Design of Fractional Delay Filters Using Convex Optimization

William Putnam (putnam@ccrma.stanford.edu*)*
Julius Smith (jos@ccrma.stanford.edu *)*

Department of Electrical Engineering and
Center For Research In Music and Acoustics (CCRMA)
Stanford University
Stanford, CA 94305-8180

ABSTRACT

Fractional sample delay (FD) filters are useful and necessary in many applications, such as the accurate steering of acoustic arrays [1], [2], delay lines for physical models of musical instruments [3] [4], and time delay estimation[5]. This paper addresses the design of finite impulse response (FIR) FD filters. The problem will be posed as a convex optimization problem in which the maximum modulus of the complex error will be minized. Several design examples will be presented, along with an empirical formula for the filter order required to meet a given worst case group delay error specification.

1. Introduction

This paper presents an optimization technique for designing FIR FD filters. FD filters are those which exhibit near unity magnitude response and a flat group delay which is not necessarily an integer multiple of the sampling interval. Essentially, FIR FD filters are discrete-time interpolators which approximate the signal in between sample points as a linear combination of sample values on either side of the desired signal value.

Designing FD filters involves determining the coefficients of an FIR filter such that its response best approximates the complex valued frequency response of the desired FD. Standard optimal filter design methods such as the Remez exchange [6] cannot be used to design FD filters since an FD filter is neither symmetric nor anti-symmetric. In this paper, the design of FD filters will be approached as a convex optimization problem. In general, a convex optimization problem is one in which a convex function is minimized subject to any number of convex constraints. Specifically, the problem is posed as a second order cone problem (SOCP), which is a general form that includes many other problems such as linear programs, quadratic programs, and quadratically constrained quadratic programs. For details on the SOCP problem, see [7].

The paper is organized as follows. Section 2 presents previous work in the areas of FIR filter design for fractional delay filters. Section 3 presents the formulation of the optimization problem specifically for FD filters. Section 4 presents design examples which depict the nature of the group delay error for even and odd length filters. Furthermore, an empirical formula is provided which relates the length of a filter to the resultant worst case group delay. Problem extensions along with a description of future work are given in section 5.

2. Previous Work

Fractional delay filtering has received quite a bit of attention in the literature. A comprehensive overview of FD filtering as well as an extensive bibliography is found in [8]. Lagrange interpolation has been suggested for fractional delay systems and is covered in [9] and [10] among others. In [11] the equivalence between Lagrange interpolation, maximally flat filters, and windowed sinc functions is established. [12] provides a comparison between several techniques including the minimization of least-square error and Lagrange interpolation.

3. Problem Formulation

The group delay of a filter is defined as:

$$\Delta_g \stackrel{\Delta}{=} -\frac{d}{d\omega}\angle H(\omega) \qquad (1)$$

An ideal fractional delay filter exhibits a constant group delay with a magnitude spectrum constant, and equal to unity. For a filter to have a constant group delay of Δ, the desired response would be:

$$H_d(\omega) \stackrel{\Delta}{=} e^{-j\Delta\omega} \qquad (2)$$

The problem of fractional delay filter design is that of choosing the coefficients of a filter such that its response best approximates the desired response $H_d(\omega)$ in some sense. In this paper, the ℓ_∞ norm on $\mathbf{C}^N$ is chosen. It is defined by $\|x\|_\infty \stackrel{\Delta}{=} \max_{i=1}^N |x_i|$. This means that the worst case modulus of the complex error over frequency will be minimized.

Using the above definition, the optimization problem becomes:

$$\text{minimize} \quad \max_\omega |H(\omega) - H_d(\omega)| \qquad (3)$$

where $H(\omega)$ is the frequency response of a filter with coefficients h_n, and is given by:

$$H(\omega) \stackrel{\Delta}{=} \sum_{n=0}^{N} h_n e^{-j\omega n}$$

Typically, in the formulation of the optimization problem, one would allow for a transition or "don't care" region in which the response of the filter is unspecified. This allows the optimization to use these degrees of freedom to best approximate the desired response in other regions. Furthermore, it was pointed out in [11] that the ℓ_∞ norm on $\mathbf{C}^N$ will not converge to zero for some FD filters since there will always be unavoidable finite error at the Nyquist frequency. This indicates a bandlimited ℓ_∞ norm must be used.

