Proc. *Midwest Symposium on Circuits and Systems* (*MWSCAS '99*), Las Cruces, NM, August 1999

# Computationally Efficient Multirate Passband Equalization for Bandpass Digital/Analog Conversion

Dan P. Scholnik and Jeffrey O. Coleman
*scholnik@nrl.navy.mil* *jeffc@alum.mit.edu*
Radar Division, Naval Research Laboratory
Washington, DC

*Abstract*— **Many systems require a digital filter for decimation/interpolation and equalization of analog filtering. Under suitable conditions, iterative design of a low-rate, nonlinear-phase FIR filter and a cascaded high-rate, linear-phase FIR filter results in a computationally efficient filter structure.**

## I. Introduction

Common architectures found in communication and radar systems and in D/A and A/D conversion involve analog filtering in cascade with digital filtering for decimation or interpolation. An example of increasing importance is the now-commonplace IF-sampling receiver [1, 2] in which a bandpass-filtered analog signal is sampled and decimation filtered. Its upconversion/transmitter counterpart [3–5] follows interpolation with analog bandpass filtering. Typically the analog-filter effects in these systems must be corrected, which requires the digital filter have nonlinear phase. However, to prevent aliasing in the rate-changing operation the digital filter must also provide substantial stopband suppression, generally leading to a relatively long filter response.

In this paper we consider the addition of a second, nonlinear-phase filter at the lower of the two rates to reduce the total computational burden by allowing the high-rate filter to have linear phase. This leads to a division of labor between the two filters—the low-rate filter provides all of the phase compensation and some of the passband amplitude compensation. The high-rate filter provides the stopband suppression and the remaining amplitude compensation in the passband. The length of the low-rate filter will be determined primarily by the passband compensation required by the analog filtering, while the length of the high-rate filter will be controlled by interpolation/decimation requirements.

This approach is complicated, however, by the need to concurrently design two digital filters. While the design of a single FIR filter for such an application can usually be formulated [2, 3, 5] as a convex program, for which there are fast and efficient global solvers [6–9], the two-filter cascade generally leads to a nonconvex optimization problem. Our approach here is to alternately optimize each FIR filter based on the last (optimized) response of the other, using an engine [9] for solving second-order cone programs (which are convex). Convergence can be guaranteed by a proper choice of constraints in the optimization.

Scholnik's involvement in this work was partially supported by National Science Foundation grant MIP-9896034 to the University of Maryland Baltimore County, where he is a graduate student and Coleman is an Adjunct Assistant Professor of CS and Electrical Engineering. The remainder was supported by the base funding of NRL through the Office of Naval Research.

## II. Designing an FIR Filter Cascade

There are three common approaches to designing an FIR filter cascade using convex optimization tools. The simplest is to design each filter individually, with (likely excessive) constraints that ensure that the overall error will be acceptable [10]. Reference [11], for example, estimates that such an approach typically leads to a $15\%$ overdesign for a class of frequency-response masking filters. The second is to sequentially design the filters, incorporating the previous designs in each step [11, 12]. This approach is nearly optimal when one of the filters has little freedom or when only one filter has a significant effect on any given spectral region. The third approach, taken here, is to iterate the second method until the filter responses converge. This approach has been used previously for the design of linear-phase IFIR filters using linear programming [13], the Parks-McClellan algorithm [14], and least-squares and eigenfilter methods [15].

No claim is made here that at convergence an iteratively designed filter cascade is jointly optimal in the combined coefficients, as the nonconvex error surface may well have suboptimal local minima. Under appropriate constraints, however, it is straightforward to show that the design must converge and perform no worse (better in almost every case) than a single-iteration design. Consider the optimization problem

$$\begin{array}{ll} \text{minimize} & f(\mathbf{g},\mathbf{h}) \\ \text{subject to} & c_i(\mathbf{g},\mathbf{h}) \le 0, \quad i = 1,\ldots,K \end{array}$$

