PERFECT-RECONSTRUCTION BIORTHOGONAL COSINE-MODULATED FILTER BANKS WITH FIXED-POINT ARITHMETIC

Tanja Karp[1] *and Alfred Mertins*[2]
[1] Texas Tech University, Department of Electrical and Computer Engineering
Box 43102, Lubbock, TX 79409-3102, USA, tanja.karp@ttu.edu
[2] University of Wollongong, School of Electrical, Computer, and Telecommunication
Engineering, Wollongong, NSW 2522, Australia, mertins@uow.edu.au

ABSTRACT

This paper describes the implementation of biorthogonal cosine-modulated filter banks on fixed-point arithmetic digital signal processors. The proposed implementation has the property that the overall filter bank keeps the perfect reconstruction property despite coefficient quantization, overflow, and rounding of intermediate results. The realization of the prototype filter is based on a factorization into zero-delay and maximum-delay matrices. We demonstrate how the frequency selectivity of the filter bank and the coding gain changes with the available wordlength of the fixed-point implementation and the dynamic range of the input signal. For speech signals it turns out that overflow and rounding errors hardly affect the frequency selectivity of the filters if the input signal uses only 75% of the available dynamic range.

1 INTRODUCTION

Biorthogonal cosine-modulated filter banks with perfect reconstruction (PR) have been studied in [1, 2, 3]. Other than paraunitary filter banks they allow to design the system delay independently of the filter length, thus, resulting in a better stopband attenuation and a smaller transition bandwidth for a given system delay than paraunitary filter banks.

However, when implementing such a filter bank on a processor with finite-precision arithmetic, the prototype and the modulating sequences need to be quantized, which results in a loss of the PR property. Rounding intermediate results to the available wordlength causes further reconstruction errors.

In this paper, we concentrate on the implementation of the lowpass prototype filter and derive a formulation that allows PR even with finite-precision arithmetic. Note that the proposed approach is different from designs which explicitly use integer prototypes such as the ones in [4, 5, 6, 7, 8] since there the number of bits needed for the implementation increases through the filter bank wherease in our implementation the number of bits needed remains constant. For the realization of the modulation matrix we refer to [6, 7, 9, 10].

The paper is organized as follows: First, we will recall basic properties of cosine-modulated filter banks and show how they can be realized using zero-delay and maximum-delay matrice. We then show that coefficient quantization and rounding operations after each multiplication and addition does not affect the PR property. Finally, we demonstrate how these nonlinear operations affect the amplitude spectrum of the subband signals and the coding gain.

2 COSINE-MODULATED FILTER BANKS

In this paper we limit our investigations to critically sampled M channel cosine-modulated filter banks with the same FIR prototype $p(n)$ for the analysis and synthesis filters and an overall delay of $D = 2sM + 2M - 1$ where s is an integer. Note that this is the most commonly treated case where the least PR constraints are imposed on the prototype filter. The analysis and synthesis filters are denoted as $h_k(n)$ and $f_k(n)$, respectively, and are obtained from the prototype filter through the following cosine modulation [2]:

$$h_k(n) = \sqrt{\frac{2}{M}} p(n)\ \cos(\frac{\pi}{M}(k+0.5)(n-D/2)+\theta_k), \tag{1}$$

$$f_k(n) = \sqrt{\frac{2}{M}} p(n)\ \cos(\frac{\pi}{M}(k+0.5)(n-D/2)-\theta_k) \tag{2}$$

with $\theta_k = (-1)^k\pi/4$. In [11, 12] we derived a computational efficient realization of the analysis and synthesis filter bank based on an $M \times M$ cosine transform and specific lifting steps called zero-delay and maximum-delay matrices for the polyphase filters. The resulting structure is shown in Figure 1. The transform matrices $\mathbf{C}_1'$ and $\mathbf{C}_2'$ in Figure 1 are defined as:
- for $\ell = 0, \ldots, M/2-1$:

$$[\mathbf{C}_1']_{k,\ell} = \sqrt{\frac{2}{M}} \cos((k+0.5)\frac{\pi}{M}(\ell - \frac{D}{2}) + \theta_k) \tag{3}$$

$$[\mathbf{C}_2']_{k,\ell} = \sqrt{\frac{2}{M}} \cos((k+0.5)\frac{\pi}{M}(2M-1-\ell-\frac{D}{2}) - \theta_k) \tag{4}$$