Defining the optimization problem over a bandlimited set Ω, results in:

$$\text{minimize} \quad \max_{\omega\in\Omega} |H(\omega) - H_d(\omega)| \tag{4}$$

Where $\Omega \triangleq \{\omega | 0 < \omega < \omega_{\text{max}}\}$, and $H_d(\omega)$ is as defined in equation 2. Typically, $\omega_{\text{max}} \approx .9\pi$.

This problem has a finite number of design variables, but an infinite number of constraints, and hence is known as a semi-infinite programming problem [13]. In practice, it is usually sufficient to perform the optimization over a finite discrete set of ω_i. Typically, the number of constraints is taken to be approximately 4N, where N is the number of design variables. Furthermore, it can be shown that the sampled-frequency solution converges to the optimal solution of the semi-infinite problem as the discretization interval becomes small [14] [7]. Additionally, an exchange algorithm can be employed to keep the number of constraints small, while still maintaining convergence to the solution of the continuous problem [15].

If $a_i^T \in \mathbf{C}^N$ is defined as:

$$a_i^T \triangleq [\ 1 \quad e^{-j\omega_i} \quad e^{-2j\omega_i} \quad \ldots \quad e^{-(N-1)j\omega_i}\ ] \tag{5}$$

then the frequency response at ω_i is then given by:

$$H(\omega_i) = \sum_{n=0}^{N} h_n e^{-j\omega_i n} = a_i^T h \tag{6}$$

where $h^T \triangleq [\ h_0 \quad h_1 \quad \ldots \quad h_{N-1}\ ]$.

At frequency ω_i, the approximation error is given by $|a_i^T h - H_d(\omega_i)|$. The problem can now be stated as:

$$\text{minimize} \quad \max_{\omega_i} |a_i^T h - H_d(\omega_i)| \tag{7}$$

By introducing a new variable t, the problem can be formulated as

$$\begin{array}{ll} \text{minimize} & t \\ \text{subject to} & |a_i^T h - H_d(\omega_i)| < t, i \in 1, \ldots, M \end{array}$$

where $a_i \in \mathbf{C}^N$, and $H_d(\omega_i) \in \mathbf{C}$. The problem can now be stated in terms of real parameters as follows:

$$\begin{aligned} |a_i^T h - H_d(\omega_i)|^2 &= |[\Re a_i^T + j\Im a_i^T]h - \\ &\quad [\Re H_d(\omega_i) + j\Im H_d(\omega_i)]|^2 \\ &= |[\Re a_i^T h - \Re H_d(\omega_i)] + \\ &\quad j[\Im a_i^T h - \Im H_d(\omega_i)]|^2 \\ &= \left\| \begin{bmatrix} \Re a_i^T h - \Re H_d(\omega_i) \\ \Im a_i^T h - \Im H_d(\omega_i) \end{bmatrix} \right\|_2^2 \\ &= \left\| \begin{bmatrix} \Re a_i^T \\ \Im a_i^T \end{bmatrix} h - \begin{bmatrix} \Re H_d(\omega_i) \\ \Im H_d(\omega_i) \end{bmatrix} \right\|_2^2 \end{aligned} \tag{8}$$

Hence the original problem can be written as:

$$\begin{array}{ll} \text{minimize} & t \\ \text{subject to} & \|A_i h - b_i\| < t, i \in 1, \ldots, M \end{array}$$

where

$$A_i \triangleq \begin{bmatrix} \Re a_i^T \\ \Im a_i^T \end{bmatrix}, b_i \triangleq \begin{bmatrix} \Re H_d^T \\ \Im H_d^T \end{bmatrix} \tag{9}$$

By extending the design variable h to include the slack variable t, the optimization problem can be stated in terms of a new variable x, where $x^T \triangleq [h^T t]$. Using this the problem becomes

$$\begin{array}{ll} \text{minimize} & f^T x \\ \text{subject to} & \|\tilde{A}_i x - b_i\| < c_i^T x, i \in 1, \ldots M \end{array}$$

In the above, $\tilde{A}_i \triangleq \begin{bmatrix} A_i & \begin{matrix} 0 \\ 0 \end{matrix} \end{bmatrix}$, and $f^T \triangleq [\ 0 \quad \ldots \quad 0 \quad 1\ ]$, hence $A_i x = \tilde{A}_i x$ and $t = f^T x$. This is the final form, and it can be seen that its nature is that of minimizing a linear functional subject to quadratic constraints. This is known as a quadratically constrained quadratic program (QCQP). Furthermore, it can be expressed as a second-order cone problem (SOCP)[7], which can be solved very efficiently. This framework also allows for the inclusion of additional problem constraints such as magnitude constraints in stopbands.

