

# DISCRETE-TIME DITHERING TECHNIQUE FOR DESIGNING TIME-VARYING SIGNAL ADAPTED FILTER BANKS

Peter G. Vouras and Trac D. Tran

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

## ABSTRACT

This paper describes the use of sinusoidal dithers in a lattice filter bank structure to create a time-varying paraunitary filter bank that approximates and tracks in near real-time the ideal principal component filter bank corresponding to a nonstationary random input process. The sinusoidal dithers are used to estimate the gradient of a cost function which is minimized using non-linear minimization techniques.

**Index Terms**— dithers, paraunitary filter banks, principal component filter banks

## 1. INTRODUCTION

The theory of filter banks is well described in [1]. Fig. 1 is a conceptual schematic of a maximally decimated filter bank with  $M$  channels implemented using polyphase matrices.

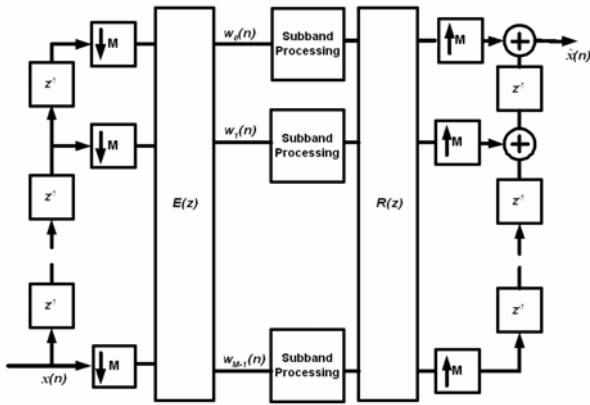


Fig. 1. Polyphase Implementation of Filter Bank

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

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

where

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

and

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

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

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

The  $M$ -by- $M$  polyphase component matrices  $\mathbf{E}(z)$  and  $\mathbf{R}(z)$  are

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

The polyphase matrix evaluated for  $z = e^{j\omega}$  is denoted by  $\mathbf{E}(\omega)$  or  $\mathbf{R}(\omega)$ .

## 2. PRINCIPAL COMPONENT FILTER BANKS

A special class of filter banks is Principal Component Filter Banks (PCFBs). These filter banks were originally described in [2]. Define the blocked version of the input sequence to a filter bank to be the  $M$ -by-1 vector

$$\mathbf{x}(n) = [x(nM) \quad x(nM-1) \quad \dots \quad x(nM-M+1)]^T$$

and define the blocked output vector to be

$$\hat{\mathbf{x}}(n) = [\hat{x}(nM) \quad \hat{x}(nM-1) \quad \dots \quad \hat{x}(nM-M+1)]^T.$$

Suppose we wish to find the optimal  $Q$ -by- $M$  and  $M$ -by- $Q$  matrices  $\mathbf{E}(n)$  and  $\mathbf{R}(n)$  with  $Q < M$  such that the time-averaged mean squared error between the vector input and output is minimized, as in

$$\min \xi = \lim_{N \rightarrow \infty} \frac{1}{2N+1} \sum_{n=-N}^N E \left\{ [\mathbf{x}(n) - \hat{\mathbf{x}}(n)]^T [\mathbf{x}(n) - \hat{\mathbf{x}}(n)] \right\}.$$

The solution filter bank is called a PCFB and is given by

$$\mathbf{E}(\omega) = \begin{bmatrix} \mathbf{v}_1^H(\omega) \\ \vdots \\ \mathbf{v}_Q^H(\omega) \end{bmatrix}$$

$$\mathbf{R}(\omega) = \mathbf{E}^H(\omega)$$

where  $\mathbf{v}_i(\omega)$  is the eigenvector corresponding to the  $i$ th largest eigenvalue of  $\hat{\mathbf{S}}_{\mathbf{x}\mathbf{x}}(\omega)$ , the time averaged spectral density matrix of the input vector process. Using this construction for the polyphase matrices  $\mathbf{E}(\omega)$  and  $\mathbf{R}(\omega)$

when  $Q = M$  results in a paraunitary (PU)  $M$ -channel perfect reconstruction filter bank, albeit one of infinite order.

