A NEW ALGORITHM FOR EFFICIENT LOW DELAY FILTER BANK DESIGN

Gerald Schuller and Mark J. T. Smith

Georgia Institute of Technology
School of Electrical and Computer Engineering
Atlanta, GA 30332-0250
email mjts@eedsp.gatech.edu

ABSTRACT

Historically, exact reconstruction FIR filter banks have had system delays of $L-1$, where L is the length of the analysis and synthesis filters. Recently it was shown that the system delay could be made less than $L-1$, which is attractive in applications like speech coding where excessive delays are annoying. In this paper, a formulation and new design algorithm are introduced for two-band low-delay filter banks. The formulation is related to that of two-band lattice filter banks and provides a broad range of design flexibility within a compact framework. Both exact reconstruction and specified system delay are guaranteed by the structure of the framework.

1. INTRODUCTION

Two-band filter banks are employed very frequently in subband coding applications. Much progress has been made in addressing the traditional design issues, such as reducing or removing distortions in the reconstruction, and placing constraints on the frequency and impulse response characteristics of the individual filters [1], [2]. Much less attention, however, has been given to system delay. System delay is particularly important in speech coding systems where long delays in transmission and reception between two parties in conversation can be disruptive and annoying.

Recently, it was shown that system delay could be controlled in the design of filter banks, suggesting that low delay systems could have a positive effect on subband speech coding [5]. In fact, in [4], it is shown that the system delay in speech coders can be reduced without degrading subjective performance. The design of low delay systems, however, is not very mature at this point. In the original work by Nayebi, et al. [5], obtaining filters with good magnitude response characteristics and good reconstruction properties involved repeated optimizations from many different starting points. In later work by Nguyen [6], the same problem was encountered. His reported results were slightly better (i.e. a few dB improvement in the stopband rejection) than those reported by Nayebi, et al.

In this paper, we introduce a new formulation and design method for low delay filter banks, specifically for two-band systems. The new method allows for control over the passband, stopband, and time domain characteristics, like other methods but has the attractive property that exact reconstruction is guaranteed structurally. So aliasing, phase, and magnitude distortions are always zero in the absence of quantization. This condition is not guaranteed in the original work by Nayebi, et al.

This work builds on the formulation reported in [7, 8], which is based on a cascaded matrix form representation. In the following sections we present the two-band matrix structure and introduce a specialized optimization algorithm.

2. THE MATRIX FRAMEWORK

The design formulation is based on a cascade of matrices, some composed solely of filter coefficients, others composed of delay elements. These matrices completely characterize the filter bank and lead to an efficient implementation, similar in structure to the lattice-form implementation [1], [3].

To begin the discussion, consider a two-channel filter bank with the input signal $x(n)$, the downsampled filter outputs $y_0(n)$, $y_1(n)$, analysis filters $h_0(n)$, $h_1(n)$, and synthesis filters $g_0(n)$, $g_1(n)$. Following the conventional lattice formulation, the input signal is viewed in terms of a 2-D vector in time

$$\mathbf{x}(n) = [x(2n), x(2n+1)]$$

[0]This work was supported in part by JSEP under contract DAAD-04-93-G-0027

with the equivalently representation

$$\mathbf{X}(z) = [X_0(z), X_1(z)]$$

in the z-domain. Similarly, $y_0(n)$ and $y_1(n)$ can be expressed as the vector,

$$\mathbf{Y}(z) = [Y_0(z), Y_1(z)]$$

resulting in

$$\mathbf{Y}(z) = \mathbf{X}(z) \cdot \mathbf{P_a}(z)$$

where $\mathbf{P_a}(z)$ is the polyphase matrix [1] given by

$$\mathbf{P_a} = \begin{bmatrix} P_{0,0}(z) & P_{0,1}(z) \\ P_{1,0}(z) & P_{1,1}(z) \end{bmatrix}. \quad (1)$$

where $P_{n,k}(z) = \sum_{m=0}^{L-1} h_k(n+2m) z^{-(L-1-m)}$. To guarantee both exact reconstruction and the realization of FIR synthesis filters of the same length as those in the analysis, we impose a mild constraint on the form of the polyphase matrix. In particular, we restrict $\mathbf{P_a}$ to be the cascade of four matrix types:

