[10] R. D. Koilpillai and P. P. Vaidyanathan. Cosine-Modulated Fir Filter Banks Satisfying Perfect Reconstruction. *IEEE Trans. in SP*, pages 770–783, April 1992.

[11] H. S. Malvar. Modulated QMF Filter Banks With Perfect Reconstruction. *Electronics Letters*, 26:906–907, June 1990.

[12] K. Nayebi, T.P. Barnwell, and M.J.T.Smith. Analysis/synthesis Systems with Time-varying Filter Bank Structures. In *Proceedings of ICASSP '92, San Francisco, CA*, pages 617–620, March 1992.

[13] K. Nayebi, T. Barnwell III, and M. J. T. Smith. The Time Domain Analysis And Design Of Exactly Reconstructing FIR Analysis/Synthesis Filter Banks. *IEEE Trans. in ASSP*, 1991.

[14] A. K. Soman, P. P. Vaidyanathan, and T. Q. Nguyen. Linear Phase Paraunitary Filter Banks : Theory, Factorizations and Applications. Technical report, California Institute of Technology, May 1992.

[15] P. P. Vaidyanathan. Improved Technique for The Design Of Perfect Reconstruction FIR QMF Banks With Lossless Polyphase Matrices. *IEEE Trans. in ASSP*, 37(7):1042–1056, 1989.

Acknowledgement

The authors would like to thank Jan Odegard and Ivan Selesnick of the DSP group at Rice for helpful suggestions and comments.

References

[1] I. Daubechies. Orthonormal Bases Of Compactly Supported Wavelets. *Communications in Pure and Applied Mathematics*, (41):909–996, 1988.

[2] I. Daubechies. *Ten Lectures on Wavelets*. SIAM, 1992. CBMS Lecture Series.

[3] R. A. Gopinath. *Wavelet And Filter Banks - New Results And Applications*. PhD thesis, Rice University, Houston, TX-77251, August 1992.

[4] R. A. Gopinath and C. S. Burrus. Wavelets and Filter Banks. Technical Report CML TR91-20, Computational Mathematics Laboratory, Rice University, September 1991. Appears in "Wavelets - A Tutorial in Theory and Applications", edited by C.K.Chui, published by Academic Press.

[5] R. A. Gopinath and C. S. Burrus. On Cosine-Modulated Wavelet Orthonormal Bases. Technical Report CML TR92-06, Computational Mathematics Laboratory, Rice University, Mar 1992. IEEE Trans. in Image Processing, to appear.

[6] R. A. Gopinath and C. S. Burrus. Theory Of Modulated Filter Banks And Modulated Wavelet Tight Frames. Technical Report CML TR92-18, August 1992. Submitted to *Applied and Computational Harmonic Analysis: Wavelets and Signal Processing.*

[7] R. A. Gopinath and C. S. Burrus. Unitary Fir Filter Banks and Symmetry. Technical Report CML TR92-17, August 1992. Submitted to IEEE Trans. in CAS II.

[8] C. Herley, J. Kovacevic, K. Ramachandran, and M. Vetterli. Time-varying Orthonormal Tilings of the Time-frequency Plane. Technical Report CU/CTR/TR 315-92-25, Columbia University, 1992.

[9] C. Herley, J. Kovacevic, K. Ramachandran, and M. Vetterli. Time-varying Orthonormal Tilings of the Time-frequency Plane. In *Proceedings of IEEE-SP Intl. Symposium on Time-Frequency and Time-Scale Analysis, Victoria, British Columbia*. IEEE, October 1992.

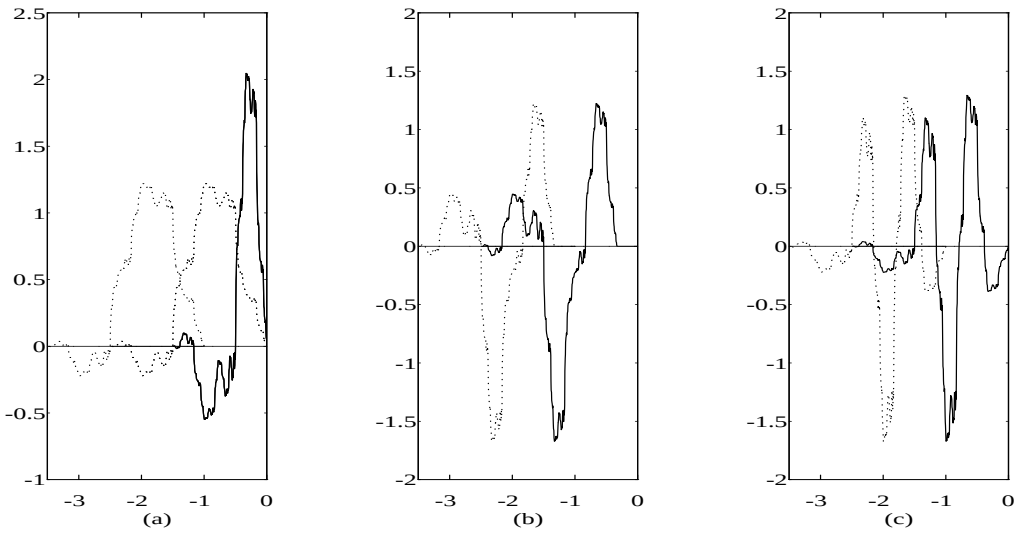

Figure 7: (a) Exit function $\nu_0(t)$, $\psi_0(t)$ and $\psi_0(t-1)$ (b) Wavelet $\psi_1(t)$ and $\psi_1(t-1)$ (c) Wavelet $\psi_2(t)$ and $\psi_2(t-1)$

wavelets, and their corresponding entry and exit functions. For instance Fig. 7 and Fig. 6 together correspond to a wavelet basis for $L^2(\mathbb{R})$, where a 3-band wavelet basis with length 6 filters is used for $t < 0$ and a 2-band wavelet basis with length 4 filters is used for $t > 0$. Certainly a degree of overlap between the exit functions on the left of a transition and entry functions on the right of the transition can be obtained by merely changing coordinates in the finite dimensional space corresponding to these functions.

8 Conclusion

Arbitrarily complex time-varying PR filter bank trees can be designed and implemented efficiently. The fundamental idea is to design time-varying PR building blocks that can then be used appropriately to form more complicated structures. The implementation of these building blocks being elementary, one has a simple implementation for the more complex time-varying trees (divide-and-conquer?). A new concept of *soft switching* between filter bank trees is also introduced. These time-varying filter banks are then used to construct time-varying wavelet bases for the real line. All the constructions are optimal in the sense that one cannot change between unitary filter bank topologies more rapidly without giving up PR. Matlab programs for generating a set of canonical entry/exit filters of a filter bank, and the entry/exit functions of a given wavelet basis can be obtained from the authors upon request.

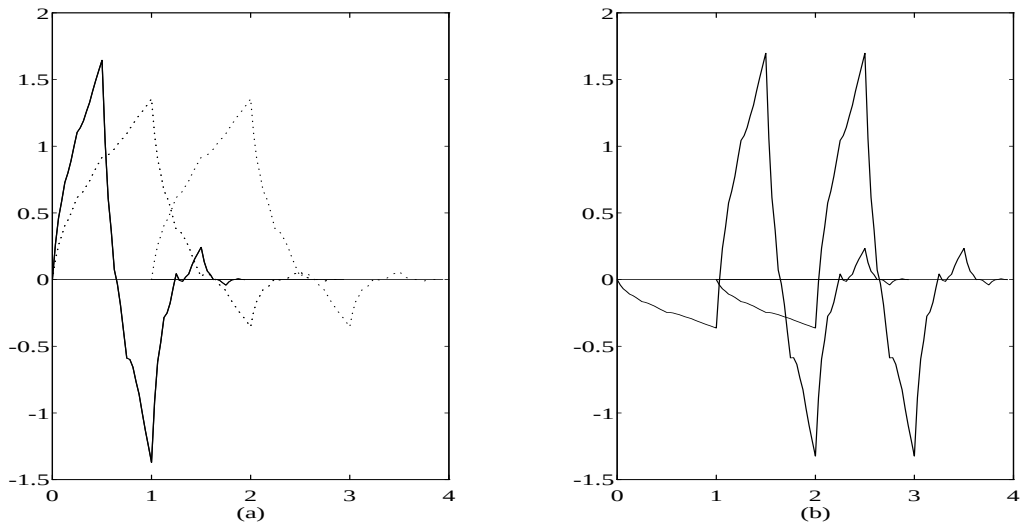

Figure 6: (a) Entry function $\chi_0(t)$, $\psi_0(t)$ and $\psi_0(t-1)$ (b) Wavelet $\psi_1(t)$ and $\psi_1(t-1)$

The functions are supported in $[-\frac{N-1}{M-1}, 0]$. Notice that in this case the scaling and wavelet vectors correspond to filters in the analysis filter bank. Let Δ be the McMillan degree of this filter bank. The spaces $W_{i,j}$ defined as in Eqn. 49 form the multiresolution analysis. Let $\{w_i\}$ be the exit filters of this filter bank. Define the *exit* functions

$$\nu_i(t) = \sqrt{M} \sum_{k=0}^{L_i-1} w_i(k)\psi_0(Mt+k), \tag{56}$$

$\mathcal{W}_j = Span\left\{M^{j/2}\nu_i(M^j t)\right\}$, and $W_{i,j}^{-} = Span\left\{\psi_{i,j,k}\right\} \stackrel{\text{def}}{=} \left\{M^{j/2}\psi_i(M^j t+k)\right\}$ for $k \in \mathbb{N}$. Then as $j \to \infty$, $W_{0,j}^{-} \to L^2((-\infty, 0])$ and

$$\{\psi_{i,j,k} \mid i \in \{1,\ldots,M-1\}, j \in \mathbf{Z}, k \in \mathbb{N}\} \cup \{\nu_{i,j} \mid i \in \{0,\ldots,\Delta-1\}, j \in \mathbf{Z}\}$$

forms a WTF for the square integrable functions on the left half line. Orthonormality of this basis is equivalent to the orthonormality of its parent basis on the line. An example with one exit function (corresponding to $M = 3$, $N = 6$) Type 1 modulated WTF obtained earlier is given in Fig. 7.

7.3 Segmented Time-Varying Wavelet Bases

Using the ideas above one can construct wavelet bases for the interval and consequently segmented wavelet bases for $L^2(\mathbb{R})$. One does the obvious - cut the real line into segments and use different wavelet bases on each segment. Each segment will be spanned by a combination of scaling functions,

vectors *considered as the filters of a synthesis unitary filter bank*. We prove this deep result (from a wavelets perspective) using the simple results in filter bank theory developed so far. First construct the entry filters of the filter bank with analysis filters given by the time reverses of the scaling and wavelet vectors. Let $u_i(n)$ denote the (synthesis) entry filters of the filter bank (i.e., these are time reverses of the analysis entry filters). Let Δ be the McMillan degree and define the *entry* functions $\mu_i(t)$, $i \in \{0, \ldots, \Delta - 1\}$ as follows:

$$\mu_l(t) = \sqrt{M} \sum_{k=0}^{L_l - 1} u_l(k) \psi_0(Mt - k). \tag{52}$$

where u_i are the entry filters of the filter bank (i.e., the rows of $\mathbf{U}$). $\mu_i(t)$ is supported in $[0, \frac{L_l - 1}{M} + \frac{N-1}{M-1}]$. Let $\mathcal{U} = Span\{\mu_{l,j}\} \stackrel{\text{def}}{=} Span\left\{M^{j/2}\mu_i(M^j t)\right\}$. Then (by considering one stage of the analysis and synthesis of the corresponding PR filter bank on right sided signals) it readily follows that Eqn. 51 holds giving a multiresolution analysis for $L^2([0, \infty))$. Moreover, the set of functions

$$\{\psi_{i,j,k} \mid i \in \{1, \ldots, M-1\}, j \in \mathbf{Z}, k \in \mathbb{N}\} \cup \{\mu_{i,j} \mid i \in \{0, \ldots, \Delta - 1\}, j \in \mathbf{Z}\}$$

form a wavelet tight frame. If one began with an orthonormal basis for the line, the newly constructed basis for the half line is also orthonormal. Indeed if $\{\psi_0(t-k)\}$ is an orthonormal system

$$\int_{t \geq 0} \mu_l(t) \psi_i(t - n) = \sum_k u_l(k) h_i(Ml + k) = 0, \tag{53}$$

and

$$\int_{t \geq 0} \mu_l(t) \mu_m(t) = \sum_k u_l(k) u_m(k) = 0. \tag{54}$$

There are precisely as many entry functions as there are entry filters and supports of these functions are explicitly given in terms of the lengths of the corresponding entry filters. Fig. 6 shows the scaling function, wavelet, their integer translates and the single entry function corresponding to Daubechies four coefficient wavelets. In this case, $u_0 = \left\{-\sqrt{3}/2, 1/2\right\}$.