PCFBs have special properties but they only exist for special classes of filter banks [3]. If a PCFB exists for an input power spectral density (PSD) matrix  $\mathbf{S}_{xx}(\omega)$  and for a class  $C$  of filter banks, then the subband variance vector of the PCFB,

$$\boldsymbol{\sigma} = \left[ \sigma_{W_0}^2 \quad \sigma_{W_1}^2 \quad \cdots \quad \sigma_{W_{M-1}}^2 \right]^T,$$

majorizes the subband variance vector created from any other filter bank in  $C$ . It has been shown that PCFBs are optimal for maximizing coding gain, and minimizing mean-squared error in the presence of quantization noise, but they are also optimal for any concave objective function of the subband variance vector [4].

If  $C$  is the class of all paraunitary filter banks with infinite order, then the PCFB exists and is the pointwise in frequency Karhunen-Loève transform (KLT) for  $\mathbf{x}(n)$ . In other words,  $\mathbf{R}(\omega)$  diagonalizes, and thereby totally decorrelates,  $\mathbf{S}_{xx}(\omega)$  for every  $\omega$  such that the frequency dependent eigenvalues are always arranged in decreasing order.

### 3. PARAUNITARY FILTER BANKS

A paraunitary filter bank satisfies the condition that

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

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

$$\mathbf{R}(\omega) = \mathbf{G}_L \boldsymbol{\Lambda}(\omega) \cdots \mathbf{G}_1 \boldsymbol{\Lambda}(\omega) \mathbf{Q} \mathbf{J}$$

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

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

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

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

$$1 \leq k \leq \frac{1}{2}(L+1)M(M-1),$$

where  $\cos(\theta_k)$  is placed in the  $i$ th row and  $i$ th 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  $\mathbf{S}_{ij}$  in the product is important. In general, an  $M$ -by- $M$  orthogonal matrix  $\mathbf{G}_m$  can be decomposed into the following product sequence of Givens rotation matrices,

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

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

$$\mathbf{G}_m(\boldsymbol{\theta}) = \left\{ \mathbf{S}_{23}(\theta_1) \right\} \left\{ \mathbf{S}_{13}(\theta_2) \mathbf{S}_{12}(\theta_3) \right\} \left\{ \mathbf{S}_{03}(\theta_4) \mathbf{S}_{02}(\theta_5) \mathbf{S}_{01}(\theta_6) \right\}$$

where  $\boldsymbol{\theta} = [\theta_1, \dots, \theta_6]^T$ .

### 4. FIR PU APPROXIMATION TO PCFB

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

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

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

After substituting the decomposition for  $\mathbf{R}(\omega)$  into  $\eta$ , there are  $\frac{1}{2}(L+1)M(M-1)$  rotation angles  $\theta_k$  that are free parameters. To restrict the value of each  $\theta_k$  to lie between  $-2\pi$  and  $2\pi$ , 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,

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

for  $\alpha$  a positive scalar. One of the more popular techniques for solving unconstrained minimization problems is the Broyden, Fletcher, Goldfarb, and Shanno (BFGS) algorithm, which falls under the category of a quasi-Newton method [5]. The MATLAB function *fminunc* implements a version of the BFGS algorithm for unconstrained minimization.

### 5. FEATURES OF THE BFGS ALGORITHM

The BFGS algorithm uses gradient information and an approximation to the Hessian matrix to minimize an objective function  $f(\mathbf{x})$ . Let  $\mathbf{B}$  denote a symmetric matrix used to approximate the Hessian. Then given  $\mathbf{x}_K$  and  $\mathbf{B}_K$ , the execution of the BFGS algorithm at step  $K+1$  to minimize  $f(\mathbf{x})$  is as follows:

1. Solve the equation

$$\mathbf{B}_K \mathbf{d}_K = -\nabla f(\mathbf{x}_K)$$

for the search direction  $\mathbf{d}_K$ .

