Design of FIR Paraunitary Approximations to Principal Component Filter Banks

Peter Vouras and Trac D. Tran, *Johns Hopkins University*

***Abstract*—In this paper we present two new algorithms for designing finite impulse response (FIR) paraunitary (PU) filter banks that are optimal approximations to ideal principal component filter banks (PCFBs). The first algorithm is optimal in the sense that it minimizes the mean square error between the desired response and the approximation. The second algorithm is optimal in the sense that it minimizes the maximum error. Both algorithms utilize a complete parameterization of FIR PU filter banks in terms of Givens rotation building blocks, and jointly optimize all the filters in the filter bank.**

***Index Terms*—FIR paraunitary filter bank design, principal component filter bank**

I. Introduction

Consider the situation shown in Fig. 1 which illustrates a maximally decimated filter bank with M channels.

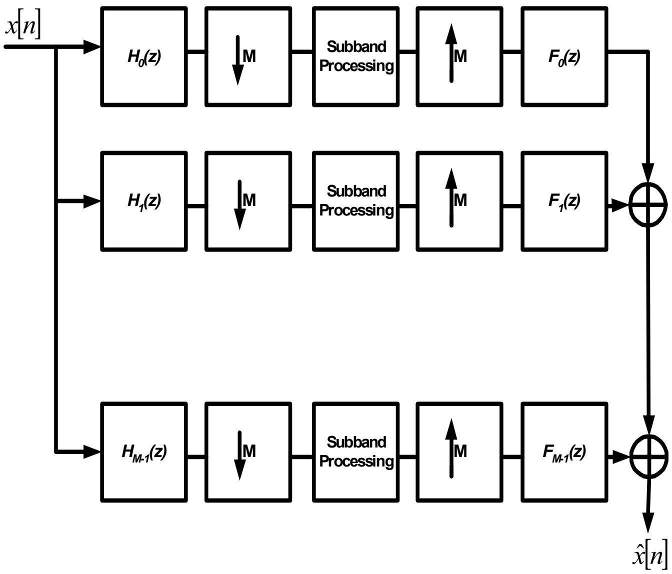

Fig. 1. Maximally Decimated Filter Bank

Each analysis filter $H_k(z)$ with impulse response $h_k(n)$ in Fig. 1 can be expressed in terms of its polyphase components as,

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

where $e_{kl}(n) = h_k(l + Mn), \quad 0 \le l \le M-1$ (2)

and $E_{kl}(z) = \sum_{n=-\infty}^{\infty} e_{kl}(n) z^{-n}$. (3)

Similarly, each synthesis filter can be expressed as,

$$F_k(z) = \sum_{l=0}^{M-1} z^{-(M-1-l)} R_{lk}\left(z^M\right), \quad k = 0,\ldots,M-1 . \tag{4}$$

Now define the M x M polyphase component matrices $E(z)$ and $R(z)$ as,

$$E(z) = \left[E_{kl}(z)\right], \quad R(z) = \left[R_{lk}(z)\right]. \tag{5}$$

Based on the polyphase representation of the analysis and synthesis filters, an equivalent but more efficient implementation of the filter bank in Fig. 1 is shown in Fig. 2.

