23

Analysis and Fast Implementation of Oversampled Modulated Filter Banks

Stephan Weiss
Dept. Electronics & Computer Science, University of Southampton, UK

Abstract. Oversampled modulated filter banks (OSFBs) are popularly employed for a number of applications such as acoustic echo cancellation in order to reduce the processing complexity of a signal processing algorithm. Hence, an efficient implementation of OSFBs themselves is mandatory. In this paper, a polyphase description is used to remove redundancies in the filter operations and to factorise the OSFB into filter components depending on the prototype filter, and the modulating transform. Based on a state-space representation of this derived polyphase factorisation, signal flow graphs can be obtained which permit a very simple and efficient OSFB implementation. The analysis is performed for a number of different classes of OSFBs, and a comparison to existing methods is drawn.

1 Introduction

Oversampled filter banks (OSFBs) find a wide range of applications, where computational reductions for resource-demanding signal processing algorithms are sought by means of subband approaches. Examples include subband adaptive filters used in acoustic echo control [1; 2], line enhancement [3], or beamforming [4]. Another use of OSFBs are, for example, transmultiplexers for the transmission of several users over a single channel [5]. Particularly in the light of computational efficiency, simple low complexity realisations for the filter banks themselves are therefore desirable. Despite this motivation and in contrast to their critically decimated counterparts [6; 7], numerically efficient implementations of non-critically sampled (or "oversampled") filter banks have received little attention.

A simple filter bank system giving K subband signals decimated by $N \leq K$ is shown in Fig. 1. An efficient implementation is based on modulation of all K analysis filters $H_k(z)$ and all synthesis filter $G_k(z)$ from one prototype filter [8]. For OSFBs with non-critical decimation $N < K$, an efficient implementation has been reported by Wackersreuther [9], where a time domain approach leads

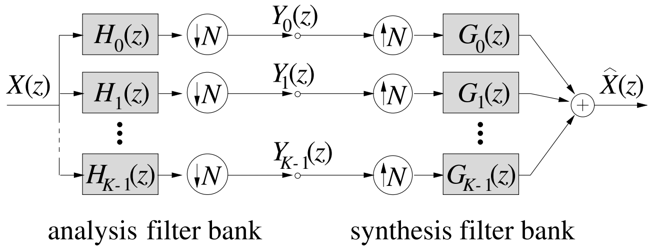

FIG. 1. Analysis and synthesis filter bank for subband decomposition of $X(z)$.

to a factorisation of the analysis filter bank operation into a filtering operation linked to the prototype filter coefficients, a cyclical shift, and the applications of the appropriate modulating transform (e.g. a DFT). A similar approach resulting in a different sequence of execution is presented in [10; 5], with a time-varying excitation of different components of the prototype filter followed by the modulating transform. More recently, polyphase factorisations in the z-domain have been presented [11; 12], which also permit a separation into filtering operations based on the prototype filter, and the modulating transform. For all cases [9; 10; 11; 12; 5], a dual implementation can be found for the synthesis filter bank operation.

The polyphase approach [11; 12] can be utilised as a starting point to derive a filter bank factorisation yielding a very simple and low cost implementation [13]. Here, this factorisation is generalised to arbitrary length prototype filters and arbitrary modulations, for which the analysis and factorisation is presented in Section 2. Based on a state-space representation of the OSFB operations in Section 3, signal flow graphs for analysis and synthesis filter bank are derived in Section 4. Further, in the latter the implementation and computational complexity of the resulting circuits is discussed and compared to existing methods.

In our notation, boldface uppercase variables are matrix valued, boldface lowercase or uppercase underlined quantities refer to vector valued variables. An $N \times N$ identity matrix is denoted by $\mathbf{I}_N$, an $N \times M$ matrix with zero elements by $\mathbf{0}_{N\times M}$.

2 Filter Bank Analysis

2.1 Polyphase Notation

Let us consider the analysis filter bank of Fig. 1 producing K subband signals. To exploit computational redundancies arising from the decimation by N, a polyphase description is utilized [7]. A polyphase notation for the kth analysis filters,

$$H_k(z) = \sum_{n=0}^{N-1} z^{-n} H_{k|n}(z^N) \quad , \tag{2.1}$$

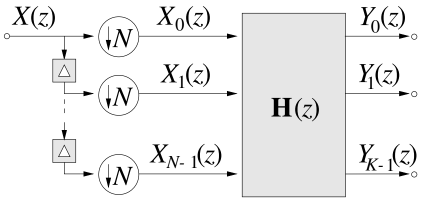

FIG. 2. Analysis filter bank with demultiplexer and $\mathbf{H}(z)$ describing an $N \times K$ MIMO system.

produces a decomposition into N type-I polyphase components $H_{k|n}(z)$ [7]. Similarly, the input signal $X(z)$ is decomposed into N type-II polyphase components $X_n(z)$,

$$X(z) = \sum_{n=0}^{N-1} z^{-N+n-1} X_n(z^N) \quad . \tag{2.2}$$

If the polyphase components are organised in vector form,

$$\underline{H}_k(z) = \left[H_{k|0}(z) \;\; H_{k|1}(z) \;\; \cdots \;\; H_{k|N-1}(z)\right]^{\mathrm{T}} \tag{2.3}$$

$$\underline{X}(z) = \left[X_0(z) \;\; X_1(z) \;\; \cdots \;\; X_{N-1}(z)\right]^{\mathrm{T}} \tag{2.4}$$

a subband signal $Y_k(z)$ can be denoted as

$$Y_k(z) = \underline{H}_k^{\mathrm{T}}(z) \cdot \underline{X}(z) \quad . \tag{2.5}$$

For compatibility, in the following we assume that filters are always subjected to type-I and signals to type-II polyphase decompositions.