2. Perform a line search along  $\mathbf{d}_K$  to find the step size  $\lambda$ .

Then set

$$\mathbf{x}_{K+1} = \mathbf{x}_K + \lambda \mathbf{d}_K.$$

3. Check for convergence using

$$\|\mathbf{x}_K - \mathbf{x}_{K+1}\| \leq \varepsilon, \quad \|\nabla f(\mathbf{x}_{K+1})\| \leq \delta$$

where  $\varepsilon$  and  $\delta$  are small constants. If the algorithm has not converged, then update  $\mathbf{B}$  by

$$\mathbf{B}_{K+1} = \mathbf{B}_K + \frac{\mathbf{y}\mathbf{y}^T}{\mathbf{y}^T \mathbf{s}} - \frac{\mathbf{B}_K \mathbf{s} \mathbf{s}^T \mathbf{B}_K}{\mathbf{s}^T \mathbf{B}_K \mathbf{s}}$$

where

$$\mathbf{s} = \mathbf{x}_{K+1} - \mathbf{x}_K, \quad \mathbf{y} = \nabla f(\mathbf{x}_{K+1}) - \nabla f(\mathbf{x}_K).$$

## 6. USING DITHERS TO ESTIMATE GRADIENT

In this paper, a novel filter bank architecture is proposed for creating a time-varying PU filter bank that tracks changes in the input signal statistics and approximates the ideal PCFB at each time instant  $t_k$ . This architecture makes use of sinusoidal dithers as described in [6] to determine the value of the gradient vector in near real-time.

Consider any objective function  $f(\mathbf{x})$  where  $\mathbf{x}$  is an  $N$ -by-1 vector. For each component of  $\mathbf{x}$ , superimpose a sinusoidal dither of different frequency, as in

$$\mathbf{x}' = \mathbf{x} + \boldsymbol{\theta} = \mathbf{x} + \alpha [\cos(\omega_1 t), \cos(\omega_2 t), \dots, \cos(\omega_N t)]^T$$

where  $\alpha$  is a small scalar. Using the Taylor series expansion of  $f(\mathbf{x})$  yields,

$$\begin{aligned} f(\mathbf{x}') &= f(\mathbf{x} + \boldsymbol{\theta}) = f(\mathbf{x}) + \nabla f(\mathbf{x})^T \boldsymbol{\theta} + \frac{1}{2} \boldsymbol{\theta}^T \nabla^2 f(\mathbf{x}) \boldsymbol{\theta} + \dots \\ &= f(\mathbf{x}) + \alpha \sum_{i=1}^N \left. \frac{\partial f}{\partial x_i} \right|_{\mathbf{x}} \cos(\omega_i t) + \dots \end{aligned}$$

The components of the gradient vector can be determined exactly once we multiply  $f(\mathbf{x}')$  with  $\cos(\omega_j t)$ , for  $j = 1, 2, \dots, N$ . The result after using trigonometric identities is,

$$\begin{aligned} f(\mathbf{x}') \cos(\omega_j t) &= f(\mathbf{x}) \cos(\omega_j t) + \frac{\alpha}{2} \left. \frac{\partial f}{\partial x_j} \right|_{\mathbf{x}} \\ &+ \frac{\alpha}{2} \left. \frac{\partial f}{\partial x_j} \right|_{\mathbf{x}} \cos(2\omega_j t) + \frac{\alpha}{2} \sum_{i \neq j} \left. \frac{\partial f}{\partial x_i} \right|_{\mathbf{x}} \cos([\omega_i - \omega_j] t) \\ &+ \frac{\alpha}{2} \sum_{i \neq j} \left. \frac{\partial f}{\partial x_i} \right|_{\mathbf{x}} \cos([\omega_i + \omega_j] t) + H.O.T. \end{aligned}$$

Note that the only constant term on the right hand side is the  $j$ th component of the gradient vector scaled by the factor  $\alpha/2$ , and can therefore be recovered exactly by low-pass filtering  $f(\mathbf{x}') \cos(\omega_j t)$ . Once the entire gradient vector is reconstructed, it can be used in the BFGS algorithm to compute the next iteration.

## 7. SIGNAL TRACKING ARCHITECTURE

Fig. 2 illustrates a lattice implementation of the Givens factorization of a PU filter bank [7]. In this case,  $M = 3$  and the polyphase matrix has Smith-McMillan degree equal to 2. Each Givens rotation is a simple butterfly connection between channels  $i$  and  $j$ . This architecture can be used to create a continuously varying PU approximation to the ideal PCFB of a nonstationary process by adding a sinusoidal dither to each rotation angle and defining the objective function in terms of the subband energy in each channel.