Transform Matrices—These are denoted by $\mathbf{T}$ and are analogous to the two-point DFT matrices in the classical two-band polyphase implementation. They have the form

$$\mathbf{T} = \begin{bmatrix} a_0 & a_2 \\ a_1 & a_3 \end{bmatrix}. \quad (2)$$

In this work, $a_0 = 0.9239$, $a_1 = 0.3827$, $a_2 = 0.3827$, $a_3 = -0.9239$ (which is a 2 point DCT type IV).

Coefficient Matrices—These are conventional matrices with real- or complex-number elements. Within this group, two matrix forms are suggested for use: $\mathbf{F}$ and $\mathbf{C}_i$ where

$$\mathbf{F} = \begin{bmatrix} d_0 & d_2 \\ d_1 & d_3 \end{bmatrix} \quad (3)$$

$$\mathbf{C}_i = \begin{bmatrix} c_0^i & 1 \\ 1 & c_1^i \end{bmatrix}. \quad (4)$$

Standard Delay Matrices—These matrices accompany the coefficient matrices above. When coefficient and delay matrices are cascaded in pairs, they produce the z-domain matrix elements in the polyphase matrix $\mathbf{P_a}$. The standard delay matrices are denoted by $\mathbf{D}$ and have the form

$$\mathbf{D} = \begin{bmatrix} z^{-1} & 0 \\ 0 & 1 \end{bmatrix}. \quad (5)$$

Zero-Delay Matrices—These are coefficient matrices that can be included in the cascade that have the special property that they do not introduce any delay to the overall analysis/synthesis system. We shall show why this is so in the next subsection. There are two types of zero-delay matrices that we have found useful, $\mathbf{E}_i$ and $\mathbf{G}_i$. These matrices and their inverses are given by:

$$\mathbf{E}_0 = \begin{bmatrix} 0 & e_2^0 \\ e_3^0 & e_1^0 z^{-1} \end{bmatrix}, \quad \mathbf{E}_0^{-1} = \begin{bmatrix} \frac{-e_1^0}{e_2^0 e_3^0} z^{-1} & \frac{1}{e_3^0} \\ \frac{1}{e_2^0} & 0 \end{bmatrix} \quad (6)$$

For $i > 0$ the elements on the anti-diagonal can be 1, leading to the more efficient matrices

$$\mathbf{E}_i = \begin{bmatrix} 0 & 1 \\ 1 & e_1^i z^{-1} \end{bmatrix}, \quad \mathbf{E}_i^{-1} = \begin{bmatrix} -e_1^i z^{-1} & 1 \\ 1 & 0 \end{bmatrix}. \quad (7)$$

Similarly, the form for the $\mathbf{G}$ matrices is

$$\mathbf{G}_i = \begin{bmatrix} g_0^i z^{-1} & 1 \\ 1 & 0 \end{bmatrix}, \quad \mathbf{G}_i^{-1} = \begin{bmatrix} 0 & 1 \\ 1 & -g_0^i z^{-1} \end{bmatrix}. \quad (8)$$

In the lattice-form implementation for two-band filter banks [1], each stage in the lattice had the same form. Here, we have expanded the variety of stages permissible by extending the set to include all the matrices listed above. This additional variety allows us to control the overall system delay as we discuss in the next section.

