ISSN (Online) 2394-6849

International Journal of Engineering Research in Electronics and Communication Engineering (IJERECE)
Vol 4, Issue 1, January 2017

Designing of Interpolation Using Matlab

[1] Dr.P.C. Srikanth [2] Seetharamegowda.S
[1][2] Department of Digital Electronics & Communication Systems, Malnad College of engineering, Hassan

Abstract: **- Digital signal interpolation systems can be implemented in a variety of ways. The most basic interpolation system for integer upsampling cascades an expander unit with an interpolation low- pass filter. More complex implementations can cascade multiple expander and low-pass filter pairs. There is also flexibility in the design of interpolation filters.**

I. INTRODUCTION

The process of digital signal interpolation is fundamental to signal processing. It is used in manycontexts, most typically for conversion between sampling rates. This thesis explores efficient designsof digital interpolation systems for integer upsamplefactors.Interpolation of a signal by an integer upsample factor can be accomplished by processingthe signal, x[n], with the cascade of an expander and low-pass filter, as shown in Figure 1-1. If the input signal x[n] has sampling frequency f, this results in the upsampled and interpolated output signal y[n] at the increased sampling frequency Lf. More complex interpolation systems can be designedas the cascade of multiple expanders and low-pass filters. A system containing two expanders and two low-pass filters is shown in Figure 1-2.

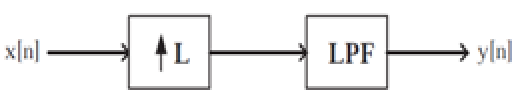


Fig 1-1: Interpolation system consisting of an expander and low pass filter

Fig1-2: Interpolation system consisting of two expander and two low pass filter

If the parameters of the cascaded interpolator in Figure 1-2 are chosen correctly, namely L1L2 = L and with appropriate choices of LPF1 and LPF2, then this system will perform equivalent interpolation to the system in Figure 1-1. In this case, assuming input sampling frequency f, the interpolated output signals y[n] and z[n] both have sampling frequencies Lf, and more specifically y[n] = z[n]. Thus, these two systems are distinct designs accomplishing the same interpolation, and can be compared in terms of computational efficiency.

This thesis studies the tradeoffs in the design of such interpolation systems for integer upsample factors. The metric used for comparison between system designs is computational cost, measured in multiplies per output sample. The following factors in system design are examined for their effect on computational cost:

- Finite impulse response (FIR) and infinite impulse response (IIR) low-pass filter designs.
- Linear-phase and minimum-phase FIR filter designs.
- Cascades of multiple expanders and low-pass filters.
- Distributions of the upsampling factor L over multiple expanders.

The processes of sampling rate reduction (often called decimation) and sampling rate increase (or interpolation). Sampling rate increase and sampling rate reduction are basically interpolation processes and can be efficiently implemented using finite impulse response (FIR) digital filters.

The processes of sampling rate reduction (often called decimation) and sampling rate increase (often called interpolation)

Sampling rate increase and sampling rate reduction are basically interpolation processes and can be efficiently implemented using finite impulse response (FIR) digital filters. The IFIR (Interpolated FIR) approach results in a Two-stage decimator/interpolator. For the multistage approach, the number of stages can be either automatically optimized or manually controlled. But multirate/multistage design introduces the most delay as compare with IFIR Design.

Increasing the sampling rate of a discrete-time signal x[n] by an integer factor I (upsampling) requires the

ISSN (Online) 2394-6849

International Journal of Engineering Research in Electronics and Communication Engineering (IJERECE)
Vol 4, Issue 1, January 2017

insertion of (I −1) samples between consecutive samples of x[n]. Thus, upsampling is an information preserving operation.

$$x_1[n] \triangleq x_c(nT_I) = x_c\left({}^{nT}/_{I}\right)$$

WhereTI $= T/I$, from the samples of the discrete-time signal

$$x[n] = xc(nT).$$

It can be implemented in matlab functions

Y=upsample (X,I)

Where 'x' is signal
I is interpolation factor
Which expands the input by a factor of I and then shifts the obtained sequence by inserting

kzeros, k = 0, 1, . . . , I −1, in the beginning.