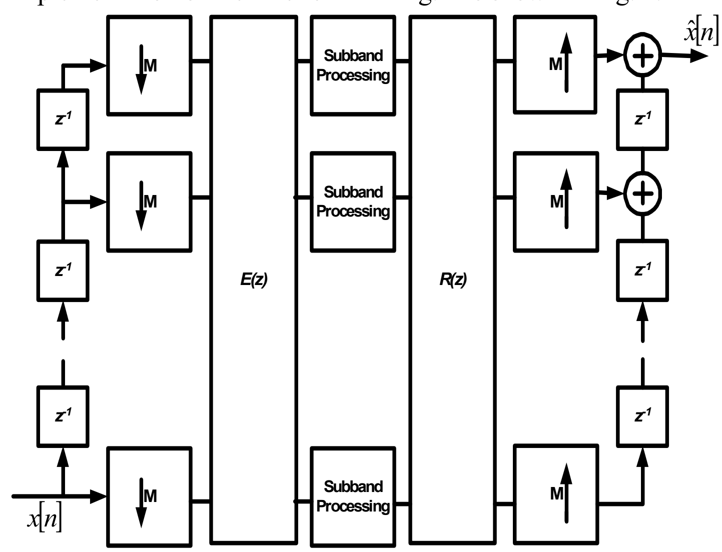

Fig. 2. Filter Bank Polyphase Implementation

Suppose we wish to find the optimum analysis and synthesis filters $H_k(z)$ and $F_k(z)$, $k = 0,\ldots,Q-1$ with $Q < M$, such that if the input $x(n)$ is a stationary random process, the filter bank output $\hat{x}(n)$ best approximates $x(n)$ in the mean square sense. The solution is that,

$$E(\omega) = \begin{bmatrix} v_1^H(\omega) \\ \vdots \\ v_Q^H(\omega) \end{bmatrix}, \quad R(\omega) = E^H(\omega) \tag{6}$$

where $v_i(\omega)$ is the eigenvector corresponding to the ith largest eigenvalue of $S_{XX}(\omega)$, the spectral density matrix of $x(n)$. Such a filter bank is called a PCFB [1].

PCFBs have been shown to be optimal for a variety of signal processing applications. For example, in compression applications, PCFBs maximize coding gain and minimize the mean-squared error caused by omitting from the reconstruction process the $(M-Q)$ subbands out of M with the lowest variance. In fact, a PCFB is optimal whenever the objective function is a concave function of the subband variances [2]. One property of PCFBs is that if the input signal $x(n)$ is wide sense stationary (WSS), then the subband variance vector of the PCFB,

$$\sigma = \left[\sigma_0^2,\quad \sigma_1^2, \cdots, \sigma_{M-1}^2\right]^T, \tag{7}$$

majorizes the subband variance vector created by any other filter bank. Furthermore, if $Q = M$ then the signals in the M channels are uncorrelated with each other.

The analysis and synthesis filters of an ideal PCFB have brick wall frequency responses and can only be approximated by FIR filters. Some techniques exist for approximating a PCFB by sequentially designing optimal FIR energy compaction filters [3]. Separately, one recent technique by Tkacenko describes an iterative greedy algorithm which jointly optimizes all the filters in a filter bank to approximate a PCFB [4]. Our paper presents two new algorithms which may also be used to simultaneously design all the filters in a FIR PU filter bank approximation to an ideal PCFB. Sample filter design results presented later in this paper and compared to the output generated by the algorithm described in [4] show better approximation performance.

II. ALGORITHM 1: MINIMUM MEAN SQUARE ERROR DESIGN

The first proposed algorithm minimizes the weighted mean squared Frobenius norm error (MSE) between the desired polyphase matrix of the ideal PCFB, $D(\omega)$, and the FIR PU synthesis polyphase matrix, $R(\omega)$. The objective function to be minimized is,

$$\eta = \frac{1}{2\pi}\int_0^{2\pi} W(\omega) \| D(\omega) - R(\omega) \|_F^2 \, d\omega \tag{8}$$

where $W(\omega)$ is a scalar nonnegative weight function. To solve the problem we decompose $R(\omega)$ into the following form,

$$R(\omega) = G_L \Lambda(\omega) \cdots G_1 \Lambda(\omega) Q J \tag{9}$$

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

$$\Lambda(\omega) = \begin{bmatrix} I & 0 \\ 0 & e^{-j\omega} I \end{bmatrix}, \quad J = \begin{bmatrix} \pm 1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \pm 1 \end{bmatrix}. \tag{10}$$

Each matrix G_k and Q is the product of $\frac{1}{2}M(M-1)$ Givens rotation matrices of the form,

$$S_{ij}(\theta_k) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & \cos(\theta_k) & \cdots & \sin(\theta_k) & 0 \\ 0 & \vdots & 1 & \vdots & 0 \\ 0 & -\sin(\theta_k) & \cdots & \cos(\theta_k) & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \tag{11}$$

$$1 \le k \le \frac{LM}{2}(M-1),$$

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 S_{ij} in the product is important. In general, a M-by-M orthogonal matrix G_k can be decomposed into the following product sequence of Givens rotation matrices [5],

$$G_k = \{S_{M-2,M-1}\} \cdots \{S_{1,M-1} \cdots S_{12}\}\{S_{0,M-1} \cdots S_{01}\}. \tag{12}$$