2.1. Reconstruction and System Delay

Given that the analysis section is composed entirely of lattice sections, each of which may be represented by a simple matrix, the synthesis filter bank can be obtained by inverting each matrix individually. The key to controlling the overall delay is the recognition that there is a system delay associated with each matrix used in the analysis section. To elaborate further, consider each of the four matrix types with respect to the system delay they contribute. Note that the transform matrices, $\mathbf{T}$, and the coefficient matrices, $\mathbf{F}$ and $\mathbf{C}_i$, contribute no delay, since their elements are real or complex numbers. Consequently the elements of their inverses are also real or complex numbers. Only matrices involving z^{-1} terms have the potential for contributing system delay. Such is the case for the standard delay matrix, $\mathbf{D}$. Its inverse results in a z term, which represents an advance. To preserve causality, it is necessary to remove this term by delaying the system by z^{-1}. Thus the causal inverse for the standard delay matrix is

$$z^{-1} \cdot \mathbf{D}^{-1} = \begin{bmatrix} 1 & 0 \\ 0 & z^{-1} \end{bmatrix}. \quad (9)$$

It is therefore apparent that we add a delay due to the inversion. In contrast, the zero-delay matrix has the property that no delay term is contributed to the overall system. Examine equations (6), (7), and (8). Note that the cascade of the matrix and inverse results in an identity matrix with no delays, although the both matrices contain a delay element z^{-1}. This special property allows a causal implementation with no added system delay.

The system delay can now be set by mixing the zero-delay and standard delay matrices in the formation of the filter banks. This leads to analysis filter banks of the general form

$$\mathbf{P_a} = \left(\prod_{i=1}^{m} \mathbf{C}_i \cdot \mathbf{D}^2\right) \cdot \mathbf{F} \cdot \mathbf{D} \cdot \left(\prod_{i=1}^{n} \mathbf{G}_i\right) \mathbf{T}. \qquad (10)$$

The synthesis filters are obtained by taking the inverse term by term, resulting in the synthesis filters $\mathbf{P_s} =$

$$\mathbf{T}^{-1} \left(\prod_{i=0}^{n-1} \mathbf{G}_{n-i}^{-1}\right) \cdot \mathbf{D}^{-1} \cdot z^{-1} \cdot \mathbf{F}^{-1} \cdot \left(\prod_{i=0}^{m-1} \mathbf{D}^{-2} \cdot z^{-2} \cdot \mathbf{C}_{m-i}^{-1}\right). \qquad (11)$$

Given this decomposition, the length $K = 4m + 2n + 4$ and the delay is $4m + 3$. The filter length and delay is set by choosing n and m. Experimental results have shown that for the extreme case of minimum delay, better frequency response characteristics are possible when the $\mathbf{E}_i$ zero-delay matrices are used in place of the $\mathbf{G}_i$ matrices. For this case

$$\mathbf{P_a} = \left(\prod_{i=0}^{m-1} \mathbf{E}_i\right) \mathbf{T}, \quad \mathbf{P_s} = \mathbf{T}^{-1} \prod_{i=0}^{m-1} \mathbf{E}_{m-1-i}^{-1}. \qquad (12)$$

The resulting length of the analysis and synthesis filters is $K = 2m+1$ and the system delay is 1. The delay that is left is the transform block delay, which is 1 sample and the minimum possible delay.

This formulation has addressed explicitly the system delay and perfect reconstruction property by virtue of the construction. To address the control of the frequency and/or impulse response properties, we propose the optimization procedure discussed in the next section.

3. OPTIMIZATION

We define $\mathbf{x}$ to be a row vector of the s unknown filter matrix entries (all of which are real for real valued filters), and $\mathbf{H}(\mathbf{x})$ to be the weighted lowpass frequency responses for the analysis and synthesis filter bank at ℓ frequency samples. For convenience, $\mathbf{H}(\mathbf{x})$ is a row vector consisting of the analysis and synthesis responses in tandem. Thus the vector length is 2ℓ. Moreover, we define $\mathbf{d}$ to be the weighted ideal frequency response for analysis and synthesis. As such it too is a row vector with 2ℓ elements, each of which we denote as d_i. The lowpass analysis and synthesis filters determine the full system completely. This is because the transform $\mathbf{T}$ imposes a relationship between the lowpass and highpass filters. By examining the relationship $\mathbf{P_s}(z) = \mathbf{P_a}^{-1}(z) \cdot z^{-d}$ (where z^{-d} is a delay of d segments and $\det(\mathbf{P_a}) = z^{-d}$) we see that the relationship between the analysis filters H and synthesis filters G is $H_1(z) = G_0(-z)$ and $G_1(z) = -H_0(-z)$. The error function f for the squared distance is then