Sampling rate increase by an integer factor

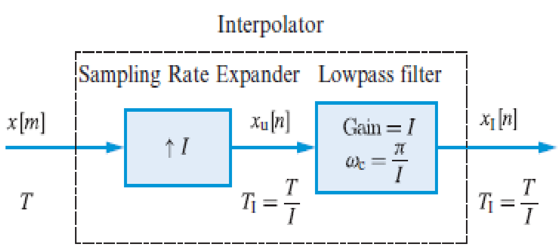


Fig :Discrete-time system for sampling rate increase by an integer factor I using ideal band limited interpolation

Which corresponds to an LTI system described by the convolution summation

$$x_I[n]=\sum_{k=-\infty}^{\infty} x_u[k]g_r[n-k]. \qquad 1$$

For increasing the sampling rate by an integer factor *I* can be implemented by a SRE followed by a filter with impulse response *gr*[*n*]. If we use an ideal low pass filter with gain *I* and cutoff frequency $\omega c = \pi/I$(see Figure b) we obtain the ideal interpolator described by (1).The sequences *g*r[*n*] used in (12.33) and (12.44) are identical; however, the interpretation is different. In the former case, *g*r[*n*] is the kernel or characteristic sequence of a linear time-varying interpolation system; in the latter case, *g*r[*n*] is the impulse response of a low pass filter.

II. FREQUENCY-DOMAIN INTERPRETATIONS

To understand the interpolation process in the frequency-domain, we start with the up sampler, whose operation is determined by since $X_u(e^{\pm j\pi})$=X($e^{\pm j\pi I}$),we conclude that I periods of x(e^{jw}) .The extra $(I-1)$copies of the compressed spectrum introduced by up sampling are called *images*. In thissense, we say that the up sampler creates an imaging effect. The interpolation filter *G*bl(e^{jw}) Removes all these images and scales the spectrum by *I*to compensate for the 1/*I* reductionin signal bandwidth. If the interpolation filter has a cutoff frequency larger than *π/I* or we use a non-ideal filter, energy from the images remains in the interpolated signal. This typeof distortion is known as "*post-aliasing*" in the computer graphics literature

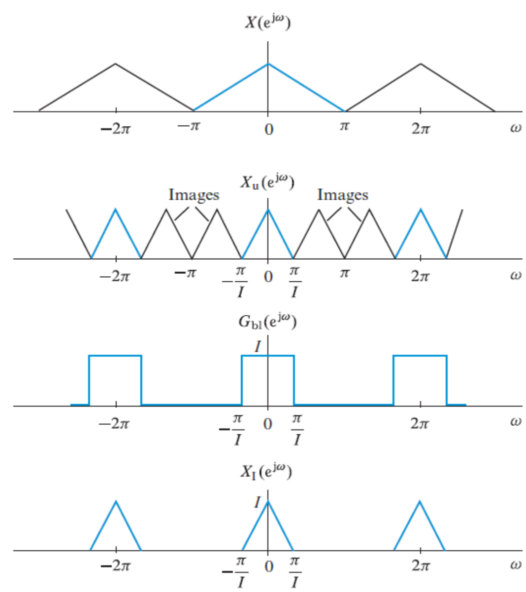


Fig: Frequency-domain interpretation of interpolation for I=3

Interpolation is used to change the sampling rate of a signal without changing its spectral content. In other words, increasing the sampling rate of a given signal (upsampling) increases the spectral separation between the images of the original spectrum. However, this process doesn't add any new information to the existing signal even though the sampling rate has increased, yielding more sample points for processing. After zero insertion, get the simplest form of signal interpolation. The upsampling process is shown in Figure 4.3.

ISSN (Online) 2394-6849

International Journal of Engineering Research in Electronics and Communication Engineering (IJERECE)
Vol 4, Issue 1, January 2017

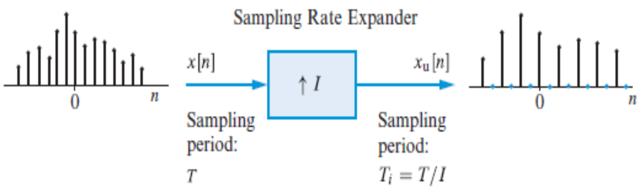


Fig: Representation of a sampling rate expander

Interpolation via zero insertion implies that zeros are inserted at the new rate between the original sample signals according to the relation (6)

$$y(n) = \begin{cases} x(\frac{n}{L}) & for\ n = 0, \pm L, \pm 2L, ... \\ 0 & otherwise \end{cases} \quad (6)$$