7.2 Wavelet Bases for $L^2((-\infty, 0])$

One way to do this would be to start with a wavelet basis for $L^2([0, \infty))$ and reflect all the functions about the origin. This is effectively equivalent to switching the analysis and synthesis filters of a filter bank. However we give an independent development. We start with a WTF for $L^2(\mathbb{R})$, the functions $\psi_i(t)$ of which are defined by

$$\psi_i(t) = \sqrt{M} \sum_{k=0}^{N-1} h_i(k) \psi_0(Mt + k). \tag{55}$$

economical continuous-time time-varying wavelet bases based on infinite iterates of time-varying filter banks.

7.1 Wavelet Bases for $L^2([0,\infty))$

Given an M channel unitary filter bank with *causal synthesis* filters $\{h_i(n)\}$ if $\sum_n h_0(n) = \sqrt{M}$ it is well known that the filter bank gives rise to a wavelet tight frame for $L^2(\mathbb{R})$ with $(M-1)$ wavelets $\psi_i(t)$, $i \in \{1,\ldots,M-1\}$ and one scaling function $\psi_0(t)$. These functions are related to the (synthesis) filters in the filter bank (time-reverses of the analysis filters) as follows:

$$\psi_i(t) = \sqrt{M} \sum_{k=0}^{N-1} h_i(k)\psi_0(Mt-k). \tag{48}$$

The sequences $\{h_i(n)\}$ are called wavelet vectors for $i > 0$ and the scaling vector for $i = 0$. The scaling function and wavelets are supported in $[0, \frac{N-1}{M-1}]$, where N is the length of these scaling and wavelet vectors. If we define the spaces $W_{i,j}$ as

$$W_{i,j} = Span\,\{\psi_{i,j,k}\} \stackrel{\text{def}}{=} \left\{M^{j/2}\psi_i(M^j t-k)\right\} \quad \text{for } k \in \mathbf{Z} \tag{49}$$

then $W_{0,j}$ forms a multiresolution analysis of $L^2(\mathbb{R})$, with the relationship

$$W_{0,j} = W_{0,j-1} \oplus W_{1,j-1} \ldots \oplus W_{M-1,j-1}. \tag{50}$$

Several authors have attempted to construct wavelet bases for the half-line and the interval [8, 9, 2]. In [1] there is an outline of an approach which essentially considers the projection of $W_{0,j}$ onto $W_{0,j}^{half}$ defined to be the space spanned by the restrictions of $\psi_{0,j,k}(t)$ to the right half line. We give an entirely different solution to the problem and is based on the following idea. In Eqn. 49 if we restrict $k \in \mathbb{N}$, then all the functions are supported in $[0,\infty)$. Let the corresponding spaces to $W_{i,j}^+$. Then it is clear that as $j \to \infty$, if $W_{0,j} \to L^2(\mathbb{R})$, then $W_{0,j}^+ \to L^2([0,\infty))$. Therefore if we can have a multiresolution analysis for $W_{0,j}^+$, then we would have a wavelet basis for $L^2(\mathbb{R})$. Unfortunately, with these new spaces there is no multiresolution analysis equivalent to Eqn. 50. The problem is precisely that $W_{0,j}^+$ is *bigger* than the direct sum of the constituents at the next coarser scale. More specifically, it is trivially seen that $W_{i,j-1}^+ \subset W_{0,j}^+$. Define $\mathcal{U}_{j-1}$ as follows:

$$W_{0,j}^+ = W_{0,j-1}^+ \oplus W_{1,j-1}^+ \ldots \oplus W_{M-1,j-1}^+ \oplus \mathcal{U}_{j-1} \tag{51}$$

If we can find an orthonormal basis for $\mathcal{U}_j$, then we have a multiresolution analysis for $L^2([0,\infty))$.

It turns out that the space $\mathcal{U}_j$ is finite dimensional. The exact dimension of this space is given by the McMillan degree of the polyphase component matrix of the scaling and wavelet

The entire approach *does not rely* on any special structure of the factorization. The two crucial features are:

1. The factorization is minimal - this ensures canonical exit filters.

2. The time-reversed factorization is minimal - this ensures canonical entry filters.

In summary we have proved the following fundamental theorem, the intuitive elegance of which is patent.

Theorem 4 *Given any minimal cascade realization/implementation of any type (general, modulated etc.) of unitary filter bank, such that its time-reversed implementation is also minimal, one has an implementation of canonical entry and exit filters for switching into and out of this filter bank. Without loss of generality the cascade can be considered to be made up of first order polynomial factors of the form* $a + z^{-1}b$ *and this partitions the entry and exit filters of the filter bank into as many sets as there are factors. The entry filters corresponding to each factor is given by the projection of the effective filter up to that factor onto the row space of* a*. The exit filters are given by the entries of the storage elements of each factor instantaneously at the time of switch. The total number of exit filters is given by* Δ*, the McMillan degree of the implementation, and the total number of entry filters is given by* $(N - M - \Delta)$*, the McMillan degree of the time-reversed implementation.*

An important consequence of the theorem is that one can start with any desirable implementation of a filter bank, and *effectively compute outputs of canonical entry and exit filters without ever knowing what the filters are.* One might wonder if it is ever important to know what the entry/exit filters are. In the next section on time-varying wavelet bases, it turns out, one requires knowledge of the entry/exit filters to define certain transition functions for these bases.

7 On Generating Time-Varying Wavelet Bases

By taking of the effective input/output map of an arbitrary unitary time-varying filter bank tree filter bank tree one readily obtains time-varying discrete-time wavelet bases. Because of the ability to switch between filter bank trees, we have such bases for one-sided and finite signals. The orthonormality of these bases is a consequence of the unitary building blocks they are built from. We now consider the case of continuous-time time varying wavelet bases that are of interest in the question of arbitrarily tiling the time-frequency plane [8, 9]. In this section we give the most

Remark: In all cases notice that the sum of the McMillan degrees adds up to $N - M$. Also Theorem 3 gives a characterization of the number of entry and exit filters for canonical MFBs. This is (a remarkable fact) in sharp contrast to the case of general unitary filter banks where a priori one cannot say anything about the number of entry and exit filters unless one is given the filters of the filter bank.

It is well known that MFBs are easy to design and implement. However, the reduction in parameters by going from a general unitary filter bank to an MFB (with comparable McMillan degree) is obviously enormous.

6.2 Unitary FIR Filter Banks with Symmetry

Unitary FIR filter banks with linear phase filters were first parameterized by Soman and Vaidyanathan [14]. Recently, an alternate proof of this result along with results for unitary filter banks with various other types of symmetry restrictions have been parameterized (using factorizations) [7]. Since all of these are unitary filter banks, they can be factored as in Eqn. 38. In this case, each of the factors are structured. The entry filters for such filter banks are essentially the outputs of each first degree factor when the storage elements of the factor are zero. One appropriately projects these outputs to correspond to full rank. Again the exit filters for these filter banks are the entries of the storage elements of each factor at the point of switching. For lack of space we do not give examples of such filter banks. Instead, we proceed to show the general unitary filter bank switching theorem.

6.3 The General Unitary Time-Varying Filter Bank Theorem

There are many unitary filter bank structures, general unitary filter banks, MFBs, unitary filter banks with symmetry etc. In all the cases, one can factor the matrix $H_p(z)$ into first degree polynomial matrices of the form $[I - f + z^{-1}f]$. Depending on the type of unitary filter bank these first order factors exhibit structure. For example we saw the corresponding structure for Type 1 and 2 MFBs. This induces a corresponding time-domain factorization of $\breve{\mathbf{H}}$. For the most general time-varying filter bank tree problem, it is sufficient to know the entry and exit filters of the constituent factors of the filter bank being pruned/added. The number of exit filters for each constituent factor is the dimension of the nullspace of f (or the McMillan degree of the factor, say δ), while the number of entry filters is the codimension $(M - \delta)$. The total number of entry/exit filters of the filter bank will be the sum of all these entry and exit filters of the factors. By soft switching one can share factors in switching between filter banks giving rise to the most time efficient way to construct the most general time-varying filter bank.

Depending on whether $\alpha = M-1$ or $\alpha = M-2$, the MFB is said to be Type 1 or Type 2 [6]. There are a special class of FIR MFBs, the canonical ones, which have the property that they are derived from a symmetric $h(n)$. In the Type 1 case they are of length $N = 2Mk$, while in the Type 2 case $N = 2Mk-1$ for some integer k. For simplicity (and because of their utility) we will consider only canonical MFBs.

First consider the unique Type 1 MFB with 4 channels and length 8 filters in [6]. The filters are given in Eqn. 22. In this case it is verified that

$$H_p(z) = F_1(z)V_0 = \left[I - f_1 + z^{-1}f_1\right]V_0 \tag{47}$$

where $f_1 = v_1 v_1^T + v_2 v_2^T$, and (say)

$$v_1 = \begin{bmatrix} 0.41636433418450 \\ -0.78450701561376 \\ 0.32495344564406 \\ 0.32495344564406 \end{bmatrix}, v_2 = \begin{bmatrix} 0.00000000000000 \\ -0.14088210492943 \\ 0.50902478635868 \\ -0.84914427477499 \end{bmatrix}, V_0 = \frac{1}{2}\begin{bmatrix} 1 & 1 & 1 & 1 \\ -\sqrt{2} & 0 & \sqrt{2} & 0 \\ 0 & \sqrt{2} & 0 & -\sqrt{2} \\ 1 & -1 & 1 & -1 \end{bmatrix}.$$

Since $\delta_1 = 2$ and $M = 4$, there are two entry filters and two exit filters given in Table 1 If one had

Table 1: Entry and Exit Filters for Type 1 MFB: $M = 4$, $N = 8$

n	$u_0(n)$	$u_1(n)$	$w_0(n)$	$w_1(n)$
0	-0.12615623124927	0.14881142113876	0.92538912054319	-0.32495344564406
1	-0.42377907352679	-0.35926255114920	0.27548222925507	0.78450701561376
2	-0.63423020353723	-0.53767440435980	-0.18407134071463	-0.52419082913093
3	-0.63423020353723	0.74812553437025	-0.18407134071463	0.06463725916124

directly used the factorization based on v_i one would have 6 entry filters and 2 exit filters. That this is not a stray example as shown by the following theorem a proof of which can be inferred by noting that the implementations in [10, 11, 6] are minimal.

Theorem 3 **Type 1,** $N = 2Mk$, M **even** *McMillan degree of* $H_p(z)$ *and* $H_p^R(z)$ *is* $M(k-1)+\frac{M}{2}$.

Type 1, $N = 2Mk$, M **odd** *McMillan degree of* $H_p(z)$ *is* $M(k-1)+\frac{M-1}{2}$ *and that of* $H_p^R(z)$ *is* $M(k-1)+\frac{M+1}{2}$ *(or vice versa).*

Type 2, $N = 2Mk-1$, M **even** *McMillan degree of* $H_p(z)$ *is* $M(k-1)+\frac{M-2}{2}$ *and that of* $H_p^R(z)$ *is* $M(k-1)+\frac{M}{2}$ *(or vice versa).*

Type 2, $N = 2Mk-1$, M **odd** *McMillan degree of both* $H_p(z)$ *and* $H_p^R(z)$ *is* $M(k-1)+\frac{M-1}{2}$.

To compute this, let us assume $N = MK - l$ and let

$$H_p(z) = \left[\begin{array}{cccc} \alpha_0(z) & \alpha_1(z) & \ldots & \alpha_{M-1}(z) \end{array} \right]. \tag{43}$$

Then it can be shown that with entries z, and I, the $(M-l) \times (M-l)$ identity matrix

$$H_p^R(z) = z^{-(K-1)} \left[\begin{array}{cccccc} \alpha_{M-l}(z^{-1}) & \ldots & \alpha_0(z^{-1}) & \alpha M - 1(z^{-1}) & \ldots & \alpha_{M-l+1}(z) \end{array} \right] \left[\begin{array}{cc} I_{M-l} & 0 \\ 0 & zI_l \end{array} \right]. \tag{44}$$

Taking the determinant on both sides and using the fact that the determinant of a unitary FIR matrix is of the form $const \times z^{-n}$ where n is the McMillan degree, one gets that the McMillan degree of $H_p^R(z)$ is given by

$$(K-1)M - \Delta - l = KM - l - M - \Delta = N - M - \Delta. \tag{45}$$

Therefore the total number of entry plus exit filters is $N - M$, and depends only on the length of the filter, *not* on the McMillan degree. □

Remark: The theorem shows that in the implementation of a filter bank (both analysis and synthesis together) one requires precisely $N - M$ storage elements. This remark must be taken with a caveat since it assumes that the analysis and synthesis filter banks are implemented according to the theory which presupposes that the inputs could be infinite. However, since practically all signals are finite, the synthesis (analysis) filter bank can be implemented by time-reversing its input, filtering with the *analysis* (synthesis) filters and time-reversing the output again. Therefore the filter bank may be implemented in this case with 2Δ storage elements where Δ is the smaller of the two McMillan degrees.

The result is valid for any implementation of the unitary filter bank, not just the specific factorization approaches considered here (though it was obtained based on intuition gathered from the factorization). In particular, if Eqn. 3 is true, then $N = MK$, and one checks that the number of exit filters is $\Delta = K - 1$, and the number of entry filters is $(M-1)(K-1) = N - M - \Delta$.

