# FACTORIZATION APPROACH TO TIME-VARYING FILTER BANKS AND WAVELETS

*R.A.Gopinath**

Department of Electrical and Computer Engineering, Rice University, Houston, TX-77251

## 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 and wavelet packet bases for the half-line and interval [?, ?, ?, ?]. For an $M$-channel filter bank the factorization allows one to construct entry/exit filters that allow the filter bank to be used on finite signals without distortion at the boundaries. One of the advantages of our approach is that an efficient implementation algorithm comes with the factorization. The factorization can be used to generate filter bank tree-structures where the tree topology changes over time. Explicit formulas for the transition filters are obtained for *arbitrary tree transitions*. The results hold for tree structures where filter banks with any number of channels or filters of any length are used. Time-varying wavelet and wavelet packet bases are also constructed using these filter bank structures.

## 1. INTRODUCTION

Traditionally filter bank theory has treated signals of infinite extent. Specialization to finite extent signals have been attempted recently [?]. This paper extends this work to *arbitrary* tree-structured filter banks. The approach is novel and based on the following idea: time-varying filter banks are factored into a cascade of building blocks which can be implemented efficiently. By adding/removing these building blocks arbitrary time-varying filter bank trees can be constructed. Time-varying filter banks can be used in data-adaptive subband coding [?, ?]. They also give rise to adaptive wavelet bases which give arbitrary tilings of the time-frequency plane [?].

The results in this paper are obtained in addressing the problem of *switching* between filter banks. We show that optimal hard switching is equivalent to switching off the first filter bank and *then* switching on the second. Sometimes one can switch on the second filter bank without completely switching off the first - a concept we will call *soft switching*. More difficult problems include hard switching between arbitrary filter bank trees, growing filter bank trees and pruning filter bank trees. This paper solves the general problem of soft-switching between arbitrary filter bank trees. Several types of unitary filter banks exist - modulated filter banks [?, ?, ?, ?], linear phase filter banks [?, ?], etc. For any minimal (McMillan degree) cascaded realization of *any* type of structured unitary filter bank, one can obtain canonical entry/exit filters for switching on/off the filter bank. canonical entry/exit filters for switching on/off the filter bank. 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.

Let $W_{i,j}$ be the multiresolution analysis spaces of an $M$-band wavelet basis [?]. Define the spaces $W^+_{i,j} = Span\left\{M^j\psi_i(M^jt-k) \mid k \geq 0\right\}$ [?]. $W^+_{i,j} \subset W^+_{0,j+1}$ and $W^+_{0,j}$ is a multiresolution analysis for $L^2([0,\infty))$. What is the dimension of the smallest space, say $\Gamma_j$, such that $W^+_{0,j+1} = W^+_{0,j} \oplus \ldots \oplus W^+_{M-1,j} \oplus \Gamma_j$? We answer this question and construct an orthonormal basis for $\Gamma_j$. This gives a wavelet basis for the positive real line. By a similar analysis using *exit functions* we obtain a basis for the negative real line. Combing these two ideas one constructs wavelet and wavelet packet bases for the interval.

*This work was supported in part by BNR and in part by AFOSR under grant 90-0334 funded by DARPA

## 2. FILTER BANKS FOR FINITE AND ONE-SIDED SIGNALS

Consider an $M$-channel unitary filter bank with filters of length $N$. The analysis and syntheis bank equations are given by $\mathbf{d} = \mathbf{Hx}$ and $\mathbf{y} = \mathbf{H^T d}$. Also by unitariness $\mathbf{HH}^T = \mathbf{H}^T\mathbf{H} = \mathbf{I}$. Let $H_p(z)$, the polyphase matrix of the analysis filter bank [?] be given by $H_p(z) = \sum_{k=0}^{K-1} h_p(k)z^{-k}$. Then [?]:

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

where $v_i$, $i \in \{1, \ldots, L-1\}$ are unit norm vectors and $V_0$ is a unitary matrix. Also the McMillan degree $L \geq K$. We will consider the case $L = K$. Section 5. characterizes situation when $L > K$.

