PARAUNITARY FILTER BANK DESIGN USING DERIVATIVE CONSTRAINTS

Peter G. Vouras and Trac D. Tran

Department of Electrical and Computer Engineering
The Johns Hopkins University
Baltimore, MD. 21218, USA

ABSTRACT

In this paper two new algorithms are presented for designing finite impulse response (FIR) paraunitary (PU) filter banks. Each algorithm minimizes the mean square error between the desired response and the FIR PU approximation subject to constraints on either the derivative of the complex frequency response or the power response of individual channel filters. The derivative constraints are useful for shaping the response of a channel filter at particular frequencies of interest. An example illustrating the utility of derivative constraints is presented whereby a FIR PU approximation is derived for an ideal principal component filter bank (PCFB).

***Index Terms*—** paraunitary filter bank design, principal component filter bank, derivative constraints, Givens factorization

1. INTRODUCTION[1]

Consider a maximally decimated filter bank with M channels. Each analysis filter $H_k(z)$ with impulse response $h_k(n)$ can be expressed in terms of its polyphase components as

$$H_k(z) = \sum_{l=0}^{M-1} z^{-l} E_{kl}\left(z^M\right), \quad k = 0, \ldots, M-1$$

where

$$e_{kl}(n) = h_k(l + Mn), \quad 0 \le l \le M-1$$

and

$$E_{kl}(z) = \sum_{n=-\infty}^{\infty} e_{kl}(n) z^{-n}.$$

Similarly, each synthesis filter $F_k(z)$ can be expressed in terms of its polyphase components $R_{lk}(z)$ as

$$F_k(z) = \sum_{l=0}^{M-1} z^{-(M-1-l)} R_{lk}\left(z^M\right), \quad k = 0, \ldots, M-1.$$

[1] This research is supported by the National Science Foundation under CAREER Grant CCR-0093262. Author emails: peter.vouras@nrl.navy.mil, trac@jhu.edu.

Now define the M x M polyphase component matrices $\boldsymbol{E}(z)$ and $\boldsymbol{R}(z)$ as

$$\boldsymbol{E}(z) = \left[E_{kl}(z)\right], \quad \boldsymbol{R}(z) = \left[R_{lk}(z)\right].$$

The polyphase matrix evaluated for $z = e^{j\omega}$ is denoted by $\boldsymbol{E}(\omega)$ or $\boldsymbol{R}(\omega)$. Based on the polyphase representation of the analysis and synthesis filters, maximally decimated filter banks may be implemented most efficiently [1].

2. FACTORIZATION OF PARAUNITARY POLYPHASE MATRIX

A paraunitary filter bank satisfies the condition that

$$\boldsymbol{E}(\omega)\boldsymbol{R}(\omega) = I, \quad \boldsymbol{E}(\omega) = \boldsymbol{R}^H(\omega) \quad \forall \omega.$$

For real filter banks, the polyphase matrix $\boldsymbol{R}(\omega)$ can be decomposed into the following factored form,

$$\boldsymbol{R}(\omega) = \boldsymbol{G}_L \Lambda(\omega) \cdots \boldsymbol{G}_1 \Lambda(\omega) \boldsymbol{QJ}$$

where L is the Smith-McMillan degree of $\boldsymbol{R}(\omega)$, $\boldsymbol{G}_k$ is the product of Givens rotation matrices, $\boldsymbol{Q}$ is an orthogonal matrix, and

$$\Lambda(\omega) = \begin{bmatrix} \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{0} & e^{-j\omega}\boldsymbol{I} \end{bmatrix}, \quad \boldsymbol{J} = \begin{bmatrix} \pm 1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \pm 1 \end{bmatrix}.$$

