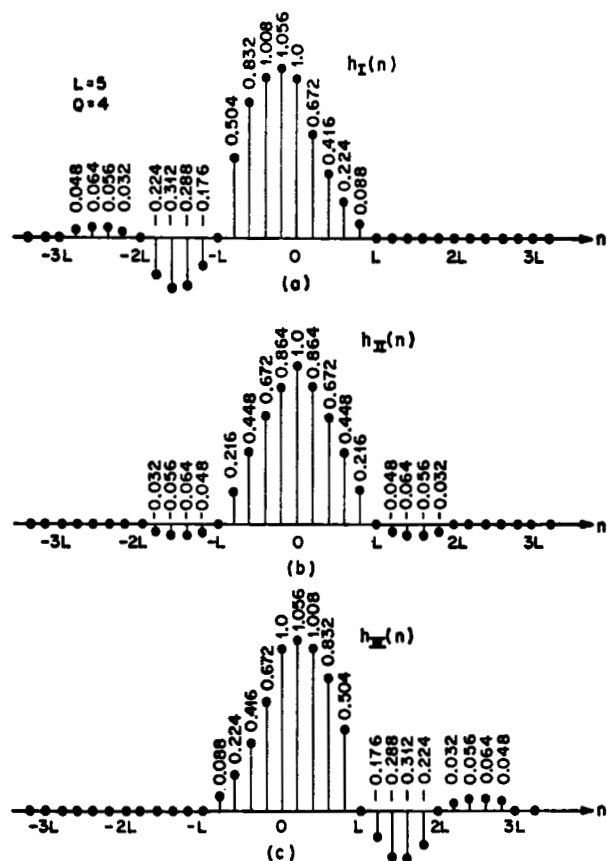


Fig. 8. Impulse responses for four-point ($Q = 4$) Lagrange interpolation.

As in the previous example we can evaluate (12) for $N = QL - 1$ and we obtain the equation

$$y(n) = \sum_{k=-1}^2 x(k)h(n - kL)$$

from which by comparison to (22) we obtain

$$\begin{aligned} h(n + L) &= A_{-1}^4(n/L) \\ h(n) &= A_0^4(n/L) \\ h(n - L) &= A_1^4(n/L) \\ h(n - 2L) &= A_2^4(n/L). \end{aligned}$$

These equations can be evaluated in the 3 intervals

- I: $-L < n \leq 0$
- II: $0 < n \leq L$
- III: $L < n \leq 2L$

to obtain the three impulse responses shown in Fig. 8 for the condition $L = 5$. The exact values for the three impulse response coefficients are also given in Fig. 8. We note that the impulse response corresponding to interpolation in the central interval $0 \leq n < L$, is symmetric and thus has zero phase. The two other impulse responses clearly do not have linear phase. Indeed, it is clear that, in general, Lagrange interpolation has phase distortion except in the Q even case and interpolation in the central interval. Thus if we wish to use impulse responses derived from the Lagrange interpolation formula for interpolation in signal processing applications we should use the conditions that yield zero phase. In general the linear

phase impulse response derived from a Q point Lagrange interpolation formula is obtained from the equations

$$\begin{aligned} h\left(n + \left(\frac{Q-2}{2}\right)L\right) &= A_{-(Q-2)/2}^Q(n/L) \\ \dots \dots \dots \dots \dots \dots \dots \\ h(n + L) &= A_{-1}^Q(n/L) \\ h(n) &= A_0^Q(n/L) \\ h(n - L) &= A_1^Q(n/L) \\ \dots \dots \dots \dots \dots \dots \dots \\ h\left(n - \frac{Q}{2}L\right) &= A_{Q/2}^Q(n/L) \end{aligned} \quad (23)$$

where $0 \leq n < L$.