This up-sampling process is illustrated for an up-sampling factor of 4 as shown in figure 5.The relationship between Y(z) and X(z) is easily derived. Let

$$Y(z) = \sum_{n=-\infty}^{\infty} y(n) z^{-n} = \sum_{\substack{n=-\infty \\ k=nL}}^{\infty} y(k) z^{-k}$$

$$= \sum_{\substack{n=-\infty \\ k=nL}}^{\infty} x(\frac{k}{L}) z^{-k} \quad (7)$$

Define l = k/L, the relation in (7) becomes:

$$Y(z) = \sum_{\substack{n=-\infty \\ k=nL}}^{\infty} x(l) z^{-lL} = X(z^{L}) \quad (8)$$

In the frequency domain, this implies that the DTFT of Y (z) is

$$Y(e^{j\omega}) = X(e^{j\omega L}) \quad (9)$$

The relationship in (9) can be depicted for the case when L=4.The spectrum of the up-sampled signal is made up of multiple compressed copies of the original signal spectrum. These compressed copies, known as images, are placed at 2π/L intervals from DC. The original signal to the frequency range π ≤ ω ≤ π implies that the upsampled signal is now limited to the range π/L ≤ ω ≤ π/L. So, the up-sampling process compresses the signal in the frequency domain by a factor of L with respect to the new sampling rate which is L times higher than the original sampling rate.

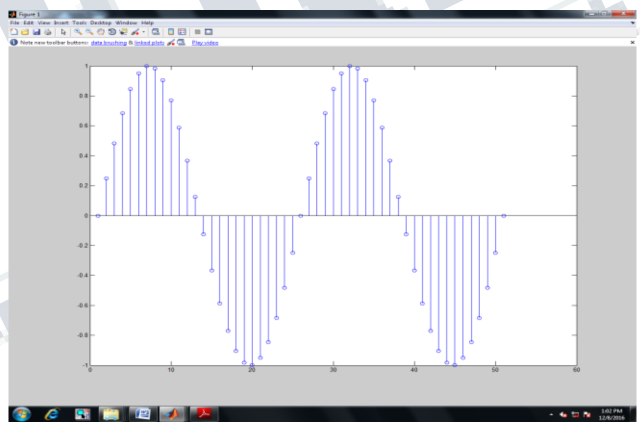
In order to suppress the images, the up-sampler of Figure 4.3 is typically followed by a low pass filter. The cutoff frequency of this filter is typically set to π/L. The resulting signal is an interpolated version of the original signal. The zero valued samples inserted between the original samples are now replaced with nonzero samples due to the convolution operation between the upsampled signal and the impulse response of the low pass filter h(n).

III. APPLICATIONS

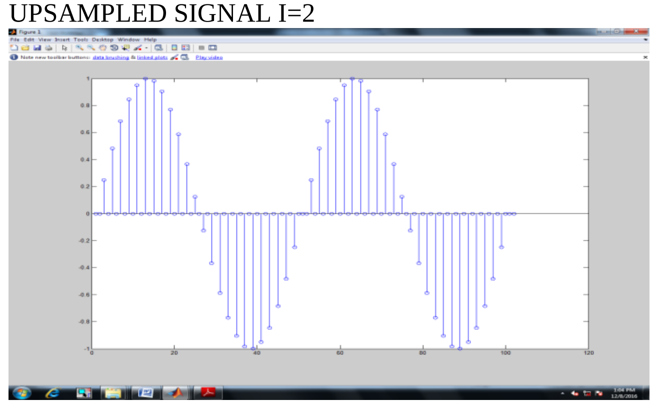
It has been widely used in Audio, SpeechProcessing, Radar Systems and Communication Systems.

SIMULATED RESULTS

ORIGNAL SIGNAL



UPSAMPLED SIGNAL I=2



IV. CONCLUSION

In this paper i have discussed the process of interpolation as a problem in digital filtering. Most of our discussion has involved frequency domain representation of the interpolation process and design criteria for digital interpolation filters.

REFERENCES

1. L.Milic, "Multirate Filtering for Digital Signal Processing: MATLAB Applications," IGI Global, 2009.

2. J. N. Fliege, "Multirate Digital Signal Processing: Multirate Systems, Filter Banks, and Wavelets," John Wiley & Sons, 1994.

3. T. Hentschel, M. Henker, and G. Fettweis, "The Digital Front-end of software radio terminals," IEEE Journal on Personal Communications, pp. 40-46, Aug. 1999.

4. N. Younis, "Power Efficient Clock/Data Distribution Techniques for Polyphase Decomposition Comb Filter in SDR Receiver," .