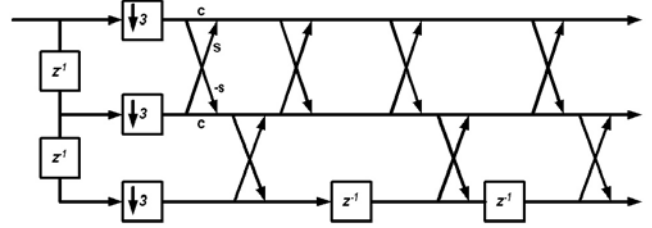


Fig. 2. Lattice Implementation of PU Filter Bank

For example, suppose initially that at time  $t_0$ , a FIR PU filter bank is available that minimizes the mean squared error between it and an ideal PCFB. This filter bank would be designed using the approach outlined previously in Section 4. Furthermore, suppose that the statistics of a zero-mean random process at the input to the filter bank are changing and that we wish to continuously update the first channel filter to track changes in the input signal and to maintain maximum energy in the first channel, just as an ideal PCFB would. Then the FIR PU filter bank could be updated at time  $t_1$  using the following approach,

1. Apply a sinusoidal dither of frequency  $\omega_j$  to each rotation angle  $\theta_j$ . The dithers can be applied in parallel and simultaneously to all the rotation angles, or they can be applied sequentially, as in a greedy algorithm. With the discrete-time sinusoidal dithers applied, the Givens rotations in Fig. 2 become,

$$\begin{aligned} c &= \cos(\theta_j + \cos(\omega_j n \Delta T)) \\ s &= \sin(\theta_j + \cos(\omega_j n \Delta T)) \end{aligned}$$

where  $\Delta T$  is the sampling period.

2. Estimate the energy of the subband vector process using the expression

$$\boldsymbol{\sigma}_w^2 = \text{diag} \left( \frac{1}{2\pi} \int_0^{2\pi} \mathbf{E}(\omega)^H \mathbf{S}_{xx}(\omega) \mathbf{E}(\omega) d\omega \right),$$

which is equivalent to obtaining the diagonal entries of the autocorrelation matrix  $\mathbf{R}_{xx}$  at zero lag. Equivalently, the energy in each channel could be estimated individually using the time-averaged power, as in

$$\hat{\sigma}_{w_k}^2 = \frac{1}{N} \sum_{n=0}^{N-1} |w_k(n)|^2,$$

where  $k$  corresponds to the  $k$ th channel. If  $N$  is large enough, then

$$\hat{\sigma}_{w_k}^2 \approx \sigma_{w_k}^2 = E[|w_k(n)|^2].$$

3. Define the objective function to be,

$$\max J = \boldsymbol{\sigma}_{w_1}^2 = \mathbf{e}_1^T \boldsymbol{\sigma}_w^2$$

where  $\mathbf{e}_1$  is a vector of zeros except for a value of unity in the first component.

4. Mix the objective function with  $\cos(\omega_j t)$  and low-pass filter to obtain the  $j$ th component of the gradient vector.

5. Once the gradient vector is constructed, use the BFGS algorithm described in Section 5 to compute the set of rotation angles for time  $t_1$ .
6. Given the next input data vector  $x(2)$ , go to Step 1 to compute the rotation angles for time  $t_2$ .

### 8. SIMULATION RESULTS

To test the architecture described in Section 7, a MATLAB simulation was created using a real, auto-regressive input process with time varying pole locations. Table 1 lists the pole locations of the process at times  $t_0$  and  $t_1$ . Fig. 3 illustrates the PSD of the input process at time  $t_0$  and Fig. 4 illustrates the PSD at time  $t_1$ .

