Iterative Greedy Algorithm for Solving the FIR Paraunitary Approximation Problem

Andre Tkacenko, *Member, IEEE,* and P. P. Vaidyanathan, *Fellow, IEEE*

***Abstract*—In this paper, a method for approximating a multi-input multi-output (MIMO) transfer function by a causal finite-impulse response (FIR) paraunitary (PU) system in a weighted least-squares sense is presented. Using a complete parameterization of FIR PU systems in terms of Householder-like building blocks, an iterative algorithm is proposed that is greedy in the sense that the observed mean-squared error at each iteration is guaranteed to not increase. For certain design problems in which there is a phase-type ambiguity in the desired response, which is formally defined in the paper, a phase feedback modification is proposed in which the phase of the FIR approximant is fed back to the desired response. With this modification in effect, it is shown that the resulting iterative algorithm not only still remains greedy, but also offers a better magnitude-type fit to the desired response. Simulation results show the usefulness and versatility of the proposed algorithm with respect to the design of principal component filter bank (PCFB)-like filter banks and the FIR PU interpolation problem. Concerning the PCFB design problem, it is shown that as the McMillan degree of the FIR PU approximant increases, the resulting filter bank behaves more and more like the infinite-order PCFB, consistent with intuition. In particular, this PCFB-like behavior is shown in terms of filter response shape, multiresolution, coding gain, noise reduction with zeroth-order Wiener filtering in the subbands, and power minimization for discrete multitone (DMT)-type transmultiplexers.**

***Index Terms*—Filter bank optimization, greedy algorithm, interpolation, principal components filter bank.**

I. Introduction

THE problem of approximating in the least-squares sense a desired response, say $D(e^{j\omega})$, by a causal finite-impulse response (FIR) filter $F(e^{j\omega})$ of length N was first considered by Tufts and Francis in 1970 [19]. In essence, the goal is to minimize a possibly weighted mean-squared error between the desired and FIR filter responses given by the following:

$$\xi \triangleq \frac{1}{2\pi}\int_{0}^{2\pi} W(\omega)\left|D(e^{j\omega}) - F(e^{j\omega})\right|^2 d\omega. \qquad (1)$$

Here, $W(\omega)$ is a nonnegative weight function that is used to emphasize the design of certain frequency ranges of interest over others. For example, $D(e^{j\omega})$ may be the response of an ideal lowpass filter that we may want to approximate over certain regions with an FIR filter $F(e^{j\omega})$. Using the trick of completing the square [4], it can be shown that the filter coefficients of $F(e^{j\omega})$, which minimize ξ from (1), can be obtained in closed form after calculating an appropriate matrix inverse [19]. Due to the completely arbitrary nature of the desired response $D(e^{j\omega})$, the least-squares method for FIR filter design can be applied to a myriad of design problems. The method can even easily be generalized to the multiple-input multiple-output (MIMO) case in which the desired and FIR filter responses are, in general, both matrices.

In many applications, we may require further constraints on the approximant, in addition to the inherent FIR assumption. If these additional constraints are linear, for example, then it turns out that the least-squares approach can be easily modified to accommodate these conditions [14]. In general, however, it may be difficult or even impossible to solve the least-squares problem with the constraints in effect.

One constraint that has received much attention from the signal processing community on account of its various applications in data compression and digital communications has been the *paraunitary* (PU) or *orthonormal* constraint [23]. This condition frequently arises in the design of multirate filter banks. One such example is the M-channel maximally decimated filter bank shown in Fig. 1(a). Here, the input signal $x(n)$ may represent a speech signal on which we would like to perform lossy data compression. For this example, the subband processors $\{\mathcal{P}_k\}$ would typically be scalar quantizers operating at a lower bit rate than the original input signal [13]. In order to introduce the PU constraint on the filter bank, we must first represent the filter bank in polyphase form [23]. If we consider the following M-fold polyphase decompositions [23] of the analysis filters $\{H_k(z)\}$ and synthesis filters $\{F_k(z)\}$

$$H_k(z) = \sum_{\ell=0}^{M-1} z^{\ell} H_{k,\ell}(z^M) \quad \text{(Type II)}$$

$$F_k(z) = \sum_{\ell=0}^{M-1} z^{-\ell} F_{k,\ell}(z^M) \quad \text{(Type I)}$$

then the system of Fig. 1(a) can be redrawn as in Fig. 1(b), where we have

$$[\mathbf{H}(z)]_{k,\ell} = H_{k,\ell}(z),\ [\mathbf{F}(z)]_{k,\ell} = F_{\ell,k}(z),\ 0 \le k,\ell \le M-1$$

The filter bank is then said to be a perfect reconstruction (PR) PU or orthonormal filter bank iff we have

$$\widetilde{\mathbf{F}}(z)\mathbf{F}(z) = \mathbf{I}, \quad \mathbf{H}(z) = \widetilde{\mathbf{F}}(z) \quad \forall z. \qquad (2)$$

where $\widetilde{\mathbf{A}}(z) \triangleq \mathbf{A}^{\dagger}(1/z^*)$ for any $p \times r$ transfer function $\mathbf{A}(z)$ [23]. Here, the first part of (2) is the PU condition on $\mathbf{F}(z)$, and

Manuscript received February 21, 2004; revised January 25, 2005. This work was supported in part by the NSF grant CCF-0428326, ONR grant N00014-06-1-0011, and the California Institute of Technology. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Henrique Malvar.

A. Tkacenko was with the Department of Electrical Engineering, California Institute of Technology, Pasadena, CA 91125 USA. He is now with the Digital Signal Processing Research Group, Jet Propulsion Laboratory, Pasadena, CA 91109 USA (e-mail: andre@systems.caltech.edu).

P. P. Vaidyanathan is with the Department of Electrical Engineering, California Institute of Technology, Pasadena, CA 91125 USA (e-mail: ppvnath@systems.caltech.edu).

Digital Object Identifier 10.1109/TSP.2005.861054

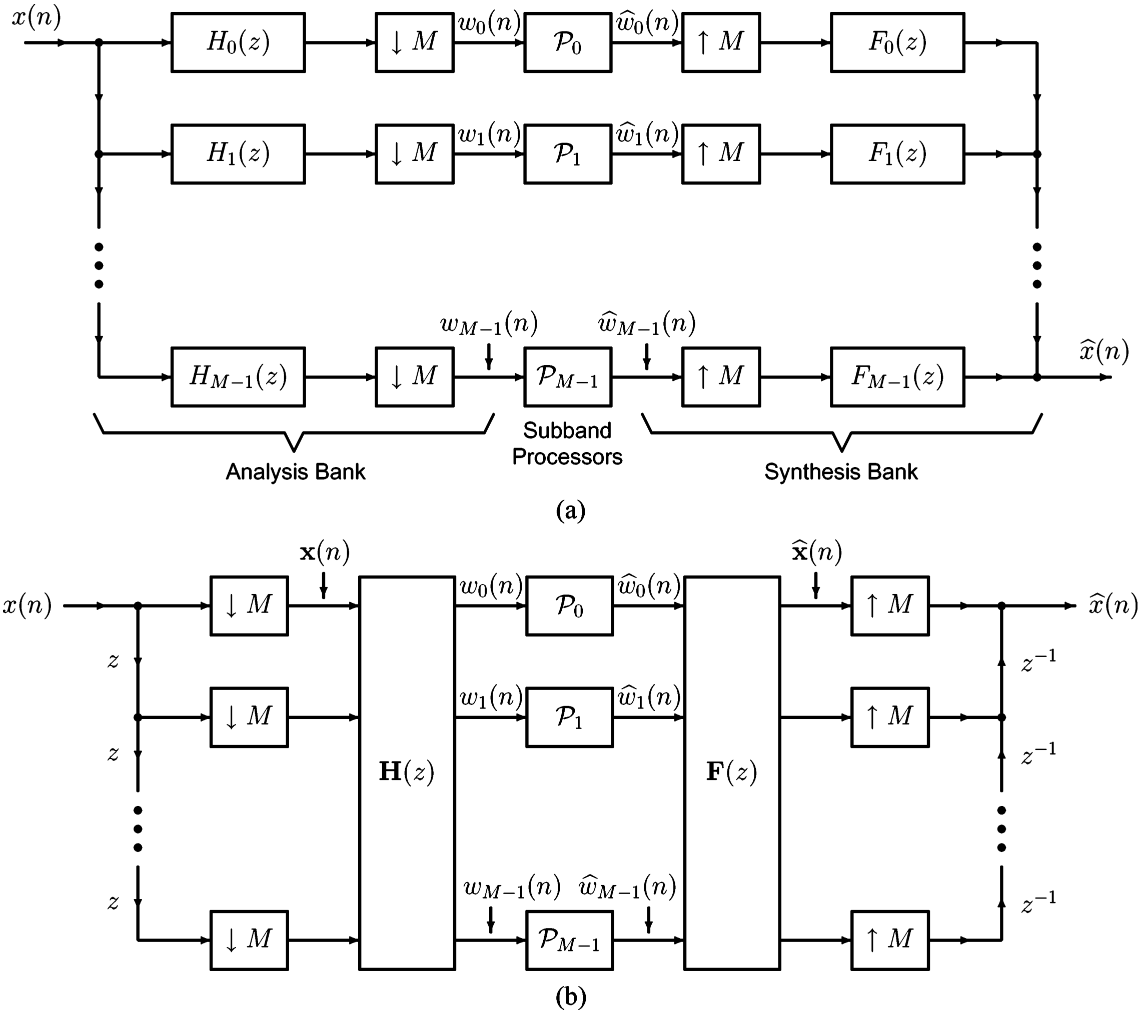


Fig. 1. (a) Typical maximally decimated filter bank system. (b) Polyphase representation of filter bank.

the second part ensures that we have PR (in the absence of the subband processors $\{\mathcal{P}_k\}$). Orthonormal filter banks have many interesting properties, which have made them very popular for use in numerous applications. For example, a PU filter bank is *lossless*, meaning that the energy observed in the subband signals $\{w_k(n)\}$ in Fig. 1(b) is precisely equal to that of the input signal $x(n)$ [23]. Furthermore, if the synthesis polyphase matrix $\mathbf{F}(z)$ is FIR, then the corresponding analysis polyphase matrix $\mathbf{H}(z) = \widetilde{\mathbf{F}}(z)$ is also necessarily FIR as well. Orthonormal filter banks have been used to generate wavelet bases [23] and have even been used for wavelet-based data compression such as that used by JPEG 2000 [13].

In this paper, we consider the weighted least-squares FIR filter design problem for the general MIMO case with the PU constraint in effect. Since the PU constraint from (2) imposes a *quadratic* constraint on the filter coefficients, there is no closed-form expression for the optimal FIR approximant. However, by using a complete parameterization of all FIR PU systems in terms of Householder-like degree-one building blocks [28], it will be shown that optimizing one set of parameters can be done in closed form, assuming all other parameters are fixed. This will lead to an iterative algorithm in which a different set of parameters is optimized at each iteration. Since a given set of parameters is optimized at each iteration, the observed mean-squared error is *guaranteed* to not increase as a function of iteration. As such, the algorithm is *greedy*, since we optimize one set of parameters while ignoring the rest.

In cases where the MIMO desired response has what we shall refer to as a *phase-type* ambiguity, which will be discussed in Section III, we propose a *phase feedback modification* to the desired response. With this modification, the "phase" of the FIR approximant is fed back to the desired response. Using this modification, it can be shown that the iterative algorithm not only still remains greedy, but also offers a better *magnitude-type* fit to the desired response. Simulation results provided here show the merit of the phase feedback modification.

Due to the arbitrary nature of the weighting function and desired response, the same proposed algorithm can be used to solve a variety of problems. In particular, by appropriately choosing the weight function $W(\omega)$, we can apply the iterative algorithm to the FIR PU interpolation problem discussed in Section I-A-2). As opposed to the traditional FIR interpolation problem, which has been well studied and can be solved easily [4], the FIR PU interpolation problem is far more difficult and has not yet been solved [26]. The iterative algorithm proposed here can be used to obtain valuable insight into the FIR PU interpolation problem, as simulation results in Section IV-B show.

Prior to analyzing the FIR PU approximation problem, we first introduce two applications in which this problem arises.