2.2 Analysis Filter Bank

For a compact notation of the analysis filter bank operations, the K subband signals are collected in a vector $\underline{Y}(z) = [Y_0(z) \; Y_1(z) \; \cdots \; Y_{K-1}(z)]^{\mathrm{T}}$. Inserting (2.5) gives

$$\underline{Y}(z) = \left[\underline{H}_0(z) \;\; \underline{H}_1(z) \;\; \cdots \;\; \underline{H}_{K-1}(z)\right]^{\mathrm{T}} \cdot \underline{X}(z) \tag{2.6}$$

$$= \mathbf{H}(z) \cdot \underline{X}(z) \quad , \tag{2.7}$$

where $\mathbf{H}(z) \in \mathbb{C}^{K \times N}(z)$ is the polyphase analysis matrix [11]. With the description (2.7), the analysis filter bank in Fig. 1 can be implemented by a demultiplexer followed by a linear time-invariant multi-input multi-output (MIMO) system $\mathbf{H}(z)$ as shown in Fig. 2. We are now interested in a particular factorisation of $\mathbf{H}(z)$.

It is assumed that the analysis filters are FIR with L_{p} coefficients, and derived from a prototype filter $P(z)$ by modulation. With the coefficients of the kth analysis filter organised in a vector $\mathbf{h}_k \in \mathbb{C}^{L_{\mathrm{p}}}$,

$$\mathbf{h}_k = \left[h_k[0] \;\; h_k[1] \;\; \cdots \;\; h_k[L_{\mathrm{p}}-1]\right]^{\mathrm{T}} \quad , \tag{2.8}$$

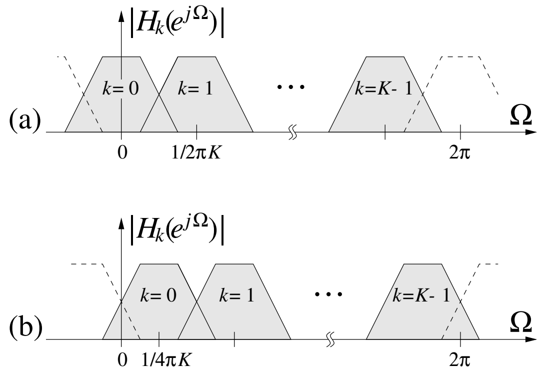

FIG. 3. (a) Even stacked and (b) odd stacked K-channel filter bank.

the polyphase components in (2.3) can be written as

$$\underline{H}_k(z) = \underbrace{\begin{bmatrix} \mathbf{I}_N & z^{-1}\mathbf{I}_N & \cdots & z^{-\lfloor L_\mathrm{p}/N \rfloor + 1}\mathbf{I}_N & \begin{matrix} z^{-\lfloor L_\mathrm{p}/N \rfloor}\mathbf{I}_R \\ \mathbf{0}_{N-R\times R} \end{matrix} \end{bmatrix}}_{\mathbf{L}_1^\mathrm{T}(z)} \cdot \mathbf{h}_k \qquad (2.9)$$

where $\mathbf{I}_N$ is an $N \times N$ identity matrix, $\lfloor \cdot \rfloor$ the floor operator, and $R = \mathrm{mod}_N L_\mathrm{p}$ the remainder of the division of the filter length L_p by the decimation factor N. The dependency of $H_k(z)$ on the underlying prototype filter with coefficients $p[i]$, $i = 0 \ldots L_\mathrm{p}-1$ is incorporated as

$$\mathbf{h}_k = \underbrace{\begin{bmatrix} p[0] & & & \mathbf{0} \\ & p[1] & & \\ & & \ddots & \\ \mathbf{0} & & & p[L_\mathrm{p}-1] \end{bmatrix}}_{\mathbf{P}} \cdot \underbrace{\begin{bmatrix} t_k[0] \\ t_k[1] \\ \vdots \\ t_k[L_\mathrm{p}-1] \end{bmatrix}}_{\mathbf{t}_k} \quad , \qquad (2.10)$$

based on the generally complex modulation sequence contained in $\mathbf{t}_k \in \mathbb{C}^{L_\mathrm{p}}$. The matrix $\mathbf{P} \in \mathbb{R}^{L_\mathrm{p} \times L_\mathrm{p}}$ is a diagonal matrix holding the prototype filter coefficients.