Table 1. Pole Locations for Time Varying AR Process

Time Instant	Pole Locations
$t_0$	$0.95e^{j2\pi(0.1)}, 0.8e^{j2\pi(0.45)}, 0.95e^{-j2\pi(0.1)}, 0.8e^{-j2\pi(0.45)}$
$t_1$	$0.8e^{j2\pi(0.1)}, 0.8e^{j2\pi(0.45)}, 0.8e^{-j2\pi(0.1)}, 0.8e^{-j2\pi(0.45)}$

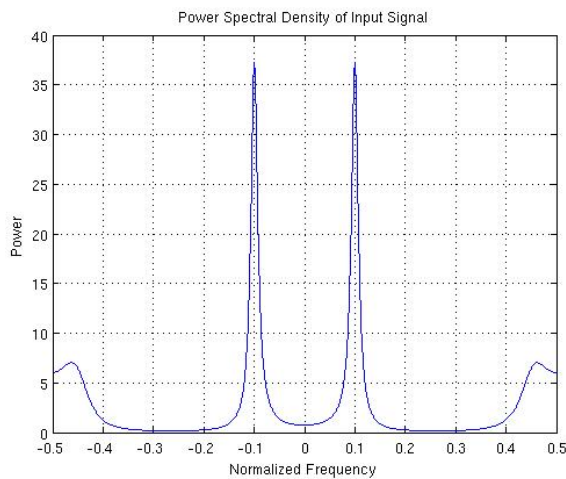


Fig. 3. Input PSD at Time  $t_0$

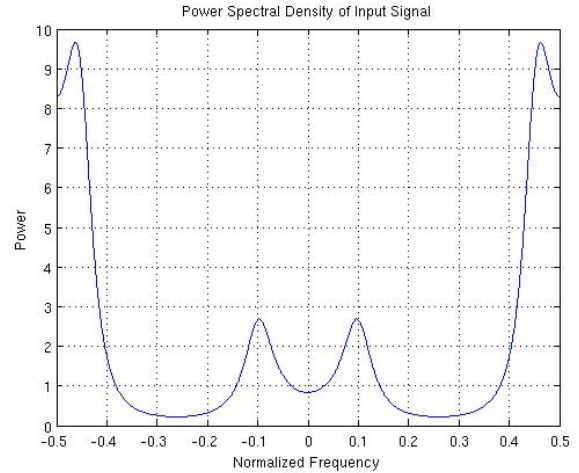


Fig. 4. Input PSD at Time  $t_1$

The design results shown in this section are for a greedy algorithm whereby the objective function  $J$  was optimized sequentially for each rotation angle in a PU filter bank with 4 channels. The dither frequency was chosen to be 100 Hz, the sampling frequency 800 Hz, and the scalar  $\alpha$  was set equal to 0.1. The low pass filter was implemented as a sliding window average of the data using a mask of 128 samples normalized to have unity DC gain.

Fig. 5 illustrates the first channel filter response from the initial PU filter bank approximation to the ideal PCFB at time  $t_0$ , which was designed to minimize the mean squared error. There are a total of 12 rotation angles in the Givens decomposition of the PU filter bank and the length of each channel filter is 8 samples. The brickwall response of each ideal PCFB channel filter is indicated with a dashed line.

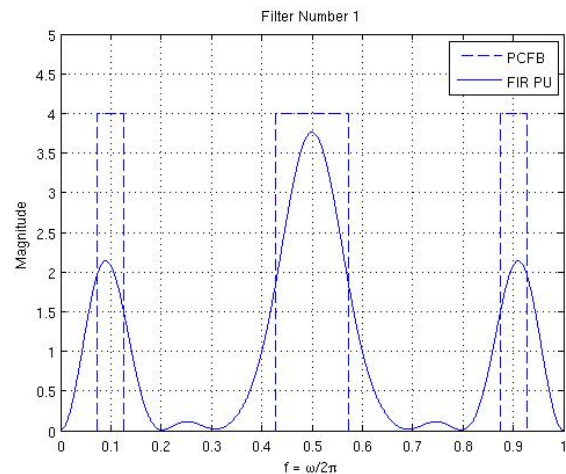


Fig. 5. Filter 1 From Initial PU Filter Bank at Time  $t_0$

Fig. 6 illustrates the response of the updated first channel filter designed using the dithering approach to maximize the subband energy in the first channel.

