LEAST-SQUARES DESIGN OF 2-D SPARSE NONSEPARABLE FILTER BANKS USING TRANSFORMATION OF VARIABLES: A GREEDY APPROACH

Bogdan C. Şicleru[1] *and Bogdan Dumitrescu*[1,2]

[1]Dept. of Automatic Control and Systems Engineering
Faculty of Automatic Control and Computers
Politehnica University of Bucharest
313 Spl. Independenţei, 060042 Bucharest, Romania

[2]Tampere Int. Center for Signal Processing
Tampere University of Technology
P.O.Box 553, SF-33101
Tampere, Finland

e-mail: bogdan.sicleru@acse.pub.ro, bogdan.dumitrescu@tut.fi

ABSTRACT

We present in this paper a design method for two-dimensional two-channel sparse nonseparable FIR filter banks using transformation of variables. In a first phase the transformation function is designed in a least-squares fashion. The second stage of the algorithm introduces the sparseness in a greedy manner by removing at each step the minimum coefficient of the filter in absolute value. The obtained results show that the sparse filter banks offer the possibility of choosing the complexity of the filters and have also better stopband energy.

***Index Terms*—** 2-D, sparse, filter bank, least-squares, greedy algorithm

1. INTRODUCTION

The problem of designing sparse filters is an issue of interest due to the focus on reducing the number of arithmetic operations and hence the costs of the implementation. The difficulty in obtaining sparsity [1] for digital filters is in maintaining their frequency specifications. The question that arises is which degrees of the filters should be left out. Setting small-valued coefficients to zero has been addressed in [2] to design sparse FIR filters. In [3] exhaustive intelligent techniques have been used. Efficient design using orthogonal matching pursuit was used in [4].

In this paper we propose the design of 2-D sparse nonseparable FIR filter banks (FBs) with quincunx [5] decimation using transformation of variables. Quincunx FBs with diamond passband [5] occupy a distinct place in signal processing due to their applications in image and video processing. A method for designing two-dimensional (2-D) perfect reconstruction (PR) FBs is through transformation of variables [6], method which allows the design of filters using polynomials of smaller degrees, thus reducing the number of coefficients. Successive thinning was recently used to study sparse filter design in one dimension [7], where the coefficients are eliminated so as the increase in error is minimized. This approach was found to be comparable with the removal of the smallest coefficient in each step. Two-dimensional sparse filters were designed in [8] using convex optimization. To the best of our knowledge this is the first attempt to design sparse FBs.

Our method uses least-squares optimization (see [9] for a 3-D non-sparse design) followed by a greedy search. The algorithm has four stages. In the first step we optimize the transformation function (TF) disregarding the properties of the filters. The second step attempts to minimize the stopband energy through a convex optimization problem. Next, in the third step we actually reach sparsity for the TF (which further gives a sparse FB) using a greedy approach where the coefficient chosen to be removed is always the minimum one. The optional last step is a nonlinear optimization of the 1-D filters used to build the 2-D filters.

We prove that our method gives filters with better stopband energy (than the non-sparse FBs) and also offers the possibility of a complexity/performance trade-off. Another advantage of our approach is that it allows the simple introduction of regularity constraints on the filters.

2. THE FILTER BANK

We aim to design FBs like the one in Figure 1, where the sampling is done on the quincunx lattice using the sampling matrix $\boldsymbol{D} = \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}$. Denoting $\boldsymbol{z} = (z_1, z_2)$ and $\boldsymbol{z}^{\boldsymbol{k}} = z_1^{k_1} z_2^{k_2}$, the filters are 2-D real symmetric trigonometric polynomials

$$H(\boldsymbol{z}) = \sum_{\boldsymbol{k}=-\boldsymbol{n}}^{\boldsymbol{n}} h_{\boldsymbol{k}} \boldsymbol{z}^{-\boldsymbol{k}}, \quad h_{\boldsymbol{k}} = h_{-\boldsymbol{k}}, \tag{1}$$

where $\boldsymbol{k} = (k_1, k_2) \in \mathbb{Z}^2$. We associate with a trigonometric polynomial $X(\boldsymbol{z})$ the vector $\boldsymbol{x}$ which contains the coefficients belonging to a halfspace.