Let $\breve{\mathbf{x}} = [\cdots, x(0), x(-1), \cdots, x(-M+1), x(M), x(M-1), \cdots]$. Then there exists a unitary matrix $\breve{H}$ such that $\mathbf{d} = \breve{\mathbf{H}}\breve{\mathbf{x}}$. Eqn. 1 induces a factorization of $\breve{\mathbf{H}}$ (and hence $\mathbf{H}$). Indeed $\breve{\mathbf{H}} = \prod_{i=K-1}^{0} \mathbf{V}_i$, where $\mathbf{V}_i$ corresponds to $v_i$. The factors $\mathbf{V}_i$, with appropriate modifications, are fundamental building blocks for time-varying unitary filter banks.

For tree-structured filter banks also, by stacking the subband signals into a vector $\mathbf{d}$, one can define a matrix $\mathbf{H}$ that relates $\mathbf{x}$ and $\mathbf{d}$. From the factorization of each filter bank in the tree one obtains a factorization of $\mathbf{H}$. The bookkeeping of the various factors in the tree can be complicated depending on how one stacks the subband signals. Conceptually it suffices to consider a single filter bank.

### 2.1. Entry and Exit Filters

For right-sided signals the matrix for the analysis bank can be truncated to a right, lower semi-infinite matrix; for left-sided signals a left, upper semi-infinite matrix and for finite signals a rectangular matrix. In all these cases one has to add additional rows to the truncated matrices to obtain a unitary filter bank. For right-sided signals one has to add $(M-1)(K-1)$ rows (corresponding to *entry* filters), say $\mathbf{U}$, so that $\begin{bmatrix} \mathbf{U} \\ \mathbf{H} \end{bmatrix}$ is unitary. For left-sided signals one has to add $(K-1)$ rows (corresponding to *exit* filters), say $\mathbf{W}$, so that $\begin{bmatrix} \mathbf{H} \\ \mathbf{W} \end{bmatrix}$ is unitary. For finite signal one requires the square matrix $\begin{bmatrix} \mathbf{U} \\ \mathbf{H} \\ \mathbf{W} \end{bmatrix}$ is unitary. Entry and exit filter outputs are information used by the synthesis filters to reconstruct the input signal.

Let $A_i$ is the $(M-1) \times M$ matrix that orthonormalizes $(I - v_i v_i^T)$, and let $B_i = \begin{bmatrix} A_i(I - v_i v_i^T) & 0 & \ldots & 0 \end{bmatrix}$, and $C_i = \begin{bmatrix} 0 & \ldots & 0 & v_i^T \end{bmatrix}$, then $\begin{bmatrix} B_i \\ \mathbf{V}_i \\ C_i \end{bmatrix}$ is unitary. If $\mathbf{T}_0 = \mathbf{V}_0$

and

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

then

$$\prod_{i=K-1}^{0} \mathbf{T}_i = \begin{bmatrix} B_1\mathbf{V}_0 \\ \vdots \\ \breve{\mathbf{H}} \\ \vdots \\ C_1\mathbf{V}_0 \end{bmatrix}, \tag{2}$$

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 while $\mathbf{W}$ has $(K-1)$ rows of exit filters in $(K-1)$ sets. 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

$$\begin{bmatrix} \mathbf{u} \\ \mathbf{d} \\ \mathbf{w} \end{bmatrix} = \begin{bmatrix} \mathbf{U} \\ \mathbf{H} \\ \mathbf{W} \end{bmatrix} \mathbf{x}. \tag{3}$$

For example, in the 4 coefficient Daubechies' filters in [?] 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:

$$\begin{bmatrix} 0.8660 & 0.5000 & 0 & 0 \\ -0.1294 & 0.2241 & 0.8365 & 0.4830 \\ -0.4830 & 0.8365 & -0.2241 & -0.1294 \\ 0 & 0 & -0.5000 & 0.8660 \end{bmatrix}. \tag{4}$$

## 3. TIME-VARYING FILTER BANKS

Results above extend to general time-varying filter banks (where input signals are two-sided).

### 3.1. Hard Switching - 1 channel to $M$

The i/o map of such a time-varying filter bank is given by

$$\mathbf{d} = \left[\begin{array}{c|c} \mathbf{I} & 0 \\ 0 & \mathbf{U} \\ 0 & \mathbf{H} \end{array}\right] \mathbf{x}. \tag{5}$$

The transition is effected by *entry* filters.