We now proceed to show that a very important class of filter banks, namely modulated unitary filter banks exhibit this property

6.1 Modulated Filter Banks

Modulated filter banks (MFBs) are ones where the filters $h_i(n)$ are given in terms of a prototype filter $h(n)$ by the following equation:

$$h_i(n) = h(n) \cos\left(\frac{\pi}{2M}(2i+1)(n - \frac{\alpha}{2}) \right) \tag{46}$$

in the storage elements at the time instant of switching. Once again there is no need to compute the outputs of the exit filters, and hence switching off is instantaneous. Even if the factorization based on Eqn. 3 is used, the number of exit filters is $\Delta = L - 1$. However, the number of entry filters in that factorization was more. To see that notice that the number of entry+exit filters in the old factorization was $M(L-1)$, while in the new factorization is $M(K-1)$ and $L \geq K$. The rationale for reducing the number of entry filters is that they cost time (since it means that much more input should be read before the switching is complete).

This approach is optimal if the filter lengths are precisely $N = MK$ for some K. However, N need not be of this form. In general to have a polynomial of degree $K-1$, $M(K-1) < N \leq MK$. Consider such a situation with $N = MK - l$. Then while designing the entry filters, the first l columns of the filter bank (with this factorization) are zero. Therefore one expects the number of entry filters to be $M(K-1) - \Delta - l$ (since one can instead of fixing these columns throw out the corresponding rows). We now prove that the result is true.

In a unitary filter bank, one may shift all the analysis filters and synthesis filters in opposite directions and still preserve PR. Also one may shift any one analysis filter or synthesis filter in opposite directions by multiples of M. Using these two elementary operations one can always ensure that the analysis filters are causal *and* their support (index of non-zero values) is as close to the the origin as possible. We will call this process (for lack of a better term) *causalization*. Notice that this process gives a unique set of analysis/synthesis filters. Also if $\{h_i(n)\}$ is a causalized set of analysis filters, then their reflections $\left\{h_i^R(n)\right\}$ form a causalized set of synthesis filters of the same filter bank. The *length* of the filters in a filter bank is defined to be N if $h_i(n) = 0$ for $n \geq N$, and $h_i(N-1) \neq 0$ for some N.

Theorem 2 *Let $\{h_i(n)\}$ a causalized set of analysis filters of a unitary filter bank with filters of length N. Then number of exit filters is given by the McMillan degree of $H_p(z)$, the corresponding polyphase matrix and the number of entry filters is given by $H_p^R(z)$ the polyphase component matrix of $\left\{h^R(n)\right\}$. If Δ is the McMillan degree of $H_p(z)$, then $N - M - \Delta$ is the McMillan degree of $H_p^R(z)$.*

Proof: We have already seen that the number of exit filters is precisely Δ. If one reflects the filters in the filter bank, then the entry filters of the new filter bank become the exit filters of the old one and vice versa. Therefore, the number of entry filters is precisely the McMillan degree of $H_p^R(z)$.

unitary matrix V_0. Hence Eqn. 38 holds. □

Remark: Since each projection f_i is of rank δ_i it can be written as $v_1 v_1^T + \ldots + v_{\delta_i} v_{\delta_i}^T$, for a *non-unique* set of orthonormal vectors v_i proving the existence of the factorization in Eqn. 3.

In the context of Eqn. 38, in the time-domain we have the following factorization: $\breve{\mathbf{H}} = \prod_{i=K-1}^{0} \mathbf{F}_i \mathbf{V}_0$ where depending on the situation (both-side, right-sided etc.,) $\mathbf{F}_i$ is defined exactly like $\mathbf{V}_i$. We will now construct the entry and exit filters for this filter bank. Let A_i be the matrix that orthonormalizes the rows of f_i. Since f_i has rank δ_i, A_i is a $(M - \delta_i) \times M$ matrix. Define $B_i = \left[\begin{array}{ccc} A_i f_i & 0 & \ldots \end{array} \right]$, D_i to be the matrix who rows are made up of the $v_j's$ that contribute $F_i(z)$. Also let $C_i = \left[\begin{array}{ccc} \ldots & 0 & D_j \end{array} \right]$. The rows of C_i are made up of the v_j's that constitute $F_i(z)$. Then for top and bottom truncation since $\left[\begin{array}{c} B_i \\ \mathbf{F}_i \\ C_i \end{array} \right]$ is unitary so is

$$\left[\begin{array}{c} B_1 \mathbf{V}_0 \\ \vdots \\ B_{K-1} \mathbf{F_{K-2}} \ldots \mathbf{F_1 V_0} \\ \breve{\mathbf{H}} \\ C_{K-1} \mathbf{F_{K-2}} \ldots \mathbf{F_1 V_0} \\ \vdots \\ C_1 \mathbf{V_0} \end{array} \right]. \tag{39}$$

Now let

$$H_p^j(z) = \prod_{i=1}^{j-1} F_i(z) V_0 = \sum_{i=0}^{j-1} h_p^j(i) z^{-i}. \tag{40}$$

We can transfer this information by considering $\mathbf{H}$ instead of $\breve{\mathbf{H}}$ to get the entry and exit filters. They are given as follows:

Entry Filters For for every $j \in \{1, \ldots, K-1\}$ there are $M - \delta_j$ entry filters given by

$$A_j (I - f_j) \left[\begin{array}{cccc} h_p^j(0) J & h_p^j(1) J & \ldots & h_p^j(j-1) J \end{array} \right]. \tag{41}$$

Exit Filters For for every $j \in \{1, \ldots, K-1\}$ there are δ_j exit filters given by

$$D_j \left[\begin{array}{cccc} h_p^j(0) J & h_p^j(1) J & \ldots & h_p^j(j-1) J. \end{array} \right]. \tag{42}$$

Let $\Delta = \sum_{i=1}^{K-1} \delta_i$. Δ is the McMillan degree of $H_p(z)$. The number of exit filters is precisely the McMillan degree of $H_p(z)$, and moreover the canonical exit filter outputs are precisely the entries

filters by that much). Anyway, from both implementational and switching points of view it is convenient to combine the two factors into a McMillan degree 2 factor:

$$\left[I - v_j v_j^T - v_{j-1} v_{j-1}^T + z^{-1}(v_j v_j^T + v_{j-1} v_{j-1}^T)\right] = \prod_{i=j}^{j-1} \left[I - v_i v_i^T + z^{-1} v_i v_i^T\right] \tag{36}$$

In this form the total implementational delay (not the number of storage elements) is reduced by one time unit (one block of input data). Notice that if there is a sequence of adjacent v_i's, l in number, that are mutually orthogonal, one combines them into a degree l factor.

$$\left[I - (\sum_{i=j}^{j-l+1} v_i v_i^T) + z^{-1}(\sum_{i=j}^{j-l+1} v_i v_i^T)\right] = \prod_{i=j}^{j-l+1} \left[I - v_i v_i^T + z^{-1} v_i v_i^T\right]. \tag{37}$$

Keeping this in mind we can factorize $H_p(z)$ as follows:

$$H_p(z) = \prod_{i=1}^{K-1} F_i(z) V_0 \tag{38}$$

where each factor $F_i(z)$ had degree one polynomial entries but with McMillan degree δ_i. For simplicity we will write $F_i(z) = I - f_i + z^{-1} f_i$. Notice that in the factorization considered previously $f_i = v_i v_i^T$. Is is easily seen that the $rank(f_i) = \delta_i$.

It is clear that what one desires is a factorization of given $H_p(z)$ of polynomial degree $K-1$ (i.e., $L = MK$) into $(K-1)$ factors of polynomial degree 1. Moreover, we desire this to be unique. We now obtain precisely such a factorization (and obtain Eqn. 3 as an immediate consequence).

Theorem 1 *Let $H_p(z)$ be the polyphase matrix of a unitary filter bank with polynomial degree $K-1$ (i.e., $N = MK$ length filters - strictly speaking $M(K-1) < N \leq MK$). Then, there exists a unique set of projection matrices f_i with rank δ_i, such that with $F_i(z) = I - f_i + z^{-1} f_i$, Eqn. 38 holds.*

Proof: Since $H_p(z)$ is unitary $h_p^T(0) h_p(K-1) = 0$ and therefore $h_p(0)$ is singular. Let f_{K-1} be the *unique* projection matrix onto the nullspace of $h_p(0)$ (say of dimension δ_{K-1}). That is, $h_p(0)^T f_{K-1} = 0 = f_{K-1} h_p(0)$ (because f_{K-1} is symmetric). Also, since $h(K-1)$ is in the nullspace, $f_{K-1} h(1K-1) = h(K-1)$ and hence $(I - f_{K-1}) h(K-1) = 0$. Therefore $[I - f_{K-1} + z f_{K-1}]\, H_p(z)$ is a matrix polynomial of degree at most degree $K-2$. However if $h(0)$ and $h(K-1)$ are not zero (a assumption one makes without loss of generality) then the degree is *precisely* $K-2$ Moreover, it is unitary since $I - f_{K-1} + z f_{K-1}$ is unitary on the unit circle (complementary projections taken together preserve energy). Repeated application of this procedure $(K-1)$ times gives a degree zero

6 Higher Order Building Blocks For Structured Filter Banks

We have discussed how *optimal* time-varying filter banks trees can be constructed under the constraint the order of $H_p(z)$ as a matrix polynomial is precisely the McMillan degree of $H_p(z)$. But this is generically true only if unitary filter banks are designed using Eqn. 3. When dealing with structured unitary filter banks and orthonormal wavelet bases one may be able to construct unitary filter banks without starting from Eqn. 3. Under such circumstances the factorization approach considered earlier *does give* PR unitary time-varying filter bank trees, but is not optimal if $L \neq K$. In this case, it turns out one can reduce the number of entry filters, while the exit filters remain the same being tied to the McMillan degree.

Consider a given unitary filter bank $\mathbf{H}$. If the filters are time-reversed, it is clear that the time-reversed entry and exit filters form exit and entry filters of this new filter bank (see Eqn. 26). But then for the new filter bank there would be $(K-1)$ entry filters and $(M-1)(K-1)$ exit filters - an apparent contradiction (unless $M=2$)! The trouble is that the polyphase component matrix of the original and the time-reversed filter banks have different McMillan degrees and the number of *optimal* entry and exit filters is tied to the McMillan degree of a filter bank and the corresponding time-reversed filter bank.

Another problem is the non-uniqueness of the factorization in Eqn. 3. During soft switching one has to compare the factors of the two filter banks. However, since the factorization is non-unique it is not very clear how one can compare the factors.

A more direct approach to motivate the problem is as follows: Consider the highest degree coefficient of $H_p(z)$ in the product in Eqn. 3 which has L factors. Clearly if it is non-zero, then the filters are of length ML, the case we have considered so far (with $L=K$). However, if it is zero then the filter length is MK, $L=K+1$. In other words, we have factored $H_p(z)$ with polynomial entries of degree $K-1$ into $K+1$ factors. There are no surprises actually since the McMillan degree (what counts!) is $K-1$. Since

$$h_p(L-1) = \left[\prod_{i=L-1}^{1} v_i v_i^T\right] V_0, \tag{35}$$

is a product of rank one factors and V_0 is a unitary matrix, $h_p(L-1)$ is zero iff a pair of adjacent vectors v_i are orthogonal (i.e., iff for some j, $v_j^T v_{j-1} = 0$). In this case, the product of the corresponding factors $\mathbf{V}_j\mathbf{V}_{j-1}$ will be zero in its left top corner in Eqn. 6 and Eqn. 25. So our earlier approach puts a number of entry filters over this column of zeros to make it full rank and unitary. Clearly, a better strategy would be to ignore these columns (reducing the number of entry

representations are given respectively by

$$\prod_{i=K-2}^{0} \left[I - v_i v_i^T + z^{-1} v_i v_i^T \right] V_0 \qquad \text{and} \qquad \prod_{i=K-1}^{0} \left[I - v_i v_i^T + z^{-1} v_i v_i^T \right] V_0.$$

This involves just the addition of one extra stage in Fig. 5. Clearly this can be done in one time step where the output of the added stage multiplied by A_{K-1} is the transition filter output and every time instant after that the output is that of the new filter bank. These is precisely one block of transition outputs (i.e., $(M-1)$ transition filters). However hard switching between the two requires $(K-2)$ exit filters and $(M-1)(K-1)$ entry filters. Soft switching may be used anywhere in a filter bank tree. Once again the transition filter is the pullback of the entry filter corresponding to adding that last block. Hard switching between filter bank trees is essentially a sequence of soft switching corresponding to all the factors.

Soft switching can lead to new filter bank design problems which will be reported in future work. One way would be to design a set of filter banks with filters of length Mj, $j \in \{1, 2, \ldots, \}$, such that each filter bank has desirable characteristics that successively improves. Then depending on the input data one could use shorter or longer filters by soft switching between them.