A. Motivation

1) Principal Component Filter Banks: Consider again the filter bank system of Fig. 1(b) in which the filter bank satisfies the PU condition of (2). Suppose that the blocked input signal vector $\mathbf{x}(n)$ is wide sense stationary (WSS) with power spectral density (psd) $\mathbf{S}_{\mathbf{xx}}(z)$. This is tantamount to saying that the scalar input signal $x(n)$ is cyclo-WSS with period M (abbreviated CWSS(M)) [12]. Recently, it has been shown that a special type of PU filter bank matched to the input statistics $\mathbf{S}_{\mathbf{xx}}(z)$ known as the principal component filter bank (PCFB) [18] is *simultaneously* optimal for a variety of objective functions [1]. Among these objectives are included several important data compression objectives such as mean-squared error under the presence of quantization noise [7] (for any bit allocation) and coding gain [24], [25] (with optimal bit allocation). By definition, a PCFB for an input psd $\mathbf{S}_{\mathbf{xx}}(z)$ and for a class $\mathcal{C}$ of filter banks, if it exists, is one whose subband variance vector

$$\boldsymbol{\sigma} \triangleq \begin{bmatrix} \sigma_{w_0}^2 & \sigma_{w_0}^2 & \cdots & \sigma_{w_{M-1}}^2 \end{bmatrix}^T \tag{3}$$

majorizes [4] any other subband variance vector arising from any other filter bank from $\mathcal{C}$. (Recall that a vector $\mathbf{a} \triangleq [a_0\ a_1\ \cdots\ a_{P-1}]^T$ with $a_0 \geq a_1 \geq \cdots \geq a_{P-1} \geq 0$ is said to *majorize* [4] a vector $\mathbf{b} \triangleq [b_0\ b_1\ \cdots\ b_{P-1}]^T$ with $b_0 \geq b_1 \geq\ \cdots\ \geq b_{P-1} \geq 0$ iff we have

$$\sum_{k=0}^{p} a_k \geq \sum_{k=0}^{p} b_k\ \forall\ 0 \leq p \leq P-2,\ \sum_{k=0}^{P-1} a_k = \sum_{k=0}^{P-1} b_k.)$$

In addition to being optimal for coding gain and mean-squared error in the presence of quantization noise, the PCFB has also been shown to be optimal for any *concave* objective function of $\boldsymbol{\sigma}$ [1].

The only problem is that for general input power spectra, PCFBs only exist for special classes of filter banks. One notable exception to this is for the special case where $M = 2$, in which case a PCFB always exists for any class of PU filter banks [1]. For general M, however, PCFBs are known to exist only for two special classes. If $\mathcal{C}$ is the class of all transform coders $\mathcal{C}^t$, in which $\mathbf{F}(z)$ is a constant unitary matrix $\mathbf{T}$, then the PCFB exists and is the Karhunen–Loève transform (KLT) for the input process $\mathbf{x}(n)$ (i.e., $\mathbf{T}$ *diagonalizes* the autocorrelation matrix $\mathbf{R}_{\mathbf{xx}}(0)$) [1], [5]. Furthermore, if $\mathcal{C}$ is the class of all (unconstrained order) PU filter banks $\mathcal{C}^u$, then the PCFB exists and is the *pointwise in frequency* KLT for $\mathbf{x}(n)$ [1], [24], [25]. By this, we mean that $\mathbf{F}(e^{j\omega})$ diagonalizes (i.e., totally decorrelates) $\mathbf{S}_{\mathbf{xx}}(e^{j\omega})$ for every ω such that the frequency-dependent eigenvalues are always arranged in decreasing order, which is a property called spectral majorization [24]. For many practical cases of inputs (for example, if the scalar input signal $x(n)$ is itself WSS), the corresponding analysis and synthesis filters are ideal bandpass filters called *compaction filters* [21], [22], [24]. As such, they are unrealizable in practice and serve only to compute an upper bound on the performance that we can expect from a PU filter bank.

The problem with the class of FIR PU filter banks in which $\mathbf{F}(z)$ has finite memory (or more appropriately finite *McMillan degree* [23]) is that it is believed that a PCFB does not exist [1], [6], [8], although this has not yet been formally proven. Instead, for this class, $\mathbf{F}(z)$ is typically chosen to optimize a *specific* objective for a given input psd, such as coding gain [2], [3], [9], [29], rate distortion [10], or a multiresolution energy compaction criterion [11]. All such methods require the numerical optimization of nonlinear and nonconvex objective functions, which offer little insight into the behavior of the solutions as the filter order (i.e., the memory of $\mathbf{F}(z)$) increases. Another common approach is to calculate an optimal FIR compaction filter [15], [20] (for the first filter $F_0(z)$) and then obtain the rest of the filters via an appropriate filter bank completion for a multiresolution criterion [11], [16]. Although this approach is elegant in the sense that the filter bank design problem is tantamount to calculating an FIR compaction filter followed by an appropriate KLT, it suffers from the ambiguity caused by the nonuniqueness of the FIR compaction filter. Different compaction filter spectral factors lead to different filter banks, which in turn yield different performances. As such, all such spectral factors need to be tested for their performance [16], which is *exponentially* computationally complex with respect to the order of the compaction filter.

In this paper, the approach that will be taken to obtain a suitable signal-adapted FIR PU filter bank will be to find the one that best approximates the *unconstrained* or *infinite order* PCFB solution in the mean-squared sense. Intuitively, we should expect that as the McMillan degree of the FIR PU system increases, the filter banks designed become more and more like the infinite-order PCFB. This will indeed be seen through simulations in Section IV-A in terms of objectives such as filter response shape, multiresolution, and coding gain. Along with these data-compression-type objectives, this PCFB-like behavior is also shown for noise reduction with zeroth-order Wiener filtering in the subbands, and power minimization for discrete multitone (DMT)-type transmultiplexers. In contrast with the methods of [11] and [16], all of the synthesis filters with this method are computed *simultaneously*, avoiding the need to compare the performance of different spectral factors of a given FIR compaction filter [16].

It should be noted that the infinite-order PCFB has a *phase-type* ambiguity or nonuniqueness (see Section III). As such, using the proposed iterative algorithm, it is not clear which infinite-order PCFB desired response will yield the overall best FIR PU approximant. To alleviate this dilemma, the phase of the desired response is mixed with that of the FIR approximant, a process that we refer to here as *phase feedback*. This modification allows the iterative algorithm to find a better FIR PU approximant to an infinite-order PCFB than without it, as will be shown in the simulation results in Section IV-A.

2) The FIR PU Interpolation Problem: In certain applications, it may be necessary for an FIR PU system, say $\mathbf{F}(e^{j\omega})$, to take on a prescribed set of values over a prescribed set of frequencies. For example, suppose that for the frequencies $\omega_0, \omega_1, \ldots, \omega_{L-1}$, we require

$$\mathbf{F}(e^{j\omega_k}) = \mathcal{U}_k \quad \forall\ 0 \leq k \leq L-1. \tag{4}$$

Evidently, the matrices $\{\mathcal{U}_k\}$ must be unitary in light of the PU assumption on $\mathbf{F}(e^{j\omega})$. The problem of finding an FIR PU system of a certain degree that satisfies (4) is known as the FIR PU interpolation problem [26].

In the traditional FIR interpolation problem, in which the only restriction made on the interpolant is the FIR constraint, we can always find an interpolant of length at most equal to the number of interpolation conditions by using the Lagrange interpolation formula [4]. However, for the FIR PU interpolation problem of (4), in general, it is not known whether there even exists an interpolant of finite degree that will satisfy all L conditions from (4). For the special case in which $\mathbf{F}(e^{j\omega})$ is scalar, it is known that in general, only one condition from (4) can be satisfied (since in this case, $\mathbf{F}(z)$ is necessarily a pure delay [26]).

Although there is no known solution to the FIR PU interpolation problem, the proposed iterative algorithm can offer valuable insight into the problem. Through proper choice of the weight function, the iterative algorithm can be used to find an FIR PU system of a particular degree that best approximates the interpolation conditions of (4) in a weighted least-squares sense. By observing the behavior of the mean-squared error at each iteration, we can conjecture whether or not an interpolant exists for the given interpolation conditions and degree. If the error tends to zero as the number of iterations increases, we can claim that such an interpolant indeed exists by construction. Simulation results for the FIR PU interpolation problem given in Section IV-B show the merit of the proposed iterative algorithm for this problem.

B. Outline of Paper

In Section II, we analyze the FIR PU approximation problem. Using the Householder-like parameterization of FIR PU systems given in [28], we show how to obtain the optimal parameters in Sections II-A and II-B. The iterative greedy algorithm for obtaining the FIR PU approximant is formally introduced in Section II-C. In Section III, we introduce the phase feedback modification to the iterative algorithm for cases in which the desired response has a phase-type ambiguity. Simulation results for the design of infinite-order PCFB-like FIR PU filter banks and for the FIR PU interpolation problem are presented in Sections IV-A and IV-B, respectively. Finally, concluding remarks are made in Section V.

II. The FIR PU Approximation Problem

Let $\mathbf{D}(e^{j\omega})$ be any $p\times r$ desired response matrix that we wish to approximate with a $p\times r$ causal FIR PU system $\mathbf{F}(e^{j\omega})$ of McMillan degree $(N-1)$. Note that we require $p\geq r$ in order to satisfy the PU condition $\widetilde{\mathbf{F}}(z)\mathbf{F}(z)=\mathbf{I}_r$. Here, we opt to choose $\mathbf{F}(e^{j\omega})$ to minimize a weighted mean-squared Frobenius norm error between $\mathbf{D}(e^{j\omega})$ and $\mathbf{F}(e^{j\omega})$ given by

$$\xi \triangleq \frac{1}{2\pi}\int_{0}^{2\pi} W(\omega)\left\|\mathbf{D}(e^{j\omega})-\mathbf{F}(e^{j\omega})\right\|_F^2 d\omega. \tag{5}$$

Here, $W(\omega)$ is a scalar nonnegative weight function as in (1), and $\|\mathbf{A}\|_F$ denotes the *Frobenius norm* of any matrix $\mathbf{A}$ given by $\|\mathbf{A}\|_F=\sqrt{\mathrm{Tr}[\mathbf{A}^\dagger\mathbf{A}]}$ [4].

Expanding (5) and using the PU condition $\widetilde{\mathbf{F}}(z)\mathbf{F}(z)=\mathbf{I}_r$ on $\mathbf{F}(z)$ yields the following:

$$\begin{aligned}\xi =& \underbrace{\frac{1}{2\pi}\int_0^{2\pi} W(\omega)\left\|\mathbf{D}(e^{j\omega})\right\|_F^2 d\omega+\frac{r}{2\pi}\int_0^{2\pi} W(\omega)d\omega}_{a}\\ &-\frac{1}{2\pi}\int_0^{2\pi} W(\omega)\mathrm{Tr}\left[\mathbf{D}^\dagger(e^{j\omega})\mathbf{F}(e^{j\omega})+\mathbf{F}^\dagger(e^{j\omega})\mathbf{D}(e^{j\omega})\right]d\omega.\end{aligned} \tag{6}$$

Note that the quantity a in (6) is simply a constant and that the only quantity that depends on the system $\mathbf{F}(z)$ is the last term of (6). Hence, with the PU constraint in effect, the error ξ is *linear* in $\mathbf{F}(z)$. This will greatly simplify the optimization problem, as will soon be shown.

To help solve this optimization problem with the PU constraint on $\mathbf{F}(z)$, we exploit the complete parameterization of causal FIR PU systems in terms of Householder-like degree-one building blocks [23], [28]. In particular, $\mathbf{F}(z)$ is a causal FIR PU system of McMillan degree $(N-1)$ iff it is of the form

$$\mathbf{F}(z)=\mathbf{V}(z)\mathbf{U} \tag{7}$$

where $\mathbf{V}(z)$ is a $p\times p$ PU matrix consisting of $(N-1)$ degree-one Householder-like building blocks of the form

$$\begin{aligned}\mathbf{V}(z) &= \prod_{i=N-1}^{1}\mathbf{V}_i(z)\\ \mathbf{V}_i(z) &= \mathbf{I}_p-\mathbf{v}_i\mathbf{v}_i^\dagger+z^{-1}\mathbf{v}_i\mathbf{v}_i^\dagger, \qquad 1\leq i\leq N-1\end{aligned} \tag{8}$$

where the vectors $\mathbf{v}_i$ are unit norm vectors, i.e., $\mathbf{v}_i^\dagger\mathbf{v}_i=1$ for all i. In addition, the matrix $\mathbf{U}$ is some $p\times r$ unitary matrix, i.e., $\mathbf{U}^\dagger\mathbf{U}=\mathbf{I}_r$.

Although it is difficult to jointly optimize the parameters $\mathbf{U}$ and $\{\mathbf{v}_k\}$ which minimize ξ from (6), it will be shown that optimizing each parameter separately while holding all other parameters fixed is very simple. This will lead to the proposed iterative algorithm whereby the parameters are individually optimized at each iteration.