This work was supported by the Romanian National Authority for Scientific Research, CNCS-UEFISCDI, project PN-II-ID-PCE-2011-3-0400.

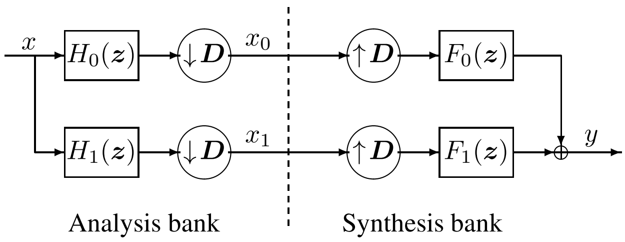


Fig. 1: Two-channel filter bank.

The lowpass filters are obtained using the transformation of variables technique [10], as

$$\begin{array}{rcl} H_0(\boldsymbol{z}) & = & H_T(M(\boldsymbol{z})) \\ F_0(\boldsymbol{z}) & = & F_T(M(\boldsymbol{z})) \end{array} \tag{2}$$

where $H_T(Z)$ and $F_T(Z)$ are 1-D filters in the real variable Z and the TF $M(\boldsymbol{z})$ is a 2-D filter as in (1) with

$$m_{\boldsymbol{k}} = 0, \quad \text{if } k_1 + k_2 \text{ is even.} \tag{3}$$

Note that in the light of (3), the support of a non-sparse TF contains only half of the coefficients of the filter. The aliasing transfer function of the FB is canceled by choosing the following highpass filters:

$$\begin{array}{rcl} H_1(z_1, z_2) & = & z_1^{-K_1} z_2^{-K_2} F_0(-z_1, -z_2) \\ F_1(z_1, z_2) & = & z_1^{K_1} z_2^{K_2} H_0(-z_1, -z_2) \end{array} \tag{4}$$

with $K_1 + K_2$ being odd.

For the purpose of illustration, we consider $H_T(Z)$ and $F_T(Z)$ [6] of degrees two and three, respectively (generalization is immediate), i.e. we have

$$\begin{array}{rcl} H_0(\boldsymbol{z}) & = & a_0 + a_1 M(\boldsymbol{z}) + a_2 M(\boldsymbol{z})^2 \\ F_0(\boldsymbol{z}) & = & b_0 + b_1 M(\boldsymbol{z}) + b_2 M(\boldsymbol{z})^2 + b_3 M(\boldsymbol{z})^3. \end{array} \tag{5}$$

Denoting $D_T(Z) = H_T(Z) F_T(Z)$, the PR condition is imposed on the FB with the constraint

$$D_T(Z) + D_T(-Z) = 1. \tag{6}$$

3. THE TRANSFORMATION FUNCTION

We describe in this section the method used to obtain the TF. We denote for brevity $H(\boldsymbol{\omega}) = H(\mathrm{e}^{j\boldsymbol{\omega}})$. To quantify the quality of a filter, we use the stopband energy, defined for a 2-D filter $H(\boldsymbol{z})$ as

$$E_s(H(\boldsymbol{z})) \triangleq \frac{1}{(2\pi)^2} \int_{V_s} |H(\boldsymbol{\omega})|^2 \, d\boldsymbol{\omega}, \tag{7}$$