Depending on which modulation is invoked for the filter bank, different periodicities of the sequence $t_k[i]$, $i = 0 \ldots L_\mathrm{p}-1$, result. If the filter bank is even stacked as shown in Fig. 3(a), the periodicity is K. In this case, a compact notation of the modulation sequence can be found

$$\mathbf{t}_k = \underbrace{\begin{bmatrix} \mathbf{I}_K & \mathbf{I}_K & \cdots & \mathbf{I}_K & \begin{matrix} \mathbf{I}_S \\ \mathbf{0}_{K-S\times S} \end{matrix} \end{bmatrix}^\mathrm{T}}_{\mathbf{L}_2^\mathrm{T}} \cdot \tilde{\mathbf{t}}_k \quad , \qquad (2.11)$$

where $\tilde{\mathbf{t}}_k \in \mathbb{C}^K$, $\mathbf{L}_2 \in \mathbb{N}^{L_\mathrm{p} \times K}$, and $S = \mathrm{mod}_K L_\mathrm{p}$. Odd-stacked filter banks as in Fig. 3(b) exhibit a $2K$ periodicity of $t_k[i]$. Additional symmetries in the modulation sequence can however be exploited by alternating sign changes on blocks of K coefficients,

$$\mathbf{t}_k = [\mathbf{I}_K \quad -\mathbf{I}_K \quad \mathbf{I}_K \quad -\mathbf{I}_K \quad \cdots]^{\mathrm{T}} \cdot \tilde{\mathbf{t}}_k \quad . \tag{2.12}$$

Instead of modifying the matrix $\mathbf{L}_2$ of odd stacked filter banks, this sign change can be incorporated into $\mathbf{P}$ in (2.10) and hence the prototype filter $P(z)$ — a trick that is also known from discrete cosine transform implementations [7]. It is assumed that for odd stacked filter banks such a modification of the prototype filter is performed by negating the coefficients' signs in every second K-block of coefficients. With this assumptions, in the following even and odd stacked filter banks can be treated alike using (2.11).

The modulation sequences $\tilde{\mathbf{t}}_k$, $k = 0 \ldots K-1$, are collected in a matrix

$$\mathbf{T} = [\tilde{\mathbf{t}}_0 \quad \cdots \quad \tilde{\mathbf{t}}_{K-1}]^{\mathrm{T}} \quad \in \mathbb{C}^{K \times K} \tag{2.13}$$

which for example for a DFT modulated filter bank would be a K-point DFT matrix. Applying (2.13) to the substitution of (2.11) and (2.10) into (2.6) gives

$$\mathbf{H}(z) = \mathbf{T} \cdot \mathbf{L}_2 \cdot \mathbf{P} \cdot \mathbf{L}_1(z) \tag{2.14}$$

as notation for the polyphase analysis matrix. With (2.14) a factorisation into prototype filter components and a rotation by a transform matrix $\mathbf{T}$ has been established similar to [11; 12]. The difference is that the diagonal matrix $\mathbf{P}$ contains no sparse filters but only the prototype filter coefficients, which will be exploited for the implementation in Section 4.

2.3 Synthesis Filter Bank

Dual to the analysis filter bank, the synthesis filter bank as shown in Fig. 1 upsamples the subband signals by a factor N and applies interpolation filters $G_k(z)$. The condition that all filters $G_k(z)$ and $H_k(z)$ are derived from the same prototype lowpass filter and that the filter bank is perfectly reconstructing is guaranteed by $\mathbf{H}(z)$ being paraunitary [11]. Reconstruction is then given by the polyphase synthesis matrix $\mathbf{G}(z) \in \mathbb{C}^{N \times K}(z)$, which is the parahermitian of $\mathbf{H}(z)$, and relates the subband samples back to the polyphase components of the fullband signal,

$$\underline{\hat{X}}(z) = \mathbf{G}(z) \cdot \underline{Y}(z) \quad . \tag{2.15}$$

For causality, a delay has to be introduced such that $\hat{X}(z) = z^{-L_{\mathrm{p}}+1} \cdot X(z)$ in Fig. 1. In the N-polyphase domain, this is expressed as [7]

$$\underline{\hat{X}}(z) = \underbrace{\begin{bmatrix} \mathbf{0}_{N-R \times R} & z^{-\lfloor L_{\mathrm{p}}/N \rfloor + 1} \cdot \mathbf{I}_{N-R} \\ z^{-\lfloor L_{\mathrm{p}}/N \rfloor} \cdot \mathbf{I}_R & \mathbf{0}_{R \times N-R} \end{bmatrix}}_{\Delta(z)} \cdot \underline{X}(z) \quad . \tag{2.16}$$

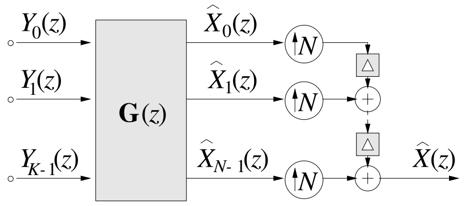

FIG. 4. Synthesis filter bank with $\mathbf{G}(z)$ describing a $K \times N$ MIMO system followed by a multiplexer.