In Fig. 6 we have plotted the frequency response of the zero-phase interpolators derived in this manner for $Q = 2, 4, 6, 8$, and $L = 5$. Since the impulse response duration is $N = QL - 1$, $N = 9, 19, 29, 39$ for these cases. Clearly, the effect of increasing Q is to improve the frequency response of the interpolator. Whereas the linear interpolator has 40-dB attenuation in a bandwidth of $0.35\pi/T$ centered around each integer multiple of $2\pi/T$, for the 4-point and 6-point interpolators, the bandwidths are $0.7\pi/T$ and $0.9\pi/T$, respectively, for at least 40-dB attenuation. Thus at the expense of increased computation, we can achieve a significantly better interpolation by using a filter derived using (23).

The interpolation filters derived from the Lagrange interpolation formula achieve high attenuation in a narrow band around integer multiples of $2\pi/T$ because the zeroes of the system function tend to be clustered about those frequencies. For example, in the case of linear interpolation ($Q = 2$), by looking at (19) we see that $H(z)$ has a double zero on the unit circle at integer multiples of $2\pi/T$. For $Q = 4$, we have found that there are clusters of 4 zeroes not precisely on the unit circle but close in the vicinity of $\omega = 2\pi/T, 4\pi/T$, etc. As a result, the attenuation close to frequencies $2\pi/T$ and $4\pi/T$ is very high. This is clear from Fig. 9, where the system function for the case $Q = 6$ is plotted on a log scale. However, it is also clear that between $2\pi/T$ and $4\pi/T$, the response of the Lagrange interpolation filter leaves much to be desired. Clearly, as Q gets larger, the impulse response gets longer and there are more zeroes to distribute so as to increase the attenuation and broaden the attenuation bands. This raises the question as to how we might design FIR digital filters so as to make the best use of the filter zeroes.

V. OPTIMUM FIR INTERPOLATION FILTERS

In practical situations, signals are often sampled at a rate that is only slightly higher than twice the Nyquist frequency in order to minimize the computation required for digital filtering and other signal processing procedures. In this case, the ideal interpolation filter for increasing the sampling rate has constant gain in the frequency range $0 \leq |\omega| < \pi/T$, and zero gain elsewhere. For such signals we are clearly interested in the best possible approximations to the ideal low-pass filter.

On the other hand, in situations where the original sequence $x(n)$ is derived by sampling at a rate considerably higher than twice the Nyquist frequency, we require a relatively narrow passband of constant gain and a number of stop-

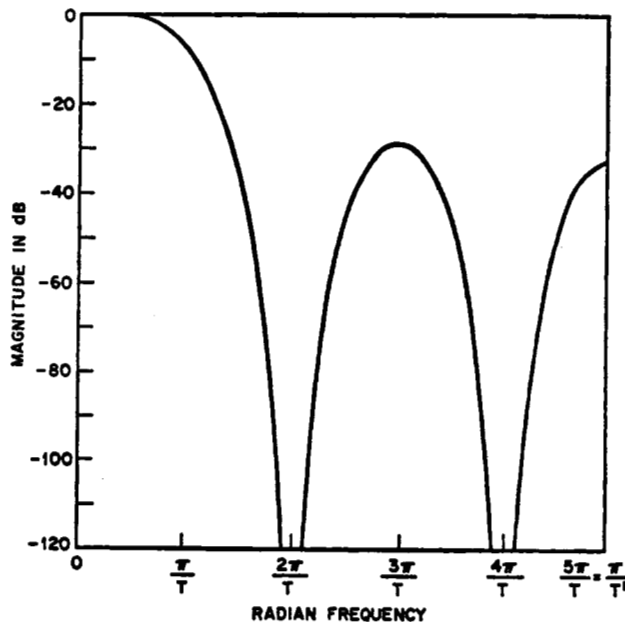


Fig. 9. Frequency response of six-point ($Q=6$) linear phase Lagrange interpolation filter. $L=5$, $N=29$.