where V_s is the stopband region. Furthermore (7) is equivalent to $E_s(H(\boldsymbol{z})) = \boldsymbol{h}^T \cdot \boldsymbol{C} \cdot \boldsymbol{h}$, where $\boldsymbol{C} = \boldsymbol{P}^T \bar{\boldsymbol{C}} \boldsymbol{P}$ is a positive definite matrix, with $\bar{\boldsymbol{C}} = \frac{1}{(2\pi)^2} \int_{V_s} \boldsymbol{C}_2(\omega_2) \otimes \boldsymbol{C}_1(\omega_1) \, d\omega_1 d\omega_2$, $\boldsymbol{C}_i = \mathrm{Toep}(\mathrm{e}^{j\omega_i n_i}, \ldots, 1, \ldots, \mathrm{e}^{-j\omega_i n_i})$, $i = 1:2$, and $\boldsymbol{P} = \begin{bmatrix} \boldsymbol{0} & 1 & \boldsymbol{0} \\ \boldsymbol{J} & \boldsymbol{0} & \boldsymbol{I} \end{bmatrix}^T$, where $\mathrm{Toep}(y_0, \ldots, y_n)$ is the Toeplitz matrix having y_k on the k-th diagonal, $\boldsymbol{J}$ is a counteridentity matrix of appropriate dimensions and the superscript ^{T} denotes transposition. Thus, the computation of the matrix $\boldsymbol{C}$ essentially reduces to solving the integral $I = \int_{V_s} \cos(k_1\omega_1 + k_2\omega_2) \, d\omega_1 d\omega_2$. (Due to space limitations we omit the computation of I.)

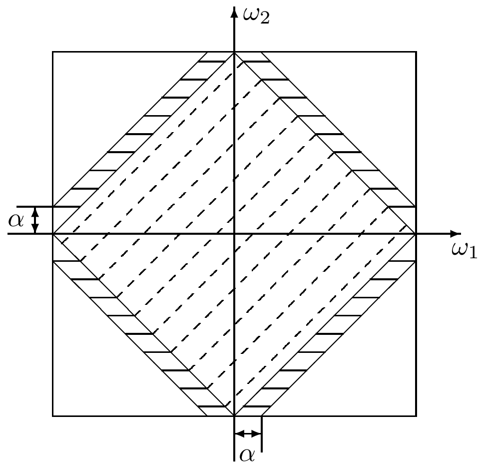


Fig. 2: The 2-D passband.

We quantify the width of the transition band with a parameter α, as shown in Figure 2 which presents the frequency bands. In white we have represented the stopband.

We describe in the next sections the algorithms used to optimize the support of the TF $M(\boldsymbol{z})$. We denote with $\mathcal{S}$ the support for the $M(\boldsymbol{z})$ filter, i.e. the set of pairs (k_1, k_2) for which m_{k_1,k_2} is nonzero. We also denote with $\mathcal{S}^{(CO)}$ the complete support in accordance with (3), i.e. the set of pairs (k_1, k_2) with $k_1 + k_2$ being an odd number, and with $\mathcal{S}^{(CU)}$ the current support of $M(\boldsymbol{z})$ at some point of the algorithm.

3.1. Non-sparse design

In the first step we optimize the TF $M(\boldsymbol{z})$ independently of the FB. When the variable Z goes from 0 to 1 the values of the polynomials $H_T(Z)$ and $F_T(Z)$ grow also from 0 to 1. Hence, in the stopband, the polynomial $\tilde{M}(\boldsymbol{z}) = 1 + M(\boldsymbol{z})$ should approximate 0. As such, we optimize the stopband energy using (7) and taking (3) into account, by solving the optimization problem

$$\begin{array}{rcl} \tilde{\boldsymbol{m}}^\star & = & \min\limits_{\tilde{\boldsymbol{m}}} \quad \tilde{\boldsymbol{m}}^T \boldsymbol{C} \tilde{\boldsymbol{m}} \\ & & \text{s.t.} \quad \tilde{M}(\boldsymbol{z}) = 1 + M(\boldsymbol{z}) \\ & & \qquad \mathcal{S} = \mathcal{S}^{(CU)} \end{array} \tag{8}$$

This is the minimization of a convex quadratic subject to linear constraints. Denoting the equality constraints as

$\mathbf{\Omega}\tilde{\boldsymbol{m}} = \boldsymbol{\gamma}$ the solution of (8) is $\tilde{\boldsymbol{m}}^{\star} = -\frac{1}{2}\boldsymbol{C}^{-1}\mathbf{\Omega}^{T}\boldsymbol{\tau}$, with $\mathbf{\Omega}\boldsymbol{C}^{-1}\mathbf{\Omega}^{T}\boldsymbol{\tau} = -2\boldsymbol{\gamma}$. For the non-sparse design the current support is the complete one. Hence, we use $\mathcal{S}^{(CU)} = \mathcal{S}^{(CO)}$.