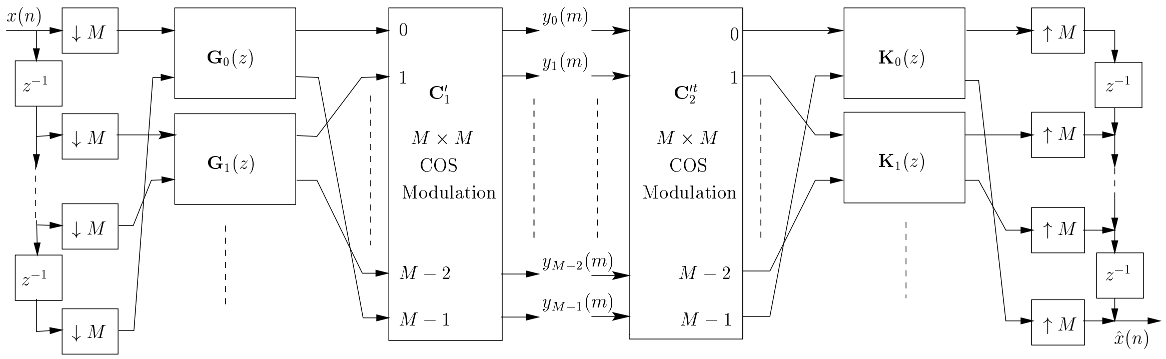

Figure 1: Efficient polyphase realization

- for $\ell = M/2, \ldots, M-1$:

$$[\mathbf{C}'_1]_{k,\ell} = \sqrt{\frac{2}{M}} \cos((k+0.5)\frac{\pi}{M}(\ell + M - \frac{D}{2}) + \theta_k \quad (5)$$

$$[\mathbf{C}'_2]_{k,\ell} = \sqrt{\frac{2}{M}} \cos((k+0.5)\frac{\pi}{M}(M - 1 - \ell - \frac{D}{2}) - \theta_k) \quad (6)$$

The polyphase filters are realized using matrices $\mathbf{G}_\ell(z)$ and $\mathbf{K}_\ell(z)$, $\ell = 0, \ldots, M/2-1$, each of which implements four polyphase filters. For a PR filter bank they can be factorized in the following form:

$$\mathbf{G}_\ell(z) = \prod_{j=1}^{j_0} \mathbf{D}_{\ell,j}(z) \prod_{i=1}^{i_0} \mathbf{B}_{\ell,i}(z) \cdot \mathbf{G}_{\ell,ini}(z), \quad (7)$$

$$\mathbf{K}_\ell(z) = \mathbf{K}_{\ell,ini}(z) \prod_{i=i_0}^{1} \mathbf{B}^{-1}_{\ell,i}(z) \prod_{j=j_0}^{1} (z^{-2}\mathbf{D}^{-1}_{\ell,j}(z)) \quad (8)$$

where $j_0 = 2s$ for a fixed value of s and and i_0 is chosen such that the desired filter length N is met. The matrices $\mathbf{B}_{\ell,i}(z)$, $\mathbf{D}_{\ell,j}(z)$, and $\mathbf{G}_{\ell,ini}(z)$ are called zero-delay, maximum delay, and initialization matrices, respectively, and have the following structures [12] :

$$\mathbf{B}_{\ell,i}(z) = \begin{bmatrix} 0 & 1 \\ 1 & b_{\ell,i}\, z^{-1} \end{bmatrix}, \; \mathbf{B}^{-1}_{\ell,i}(z) = \begin{bmatrix} -b_{\ell,i}\, z^{-1} & 1 \\ 1 & 0 \end{bmatrix} \quad (9)$$

$$\mathbf{D}_{\ell,j}(z) = \begin{bmatrix} d_{\ell,j} & z^{-1} \\ z^{-1} & 0 \end{bmatrix}, \; z^{-2}\, \mathbf{D}^{-1}_{\ell,j}(z) = \begin{bmatrix} 0 & z^{-1} \\ z^{-1} & -d_{\ell,j} \end{bmatrix} \quad (10)$$