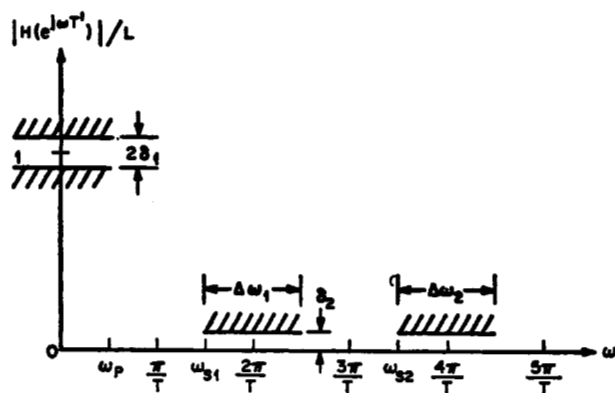


Fig. 10. General tolerance scheme for digital interpolation.

bands of zero gain, with the frequency response being somewhat arbitrary elsewhere. This means that high-order polynomial interpolation filters may be quite satisfactory for signals that are sufficiently oversampled. However, it is generally possible to achieve significantly better interpolation filters using optimization techniques.

A. Design Specifications

A scheme for approximating the ideal interpolator is shown in Fig. 10. There is a band of frequencies, $0 \leq \omega \leq \omega_p$, called the passband in which the frequency response should be close to 1.⁴ In practice we allow for an error of $\pm \delta_1$. In addition, there are one or more stopbands in which the frequency response is required to be within $\pm \delta_2$ of zero. The choice of the parameters in Fig. 10 depends upon the desired accuracy of interpolation and to some extent upon the nature of the input sequence. The choice of passband and stopbands depends upon the Nyquist frequency of the original signal. If the original sampling period is such that $\pi/T \approx \Omega$, then ω_p must be very close to π/T , and the first stopband, beginning at ω_{s1} ,

must also be close to π/T . Thus the transition band $\omega_p \leq \omega \leq \omega_{s1}$, must be very narrow, which implies that a large value of N will be required.

In such cases, it is reasonable to define only one stopband, $\omega_{s1} \leq \omega \leq \pi/T$. However, in cases where π/T is significantly higher than the Nyquist frequency, the transition band between passband and stopband can be wider, and it makes sense to define stopbands around each integer multiple of $2\pi/T$, with transition bands $\omega_{s1} + \Delta\omega_1 \leq \omega \leq \omega_{s2}$, etc., in which the frequency response is unconstrained. As can be seen from Figs. 6 and 9, this is the type of frequency response that characterizes the Lagrange interpolation filters.

B. Design Techniques

The frequency response of a zero-phase FIR filter can be expressed as

$$H(e^{j\omega T}) = h(0) + 2 \sum_{n=1}^{(N-1)/2} h(n) \cos(\omega n T). \quad (24)$$

The tolerance scheme depicted in Fig. 10 can be expressed as the following set of inequalities:

$$1 - K\delta_2 \leq H(e^{j\omega T}) \leq 1 + K\delta_2, \quad 0 \leq \omega \leq \omega_p \quad (25)$$

$$-\delta_2 \leq H(e^{j\omega T}) \leq \delta_2, \quad \omega_{s1} \leq \omega \leq \omega_{s1} + \Delta\omega_1$$

$$\omega_{s2} \leq \omega \leq \omega_{s2} + \Delta\omega_2 \quad (26)$$

where we have defined $K = \delta_1/\delta_2$ as the ratio of the passband to stopband error. For a given N and K , these equations may be evaluated on a dense set of frequencies in the specified passband and stopbands and may be solved for $\{h(n)\}$ and δ_2 using either linear programming techniques [11] or discrete Chebyshev approximation techniques [10]. It has been shown [10] that the filters designed by these techniques are optimum in the sense of having the narrowest transition bands for given δ_1 and δ_2 . For these filters, the error in the passband and stopbands exhibits an equiripple behavior. An example of a low-pass filter designed by linear programming is shown in Fig. 11. In this case $L=5$, $N=29$ ($Q=6$), $\omega_p = 0.6\pi/T$, $\omega_{s1} = 1.4\pi/T$, $K=1.0$. The resulting filter has $\delta_1 = \delta_2 = 0.00586$. (The filter gain has been normalized to 0 dB for convenience in plotting.) The equiripple behavior of the stopband is readily apparent.