In the second step, we aim to attain the actual optimization purpose, which is the minimization of the stopband energy of the filters (2), for given 1-D filters $H_T(Z)$ and $F_T(Z)$. We solve the optimization problem

$$\min_{M(\boldsymbol{z})} \quad \mathcal{E}_s(H_T(M(\boldsymbol{z})), F_T(M(\boldsymbol{z}))), \tag{9}$$

with $\mathcal{E}_s(G_1(\boldsymbol{z}), G_2(\boldsymbol{z})) \triangleq \lambda E_s(G_1(\boldsymbol{z})) + (1-\lambda)E_s(G_2(\boldsymbol{z}))$, where the weight λ is known. The stopband energies that appear in (9) are nonconvex functions in the variable $M(\boldsymbol{z})$. To make the problem tractable, we employ the following trick. Let $M_0(\boldsymbol{z})$ be the TF designed using (8). Since this should be a relatively good approximation of the optimal TF, we optimize instead of (5) the filters from (10), which can be found on the top of the next page. We obtain the optimization problem

$$\boldsymbol{m}^{\star} \quad = \quad \min_{M(\boldsymbol{z})} \quad \mathcal{E}_s(\tilde{H}_0(\boldsymbol{z})), \tilde{F}_0(\boldsymbol{z})). \tag{11}$$

This is a convex quadratic in the coefficients of $M(\boldsymbol{z})$. The problem (11) can be written as [9] $\boldsymbol{m}^{\star} = \min_{\boldsymbol{m}} \boldsymbol{m}^T\mathbf{\Theta}\boldsymbol{m} + 2\boldsymbol{\eta}\boldsymbol{m}$, with unique solution $\boldsymbol{m}^{\star} = -\mathbf{\Theta}^{-1}\boldsymbol{\eta}^T$. Note that we do not iterate problem (11) (using $\beta M^{*}(\boldsymbol{z}) + (1-\beta)M_0(\boldsymbol{z})$, $\beta \in (0, 1]$ as $M_0(\boldsymbol{z})$ for the next step) because the stopband energy does not improve.

3.2. Sparse design

In the third step we focus on designing sparse FBs by creating a sparse TF, using a modified version of (11) in a greedy algorithm. After obtaining the non-sparse TF using (8), call it $M_0^{\star}(\boldsymbol{z})$, we replace the minimum coefficient (in absolute value) with zero and using this new TF we solve problem (11) with the current support; note that introducing one zero in the TF actually creates four zeros, due to the fact that the filters are 4-symmetric, so the decrement step for the number of nonzero coefficients for a filter is four. These steps are repeated until the size of the support is satisfactory. Note that the optimization step with the new support does not modify the size of the support.

Algorithm 3.1: SPARSETF($\boldsymbol{n}, \alpha, N_d$)

$$\begin{cases} i \leftarrow 0 \\ \textbf{while } N_i \triangleq \text{nnz}(\boldsymbol{m}_i^{\star})^a > N_d \\ \quad \textbf{do} \begin{cases} \text{set the minimum coefficient of } \boldsymbol{m}_i^{\star} \text{ to zero} \\ \tilde{H}_i(\boldsymbol{z}) \triangleq \mathcal{H}(M(\boldsymbol{z}), M_i^{\star}(\boldsymbol{z})) \\ \tilde{F}_i(\boldsymbol{z}) \triangleq \mathcal{F}(M(\boldsymbol{z}), M_i^{\star}(\boldsymbol{z})) \\ i \leftarrow i+1 \\ \boldsymbol{m}_i^{\star} \leftarrow \operatorname*{argmin}_{M(\boldsymbol{z})} \mathcal{E}_i \triangleq \mathcal{E}_s(\tilde{H}_{i-1}(\boldsymbol{z}), \tilde{F}_{i-1}(\boldsymbol{z})) \end{cases} \\ \textbf{return } (\boldsymbol{m}_i^{\star}) \end{cases}$$