$$\mathbf{G}_{\ell,ini}(z) = (-1)^s \begin{bmatrix} 1 & 0 \\ \tilde{g}_{\ell,0}\, z^{-1} & z^{-1} \end{bmatrix} \begin{bmatrix} 1 & \tilde{g}_{\ell,1} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ \tilde{g}_{\ell,2} & 1 \end{bmatrix} \quad (11)$$

$$\mathbf{K}_{\ell,ini}(z) = \begin{bmatrix} 1 & 0 \\ -\tilde{g}_{\ell,2} & 1 \end{bmatrix} \begin{bmatrix} 1 & -\tilde{g}_{\ell,1} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} z^{-1} & 0 \\ -\tilde{g}_{\ell,0}\, z^{-1} & 1 \end{bmatrix} \quad (12)$$

Zero-delay matrices increase the length of the polyphase filters but keep the system delay unchanged whereas maximum-delay matrices increase both, the filter length and the system delay.

3 FIXED-POINT IMPLEMENTATION

An implementation of the filter bank on a fixed-point processor has been simulated in Matlab using Simulink Fixed Point Blockset which automatically takes care of rounding, scaling, and overflow. It is possible to assign different wordlengths to signals and gains. Figures 2 and 3 show the realization of a zero-delay matrix and its inverse and Figures 4 and 5 show the realization of an initialization matrix and its inverse.

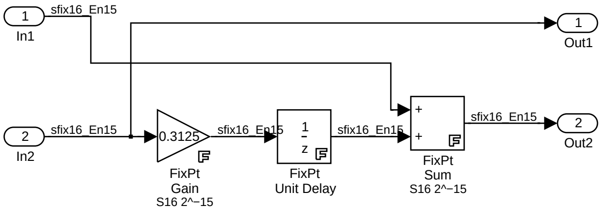

Figure 2: Zero-delay matrix using Matlab Simulink Fixed-Point Blockset. Wordlength: 16 bit for both signals and coefficient.

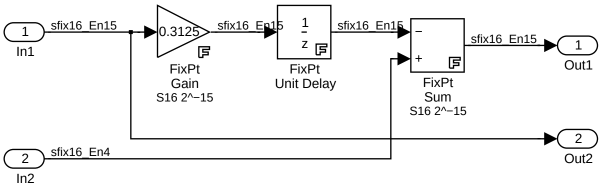

Figure 3: Inverse zero-delay matrix using Matlab Simulink Fixed-Point Blockset. Wordlength: 16 bit for both signals and coefficient.

For the fixed point implementation, the entries of the matrices (9)-(12) must be quantized to a specified wordlength. Fortunately, each matrix contains only one coefficient that is subject to quantization. Because the inverse matrix contains the same coefficient, coefficient quantization does not alter the PR property of the filter bank as long as the same quantization scheme (e.g. floor, ceiling, nearest) is applied in the analysis and synthesis filter banks.

The same holds true when rounding the multiplica-

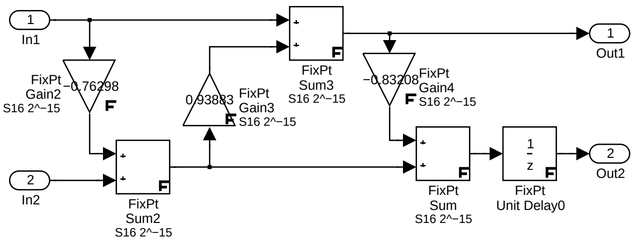

Figure 4: Initialization matrix using Matlab Simulink Fixed-Point Blockset. Wordlength: 16 bit for both signals and coefficients.

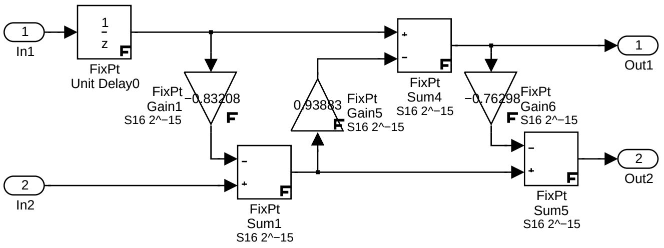

Figure 5: Inverse initialization matrix using Matlab Simulink Fixed-Point Blockset. Wordlength: 16 bit for both signals and coefficients.

tion results, to the original wordlength of the input data stream since exactly the same rounded products occur on the analysis and synthesis sides and rounding errors compensate. Note that the same argumentation has been followed in [13] for the design of integer-to-integer wavelet transforms.