The filter in this example does not satisfy the constraints on $h(n)$ given in (14); however, these constraints are linear and easily can be added to the constraints of (25) and (26). Thus in the optimization procedure $h(0)$ is constrained to 1.0 and $h(\pm L)$, $h(\pm 2L)$, etc., are set to 0. The filter performance is only slightly degraded by the addition of these constraints. Fig. 11 shows an example where all the fixed-design parameters were the same as in the previous example. In this case the resulting value of $\delta_1 = \delta_2$ was 0.00599. It can be seen from Fig. 12 that the equiripple nature of the frequency response is destroyed, but with little sacrifice in performance.

If the original signal bandwidth is much less than π/T , a bandstop filter may provide superior performance to a comparable low-pass filter. Fig. 13 shows an example for $L=5$ where $N=29$, $\omega_p = 0.3\pi/T$, $\omega_{s1} = 1.7\pi/T$, $\Delta\omega_1 = \Delta\omega_2 = 0.6\pi/T$, $\omega_{s2} = 3.7\pi/T$, and $K=1.0$. In this case $\delta_1 = \delta_2 = 0.0000681$. Thus this filter does a much better job of attenuating the images of the signal spectrum around the frequencies $2\pi/T$ and $4\pi/T$, but only if the original signal bandwidth was much less than π/T . It can be seen that the extra attenuation around $2\pi/T$ and $4\pi/T$ is obtained at the expense of the regions around

⁴ If the sampling rate is increased by a factor of L , the filter gain must approximate L in the passband instead of 1.0. This is easily accomplished by multiplying the impulse response samples by L .

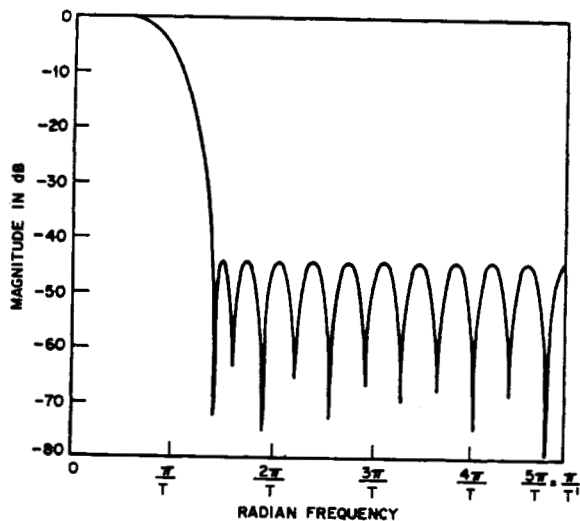


Fig. 11. Frequency response of a typical low-pass interpolation filter designed by linear programming. $L=5$, $Q=6$, $N=29$, $\omega_p=0.6\pi/T$, $\omega_{s1}=1.4\pi/T$, $\delta_1=\delta_2=0.00586$. (No impulse response constraints.)

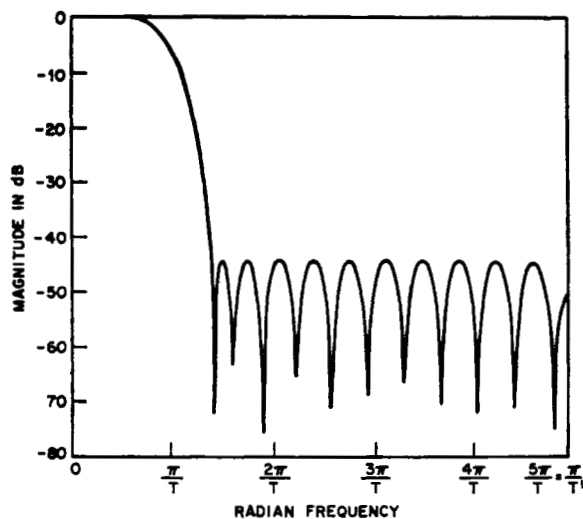


Fig. 12. Frequency response for the example of Fig. 11, with impulse response constraints. In this case $\delta_1=\delta_2=0.00599$.