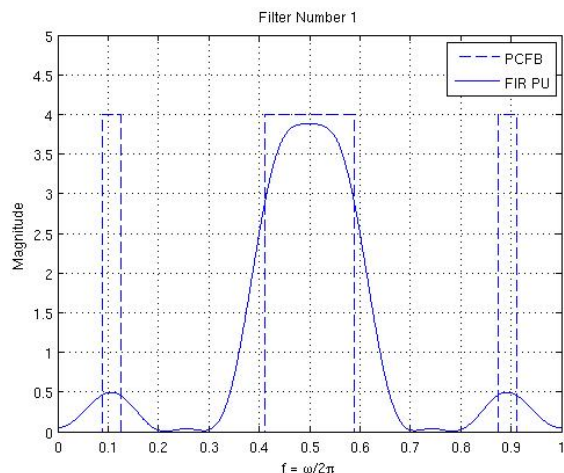


Fig. 6. Updated Filter 1 at Time  $t_1$

As is clear from Fig. 6 the frequency response of the filter has widened in the largest passband region and has adapted nicely to maintain maximum energy in the first subband, as desired. However, Fig. 7 shows that the frequency response of other channel filters deviates from the ideal.

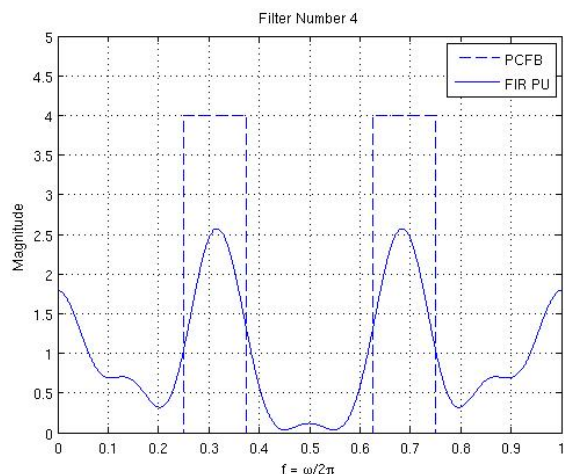


Fig. 7. Updated Filter 4 at Time  $t_1$

## 9. JOINT OPTIMIZATION OVER ALL ANGLES

Although the sequential greedy approach yields satisfactory results for tracking the first channel filter, which is the most important in a PCFB, a better solution can be obtained by optimizing the objective function over all rotation angles simultaneously. By carefully choosing the dither frequencies and the bandwidth of the low-pass filter, it is possible to apply the dithers in parallel to estimate all components of the gradient vector simultaneously. Specifically, the primary concern with applying all the dithers in parallel is to avoid crosstalk between the dithering signals that could corrupt the estimate of the gradient. If the

dither frequencies  $\omega_j$  and the differences between the dither frequencies,  $\omega_j - \omega_i$ ,  $i \neq j$ , are larger than the bandwidth of the low-pass filter, then crosstalk will be negligible. Fig. 8 illustrates the placement of 12 dither frequencies applied in parallel and the bandwidth of the low-pass filter used in conjunction with the dithers to jointly optimize the objective function  $J$  over all rotation angles. The sampling frequency for this configuration was 3200 Hz. The length of the impulse response of the low-pass filter was 128 samples and the dithers were spaced 100 Hz apart with the first dither frequency at 100 Hz. The scalar  $\alpha$  was set equal to 0.01.

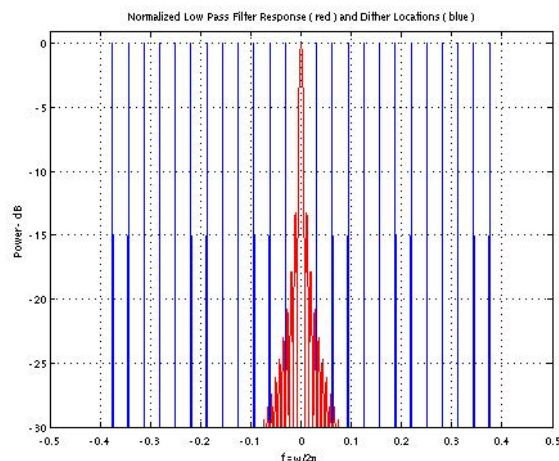


Fig. 8. Parallel Dithering Configuration

Compared to the sequential greedy algorithm, the advantage of applying the dither signals in parallel, and jointly optimizing over all rotation angles simultaneously, is that the BFGS algorithm will converge to a filter bank solution that is an excellent approximation to the ideal brickwall filter responses not only in the desired first channel filter, but also in the other channel filters as well. One disadvantage of a parallel dithering configuration compared to a sequential approach, however, is that a higher sampling rate is necessary. Figures 9 and 10 illustrate two of the FIR filters designed using the parallel dithering configuration.