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

$$G_k(\Theta) = \{S_{23}(\theta_1)\}\{S_{13}(\theta_2) S_{12}(\theta_3)\}\{S_{03}(\theta_4) S_{02}(\theta_5) S_{01}(\theta_6)\} \tag{13}$$

where we start counting rows and columns from zero, and $\Theta = [\theta_1, \ldots, \theta_6]^T$. By embedding the polyphase matrix decomposition (9) into the objective function (8), the solution filter bank is guaranteed to be PU with $E(\omega) = R^H(\omega)$.

A. Unconstrained Minimization Algorithm

After substituting the decomposition for $R(\omega)$ given in (9) into (8), there are $\frac{LM(M-1)}{2}$ 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,

$$\varphi = \sum_{j=1}^{\frac{LM(M-1)}{2}} \left(\max\{0, \theta_j - 2\pi\}\right)^2 + \left(\max\{0, -2\pi - \theta_j\}\right)^2. \tag{14}$$

We can now solve the unconstrained minimization problem,

$$\min \quad \xi = \eta + \alpha\varphi, \tag{15}$$

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.

Since the objective function $\xi(\Theta)$, with $\Theta = \left[\theta_1, \theta_2, \ldots, \theta_{\frac{1}{2}LM(M-1)}\right]^T$, is twice continuously differentiable, whenever the gradient at a point $\hat{\Theta}$ equals zero, and the Hessian matrix is positive definite when evaluated at $\hat{\Theta}$, then $\hat{\Theta}$ is guaranteed to be a strict local minimum of $\xi(\Theta)$, and the algorithm terminates. The algorithm may converge to different local minima depending on the initial starting point.

The gradient of the objective function in (15) is given by,

$$\nabla\xi(\Theta)=\left[\frac{\partial\xi(\Theta)}{\partial\theta_1},\ldots,\frac{\partial\xi(\Theta)}{\partial\theta_{LM(M-1)/2}}\right]^T. \qquad (16)$$

If the matrix G_k depends on θ_j through the factor $S_{pq}(\theta_j)$, then using the matrix

$$V_j(\Theta,\omega)=D^H(\omega)G_L\Lambda(\omega)\cdots\frac{\partial G_k}{\partial\theta_j}\Lambda(\omega)\cdots G_1\Lambda(\omega)QJ \qquad (17)$$

we have

$$\frac{\partial\xi(\Theta)}{\partial\theta_j}=\frac{-1}{2\pi}\int_0^{2\pi}Trace\left\{V_j(\Theta,\omega)+V_j(\Theta,\omega)^H\right\}d\omega+\alpha\frac{\partial\varphi}{\partial\theta_j} \qquad (18)$$

where

$$\frac{\partial G_k}{\partial\theta_j}=\{S_{M-2,M-1}\}\cdots\frac{\partial S_{pq}(\theta_j)}{\partial\theta_j}\cdots\{S_{0,M-1}\cdots S_{01}\} \qquad (19)$$

and

$$\frac{\partial S_{pq}(\theta_j)}{\partial\theta_j}=\begin{bmatrix}0&0&0&0&0\\0&-\sin(\theta_j)&\cdots&\cos(\theta_j)&0\\0&\vdots&0&\vdots&0\\0&-\cos(\theta_j)&\cdots&-\sin(\theta_j)&0\\0&0&0&0&0\end{bmatrix}, \qquad (20)$$

$$\frac{\partial\varphi}{\partial\theta_j}=2\cdot\max\{0,\theta_j-2\pi\}+2\cdot\max\{0,-2\pi-\theta_j\}.$$

The Hessian matrix is given by,

$$\nabla^2\xi(\Theta)=\begin{bmatrix}\frac{\partial^2\xi(\Theta)}{\partial\theta_1\partial\theta_1}&\cdots&\frac{\partial^2\xi(\Theta)}{\partial\theta_{LM(M-1)/2}\partial\theta_1}\\\vdots&\ddots&\vdots\\\frac{\partial^2\xi(\Theta)}{\partial\theta_1\partial\theta_{LM(M-1)/2}}&\cdots&\frac{\partial^2\xi(\Theta)}{\partial\theta_{LM(M-1)/2}\partial\theta_{LM(M-1)/2}}\end{bmatrix}. \qquad (21)$$