where $\mathbf{g}$ and $\mathbf{h}$ are vectors of optimization variables representing the coefficients of the two FIR filters, and both the objective $f()$ and constraints $c_i()$ are convex functions of $\mathbf{g}$ and $\mathbf{h}$ individually (but not jointly). Fixing either $\mathbf{g}$ or $\mathbf{h}$ results in a convex optimization problem in the other set of variables. With the value of $\mathbf{h}$ fixed at $\mathbf{h_0}$, the above optimization can be performed to obtain $\mathbf{g_1}$, with the corresponding objective value $f(\mathbf{g_1},\mathbf{h_0})$. If now $\mathbf{g}$ is left fixed at $\mathbf{g_1}$ and $\mathbf{h}$ is optimized to obtain $\mathbf{h_1}$, it is clear that $f(\mathbf{g_1},\mathbf{h_1}) \le \mathbf{f}(\mathbf{g_1},\mathbf{h_0})$. It follows that the sequence of successive values of the objective as the design is iterated is monotonically nonincreasing. If $f()$,

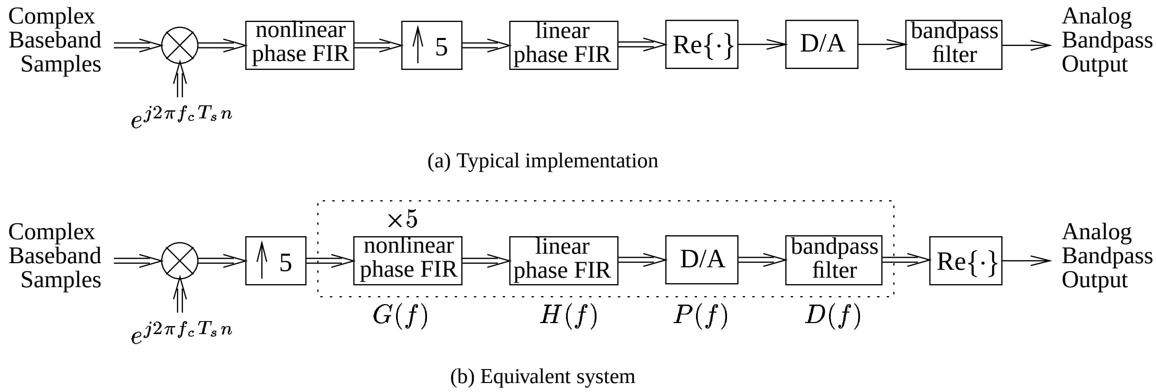


Fig. 1. Example bandpass synthesis system.

which generally represents some error function, is bounded below (as it will be in all practical designs) then the sequence must converge. Thus at each iteration performance cannot decrease (and will usually increase), and eventually a stable state will be reached. In most cases tried by the authors convergence was achieved in five to fifteen iterations.

In practice, the constraints imposed at each of the two iteration stages need not be identical, but just sufficient to avoid degrading the performance of the other stage. For example, extra constraints on the individual filters are often needed in transition bands to keep responses reasonable but have minimal effect on the objective, and these will be different for each filter. An example will help illustrate.

## III. Example

Consider Fig. 1(a), which depicts a bandpass-synthesis system[1] in which rate $f_s$ input samples feed a nonlinear-phase FIR equalizer followed by a linear-phase interpolation-by-five filter, D/A converter and an analog filter. Here the digital filtering must correct for the D/A hold response and the analog-filter amplitude and group-delay distortion while supporting the interpolation. Figure 1(b) shows an equivalent system in which a noble identity is used to exchange the low-rate filter and interpolator and the $\mathrm{Re}\{\cdot\}$ operation is moved to the end. The result is to isolate all the filter responses in one equivalent block (dotted box), so that the overall response to be designed is the cascade $G(f)H(f)P(f)D(f)$, where equalizer response $G(f)$ has a period of $f_s$, interpolator response $H(f)$ has a period of $5f_s$, hold response $P(f) = \mathrm{sinc}(f/5f_s)$, and $D(f)$ is the response of the analog bandpass filter.