Here is another scenario where soft switching is very useful: a signal is being analyzed by a time-varying filter bank and at $n = 0$, one decides to grow the tree by an M channel filter bank. As seen before the first true output occurs at time index $MK - 1$. However, if intervening data leads one to choose an entirely different filter bank (say with more channels or different filters) at some time instant $n_0 < MK - 1$, then one can use the soft switching concept to *prune only the added blocks*, before growing the chosen filter bank blocks - evident temporal economy.

An important outcome of the analysis is that the only freedom in the design of the transition filters corresponds to the freedom in the design of the entry and exit filters. Clearly the exit filters are fixed. The entry filters have a degree of freedom (corresponding to taking out the transition outputs from the implementation and probably doing an orthogonal change of basis). Now consider having overlapping transitions. As far as implementation goes, there is no difference. Once takes the transition outputs corresponding to different filters (or blocks) and does a unitary transformation on them. Under the circumstances it is not clear whether one needs to do that. After all this is not going to have any influence on the reconstruction. Therefore is is probably not very important to design the transition filters (contrast this with the development in [8, 9] where one has to design transition filters).

need to know the entry filters! The transition time is used to initialize the $K-1$ memory elements (states) in the implementation of the M channel filter bank. Obviously one cannot do better than that!

Now consider going from an M channel to a one channel filter bank at $n=0$. The last input $[x(-M),\cdots,x(-1)]$ is fed into the structure in Fig. 5 giving the last output of the first filter bank. At the next instant, the switching has taken place and the output is given by $x(0)$. What about the exit filters in Eqn. 19? Does one have to compute them? No! They are already computed. Consider just the last row. This is just given by inner product of the last input with v_1^T, which is precisely the number stored in the state of the first block. Similarly, the rest of the exit filter outputs are contained (at $n=0$) in each of the $(K-1)$ storage elements in Fig. 5. The transition outputs are obtained by just flushing out all the states of the original filter bank. The relationship between the (input) time index, number of inputs/outputs is given as follows:

	One Channel			M Channel		
Time n	...	-M-1	-1	0	1	...
# Inputs	...	M	M	1	1	...
# Outputs	...	M	M+K-1	1	1	...

In summary switching on an M channel filter bank takes $(M-1)$ time instants for the first output to come, while switching it off is instantaneous. The same goes with pruning or growing a tree. However, in this case the $(M-1)$ time instants is measured from the perspective of the channel at the point of tree growth. However, from the point of view of the *fundamental input* to the tree it could be several more global time instants depending on the topology of the tree.

5.1 Soft Switching

When switching from an M_1 channel filter bank to an M_2 channel filter bank (at $n=0$), the transition outputs occur from $n=(M_2-1)$ to $n=(K_2-1)M_2-1$. Can the first true output ever be obtained earlier than $n=M_2K_2-1$? The answer is yes, and it depends on a special structure between the two filter banks. The crucial point is to realize that time-varying nature is introduced by adding or removing the building blocks $[I-v_iv_i^T+z^{-1}v_iv_i^T]$ at various stages in the system. If the two filter banks that one is switching between has common building blocks, one only needs to remove and add appropriate blocks (rather than completely switching off one filter bank and switching on the next). Consider switching between M channel filter banks whose polyphase

stream of data comes in every instant, and a continuous stream of outputs comes out every M^{th} instant. The input block at any given instant is given by, $[x(n), x(n-1), \cdots, x(n-(M-1))]$. Therefore the output at instant n depends not only on the input at time-instant n but the $M-1$ previous values. This degree of non-causality is the cause for a seldom appreciated *structural delay* caused by the filter bank (including analysis and synthesis) [3]. This structural delay is independent of the non-causality that comes in the implementation of a PR filter bank by using non-causal filters (unitary filter bank always have non-causal analysis or synthesis filters). Perhaps, it is more natural to define the downsampling operator to give outputs at $n = M-1, 2M-1, \ldots$ in which case this degree of non-causality is hidden - a moot point, to say the least!

Now consider the problem of hard switching from a one channel filter bank to an M channel filter bank at $n = 0$. After a certain transition, corresponding to the entry filters in the switch, the implementation is exactly as in Fig. 5. What happens during the transition period? Firstly, notice that the first output value has to come only at $n = (M-1)$. This is because, the input is available only from $n = 0$, and the first block of M data points is available only at $n = (M-1)$. From Eqn. 10 one sees that the first $(K-1)$ blocks of data are operated on by entry filters, during which time for every block of M inputs, the output block has only $(M-1)$ numbers. However, from the K^{th} block onwards the output is precisely the output of the filter bank. The relationship between the (input) time index and the number of input and output points is given below:

	One Channel			Transition				M Channels		
Time n	...	-2	-1	M-1	2M-1	...	MK-1	M(K+1)-1	M(K+2)-1	...
# Inputs	...	1	1	M	M	...	M	M	M	...
# Outputs	...	1	1	M-1	M-1	...	M	M	M	...

It is clear that the switching has been instantaneous (in so far has one had to wait till a block of M inputs is available). The first "true" output of the filter bank comes at $n = MK-1$. This is the best possible that can be achieved since the filter length $N = MK$! Consider the first $(M-1)$ transition outputs given by $A_1(I - v_1 v_1^T)$. Clearly this is the output of running just the first stage in Fig. 5 for one step, and multiplying the output by A_1. Similarly, the second transition is seen to be given by running both the first and second stages for one time step and multiplying the output by A_2. In other words once we decide to switch to the new filter bank, we wait till we have a block of M input data and add the blocks of the cascade, one for each new block of input data, simultaneously picking out the transition outputs from new block. Once the transition output from the last added stage is taken, the filter bank is completely functional, and every other block of input gives the "true" filter bank output. Notice the beauty of the implementation:. *There is no*

4.6 Hard Switching Between Arbitrary Subtrees

Clearly one can switch between arbitrary trees by first pruning the first tree until it becomes a one channel filter bank and then growing the second tree. In general this is the best that one can do (in terms of quick transition to the second tree). However if the switching is between two trees that differ only in a subtree, then one can prune that particular section and grow whatever else one pleases there. Under special circumstances that we prefer to call soft-switching (understanding which is impossible in the context of just switching filter bank trees) one can do better as we shall see later. However, it is useful to remember that in every case the transition filters are merely the *pullback* of the entry/exit filters of the added/pruned filter bank by the extant tree preceding the transition.

5 Implementation of Time-Varying Trees

We have seen that arbitrary time-varying filter bank trees are possible, and that transition filters can be explicitly calculated, for both switching between filter banks and for tree growth and pruning. For an arbitrary tree at each instant that a switching takes place there are corresponding transition filters etc., and the entire bookkeeping process for a deep tree gets fairly complicated. The goal of this section is to show that, one can indeed to all of this very easily if one uses the factorization results obtained earlier. Moreover, implementation of time-varying filter bank trees is essentially no more complex that implementing a simple time-invariant filter bank. One of the by products of the implementation would be the concept of *soft switching* which is impossible to see without the factorization approach.

First consider the implementation of an M-channel filter bank shown in Fig. 5. The input

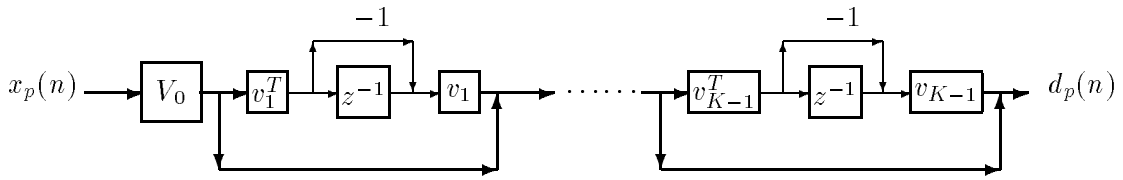

Figure 5: Implementation of Unitary FB: Analysis Bank

$x(n)$ is segmented into blocks of size M. The sequences of blocks goes through a cascade of simple building blocks to give the output of the M channels. If the input is infinite, then a continuous

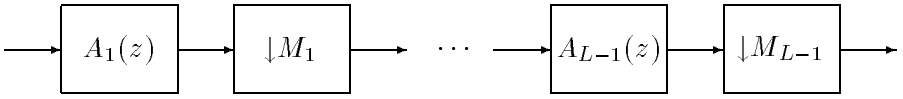

Figure 2: A Branch of an Existing Tree

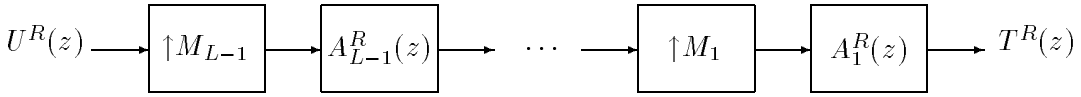

Figure 3: Transition Filter For Tree Growth

4.5 Pruning a Filter Bank Tree

Assume that from $n = -\infty$, a tree is operational consisting of an M_1 channel filter bank having an M_2 channel filter bank sticking out of its lowpass channel. At time $n = 0$, we wish to prune this tree into a simple M_1 channel filter bank. The mapping from input to the output of the section of the tree to be pruned is given by $\mathbf{H_2 P H_1}$. Therefore the pruning operation is accomplished as follows:

$$\mathbf{d} = \begin{bmatrix} \mathbf{H_2} & 0 \\ \mathbf{W_2} & 0 \\ 0 & I \end{bmatrix} \mathbf{P H_1}. \tag{34}$$

Clearly, there are as many transition filters as there are exit filters of $\mathbf{H_2}$. Moreover if $W(z)$ is any particular row of $\mathbf{W_2}$, the corresponding transition filter is given by $W(z^2)H_0^R(z)$. From this formula one sees that for $j \in \{1, \ldots, K_2 - 1\}$ there is one transition filter each of length $M_1(M_2 j + K_1 - 1)$. This matches with the intuition that if $M_1 = 1$, one is just switching from an M_2 channel filter bank to a one channel filter bank. The transition filters above are canonical. In the more general case of tree pruning, if the map from the input to the point of pruning is given as in Fig. 2, then the transition filters are given by Fig. 4.

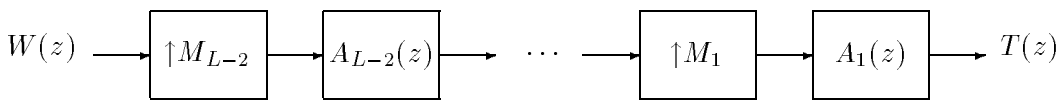

Figure 4: Transition Filter For Pruning

essentially does is:

$$\mathbf{U_2}\begin{bmatrix} h_0(N-1) & \dots & h_0(N-M-1) & \dots & h_0(0) & 0 & \dots & \dots & \dots \\ 0 & \dots & h_0(N-1) & \dots & h_0(N-M-1) & \dots & h_0(0) & 0 & \ddots \\ \vdots & \vdots & \vdots & \vdots & \dots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}, \quad (33)$$

where $h_0(n)$ is the impulse response of the lowpass filters of the first filter bank. By inspection it is clear that there are $(K_2 - 1)$ transition filters in sets of $(M_2 - 1)$. In set j, $j \in \{1, \dots, K_2 - 1\}$ the filters are of length $M_1(M_2 j + K_1 - 1)$. If $M_1 = 1$, then the length becomes $M_2 j$, which agrees with our intuition that the problem reduces to switching from a one channel filter bank to an M_2 channel filter bank. Stated differently, if $U(z)$ denotes the $\mathcal{Z}$ transform of any given entry filter, and $T(z)$ the transition filter, then $T^R(z) = U^R(z^{M_1})H_0^R(z)$, where $H_0^R(z)$ is the reflection of the $H_0(z)$, the lowpass filter of $\mathbf{H_1}$, and $U^R(z)$ and $T^R(z)$ are reflections of the corresponding filters. For example consider growing a binary tree using Daubechies' 4 coefficient filter banks. In this case there is one row $U(z) = 1/2 + \sqrt{3}/2z^{-1}$. Therefore, the transition filter is given by $T^R(z) = U^R(z^2)H_0^R(z)$. Indeed the single six coefficient transition filter is given by

$t_0(0)$	0.24148145657227
$t_0(1)$	0.41825815186890
$t_0(2)$	0.53033008588991
$t_0(3)$	0.65973960844117
$t_0(4)$	0.19411428382689
$t_0(5)$	−0.11207193402101

Now consider the more general problem of growing an arbitrary tree. One can easily obtain the transition rows algorithmically. The crux of the issue is that every transition filter is the net effect of the entry filter at the point of tree growth seen from the perspective of the input rather than the output point at which the tree is grown. Does one really have to know these entry/exit/transition filters? How does one implement these transition filters? All these questions will be answered in Section 5, where we will generate a machine that essentially takes care of doing anything time-varying in a unitary fashion. In any case, we will indeed give the explicit formula (actually algorithm) for the transition filters. Let the mapping from the input channel to the output "growth" channel by as shown in Fig. 2. Then the corresponding transition filters are given by the system in Fig. 3, when driven by the entry filters of the newly added filter bank. The transition filters are said to be canonical if the entry filters that they come from are canonical. Clearly the transition filters are non-unique. Notice that the transition filters are obtained by running time-reversed entry filter through the synthesis bank of the corresponding branch of the extant tree.