Each matrix $\boldsymbol{G}_m$ and $\boldsymbol{Q}$ is the product of ½M(M-1) Givens rotation matrices of the form,

$$\boldsymbol{S}_{ij}(\theta_k) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & \cos(\theta_k) & \cdots & \sin(\theta_k) & 0 \\ 0 & \vdots & 1 & \vdots & 0 \\ 0 & -\sin(\theta_k) & \cdots & \cos(\theta_k) & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

$$1 \le k \le \tfrac{1}{2}(L+1)M(M-1),$$

where $\cos(\theta_k)$ is placed in the ith row and ith column, $\sin(\theta_k)$ is in position (i, j), -$\sin(\theta_k)$ is in position (j, i), and $\cos(\theta_k)$ is in position (j, j). The order of the matrices $\boldsymbol{S}_{ij}$ in the product is important. In general, a M-by-M orthogonal matrix $\boldsymbol{G}_m$ can be decomposed into the following product sequence of Givens rotation matrices [2],

$$\boldsymbol{G}_m = \left\{\boldsymbol{S}_{M-2,M-1}\right\} \cdots \left\{\boldsymbol{S}_{1,M-1} \cdots \boldsymbol{S}_{12}\right\}\left\{\boldsymbol{S}_{0,M-1} \cdots \boldsymbol{S}_{01}\right\}.$$

For instance, if $M = 4$, the decomposition becomes,

$$G_m(\Theta)=\{S_{23}(\theta_1)\}\{S_{13}(\theta_2)S_{12}(\theta_3)\}\{S_{03}(\theta_4)S_{02}(\theta_5)S_{01}(\theta_6)\}$$
where $\Theta = [\theta_1, \ldots, \theta_6]^T$.

3. MINIMUM MEAN SQUARE ERROR (MMSE) FILTER BANK DESIGN

One approach to designing a FIR PU filter bank that closely approximates a desired filter bank is to minimize the weighted mean squared Frobenius norm error between the polyphase matrix of the desired filter bank, $\boldsymbol{D}(\omega)$, and the FIR PU synthesis polyphase matrix, $\boldsymbol{R}(\omega)$. The objective function to be minimized is

$$\eta = \frac{1}{2\pi}\int_0^{2\pi} W(\omega)\left\| \boldsymbol{D}(\omega) - \boldsymbol{R}(\omega)\right\|_F^2 d\omega$$

where $W(\omega)$ is a scalar nonnegative weight function. By embedding a Givens decomposition of the polyphase matrix $\boldsymbol{R}(\omega)$ into the objective function, the solution filter bank is guaranteed to be PU with $\boldsymbol{E}(\omega) = \boldsymbol{R}^H(\omega)$.

After substituting the decomposition for $\boldsymbol{R}(\omega)$ into η, there are $\frac{1}{2}(L+1)M(M-1)$ rotation angles θ_k that are free parameters. To restrict the value of each θ_k to lie between -2π and 2π, we define the following penalty function,

$$\varphi = \sum_{j=1}^{\frac{(L+1)M(M-1)}{2}} \left(\max\{0, \theta_j - 2\pi\}\right)^2 + \left(\max\{0, -2\pi - \theta_j\}\right)^2 .$$

We can now solve the unconstrained minimization problem,

$$\min \quad \xi = \eta + \alpha\varphi ,$$

where α is a positive scalar. One of the more popular techniques for solving unconstained minimization problems is the Broyden, Fletcher, Goldfarb, and Shanno (BFGS) algorithm, which falls under the category of a quasi-Newton method. The MATLAB function *fminunc* implements a version of the BFGS algorithm for unconstrained minimization. More details on designing FIR PU filter banks using unconstrained minimization can be found in [3].

4. DERIVATIVE CONSTRAINTS

It is often desirable to shape the frequency response of specific channel filters in an MMSE-optimal filter bank such that the gain is flat at particular frequencies of interest. One way to accomplish this task is to minimize the objective function η subject to zero constraints on either the derivatives of the complex frequency response or the derivatives of the real power response of individual channel filters.

In the first case, the minimization problem becomes

$$\min \quad \xi = \eta + \alpha\varphi$$

subject to

$$\mathrm{Re}\left\{\frac{\partial^k F_j(\omega)}{\partial \omega^k}\right\}\Bigg|_{\omega=\omega_i} = 0$$

and

$$\mathrm{Im}\left\{\frac{\partial^k F_j(\omega)}{\partial \omega^k}\right\}\Bigg|_{\omega=\omega_i} = 0$$

where k denotes the kth derivative with respect to ω, $F_j(\omega)$ is the complex frequency response of the jth synthesis filter, and ω_i is the ith frequency the constraint is imposed at.