The input signal has a bandwidth of $0.8f_s$, and the desired output is a bandpass signal with a carrier frequency of $1.25f_s$. The cascade of equalizer, interpolator, hold response, and analog filter is therefore required to form an analytic (one-sided) bandpass filter, with desired specifications of 40 dB stopband suppression and $1\%$ rms passband error. The passband is the interval $[0.85f_s, 1.65f_s]$ and to prevent aliasing the stopband is defined by $(-\infty, 0.65f_s] \cup [1.85f_s, \infty)$. The remaining two intervals form the transition region. The analog bandpass filter is generated from a sixth-order lowpass Chebychev prototype with 1 dB passband ripple, chosen to provide relatively sharp transitions. The passband edges of the analog filter are chosen somewhat wider than the overall passband width as the most severe group-delay distortion occurs at the band edges. It was determined empirically that an equalizer of length six is sufficient to correct for the group delay of the analog filter, and a length-21 interpolator provides the remaining required amplitude shaping.

The design process proceeds as follows. The nonlinear phase filter is initialized by minimizing the rms passband error of the cascade with the analog filter:

$$\text{minimize} \quad \sum_{f \in \mathcal{F}_{\mathrm{pb}}} \left| G(f)D(f) - e^{-j2\pi f \tau} \right|^2$$

Parameter $\tau$ represents the desired group delay of the filter cascade. The performance of an FIR equalizer is typically very sensitive to the choice of $\tau$, empirically determined for this example to be $2.8T_s$. The $L_2$ integral is here approximated by a Riemann sum over an appropriate grid of frequencies. Then, with the low-rate filter response $G(f)$ fixed, the optimization

$$\begin{aligned} &\text{minimize} && \sum_{f \in \mathcal{F}_{\mathrm{pb}}} \left| G(f)H(f)P(f)D(f) - e^{-j2\pi f \tau} \right|^2 \\ &\text{subject to} && \left| G(f)H(f)P(f)D(f) \right|^2 \leq 10^{-4}, \quad f \in \mathcal{F}_{\mathrm{sb}} \end{aligned}$$

is performed over the possible interpolator responses $H(f)$. At this point the optimization begins iterating between designing $G(f)$ with $H(f)$ fixed according to

$$\text{minimize} \quad \sum_{f \in \mathcal{F}_{\mathrm{pb}}} \left| D(f)G(f)H(f)P(f) - e^{-j2\pi f \tau} \right|^2$$

and designing $H(f)$ as before. Note that the stopband constraint is not enforced when designing the low-rate filter. This filter has little effect on the stopband, and what little degradation may occur is easily compensated in the interpolator. Not

[1] This example is derived from an example in [5].

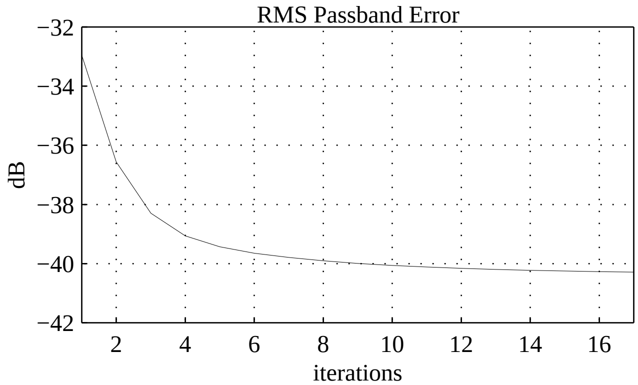


Fig. 2. Passband error vs. iteration.

shown are the auxiliary constraints on the transition bands of the filters used to avoid extreme responses. Figure 2 shows the optimized value of the rms passband error after each two-part iteration. After eight iterations the responses have nearly settled, with a 7 dB improvement over the first-pass design.

Figures 3 and 4 show the responses of the two iteratively optimized FIR filters along with the analog-filter and cascade responses. In the passband the low-rate filter provides the group-delay compensation and part of the amplitude correction. The high-rate filter provides the stopband suppression and the rest of the needed amplitude shaping in the passband. The nonlinear-phase filter has complex inputs, outputs, and coefficients, and thus requires four real multiplies per coefficient per input sample, or 24 total. The interpolator has linear phase and real outputs, so it requires just one real multiply per coefficient per input sample, or 21 total. The two-filter cascade thus requires 45 real multiplies per input sample. For comparison, a single nonlinear-phase FIR interpolation filter of length 32 was designed to the same specifications and required 64 multiplies per input sample. This single filter requires 38% more multiplies than the factored cascade.