4.3 Hard Switching - M_1 channel to M_2 channel

Here we assume that we want to switch between two filter banks. At $n = 0$ one switches into the second filter bank. One can accomplish this as follows:

$$\mathbf{d} = \left[\begin{array}{c|c} \mathbf{H_1} & 0 \\ \mathbf{W_1} & 0 \\ 0 & \mathbf{U_2} \\ 0 & \mathbf{H_2} \end{array}\right] \mathbf{x}. \tag{31}$$

The transition region is given by the exit filters of the first filter bank and the entry filters of the second. Clearly the transition filters are abrupt (they do not overlap). One can obtain overlapping transition filters as follows: replace it by any orthogonal basis for the row space of the matrix $\left[\begin{array}{c|c} \mathbf{W_1} & 0 \\ 0 & \mathbf{U_2} \end{array}\right]$. For example consider switching between two channel filter banks with length 4 and length 6 Daubechies' filters. In this case, there is one exit filter ($\mathbf{W_1}$) and two entry filters ($\mathbf{U_2}$). In [8, 9] for the same example they construct four transition filters, one more than required.

4.4 Growing a Filter Bank Tree

Growing a filter bank into a tree relies upon the construction that goes from a one channel to an M channel filter bank. The channel at which one wants to grow the tree (by introducing an M channel filter bank) may be effectively considered to be the output of a one channel filter bank. One replaces this one channel filter bank by an M channel filter bank. How many transition filters are there? What are they? These questions can be answered explicitly, but the notation would be cumbersome (since the tree one begins with could be arbitrary). To illustrate this fundamental idea we will consider the following scenario: We have an M_1 channel filter bank from which at $n = 0$ we wish to grow, say, the lowpass channel into an M_2 channel filter bank. Let $\mathbf{P}$ be a matrix that selects the outputs of the lowpass channel of the the original filter bank. Then the output of the portion of the tree corresponding to the lowpass channel is given by

$$\mathbf{d} = \left[\begin{array}{cc} \mathbf{I} & 0 \\ 0 & \mathbf{U_2} \\ 0 & \mathbf{H_2} \end{array}\right] \mathbf{P}\mathbf{H_1}. \tag{32}$$

And transition filters are given by the following matrix product: $\mathbf{U_2}\mathbf{P}\mathbf{H_1}$. One immediately sees that there are precisely as many transition filters as there are entry filters of the second filter bank. The expression for the transition filters may seem complicated at first sight, but what one

For example, in the 4 coefficient Daubechies' filters in [1] case, there is one entry filter (the same as in the example for top-truncation). Also there is one exit filter ($v_1^T V_0 J$) given by $\begin{bmatrix} \sqrt{3}/2 & 1/2 \end{bmatrix}$. Indeed the following matrix (with the specific choice $L = 4$) is unitary:

$$\begin{bmatrix} 0.8660 & 0.5000 & 0 & 0 \\ -0.1294 & 0.2241 & 0.8365 & 0.4830 \\ -0.4830 & 0.8365 & -0.2241 & -0.1294 \\ 0 & 0 & -0.5000 & 0.8660 \end{bmatrix}. \tag{28}$$

4 Time Varying Filter Banks

The filter banks for right and left sided signals in the previous section, and their corresponding factorizations, act as building blocks for the construction of more general time-varying filter banks (where by default input signals will always assumed to be two-sided). We will consider a series of time-varying filter bank problems and in each case give an explicit formula for the filters corresponding to the transition region (which will appropriately turn out to be built out of the entry and exit filters in the previous section).

4.1 Hard Switching - 1 channel to M

Until $n = -1$, the input is the same as the output. At $n = 0$, one switches into an M-channel filter bank as quickly as possible. In this case, the transition is accomplished precisely by the entry filters (hence the name entry) of the filter bank. The input/output of this time-varying filter bank is given by

$$\mathbf{d} = \left[\begin{array}{c|c} \mathbf{I} & 0 \\ 0 & \mathbf{U} \\ 0 & \mathbf{H} \end{array}\right] \mathbf{x}. \tag{29}$$

4.2 Hard Switching - M channel to 1

Until $n = -1$, the filter bank is operational. From $n = 0$ onwards the inputs "leaks" to the output. The fastest way to accomplish this is given by the following input/output map:

$$\mathbf{d} = \left[\begin{array}{c|c} \mathbf{H} & 0 \\ \mathbf{W} & 0 \\ 0 & \mathbf{I} \end{array}\right] \mathbf{x}. \tag{30}$$

end we start with the factorization of $H_p(z)$. If we define the $L \times L$ matrix $\mathbf{V}_0 = diag(V_0)$, and for $i \in \{1, \ldots, K-1\}$ the $(L - Mi) \times (L - Mi + M)$ matrices

$$\mathbf{V}_i = \begin{bmatrix} v_i v_i^T & I - v_i v_i^T & 0 & \ldots & \ldots & 0 \\ 0 & v_i v_i^T & I - v_i v_i^T & \ldots & \ldots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 & \vdots \\ \vdots & \ddots & 0 & v_i v_i^T & I - v_i v_i^T & 0 \\ 0 & \ldots & \ldots & 0 & v_i v_i^T & I - v_i v_i^T \end{bmatrix}. \tag{25}$$

then $\breve{\mathbf{H}}$ is readily verified by induction to be $\prod_{i=K-1}^{0} \mathbf{V}_i$. Since each of the factors (except $\mathbf{V}_0$) has M more columns than rows, they can be made unitary by appending appropriate rows. From the analysis in the previous sections it is clear that if we define the matrices $B_j = \begin{bmatrix} A_j(I - v_j v_j^T) & 0 & \ldots & 0 \end{bmatrix}$, and $C_j = \begin{bmatrix} 0 & \ldots & 0 & v_j^T \end{bmatrix}$, then $\begin{bmatrix} B_i \\ \mathbf{V}_i \\ C_i \end{bmatrix}$ is unitary. Moreover, if we define $\mathbf{T}_0 = \mathbf{V}_0$ and for $i \in \{1, \ldots, K-1\}$,

$$\mathbf{T}_i = \begin{bmatrix} I_{(M-1)(i-1)} & 0 & 0 \\ 0 & B_i & 0 \\ 0 & \mathbf{V}_i & 0 \\ 0 & C_i & 0 \\ 0 & 0 & I_{i-1} \end{bmatrix}$$

then each of the factors $\mathbf{T}_i$ is a square unitary matrix of size $L - N + M$ and

$$\prod_{i=K-1}^{0} \mathbf{T}_i = \begin{bmatrix} B_1 \mathbf{V}_0 \\ \vdots \\ \breve{\mathbf{H}} \\ \vdots \\ C_1 \mathbf{V}_0 \end{bmatrix}, \tag{26}$$

is the unitary matrix that acts on the data. The corresponding unitary matrix that acts on $\mathbf{x}$ (rather than $\breve{\mathbf{x}}$) is of the form $\begin{bmatrix} \mathbf{U} \\ \mathbf{H} \\ \mathbf{W} \end{bmatrix}$, $\mathbf{U}$ has $(M-1)(K-1)$ rows of entry filters in $(K-1)$ sets given by Eqn. 12, while $\mathbf{W}$ has $(K-1)$ rows of exit filters given by Eqn. 20. The rows of $\mathbf{U}$ and $\mathbf{W}$ form the entry and exit filters respectively. Clearly they are non-unique and the solution above is said to be a canonical set. The input/output behavior is captured in

$$\begin{bmatrix} \mathbf{u} \\ \mathbf{d} \\ \mathbf{w} \end{bmatrix} = \begin{bmatrix} \mathbf{U} \\ \mathbf{H} \\ \mathbf{W} \end{bmatrix} \mathbf{x}. \tag{27}$$

Here $K = 2$ and one has $v_1^T = \begin{bmatrix} 0.40824829046386 & -0.81649658092773 & 0.40824829046386 \end{bmatrix}$ and

$$V_0 = \begin{bmatrix} 0.57735026918963 & 0.57735026918963 & 0.57735026918963 \\ -0.78867513459481 & 0.57735026918963 & 0.21132486540519 \\ 0.21132486540519 & 0.57735026918963 & -0.78867513459481 \end{bmatrix}. \tag{23}$$

The exit filter row is given by $v_1^T V_0 J = \begin{bmatrix} -0.25881904510252 & 0.00000000000000 & 0.96592582628907 \end{bmatrix}$. Indeed one easily verifies that $\begin{bmatrix} \mathbf{H} \\ \mathbf{W} \end{bmatrix}$ given below is unitary.

$$\begin{bmatrix} & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ & -0.1057 & 0.0000 & 0.3943 & 0.6830 & 0.5774 & 0.1830 & \vdots & \vdots & \vdots \\ & 0.2113 & -0.0000 & -0.7887 & 0.0000 & 0.5774 & 0.0000 & \vdots & \vdots & \vdots \\ \ddots & -0.1057 & 0.0000 & 0.3943 & -0.6830 & 0.5774 & -0.1830 & 0 & 0 & 0 \\ & \dots & \dots & 0 & -0.1057 & 0.0000 & 0.3943 & 0.6830 & 0.5774 & 0.1830 \\ & \dots & \dots & 0 & 0.2113 & -0.0000 & -0.7887 & 0.0000 & 0.5774 & 0.0000 \\ & \dots & \dots & 0 & -0.1057 & 0.0000 & 0.3943 & -0.6830 & 0.5774 & -0.1830 \\ & \dots & \dots & \dots & \dots & \dots & 0 & -0.2588 & 0.0000 & 0.9659 \end{bmatrix}$$

3.3 Finite $x(n)$ - Top and Bottom Truncation

We now consider the situation where the input is in some finite interval. Not surprisingly the result is obtained by a combination of top and bottom truncation. Since by its very nature, an M-channel filter banks works on M-vector signals (the polyphase components of $x(n)$), it is natural to expect that the input signal is some multiple of M. Let $\mathbf{x} = [x(0), x(1), \cdots, x(L-1)]$, where $L = M\breve{L}$ and let $\breve{\mathbf{x}} = [x(M-1), \dots, x(0), x(M), \cdots, x(L-1), \cdots, x(L-M)]$. Let $\mathbf{d}$ be the output signal in the channel corresponding to filtering by the filter bank. Then,

$$\mathbf{d} = \breve{\mathbf{H}}\breve{\mathbf{x}} \stackrel{\text{def}}{=} \begin{bmatrix} h_p(K-1) & \dots & h_p(0) & 0 & \dots & \dots & \dots & 0 \\ 0 & h_p(K-1) & \dots & h_p(0) & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \vdots & \vdots \\ \vdots & \dots & \dots & 0 & h_p(K-1) & \dots & h_p(0) & 0 \\ 0 & \dots & \dots & \dots & 0 & h_p(K-1) & \dots & h_p(0) \end{bmatrix} \breve{\mathbf{x}}. \tag{24}$$

Notice that $\breve{\mathbf{H}}$ is an $(L - N + M) \times L$ matrix, where $N = MK$ is the length of the filters. Now since the rows of $\breve{\mathbf{H}}$ are mutually orthonormal (i.e., has rank L), it is clear that one has to append precisely $N - M = M(K-1)$ rows from the orthogonal complement of $\breve{\mathbf{H}}$ to make the map from $\mathbf{x}$ to an appended $\mathbf{d}$ unitary. Our goal is not just to do this, rather to obtain a factorization. To this

the form of $\mathbf{V}_i$, it is obvious that with $C_i = \begin{bmatrix} \dots & 0 & v_i^T \end{bmatrix}$, $\begin{bmatrix} \mathbf{V_i} \\ C_i \end{bmatrix}$ is an upper left semi-infinite matrix that is unitary. With $\mathbf{T}_0 = \mathbf{V}_0$, and for $i \in \{1, \dots, K-1\}$,

$$\mathbf{T}_i = \begin{bmatrix} \mathbf{V}_i & 0 \\ C_i & 0 \\ 0 & I_{i-1} \end{bmatrix}, \tag{18}$$

it follows that

$$\mathbf{T}_{i-1} \dots \mathbf{T}_0 = \begin{bmatrix} \mathbf{V}_{i-1} \dots \mathbf{V_0} \\ \vdots \\ C_j \mathbf{V}_{j-1} \dots \mathbf{V_0} \\ \vdots \\ C_1 \mathbf{V_0} \end{bmatrix} \text{ and hence } \prod_{i=K-1}^{0} \mathbf{T}_i = \begin{bmatrix} \breve{\mathbf{H}} \\ C_{K-1} \mathbf{V}_{K-2} \dots \mathbf{V_0} \\ \vdots \\ C_j \mathbf{V}_{j-1} \dots \mathbf{V_0} \\ \vdots \\ C_1 \mathbf{V_0} \end{bmatrix}. \tag{19}$$