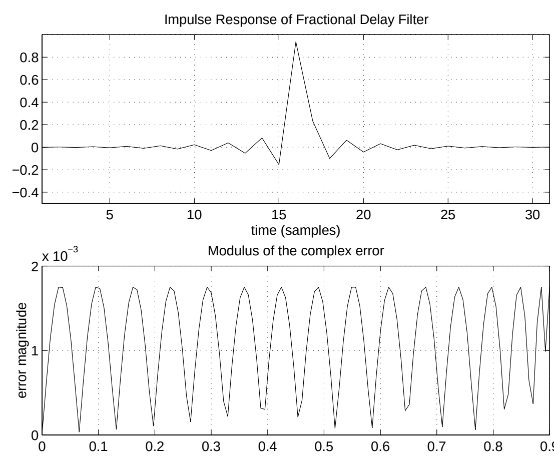

Figure 1: Impulse response and equiripple error modulus of a length 31 filter with a fractional delay of .2 samples.

4. Examples

All the optimization problems presented herein were solved using the code presented in [7] which is available via ftp [1] [16]. Code for the fractional delay filter design, as well as all the examples in this paper is also available via the web. [2]

[1] `http://www-isl.Stanford.EDU/~boyd/SOCP.html`
[2] `http://www-ccrma.stanford.edu/~putnam`

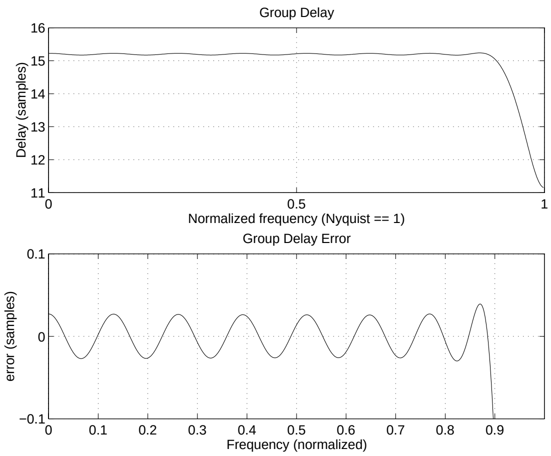

Figure 2: Plot showing the resultant group delay and group delay error for the filter in Figure 1

As mentioned previously, the complex Chebychev optimization results in the filter which best approximates the complex valued frequency response of the desired constant group delay filter in a minmax sense. Figure 1 shows the impulse response and the the magnitude of the complex error in the frequency response for a typical FD design problem. In this example, the desired fractional component of the delay of the filter is .2 samples. Specifically, the overall specified group delay of the filter was 15.2 samples. The results clearly show the equiripple nature of the solution. Figure 2 shows the group delay, and group delay error for the same filter.

4.1. Error vs. Delay

Figures 3-4 show the group delay error versus frequency for fractional delays ranging from 0 to 1 sample. Data are presented for both even and odd length filters since they exhibit different behaviour in

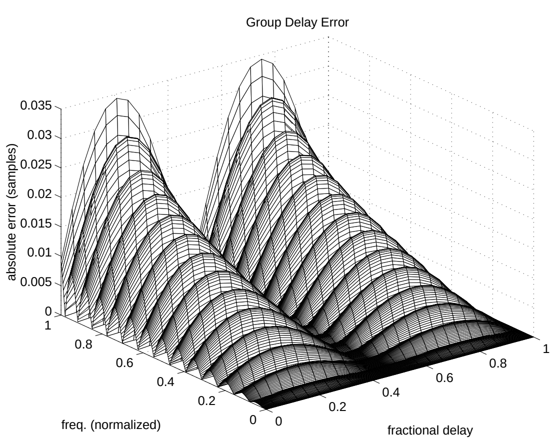

Figure 3: Group delay error vs. frequency as a function of fractional delay for an even length filter.