A. Optimal Choice of $\mathbf{U}$

Substituting (7) into (6) yields the following:

$$\begin{aligned}\xi =& a-\frac{1}{2\pi}\int_0^{2\pi} W(\omega)\mathrm{Tr}\left[\mathbf{D}^\dagger(e^{j\omega})\mathbf{V}(e^{j\omega})\mathbf{U}\right]d\omega\\ &-\frac{1}{2\pi}\int_0^{2\pi} W(\omega)\mathrm{Tr}\left[\mathbf{U}^\dagger\mathbf{V}^\dagger(e^{j\omega})\mathbf{D}(e^{j\omega})\right]d\omega\\ =& a-\mathrm{Tr}\left[\underbrace{\left(\frac{1}{2\pi}\int_0^{2\pi} W(\omega)\mathbf{D}^\dagger(e^{j\omega})\mathbf{V}(e^{j\omega})d\omega\right)}_{\mathbf{A}^\dagger}\mathbf{U}\right]\\ &-\mathrm{Tr}\left[\mathbf{U}^\dagger\underbrace{\left(\frac{1}{2\pi}\int_0^{2\pi} W(\omega)\mathbf{V}^\dagger(e^{j\omega})\mathbf{D}(e^{j\omega})d\omega\right)}_{\mathbf{A}}\right]\end{aligned} \tag{9}$$

$$= a-2\underbrace{\mathrm{Re}\left[\mathrm{Tr}[\mathbf{U}^\dagger\mathbf{A}]\right]}_{\mu}. \tag{10}$$

Note that minimizing ξ from (10) is equivalent to maximizing μ. To find the optimal $p \times r$ unitary matrix $\mathbf{U}$ that maximizes μ, we must exploit the singular value decomposition (SVD) [4] of $\mathbf{A}$. Suppose that $\mathbf{A}$ has the following SVD:

$$\mathbf{A} = \mathbf{T}\mathbf{\Sigma}\mathbf{W}^\dagger \tag{11}$$

Here, $\mathbf{T}$ and $\mathbf{W}$ are, respectively, $p \times p$ and $r \times r$ unitary matrices. The quantity $\mathbf{\Sigma}$ is a $p \times r$ diagonal matrix of the form

$$\mathbf{\Sigma} = \begin{bmatrix} \mathbf{\Sigma}_0 & \mathbf{0}_{\rho \times (r-\rho)} \\ \mathbf{0}_{(p-\rho)\times\rho} & \mathbf{0}_{(p-\rho)\times(r-\rho)} \end{bmatrix} \tag{12}$$

where $\rho = \text{rank}(\mathbf{A})$ and $\mathbf{\Sigma}_0$ is a diagonal matrix of the singular values of $\mathbf{A}$. In other words, we have $\mathbf{\Sigma}_0 = \text{diag}(\sigma_0, \sigma_1, \ldots, \sigma_{\rho-1})$ where $\{\sigma_i\}$ are the singular values of $\mathbf{A}$ which satisfy $\sigma_i > 0$ for all $0 \le i \le \rho - 1$. Substituting (11) into (10) yields the following:

$$\mu = \text{Re}\left[\text{Tr}[\mathbf{U}^\dagger\mathbf{T}\mathbf{\Sigma}\mathbf{W}^\dagger]\right] = \text{Re}\left[\text{Tr}[\mathbf{\Sigma}\underbrace{\mathbf{W}^\dagger\mathbf{U}^\dagger\mathbf{T}}_{\mathbf{G}^\dagger}]\right]. \tag{13}$$

Note that the $p \times r$ matrix $\mathbf{G} = \mathbf{T}^\dagger\mathbf{U}\mathbf{W}$ is unitary, i.e., $\mathbf{G}^\dagger\mathbf{G} = \mathbf{I}_r$. Using (12) in (13) yields

$$\begin{aligned}\mu &= \text{Re}\left[\text{Tr}[\mathbf{\Sigma}\mathbf{G}^\dagger]\right] = \text{Re}\left[\sum_{i=0}^{\rho-1}\sigma_i[\mathbf{G}]^*_{i,i}\right] \\ &= \sum_{i=0}^{\rho-1}\sigma_i\text{Re}\left[[\mathbf{G}]^*_{i,i}\right] = \sum_{i=0}^{\rho-1}\sigma_i\text{Re}\left[[\mathbf{G}]_{i,i}\right].\end{aligned} \tag{14}$$

Since $\mathbf{G}$ is a unitary matrix, we have

$$\text{Re}\left[[\mathbf{G}]_{k,\ell}\right] \le 1 \tag{15}$$

with equality iff $[\mathbf{G}]_{k,q} = \delta(q-\ell)$ and $[\mathbf{G}]_{s,\ell} = \delta(s-k)$, as the columns of $\mathbf{G}$ form an orthonormal set of vectors [4]. In light of (15) and the fact that $\sigma_i > 0$ for all i, from (14), we have

$$\mu \le \sum_{i=0}^{\rho-1}\sigma_i \tag{16}$$

with equality iff $[\mathbf{G}]_{k,\ell} = \delta(k-\ell)$ for $0 \le k, \ell \le \rho - 1$. Since $\mathbf{G}$ is unitary, we have equality iff

$$\mathbf{G} = \mathbf{G}_{\text{opt}} = \begin{bmatrix} \mathbf{I}_\rho & \mathbf{0}_{\rho\times(r-\rho)} \\ \mathbf{0}_{(p-p)\times\rho} & \mathbf{G}_0 \end{bmatrix} \tag{17}$$

where $\mathbf{G}_0$ is an arbitrary $(p-\rho) \times (r-\rho)$ unitary matrix, i.e., $\mathbf{G}_0^\dagger\mathbf{G}_0 = \mathbf{I}_{(r-\rho)}$. As $\mathbf{G} = \mathbf{T}^\dagger\mathbf{U}\mathbf{W}$, we have $\mathbf{U} = \mathbf{T}\mathbf{G}\mathbf{W}^\dagger$, and so the optimum $\mathbf{U}$ and corresponding optimal value of ξ is given by (16) and (10) to be the following:

$$\begin{aligned}\mathbf{U}_{\text{opt}} &= \mathbf{T}\mathbf{G}_{\text{opt}}\mathbf{W}^\dagger \text{ with } \mathbf{G}_{\text{opt}} \text{ as in (17)} \\ \xi_{\text{opt}} &= a - 2\left(\sum_{i=0}^{\rho-1}\sigma_i\right).\end{aligned} \tag{18}$$

In the special case where $\rho = r$ (i.e., $\mathbf{A}$ as full rank), we have

$$\mathbf{U}_{\text{opt}} = \mathbf{T}\begin{bmatrix} \mathbf{W}^\dagger \\ \mathbf{0}_{(p-r)\times r} \end{bmatrix}, \quad \xi_{\text{opt}} = a - 2\left(\sum_{i=0}^{r-1}\sigma_i\right).$$

Since the matrices $\mathbf{T}$ and $\mathbf{W}$ from (18) depend on $\mathbf{V}(z)$, the choice of $\mathbf{U}$ from (18) is optimal for *fixed* $W(\omega)$, $\mathbf{D}(e^{j\omega})$, and $\mathbf{V}(z)$.

B. Optimal Choice of $\mathbf{v}_k$

In order to find the optimal choice of $\mathbf{v}_k$ assuming that all other parameters are fixed, we must cleverly extract only those portions of ξ that depend on $\mathbf{v}_k$. For simplicity, let us define the following $p \times p$ matrices:

$$\mathcal{L}_k(z) \triangleq \begin{cases} \prod_{i=N-1}^{k+1}\mathbf{V}_i(z), & 0 \le k \le N-2 \\ \mathbf{I}_p, & k = N-1 \end{cases} \tag{19}$$

$$\mathcal{R}_k(z) \triangleq \begin{cases} \mathbf{I}_p, & k = 1 \\ \prod_{i=k-1}^{1}\mathbf{V}_i(z), & 2 \le k \le N \end{cases}. \tag{20}$$

Note that $\mathcal{L}_k(z)$ and $\mathcal{R}_k(z)$ are, respectively, the left and right neighbors of the matrix $\mathbf{V}_k(z)$ for $1 \le k \le N-1$ appearing in $\mathbf{V}(z)$ from (8). In other words, we have

$$\mathbf{V}(z) = \mathcal{L}_k(z)\mathbf{V}_k(z)\mathcal{R}_k(z), \quad 1 \le k \le N-1. \tag{21}$$

Also note that by construction, we have $\mathcal{L}_0(z) = \mathcal{R}_N(z) = \mathbf{V}(z)$. Substituting (21) and (8) into (7) and (6) yields (22)–(24), shown at bottom of the next page. Here, the quantity c defined in (22) depends on all of the parameters *except* $\mathbf{v}_k$. Hence, to minimize ξ with respect to $\mathbf{v}_k$, we must minimize the quantity ν from (24). Note, however, that $\nu = \mathbf{v}_k^\dagger\mathbf{Q}\mathbf{v}_k$ is simply a *quadratic form* corresponding to the Hermitian matrix $\mathbf{Q}$ [4]. As $\mathbf{v}_k$ must satisfy $\mathbf{v}_k^\dagger\mathbf{v}_k = 1$, it follows from *Rayleigh's principle* [4] that the optimal $\mathbf{v}_k$ must be a unit norm eigenvector corresponding to the smallest eigenvalue of $\mathbf{Q}$. If $\lambda_{\min}$ denotes the smallest eigenvalue of $\mathbf{Q}$ and $\mathbf{w}_{\min}$ is any unit norm eigenvector corresponding to $\lambda_{\min}$, then the optimum choice of $\mathbf{v}_k$ and corresponding optimal ξ are given by (24) to be the following:

$$\mathbf{v}_{k,\text{opt}} = \mathbf{w}_{\min}, \quad \xi_{\text{opt}} = a - 2\text{Re}[c] + \lambda_{\min}. \tag{25}$$

Note that since $\mathbf{w}_{\min}$ from (25) depends on $\mathcal{L}_k(z)$, $\mathcal{R}_k(z)$, and $\mathbf{U}$, it follows that the choice of $\mathbf{v}_k$ from (25) is optimal for *fixed* $W(\omega)$, $\mathbf{D}(e^{j\omega})$, $\mathbf{U}$, and all $\mathbf{v}_i$ for which $i \ne k$.

In summary, finding the optimal parameters corresponding to the Householder-like factorization of causal FIR PU systems is simple if the parameters are optimized individually. The process of updating the individual parameters to their optimal values forms the basis of the proposed iterative algorithm for solving the FIR PU approximation problem, which we now present.

C. Iterative Greedy Algorithm for Solving the Approximation Problem

For the iterative algorithm presented below, each set of Householder-like parameters is optimized in a random order. Furthermore, a complete set of N parameters is optimized before moving on to a new set of parameters. This is explained mathematically below as follows.

Let ξ_m denote the mean-squared error at the mth iteration for $m \ge 0$. In addition, let $\Pi_\ell(k)$ denote a random permutation of the integers $0 \le k \le N-1$ for any $\ell \ge 0$. Then, the iterative algorithm for solving the FIR PU approximation problem is as follows.

```
Initialization:
    Generate a random $p \times r$ unitary matrix
    $\mathbf{U}$ and $(N-1)$ $p\times 1$ random unit norm vec-
    tors $\mathbf{v}_i$, $1 \leq i \leq N-1$.
Iteration: For $m \geq 0$, do the following.
1) If $\Pi_{\lfloor m/N \rfloor}(m \bmod N) = 0$, calculate the
    optimal unitary matrix $\mathbf{U}$ and corre-
    sponding $\xi_m$ using (18), (11), and (9).
      Otherwise, if $\Pi_{\lfloor m/N \rfloor}(m \bmod N) = k$ for
    some $1 \leq k \leq N-1$, calculate the optimal
    unit norm vector $\mathbf{v}_k$ and corresponding
    $\xi_m$ using (25), (24), (23), and (22).
2) Increment $m$ by 1 and return to
    Step 1).
```

Since at each stage in the iteration, we are globally optimizing one parameter while fixing the rest, the above technique is a *greedy algorithm*. As such, the mean-squared error ξ_m is guaranteed to be monotonic nonincreasing as a function of the iteration index m. Furthermore, as ξ_m has a lower bound (i.e., we always have $\xi_m \geq 0$), ξ_m is guaranteed to have a limit as $m \to \infty$ [23]. Thus, the algorithm is guaranteed to converge monotonically to a local optimum. Simulation results provided in Section IV verify this monotonic and limiting behavior.