It is easily seen that the last row of $\mathbf{V}_{j-1} \dots \mathbf{V}_0$ is given by $\begin{bmatrix} \dots & 0 & R_j \end{bmatrix}$ and therefore the non-zero entries in each of the added rows are of the form $v_j^T R_j$. If $\mathbf{x} = [\dots, x(-2), x(-1)]$, and $\breve{\mathbf{x}} = [\dots, x(-M), x(-1), x(-2), \dots, x(-M+1)]$, and if $\mathbf{W}$ denotes the corresponding rows added to $\mathbf{H}$, then $\begin{bmatrix} \mathbf{H} \\ \mathbf{W} \end{bmatrix}$ is unitary and the $(K-1)$ rows of $\mathbf{W}$ are given by

$$v_j^T \begin{bmatrix} h_p^j(j-1)J & h_p^j(j-2)J & \dots & h_p^j(0)J \end{bmatrix}. \tag{20}$$

These added rows correspond to *exit filters* given by

$$v_j^T \begin{bmatrix} h_p^j(0)J & v_j^T h_p^j(j-2)J & \dots & h_p^j(0)J \end{bmatrix}.$$

There is one exit filter each of length Mj for $j \in \{1, \dots, K-1\}$. Clearly the exit filters are non-unique since the identity matrices in $\mathbf{T}_i$ could be replaced by an arbitrary unitary matrix. However, the set of filters above is said to be *the* canonical set of *exit* filters. Canonical exit filters, in contrast to canonical entry filters are unique. One summarizes the input/output of this system as follows:

$$\begin{bmatrix} \mathbf{d} \\ \mathbf{w} \end{bmatrix} = \begin{bmatrix} \mathbf{H} \\ \mathbf{W} \end{bmatrix} \mathbf{x}. \tag{21}$$

We now consider an example of bottom truncation for the (unique) three-channel Type 1 modulated filter bank associated with a wavelet orthonormal basis and with length 6 filters in [6, 5]. For $i \in \{0, \dots, M-1\}$ and for $n \in \{0, \dots, 2M-1\}$, the filters are given by

$$h_i(n) = \sqrt{\frac{1}{2M}} \left[\sin\left(\frac{\pi(i+1)(n+.5)}{M} - (2i+1)\frac{\pi}{4} \right) - \sin\left(\frac{\pi i(n+.5)}{M} - (2i+1)\frac{\pi}{4} \right) \right]. \tag{22}$$

There is one canonical entry filter of length 2. In this simple case, since $\begin{bmatrix} \cos(\pi/12) & -\sin(\pi/12) \end{bmatrix}$ is orthogonal to v_1^T, it is the *unique* (up to negation) row vector $A_1(I - v_1 v_1^T)$ and $R_0 = V_0$. Therefore the entry filter coefficients are given by

$$\begin{bmatrix} u_0(0) \\ u_0(1) \end{bmatrix} = \begin{bmatrix} (\cos(\pi/12) - \sin(\pi/12))/\sqrt{2} \\ (\cos(\pi/12) + \sin(\pi/12))/\sqrt{2} \end{bmatrix} = \begin{bmatrix} 1/2 \\ \sqrt{3}/2 \end{bmatrix}. \tag{15}$$

It is easily verified that the following lower semi-infinite matrix corresponding to the impulse response is both left and right unitary:

$$\begin{bmatrix} 0.8660 & 0.5000 & 0 & 0 & \dots & \dots & \dots \\ -0.1294 & 0.2241 & 0.8365 & 0.4830 & \ddots & \vdots & \\ -0.4830 & 0.8365 & -0.2241 & -0.1294 & 0 & \ddots & \\ 0 & 0 & -0.1294 & 0.2241 & 0.8365 & 0.4830 & \ddots \\ 0 & 0 & -0.4830 & 0.8365 & -0.2241 & -0.1294 & \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \ddots \end{bmatrix}.$$

This is in contrast to the same example in [8, 9], where the authors construct two transition filters of length 3 each: clearly, the example in their paper is non-canonical (the transition is done at a slower pace than required). The probable reasons why the authors in that paper obtain non-canonical filters, and some of the merits behind the choice will be discussed later. in Section 5.

3.2 Left-sided $x(n)$ - Bottom Truncation

If $x(n)$ is a left-sided (ending at, say, $n = 0$), then one can truncate the doubly infinite matrices in Eqn. 1, or equivalently Eqn. 2 at the bottom to give upper left semi-infinite matrices. One can also easily show that $\breve{\mathbf{H}} = \prod_{i=K-1}^{0} \mathbf{V_i}$ with

$$\breve{\mathbf{H}} = \begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots \\ \ddots & h_p(K-1) & h_p(K-2) & \dots & h_p(0) & 0 \\ & 0 & h_p(K-1) & \dots & h_p(1) & h_p(0) \end{bmatrix}, \mathbf{V_i} = \begin{bmatrix} \vdots & \vdots & \vdots \\ \ddots & v_i v_i^T & I - v_i v_i^T & 0 \\ & 0 & v_i v_i^T & I - v_i v_i^T \end{bmatrix}. \tag{16}$$

One readily verifies that

$$\mathbf{V}_i \mathbf{V}_i^T = \begin{bmatrix} \ddots & \dots & 0 \\ \vdots & I & \vdots \\ 0 & \dots & I \end{bmatrix} \quad \text{and} \quad \mathbf{V}_i^T \mathbf{V}_i = \begin{bmatrix} \ddots & 0 & \dots \\ 0 & I & \vdots \\ \vdots & \dots & I - v_i v_i^T \end{bmatrix}, \tag{17}$$

and once again $\breve{\mathbf{H}}\breve{\mathbf{H}}^T = I$ while $\breve{\mathbf{H}}^T\breve{\mathbf{H}} \neq I$. Each factor $\mathbf{V}_i$, $i \neq 0$, is row rank deficient by one since $(I - v_i v_i^T)$ has rank $(M-1)$. Therefore for each factor $\mathbf{V}_i$, one has to append a row. From

Then the first row of $\mathbf{V}_{j-1} \ldots \mathbf{V}_1 \mathbf{V}_0$ is given by

$$\left[\begin{array}{cccccc} h_p^j(j-1) & h_p^j(j-2) & \ldots & h_p^j(0) & 0 & \ldots \end{array} \right] \stackrel{\text{def}}{=} \left[\begin{array}{cccc} R_j & 0 & \ldots \end{array} \right].$$

Hence from Eqn. 8, the non-zero elements of the added rows are of the form $A_j(I - v_j v_j^T)R_j$. In order to go from this to the rows of $\mathbf{U}$, the *entry filters* from the relationship between $\mathbf{x} = [x(0), x(1), \ldots]$, and $\breve{\mathbf{x}} = [x(M-1), \ldots, x(0), x(M), \ldots]$, it follows that the rows corresponding to the $j^{\,th}$ set of entry filters is given by

$$A_j(I - v_j v_j^T) \left[\begin{array}{cccc} h_p^j(j-1)J & h_p^j(j-2)J & \ldots & h_p^j(0)J, \end{array} \right] \tag{12}$$

where J is the $M \times M$ exchange matrix that reverses the columns of any matrix it operates from the right on. The filters themselves are given by

$$A_j(I - v_j v_j^T) \left[\begin{array}{cccc} h_p^j(0)J & h_p^j(1)J & \ldots & h_p^j(j-1)J, \end{array} \right].$$

Each set has $(M-1)$ filters each of length Mj. The entry filters as given above will be called a *canonical set* of entry filters. To motivate the definition notice that in each of the factors $\mathbf{T}_i$ in Eqn. 9, the identity matrix at the top left corner could be replaced by an arbitrary unitary matrix and still unitariness would be preserved. Therefore, it is clear that any set of entry filters (of size $(M-1)(K-1)$) have to necessarily be the canonical set multiplied on the left by an arbitrary $(M-1)(K-1) \times (M-1)(K-1)$ unitary matrix. In the non-canonical case, the lengths of the filters no longer have to be of the form Mj. Another reason that they are called canonical entry filters is that they also represent a minimal set of entry filters - any other set of entry filters that makes $\breve{\mathbf{H}}$ unitary has to necessarily have more rows. Since a canonical set it is already unitary, adding rows would incur adding columns to preserve unitariness (since otherwise the column rank would fall). Notice that even a canonical set is not unique since A_j is not unique. The importance of canonical sets also arise when we discuss implementation issues and the most general tree-structured time-varying filter bank. Finally the input/output behavior of the unitary FIR filter bank can be expressed as follows:

$$\left[\begin{array}{c} \mathbf{u} \\ \mathbf{d} \end{array} \right] = \left[\begin{array}{c} \mathbf{U} \\ \mathbf{H} \end{array} \right] \mathbf{x}. \tag{13}$$

For example consider switching to a two channel filter bank with length four filters (corresponding to the Daubechies filters [1]). In this case, $M = 2$, $N = 4$, and $K = 2$ and it can be shown that $v_1 = \left[\begin{array}{cc} \sin(\pi/12) & \cos(\pi/12) \end{array} \right]$ and $V_0 = \left[\begin{array}{cc} 1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{2} & -1/\sqrt{2} \end{array} \right]$ and $H_p(z)$ is given by

$$H_p(z) = \frac{1}{4\sqrt{2}} \left[\begin{array}{cc} (1+\sqrt{3}) + z^{-1}(3-\sqrt{3}) & (3+\sqrt{3}) + z^{-1}(1-\sqrt{3}) \\ (3+\sqrt{3}) + z^{-1}(1-\sqrt{3}) & (-3+\sqrt{3}) - z^{-1}(1+\sqrt{3}) \end{array} \right]. \tag{14}$$

filter bank for one sided signals! However, this cannot be since every factor $\mathbf{V}_i$, $i \neq 0$, is row rank deficient by $(M-1)$. Therefore $\breve{\mathbf{H}}$ has a non-trivial kernel, precluding PR. However, it has full column rank since each of the $\mathbf{V}_i$ do. One readily verifies that

$$\mathbf{V}_i \mathbf{V}_i^T = \begin{bmatrix} I & 0 & \dots \\ 0 & I & \vdots \\ \vdots & \dots & \ddots \end{bmatrix} \quad \text{and} \quad \mathbf{V}_i^T \mathbf{V}_i = \begin{bmatrix} v_i v_i^T & 0 & \dots \\ 0 & I & \vdots \\ \vdots & \dots & \ddots \end{bmatrix}. \tag{7}$$

Since the $M \times M$ matrix $v_i v_i^T$ has rank one, the row rank deficiency is precisely $(M-1)$. Also clearly $\breve{\mathbf{H}}^T \breve{\mathbf{H}} = I$. To fix this one has to append $(M-1)$ rows to each factor $\mathbf{V_i}$. Effectively therefore one has to append precisely $(K-1)(M-1)$ rows to $\breve{\mathbf{H}}$ to make it unitary.

From the form of $\mathbf{V}_i$, it is clear that the $(M-1)$ rows must span the $(M-1)$ dimensional space corresponding to the M rows $\begin{bmatrix} I - v_i v_i^T & 0 & \cdots \end{bmatrix}$. If A_i is the $(M-1) \times M$ matrix that orthonormalizes $(I - v_i v_i^T)$, and if

$$B_i = \begin{bmatrix} A_i(I - v_i v_i^T) & 0 & \dots \end{bmatrix}, \tag{8}$$

then $\begin{bmatrix} B_i \\ \mathbf{V_i} \end{bmatrix}$ is a lower semi-infinite matrix that is unitary (and same as $\mathbf{V_i}$ but for the *entry filters* B_i). We now define unitary building blocks $\mathbf{T}_i$ as follows:

$$\mathbf{T}_i = \begin{bmatrix} I_{(M-1)(i-1)} & 0 \\ 0 & B_i \\ 0 & \mathbf{V}_i \end{bmatrix}. \tag{9}$$

Obviously for all i, $\mathbf{T}_i$ is unitary. We also define $\mathbf{T}_0 = \mathbf{V}_0$. Now the product $\mathbf{T}_1 \mathbf{T}_0$ is given by $\begin{bmatrix} B_1 \mathbf{V}_0 \\ \mathbf{V}_1 \mathbf{V}_0 \end{bmatrix}$. It is easily seen that for any i,

$$\mathbf{T}_{i-1} \dots \mathbf{T}_0 = \begin{bmatrix} B_1 \mathbf{V_0} \\ \vdots \\ B_j \mathbf{V}_{j-1} \dots \mathbf{V_0} \\ \vdots \\ \mathbf{V}_{i-1} \dots \mathbf{V_0} \end{bmatrix} \text{ and hence } \prod_{i=K-1}^{0} \mathbf{T}_i = \begin{bmatrix} B_1 \mathbf{V_0} \\ \vdots \\ B_j \mathbf{V}_{j-1} \dots \mathbf{V}_0 \\ \vdots \\ B_{K-1} \mathbf{V}_{K-2} \dots \mathbf{V}_0 \\ \breve{\mathbf{H}} \end{bmatrix}. \tag{10}$$

Notice that the product has precisely $(M-1)(K-1)$ extra rows added to $\breve{\mathbf{H}}$. The expression for these rows can be simplified. For $j \geq 1$ define

$$H_p^j(z) = \prod_{i=j-1}^{1} \left[I - v_i v_i^T + z^{-1} v_i v_i^T \right] V_0 = \sum_{i=0}^{j-1} h_p^j(i) z^{-i}. \tag{11}$$

In the following we will construct filter banks that act respectively on right sided, left sided and finite signals. The approaches to right and left sided signals can be combined to give the third case. However, it is clear that a unitary FIR filter bank acting on a finite signal has to be a unitary matrix (hopefully) most of the rows of which are made up of the analysis filters. One expects a few rows of outputs that help to switch into the filter bank mode and some rows to switch out of the filter bank mode.