Incorporating this delay into the perfect reconstruction condition, $\mathbf{G}(z) \cdot \mathbf{H}(z) = \Delta(z)$, the polyphase synthesis matrix is given by

$$\begin{aligned} \mathbf{G}(z) &= \Delta(z) \cdot \mathbf{H}^{\mathrm{H}}(z^{-1}) & (2.17) \\ &= \Delta(z) \cdot \mathbf{L}_1^{\mathrm{T}}(z^{-1}) \cdot \mathbf{P} \cdot \mathbf{L}_2^{\mathrm{T}} \cdot \mathbf{T}^{\mathrm{H}} \quad . & (2.18) \end{aligned}$$

Evaluating $\Delta(z) \cdot \mathbf{L}_1^{\mathrm{T}}(z^{-1})$ in (2.18) gives

$$\hat{\mathbf{L}}_1^{\mathrm{T}}(z) = \Delta(z) \cdot \mathbf{L}_1^{\mathrm{T}}(z^{-1}) = \mathbf{L}_1^{\mathrm{T}}(z) \cdot \mathbf{J}_{L_{\mathrm{p}}} \quad , \tag{2.19}$$

where $\mathbf{J}_{L_{\mathrm{p}}}$ is an $L_{\mathrm{p}} \times L_{\mathrm{p}}$ reverse identity matrix left-right flipping the causal matrix $\mathbf{L}_1^{\mathrm{T}}(z)$. The polyphase synthesis matrix in (2.18) leads to the signal flow graph in Fig. 4 with the MIMO system $\mathbf{G}(z)$ followed by a multiplexer. Its functionality is identical to the original synthesis filter bank in Fig. 1, but multiplications with expanding zeros in the interpolations filters are avoided.

Paraunitarity of $\mathbf{H}(z)$ is equivalent to the filter bank implementing a tight frame decomposition, which offers useful properties such as a fixed energy relation between the fullband signal $X(z)$ and the subband samples in $\underline{Y}(z)$. In a more general case, $H_k(z)$ and $G_k(z)$ can be based on different prototype lowpass filters (with different coefficients or even filter lengths) and therefore the filter bank system is not necessarily perfectly reconstructing. In this case the factorisation of the polyphase analysis matrix remains as in Section 2.2, while the factorial matrices of $\mathbf{G}(z)$ in (2.18) are built accordingly, whereby the parameters of the synthesis prototype filter have to be applied for $\mathbf{L}_1(z)$, $\mathbf{P}$, and $\mathbf{L}_2$.

3 Canonical State-Space Representations

Before trying to find suitable implementations for the previously factorised filter bank operations, canonical state-space representations for Figs. 2 and 4 are derived in this section. These representations take the form

$$\begin{bmatrix} z\underline{W}(z) \\ \underline{V}(z) \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix} \cdot \begin{bmatrix} \underline{W}(z) \\ \underline{U}(z) \end{bmatrix} \quad , \tag{3.1}$$

A flow graph of (3.1) is given in Fig. 5. Appropriate system matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$, a state vektor $\underline{W}(z)$, input $\underline{U}(z)$ and output $\underline{V}(z)$ need to be defined in the following.

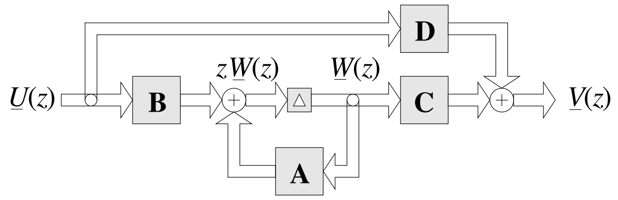

FIG. 5. Flow graph of state-space representation with states $\underline{W}(z)$.

3.1 Analysis Filter Bank Representation

For the analysis filter bank, $\underline{U}(z)$ contains N demultiplexed samples of the input signal (i.e. $\underline{U}(z) = \underline{X}(z)$, the output $\underline{V}(z) = \underline{Y}(z)$ holds the K subband signals. For a canonical form, the state vector $\underline{W}(z)$ must not have more than $L_\mathrm{p}-N$ elements and can be found by reformulating the analysis operation by introduction of an intermediate variable $\underline{Q}(z)$:

$$\underline{Y}(z) = \mathbf{T}\cdot\mathbf{L}_2\cdot\mathbf{P}\cdot\underline{Q}(z) \qquad \text{with} \quad \underline{Q}(z) = \mathbf{L}_1(z)\cdot\underline{X}(z) \tag{3.2}$$

$$\underline{Q}(z) = \begin{bmatrix} \mathbf{0}_{N\times L_\mathrm{p}-N} & \mathbf{0}_{N\times N} \\ \mathbf{I}_{L_\mathrm{p}-N} & \mathbf{0}_{L_\mathrm{p}-N\times N} \end{bmatrix} z^{-1}\cdot\underline{Q}(z) + \begin{bmatrix} \mathbf{I}_N \\ \mathbf{0}_{L_\mathrm{p}-N\times N} \end{bmatrix} \underline{X}(z)\,. \tag{3.3}$$