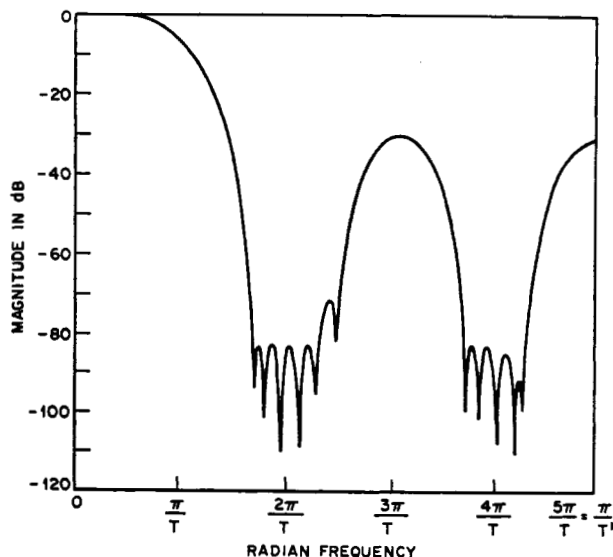


Fig. 13. Frequency response of a typical bandstop interpolation filter designed by linear programming. $L=5$, $Q=6$, $N=29$, $\omega_p=0.3\pi/T$, $\omega_{s1}=1.7\pi/T$, $\omega_{s2}=3.7\pi/T$, $\Delta\omega_1=\Delta\omega_2=0.6\pi/T$, and $\delta_1=\delta_2=0.0000681$.

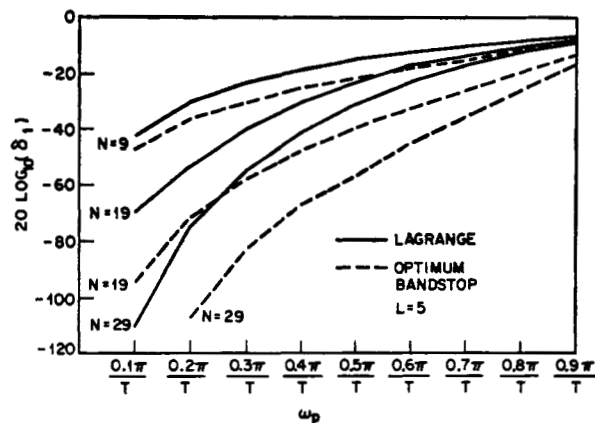


Fig. 14. Comparison of linear-phase Lagrange interpolators and optimum bandstop filters ($L=5$). Plots show minimum stopband attenuation as a function of half the stopband width. For the optimum bandstop filter ($N=29$), the curve is off scale for $\omega_p < 0.2\pi/T$.

$3\pi/T$ and $5\pi/T$ where the attenuation is much less. Thus optimum filters of this type (and classical filters) should only be used if one is certain of the bandwidth of the input sequence.

C. Comparison to Classical Interpolators

It is of considerable interest to compare the filters derived from classical interpolation formulas and those designed by linear programming. In the design of the optimum filters, the parameters of the tolerance scheme of Fig. 10 were set so that $\delta_1=\delta_2$ with symmetrical stopbands of width $\Delta\omega_1=\Delta\omega_2=\dots=2\omega_p$ centered at integer multiples of $2\pi/T$. This is reasonable since in interpolation, preservation of the passband is generally as important as rejection of stopbands. In order to compare the Lagrange filters to the optimal filters, values of δ_1 and δ_2 were measured at the edges of the passband and the first stopband, respectively. Since for the Lagrange filters, δ_1 is always greater than δ_2 for the above definition of passband and stopbands, the comparison has been made on the basis of passband error.