3.1 Right-sided Signals - Top Truncation

If $x(n)$ is a right-sided signal (starting at say $(n = 0)$), then the doubly-infinite matrix in Eqn. 1 may be truncated at the top to obtain a lower, right semi-infinite matrix, say $\mathbf{H}$. It turns out that one can always show that the truncated matrix is not unitary. In [8, 9] the authors fix this problem by adding additional rows/columns to $\mathbf{H}$. Adding a row is equivalent to getting an extra output value that does not correspond to the filters in the filter bank. Equivalently one might view this as an output delay. In a tree-structured filter bank these output delays would be input delays to the second stage and so on. Adding columns however, is equivalent to input delays, which immediately causes extra output delays also. Therefore, one wishes to add the minimal number of rows/columns. There are a number of questions: What is the minimum number of rows and columns to be added? How does it depend on the number of channels M? Can one obtain explicit formulas for these rows? What are the degrees of freedom in their choice?

The answer is that one has to append precisely (zero columns and) $(M-1)(K-1)$ rows, say $\mathbf{U}$, so that $\begin{bmatrix} \mathbf{U} \\ \mathbf{H} \end{bmatrix}$ is unitary. These rows will be called *entry* filters. The outputs of the entry filters are information used by the synthesis filters to reconstruct the input signal.

The analysis uses the fact that if $\mathbf{x}$ is right sided, so is $\breve{\mathbf{x}}$(given by $[x(M-1), \cdots, x(0), x(M), \cdots]$). Since the analysis filter bank gives out M outputs for every block of M inputs, the first output can only be expected at $n = (M-1)$ (we assume without loss of generality that all the filters are causal). In other words, the filter bank will give outputs at time instants $n = M-1, 2M-1, 3M-1, \ldots$. Let $\mathbf{d} = \mathbf{H}\mathbf{x}$ be the desired input/output behavior. If we define

$$\breve{\mathbf{H}} = \begin{bmatrix} h_p(K-1) & h_p(K-2) & \ldots & h_p(0) & 0 & \\ 0 & h_p(K-1) & \ldots & h_p(1) & h_p(0) & \ddots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \end{bmatrix}, \mathbf{V}_i = \begin{bmatrix} v_i v_i^T & I - v_i v_i^T & 0 & \ddots \\ & v_i v_i^T & I - v_i v_i^T & \\ \vdots & \vdots & \vdots & \end{bmatrix}, \tag{6}$$

then $\breve{\mathbf{H}} = \prod_{i=K-1}^{0} \mathbf{V_i}$ and $\mathbf{d} = \breve{\mathbf{H}}\breve{\mathbf{x}}$. In this case if $\breve{\mathbf{H}}\breve{\mathbf{H}}^T = \breve{\mathbf{H}}^T\breve{\mathbf{H}}I$, then one has a PR unitary

filter banks are designed from Eqn. 3 which should generically give a polynomial of degree $L-1$, we will assume $L = K$. The essential features of optimal time-varying filter bank trees can be seen in this case. However, when $L > K$ the optimal time-varying filter bank trees rely on a new factorization that is very similar to Eqn. 38. Section 6 characterizes situations when $L > K$. In practice this occurs in structured filter banks (and more generally unitary filter banks that are constructed without relying on Eqn. 3).

The trick is to use Eqn. 3 to induce the corresponding (matrix) factorization of $\breve{\mathbf{H}}$. If we define the doubly infinite matrices $\mathbf{V}_0 = diag(V_0)$, and

$$\mathbf{V}_i = \begin{bmatrix} & \vdots & \vdots & \vdots & \\ \cdots & v_i v_i^T & I - v_i v_i^T & 0 & \cdots \\ & 0 & v_i v_i^T & I - v_i v_i^T & \\ & \vdots & \vdots & \vdots & \end{bmatrix}, \quad \text{for } i \in \{1, \ldots, K-1\} \tag{4}$$

it follows that

$$\breve{\mathbf{H}} = \prod_{i=K-1}^{0} \mathbf{V}_i. \tag{5}$$

The factors $\mathbf{V}_i$, with a few modifications to make it time-varying will be fundamental building blocks for a factorization of arbitrary time-varying unitary filter banks.

In a tree structured (unitary) filter bank also, if we appropriately stack the outputs of all the analysis channels, one can define a corresponding $\breve{\mathbf{H}}$, that relates this output to the input. Once again the factorization in Eqn. 5, used for each of the various analysis banks in the tree induces a factorization of $\breve{\mathbf{H}}$. However, bookkeeping of the various types of factors may be involved depending on how one stacks the outputs of the channels. We will see later that this bookkeeping problem, when viewed just the right way ceases to be a problem after all! In any case, conceptually it is sufficient to consider just a single stage analysis filter bank.

Consider a single stage analysis filter bank. A fundamental assumption is that the filter bank operates on infinite input signals. Another assumption is that over the entire time horizon, the filter bank splits the input signal into M channels. Time varying filter banks relax the second assumption by allowing the following: over different time-segments of the input signal (usually non-overlapped, but sometimes overlapped) different analysis filter banks are used to split the signal. With time-varying filter banks we can remove the first assumption also since finite input signals are infinite signals non-zero on a particular segment, and one may let the filter operate on that particular segment.

related by the following doubly infinite block Toeplitz matrix:

$$\mathbf{d} = \mathbf{H}\mathbf{x} \stackrel{\text{def}}{=} \begin{bmatrix} & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \\ & h_0(N-1) & \dots & h_0(N-M-1) & \dots & h_0(0) & 0 & \\ & \vdots & \dots & \dots & \dots & \dots & \dots & \\ \ddots & h_{M-1}(N-1) & \dots & h_{M-1}(N-M-1) & \dots & h_{M-1}(0) & 0 & \ddots \\ & 0 & 0 & h_0(N-1) & \dots & \dots & \dots & \\ & \vdots & \vdots & \vdots & \dots & \dots & \dots & \\ & 0 & 0 & h_{M-1}(N-1) & \dots & \dots & \dots & \\ & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \end{bmatrix} \mathbf{x}. \qquad (1)$$

If the filter bank is to be perfect reconstruction (PR) $\mathbf{H}$ must be invertible. An important class of PR filter banks are the unitary ones characterized by the fact that $\mathbf{H}\mathbf{H}^T = \mathbf{H}^T\mathbf{H} = \mathbf{I}$. In particular, it follows that a *one* channel filter bank (where the input is leaked to the output) is unitary iff (modulo delays) $\mathbf{H} = \mathbf{I}$.

From an implementation point of view the best possible structure uses the polyphase representation [15, 4]. Moreover, this computational structure is also the fountainhead for the elegant factorization theory of the filters [15]. A fundamental tool in this paper will be the time-domain equivalent of this factorization (and its generalization).

Let $H_p(z)$ be the polyphase component matrix of the analysis filter bank. Generically N, the length of the filters is of the form MK and therefore the polyphase matrix is of the form $H_p(z) = \sum_{k=0}^{K-1} h_p(k)z^{-k}$. In order to obtain the factorization it is convenient to introduced a new sequence $\breve{\mathbf{x}} = [\cdots, x(0), x(-1), \cdots, x(-M+1), x(M), x(M-1), \cdots]$ obtained from $\mathbf{x}$ by a permutation. In terms of $\breve{\mathbf{x}}$ and the matrices $h_p(k)$ the output of the analysis filter bank is seen to be given by

$$\mathbf{d} = \breve{\mathbf{H}}\breve{\mathbf{x}} \stackrel{\text{def}}{=} \begin{bmatrix} & \vdots & \vdots & \vdots & \vdots & \vdots & \\ \ddots & h_p(K-1) & h_p(K-2) & \dots & h_p(0) & 0 & \ddots \\ & 0 & h_p(K-1) & \dots & h_p(1) & h_p(0) & \\ & \vdots & \vdots & \vdots & \vdots & \vdots & \end{bmatrix} \breve{\mathbf{x}}. \qquad (2)$$

Since $\mathbf{x}$ and $\breve{\mathbf{x}}$ are related by an obvious permutation, $\mathbf{H}$ is unitary iff $\breve{\mathbf{H}}$ is unitary. For unitary FIR filter banks $H_p(z)$ is of the form [15]:

$$H_p(z) = \prod_{i=L-1}^{1} \left[I - v_i v_i^T + z^{-1} v_i v_i^T \right] V_0, \qquad (3)$$

where v_i, $i \in \{1, \dots, L-1\}$ are unit norm vectors and V_0 is a unitary matrix. One has essentially factored the degree $K-1$ polynomial matrix into $L-1$ first order factors. Since *usually* unitary

can do efficient subband coding say), it is clear that one has to design rows corresponding to the transition period of the consolidated FIR analysis system. A similar analysis of the synthesis bank where one is interested in representing the output signal as a linear combination of basis functions shows that switching from one PR synthesis bank to another has to be done by designing columns of transition region of the consolidated system input/output matrix.

This conceptual distinction between transitions of analysis and synthesis filter banks becomes latent if one considers switching between filters of unitary filter banks (where the synthesis bank is the adjoint of the analysis filter bank). Here transition rows of the analysis bank becomes transition columns of the synthesis bank. All through this paper we will only consider unitary filter banks and therefore we will only consider time-varying analysis filter banks - theory, design and implementation.

3 Filter Banks for Finite and One-sided Signals

The theory of time-invariant filter banks had been developed in both the time and $\mathcal{Z}$-transform domains. While the time-domain theory ([13]) is flexible, design and implementation of good filter banks with many channels is cumbersome. The factorization-based theory of unitary filter banks in the $\mathcal{Z}$-transform domain ([15]), however, is practically efficient to design and implement. Moreover, the unitariness of these filter banks gives rise to orthonormal wavelet bases. Also, the factorization based approach can be easily specialized to cases of filter banks with different types of symmetry restrictions on the filters [14, 7]. A third powerful approach to PR filter bank design is based on the idea of modulated filter banks a general treatment of which is found in [6]. A natural approach to attack the time-varying filter bank seems to be in the time domain. However, we will shortly see that even in the time-domain the knowledge gained from the factorization in the $\mathcal{Z}$ domain when interpreted in the time domain with appropriate modification provides a solution to the general time-varying filter bank problem. However, in order to obtain the optimal solution, it is necessary to introduce a new factorization of polynomial matrices unitary on the unit circle. This factorization and the corresponding results will be given at the end of this paper. The basic ideas can be seen from the more popular McMillan degree based factorization (which has a disadvantage in that the number of factors is more than the order of the polynomial matrix).

Consider the filter bank in Fig. 1. Let the filters be causal FIR of length N, let $\mathbf{x} = [\cdots, x(0), x(1), \cdots]$ and $\mathbf{d} = [\cdots, d_0(0), d_1(0), \cdots, d_{M-1}(0), d_0(1), d_1(1), \cdots]$. Then, $\mathbf{x}$ and $\mathbf{d}$ are

weighted by the expansion coefficients $x(n)$. For instance consider trying to capture local polynomial behavior in the signal. In this case, one designs rows that act as basis functions of polynomials. Dually, if the input signal is to be smoothed, then each input sample can be replaced by samples of a polynomial locally. In this case one designs the columns of $\mathbf{H}$ to be the basis functions that replace the input samples/coefficients.