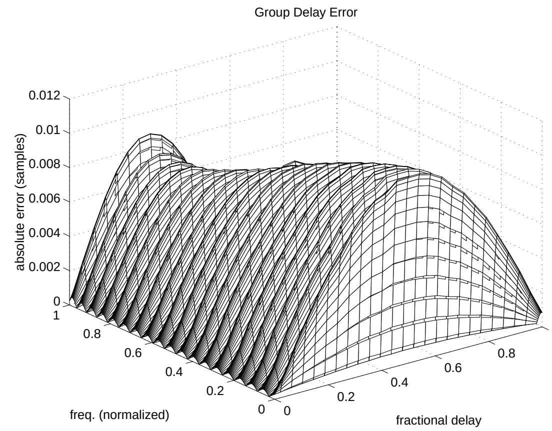

Figure 4: Group delay error vs. frequency as a function of fractional delay for an odd length filter.

terms of group delay. As might be expected, an even-length filter has no problem achieving a delay of .5 samples. As seen in Figure 3, the worst case error occurs at .25 and .75 samples. Likewise, an odd length filter achieves its worst case group delay error at .5 samples.

4.2. Error / Length tradeoff Curves

Figure 5 depicts the tradeoff curve for worst case group delay error as function of the filter length for both even and odd length filters. This error is presented both on a linear and a logarithmic axis. Worst case error was determined by designing a large number of filters over a range of fractional delays and then taking the maximum error for

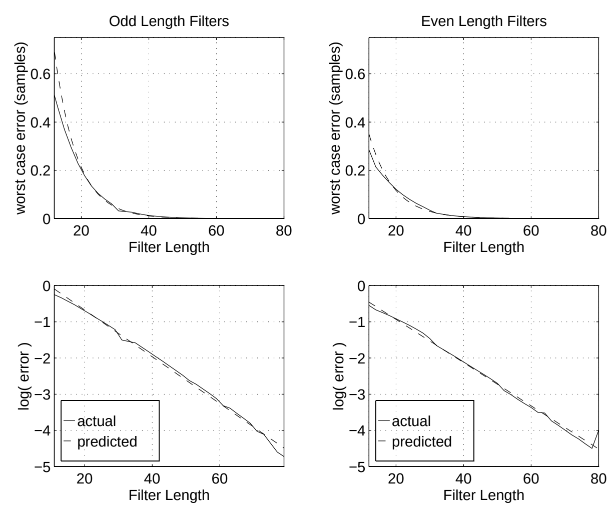

Figure 5: Worst case group delay error as function of filter length for both odd and even length filters. The error is plotted on both a linear and logarithmic axis. The dashed line is the error predicted by equation 10, using the appropriate parameters from Table 1.

parameter:	a	b
even	-0.0597	0.2598
odd	-0.0646	0.6192

Table 1: Parameters determined by a least squares fit of the data in Figure 5. These parameters are used in equation 11 to determine the minimum length filter required to meet a groupd delay error specification.

all filters over a dense grid of frequencies.

It is striking that the group delay error varies approximately linearly on a logarithmic scale. This allows for the a simple closed form expression predicting the necessary filter order for a given desired group delay or magnitude error. As can be seen in the bottom plots in Figure 5, $\log_{10} E_D$ varies approximately linearly with the length of the filter, and hence can be expressed as:

$$\log_{10} E_D \approx aN + b \quad (10)$$

Using a least squares fit, the parameters a and b were determined for both the even and odd length cases, and are given in Table 1.

Equation 10 can be rearranged into the following form which expresses the minimum length filter necessary to meet a group delay error specification:

$$N \geq \frac{\log_{10} E_D - a}{b} \quad (11)$$

5. Problem Extensions and Future Work

This design technique can be extended to include many convex constraints on the design parameters, or on the frequency response of a filter. This would be useful in order to provide constraints on the maximum magnitude of the response of a filter in its stopbands. This is important in the design of filters for asynchronous sample rate conversion, where one must low-pass filter to avoid aliasing, as well as provide fractional delay to reconstruct the signal at non rational multiples of the sampling rate [17].