Fig. 14 shows such a comparison for the case $L=5$.⁵ The solid curves are for linear-phase filters derived from the Lagrange interpolation formula where $N=QL-1$ and $Q=2, 4$, and 6 . These curves show the error in decibels at the edge of the passband, i.e., $20 \log_{10} \delta_1$, as a function of ω_p . As an example, for linear interpolation ($Q=2$ and $N=9$) we see that the passband error is -42 dB for $\omega_p=0.1\pi/T$. Furthermore, since $\delta_1 > \delta_2$ we can say that the attenuation is at least 42 dB in the region $(2\pi/T - 0.1\pi/T) \leq \omega \leq (2\pi/T + 0.1\pi/T)$. For wider bandwidths, the performance is worse; however, for higher order interpolators the performance becomes appreciably better as is expected.

The dotted curves in Fig. 14 show passband error for bandstop filters designed by linear programming. By comparing corresponding curves, it can be seen that the optimum designs are always significantly better than the corresponding classical interpolator, with the improvement being most striking for narrow bandwidths and for the higher order filters.

Clearly, there are a variety of optimum designs corresponding to situations in which passband and stopband approximation errors are not treated as being of equal importance. For example if $\delta_1=K\delta_2$, then the case $K > 1$ corresponds to placing more importance on stopband attenuation than on passband

⁵ The comparisons are similar for larger values of L .

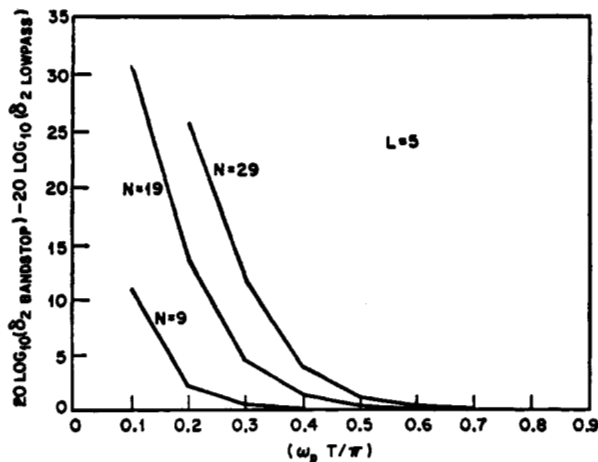


Fig. 15. Comparison of optimum bandstop and optimum low-pass interpolation filters ($L=5$). The curve for $N=29$ is off scale for $\omega_p < 0.2\pi/T$.

error. This situation would be a more favorable situation for the classical filters, although it is always possible to design a better filter using the optimum design procedures.

Another interesting comparison is between optimum low-pass filters and optimum bandstop filters. To compare these filters, we set

$$\omega_p = \frac{2\pi}{T} - \omega_{s1} = \frac{4\pi}{T} - \omega_{s2}$$

and $\Delta\omega_1 = \Delta\omega_2 = 2\omega_p$ for the bandstop filters and

$$\omega_p = \frac{2\pi}{T} - \omega_{s1}$$

and $\Delta\omega = \pi/T - \omega_{s1}$ for the low-pass filters. That is, both filters were designed to accommodate the same input signal bandwidth. Fig. 15 shows the difference between stopband attenuations for the bandstop filters and the low-pass filters as a function of bandwidth. From these curves we see that for narrow bandwidths the bandstop filters have significantly greater attenuation; however as the bandwidth approaches half the original sampling frequency, there is no difference between the two types of filters.

VI. CONCLUSIONS

In this paper we have discussed the process of interpolation as a problem in digital filtering. Most of our discussion has involved frequency-domain representations of the interpolation process and design criteria for digital interpolation filters. We have taken this approach because it is the most reasonable for digital signal processing applications where it is necessary to either raise or lower the sampling rate of a signal. This point of view is in contrast to that of interpolation in tables where one is concerned primarily with minimizing the error in a particular interpolated sample. Because of the variety of factors involved in the design of an interpolation filter, we have not tried to give design formulas and error bounds that would have limited value, but rather we have chosen to attempt to illuminate the important factors involved in the

interpolation process and to discuss general design procedures that can be adapted to a variety of situations.