The above discussion is crucial to understanding what designing a time-varying filter bank means. An M-channel (time-invariant) filter bank is shown in Fig. 1. One designs the filter

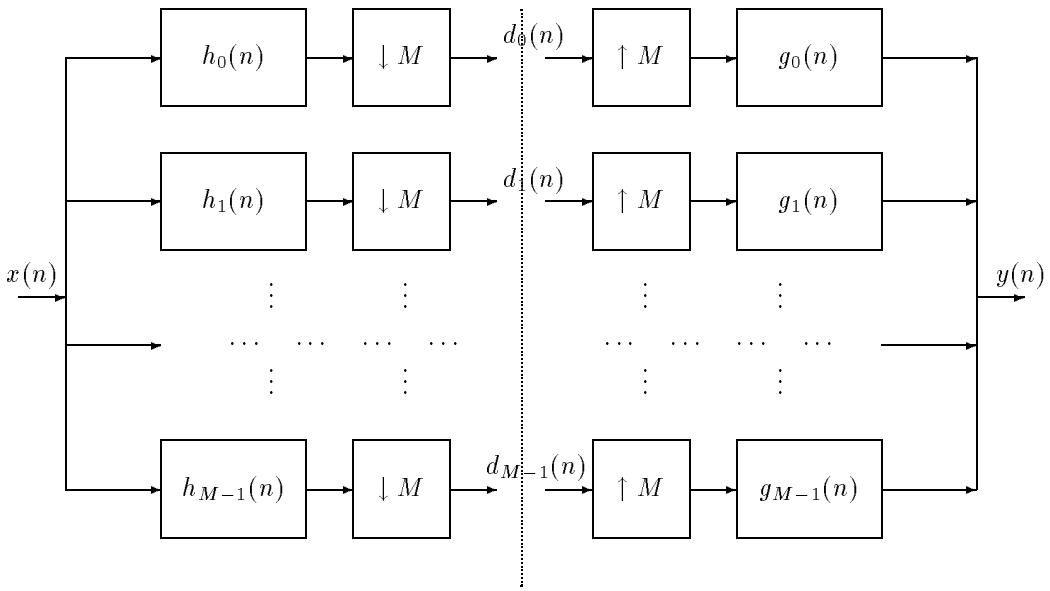

Figure 1: An M-channel Filter Bank

bank by choosing the analysis filters and synthesis filters, such that the overall system is perfect reconstruction (PR) and satisfies some objective criterion. For simplicity consider two M channel PR analysis filter banks. At some time instant, say $n = 0$, assume we wish to abruptly replace all the *analysis* filters of the first bank with that of the second. Since all filters are FIR, after an initial transition, the output would correspond to that of the second PR bank. In order to ensure the combined time-varying analysis bank is PR, at least intuitively it is clear that one has to incorporate additional outputs that *essentially drain out* the states of the first analysis bank at $n = 0$. Since the goal of the analysis bank is to capture trends in the signal $x(n)$ (so that one

(entry/exit) functions in the wavelet bases and the entry/exit filters of the corresponding filter bank with the scaling and wavelet vectors as *synthesis filters* (caveat: not analysis filters). This brings about a deep connection between the codimension of a truncated multiresolution analysis in its next finer approximation space (dimension of Γ_j) and the McMillan degree of the scaling and wavelet vectors, a fact that is far from obvious from a wavelets perspective.

Both the results on time-varying filter banks (and associated wavelet bases) naturally specialize to the case of *structured unitary* filter banks like unitary filter banks with symmetry [14, 7] and modulated filter banks [10, 11, 6, 5].

The outline of the paper is as follows. In Section 2 some issues regarding the design of time-varying linear systems are discussed. Section 3 introduces *a* time-domain factorization of unitary filter banks, and constructs PR filter banks for one-sided and finite signals. Time-varying PR filter bank trees are constructed in Section 4, while Section 5 shows how the most general time-varying unitary filter bank can be implemented with a simple and elegant algorithm. Section 6 generalizes these results to structured filter banks like modulated filter banks and filter banks with symmetry restrictions on the filters. A novel factorization of the polyphase matrix of a unitary filter bank is also given. In this case, we shown the crucial connection between the McMillan degree of the filter bank structure and the number of exit/entry entry filters. Time-varying wavelet tight frames and orthonormal bases are constructed in Section 7.

2 On Designing Time-Varying Systems

Consider the design of a (SISO) time-varying (FIR) system defined by $\mathbf{y} = \mathbf{H}\mathbf{x}$, where $\mathbf{H}$ is a doubly infinite matrix. Designing such a system is equivalent to designing the rows or columns of $\mathbf{H}$ (by the FIR assumption there are finitely many non-zero entries in any row/column). How does one choose whether to designs rows or columns? Note that the columns correspond to impulse responses of the system (to shifts of the delta impulse). To understand this better, consider the time-invariant case, where all columns are shifts of the first column and all rows are shifts of the first row. Clearly, in this case designing a row is the same as designing a column. However, in the time-varying case designing rows is different from designing columns.

The different choices depend upon the duality between considering the sequence of numbers $x(n)$ as a signal or a set of impulses (expansion coefficients). In the former case, $y(n)$ is the inner product of the signal corresponding to the basis functions made up of the rows of $\mathbf{H}$, while in the latter it is the linear combination of the basis functions (corresponding to the columns of $\mathbf{H}$)

torization result in this case, their results are based on detailed analysis of the filter coefficients across the boundaries of the input signal segments. In [8, 9], the analysis is done in the context of unitary filter banks. However, in that paper the authors *postulate* the form of the *boundary filters* and exhibit solutions. In our factorization approach, we exhibit formulas for the minimal set of canonical boundary (we call them entry and exit filters). The examples in [8, 9] have the same or more boundary filters that those constructed in this paper. Moreover, our factorization approach leads to efficient implementations of time-varying filter banks (including the entry/exit filters). Also the non-uniqueness of these entry/exit filters unravels itself nicely in our approach. An important consequence of our analysis is that there are as many canonical exit filters as the McMillan degree of the analysis polyphase matrix, and there are as many canonical entry filters as the McMillan degree of synthesis polyphase matrix.

Recently, (multiplicity 2) orthonormal wavelet bases for the half-line has been constructed by Yves Meyer, and an outline of his approach appears in [2]. The basic idea is to start with a multiresolution analysis giving rise to an ON wavelet basis for $L^2(\mathbb{R})$, and project each space in the multiresolution onto the half line. Using *border* scaling functions and wavelets this new multiresolution analysis is given an orthonormal basis. The authors in [8, 9] also construct time-varying wavelet bases for a finite interval exploiting the well-known relationship between unitary filter banks and orthonormal wavelet bases. Moreover, most of the results in that paper were obtained in the context of two channel filter banks (and trees constructed thereof). Let $W_{i,j}$ denote the multiresolution analysis corresponding to the scaling functions and wavelets in a multiplicity M orthonormal wavelet basis. Instead of defining the new spaces $W_{i,j}^{half}$ obtained by restricting the functions to be zero on the left half line, say, we approach the problem in a more basic manner. Let $\psi_0(t)$ be the scaling function and $\psi_i(t)$, $i \in \{1, \ldots, M-1\}$ the wavelets in a multiplicity M orthonormal wavelet basis. Let the scaling and wavelet vectors of this basis be denoted by h_i. If they of length N, then the scaling function (by translation if necessary) is supported in $[0, \frac{N-1}{M-1})$. We define the spaces $W_{i,j}^{+} = Span\{M^j \psi_i(M^j t - k) \mid k \geq 0\}$. Clearly $W_{i,j}^{+} \subset W_{0,j+1}^{+}$ and moreover $W_{0,j}^{+}$ is a multiresolution analysis for $L^2([0,\infty))$. Therefore, we ask the following natural question: what is the dimension of the smallest space, say Γ_j such that $W_{0,j+1}^{+} = W_{0,j}^{+} \oplus \ldots \oplus W_{M-1,j}^{+}$. We answer this question explicitly and construct an orthonormal basis for Γ_j (made of what we call *entry* functions) giving rise to a wavelet basis for the (right) half line. Similarly, for the left half line one can do the same (but this time with *exit* functions). Combining these two ideas we construct segmented orthonormal wavelet bases. Ideas to construct wavelet bases with overlapping transitions are also given. In our construction there is an one-to-one correspondence between

operational and for $n \geq 0$ another filter bank is operational. Such *switching* between filter banks will be called hard switching - examples of which can be found in [8, 9, 12]. Hard switching essentially switches off one filter bank, and switches on the other. This paper shows that the optimal way to switch from one *unitary* filter bank to another is equivalent to optimally switching off the first and *then* switching on the second. Moreover, under some conditions that relate the two filter banks, one can switch on the second filter bank before switching off the first - a new concept which we call *soft switching*. More difficult problems involve hard switching between arbitrary trees of filter banks with different number of channels, growing filter bank trees and pruning filter bank trees. This paper gives a solution for the most general problem of soft-switching (which includes hard switching as a special case) between arbitrary unitary filter bank trees. Moreover, independent of the depth and complexity of tree, we give a simple and efficient implementation of the time-varying filter bank tree. There are many different types of unitary filter banks depending on additional structure that one might impose on the filters - like modulated filter banks, linear phase filter banks etc. The design and implementation of these structured filter banks are quite different from that of general unitary filter banks. Even though the general theory developed in this paper may be applied to these structured problems, it is computationally advantageous to specialize the theory to these special types of filter banks. We prove that for any minimal (McMillan degree) cascaded realization of *any* type of structured unitary filter bank (that may potentially be discovered in the future), one can immediately obtain canonical entry and exit filters for switching on and off the filter bank. The basic result (which is very intuitive) is that switching off a filter bank is related to the McMillan degree of the analysis filter bank and switching on a filter bank is related to the McMillan degree of the synthesis bank.

Consider the following scenario: At $n = -\infty$ an arbitrary tree structured filter bank is operational. A stream of data enters at the root of the tree and streams out the output channels. At $n = 0$ one decides to prune a subtree, and simultaneously grow another subtree. Assume that the rationale based on which the original tree was designed becomes suspect at a later time instant and the entire extant tree is replaced by a newly designed tree. How does one do all this efficiently, economically and moreover without complicated bookkeeping to take care of all the channels? This paper gives an algorithmic solution to this, arguably most general, time-varying PR filter bank tree problem.

Time-varying filter banks have been studied recently [12, 8, 9]. In [12] the more difficult problem in the context of general (not unitary) filter banks is considered (the problem becomes intractable for moderate number of channels or depth of the tree). Since there is no known fac-

1 Introduction

This paper gives a factorization theory of time-varying unitary FIR filter banks. Filter banks are used to decompose signals into its frequency components. If one can reconstruct the signal from these components then the filter bank is said to be a perfect reconstruction (PR) filter bank. PR FIR filter banks give very general decomposition and recomposition results for arbitrary separable Hilbert spaces. Since filter banks have efficient computational algorithms, the decompositions also have corresponding fast algorithms. Filter banks can also be used in tree structures. A particular class of tree structured FIR unitary filter banks in an infinite tree gives rise to orthonormal wavelet bases (both octave band and M-band). These wavelet bases are of great interest in the mathematical community since they give unconditional bases for a number of the classical function spaces. Arbitrary tree structured filter banks also give rise to functions called wavelet packets. Wavelet packets, Short-Time Fourier Transform and the Wavelet Transform provide different ways to tile the time-frequency plane. These decomposition techniques are static in time.

Recently, it has been realized that one can construct arbitrary tilings of the time-frequency plane by allowing filter banks to be time-varying. There have been several attempts at constructing time-varying filter banks [8, 9, 12]. Besides the tiling of the time-frequency plane (in continuous-time), in discrete-time also time-varying filter banks are useful to do data-adaptive subband coding. Can flexible time-varying filter banks that are easy to design and implement be constructed? If so, how does one choose good filter bank tree topologies, and what criterion does one use to decide to change from one topology to another, and more importantly how can this be accomplished most economically?

This paper does not attempt to answer all these practical issues. It makes the fundamental assumption that at some time instant $n = 0$, one wants to *change* a PR filter bank tree into another while still maintaining PR. Why and how one comes to decide upon the change will not be addressed. However, once the decision is made, we show how the transition can be done optimally, give an explicit formula for the transition filters, and show how one can implement the transition with practically no additional computational cost relative to a static filter bank tree. Of course, the overhead of when the transitions is done must be known to the synthesis bank. Given the adaptive criterion that was used to decide the transition between filter banks during analysis is known to the synthesis banks can one avoid this overhead information? Probably not, but this question of using some backward adaptive scheme is currently being investigated.

A simple example of a time-varying filters bank is as follows: For $n < 0$, one filter bank is

FACTORIZATION APPROACH TO TIME-VARYING UNITARY FILTER BANK TREES

R.A.Gopinath and C.S.Burrus*
Department of Electrical and Computer Engineering,
Rice University, Houston, TX-77251

CML TR-92-23
3rd December '92

Abstract

A complete factorization of all optimal (in terms of quick transition) time-varying FIR unitary filter bank tree topologies is obtained. This has applications in adaptive subband coding, tiling of the time-frequency plane and the construction of orthonormal wavelet bases for the half-line and interval [8, 9, 12, 2]. A simple efficient implementation algorithm also comes with the factorization ensuring that even the most complex tree topology can be adapted with minimal overhead. Explicit formulas for entry/exit filters are given for *arbitrary tree transitions*. The results are independent of the number of channels and the length of the filters (as long as they are FIR), implying that some of the efficiency reasons for considering only binary time-varying trees is not valid any more. Time-varying wavelet bases (different bases for different segments of the real line) are also constructed.

Contact Address:
Ramesh A. Gopinath
Department of EE, A235
Rice University,
Houston, TX-77251
Phone (713) 527-8750 x3577 x3508
email: ramesh@rice.edu
fax : (713)-524-5237

*This work was supported by AFOSR under grant 90-0334 funded by DARPA and Bell-Northern Research

To be Submitted to IEEE Transactions in Signal Processing

FACTORIZATION APPROACH TO UNITARY TIME-VARYING FILTER BANKS

R.A.Gopinath and C.S.Burrus
Department of Electrical and Computer Engineering,
Rice University, Houston, TX-77251

CML TR-92-23
3rd December 1992