1) Fast Iterative Greedy Algorithm: For the special case in which the random permutation $\Pi_\ell(k)$ is simply k for all ℓ, we can exploit the order of the parameter updates to obtain a slight improvement in the computational complexity of the algorithm. This results in what we refer to as the *fast* iterative greedy algorithm described below.

```
Initialization:
1) Generate a random $p \times r$ unitary matrix
    $\mathbf{U}$ and $(N-1)$ $p\times 1$ random unit norm vec-
    tors $\mathbf{v}_i$, $1 \leq i \leq N-1$.
2) Compute the matrix $\mathcal{R}_N(z)$ using (20).

Iteration: For $m \geq 0$, do the following.
1) If $m$ is a multiple of $N$:
    a) Calculate the optimal $\mathbf{U}$ and cor-
       responding $\xi_m$ using (18), (11), and
       (9) with $\mathbf{V}(z) = \mathcal{R}_N(z)$.
    b) Compute $\mathcal{L}_0(z) = \mathbf{V}(z)$ and $\mathcal{R}_1(z) = \mathbf{I}_p$.
```

$$
\begin{aligned}
\xi &= a - \underbrace{\frac{1}{2\pi}\int_0^{2\pi} W(\omega)\mathrm{Tr}\left[\mathbf{D}^\dagger(e^{j\omega})\mathcal{L}_k(e^{j\omega})\mathcal{R}_k(e^{j\omega})\mathbf{U}\right] d\omega}_{c} \\
&\quad + \frac{1}{2\pi}\int_0^{2\pi} W(\omega)\mathrm{Tr}\left[\mathbf{D}^\dagger(e^{j\omega})\mathcal{L}_k(e^{j\omega})\mathbf{v}_k\mathbf{v}_k^\dagger\mathcal{R}_k(e^{j\omega})\mathbf{U}\right] d\omega \\
&\quad - \frac{1}{2\pi}\int_0^{2\pi} e^{-j\omega}W(\omega)\mathrm{Tr}\left[\mathbf{D}^\dagger(e^{j\omega})\mathcal{L}_k(e^{j\omega})\mathbf{v}_k\mathbf{v}_k^\dagger\mathcal{R}_k(e^{j\omega})\mathbf{U}\right] d\omega \\
&\quad - \underbrace{\frac{1}{2\pi}\int_0^{2\pi} W(\omega)\mathrm{Tr}\left[\mathbf{U}^\dagger\mathcal{R}_k^\dagger(e^{j\omega})\mathcal{L}_k^\dagger(e^{j\omega})\mathbf{D}(e^{j\omega})\right] d\omega}_{c^*} \\
&\quad + \frac{1}{2\pi}\int_0^{2\pi} W(\omega)\mathrm{Tr}\left[\mathbf{U}^\dagger\mathcal{R}_k^\dagger(e^{j\omega})\mathbf{v}_k\mathbf{v}_k^\dagger\mathcal{L}_k^\dagger(e^{j\omega})\mathbf{D}(e^{j\omega})\right] d\omega \\
&\quad - \frac{1}{2\pi}\int_0^{2\pi} e^{j\omega}W(\omega)\mathrm{Tr}\left[\mathbf{U}^\dagger\mathcal{R}_k^\dagger(e^{j\omega})\mathbf{v}_k\mathbf{v}_k^\dagger\mathcal{L}_k^\dagger(e^{j\omega})\mathbf{D}(e^{j\omega})\right] d\omega \qquad (22) \\
&= a - 2\mathrm{Re}[c] + \mathbf{v}_k^\dagger \underbrace{\left[\frac{1}{2\pi}\int_0^{2\pi} W(\omega)(1-e^{-j\omega})\mathcal{R}_k(e^{j\omega})\mathbf{U}\mathbf{D}^\dagger(e^{j\omega})\mathcal{L}_k(e^{j\omega})d\omega\right]}_{\mathbf{B}} \mathbf{v}_k \\
&\quad + \mathbf{v}_k^\dagger \underbrace{\left[\frac{1}{2\pi}\int_0^{2\pi} W(\omega)(1-e^{j\omega})\mathcal{L}_k^\dagger(e^{j\omega})\mathbf{D}(e^{j\omega})\mathbf{U}^\dagger\mathcal{R}_k^\dagger(e^{j\omega})d\omega\right]}_{\mathbf{B}^\dagger} \mathbf{v}_k \qquad (23) \\
&= a - 2\mathrm{Re}[c] + \mathbf{v}_k^\dagger \underbrace{\left(\mathbf{B}+\mathbf{B}^\dagger\right)}_{\mathbf{Q}} \mathbf{v}_k = a - 2\mathrm{Re}[c] + \underbrace{\mathbf{v}_k^\dagger\mathbf{Q}\mathbf{v}_k}_{\nu} \qquad (24)
\end{aligned}
$$

```
   Otherwise, if $m \equiv k \bmod N$ where
   $1 \leq k \leq N-1$:
      a) From (19), update the left matrix
         as $\mathcal{L}_k(z) = \mathcal{L}_{k-1}(z)\widetilde{\mathbf{V}}_k(z)$.
      b) Calculate the optimal $\mathbf{v}_k$ and
         corresponding $\xi_m$ using (25), (24),
         (23), and (22).
      c) From (20), update the right ma-
         trix as $\mathcal{R}_{k+1}(z) = \mathbf{V}_k(z)\mathcal{R}_k(z)$.
2) Increment $m$ by 1 and return to
   Step 1).
```

As the iterations progress, the left matrix is shortened by the old optimal vectors $\mathbf{v}_k$, whereas the right matrix is lengthened by the newly computed ones. After all of the $\mathbf{v}_k$s have been optimized, the left matrix assumes the value of the right matrix, while the right matrix is then refreshed to be the identity matrix. This offers a slight improvement in the overall computational complexity of the algorithm, as the left and right matrices need not be completely recalculated at each iteration, as is required in the general case from above. As is shown through simulations in Section IV, in addition to being faster than the general random update algorithm, the fast algorithm performs nearly identically to the general one.

Prior to presenting the simulation results, we first introduce the phase feedback modification to the iterative algorithm for cases in which the desired response $\mathbf{D}(e^{j\omega})$ has a *phase-type ambiguity*, which we will define shortly.

III. Phase Feedback Modification

A. Phase-Type Ambiguity

Referring back to Fig. 1(b), suppose that we would like to design an FIR PU synthesis polyphase matrix approximant to that of the infinite-order PCFB as described in Section I-A-1). In this case, the desired response $\mathbf{D}(e^{j\omega})$ is any system that totally decorrelates and spectrally majorizes the blocked input signal $x(n)$ (i.e., $\mathbf{D}(e^{j\omega})$ diagonalizes $\mathbf{S_{xx}}(e^{j\omega})$ for every ω in such a way that the eigenvalues are arranged in descending order [1], [24]). This implies a *nonuniqueness* for the desired response $\mathbf{D}(e^{j\omega})$. To see this, note that $\mathbf{D}(e^{j\omega})$ must contain the unit norm eigenvectors of $\mathbf{S_{xx}}(e^{j\omega})$ arranged in some order to preserve the spectral majorization property. Partitioning $\mathbf{D}(e^{j\omega})$ into its columns as

$$\mathbf{D}(e^{j\omega}) = \begin{bmatrix}\mathbf{d}_0(e^{j\omega}) & \mathbf{d}_1(e^{j\omega}) & \cdots & \mathbf{d}_{M-1}(e^{j\omega})\end{bmatrix} \qquad (26)$$

it follows that $\mathbf{d}_k(e^{j\omega})$ is a unit norm eigenvector of $\mathbf{S_{xx}}(e^{j\omega})$ for all ω. As any unit magnitude scale factor of a unit norm eigenvector is itself a unit norm eigenvector, it follows that any system of the form

$$\begin{aligned}\mathbf{D}_a(e^{j\omega}) &= \Big[\mathbf{d}_0(e^{j\omega})e^{j\phi_0(\omega)} \quad \mathbf{d}_1(e^{j\omega})e^{j\phi_1(\omega)} \quad \cdots \\ &\qquad \mathbf{d}_{M-1}(e^{j\omega})e^{j\phi_{M-1}(\omega)}\Big] \\ &= \mathbf{D}(e^{j\omega})\boldsymbol{\Lambda}(e^{j\omega}) \qquad (27)\end{aligned}$$

where $\boldsymbol{\Lambda}(e^{j\omega}) = \operatorname{diag}\left(e^{j\phi_0(\omega)}, e^{j\phi_1(\omega)}, \ldots, e^{j\phi_{M-1}(\omega)}\right)$ is a valid desired response for an infinite-order PCFB. If the eigenvalues of $\mathbf{S_{xx}}(e^{j\omega})$ are distinct for all ω, then all valid desired responses are related to each other as in (27). On the other hand, if the eigenvalues are not distinct at some frequency, say ω_0, then at that frequency, the columns of any one desired response corresponding to the nondistinct eigenvalues can be expressed as a *unitary* combination of the same columns of any other desired response. As an example, suppose that at ω_0, the largest eigenvalue of $\mathbf{S_{xx}}(e^{j\omega})$ has multiplicity 2. Then, given any desired response $\mathbf{D}(e^{j\omega})$ of the form given in (26), we can obtain another desired response $\mathbf{D}_a(e^{j\omega})$ in which we have

$$\begin{aligned}\mathbf{D}_a&(e^{j\omega_0}) \\ &= \Big[\big[\mathbf{d}_0(e^{j\omega_0}) \ \ \mathbf{d}_1(e^{j\omega_0})\big]\,\boldsymbol{\Phi}(\omega_0) \ \ \mathbf{d}_2(e^{j\omega_0})e^{j\phi_2(\omega_0)} \ \ \cdots \\ &\qquad \mathbf{d}_{M-1}(e^{j\omega_0})e^{j\phi_{M-1}(\omega_0)}\Big] \\ &= \mathbf{D}(e^{j\omega_0}) \underbrace{\begin{bmatrix} \boldsymbol{\Phi}(\omega_0) & \mathbf{0} & \cdots & \cdots & \mathbf{0} \\ \mathbf{0} & e^{j\phi_2(\omega_0)} & 0 & \cdots & 0 \\ \vdots & 0 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \mathbf{0} & 0 & \cdots & 0 & e^{j\phi_{M-1}(w_0)} \end{bmatrix}}_{\boldsymbol{\Lambda}(e^{j\omega_0})}\end{aligned}$$

where $\boldsymbol{\Phi}(\omega_0)$ is a 2 × 2 unitary matrix. In general, for an eigenvalue with multiplicity μ, the corresponding eigenvectors of one desired response can be related in terms of any other via a $\mu \times \mu$ unitary matrix.

Any $p \times r$ desired response $\mathbf{D}_a(e^{j\omega})$ that has a nonuniqueness of the form

$$\mathbf{D}_a(e^{j\omega}) = \mathbf{D}(e^{j\omega})\boldsymbol{\Lambda}(e^{j\omega}) \qquad (28)$$

where $\mathbf{D}(e^{j\omega})$ is some $p \times r$ given desired response and $\boldsymbol{\Lambda}(e^{j\omega})$ is an $r \times r$ block diagonal matrix of unitary matrices will be said to have a *phase-type ambiguity*, since the phases of the columns are arbitrary in this case. (In the PCFB example described here, the number of blocks of $\boldsymbol{\Lambda}(e^{j\omega})$ is equal to the number of distinct eigenvalues of $\mathbf{S_{xx}}(e^{j\omega})$ and the size of each block is equal to the multiplicity of each of these eigenvalues.) When the desired response has a phase-type ambiguity, some desired responses may yield a better overall FIR PU approximant than others. The reason for this is that the causal FIR constraint we assume here imposes severe restrictions on the allowable phase of the FIR PU approximant. Since we do not know the best desired response to choose *a priori*, we propose a *phase feedback modification* to the iterative greedy algorithm of Section II-C in order to learn the proper desired response.

B. Derivation of the Phase Feedback Modification

Suppose that we are given a desired response $\mathbf{D}(e^{j\omega})$ with a phase-type ambiguity as in (28). In addition, suppose that the matrix $\boldsymbol{\Lambda}(e^{j\omega})$ from (28) corresponds only to a simple phase-type ambiguity of the form

$$\boldsymbol{\Lambda}(e^{j\omega}) = \operatorname{diag}\left(e^{j\phi_0(\omega)}, e^{j\phi_1(\omega)}, \ldots, e^{j\phi_{r-1}(\omega)}\right) \quad \forall\omega. \qquad (29)$$

The question then arises as to how to choose the phases $\{\phi_k(\omega)\}$ to minimize the mean-squared error in (5) with the desired response $\mathbf{D}(e^{j\omega})$ replaced by $\mathbf{D}_a(e^{j\omega})$ given to be

$$\xi = \frac{1}{2\pi}\int_0^{2\pi} W(\omega)\left\|\mathbf{D}_a(e^{j\omega}) - \mathbf{F}(e^{j\omega})\right\|_F^2 d\omega. \qquad (30)$$

To solve this problem, we partition the old given desired response $\mathbf{D}(e^{j\omega})$ and the FIR PU approximant $\mathbf{F}(e^{j\omega})$ as follows:

$$\begin{aligned}\mathbf{D}(e^{j\omega}) &= \begin{bmatrix}\mathbf{d}_0(e^{j\omega}) & \mathbf{d}_1(e^{j\omega}) & \cdots & \mathbf{d}_{r-1}(e^{j\omega})\end{bmatrix}\\ \mathbf{F}(e^{j\omega}) &= \begin{bmatrix}\mathbf{f}_0(e^{j\omega}) & \mathbf{f}_1(e^{j\omega}) & \cdots & \mathbf{f}_{r-1}(e^{j\omega})\end{bmatrix}.\end{aligned}$$

Then, from (30), it can easily be shown that we have

$$\xi = \frac{1}{2\pi}\int_0^{2\pi} W(\omega)\left(\sum_{k=0}^{r-1}\left\|\mathbf{d}_k(e^{j\omega})e^{j\phi_k(\omega)} - \mathbf{f}_k(e^{j\omega})\right\|_2^2\right)d\omega. \tag{31}$$