## IV. Concluding Remarks

Why the relatively modest gains compared to, say, the near-factor of $N$ savings often associated with IFIR filtering? There are two reasons. First, factoring the single complex interpolator into two filters results in isolating the low-rate filter from the $\mathrm{Re}\{\cdot\}$ operation, doubling the required computation for the equalizer. Thus even with only six taps the equalizer of the example is more computationally demanding than the high-rate filter. Second, the proposed approach is fundamentally distinct from IFIR filtering, where the typically linear-phase low-rate shaping filter is responsible for the sharp transitions and part of the stopband suppression. Here the low-rate equalizer is responsible only for passband shaping, since typically its frequency-response period is only slightly greater than the bandwidth of the input data. The relatively sharp transitions of the example are produced by the interpolator and analog filter, and in fact an IFIR filter might be used here to save compu-

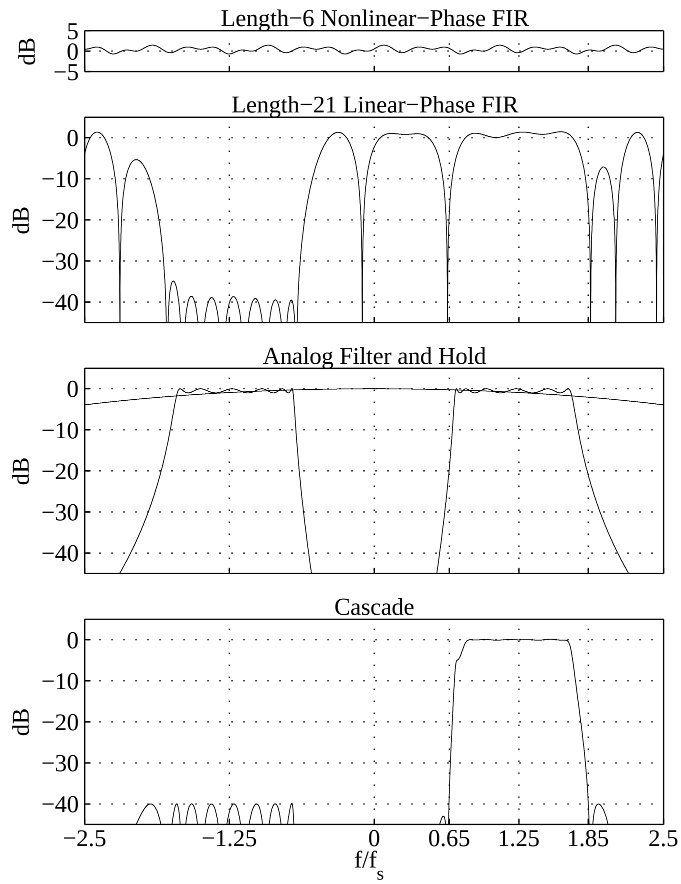


Fig. 3. Magnitude responses of the example design.

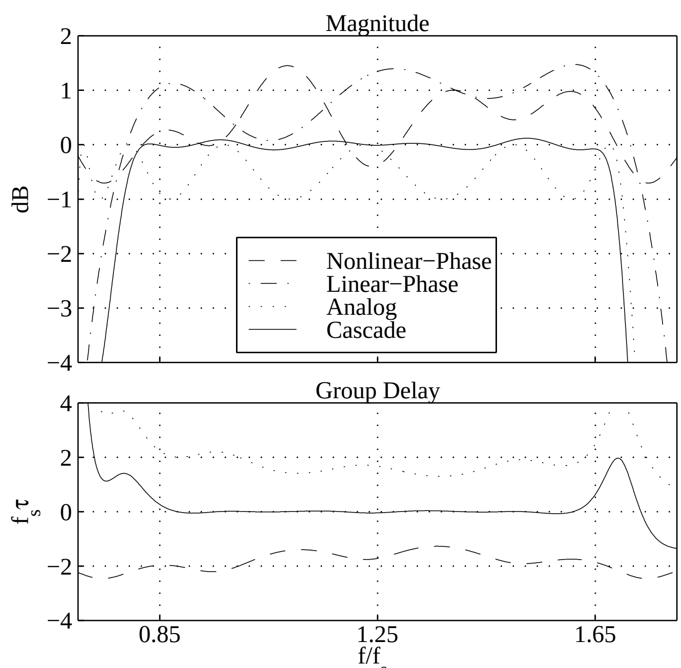


Fig. 4. Passband details.

tation and/or lower the analog filter order. The configuration used was chosen to isolate the interaction between the linear and nonlinear phase sections.