During addition, a carry over bit may occur that usually increases the wordlength of the sum by one bit. To avoid increased wordlengths, we carry out finite-field additions in our implementation. That is, the sum of two numbers A and B in the range $[-1, 1]$ is computed as

$$C = ((A + B + 1) \mod 2) - 1. \tag{13}$$

Most importantly, with this type of addition, overflow does not affect the PR property.

4 DESIGN EXAMPLES

For a filter bank with $M = 8$ subbands, we implement a lowdelay prototype filter of length $N = 32$ that causes an overall system delay of $D = 15$. For this setting, $\mathbf{G}_\ell(z)$ according to (7) consists of the initialization matrix and two zero-delay matrices. The prototype coefficients as well as the initialization and zero-delay matrix coefficients for the original floating point prototype filter are given in Table 1. The magnitude responses of the floating-point analysis filters are shown in Figure 6.

We only consider the influence of the polyphase realization of the prototype and assume that the modulation matrix is perfect.

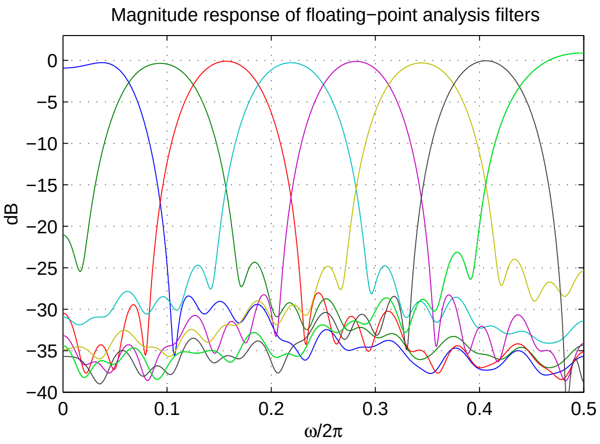

Figure 6: Magnitude responses of the analysis filters.

Table 1: Coefficients of the floating-point prototype filter and the initialization and zero-delay matrix coefficients

p_0	0.28368881433396	p_{16}	0.23802829296291
p_1	0.38739487911472	p_{17}	0.14898493689125
p_2	0.49853892787010	p_{18}	0.07230700728739
p_3	0.61003862299352	p_{19}	0.01265254491271
p_4	0.71482988586654	p_{20}	−0.02745312720718
p_5	0.80617651327097	p_{21}	−0.05212221331985
p_6	0.88110239746544	p_{22}	−0.05705821166075
p_7	0.93883034147570	p_{23}	−0.05213520966147
p_8	0.95232579397068	p_{24}	−0.03919070847929
p_9	0.92151456192117	p_{25}	−0.01917604182496
p_{10}	0.87044546070271	p_{26}	0.00180987945862
p_{11}	0.79737935441808	p_{27}	0.00636354385339
p_{12}	0.70488882323366	p_{28}	0.01380743400644
p_{13}	0.59828289592636	p_{29}	0.00130464427674
p_{14}	0.48542538977717	p_{30}	−0.00734403541791
p_{15}	0.37339381850171	p_{31}	−0.00858391991101
$\tilde{g}_{01}$	−0.83208008854513	$\tilde{g}_{21}$	−0.47326870567427
$\tilde{g}_{02}$	0.93883034147570	$\tilde{g}_{22}$	0.80617651327097
$\tilde{g}_{03}$	−0.76298256886340	$\tilde{g}_{23}$	−0.62202391644390
b_{01}	0.16464726941260	b_{21}	−0.02503048496292
b_{02}	0.23825835937716	b_{22}	0.08427716336670
$\tilde{g}_{11}$	−0.71272354814690	$\tilde{g}_{31}$	0.09010463437248
$\tilde{g}_{12}$	0.88110239746544	$\tilde{g}_{32}$	0.71482988586654
$\tilde{g}_{13}$	−0.69527119963296	$\tilde{g}_{33}$	−0.54553032087313
b_{11}	0.12871127930841	b_{31}	−0.50294576287202
b_{12}	0.15337504408994	b_{32}	0.02579188515420

Although nonlinear operations such as quantization, rounding and overflow do not affect the PR property of the filter bank they can have a significant influence on the frequency selectivity of the filter bank. In the following we present simiulation results for a speech input signal with the spectrum shown in Figure 7.