Note that we can minimize ξ from (31) by minimizing each term of the summation *pointwise in frequency*. This can be done here since the phases $\{\phi_k(\omega)\}$ are independent functions of k that have arbitrary response (in terms of ω). Hence, minimizing ξ is tantamount to minimizing

$$\ell_k(\omega) \triangleq \left\|\mathbf{d}_k(e^{j\omega})e^{j\phi_k(\omega)} - \mathbf{f}_k(e^{j\omega})\right\|_2^2 \tag{32}$$

for each k. Upon expanding $\ell_k(w)$ in (32), we get the following:

$$\begin{aligned}\ell_k(\omega) &= \left\|\mathbf{d}_k(e^{j\omega})\right\|_2^2 + \left\|\mathbf{f}_k(e^{j\omega})\right\|_2^2\\ &\quad - e^{-j\phi_k(\omega)}\mathbf{d}_k^\dagger(e^{j\omega})\mathbf{f}_k(e^{j\omega}) - \mathbf{f}_k^\dagger(e^{j\omega})\mathbf{d}_k(e^{j\omega})e^{j\phi_k(\omega)}.\end{aligned} \tag{33}$$

Expressing $\mathbf{d}_k^\dagger(e^{j\omega})\mathbf{f}_k(e^{j\omega})$ as

$$\mathbf{d}_k^\dagger(e^{j\omega})\mathbf{f}_k(e^{j\omega}) = \left|\mathbf{d}_k^\dagger(e^{j\omega})\mathbf{f}_k(e^{j\omega})\right| e^{j\theta_k(\omega)} \tag{34}$$

then we have $\mathbf{f}_k^\dagger(e^{j\omega})\mathbf{d}_k(e^{j\omega}) = |\mathbf{d}_k^\dagger(e^{j\omega})\mathbf{f}_k(e^{j\omega})|e^{-j\theta_k(\omega)}$, and so from (33), we get

$$\begin{aligned}\ell_k(\omega) &= \left\|\mathbf{d}_k(e^{j\omega})\right\|_2^2 + \left\|\mathbf{f}_k(e^{j\omega})\right\|_2^2\\ &\quad -2\left|\mathbf{d}_k^\dagger(e^{j\omega})\mathbf{f}_k(e^{j\omega})\right|\cos\left(\theta_k(\omega) - \phi_k(\omega)\right).\end{aligned} \tag{35}$$

Hence, to minimize $\ell_k(\omega)$, we must choose $\phi_k(\omega)$ as follows:

$$\phi_{k,\mathrm{opt}}(\omega) = \theta_k(\omega). \tag{36}$$

Thus, from (34), it can be seen that the optimal thing to do for each column of the desired response is to *mix* its phase with that of the FIR PU approximant. In other words, the phase of the FIR PU approximant must be fed back to the desired response in order to minimize the mean-squared error.

C. Greediness of the Phase Feedback Modification

With the phase feedback modification of (36) in effect, it can be shown that the iterative algorithm from Section II-C still remains greedy. To see this, suppose that a phase feedback is performed at the mth iteration and let $\xi_{m,\mathrm{before}}$ and $\xi_{m,\mathrm{after}}$ denote, respectively, the error before and after the phase feedback. Note that $\xi_{m,\mathrm{before}}$ and $\xi_{m,\mathrm{after}}$ are given by (5) and (30), respectively. For simplicity of notation, let $\mathbf{f}_{k;m}(e^{j\omega})$ denote the kth column of $\mathbf{F}(e^{j\omega})$ at the mth iteration and let $\theta_{k;m}(\omega)$ denote the phase of the inner product $\mathbf{d}_k^\dagger(e^{j\omega})\mathbf{f}_{k;m}(e^{j\omega})$ as in (34). Using (35) and (32) in (31), we get

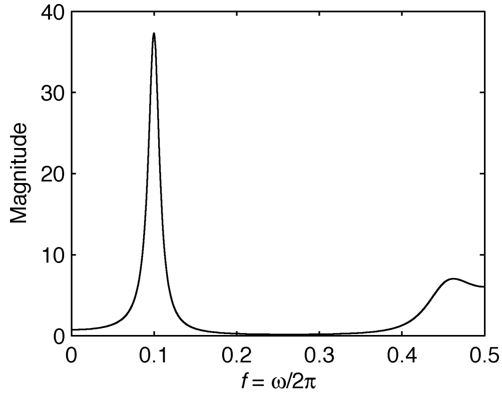


Fig. 2. Input psd $S_{xx}(e^{j\omega})$ of the AR(4) process $x(n)$.

$$\begin{aligned}&\xi_{m,\mathrm{before}} - \xi_{m,\mathrm{after}}\\ &= \frac{1}{\pi}\int_0^{2\pi} W(\omega)\left(\sum_{k=0}^{r-1}\left|\mathbf{d}_k^\dagger(e^{j\omega})\mathbf{f}_{k;m}(e^{j\omega})\right|\left(1-\cos\left(\theta_{k;m}(\omega)\right)\right)\right)d\omega\\ &\geq 0\end{aligned}$$

since the integrand from above is always nonnegative. Hence, it follows that $\xi_{m,\mathrm{after}} \leq \xi_{m,\mathrm{before}}$. As $\xi_{m+1,\mathrm{after}} \leq \xi_{m,\mathrm{after}}$, since the unmodified algorithm is greedy, we have $\xi_{m+1,\mathrm{after}} \leq \xi_{m,\mathrm{before}}$. Thus, the algorithm remains greedy even with the phase feedback modification in effect. As will be shown in Section IV-A regarding the design of PCFB-like FIR PU filter banks, the phase feedback modification can offer a better *magnitude-type* fit to the desired response than the unmodified algorithm.

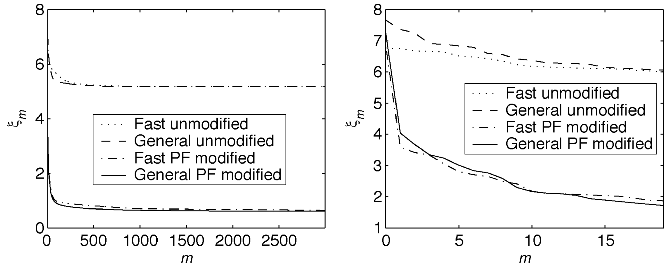
IV. Simulation Results

A. Design of PCFB-Like FIR PU Filter Banks

Recall that the proposed iterative algorithm can be used to design a *PCFB-like* filter bank when the desired response $\mathbf{D}(e^{j\omega})$ is the synthesis polyphase matrix of any infinite-order PCFB for the psd $\mathbf{S}_{\mathbf{xx}}(z)$ of the blocked filter bank input $\mathbf{x}(n)$ from Fig. 1(b). Suppose that the unblocked scalar input signal $x(n)$ from Fig. 1 is a real WSS autoregressive order 4 (AR(4)) process whose psd $S_{xx}(e^{j\omega})$ is as shown in Fig. 2. (As $x(n)$ is itself WSS, it follows that the psd of the blocked process $\mathbf{S}_{\mathbf{xx}}(z)$ is a pseudocirculant matrix [23] formed from the scalar psd $S_{xx}(z)$.) In the case where the scalar input signal $x(n)$ is WSS, the synthesis filters $\{F_k(z)\}$ corresponding to any infinite-order PCFB are ideal bandpass *compaction filters* corresponding to $S_{xx}(e^{j\omega})$ and its *peeled spectra* [24]. (Because of the orthnormality condition of (2), it follows that the corresponding analysis filters $\{H_k(z)\}$ are also ideal compaction filters.)

To test the proposed iterative greedy algorithm, we chose the following input parameters:

- $M = 4$, $N = 10$, $W(\omega) = 1$;
- 512 uniformly spaced frequency samples for numerical integration;



(a) (b)

Fig. 3. Mean-squared error ξ_m as a function of the iteration index m for the unmodified and phase feedback modified general and fast algorithms. (a) Plot of $KN = 3000$ iterations. (b) Magnified plot of the first 20 iterations.

TABLE I
AVERAGE TIME REQUIRED PER ITERATION FOR THE ITERATIVE GREEDY ALGORITHMS PROPOSED. (744 MHz INTEL PENTIUM III RUNNING MATLAB WAS USED.)

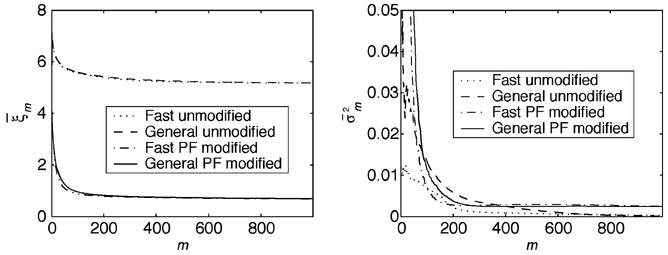
Algorithm	Time per iteration (sec)
Fast unmodified	0.4705
General unmodified	0.4843
Fast PF modified	0.5502
General PF modified	0.5904

- KN total iterations for some integer K.[1]

This implies that the synthesis filters $\{F_k(z)\}$ are causal and FIR of length $MN = 40$.

In Fig. 3(a), we have plotted the observed mean-squared error ξ_m as a function of the iteration index m for both the unmodified and phase feedback modified methods employing both the general as well as fast algorithms. As can be seen in all cases, the error decreased monotonically with iteration, as expected. In addition, the fast algorithm performed nearly identically to the general algorithm for both cases, and the phase feedback modified methods yielded a lower overall error than the unmodified ones. A magnified view of the observed error for the first 20 iterations is shown in Fig. 3(b). It can be seen that even though all of the algorithms exhibited similar initial errors, the errors of the phase feedback modified methods decreased more quickly than those of the unmodified algorithms.

In Table I, we have listed the processing time required per iteration for all of the algorithms proposed here. The processor used was an Intel Pentium III operating at 744 MHz running Matlab. As can be seen from Table I, there is a slight improvement in processing time required for the fast algorithms as opposed to the general ones. For a large number of iterations, this improvement becomes quite noticeable. In addition, as will soon be shown, the performance of the fast algorithms is nearly identical to those of the general ones, further justifying their use in practice.



(a) (b)

Fig. 4. (a) Average mean-squared error $\overline{\xi}_m$ and (b) average error variance $\overline{\sigma}^2_m$ as a function of the iteration index m for a total of $L = 30$ trial runs.

1) Convergence Analysis: To analyze the convergence properties of the iterative algorithms, as well as their sensitivity with respect to random initial conditions, the algorithms were run several times, each time with a different initial condition. Suppose that each algorithm was run a total of L times and that $\xi_{\ell;m}$ denotes the observed mean-squared error of the ℓth trial at the mth iteration, where here $0 < \ell < L - 1$. To gauge the behavior of the algorithms, we opted to calculate the average mean-squared error per iteration $\overline{\xi}_m$ as well as the average error variance per iteration $\overline{\sigma}^2_m$ defined as follows:

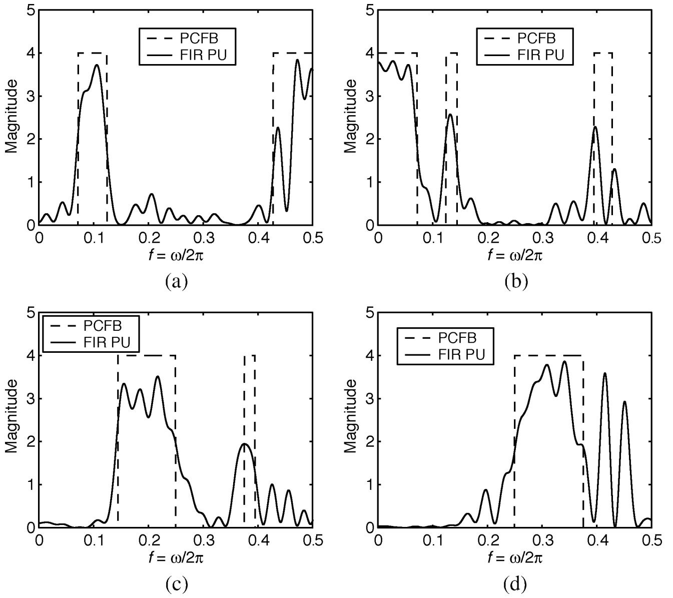
$$\overline{\xi}_m \triangleq \frac{1}{L}\sum_{\ell=0}^{L-1} \xi_{\ell;m} \tag{37}$$

$$\begin{aligned}\overline{\sigma}^2_m &\triangleq \frac{1}{L}\sum_{\ell=0}^{L-1} |\xi_{\ell;m} - \overline{\xi}_m|^2 \\ &= \frac{1}{L}\sum_{\ell=0}^{L-1} |\xi_{\ell;m}|^2 - |\overline{\xi}_m|^2.\end{aligned} \tag{38}$$

In Fig. 4(a) and (b), we have plotted, respectively, the average error per iteration $\overline{\xi}_m$ and average error variance per iteration $\overline{\sigma}^2_m$ for a total of $L = 30$ trial runs and for 1000 iterations. As can be seen from Fig. 4(a), all methods yielded a monotonic decreasing error and that the phase feedback modified methods outperformed the unmodified ones. In addition, the performance of the fast algorithms can be seen to be nearly identical to that of the general ones. More important, however, it can be seen from Fig. 4(b) that the variance of the error becomes very small once the number of iterations is large enough. At $m = 1000$, we had $\overline{\sigma}^2_m = 2.5937 \times 10^{-4}, 1.5142 \times 10^{-4}, 24.5190 \times 10^{-4}$, and 24.5783×10^{-4} for the fast unmodified, general unmodified, fast phase feedback, and general phase feedback methods, respectively. Although the variances of the phase feedback methods are larger than those of the unmodified algorithms, they are still relatively small given the inherent additional amount of randomness of the phase feedback methods over the unmodified ones. This suggests that the algorithm is relatively *insensitive* with respect to the choice of the initial condition. Furthermore, this suggests that the local optimality guaranteed by the iterative greedy algorithms is perhaps close to being *global*.