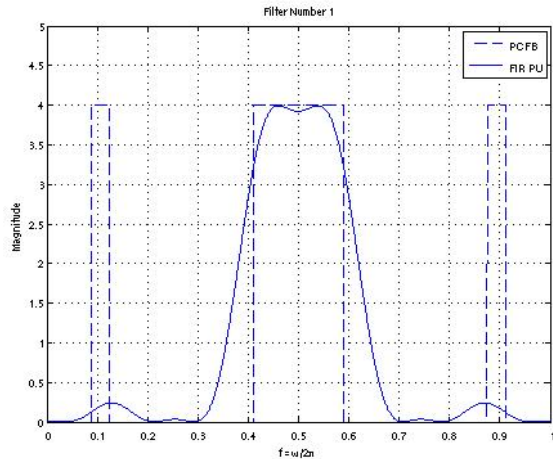


Fig. 9. Filter 1 at Time  $t_1$  Designed Using Parallel Dithers

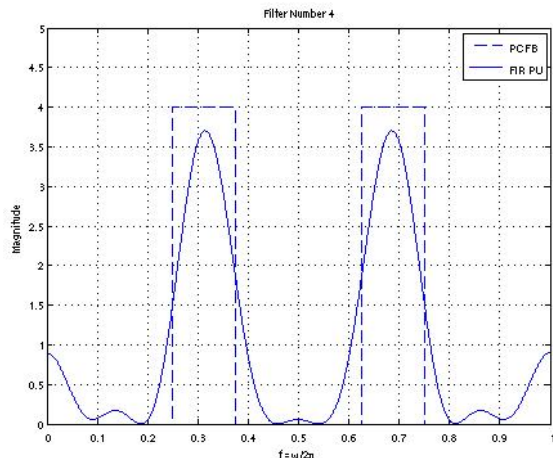


Fig. 10. Filter 4 at Time  $t_1$  Designed Using Parallel Dithers

## 10. CONCLUSIONS AND FUTURE PLANS

The use of dithers to estimate the gradient of a cost function in near real-time appears to be a very promising technique for designing signal adapted filter banks that can track changes in the input signal. The dithers can be applied sequentially as in a greedy algorithm or they can be applied simultaneously in parallel. A parallel configuration requires careful attention to the bandwidth of the low pass filter and the placement of the dither frequencies to avoid crosstalk. Although both the greedy algorithm and the simultaneous optimization of the objective function over all rotation angles yield excellent solutions for tracking the first channel filter, the advantage of employing the joint optimization approach is that the other channel filters are also close approximations to the ideal.

Future plans include verifying the performance of this technique on a high performance computing architecture, such as the Cray XD1, where the dithering operation can be

performed on FPGAs and the BFGS routine on a general purpose processor for maximum computational throughput.

## 11. REFERENCES

- [1] P. P. Vaidyanathan, *Multirate Systems and Filter Banks*, Prentice Hall, Upper Saddle River, NJ, 1993.
- [2] M. K. Tsatsanis and G. B. Giannakis, "Principal Component Filter Banks for Optimal Multiresolution Analysis," *IEEE Trans. Signal Processing*, pp. 1766 - 1777, Aug. 1995.
- [3] 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.
- [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] D. P. Bertsekas, *Nonlinear Programming*, Athena Scientific, Belmont, MA, 1999.
- [6] D. N. Loizos, P. P. Sotiriadis, and G. Cauwenberghs, "A Robust Continuous-Time Multi-Dithering Technique for Laser Communications Using Adaptive Optics," *Proceedings IEEE International Symposium on Circuits and Systems*, Kos, Greece, May 2006.
- [7] G. Strang and T. Nguyen, *Wavelets and Filter Banks*, Wellesley-Cambridge Press, Wellesley, MA, 1997.