PSD of input signal

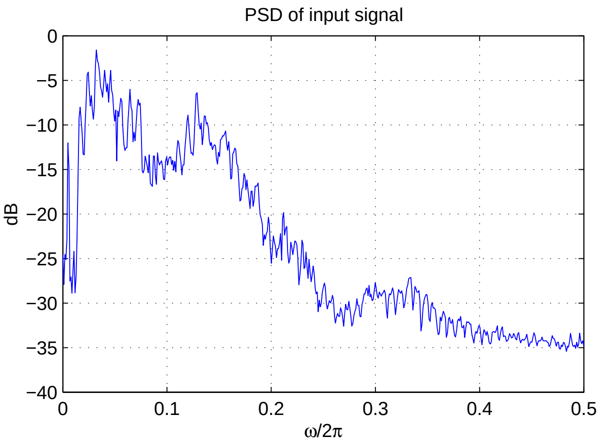

Figure 7: Amplitude spectrum of the input signal.

4.1 The Effect of Overflow

In a first step we investigate the influence of overflow occuring in the adders of the zero-delay and initialization matrices in Figures 2 and 4. Note that no overflow occurs during the multiplications since the absolute value of all coefficients in Table 1 is smaller than one. All signals and coefficients have a wordlength of 16 bits to represent numbers in the range $[-1\ 1]$. The input sequence of length 45578 is scaled such that it covers the ranges of $[-0.5\ 0.5]$, $[-0.75\ 0.75]$, and $[-1\ 1]$, respectively. The amount of overflow occuring for the different realizations is given in Table 2.

Table 2: Amount of overflow occuring in the polyphase realization (input signal is of length 45578).

input range	no. of overflows
$[-0.5\ 0.5]$	none
$[-0.75\ 0.75]$	1077
$[-1\ 1]$	4761

The magnitude responses of the analysis filters in dB are estimated using the power spectrum density of the subband signals by taking the subband power spectrum density (PSD) in dB prior to subsampling minusd the input PSD in dB. All PSD estimates are obtained by averaging the squared magnitudes of DFT's of non-overlapping rectangular signal windows of length 1024. The results are shown in Figure 8 for dynamic ranges of the input signal of $[-0.5\ 0.5]$ and $[-0.75\ 0.75]$, respectively, as well as for the normalized spectrum of the lowpass band for all three different input ranges together with the normalized input spectrum.

Effective analysis filter response for input range: [−0.5 0.5]

Effective analysis filter response for input range: [−0.75 0.75]