With the recursive update in (3.3), the lower portion with $L_\mathrm{p}-N$ elements of $\underline{Q}(z)$ now forms the state vector $\underline{W}(z)$, and the state space system matrices can be identified as

$$\mathbf{A} = \begin{bmatrix} \mathbf{0}_{N\times L_\mathrm{p}-2N} & \mathbf{0}_{N\times N} \\ \mathbf{I}_{L_\mathrm{p}-2N} & \mathbf{0}_{L_\mathrm{p}-2N\times N} \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} \mathbf{I}_N \\ \mathbf{0}_{L_\mathrm{p}-2N\times N} \end{bmatrix} \tag{3.4}$$

$$\mathbf{C} = \mathbf{T}\cdot\mathbf{L}_2\cdot\mathbf{P}\cdot\begin{bmatrix} \mathbf{0}_{N\times L_\mathrm{p}-N} \\ \mathbf{I}_{L_\mathrm{p}-N} \end{bmatrix} \qquad \mathbf{D} = \mathbf{T}\cdot\mathbf{L}_2\cdot\mathbf{P}\cdot\begin{bmatrix} \mathbf{I}_N \\ \mathbf{0}_{L_\mathrm{p}-N\times N} \end{bmatrix}\,. \tag{3.5}$$

Note that all memory exhibiting matrices in (2.14) have been replaced by memoryless operations due to the recursive updating in (3.1).

3.2 Synthesis Filter Bank Representation

For the synthesis, the input in Fig. 5 now contains the K subband samples, $\underline{U}(z) = \underline{Y}(z)$, which are used for the reconstruction of N polyphase components of the fullband signal $\hat{X}(z)$, held in the output $\underline{V}(z) = \underline{\hat{X}}(z)$. A suitable state vector $\underline{W}(z)$ is sought from (2.18) with (2.19) inserted:

$$\underline{\hat{X}}(z) = \hat{\mathbf{L}}_1^\mathrm{T}(z)\cdot\mathbf{P}\cdot\mathbf{L}_2^\mathrm{T}\cdot\mathbf{T}^\mathrm{H}\cdot\underline{Y}(z) = \begin{bmatrix} \mathbf{0}_{N\times L_\mathrm{p}-N} & \mathbf{I}_N \end{bmatrix}\cdot\underline{Q}(z) \tag{3.6}$$

$$\underline{Q}(z) = \begin{bmatrix} \mathbf{0}_{N\times L_\mathrm{p}-N} & \mathbf{0}_{N\times N} \\ \mathbf{I}_{L_\mathrm{p}-N} & \mathbf{0}_{L_\mathrm{p}-N\times N} \end{bmatrix} z^{-1}\cdot\underline{Q}(z) + \mathbf{P}\cdot\mathbf{L}_2^\mathrm{T}\cdot\mathbf{T}^\mathrm{H}\cdot\underline{Y}(z) \tag{3.7}$$

The upper $L_\mathrm{p}-N$ elements of the intermediate variable $\underline{Q}(z)$ form the state vector $\underline{W}(z)$, which together with the recursive formulation (3.7) gives rise to the following state-space matrices:

$$\mathbf{A} = \begin{bmatrix} \mathbf{0}_{N\times L_\mathrm{p}-2N} & \mathbf{0}_{N\times N} \\ \mathbf{I}_{L_\mathrm{p}-2N} & \mathbf{0}_{L_\mathrm{p}-2N\times N} \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} \mathbf{I}_{L_\mathrm{p}-N} & \mathbf{0}_{L_\mathrm{p}-N\times N} \end{bmatrix} \cdot \mathbf{P} \cdot \mathbf{L}_2^\mathrm{T} \cdot \mathbf{T}^\mathrm{H} \quad (3.8)$$

$$\mathbf{C} = \begin{bmatrix} \mathbf{0}_{N\times L_\mathrm{p}-2N} & \mathbf{I}_N \end{bmatrix} \qquad \mathbf{D} = \begin{bmatrix} \mathbf{0}_{N\times L_\mathrm{p}-N} & \mathbf{I}_N \end{bmatrix} \cdot \mathbf{P} \cdot \mathbf{L}_2^\mathrm{T} \cdot \mathbf{T}^\mathrm{H} \quad (3.9)$$

The system matrices $\mathbf{A}$ in both (3.4) and (3.8) represent tapped delay lines (TDL), in which the state values are shifted by N states for every update iteration.

4 Filter Bank Implementation

Based on the factorisations in Section 2 and the state-space representations in Section 3, we now aim to find signal flow graphs that provide simple and efficient OSFB implementations.