6. Conclusions

This paper presented the design of fractional delay filters whose frequency response is optimal in a complex Chebchev sense. Design examples were provided which demonstrate the usefullness of this method. An empirical closed form approximate expression was presented which expresses the filter length required in order to meet a desired worst case group delay error.

7. Acknowledgments

The authors would like to thank Stephen Boyd for help in the area of convex optimization and Miguel Lobo for support and help in the use of his SOCP software.

References

1. Roger G. Pridham and Ronald A. Mucci, “Digital Interpolation Beamforming for Low-Pass and BandPass Signals,” *Journal of the Acoustical Society of America*, Vol. 67, No. 6, 1979, pp. 904-919.
2. William Putnam and Julius Smith, “Optimal Chebychev interpolation Filters for Accurate Beam Steering,” *Third Joint Meeting of the Acoustical Society of America and the Acoustical Society of Japan*, Honolulu, Hawaii, December 1996.
3. Vesa Välimäki, Matti Karajaleinen, and T. Kuisma “Modeling of Woodwind Bores With Finger Holes,” *Proc. of The International Computer Music Conference(ICMC ’93)*, Tokyo, Japan, pp. 32-29, 1993.
4. David A. Jaffe and Julius O. Smith, “Extensions of the Karplus-Strong Plucked String Algorithm,” *Computer Music Journal* Vol. 7, No. 2, 1983, pp. 56-69.
5. Julius O. Smith and Benjamin Friedlander, “Estimation of Multipath Delay,” *Proceedings of the International Conference on Acoustics Speech and Signal Processing*, San Diego, pp. 15.9.1–15.9.4, 1984.
6. E.Y. Remez *General Computational Methods of Chebyshev Approximation for Nonrecursive Digital Filters with Linear Phases*, U.S. Atomic Energy Commision, Oak Ridge, Tenn., 1957.
7. M. S. Lobo and L. Vandenberghe and S. Boyd and H. Lebret “Second-order cone programming: interior-point methods and engineering applications,” *In Preparation*.
8. T.I. Laakso, Vesa Valimaki and Matti Karjalainen “Splitting the unit delay FIR/all pass filters design,” *IEEE Signal Processing Magazine*, Vol. 13, No. 1, 1996, pp. 30-60.
9. Shailey Minocha, S.C. Dutta Roy, Balbir Kumar “A Note On The FIR Approximation of a Fractional Sample Delay,” *International Journal of Circuit Theory and Applications*, Vol. 21, 1993, pp. 265-274.
10. C.C. Ko and Y.C. Lim, “Approximation of a Variable-Length Delay Line By Using Tapped Delay Line Processing,” *Signal Processing*, Vol. 14, No. 4, June 1988, pp. 363-369.
11. Peter J. Kootsookos and Robert C. Williamson, “FIR Approximation of Fractional Sample Delay Systems,” *IEEE Transactions on Circuits and Systems-II: Analog and Digital Signal Processing* Vol. 43, No. 3, 1996, pp. 269-271.
12. G. D. Cain, N.P. Murphy and A. Tarczynski, “Evaluation of Several Variable FIR Fractional Sample Delay Filters,” *Proceedings of the International Conference on Acoustics Speech and Signal Processing*, Tokyo, Japan, pp. 32-29, 1994.
13. D. G. Luenberger *Optimization By Vector Space Methods*, John Wiley & Sons, New York, N. Y., 1969.
14. E. W. Cheney *Introduction to Approximation Theory*, Chelsea Publishing Company, New York, N. Y., 1982.
15. William Putnam and Stephen Boyd “FIR filter design using convex optimization,” *In Preparation*.
16. M. S. Lobo, L. Vandenberghe and S. Boyd “SOCP: Software for Second-Order Cone Programming,” Information Systems Laboratory, Stanford University.
17. Julius O. Smith and P. Gossett, “A flexible sampling-rate conversion method,” *Proceedings of the International Conference on Acoustics Speech and Signal Processing*, San Diego, March 1984, vol. 2, pp. 19.4.1-19.4.2.