The steps of the greedy algorithm are summarized in Algorithm 3.1. N_d is the maximum desired number of nonzero coefficients for the TF. card($\cdot$) is the function that returns the cardinal of a set and the function nnz($\cdot$) returns the number of nonzero elements of its argument.

4. OPTIMIZATION OF THE 1-D FILTERS

In the fourth step, our results can be further improved by optimizing the 1-D filters. Let $\overline{M}(\boldsymbol{z})$ be the solution of the problem (11), considered now fixed. Let $\boldsymbol{a} = [a_0\ a_1\ a_2]^T$ and $\boldsymbol{b} = [b_0\ b_1\ b_2\ b_3]^T$ be the coefficients of the 1-D filters $H_T(Z)$ and $F_T(Z)$, respectively, that appear in (5). Considering these vectors as variables, we solve the optimization problem

$$\begin{aligned} \min_{\boldsymbol{a},\boldsymbol{b}} \quad & \mathcal{E}_s(H_T(\overline{M}(\boldsymbol{z})), F_T(\overline{M}(\boldsymbol{z}))) \\ \text{s.t.} \quad & (6) \end{aligned} \tag{12}$$

Although the criterion is a convex quadratic, the constraint (6) is bilinear, hence the problem (12) is not convex.

In order to obtain a sparse FB we use the solution of the greedy algorithm for $\overline{M}(\boldsymbol{z})$.

5. THE REGULARITY CONDITIONS

Our method provides the possibility of easily adding regularity conditions. According to e.g. [11] in order to achieve Q order regularity for the lowpass filters one must impose S order regularity on the TF $M(\boldsymbol{z})$ and/or R order regularity on both the 1-D filters $H_T(Z)$ and $F_T(Z)$. We describe next the first case. (See Table 1 from [11] for the connection between Q, S and R.)

For an order S of zero derivatives the conditions on the filter $M(\boldsymbol{z})$ are $\left.\frac{\partial^L M(\omega_1,\omega_2)}{\partial\omega_1^{\ell_1}\partial\omega_2^{\ell_2}}\right|_{\omega_1=\omega_2=\pi} = 0$, $\forall L = 1 : S$, $\forall \ell_1, \ell_2,\ \ell_1 + \ell_2 = L$. We also require that $M(\pi, \pi) = -1$. To design a TF, we embed the equality constraints

$$\mathbf{\Upsilon}\boldsymbol{m} \quad = \quad \boldsymbol{\zeta} \tag{13}$$

over $M(\boldsymbol{z})$, in the optimization problem (11). Thus we obtain the problem

$$\begin{aligned} \boldsymbol{m}_R^{\star} \quad = \quad \min_{\boldsymbol{m}} \quad & \boldsymbol{m}^T\mathbf{\Theta}\boldsymbol{m} + 2\boldsymbol{\eta}\boldsymbol{m} \\ \text{s.t.} \quad & (13) \end{aligned} \tag{14}$$

The solution of the problem (14) is $\boldsymbol{m}_R^{\star} = -\mathbf{\Theta}^{-1}\boldsymbol{\eta}^T - \frac{1}{2}\mathbf{\Theta}^{-1}\mathbf{\Upsilon}^T\boldsymbol{\xi}$, where $\mathbf{\Upsilon}\mathbf{\Theta}^{-1}\mathbf{\Upsilon}^T\boldsymbol{\xi} = -2\mathbf{\Upsilon}\mathbf{\Theta}^{-1}\boldsymbol{\eta}^T - 2\boldsymbol{\zeta}$.