### 3.2. Hard Switching - $M$ channel to 1

In this case the i/o map for time-optimal transition is

$$\mathbf{d} = \left[\begin{array}{c|c} \mathbf{H} & 0 \\ \mathbf{W} & 0 \\ 0 & \mathbf{I} \end{array}\right] \mathbf{x}. \tag{6}$$

### 3.3. Hard Switching - $M_1$ channel to $M_2$ channel

Now the transition filters are given by the exit filters of the first filter bank and the entry filters of the second.

$$\mathbf{d} = \left[\begin{array}{c|c} \mathbf{H_1} & 0 \\ \mathbf{W_1} & 0 \\ 0 & \mathbf{U_2} \\ 0 & \mathbf{H_2} \end{array}\right] \mathbf{x}. \tag{7}$$

The transition is abrupt (i.e., the filters are do not overlap). Overlapping transitions are obtained by using any orthogonal basis for the row space of $\left[\begin{array}{c|c} \mathbf{W_1} & 0 \\ 0 & \mathbf{U_2} \end{array}\right]$ as the transition filters.

### 3.4. Growing a Filter Bank Tree

Growing a tree is equivalent to switching from a one-channel to, say, an $M$-channel filter bank at the point of tree growth. Fro example consider growing an $M_2$ channel filter bank in the lowpass channel of an $M_1$ channel filter bank. If $\mathbf{P}$ be the matrix that selects the lowpass outputs of the first filter bank. Then the output of that portion of the the time-varying filter bank tree is related to the input as

$$\mathbf{d} = \begin{bmatrix} \mathbf{I} & 0 \\ 0 & \mathbf{U_2} \\ 0 & \mathbf{H_2} \end{bmatrix} \mathbf{P}\mathbf{H_1}\mathbf{x}. \tag{8}$$

Transition filters are given by $\mathbf{U_2}\mathbf{P}\mathbf{H_1}$. There are as many transition filters as entry filters of the second filter bank. Transition filters for arbitrary tree-growth can be obtained algorithmically. The crucial point 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. Let the mapping from the input to the output "growth" channel be as shown in Fig. 2. The transition filters are given by the system in Fig. 3 which is driven by the entry filters of the newly added filter bank. Transition filters are non-unique.

### 3.5. 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 an $M_1$ channel filter bank. The mapping from input to the output of the section of the tree to be pruned is given by $\mathbf{H_2}\mathbf{P}\mathbf{H_1}$. Pruning is accomplished as follows:

$$\mathbf{d} = \begin{bmatrix} \mathbf{H_2} & 0 \\ \mathbf{W_2} & 0 \\ 0 & \mathbf{I} \end{bmatrix} \mathbf{P}\mathbf{H_1}x. \tag{9}$$

There are as many transition filters as entry filters of $\mathbf{H_2}$. For arbitrary 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.

### 3.6. Hard Switching Between Arbitrary Subtrees

One can switch between arbitrary trees by pruning the first tree (till 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). If the two trees differ only in a subtree, then one can prune and grow the subtree only. Under special circumstances a mode of switching known as soft-switching can be used to give better results. In every case transition filters are the *pullback* of the entry/exit filters of the added/pruned filter bank by the extant tree preceding the transition.

## 4. IMPLEMENTATION OF TIME-VARYING TREES

Bookkeeping for a deep tree with several transitions can get complicated. This can be avoided and moreover, time-varying filter bank trees can be implemented with the same complexity as a fixed filter bank trees.

Eqn. 1 shows that a unitary filter bank can be implemented using a cascade of building blocks one corresponding to each factor. Now consider switching from a one channel filter bank to an $M$ channel filter bank at $n = 0$. For $n < 0$, the input is fed to the output. For $n \geq 0$ for each block of $M$-inputs one can add one factor in Eqn. 1 so that in $K-1$ steps the $M$-channel filter bank is operational. Each added block has its *state* set to zero (i.e., the delays corresponding to $z^{-1}$) for one instant. The corresponding output is $(I - v_i v_i^T)$ times the input $M$-vector. This transition output can be remembered as $M-1$ numbers (since rank of $I - v_i v_i^T$ is $M-1$). These are precisely the output of the entry filters. Now consider switching from an

$M$-channel filter bank to a one channel filter bank at $n = 0$. At $n = 0$ each building block of the filter bank has states stored in the storage elements. One easily sees that the exit filter outputs are just these states. As such entry and exit filters can be implemented *without ever knowing* them. The efficiency of this implementation for fixed filter banks naturally carries over to the time-varying case.