4.1 Analysis Filter Bank Implementation

Inspecting (3.4) and (3.5), the analysis filter bank operation in (2.14) can be executed in two steps. As mentioned above, the state values from a TDL, which is shifted by $\mathbf{A}$ and updated with N fresh samples in every subband sampling period by $\mathbf{B}$. Hence, $\mathbf{C}$ and $\mathbf{D}$ are excited by $L_\mathrm{p}-N$ old and N current input samples in $\underline{W}(z)$ and $\underline{U}(z)$, respectively. Due the similiarity of $\mathbf{C}$ and $\mathbf{D}$, a single TDL $[\underline{U}^\mathrm{T}(z)\ \ \underline{W}^\mathrm{T}(z)]^\mathrm{T}$ holding a total of L_p current and past input samples can be assembled. The result is the flow graph in Fig. 6. There, the demultiplexing of the scalar OSFB input into N parallel samples in $\underline{U}(z)$ as shown in Fig. 2 is already incorporated into the TDL.

According to (3.5), in Fig. 6 the TDL vector $[\underline{U}^\mathrm{T}(z)\ \ \underline{W}^\mathrm{T}(z)]^\mathrm{T}$ is passed into the block $\mathbf{P}$, multiplying each value by a prototype filter coefficients $p[i]$. From the results, $\mathbf{L}_2$ creates K subsummations. After rotated of these subsums by the modulation matrix $\mathbf{T}$, finally a set of K subband samples has been calculated.

Note, that the only memory-exhibiting operation in this analysis filter bank realisation is the TDL holding L_p samples of the input signal. The number of states has increased by N over the canonical state-space representation in (3.4) and (3.5) due to the inclusion of the demultipled of Fig. 2. Therefore with respect to the overall circuit running at the fullband rate, the circuit in Fig. 6 is canonical.

4.2 Synthesis Filter Bank Implementation

Let us consider the system matrices $\mathbf{B}$ and $\mathbf{D}$ defined for the synthesis OSFB in (3.8) and (3.9), respectively, and the state-space representation in Fig. 5. It is obvious that the subband samples in $\underline{U}(z) = \underline{Y}(z)$ are derotated by $\mathbf{T}^\mathrm{H}$ and duplicated to L_p values by $\mathbf{L}_2^\mathrm{T}$ to finally excite the L_p prototype filter coefficients in $\mathbf{P}$. Of these L_p products, the upper $L_\mathrm{p}-N$ values are latched by $\mathbf{B}$ onto the TDL implemented by $\mathbf{A}$ in (3.8). The lower N product values are added with the lower N elements of the state vector to form the output $\underline{V}(z) = \underline{\hat{X}}(z)$. Incorporating the multiplexing of the output $\underline{\hat{X}}(z)$ to $\hat{X}(z)$, the TDL structure

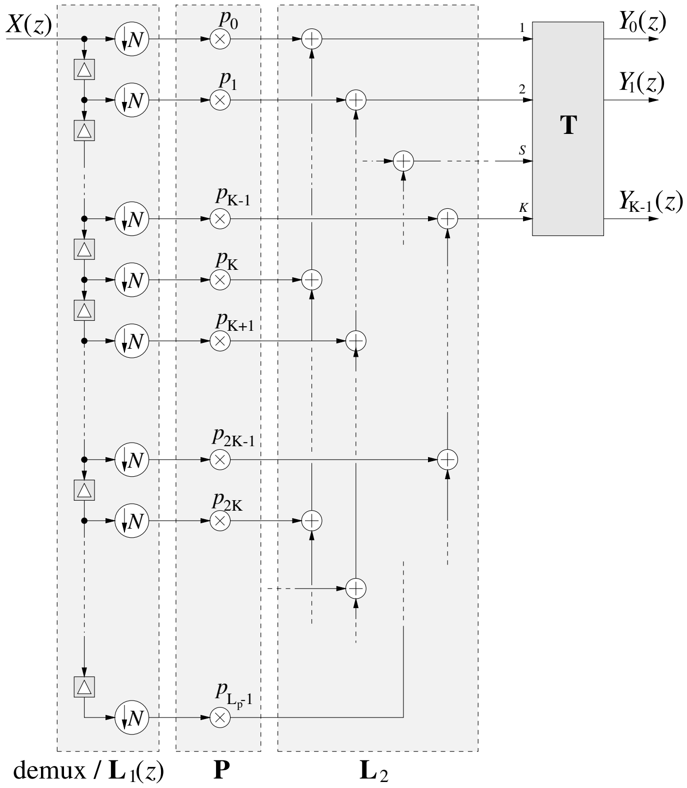

FIG. 6. Analysis filter bank signal flow graph.

can be realised as shown in Fig. 7. This can also be motivated by merging the multiplexer of Fig. 4 with $\hat{\mathbf{L}}_1^{\mathrm{T}}(z)$.

Note that similar to the analysis OSFB in Fig. 6 this circuit only requires a single TDL, onto which the results of the operation $\mathbf{P}\cdot\mathbf{L}_2^{\mathrm{T}}\cdot\mathbf{T}^{\mathrm{H}}\cdot\underline{\hat{Y}}(z)$ are accumulated. This operation is performed once in every subband sampling period. Afterwards, the values in the TDL can be shifted N times, clocked at the fullband rate, to form the fullband output samples. Similar arguments as in Section 4.1 hold for the canonical property of the flow graph in Fig. 7.