Lowpass subband PSD for different input ranges

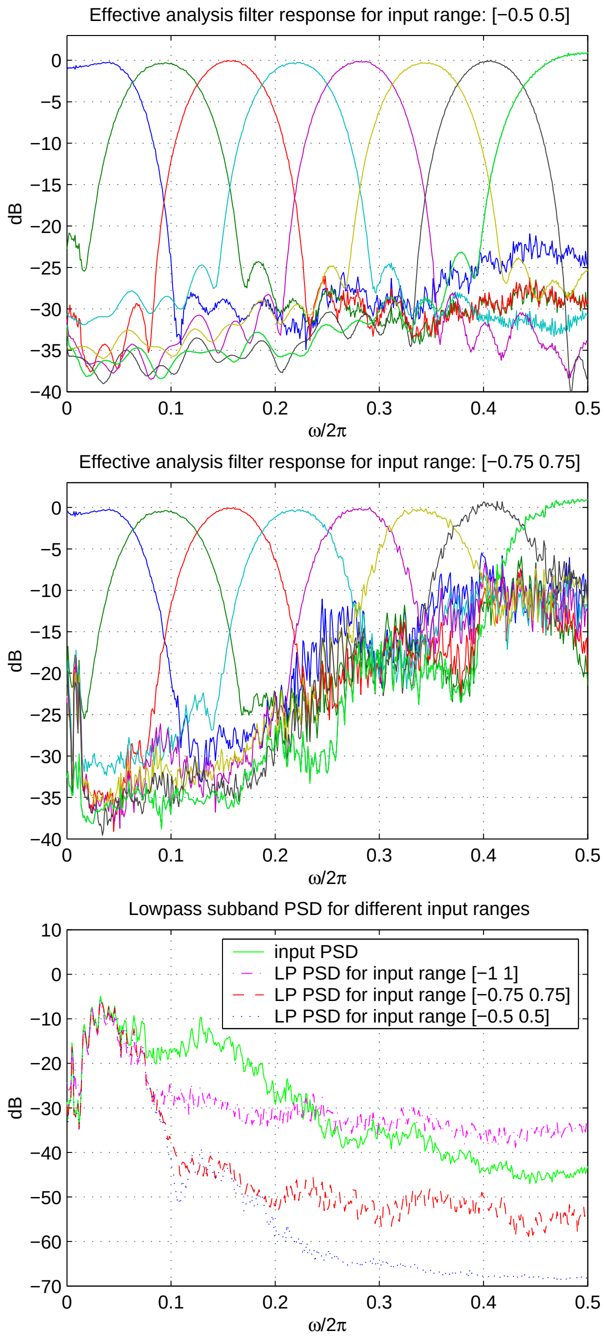

Figure 8: Influence of overflow on the transfer characteristic of the filter bank and the lowpass spectrum.

It can be seen that the realization is rather sensitive to the occurrence of overflow. High frequency components that pass through the lowpass filter increase with the range of the input signal and thus the amount of overflow. For the input range of $[-1\ 1]$ there are more high frequency components in PSD of the lowpass band than in the original input PSD. This is due to the finite field addition in our realization which creates new frequency components in the spectrum. On the other hand, a security margin of one bit for the input signal is sufficient to prevent overflow, see Table 2. Fortunately, most audio/speech signals satisfy this margin most of the time.

4.2 Coefficient Quantization

In a next step we investigate the sensitivity of the structure to coefficient quantization. Realizing the filter banks with coefficients of a small wordlength has advantages for VLSI implementations where multiplications are realized using shift and add operations.

The filter coefficients are quantized to 4, 8, and 16 bit wordlength. We can see from Figure 9 that the filter coefficients are not very sensitive to quantization. The result for 8 and 16 bit superpose. Only when quantizing the coefficients to 4 bit wordlength an increase of the power in the stopband can be observed for $0.1 \leq \omega/2\pi \leq 0.2$.

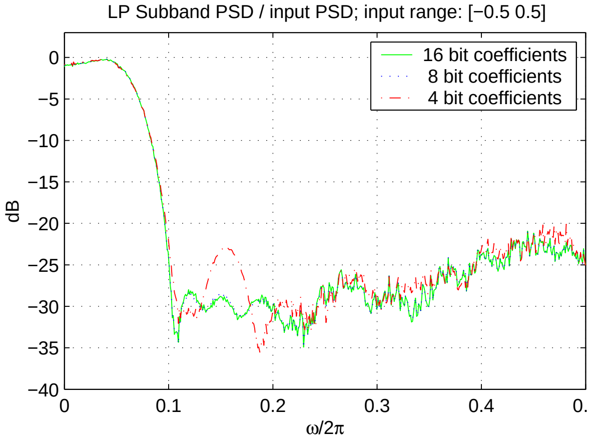

Figure 9: Influence of coefficient quantization on the transfer characteristic of the filter bank and the lowpass spectrum.

Figure 10 shows the magnitude frequency response of the lowpass filter if only coefficient quantization and no rounding of intermediate signals is applied. Comparing Figures 9 and 10, we can draw the conclusion that the decrease of the stopband attenuation with respect to high frequencies is mainly due to the rounding operation that keeps the wordlength constant. As long as rounding is performed, coefficients can be implemented using a small wordlength without significant additional loss of performance.