### 4.1. Soft Switching

Consider switching from an $M_1$ channel filter bank to an $M_2$ channel filter bank. If the two filter banks have common factors (building blocks) one can remove/add appropriate blocks (rather than completely switching off one filter bank and switching on the next). This concept called *soft switching* is time-efficient and can be used anywhere in a filter bank tree. As always the transition filter is the pullback of the entry filter corresponding to adding each block. A hard switch between filter bank trees then becomes a sequence of soft switches.

Soft switching can lead to new filter bank design problems. One could 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. Depending on the input data one could use shorter or longer filter banks by soft switching between them.

## 5. HIGHER ORDER BUILDING BLOCKS FOR STRUCTURED FILTER BANKS

Until now we assumed $L = K$ (in Eqn. 1). When $L > K$ also the theory works. However, the switching is not optimal - i.e., one can switch faster. For optimal switching we introduce a novel *unique* factorization of $H_p(z)$ [?]:

$$H_p(z) = \prod_{i=1}^{K-1} F_i(z) V_0, \tag{10}$$

where $F_i(z) = I - f_i + z^{-1} f_i$ and $f_i$ is an orthogonal projection matrix of rank $\delta_i$. $F_i(z)$ is of polynomial degree one and McMillan degree $\delta_i$. Since $f_i$ is of rank $\delta_i$ one can write $f_i = u_i u_i^T$, where $u_i$ is some (non-unique) $M \times \delta_i$ matrix. Eqn. 10 induces a time-domain factorization that gives rise to the optimal factorization of time-varying filter banks. In effect one replaces the vector $v_i$ by the matrix $u_i$ in the theory developed earlier. In this case for the $i^{th}$ factor there are $M - \delta_i$ entry filters and $\delta_i$ exit filters. Let $H_p^j(z) = \prod_{i=1}^{j-1} F_i(z) V_0 = \sum_{i=0}^{j-1} h_p^j(i) z^{-i}$. Then the entry and exit filters are given by

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

$$A_j (I - f_j) \left[ \begin{array}{cccc} h_p^j(0) J & h_p^j(1) J & \ldots & h_p^j(j-1) J \end{array} \right]. \tag{11}$$

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

$$u_j \left[ \begin{array}{cccc} h_p^j(0) J & h_p^j(1) J & \ldots & h_p^j(j-1) J. \end{array} \right], \tag{12}$$

Let $\Delta = \sum_{i=1}^{K-1} \delta_i$. $\Delta$ is the McMillan degree of $H_p(z)$. The number of exit filters is precisely $\Delta$. Canonical exit filter outputs are precisely the entries in the storage elements at the switching instant. Notice that even with Eqn. 1 one has the same number of exit filters. However, in the earlier factorization the number of entry+exit filters is $M(L-1)$, while in the new factorization it is $M(K-1)$ and $L \geq K$. Therefore there are less entry filters (which cost time since it means that much more input should be read before switching is complete).

This approach is optimal if the filter lengths are precisely $N = MK$ for some $K$. One can modify this factorization for general $N$ also [?]. The result is that there are $\Delta$ exit filters and $N - M - \Delta$ entry filters. The number of entry and exit filters are the McMillan degrees of $H_p(z^{-1})$ and $H_p(z)$ respectively.

The factorization approach can be specialized to the case of structured filter banks like modulated filter banks [?] and linear-phase filter banks [?, ?]. More generally for any unitary filter bank structure that has a cascaded minimal factorization of both $H_p(z)$ and $H_p(z^{-1})$ into first order factors, one has a factorization of the corresponding time-varying filter bank. For each factor $p + qz^{-1}$ there are entry and exit filters. The entry filters are given by any ON basis for the rowspace of $p$ while the exit filters are given by any ON basis for the rowspace of $q$. The ranks of $p$ and $q$ give the number of entry and exit filters respectively. Implementation of the time-varying filter bank is also immediate. One can start with any (structured) minimal implementation of a filter bank and *effectively compute outputs of entry and exit filters without ever knowing what the filters are.* However, in some applications like the construction of time-varying wavelet bases (for example), one must know the entry/exit in order to define transition functions for these bases.

## 6. TIME-VARYING WAVELET PACKET BASES

By taking of the effective input/output map of an arbitrary unitary time-varying filter bank tree one readily obtains time-varying discrete-time wavelet packet bases. We construct *continuous-time* time varying wavelet and wavelet packet bases using entry/exit functions.