In the second case the minimization problem becomes

$$\min \quad \xi = \eta + \alpha\varphi$$

subject to

$$\frac{\partial^k \left|F_j(\omega)\right|^2}{\partial \omega^k}\Bigg|_{\omega=\omega_i} = 0.$$

The primary difference between the two types of derivative constraints is that in the first case the filter's phase response will be constrained, whereas in the second case the filter's phase response can be arbitrary. This paper will examine both constraint types.

4.1. Phase Constrained Solution

The complex frequency response of the jth synthesis filter can be written using the polyphase matrix $\boldsymbol{R}(\omega)$ as

$$F_j(\omega) = \boldsymbol{e}^H(\omega)\boldsymbol{R}(\omega M)\boldsymbol{u}_j$$

where

$$\boldsymbol{e}(\omega) = \begin{bmatrix} e^{j(M-1)\omega} & e^{j(M-2)\omega} & \ldots & e^{j\omega} & 1 \end{bmatrix}^T$$

and $\boldsymbol{u}_j$ is a column vector of zeroes except for the jth component which is equal to one.

Next define the following recursive sequence of matrices,

$$\begin{aligned}
\boldsymbol{U}_L &= \boldsymbol{G}_L \\
\boldsymbol{U}_{L-1} &= \boldsymbol{G}_L \boldsymbol{\Lambda}(\omega M)\boldsymbol{G}_{L-1} = \boldsymbol{U}_L \boldsymbol{\Lambda}(\omega M)\boldsymbol{G}_{L-1} \\
\boldsymbol{U}_{L-2} &= \boldsymbol{G}_L \boldsymbol{\Lambda}(\omega M)\boldsymbol{G}_{L-1}\boldsymbol{\Lambda}(\omega M)\boldsymbol{G}_{L-2} \\
&= \boldsymbol{U}_{L-1}\boldsymbol{\Lambda}(\omega M)\boldsymbol{G}_{L-2} \\
&\vdots \\
\boldsymbol{U}_2 &= \boldsymbol{G}_L \boldsymbol{\Lambda}(\omega M)\boldsymbol{G}_{L-1}\boldsymbol{\Lambda}(\omega M)\boldsymbol{G}_{L-2}\cdots \boldsymbol{G}_2 \\
&= \boldsymbol{U}_3 \boldsymbol{\Lambda}(\omega M)\boldsymbol{G}_2 \\
\boldsymbol{U}_1 &= \boldsymbol{G}_L \boldsymbol{\Lambda}(\omega M)\boldsymbol{G}_{L-1}\boldsymbol{\Lambda}(\omega M)\boldsymbol{G}_{L-2}\cdots \boldsymbol{G}_1 \\
&= \boldsymbol{U}_2 \boldsymbol{\Lambda}(\omega M)\boldsymbol{G}_1
\end{aligned}$$

$$\begin{aligned}\boldsymbol{V}_L &= \boldsymbol{G}_{L-1}\boldsymbol{\Lambda}(\omega M)\boldsymbol{G}_{L-2}\cdots\boldsymbol{G}_1\boldsymbol{\Lambda}(\omega M)\boldsymbol{QJ}\\ &= \boldsymbol{G}_{L-1}\boldsymbol{\Lambda}(\omega M)\boldsymbol{V}_{L-1}\\ \boldsymbol{V}_{L-1} &= \boldsymbol{G}_{L-2}\boldsymbol{\Lambda}(\omega M)\boldsymbol{G}_{L-3}\cdots\boldsymbol{G}_1\boldsymbol{\Lambda}(\omega M)\boldsymbol{QJ}\\ &= \boldsymbol{G}_{L-2}\boldsymbol{\Lambda}(\omega M)\boldsymbol{V}_{L-2}\\ &\vdots\\ \boldsymbol{V}_2 &= \boldsymbol{G}_1\boldsymbol{\Lambda}(\omega M)\boldsymbol{QJ} = \boldsymbol{G}_1\boldsymbol{\Lambda}(\omega M)\boldsymbol{V}_1\\ \boldsymbol{V}_1 &= \boldsymbol{QJ}.\end{aligned}$$