In particular, we have argued that linear-phase FIR filters have many attractive features for discrete-time interpolation and have shown how they may be efficiently utilized. Classical polynomial interpolation has been discussed in the context of digital signal processing. Interpolation filters derived from polynomial interpolation formulas are attractive because the impulse response can be easily computed or looked up in a table. However, we have seen that the frequency response of such systems leaves much to be desired in digital signal processing applications where the original sampling rate may be only slightly above the Nyquist rate.

As an alternative to filters based on classical interpolation formulas, we discussed optimum low-pass and bandstop FIR filters that were designed by linear programming. The bandstop filters have frequency responses that are very similar to the classical interpolators, but are always superior. The bandstop designs appear to be most important for cases when the original sampling rate is several times the Nyquist rate, while the low-pass designs are appropriate when the original sampling rate is close to the Nyquist rate.

ACKNOWLEDGMENT

The authors would like to acknowledge the helpful comments and criticisms provided by Dr. O. Herrmann and Dr. J. F. Kaiser.

REFERENCES

- [1] A. E. Rosenberg, R. W. Schafer, and L. R. Rabiner, "Effects of smoothing and quantizing the parameters of formant-coded voiced speech," *J. Acoust. Soc. Amer.*, vol. 50, pp. 1532-1538, Dec. 1971.
- [2] R. W. Schafer and L. R. Rabiner, "Design and simulation of a speech analysis-synthesis system based on short-time Fourier analysis," *IEEE Trans. Audio Electroacoust.*, vol. AU-21, pp. 165-174, June 1973.
- [3] S. L. Freeney, R. B. Kieburz, K. V. Mina, and S. K. Tewksbury, "Design of digital filters for an all digital frequency division multiplex-time division multiplex translator," *IEEE Trans. Circuit Theory*, vol. CT-18, pp. 702-711, Nov. 1971.
- [4] D. J. Goodman, "The application of delta modulation to analog-to-PCM encoding," *Bell Syst. Tech. J.*, vol. 48, pp. 321-344, Feb. 1969.
- [5] H. Freeman, *Discrete-Time Systems*. New York: Wiley, 1965.
- [6] A. J. Gibbs, "On the frequency-domain responses of causal digital filters," Ph.D. dissertation, University of Wisconsin, Madison, 1969.
- [7] L. R. Rabiner, "Techniques for designing finite-duration impulse-response digital filters," *IEEE Trans. Commun. Technol.*, vol. COM-19, pp. 188-195, Apr. 1971.
- [8] O. Herrmann, "On the design of nonrecursive digital filters with linear phase," *Electron. Lett.*, vol. 6, pp. 328-329, 1970.
- [9] E. Hofstetter, A. V. Oppenheim, and J. Siegel, "A new technique for the design of nonrecursive digital filters," in *Proc. 5th Ann. Princeton Conf. on Information Sciences and Systems*, pp. 64-72, 1971.
- [10] T. W. Parks and J. H. McClellan, "Chebyshev approximation for nonrecursive digital filters with linear phase," *IEEE Trans. Circuit Theory*, vol. CT-19, pp. 189-194, Mar. 1972.
- [11] L. R. Rabiner, "The design of finite impulse response digital filters using linear programming techniques," *Bell Syst. Tech. J.*, vol. 51, pp. 1177-1198, July-Aug. 1972.
- [12] L. R. Rabiner and R. W. Schafer, "Recursive and nonrecursive realizations of digital filters designed by frequency sampling techniques," *IEEE Trans. Audio Electroacoust.*, vol. AU-19, pp. 200-207, Sept. 1971.
- [13] R. W. Hamming, *Numerical Methods for Scientists and Engineers*. New York: McGraw-Hill, 1962.
- [14] F. B. Hildebrand, *Introduction to Numerical Analysis*. New York: McGraw-Hill, 1956.
- [15] *Handbook of Mathematical Functions*, M. Abramowitz and I. A. Stegun, Eds. (Nat. Bur. Stand. Appl. Math. Ser.-55.) Washington, D. C.: U. S. Gov. Printing Office, 1964.