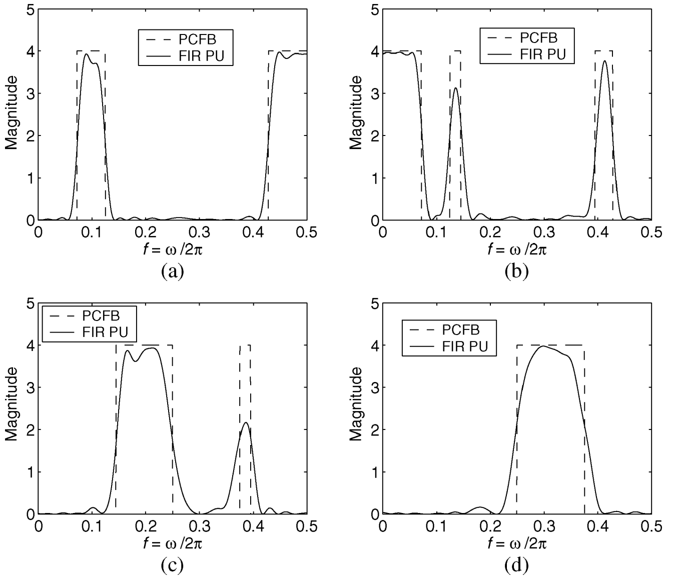
2) Filter Response Results: To see the effects of the phase feedback modification more clearly, in Figs. 5 and 6, we have plotted, respectively, the magnitude squared responses of the resulting synthesis filters $\{F_k(z)\}$ for the unmodified and phase feedback modified general algorithms together with the

[1] We opted for an integer multiple of N iterations to ensure that all of the parameters were optimized the same number of times. In addition, for all of the simulation results presented in this section, unless otherwise stated, we chose $K = \lceil 3000/N \rceil$.



(a) (b) (c) (d)

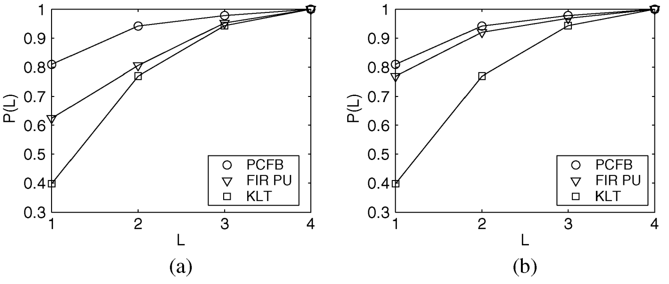
Fig. 5. Magnitude-squared responses of the PCFB and FIR PU synthesis filters using the unmodified general iterative algorithm: (a) $F_0(z)$; (b) $F_1(z)$; (c) $F_2(z)$; and (d) $F_3(z)$.



(a) (b) (c) (d)

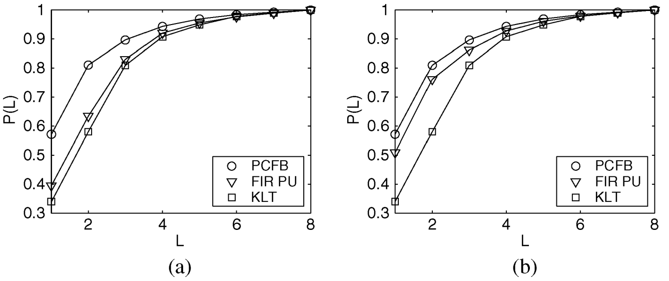
Fig. 6. Magnitude-squared responses of the PCFB and FIR PU synthesis filters using the phase feedback modified general iterative algorithm: (a) $F_0(z)$; (b) $F_1(z)$; (c) $F_2(z)$; and (d) $F_3(z)$.

responses of the infinite order PCFB synthesis filters. (Due to the phase-type ambiguity present in $\mathbf{D}(e^{j\omega})$ here, only the magnitude has been plotted since the infinite order PCFB filters can have arbitrary phase.) As can be seen, the FIR synthesis filters designed with the phase feedback modification offer a better *magnitude-type* fit to the infinite-order PCFB filters than those designed with the unmodified algorithm. Due to this observed phenomenon, we opted to carry out the rest of the PCFB simulations using the phase feedback modification. It should also be noted that the remainder of the PCFB simulations in this section were carried out for the real AR(4) process $x(n)$ with psd $S_{xx}(e^{j\omega})$ as in Fig. 2.



(a) (b)

Fig. 7. Proportion of the total variance $P(L)$ as a function of the number of subbands kept L for an $M = 4$ channel system with (a) $N = 3$ and (b) $N = 10$.



(a) (b)

Fig. 8. Proportion of the total variance $P(L)$ as a function of the number of subbands kept L for an $M = 8$ channel system with (a) $N = 3$ and (b) $N = 10$.

3) Multiresolution Optimality Results: Referring to Fig. 1, recall from Section I-A-1) that by definition, the PCFB, if it exists for a class of filter banks, is such that its subband variance vector $\boldsymbol{\sigma}$ from (3) *majorizes* the subband variance vector of any other filter bank in the class under consideration [1], [8]. As such, it is optimal in a multiresolution sense in that it successively compacts as much of the signal energy as possible into each subband starting with the first [18]. One suitable measure of multiresolution optimality is the proportion of the partial subband variances to the total. By preserving only L out of M subbands, this proportion is given by

$$P(L) \triangleq \frac{\left(\sum_{k=0}^{L-1} \sigma_{w_k}^2\right)}{\left(\sum_{k=0}^{M-1} \sigma_{w_k}^2\right)}, \quad 1 \leq L \leq M.$$

Because of the subband majorization property of the PCFB, the PCFB maximizes $P(L)$ for all L.

Using the proposed iterative algorithm for the design of a PCFB-like filter bank for the real AR(4) process $x(n)$ considered here, a plot of the observed proportion $P(L)$ as a function of the number of subbands preserved L is shown in Fig. 7 for $N = 3$ and $N = 10$. Included in Fig. 7 are the performances of the zeroth-order PCFB (namely the KLT) as well as the infinite-order one. As can be seen, both FIR filter banks designed outperform the KLT. Furthermore, by comparing Fig. 7(a) and (b), it can be seen that as the filter order increased, the subband variances came closer to those of the infinite-order PCFB.

To show another example of this phenomenon, we considered the design of an $M = 8$ channel system. For this case, a plot of $P(L)$ as a function of L is shown in Fig. 8(a) and (b) for $N = 3$ and $N = 10$, respectively. As before, it can be seen that

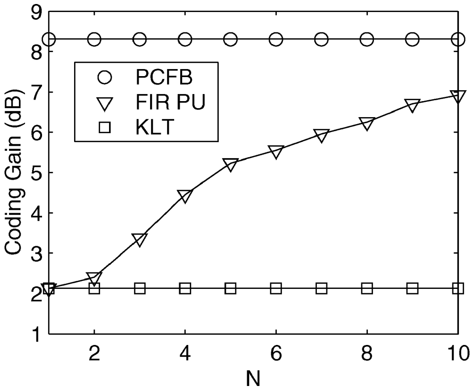
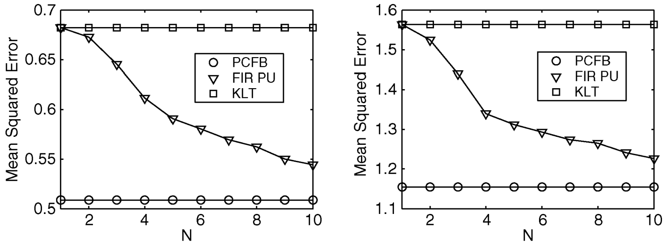


Fig. 9. Observed coding gain G_{code} as a function of the FIR PU filter order parameter N.



(a)

(b)

Fig. 10. Noise reduction performance (ξ from (39)) with zeroth-order subband Wiener filters as a function of the FIR PU filter order parameter N for (a) noise variance (η^2) of 1 and (b) noise variance of 4.

as the filter order increased, the subband variances of the FIR filter banks came closer to those of the infinite-order PCFB.[2] This is in accordance with intuition that states that as the filter order increases, the designed FIR PU filter banks should behave more and more like the infinite-order PCFB. Upon considering other objectives for which the PCFB is optimal, this PCFB-like behavior for the optimized FIR filter banks will become more apparent.

4) Coding Gain Results: Recall from Section I-A-1) that the PCFB, if it exists, is simultaneously optimal for a variety of objective functions of the vector of subband variances $\boldsymbol{\sigma}$ from (3). In particular, it is optimal for coding gain with optimal bit allocation in the subbands [1], [8]. Assuming optimal bit allocation, the coding gain is given by [23]

$$G_{\text{code}} = \frac{\left(\frac{1}{M}\sum\limits_{k=0}^{M-1}\sigma_{w_k}^2\right)}{\left(\prod\limits_{k=0}^{M-1}\sigma_{w_k}^2\right)^{\frac{1}{M}}}.$$

In other words, the coding gain is the arithmetic mean/geometric mean (AM/GM) ratio of the subband variances in this case. The coding gain is lower bounded by unity (because of the AM/GM inequality) and upper bounded by the gain produced by the PCFB.

Here, the proposed iterative algorithm was used to design an $M = 4$ channel PCFB-like filter bank in which the synthesis polyphase matrix length N was varied from 1 to 10. A plot of the coding gain observed as a function of N is shown in Fig. 9. In addition, we have included the coding gain of the KLT (2.1276 dB) along with that of the infinite- order PCFB (8.3081 dB). From Fig. 9, we can see that even at small filter orders, the FIR PU filter banks designed yielded a much larger coding gain than the KLT. Furthermore, the optimized FIR filter banks exhibited a *monotonically increasing* coding gain. This is consistent with intuition, which dictates that as the filter order increases, the FIR filter banks designed should become more and more PCFB-like. From Fig. 9, it appears as though the coding gain of the FIR filter banks will asymptotically achieve the infinite-order PCFB performance as $N \to \infty$.

5) Noise Reduction Using Zeroth-Order Wiener Filters: In addition to being optimal for coding gain, the PCFB, if it exists, is optimal for any concave objective of $\boldsymbol{\sigma}$ [1]. One such objective is noise reduction with zeroth-order Wiener filters in the subbands if the input noise is white [1]. In other words, if the input to the filter bank of Fig. 1(a) is $x(n) = s(n) + \mu(n)$, where $s(n)$ is a pure signal and $\mu(n)$ is a white noise process, and if the subband processors $\{\mathcal{P}_k\}$ are taken to be zeroth-order Wiener filters (i.e., multipliers), then the PCFB for $x(n)$ (which is also the PCFB for $s(n)$ in this case) is optimal in terms of minimizing the mean-squared value of the error $e(n) \triangleq s(n) - \widehat{x}(n)$ [1]. With the presence of zeroth-order Wiener filters, the mean-squared error ξ is in general given by

$$\xi = \frac{1}{M}\sum_{i=0}^{M-1}\frac{\sigma_{w_i}^2\eta^2}{\sigma_{w_i}^2 + \eta^2} \tag{39}$$

where $\sigma_{w_i}^2$ denotes the variance of the ith subband when the input is the desired signal $s(n)$ and η^2 denotes the variance of the white noise process $\mu(n)$. As ξ is a *concave* function of the subband variance vector $\boldsymbol{\sigma}$ from (3), the PCFB for $s(n)$, if it exists, is optimal for this objective function [1].

Using the same FIR PU filter banks as those computed in Section IV-A-4), the observed mean-squared error ξ from (39) as a function of N is shown in Fig. 10 for (a) $\eta^2 = 1$ and (b) $\eta^2 = 4$. As can be seen in both cases, the FIR filter banks significantly outperform the KLT Furthermore, it can be seen that the error *monotonically decreased* as N increased, in accordance with intuition. Asymptotically, it appears as though the optimized FIR filter bank is trying to emulate the behavior of the infinte order PCFB.

6) Power Minimization for DMT-Type Transmultiplexers: In addition to applications in data-compression-related objectives, the theory of PCFBs has also been found useful in digital communications involving the design of optimal DMT-type PU transmultiplexers [27]. A typical nonredundant PU transmultiplexer [23] in polyphase form is shown in Fig. 11. We distinguish nonredundant transmultiplexers from redundant ones such as those used in typical DMT transceivers in which the polyphase matrix $\mathbf{F}(z)$ is $L \times M$ with $L < M$. The system of Fig. 11 represents a digital communications system in which M users $\{x_k(n)\}$ transmit data over a common path. Prior to receiving the data and separating the users at the receiver, the

[2] It should be noted that this phenomenon continues to hold true for larger M; however, the results become less dramatic since the gap between the KLT and infinite-order PCFB shrinks as M increases.