This iterative approach to filter design is not limited to minimum MSE objectives or equiripple stopbands, but is applicable to almost any filter response error measure. How the nature of the objective and the type and number of constraints affects either convergence rate or iteration gain is an open question. Although not attempted, this approach could also be extended to three or more filters.

## References


[1] R. L. Mitchell, "Creating complex signal samples from a band-limited real signal," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 25, no. 3, pp. 425–427, May 1989.

[2] D. P. Scholnik and J. O. Coleman, "Periodically nonuniform bandpass sampling as a tapped-delay-line filtering problem," in *Proc. 1999 Int'l Conf. Acoustics, Speech, and Signal Processing (ICASSP '99)*, Mar. 1999.

[3] H. Samueli and B. C. Wong, "A VLSI architecture for a high-speed all-digital quadrature modulator and demodulator for digital radio applications," *IEEE J. Select. Areas in Commun.*, vol. 8, no. 8, pp. 1512–1519, Oct. 1990.

[4] D. P. Scholnik and J. O. Coleman, "Simple, exact models of sample-interleaving demodulators/modulators for quadrature bandpass sampling/reconstruction," in *Proc. Conf. Information Science and Systems*, Baltimore, MD, Mar. 1997.

[5] D. P. Scholnik and J. O. Coleman, "Nonuniformly offset polyphase synthesis of a bandpass signal from complex-envelope samples," in *Proc. 1999 Int'l Symp. Circuits and Systems (ISCAS '99)*, May 1999.

[6] M. S. Lobo, L. Vandenberghe, S. Boyd, and H. Lebret, "Applications of second-order cone programming," *Linear Algebra and its Applications*, vol. 284, pp. 193–228, Nov. 1998.

[7] R. Vanderbei, "LOQO: An interior point code for quadradic programming," Technical Report SOR 94-15, Princeton University, 1994, revised 11/30/98, to appear in *Optimization Methods and Software*.

[8] R.J. Vanderbei and D.F. Shanno, "An interior-point algorithm for non-convex nonlinear programming," Tech. Rep. SOR-97-21, Statistics and Operations Research, Princeton University, 1997, To appear in *Computational Optimization and Applications.*

[9] J. F. Sturm, "Sedumi," MATLAB toolbox, May 1998, http://www.unimaas.nl/~ sturm/software/sedumi.html.

[10] G. Rajan, Y. Nuevo, and S. K. Mitra, "On the design of sharp cutoff wide-band FIR filters with reduced arithmetic complexity," *IEEE Trans. Circuits and Systems*, vol. 35, no. 11, pp. 1447–1454, Nov. 1988.

[11] R. Yang, B. Liu, and Y. C. Lim, "A new structure of sharp transition FIR filters using frequency-response masking," *IEEE Trans. Circuits and Systems*, vol. 35, no. 8, pp. 955–966, Aug. 1988.

[12] Y. C. Lim, "Frequency-response masking approach for the synthesis of sharp linear phase digital filters," *IEEE Trans. Circuits and Systems*, vol. 33, no. 4, pp. 357–364, Apr. 1986.

[13] J. O. Coleman, *The Use of the FF Design Language for the Linear-Programming Design of Finite-Impulse-Response Digital Filters for Digital-Communication and Other Applications,* Ph.D. thesis, University of Washington, Seattle, WA, Dec. 1991.

[14] T. Saramäki, Y. Nuevo, and S. K. Mitra, "Design of computationally efficient interpolated FIR filters," *IEEE Trans. Circuits and Systems*, vol. 35, no. 1, pp. 70–88, Jan. 1988.

[15] Y.-P. Lin and P. P. Vaidyanathan, "An iterative approach to the design of IFIR matched filters," in *Proc. 1997 Int'l Symp. Circuits and Systems (ISCAS '97)*, June 1997.