### 6.1. Wavelet Bases for $L^2([0, \infty))$

An $M$ channel unitary filter bank (with synthesis filters $\{h_i\}$) such that $\sum_n h_0(n) = \sqrt{M}$ gives rise to an $M$-band wavelet tight frame $L^2(\mathbb{R})$ [?]. The scaling function $\psi_0(t)$ and wavelets $\psi_i(t)$, $i \in \{1, \ldots, M-1\}$ are related to these synthesis filters as

$$\psi_i(t) = \sqrt{M} \sum_{k=0}^{N-1} h_i(k) \psi_0(Mt - k). \tag{13}$$

$h_0(n)$ is the scaling filter and $\{h_i(n) \mid i \neq 0\}$ are the wavelet filters. The scaling function and wavelets are supported in $[0, \frac{N-1}{M-1}]$, where $N$ is the length of the filters. If

$$W_{i,j} = Span\,\{\psi_{i,j,k}\} \stackrel{\text{def}}{=} \left\{ M^{j/2} \psi_i(M^j t - k) \right\} \quad \text{for } k \in \mathbf{Z}, \tag{14}$$

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

$$W_{0,j} = W_{0,j-1} \oplus W_{1,j-1} \ldots \oplus W_{M-1,j-1} \qquad \forall j \in \mathbf{Z}. \tag{15}$$

For $k \in \mathbb{N}$, support of $\psi_{i,j,k}(t)$ is in $[0, \infty)$. With this restriction (in Eqn. 14) define the spaces $W_{i,j}^+$. As $j \to \infty$ (since $W_{0,j} \to L^2(\mathbb{R})$) $W_{0,j}^+ \to L^2([0, \infty))$. Hence it suffices to have a multiresolution analysis for $W_{0,j}^+$ to get a wavelet basis for $L^2([0, \infty))$. Eqn. 15 does not hold with $W_{i,j}$ replaced by $W_{i,j}^+$ because $W_{0,j}^+$ is *bigger* than the direct sum of the constituents at the next coarser scale. Let $\mathcal{U}_{j-1}$ be this *difference* space:

$$W_{0,j}^+ = W_{0,j-1}^+ \oplus W_{1,j-1}^+ \ldots \oplus W_{M-1,j-1}^+ \oplus \mathcal{U}_{j-1} \tag{16}$$

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

We proceed as follows. Construct entry filters (for the analysis filters) of the filter bank with synthesis filters $\{h_i\}$. Time-reverse them to obtain entry filters (for the synthesis filters). If $\Delta$ is the McMillan degree of the synthesis bank, there are $\Delta$ entry filters. Let $u_i(n)$ denote the $i^{th}$ synthesis entry filters. Define the *entry* functions

$$\mu_l(t) = \sqrt{M} \sum_{k=0}^{L_l - 1} u_l(k) \psi_0(Mt - k), \qquad l \in \{0, \ldots, \Delta - 1\}. \tag{17}$$

$\mu_i(t)$ is *compactly supported* in $[0, \frac{L_l-1}{M} + \frac{N-1}{M-1}]$. Let $\mathcal{U}_j = Span\{\mu_{l,j}\} \stackrel{\text{def}}{=} Span\left\{M^{j/2}\mu_i(M^j t)\right\}$. By considering one stage of the analysis and synthesis stages of this PR filter bank on right sided signals) it readily follows that Eqn. 16 holds. Therefore the union of $\{\psi_{i,j,k} \mid i \in \{1,\ldots,M-1\}, j \in \mathbf{Z}, k \in \mathbb{N}\}$ and $\{\mu_{i,j} \mid i \in \{0,\ldots,\Delta-1\}, j \in \mathbf{Z}\}$ forms a wavelet tight frame for $L^2([0,\infty))$. If on started with an ON basis for $L^2(\mathbb{R})$ the newly constructed basis is an ON basis for $L^2([0,\infty))$. Indeed if $\{\psi_0(t-k)\}$ is an orthonormal system

$$\int_{t\geq 0} \mu_l(t)\psi_i(t-n)\,dt = \sum_k u_l(k)h_i(Ml+k) = 0, \qquad (18)$$

and

$$\int_{t\geq 0} \mu_l(t)\mu_m(t)\,dt = \sum_k u_l(k)u_m(k) = 0. \qquad (19)$$

The dimension of $\mathcal{U}_j$ is precisely the McMillan degree of the polyphase component matrix of the scaling and wavelet filters *considered as the filters of the synthesis bank*. 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. 1 shows the scaling function, wavelet, their integer translates and the single entry function corresponding to Daubechies four coefficient wavelets. In this case, $u_0 = \{-\sqrt{3}/2, 1/2\}$. One could start