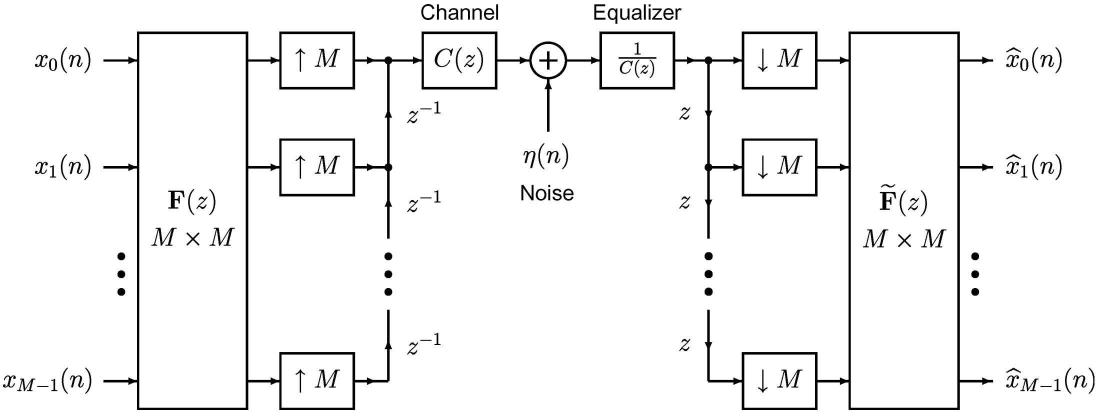


Fig. 11. Uniform PU nonredundant transmultiplexer.

incoming signal undergoes a linear distortion in the form of the channel $C(z)$, and a noise process $\eta(n)$ is added to it. To undo the effects of the channel, we assume that a *zero-forcing equalizer* [27] of $1/C(z)$ has been used, as can be seen in Fig. 11.

Assuming that the kth input signal $x_k(n)$ consists of pulse-amplitude-modulated (PAM) symbols with b_k bits and power P_k, then if the noise $\eta(n)$ is *Gaussian*, the probability of error in detecting the symbol $x_k(n)$ is given by [27] to be

$$\mathcal{P}_e(k) = 2(1-2^{-b_k})\mathcal{Q}\left(\sqrt{\frac{3P_k}{(2^{2b_k}-1)\sigma_{q_k}^2}}\right). \tag{40}$$

Here, $\mathcal{Q}(x)$ is the *Marcum* $\mathcal{Q}$ *function*, which is frequently used in communications. In addition, $\sigma_{q_k}^2$ denotes the noise power seen at the kth output $\widehat{x}_k(n)$. Solving (40) for P_k yields

$$P_k = \beta\left(\mathcal{P}_e(k), b_k\right)\sigma_{q_k}^2$$

where

$$\beta\left(\mathcal{P}_e(k), b_k\right) = \frac{(2^{2b_k}-1)}{3}\left[\mathcal{Q}^{-1}\left(\frac{\mathcal{P}_e(k)}{2(1-2^{-b_k})}\right)\right]^2.$$

As P_k is a linear function of $\sigma_{q_k}^2$, it follows that the total power P given by

$$P = \sum_{k=0}^{M-1} P_k \tag{41}$$

is a *convex* function of the variances $\{\sigma_{q_k}^2\}$. As such, this power is minimized iff the PU filter bank $\mathbf{F}(z)$ is chosen to be a PCFB for the effective noise process seen at the input to the receiver. If $\eta(n)$ is a WSS process with psd $S_{\eta\eta}(e^{j\omega})$, then the effective noise seen at the receiver input is WSS with psd $S_{\eta\eta}(e^{j\omega})/|C(e^{j\omega})|^2$ Hence, the total power P from (41) is minimized iff $\mathbf{F}(z)$ is a PCFB for the psd $S_{\eta\eta}(e^{j\omega})/|C(e^{j\omega})|^2$.

As an example, suppose that the desired probability of error is $\mathcal{P}_e(k) = 10^{-9}$ for all k. In addition, suppose that we have $b_0 = 2, b_1 = 3, b_2 = 4$, and $b_3 = 5$. It should be noted that this is a not an optimal bit allocation [27] and is only chosen here as such for simplicity. Finally, suppose that the effective noise psd $S_{\eta\eta}(e^{j\omega})/|C(e^{j\omega})|^2$ is simply the psd $S_{xx}(e^{j\omega})$ shown in Fig. 2. Then, using the proposed iterative algorithm, the required powers as a function of the synthesis polyphase order N is shown in Fig. 12. As can be seen, the FIR filter banks designed here significantly outperform the KLT and exhibit a *monotonically decreasing* power as a function of N, in accordance with intuition. Furthermore, as before, the optimized FIR filter bank appears to be approaching the performance of the infinite-order PCFB as the order increases.

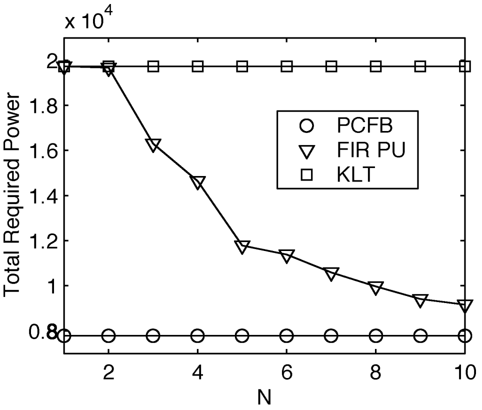


Fig. 12. Nonredundant DMT-type transmultiplexer total required power P as a function of the FIR PU filter order parameter N.

B. FIR PU Interpolation Problem

Recall from Section I-A-2) that the FIR PU interpolation problem involves finding an FIR PU system of a certain McMillan degree, say $\mathbf{F}(e^{j\omega})$, which takes on a prescribed set of L values, say $\mathcal{U}_0, \mathcal{U}_1, \ldots, \mathcal{U}_{L-1}$, over a prescribed set of L frequencies, say $\omega_0, \omega_1, \ldots, \omega_{L-1}$. In other words, we seek an FIR PU $\mathbf{F}(z)$ of a certain degree such that $F(e^{j\omega_k}) = \mathcal{U}_k$ for all $0 \le k \le L-1$. (Clearly, the matrices $\{\mathcal{U}_k\}$ must be unitary.) As mentioned in Section I-A-2), there is no known solution to the FIR PU interpolation problem. However, for this problem, the proposed iterative algorithm can be used to approximate an interpolant. In this case, the desired response $\mathbf{D}(e^{j\omega})$ is as follows:

$$\mathbf{D}(e^{j\omega}) = \begin{cases} \mathcal{U}_k, & \omega = \omega_k \; \forall \, 0 \le k \le L-1 \\ \text{do not care}, & \text{otherwise.} \end{cases}$$

As we do not care about the response at all frequencies not in the set $\{\omega_k\}$, it only makes sense that these regions be given no weight in the approximation problem. One weight function that accommodates this need is the interpolation weight function $W_{\text{int}}(\omega)$, given by the following:

$$W_{\text{int}}(\omega) = 2\pi \sum_{k=0}^{L-1} p_k \delta(\omega - \omega_k). \tag{42}$$

Here, the p_ks are discrete weight parameters used to emphasize the design of some interpolation conditions over others, which satisfies

$$p_k \geq 0, \quad \sum_{k=0}^{L-1} p_k = 1.$$

In other words, $\{p_k\}$ is a discrete probability density function (pdf). Substituting (42) into the expression for the weighted mean-squared error ξ from (5) yields

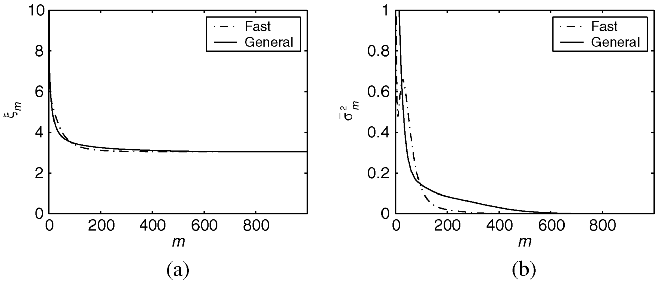
$$\xi = \sum_{k=0}^{L-1} p_k \left\| \mathbf{D}(e^{j\omega_k}) - \mathbf{F}(e^{j\omega_k}) \right\|_F^2 = \sum_{k=0}^{L-1} p_k \left\| \mathcal{U}_k - \mathbf{F}(e^{j\omega_k}) \right\|_F^2.$$

Hence, with the interpolation weight function $W_{\text{int}}(\omega)$ from (42), the mean-squared error integral becomes a *discrete summation*. This simplifies the proposed algorithm since no numerical integration is required.

1) Example 1: As an example, suppose that we seek a 3 × 2 FIR PU system $\mathbf{F}(z)$ such that $\mathbf{F}(e^{j\omega_k}) = \mathcal{U}_k$ for $0 \leq k \leq 3$, where $\mathcal{U}_0, \ldots, \mathcal{U}_3$ are randomly chosen 3 × 2 unitary matrices. Furthermore, suppose that the frequencies are chosen as

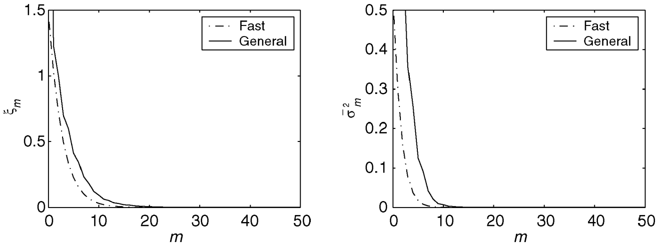
$$\omega_0 = 0, \quad \omega_1 = \frac{\pi}{2}, \quad \omega_2 = \frac{3\pi}{4}, \quad \omega_3 = \frac{5\pi}{4}.$$

Since there are four interpolation conditions, we might expect that we need $N \geq 4$ for the FIR PU interpolant in general. Using the proposed iterative algorithm for $N = 4$, the observed average mean-squared error $\overline{\xi}_m$ from (37) and average error variance $\overline{\sigma}_m^2$ from (38) are shown in Fig. 13(a) and (b), respectively, for both the fast and general algorithms using a total of 30 trial runs of each method. Here, we used $p_0 = p_1 = p_2 = p_3 = 1/4$ (i.e., uniform weighting) and KN iterations, where we chose $K = \lceil 1000/N \rceil$. From Fig. 13(b), the error appears to be rather insensitive with respect to the choice of initial condition. Here, we have $\overline{\sigma}_m^2 = 7.8708 \times 10^{-14}$ and 3.7253×10^{-7} at $m = 1000$ for the fast and general algorithms, respectively. This suggests that the algorithms perhaps converge to a global optimum, although there is no proof of this statement. As the error appears to have saturated at a nonzero value (in this case, $\overline{\xi}_m = 3.0429$ for both algorithms), this suggests that there may not exist an FIR PU system with $N = 4$, which satisfies the desired interpolation conditions. Despite this, the algorithms have found a good approximant to the desired interpolant.



(a) (b)

Fig. 13. FIR PU_ interpolation problem—Example 1: (a) Average mean-squared error $\overline{\xi}_m$ and (b) average error variance $\overline{\sigma}_m^2$ as a function of the iteration index m for a total of $KN = 1000$ iterations and $L = 30$ trial runs.



(a) (b)

Fig. 14. FIR PU_ interpolation problem—Example 2: (a) Average mean-squared error $\overline{\xi}_m$ and (b) average error variance $\overline{\sigma}_m^2$ as a function of the iteration index m for a total of $KN = 50$ iterations and $L = 30$ trial runs.

2) Example 2: To further test the performance of the proposed iterative algorithm, we can use it to obtain an FIR PU system for which we know that an interpolant exists. For example, suppose that we seek a 3 × 2 FIR PU system $\mathbf{F}(z)$ such that

$$\begin{aligned} \mathbf{F}(e^{j\omega_0}) &= \mathcal{U}_0 = (\mathbf{I} - \mathbf{v}\mathbf{v}^\dagger + e^{-j\omega_0}\mathbf{v}\mathbf{v}^\dagger)\mathbf{U} \\ \mathbf{F}(e^{j\omega_1}) &= \mathcal{U}_1 = (\mathbf{I} - \mathbf{v}\mathbf{v}^\dagger + e^{-j\omega_1}\mathbf{v}\mathbf{v}^\dagger)\mathbf{U}. \end{aligned}$$

Here, $\mathbf{v}$ is an arbitrary 3 × 1 unit norm vector, and $\mathbf{U}$ is a 3 × 2 arbitrary unitary matrix. As there are two interpolation conditions, we expect that in general, we need $N \geq 2$ here. Clearly, for $N = 2$, the choice

$$\mathbf{F}(z) = (\mathbf{I} - \mathbf{v}\mathbf{v}^\dagger + z^{-1}\mathbf{v}\mathbf{v}^\dagger)\mathbf{U} \tag{43}$$

satisfies the desired interpolation conditions. Using the proposed iterative algorithm, we can see if the algorithm can *converge* to an interpolant similar to the one from (43). For this simulation, we chose $\omega_0 = 0$ and $\omega_1 = 3\pi/4$ and $p_0 = p_1 = 1/2$ (i.e., uniform weighting). A plot of the observed average mean-squared error $\overline{\xi}_m$ and average error variance $\overline{\sigma}_m^2$ is shown in Fig. 14(a) and (b), respectively, for both the fast and general algorithms using a total of 30 trial runs of both methods. Here, the number of iterations chosen was KN with $K = \lceil 50/N \rceil$. As we can see, it appears as though the algorithm does in fact converge to desired interpolant. At $m = 50$, we have $\overline{\xi}_m = 4.1796 \times 10^{-9}$ and 7.4645×10^{-7} for the fast and general algorithms, respectively, both of which are very close to zero. Furthermore, we have $\overline{\sigma}_m^2 = 6.1949 \times 10^{-18}$ and 8.9227×10^{-13}, respectively, for the fast and general algorithms at $m = 50$. This strongly suggests that the proposed algorithms indeed converge to a *global* optimum in this case.