Also define the terms

$$\boldsymbol{D}_k(\omega) = \frac{\partial^k \boldsymbol{\Lambda}(\omega M)}{\partial \omega^k} = \begin{bmatrix}\boldsymbol{0} & \boldsymbol{0}\\ \boldsymbol{0} & (-jM)^k e^{-j\omega M}\boldsymbol{I}\end{bmatrix},$$

$$\boldsymbol{d}_k(\omega) = \left[(-j[M-1])^k e^{-j(M-1)\omega} \cdots (-j)^k e^{-j\omega} \quad 0\right]^T,$$

$$\boldsymbol{B}(\omega) = \begin{bmatrix}\boldsymbol{U}_1^T\\ \vdots\\ \boldsymbol{U}_L^T\end{bmatrix}, \quad \boldsymbol{C}(\omega) = \begin{bmatrix}\boldsymbol{V}_1\\ \vdots\\ \boldsymbol{V}_L\end{bmatrix},$$

$$\boldsymbol{Y}_k(\omega) = \begin{bmatrix}\boldsymbol{D}_k(\omega) & \boldsymbol{0} & \boldsymbol{0}\\ \boldsymbol{0} & \ddots & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{D}_k(\omega)\end{bmatrix},$$

$$\boldsymbol{L}_k(\omega) = \boldsymbol{B}(\omega)^T \boldsymbol{Y}_k(\omega)\boldsymbol{C}(\omega).$$

Then for a first order derivative constraint one can use the expression

$$\frac{\partial F_j(\omega)}{\partial \omega} = \boldsymbol{d}_1^T(\omega)\boldsymbol{R}(\omega M)\boldsymbol{u}_j + \boldsymbol{e}^H(\omega)\boldsymbol{L}_1(\omega)\boldsymbol{u}_j.$$

Furthermore, define the matrices

$$\boldsymbol{B}_k(\omega) = \begin{bmatrix}\partial^k \boldsymbol{U}_1^T / \partial\omega^k\\ \vdots\\ \partial^k \boldsymbol{U}_L^T / \partial\omega^k\end{bmatrix}, \quad \boldsymbol{C}_k(\omega) = \begin{bmatrix}\partial^k \boldsymbol{V}_1 / \partial\omega^k\\ \vdots\\ \partial^k \boldsymbol{V}_L / \partial\omega^k\end{bmatrix}$$

$$\boldsymbol{L}_1'(\omega) = \boldsymbol{B}_1(\omega)^T \boldsymbol{Y}_1(\omega)\boldsymbol{C}(\omega) + \boldsymbol{L}_2(\omega) + \boldsymbol{B}(\omega)^T \boldsymbol{Y}_1(\omega)\boldsymbol{C}_1(\omega),$$

and note that

$$\frac{\partial \boldsymbol{U}_j}{\partial \omega} = \frac{\partial \boldsymbol{U}_{j+1}}{\partial \omega}\boldsymbol{\Lambda}(\omega M)\boldsymbol{G}_j + \boldsymbol{U}_{j+1}\boldsymbol{D}_1(\omega)\boldsymbol{G}_j$$

$$\frac{\partial \boldsymbol{V}_j}{\partial \omega} = \boldsymbol{G}_{j-1}\boldsymbol{D}_1(\omega)\boldsymbol{V}_{j-1} + \boldsymbol{G}_{j-1}\boldsymbol{\Lambda}(\omega M)\frac{\partial \boldsymbol{V}_{j-1}}{\partial \omega}.$$

Then a second order derivative constraint can be specified using the formula

$$\frac{\partial^2 F_j(\omega)}{\partial \omega^2} = \boldsymbol{d}_2^T(\omega)\boldsymbol{R}(\omega M)\boldsymbol{u}_j + 2\boldsymbol{d}_1^T(\omega)\boldsymbol{L}_1(\omega)\boldsymbol{u}_j + \boldsymbol{e}^H(\omega)\boldsymbol{L}_1'(\omega)\boldsymbol{u}_j.$$

Higher order derivatives can be computed by repeated application of the product rule. A filter that satisfies this constraint for a particular frequency ω_i will have its phase response fixed.