$$f(\mathbf{x}) = \sum_{i=1}^{2l} |H_i(\mathbf{x}) - d_i|^2$$

To optimize the magnitude of the frequency response, we use the error function,

$$f(\mathbf{x}) = \sum_{i=1}^{2\ell} (|H_i(\mathbf{x})| - d_i)^2 = \sum_{i=1}^{2\ell} |H_i(\mathbf{x}) - \frac{H_i(\mathbf{x})}{|H_i(\mathbf{x})|} \cdot d_i|^2 =$$

$$\sum_{i=1}^{2\ell} |H_i(\mathbf{x}) - d_i'|^2$$

There are several reasonable methods for minimization. The method of conjugate directions (see [9]) was found to have a robust and relatively fast convergence behavior for this function. To minimize $f(\mathbf{x})$, its second derivative or Hessian matrix is used. The minimization is done as an iterative process with separate (one dimensional) line minimizations, where the direction of each line minimization is determined by the eigenvectors of the Hessian. The line minimization is performed by using Newtons method. To illustrate the idea, let $\mathbf{x}_0$ be the starting point in the iteration. The Newton step is then

$$\mathbf{x}_1 = \mathbf{x}_0 - \Delta\mathbf{x}\,, \quad \Delta\mathbf{x} = \frac{\partial f/\partial \mathbf{v}_i|_{\mathbf{x}_0}}{\partial^2 f/\partial \mathbf{v}_i^2|_{\mathbf{x}_0}} \cdot \mathbf{v}_i$$

The derivatives can be computed as

$$\frac{\partial f}{\partial \mathbf{v}_i} = 2\mathrm{Re}\{(\mathbf{H} - \mathbf{d}) \overline{\frac{\partial \mathbf{H}}{\partial \mathbf{v}_i}}^T\}$$

$$\frac{\partial^2 f}{\partial \mathbf{v}_i^2} \approx 2\mathrm{Re}\{\frac{\partial \mathbf{H}}{\partial \mathbf{v}_i} \cdot \overline{\frac{\partial \mathbf{H}}{\partial \mathbf{v}_i}}^T\}$$

where the overbar means complex conjugate. Here it can be seen that this step approaches a minimum, because the second derivative is always greater than zero.

If $f(\mathbf{x}_1) > f(\mathbf{x}_0)$ then the magnitude of $\Delta\mathbf{x}$ is reduced, and if that brings no improvement, $\mathbf{x}$ is left unchanged for this $\mathbf{v}_i$.

The directions $\mathbf{v}_i$ are chosen such that a small change in one direction does not change the location of the minimum (i.e. the first derivative) of the other directions. If $\mathbf{B}$ is the Hessian matrix of f, then this means $\mathbf{v}_i \mathbf{B} \mathbf{v}_j^T = 0$. This is true for the s eigenvectors of $\mathbf{B}$. $\mathbf{B}$ can be approximated by the first derivative of $\mathbf{H}$. Define $\mathbf{A} = \nabla \mathbf{H}^T$ with its elements as $a_{ij} = \partial H_j / \partial x_i$. Then $\mathbf{B}$ is approximated by neglecting higher order derivatives of $\mathbf{H}$ as

$$\mathbf{B} \approx 2\mathrm{Re}\{\mathbf{A}\overline{\mathbf{A}}^T\}.$$

A new $\mathbf{B}$ is computed after the full previous set of eigenvectors $\mathbf{v}_i$ of $\mathbf{B}$ is used to update $\mathbf{x}$.