If the matrix G_k depends on θ_j through the factor $S_{pq}(\theta_j)$, and the matrix G_n depends on θ_m through the factor $S_{rs}(\theta_m)$, then using the matrix

$$V_{jm}(\Theta,\omega)=D^H(\omega)G_L\Lambda(\omega)\cdots\frac{\partial G_k}{\partial\theta_j}\cdots\frac{\partial G_n}{\partial\theta_m}\cdots G_1\Lambda(\omega)QJ \qquad (22)$$

we have

$$\frac{\partial^2\xi(\Theta)}{\partial\theta_j\partial\theta_m}=\frac{-1}{2\pi}\int_0^{2\pi}Trace\left\{V_{jm}(\Theta,\omega)+V_{jm}(\Theta,\omega)^H\right\}d\omega+\frac{\alpha\partial^2\varphi}{\partial\theta_j\partial\theta_m}, \qquad (23)$$

and define

$$V_{jj}(\Theta,\omega)=D^H(\omega)G_L\Lambda(\omega)\cdots\frac{\partial^2 G_k}{\partial\theta_j\partial\theta_j}\cdots G_1\Lambda(\omega)QJ \qquad (24)$$

so that

$$\frac{\partial^2\xi(\Theta)}{\partial\theta_j\partial\theta_j}=\frac{-1}{2\pi}\int_0^{2\pi}Trace\left\{V_{jj}(\Theta,\omega)+V_{jj}(\Theta,\omega)^H\right\}d\omega+\frac{\alpha\partial^2\varphi}{\partial\theta_j\partial\theta_j}. \qquad (25)$$

Furthermore,

$$\frac{\partial^2 G_k}{\partial\theta_j\partial\theta_j}=\{S_{M-2,M-1}\}\cdots\frac{\partial^2 S_{pq}(\theta_j)}{\partial\theta_j\partial\theta_j}\cdots\{S_{0,M-1}\cdots S_{01}\} \qquad (26)$$

and

$$\frac{\partial^2 S_{pq}(\theta_j)}{\partial\theta_j\partial\theta_j}=\begin{bmatrix}0&0&0&0&0\\0&-\cos(\theta_j)&\cdots&-\sin(\theta_j)&0\\0&\vdots&0&\vdots&0\\0&\sin(\theta_j)&\cdots&-\cos(\theta_j)&0\\0&0&0&0&0\end{bmatrix}. \qquad (27)$$

If the matrix G_k depends on θ_j through the factor $S_{pq}(\theta_j)$, and it depends on θ_m through the factor $S_{rs}(\theta_m)$, then

$$V_{jm}(\Theta,\omega)=D^H(\omega)G_L\Lambda(\omega)\cdots\frac{\partial^2 G_k}{\partial\theta_j\partial\theta_m}\cdots G_1\Lambda(\omega)QJ, \qquad (28)$$

and

$$\frac{\partial^2 G_k}{\partial\theta_j\partial\theta_m}=S_{M-2,M-1}\cdots\frac{\partial S_{pq}(\theta_j)}{\partial\theta_j}\cdots\frac{\partial S_{rs}(\theta_m)}{\partial\theta_m}\cdots S_{01}. \qquad (29)$$

Lastly,

$$\frac{\partial^2\varphi}{\partial\theta_j\partial\theta_m}=0, \qquad (30)$$

and

$$\frac{\partial^2\varphi}{\partial\theta_j\partial\theta_j}=\begin{Bmatrix}0 & if\ -2\pi\le\theta_j\le 2\pi\\2 & if\ \theta_j>2\pi\\-2 & if\ \theta_j<-2\pi\end{Bmatrix}. \qquad (31)$$