In summary, even though there is no general solution to the FIR PU interpolation problem, the proposed algorithms offer a way to approximate a suitable interpolant.

V. Concluding Remarks

In this paper, we proposed an iterative greedy algorithm to solve the FIR PU approximation problem using the complete parameterization of such systems in terms of Householder-like building blocks. Furthermore, we proposed a phase feedback modification to our algorithm for cases in which the desired response has a *phase-type* ambiguity as discussed in Section III.

Simulation results presented showed the usefulness of the proposed iterative algorithm for designing PCFB-like filter banks. As opposed to other methods, which compute the first filter required (an FIR compaction filter) and then complete the filter bank via an appropriate KLT [11], [16], this method *simultaneously* calculates all of the filters at once. This has the advantage that we do not have to worry about different filter banks formed from different spectral factors of the FIR compaction filter. The FIR PU filter banks designed here were shown to behave more and more like the PCFB as the filter order increased, in terms of numerous objective functions.

In addition to designing PCFB-like filter banks, we showed that the proposed iterative algorithm could also be used for the FIR PU interpolation problem. Although there is no known solution for this problem, the proposed algorithm can always provide a way to approximate an interpolant. As the iterative algorithm is only guaranteed to reach a *local optimum*, it cannot be used to solve the FIR PU interpolation problem, except for cases in which the mean-squared error goes to zero.

There are still several open problems that remain. For many practical filter banks, a linear phase constraint on the analysis/synthesis filters is desired in addition to the PU condition imposed here. At this time, it is unclear as to how to generalize the iterative algorithm to account for a linear phase constraint and how the resulting algorithm would behave with the phase feedback modification in effect. In addition to the FIR PU interpolation problem mentioned previously, the problem of generalizing the iterative algorithm to the multidimensional case still remains open. This is because in the general multidimensional case, there is no known way to completely parameterize FIR PU systems using Householder-like building blocks [23]. The reason for this is that the notion of poles and zeros does not exist in the multidimensional case. Such problems may not exist if we restrict our attention to special classes of FIR PU systems, such as *separable* systems. These open problems are currently the subject of further research.

Matlab code for the proposed iterative algorithm presented here is available online at [17].

References

[1] S. Akkarakaran and P. P. Vaidyanathan, "Filterbank optimization with convex objectives and the optimality of principal component forms," *IEEE Trans. Signal Process.*, vol. 49, no. 1, pp. 100–114, Jan. 2001.

[2] H. Caglar, Y. Liu, and A. N. Akansu, "Statistically optimized PR-QMF design," in *Proc. SPIE 1605, Wavelet Appl. Signal Image Processing*, San Diego, CA, 1991, pp. 86–94.

[3] P. Delsarte, B. Macq, and D. T. M. Slock, "Signal-adapted multiresolution transform for image coding," *IEEE Trans. Inform. Theory*, vol. 38, no. 2, pp. 897–904, Mar. 1992.

[4] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 1985.

[5] Y. Huang and P. M. Schultheiss, "Block quantization of correlated Gaussian random variables," *IEEE Trans. Commun. Syst.*, vol. C-10, no. 3, pp. 289–296, Sep. 1963.

[6] O. S. Jahromi, B. A. Francis, and R. H. Kwong, "Algebraic theory of optimal filterbanks," *IEEE Trans. Signal Process.*, vol. 51, no. 2, pp. 442–457, Feb. 2003.

[7] A. Kiraç and P. P. Vaidyanathan, "Optimality of orthonormal transforms for subband coding," presented at the IEEE DSP Workshop, Bryce Canyon, UT, Aug. 1998.

[8] ——, "On existence of FIR principal component filter banks," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, vol. 3, Seattle, WA, May 1998, pp. 1329–1332.

[9] P. Moulin, M. Anitescu, K. O. Kortanek, and F. A. Potra, "The role of linear semi-infinite programming in signal-adapted QMF bank design," *IEEE Trans. Signal Process.*, vol. 45, no. 9, pp. 2160–2174, Sep. 1997.

[10] P. Moulin, M. Anitescu, and K. Ramchandran, "Theory of rate-distortion-optimal, constrained filterbanks-application to IIR and FIR biorthogonal designs," *IEEE Trans. Signal Process.*, vol. 48, no. 4, pp. 1120–1132, Apr. 2000.

[11] P. Moulin and M. K. Mihçak, "Theory and design of signal-adapted FIR paraunitary filter banks," *IEEE Trans. Signal Process.*, vol. 46, no. 4, pp. 920–929, Apr. 1998.

[12] V. P. Sathe and P. P. Vaidyanathan, "Effects of multirate systems on the statistical properties of random signals," *IEEE Trans. Signal Process.*, vol. 41, no. 1, pp. 131–146, Jan. 1993.

[13] K. Sayood, *Introduction to Data Compression*, 2nd ed. San Diego, CA: Academic, 2000.

[14] A. Tkacenko and P. P. Vaidyanathan, "On the eigenfilter design method and its applications: a tutorial," *IEEE Trans. Circuits Syst. II*, vol. 50, no. 9, pp. 497–517, Sep. 2003.

[15] ——, "Iterative gradient technique for the design of least squares optimal FIR magnitude squared Nyquist filters," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, vol. 1, Montreal, QC, Canada, May 2004, pp. 1–4.

[16] ——, "On the spectral factor ambiguity of FIR energy compaction filter banks," *IEEE Trans. Signal Process.*, vol. 54, no. 1, pp. 380–385, Jan. 2006.

[17] A. Tkacenko. (2004) Matlab m-Files. [Online]. Available: http://www.systems.caltech.edu/dsp/students/andre/index.html

[18] M. K. Tsatsanis and G. B. Giannakis, "Principal component filter banks for optimal multiresolution analysis," *IEEE Trans. Signal Process.*, vol. 43, no. 8, pp. 1766–1777, Aug. 1995.

[19] D. W. Tufts and J. T. Francis, "Designing digital lowpass filters: Comparison of some methods and criteria," *IEEE Trans. Audio Electroacoust.*, vol. AU-18, pp. 487–494, Dec. 1970.

[20] J. Tuqan and P. P. Vaidyanathan, "A state space approach to the design of globally optimal FIR energy compaction filters," *IEEE Trans. Signal Process.*, vol. 48, no. 10, pp. 2822–2838, Oct. 2000.

[21] M. Unser, "An extension of the KLT for wavelets and perfect reconstruction filter banks," in *Proc. SPIE 2034, Wavelet Appl. Signal Image Processing*, San Diego, CA, 1993, pp. 45–56.

[22] ——, "On the optimality of ideal filters for pyramid and wavelet signal approximation," *IEEE Trans. Signal Process.*, vol. 41, no. 12, pp. 3591–3596, Dec. 1993.

[23] P. P. Vaidyanathan, *Multirate Systems and Filter Banks*. Englewood Cliffs, NJ: Prentice-Hall, 1993.

[24] ——, "Theory of optimal orthonormal subband coders," *IEEE Trans. Signal Process.*, vol. 46, no. 6, pp. 1528–1543, Jun. 1998.

[25] P. P. Vaidyanathan and S. Akkarakaran, "A review of the theory and applications of optimal subband and transform coders," *J. Appl. Computational Harmonic Anal.*, vol. 10, pp. 254–289, 2001.

[26] P. P. Vaidyanathan and A. Kiraç, "Cyclic lti systems and the paraunitary interpolation problem," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, vol. 3, Seattle, WA, May 1998, pp. 1445–1448.

[27] P. P. Vaidyanathan, Y.-P. Lin, S. Akkarakaran, and S.-M. Phoong, "Discrete multitone modulation with principal component filter banks," *IEEE Trans. Circuits Syst. I*, vol. 49, no. 10, pp. 1397–1412, Oct. 2002.

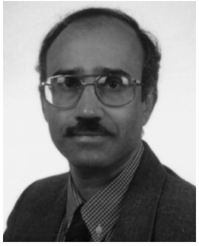
[28] P. P. Vaidyanathan, T. Q. Nguyen, Z. Doğanata, and T. Saramäki, "Improved technique for design of perfect reconstruction FIR QMF banks with lossless polyphase matrices," *IEEE Trans. Acoustics, Speech, Signal Processing*, vol. 37, no. 7, pp. 1042–1056, Jul. 1989.

[29] B. Xuan and R. H. Bamberger, "FIR principal component filter banks," *IEEE Trans. Signal Process.*, vol. 46, no. 4, pp. 930–940, Apr. 1998.

Andre Tkacenko (S'00–M'05) was born in Santa Clara, CA, on February 24, 1977. He received the B.S., M.S., and Ph.D. degrees in electrical engineering from the California Institute of Technology (Caltech), Pasadena, in 1999, 2001, and 2004, respectively.

Currently, he is a member of the Digital Signal Processing Research Group at the Jet Propulsion Laboratory, Pasadena, CA. His research interests include digital signal processing, multirate systems, optimization algorithms, and their applications in digital communications and data compression.

Dr. Tkacenko was awarded the Graduate Division Fellowship from Caltech in 1999. He received the Charles Wilts Prize in 2004 for outstanding independent research in electrical engineering for his Ph.D. dissertation titled "Optimization Algorithms for Realizable Signal-Adapted Filter Banks."



P. P. Vaidyanathan (S'80–M'83–SM'88–F'91) was born in Calcutta, India, on October 16, 1954. He received the B.Sc. (Hons.) degree in physics and the B.Tech. and M.Tech. degrees in radiophysics and electronics, all from the University of Calcutta, Calcutta, India, in 1974, 1977, and 1979, respectively, and the Ph.D. degree in electrical and computer engineering from the University of California, Santa Barbara, in 1982.

He was a Postdoctoral Fellow at the University of California, Santa Barbara, from September 1982 to March 1983. In March 1983, he joined the Electrical Engineering Department, Calfornia Institute of Technology (Caltech), as an Assistant Professor, where since 1993, he has been Professor of electrical engineering. He has authored a number of papers in IEEE journals and is the author of the book *Multirate Systems and Filter Banks* (Englewood Cliffs, NJ: Prentice-Hall, 1993). He has written several chapters for various signal processing handbooks. His main research interests are in digital signal processing, multirate systems, wavelet transforms, and signal processing for digital communications. He is a consulting editor for the journal *Applied and Computational Harmonic Analysis*.

Dr. Vaidyanathan served as Vice-Chairman of the Technical Program committee for the 1983 IEEE International Symposium on Circuits and Systems and as the Technical Program Chairman for the 1992 IEEE International Symposium on Circuits and Systems. He was an Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS from 1985 to 1987 and is currently an Associate Editor for the IEEE SIGNAL PROCESSING LETTERS. He was a Guest Editor in 1998 for special issues of the IEEE TRANSACTIONS ON SIGNAL PROCESSING and the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS II on the topics of filterbanks, wavelets, and subband coders. He was a recepient of the Award for Excellence in Teaching from the California Institute of Technology for the years 1983–1984, 1992–1993, and 1993–1994. He also received the NSF's Presidential Young Investigator Award in 1986. In 1989, he received the IEEE ASSP Senior Award for his paper on multirate perfect-reconstruction filterbanks. In 1990, he was recepient of the S. K. Mitra Memorial Award from the Institute of Electronics and Telecommuncations Engineers, India, for his joint paper in the *IETE Journal*. He was also the coauthor of a paper on linear-phase perfect reconstruction filterbanks in the IEEE TRANSACTIONS ON SIGNAL PROCESSING, for which the first author (T. Nguyen) received the Young Outstanding Author Award in 1993. He received the 1995 F. E. Terman Award of the American Society for Engineering Education, sponsored by Hewlett Packard Co., for his contributions to engineering education, especially the book *Multirate Systems and Filter Banks*. He has given several plenary talks, including the Sampta'01, Eusipco'98, SPCOM'95, and Asilomar'88 conferences on signal processing. He was chosen as a distinguished lecturer for the IEEE Signal Processing Society for the year 1996–1997. In 1999, he received the IEEE CAS Society's Golden Jubilee Medal, and in 2002, he received the IEEE Signal Processing Society's Technical Achievement Award.