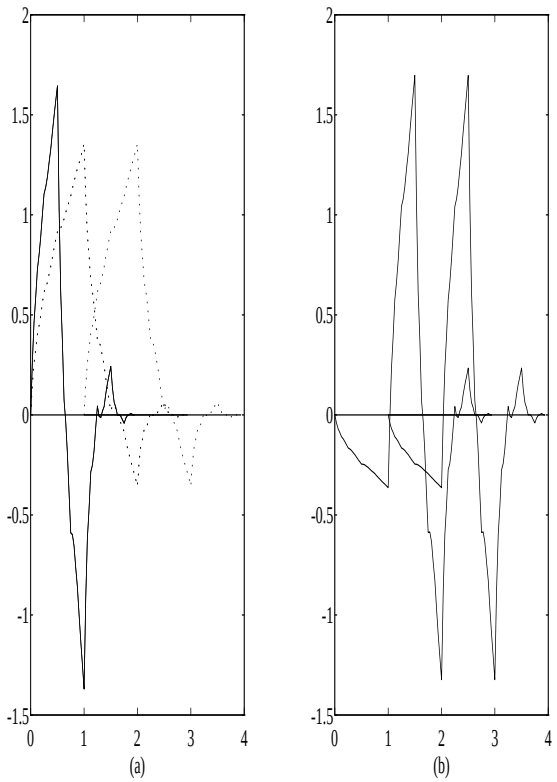


Figure 1. (a) Entry function $\mu_0(t)$, $\psi_0(t)$ and $\psi_0(t-1)$ (b) Wavelet $\psi_1(t)$ and $\psi_1(t-1)$

with a wavelet basis for $L^2([0,\infty))$ and reflect all the functions about $t=0$. This is equivalent to swapping the analysis and synthesis filters of the filter bank. Equivalently this is equivalent to using exit filters instead of entry filters above.

### 6.2. Time-Varying Wavelet Packet Bases

Using the ideas above one can construct wavelet bases for the interval and consequently segmented wavelet bases for $L^2(\mathbb{R})$. One write $\mathbb{R}$ as a disjoint union of intervals and uses a different wavelet basis in each interval. Each interval is will be spanned by a combination of scaling functions, wavelets, and corresponding entry and exit functions. 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. Extension of these ideas to obtain segmented *wavelet packet* bases is also immediate.

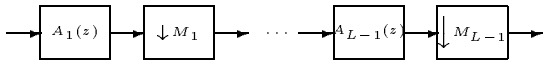


Figure 2. A Branch of an Existing Tree

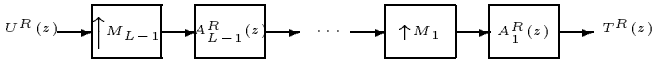


Figure 3. Transition Filter For Tree Growth

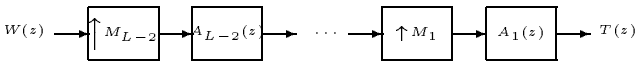


Figure 4. Transition Filter For Pruning

# FACTORIZATION APPROACH TO TIME-VARYING FILTER BANKS AND WAVELETS

*R.A.Gopinath*[1]

Department of Electrical and Computer Engineering, Rice University, Houston, TX-77251

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 and wavelet packet bases for the half-line and interval [?, ?, ?, ?]. For an $M$-channel filter bank the factorization allows one to construct entry/exit filters that allow the filter bank to be used on finite signals without distortion at the boundaries. One of the advantages of our approach is that an efficient implementation algorithm comes with the factorization. The factorization can be used to generate filter bank tree-structures where the tree topology changes over time. Explicit formulas for the transition filters are obtained for *arbitrary tree transitions*. The results hold for tree structures where filter banks with any number of channels or filters of any length are used. Time-varying wavelet and wavelet packet bases are also constructed using these filter bank structures.