4.3 Special Filter Bank Cases

A number of special filter bank implementations arise from different choices of the transform matrix $\mathbf{T}$, which in the following is to be refined in its implementation. In case of a DFT modulated filter bank, resulting in an even stacked design as shown in Fig. 3(a), $\mathbf{T}$ is a $K\times K$ DFT matrix, $\mathbf{T}=\mathbf{T}_{\mathrm{DFT}}$. In case of a generalised DFT (GDFT, [14]) filter bank, $\mathbf{T}$ is a GDFT matrix. This GDFT matrix comprises of elements $t_{k,n}=e^{j\frac{2\pi}{K}(k+k_0)(n+n_0)}$, with $n=0\dots L_{\mathrm{p}}-1$ and $k=0\dots L_{\mathrm{p}}-1$. Note that for $k_0=\frac{1}{2}$, the odd stacked filter bank characteristic

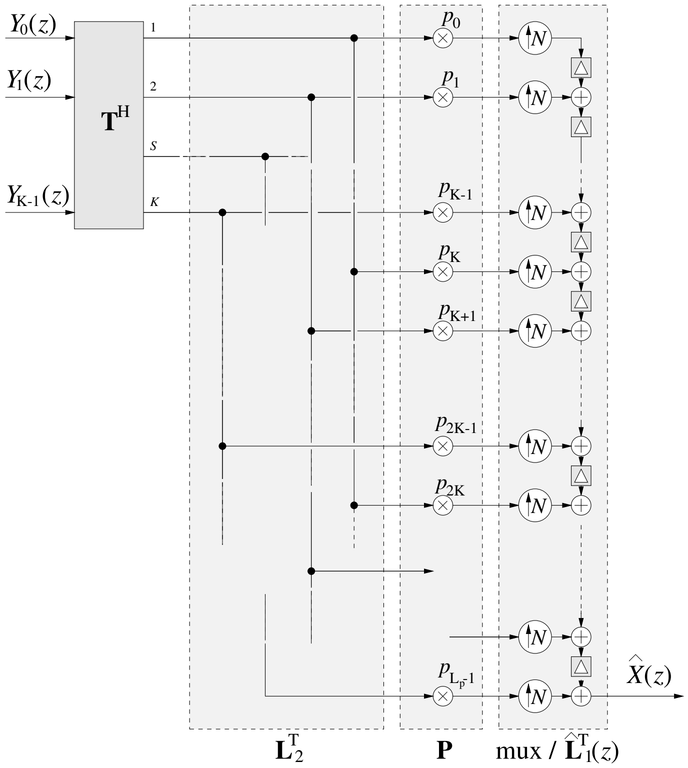

FIG. 7. Synthesis filter bank signal flow graph.

of Fig. 3(b) arises. A GDFT matrix generally permits a factorisation

$$\mathbf{T} = \mathbf{D}_1 \cdot \mathbf{T}_{\mathrm{DFT}} \cdot \mathbf{D}_2 \qquad . \tag{4.1}$$

For an odd stacked filter bank, all matrices are $K \times K$ and $\mathbf{D}_1$ and $\mathbf{D}_2$ have diagonal form if inversion of the signs of prototype coefficients are performed as discussed in Section 2.2 [12].

Efficient implementations make use of an FFT routine instead of performing $\mathbf{T}_{\mathrm{DFT}}$ as a matrix operation. Further, if the input signal $X(z)$ is real valued, (i) all operations except the transform matrix can be performed with real arithmetic and (ii) redundancies arise due to approximately half of the subband signals being complex conjugate copies of others. Hence about half (depending on the DFT/GDFT transform and K being odd or even) of the subbands do not need to be calculated nor processed.

Single sideband (SSB) modulated OSFBs for real valued subband signals can be obtained by modification of a GDFT modulated filter bank decimated by only $N/2$ [8]. An additional complex modulation is performed on the subband signals $\underline{Y}(z)$ followed by a real operation. On the synthesis side, this is compensated by a matching demodulation prior to feeding into a GDFT synthesis filter bank.

Table 1 Computational Complexity of Filter Banks Implementations

	C_{real} / [MACs]	C_{complex} / [MACs]
DFT	$\frac{1}{N}(L_{\text{p}} + 4K \log_2 K)$	$\frac{1}{N}(2L_{\text{p}} + 4K \log_2 K)$
GDFT	$\frac{1}{N}(L_{\text{p}} + 4K \log_2 K + 4K)$	$\frac{1}{N}(2L_{\text{p}} + 4K \log_2 K + 8K)$
SSB	$\frac{2}{N}(L_{\text{p}} + 4K \log_2 K + 5K)$	$\frac{2}{N}(2L_{\text{p}} + 4K \log_2 K + 10K)$

4.4 Computational Complexity

From the signal flow graphs for analysis and synthesis in Figs. 6 and 7, the computational complexities for both operations can be evaluated in terms of multiply-accumulates (MACs) per sampling period. The latter is the period of the fullband signals prior analysis or after synthesis. The complexities are given in Tab. 1 and are identical for analysis and synthesis, but differ for the choice of T and depend on whether the input signal is real or complex valued. Note, that the multiplication of the complex samples with the real valued prototype filter coefficients accrues to $2L_p$ MACs. The modulation matrix $\mathbf{T}_{\text{DFT}}$ is assumed to be implemented by a K-point FFT requiring $4K \log_2 K$ real valued MACs, which is invariably applied for all types of filter banks.