B. Phase Ambiguity

Each column, $d_j(\omega)$, of the desired polyphase matrix, $D(\omega)$, is an unit norm eigenvector of the power spectral density matrix $S_{XX}(\omega)$. Since multiplying a unit norm eigenvector by a scalar with unit magnitude yields a unit norm eigenvector, $D(\omega)$ can be multiplied by a diagonal matrix of complex exponentials and still be considered a valid desired response. If a given desired response is modified as,

$$D_m(\omega)=D(\omega)L(\omega), \qquad (32)$$

with

$$L(\omega)=\begin{bmatrix}e^{j\psi_0(\omega)}&0&0&0\\0&e^{j\psi_1(\omega)}&0&0\\0&0&\ddots&0\\0&0&0&e^{j\psi_{M-1}(\omega)}\end{bmatrix}, \qquad (33)$$

then choosing the phases $\psi_j(\omega)$ judiciously can result in lower mean squared error. Define the columns of the FIR PU polyphase approximation matrix $R(\omega)$ to be $r_j(\omega)$ and consider the complex representation of the inner product,

$$d_j^H(\omega)r_j(\omega)=\left|d_j^H(\omega)r_j(\omega)\right|e^{j\chi_j(\omega)}. \qquad (34)$$

Setting $\psi_j(\omega)=\chi_j(\omega)$, as stipulated by Tkacenko in [4], results in the lowest mean squared error.

C. Pseudocode for Algorithm 1

The steps for computing a FIR PU approximation to the ideal PCFB which minimizes the mean squared Frobenius norm error defined in (8) are as follows;

1. Choose a random initial starting vector, $\in$, of angles and an initial value for the matrix J
2. Run the BFGS algorithm with inexact line search until convergence (can use MATLAB function *fminunc*)
3. Apply phase modification
4. Evaluate gradient and Hessian. If sufficient conditions for local minimum are not satisfied, go to Step 2 and repeat.

An exhaustive search of all 2^M possible values of the matrix J is necessary to verify the optimal solution.

D. Results for Algorithm 1

A simulation was used to test the performance of Algorithm 1 and its results were compared to the filters designed using the iterative greedy algorithm described in [4], which also minimizes a mean squared error criterion. The input for the simulation was an auto-regressive process with 4 poles and the power spectral density shown in Fig. 3.

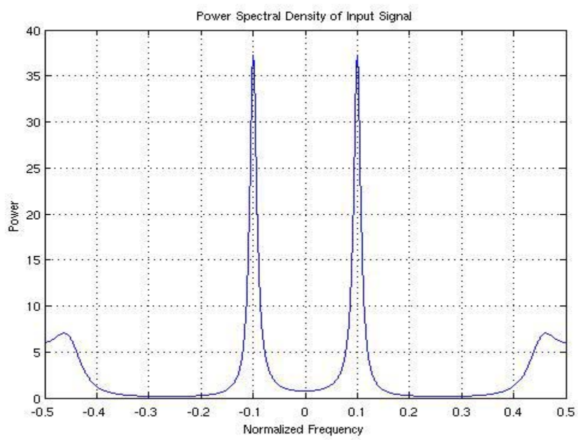

Fig. 3. Power Spectral Density of Input Process

The FIR PU approximant was a filter bank of Smith-McMillan degree equal to 2 and $M = 4$ channels. The resulting length of the FIR filters was 8. All the designed filters are orthogonal to each other. The final value of the mean squared error at the conclusion of Algorithm 1 was 1.5551, compared to 2.4028 at the conclusion of the iterative greedy algorithm. At the optimal solution, the infinity norm of the gradient vector was equal to 3.707e-5. Also, the Hessian matrix evaluated at the optimal vector of angles was verified to be positive definite. After an exhaustive search of all the possible values for the matrix *J*, the value which yielded optimal results was found to be

$$J = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{35}$$

Fig. 4 plots the frequency response of the optimal filter for channel 1, and for comparison Fig. 5 shows the filter designed using Tkacenko's algorithm in [4]. The remaining plots in Figs. 6 through 11 show the responses of the other channel filters, arranged in similar fashion. The brickwall response of the ideal filter is superimposed on each plot.