4.2. Phase Unconstrained Solution

A first order derivative constraint imposed on the power response of a filter can be written using the expression

$$\frac{\partial |F_j(\omega)|^2}{\partial \omega} = 2\,\mathrm{Re}\left\{\boldsymbol{w}^H(\omega)\boldsymbol{u}_j\boldsymbol{u}_j^T\boldsymbol{R}^H(\omega M)\boldsymbol{e}(\omega)\right\}$$

where

$$\boldsymbol{w}^H(\omega) = \boldsymbol{d}_1^H(\omega)\boldsymbol{R}(\omega M) + \boldsymbol{e}^H(\omega)\boldsymbol{L}_1(\omega).$$

Higher order derivatives can be obtained by repeated application of the product rule. Filters that satisfy this type of constraint are free to have arbitrary phase.

5. RESULTS

The approach described in this paper was used to design a FIR PU approximation to an ideal PCFB for the case where $M = 4$ and the length of the channel filters is 8. PCFBs are optimal filter banks for a variety of applications, such as maximizing coding gain [4]. They were first described by Tsatsanis in [5] and are filter banks that depend on the characteristics of the input signal. Ideal PCFBs consist of brickwall filters with possibly many narrow band pass regions. FIR approximations to PCFBs are difficult filters to design with high frequency selectivity in the narrow bandpass regions. Previously developed algorithms for designing FIR compaction filters can be used to sequentially approximate the filters in a PCFB [6], [7]. Recent techniques for jointly optimizing all the filters in a FIR approximation to a PCFB include an elegant greedy algorithm reported by Tkacenko [8]. The results in this paper will be compared to filters designed using Tkacenko's approach.

Figures 1 and 2 illustrate the brickwall response of selected filters in an ideal PCFB and superimpose the response of filters derived using unconstrained optimization and the filters derived using a first order derivative constraint of the first type. Also shown are the filters derived using Tkacenko's methodology. The derivative constraint was specified for the second filter in the filter bank and set equal to zero at the normalized frequency $\omega = 0.135$, which effectively flattens the gain of the filter at that frequency. Note that the imposed constraint is also satisfied at $\omega = 0.865$ because the filter coefficients are real. As is clear from the figures, the filters designed using derivative constraints have higher gain and better selectivity in the pass band region than the Tkacenko filters as well as the filters designed using unconstrained optimization. Figure 3 is a magnified view of the pass band region where the derivative constraint is satisfied, clearly showing a desirable flat filter gain.

Figures 4 and 5 compare the frequency response of a phase constrained solution to a phase unconstrained solution where both solutions satisfy a first order derivative constraint of the second type. The constraint was set for the second filter in the filter bank at $\omega = 0.135$. Both the phase constrained and the phase unconstrained solutions exhibit excellent response in the pass band regions, but the benefit of the phase unconstrained solution is that the filters can have any phase response.

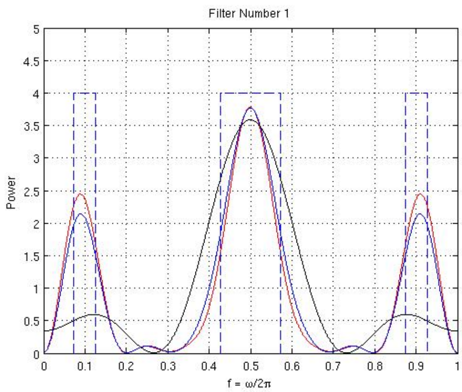

Figure 1. Uncons. (blue), Phase Cons. (red), Tkacenko (black)