Note that only the s derivatives of $\mathbf{H}$ are used. No explicit computations of second derivatives are needed and no stepsize parameter α is required. Experiments have shown that this algorithm is less sensitive to the starting point than others we have tried. However, as with any iterative algorithm, choosing a starting is an issue. The strategy employed in this work that was found to be effective is to start with a small filter length using random initial element values. Iterations are performed to obtain good filters for this length. The length is then increased by appending zero-valued matrix elements to achieve the desired length. This effectively allows you to integrate the initialization process into the design. Sometimes it can be beneficial to try a second random starting point. Convergences usually occurs in less than a few hundred iterations and the overall design process is relatively rapid.

As an illustration, Figure 1 shows the spectral magnitude response of a low-delay 32 tap filter bank and a 16 tap conventional filter bank, both with system delays of 15 samples. The low-delay filter bank is able to achieve a narrower transition band and about 5 dB improvement in stopband rejection.

The upper bounds on performance for this design approach are not yet clear. Perhaps further improvement in design performance is possible. However, what seems interesting from our perspective is the notion that delay can be imposed structurally in the lattice design framework and therefore exact reconstruction can be guaranteed.

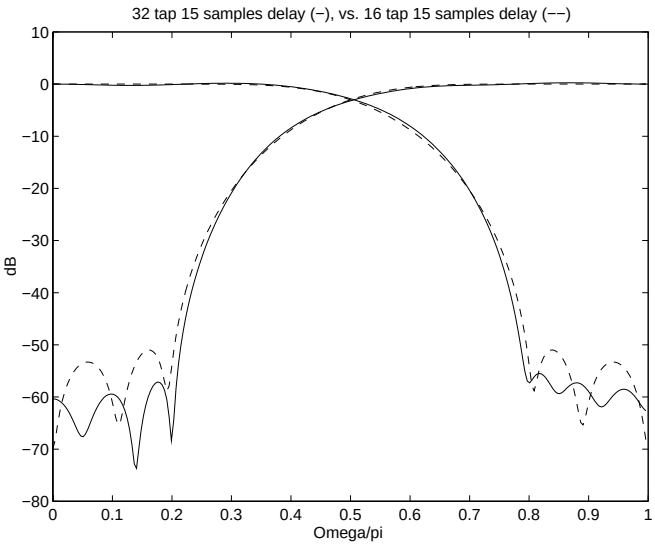

Figure 1: Analysis filter bank frequency responses for 32 tap and 16 tap filters, both with 15 sample system delay.

4. REFERENCES

[1] P.P. Vaidyanathan, *Multirate Systems and Filter Banks*, Prentice Hall, 1993.

[2] A. Akansu and R. Haddad, *Multiresolution Signal Decomposition*, Academic Press, 1992.

[3] H. Malvar, *Signal Processing with Lapped Transforms*, Artech House, 1991.

[4] K. Nayebi, T. Barnwell, M. Smith, "Low Delay Coding of Speech and Audio Using Nonuniform Band Filter Banks," IEEE Workshop on Speech Coding for Telecom. Sept. 1991.

[5] K. Nayebi, T. Barnwell, M. Smith, "Design of Low Delay FIR Analysis-Synthesis Filter Bank Systems," Proc. Conf. on Info. Sci. and Sys., Mar. 1991.

[6] T. E. Tuncer and T. Nguyen, "General Analysis of Two-Band QMF Banks," to appear in IEEE Trans. on Signal Processing.

[7] G. Schuller and M. J. T. Smith, "A General Formulation for Modulated Perfect Recon. Filter Banks with Variable System Delay," NJIT 94 Sym. on Appl. of Subbands and Wavelets, Mar. 1994.

[8] G. Schuller and M. J. T. Smith, "Efficient Low Delay Filter Banks", DSP Workshop, Oct. 1994.

[9] W.H.Press at al., "Numerical Recipes", Cambridge University Press,1992