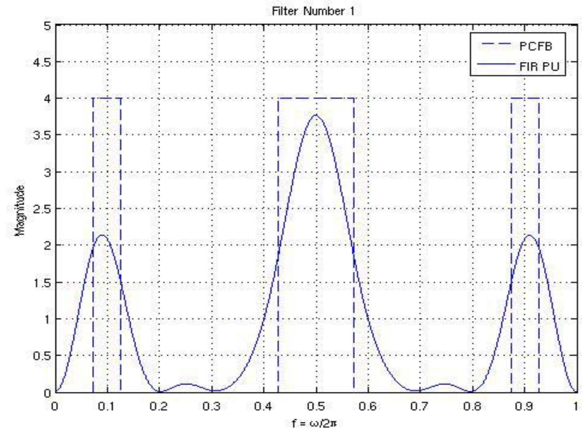

Fig. 4. MSE Optimal Filter for Channel 1

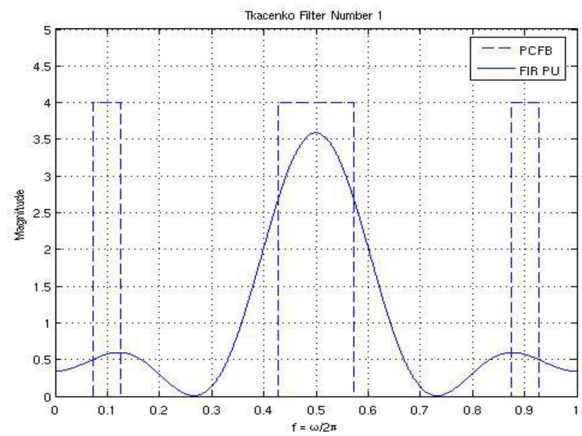

Fig. 5. Filter 1 Designed Using Greedy Algorithm

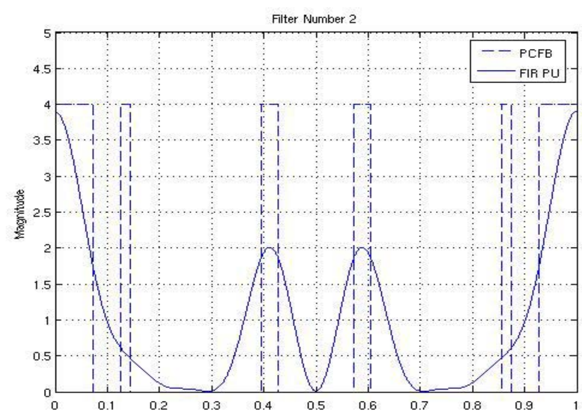

Fig. 6. MSE Optimal Filter for Channel 2

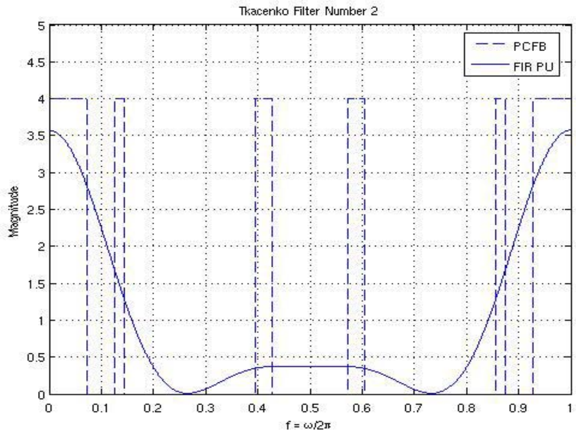

Fig. 7. Filter 2 Designed Using Greedy Algorithm

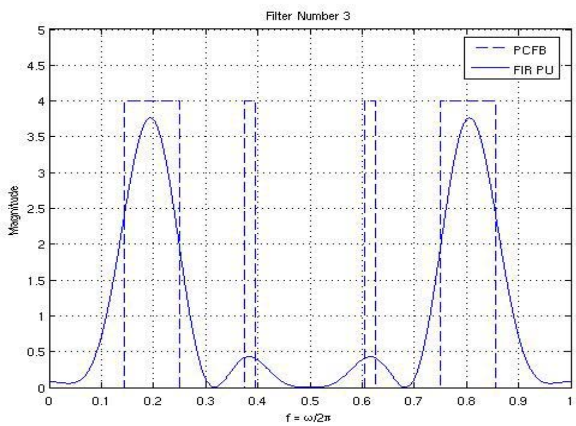

Fig. 8. MSE Optimal Filter for Channel 3

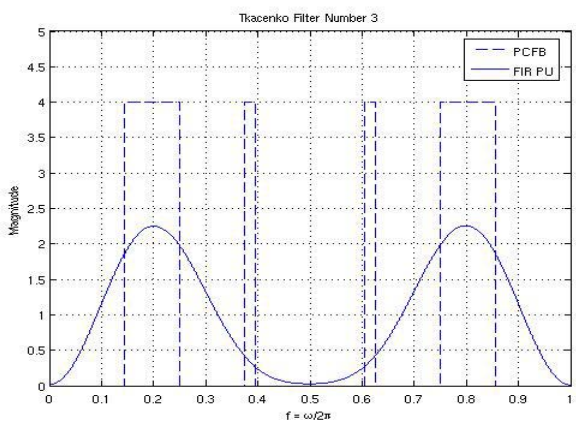

Fig. 9. Filter 3 Designed Using Greedy Algorithm

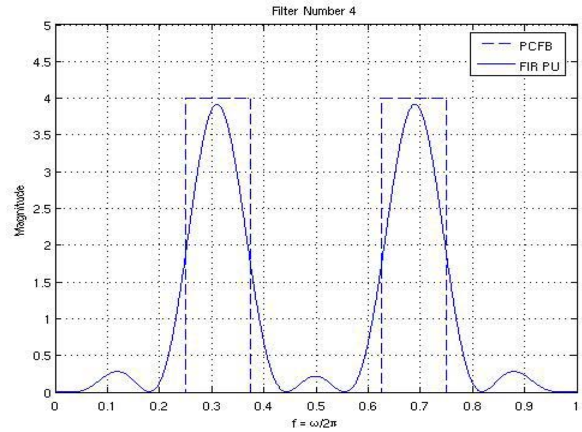

Fig. 10. MSE Optimal Filter for Channel 4

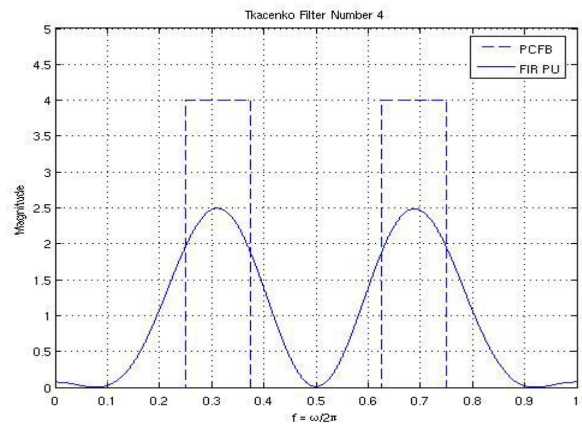

Fig. 11. Filter 4 Designed Using Greedy Algorithm

As is evident from the figures, the filters designed using Algorithm 1 tend to have high gain in the wide passband regions and somewhat smaller gain in the narrower passband regions.

III. ALGORITHM 2: MINIMUM MAXIMUM ERROR DESIGN

A different objective function may be used to estimate a FIR PU approximation to the ideal PCFB, resulting in filters that are excellent solutions, but with a different response than the filters designed using Algorithm 1. These filters may be better suited to some specific applications. The second proposed algorithm minimizes the maximum Frobenius norm squared error over a finite, but large, set of N discrete frequencies $\{\omega_k\}$. The objective function in this case is defined as,

$$\min_{\omega_i} \max \xi_i , \qquad (36)$$

where

$$\xi_i = W(\omega_i)\| D(\omega_i) - R(\omega_i) \|_F^2 + \alpha\varphi, \quad 1 \le i \le N. \qquad (37)$$

Algorithm 2 minimizes the maximum value of ξ_i over all the frequencies ω_i using a Sequential Quadratic Programming (SQP) method.

A. *Pseudocode for Algorithm 2*

The steps for Algorithm 2, which computes a FIR PU filter bank that minimizes the maximum error, are as follows;

1. Choose a random initial starting vector, $\in$, of angles and an initial value for the matrix J
2. Pick N and define a frequency grid, ω_i, $1 \le i \le N$
3. Run the SQP algorithm to minimize the maximum value of ξ_i until convergence (can use MATLAB function *fminimax*)
4. Apply phase modification
5. Go to Step 3 and repeat K times until $\xi_{K+1} - \xi_K < \varepsilon$, where ε is some small user defined threshold.