In order to design sparse FBs with regularity conditions we design a sparse TF using the Algorithm 3.1; we denote this TF as $\hat{M}(\boldsymbol{z})$. Using $\hat{M}(\boldsymbol{z})$ we solve the optimization problem

$$\begin{aligned} \hat{\boldsymbol{m}}_R^{\star} \quad = \quad \min_{M(\boldsymbol{z})} \quad & \mathcal{E}_s(\hat{H}(\boldsymbol{z}), \hat{F}(\boldsymbol{z})) \\ \text{s.t.} \quad & (13) \end{aligned} \tag{15}$$

$$\begin{array}{rcl}\tilde{H}_0(\boldsymbol{z}) & = & \mathcal{H}(M(\boldsymbol{z}), M_0(\boldsymbol{z})) \triangleq a_0 + a_1 M(\boldsymbol{z}) + a_2 M_0(\boldsymbol{z}) M(\boldsymbol{z}) \\ \tilde{F}_0(\boldsymbol{z}) & = & \mathcal{F}(M(\boldsymbol{z}), M_0(\boldsymbol{z})) \triangleq b_0 + b_1 M(\boldsymbol{z}) + b_2 M_0(\boldsymbol{z}) M(\boldsymbol{z}) + b_3 M_0(\boldsymbol{z})^2 M(\boldsymbol{z}).\end{array} \quad (10)$$

where

$$\begin{array}{rcl}\hat{H}(\boldsymbol{z}) & = & \mathcal{H}(M(\boldsymbol{z}), \hat{M}(\boldsymbol{z})) \\ \hat{F}(\boldsymbol{z}) & = & \mathcal{F}(M(\boldsymbol{z}), \hat{M}(\boldsymbol{z})).\end{array} \quad (16)$$

The 1-D filters have R order zeros in -1 if $H_T^{(i)}(-1) = 0$, $F_T^{(i)}(-1) = 0, \forall\, i = 0 : R$, constraints which are equivalent to imposing factors of the form $(1 + Z)^{R+1}$ on the 1-D filters [9].

6. RESULTS

We consider designing sparse FBs using a TF of degree $\boldsymbol{n} = [\ 7\ 7\ ]$ which is a filter of size 15×15. We use the following 1-D filters [6]

$$\begin{array}{rcl}H_T(Z) & = & -\frac{1}{4}(Z+1)(Z-3) \\ F_T(Z) & = & -\frac{1}{12}(Z+1)(Z^2+Z-8).\end{array} \quad (17)$$

(It can be easily checked that the filters from (17) satisfy the PR condition (6).) We have made tests using $\alpha = [0.10, 0.15, 0.20]\pi$ and we have considered $\lambda = 0.5$ in all cases.

We discuss first the non-sparse case. Table 1 presents a comparison between several algorithms. Remind that (11) and (12) represent the third and the fourth stages of our algorithm, respectively. The nonlinear optimization in (12) gives an advantage over the convex optimization problem (11). Comparing the results provided by (11) with the window method from [11], our approach performs slightly better; taking e.g. $\boldsymbol{n} = [\ 7\ 7\ ]$, $\alpha = 0.1\pi$, the window method gives a stopband energy of 0.0004294. In the problem (14) we introduce the regularity constraints, reason for which the stopband energy increases. (To solve the problem (12) we have used the `fmincon` function from MATLAB.)

We analyze now sparse FBs. We denote with N the number of nonzero coefficients for a filter. The decrease of the number of coefficients is made using Algorithm 3.1. We show in Figure 3 a comparison for the number of nonzero coefficients between the sparse filters versus the non-sparse ones. When comparing filters with the same number of nonzero coefficients one can see that the sparse filters obtained have a smaller stopband energy than the non-sparse filters obtained using (11), in most cases. Moreover, Algorithm 3.1 offers the possibility of selecting the desired number of nonzero coefficients for the TF. We present in Table 2 the stopband energies for different algorithms for the case of sparse filters with $\alpha = 0.1\pi$, starting from $\boldsymbol{n} = [\ 7\ 7\ ]$.