Although a number of methods reported in the literature give identical complexities in terms of MACs, the realisations in Figs. 6 and 7 do not require any additional time-varying circular shifts [9] or switching [5], the indexing of time-varying filters [10], or filters with sparse coefficients [11; 12]. Further, the signal flow graphs in Figs. 6 and 7 only require a single circular buffer and hence a minimum amount of pointers for addressing, and permit an arbitrary prototype filter length L_{p} independent of both the decimation ratio N and the channel number K.

5 Conclusion

Oversampled filter banks where all filters are derived from a prototype filter by modulation have been analysed using the well-known polyphase decomposition. Similar to previous analyses in the literature, this decomposition was factorised to reduce all filter operations to operations on the prototype filter coefficients, and a multiplication by the modulation matrix. This ensured a minimum amount of multiply-accumulate operations. However, the factorisation was exploited via a state-space representation to locate all memory-requiring operations next to the multiplexers and demultiplexers of the circuit. The benefit is an implementation with a only single TDL that can be conveniently updated.

The presented implementation can be applied to a variety of modulated filter banks, such as shown for DFT, GDFT, or SSB filter banks, and with a large flexibility for the length of the prototype filter. Although not explicitly derived here, the analysis and synthesis filter bank implementations can be similarly applied to filter banks where filters $H_k(z)$ and $G_k(z)$ originate from more than one prototype filter [15].

Bibliography

1. W. Kellermann. Analysis and Design of Multirate Systems for Cancellation of Acoustical Echoes. In *Proc. IEEE ICASSP*, volume 5, pages 2570–2573, New York, 1988.
2. C. Breining, P. Dreiseitel, E. Hänsler, A. Mader, B. Nitsch, H. Puder, T. Schertler, G. Schmidt, and J. Tilp. Acoustic Echo Control. *IEEE Signal Processing Magazine*, 16(4):42–69, July 1999.
3. V.S. Somayazulu, S.K. Mitra, and J.J. Shynk. Adaptive Line Enhancement Using Multirate Techniques. In *Proc. IEEE ICASSP*, volume 2, pages 928–931, Glasgow, Scotland, UK, May 1989.
4. W. Kellermann. "Strategies for Combining Acoustic Echo Cancellation and Adaptive Beamforming Microphone Arrays". In *Proc. IEEE ICASSP*, volume I, pages 219–222, Munich, Germany, April 1997.
5. G. Cherubini, E. Eleftheriou, S. Ölcer, and J. M. Cioffi. Filter Bank Modulation Techniques for Very High-Speed Digital Subscriber Lines. *IEEE Communications Magazine*, 38(5):98–104, May 2000.
6. M. Bellanger, G. Bonnerot, and M. Coudreuse. Digital Filtering by Polyphase Network: Application to Sample Rate Alteration and Filter Banks. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-24(No.4):109–114, April 1978.
7. P. P. Vaidyanathan. *Multirate Systems and Filter Banks*. Prentice Hall, Englewood Cliffs, 1993.
8. G. Wackersreuther. Some New Aspects of Filters for Filter Banks. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-34(10):1182–1200, October 1986.
9. G. Wackersreuther. *Ein Beitrag zum Entwurf digitaler Filterbänke*. PhD thesis, Institut für Nachrichtentechnik, Universität Erlangen-Nürnberg, 1987.
10. R. Stasinski. Efficient Implementation of Uniform Filter Banks in the Absence of Critical Sampling. *IEE Electr. Let.*, 30(2):118–120, January 1994.
11. Z. Cvetković and M. Vetterli. Tight Weyl-Heisenberg Frames in $l^2(\mathbb{Z})$. *IEEE Transactions on Signal Processing*, 46(5):1256–1259, May 1998.
12. S. Weiss. *On Adaptive Filtering in Oversampled Subbands*. PhD thesis, Signal Processing Division, University of Strathclyde, Glasgow, May 1998.
13. S. Weiss and R. W. Stewart. Fast Implementation of Oversampled Modulated Filter Banks. *IEE Electr. Let.*, 36(17):1502–1503, August 2000.
14. R. E. Crochiere and L. R. Rabiner. *Multirate Digital Signal Processing*. Prentice Hall, Englewood Cliffs, NJ, 1983.
15. M. Harteneck, S. Weiss, and R. W. Stewart. Design of Near Perfect Reconstruction Oversampled Filter Banks for Subband Adaptive Filters. *IEEE Transactions on Circuits & Systems II*, 46(8):1081–1086, August 1999.