B. *Results for Algorithm 2*

Using the same input process described by Fig. 3, filters designed using Algorithm 2 were computed by simulation. The results are shown in Figs. 12 through 15.

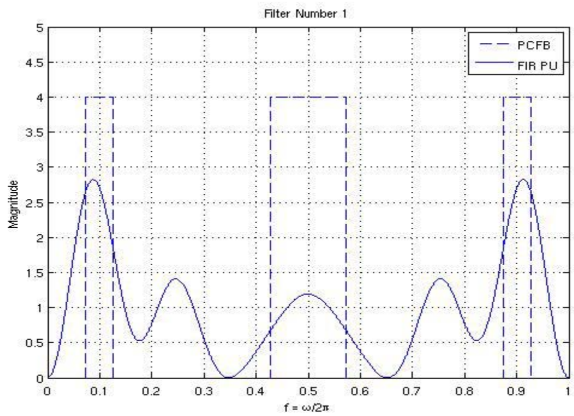

Fig. 12. Min-Max Optimal Filter for Channel 1

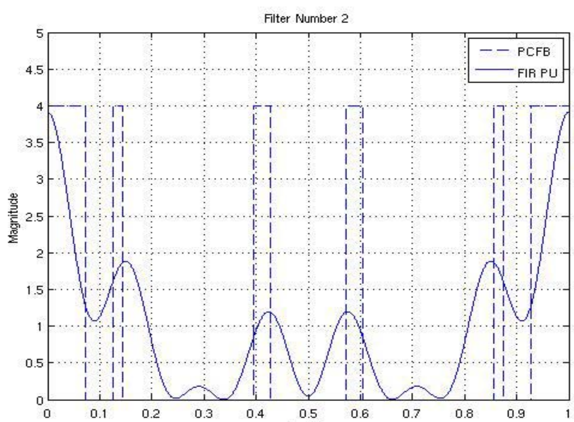

Fig. 13. Min-Max Optimal Filter for Channel 2

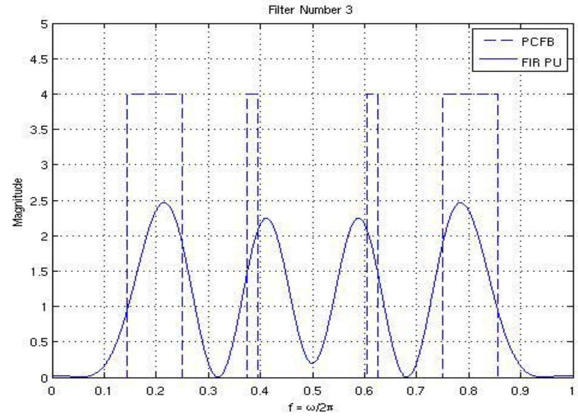

Fig. 14. Min-Max Optimal Filter for Channel 3

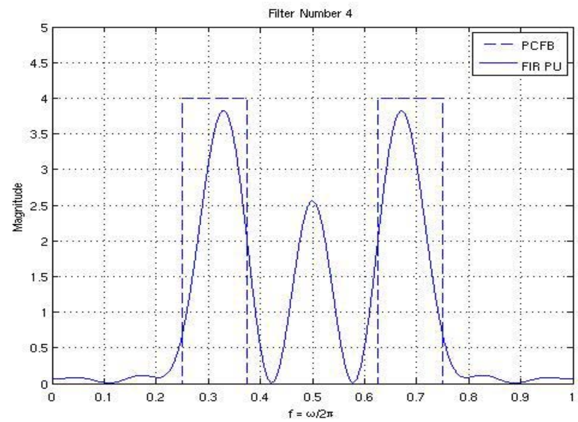

Fig. 15. Min-Max Optimal Filter for Channel 4

The filters designed using Algorithm 2, show higher gain in some of the narrow passband regions than the filters designed using Algorithm 1.

REFERENCES

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

[2] 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.

[3] 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.

[4] 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.

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