We discuss now several aspects of our algorithm. All the results presented onwards use $\alpha = 0.2\pi$. The tests start with a non-sparse TF of degree $\boldsymbol{n} = [\ 7\ 7\ ]$, i.e. $\mathcal{S}^{(CO)} = 112$, and stop when reaching $N = 80$.

We note that in Algorithm 3.1, we do not re-optimize the new TF with (8), before re-optimizing the stopband energy $\mathcal{E}_i$ with the new support. The reason is because re-optimizing with (8) produces higher stopband energies than our algorithm. Applying Algorithm 3.1 we have obtained $\mathcal{E}_8 = 0.0000176$, while introducing the optimization with (8) would give $\mathcal{E}_8 = 0.0000408$.

We stress now the importance of the greedy approach. Our algorithm sets the minimum coefficient to zero in each step, hence after i steps we have removed i unique coefficients. On the other hand, if we were to remove the first i smallest coefficients the results would be less satisfying. As an example, removing the 8 smallest (unique) coefficients from the start, would give a stopband energy equal to 0.0000258.

Another variant of our approach would be to remove at each step the coefficient that gives the highest stopband energy if kept. The drawback of such an approach is that it is much too slow, the search for the right element to remove being of complexity $\mathcal{O}(n^2)$ rather than just $\mathcal{O}(n)$, which is the case of removing the minimum coefficient, where we have considered n to be the total number of coefficients of the filter. As such, while running Algorithm 3.1 takes about 12 minutes, the minimum stopband energy algorithm takes roughly 6 hours on a PC with Intel Core 2 Duo at 2.33 GHz and 2 GB of RAM. Moreover, the improvement over the minimum coefficient algorithm is insignificant; with this approach we have obtained a stopband energy of 0.0000175.

7. CONCLUSIONS

We have proposed a new method for designing 2-D sparse FBs with a diamond passband with the aid of transformation of variables. Our approach reduces the initial problem to designing filters with minimum stopband energy. The sparseness of the filters is obtained in a greedy fashion by removing at each step the minimum coefficient (in absolute value) and therefore controlling the complexity of the filters. The sparse filters thus obtained offer FBs with smaller stopband energy than the non-sparse FBs with the same number of nonzero coefficients. Furthermore, using our method one can easily introduce regularity constraints in the design problem.

8. REFERENCES

[1] M. Elad, *Sparse and redundant representations*, Springer, 2010.

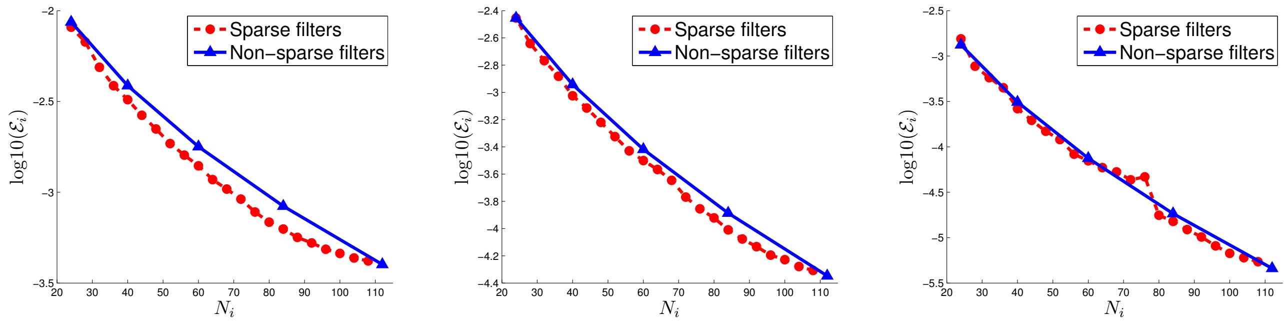
[2] M. Smith and D. Farden, "Thinning the impulse response of FIR digital filters," in *Proc. IEEE Int. Conf.*

Table 1: Stopband energies using different algorithms; $\boldsymbol{n} = [\,7\;7\,]$.