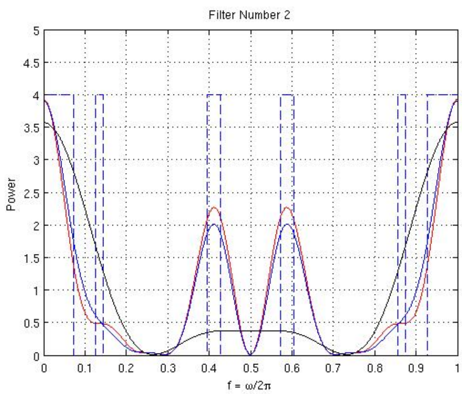

Figure 2. Uncons., Phase Cons., Tkacenko - Filter 2

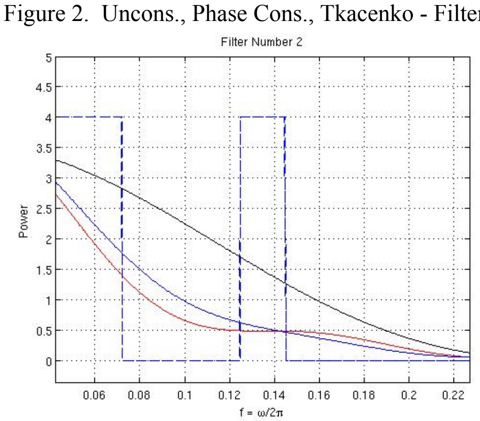

Figure 3. Filter 2 Close-Up

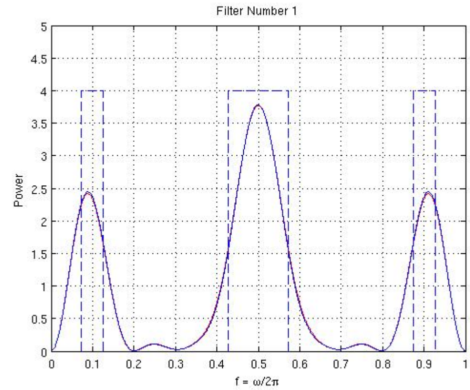

Figure 4. Phase Cons. (blue) vs Phase Uncons. (red)

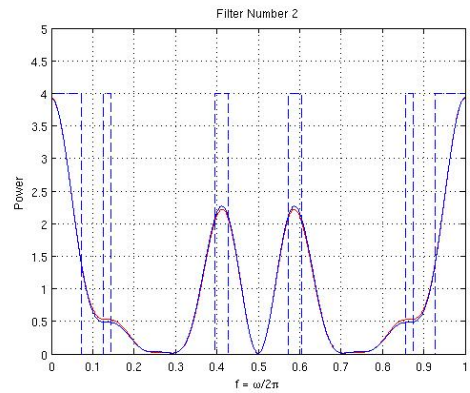

Figure 5. Phase Cons. vs Phase Uncons. - Filter 2

6. REFERENCES

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

[2] G. Strang and T. Nguyen, *Wavelets and Filter Banks*, Wellesley-Cambridge Press, Wellesley, MA, 1997.

[3] P. Vouras and T. Tran, "Design of FIR Paraunitary Approximations to Principal Component Filter Banks," *Proceedings Conference on Information Sciences and Systems*, Princeton, NJ, March 2006.

[4] S. Akkarakaran and P. P. Vaidyanathan, "Filterbank Optimization with Convex Objectives and the Optimality of Principal Component Forms," *IEEE Trans. Signal Processing*, pp. 100 - 114, Jan. 2001.

[5] M. K. Tsatsanis and G. B. Giannakis, "Principal Component Filter Banks for Optimal Multiresolution Analysis," *IEEE Trans. Signal Processing*, pp. 1766 - 1777, Aug. 1995.

[6] P. Moulin and M. Mihçak, "Theory and Design of Signal-Adapted FIR Paraunitary Filter Banks," *IEEE Trans. Signal Processing*, pp. 920 - 929, April 1998.

[7] J. Tuqan and P. P. Vaidyanathan, "A State Space Approach to the Design of Globally Optimal FIR Energy Compaction Filters," *IEEE Trans. Signal Processing*, pp. 2822 – 2838, Oct. 2000.

[8] A. Tkacenko and P. P. Vaidyanathan, "Iterative Greedy Algorithm for Solving the FIR Paraunitary Approximation Problem," *IEEE Trans. Signal Processing*, pp. 146 - 160, Jan. 2006.