α/π	Problem (11)	Problem (12)	Problem (14), $S=3$	Problem (14), $S=5$
0.10	0.0004001	0.000371102	0.0004812	0.0006211
0.15	0.0000449	0.000042757	0.0000585	0.0000839
0.20	0.0000045	0.000004355	0.0000064	0.0000102

Table 2: Stopband energies for sparse filter banks; $\boldsymbol{n} = [\,7\;7\,]$, $\alpha = 0.1\pi$.

N	Algorithm 3.1	Optim. of $H_T(Z), F_T(Z)$	Regularity, $S=3$	Regularity, $S=5$
108	0.0004185	0.0003889	0.0004888	0.0006225
88	0.0005638	0.0005250	0.0005908	0.0007070
68	0.0010414	0.0009711	0.0010547	0.0012002
48	0.0022304	0.0020957	0.0026067	0.0027723
28	0.0067229	0.0063201	0.0076859	0.0097672



(a) $\boldsymbol{n} = [\,7\;7\,]$, $\alpha = 0.1\pi$. (b) $\boldsymbol{n} = [\,7\;7\,]$, $\alpha = 0.15\pi$. (c) $\boldsymbol{n} = [\,7\;7\,]$, $\alpha = 0.2\pi$.

Fig. 3: Number of nonzero coefficients comparison for the TF.

Acc. Speech and Sign. Proc. (ICASSP), 1981, pp. 240–242.

[3] Y.S. Song and Y.H. Lee, "Design of sparse FIR filters based on branch and bound algorithm," in *Proc. 40th Midwest Symp. Circ. Syst.*, 1997, vol. 2, pp. 1445–1448.

[4] D. Mattera, F. Palmieri, and S. Haykin, "Efficient sparse FIR filter design," in *Proc. Int. Conf. Acc. Speach and Sign. Proc. (ICASSP)*, 2002, vol. 2, pp. 1537–1540.

[5] W.S. Lu, A. Antoniou, and H. Xu, "A direct method for the design of 2-D nonseparable filter banks," *IEEE Trans. Circ. Syst. II*, vol. 45, no. 8, pp. 1146–1150, 1998.

[6] D.B.H. Tay and N.G. Kingsbury, "Flexible design of multidimensional perfect reconstruction FIR 2-band filters using transformations of variables," *IEEE Trans. Image Proc.*, vol. 2, no. 4, pp. 466–480, October 1993.

[7] T. Baran, D. Wei, and A.V. Oppenheim, "Linear programming algorithms for sparse filter design," *IEEE Trans. Signal Proc.*, vol. 58, no. 3, pp. 1605–1617, 2010.

[8] W.S. Lu and T. Hinamoto, "Two-dimensional digital filters with sparse coefficients," *Multidim. Syst. Sign. Proc.*, vol. 22, no. 1–3, pp. 173–189, 2010.

[9] B.C. Şicleru and B. Dumitrescu, "Least squares design of three-dimensional filter banks using transformation of variables," in *Proc. Int. Conf. Acc. Speech and Sign. Proc. (ICASSP)*, Dallas, Texas, USA, March 2010, pp. 3682–3685.

[10] D.B.H. Tay, "Design of filter banks/wavelets using TROV: a survey," *Digital Sign. Proc.*, vol. 7, pp. 229–238, 1997.

[11] D.B.H. Tay and N.G. Kingsbury, "Design of nonseparable 3-D filter banks/wavelet bases using transformations of variables," in *IEE Proc. Vis. Image Sign. Proc.*, February 1996, vol. 143, pp. 51–61.