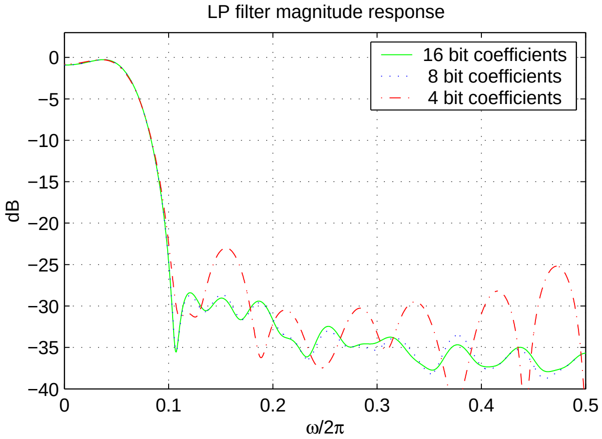

Figure 10: Lowpass magnitude response for different coefficient wordlength.

4.3 Coding Gain

The coding gain of the filter bank implementations is given by [14]

$$G = \frac{\sigma_x^2}{\left(\prod_{k=0}^{M-1} \sigma_{y_k}^2 ||f_k||^2\right)^{1/M}} \tag{14}$$

$$G = \frac{\frac{1}{M}\sum_{k=0}^{M-1} \sigma_{y_k}^2}{(\prod_{k=0}^{M-1} \sigma_{y_k}^2)^{1/M}} \tag{15}$$

where σ_x^2, denotes the variance of the input signal, $\sigma_{y_k}^2$ the variance of the k-th subband signal, and $||f_k||^2$ the norm of the k-th synthesis filter. The resulting coding gains for the input signal from Figure 7 are stated in Table 3.

Table 3: Coding gains for different implementations

coeff. wordlength	input range	coding gain
16 bit	[−1 1]	1.97
16 bit	[−0.75 0.75]	8.96
16 bit	[−0.5 0.5]	9.77
8 bit	[−0.5 0.5]	9.75
4 bit	[−0.5 0.5]	7.66

5 CONCLUSIONS

In this paper we have described a structure for the implementation of biorthogonal cosine-modulated filter banks with fixed-point arithmetic. The approach is based a factorization of the polyphase components into zero-delay and maximum delay matrices and keeps the perfect reconstruction property despite quantization, rounding and overflow. Design examples demonstrate the influence of overflow, quantization, and rounding on the frequency selectivity and the coding gain.

REFERENCES

[1] G. T. D. Schuller and M. J. T. Smith, “A new framework for modulated perfect reconstruction filter banks,” *IEEE Trans. on Signal Processing*, vol. 44, August 1996.

[2] P. N. Heller, T. Karp, and T. Q. Nguyen, “A general formulation of modulated filter banks,” *IEEE Trans. on Signal Processing*, vol. 47, pp. 986–1002, April 1999.

[3] G. D. T. Schuller and T. Karp, “Modulated filter banks with arbitrary system delay: Efficient implementation and the time-varying case,” *IEEE Trans. on Signal Processing*, vol. 48, pp. 737–748, March 2000.

[4] T. Karp, A. Mertins, and T. Q. Nguyen, "Efficiently VLSI-realizable prototype filters for modulated filter banks," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Munich, Germany, May 1997.

[5] A. Mertins, "Subspace approach for the design of cosine-modulated filter banks with linear-phase prototype filter," *IEEE Trans. on Signal Processing*, vol. 46, pp. 2812–2818, October 1998.

[6] M. Bi, S. H. Ong, and Y. H. Ang, "Integer-modulated FIR filter banks for image compression," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 8, pp. 923–927, December 1998.

[7] A. Mertins, T. Karp, and J. Kliewer, "Design of perfect reconstruction integer-modulated filter banks," in *Proc. IEEE International Symposium on Signal Processing and its Applications*, Brisbane, Qld, Australia, vol. 2, pp. 591–594, August 1999.

[8] A. Mertins, T. Karp, and J. Kliewer, "Integer-modulated filter banks providing perfect reconstruction," in *Proc. EURASIP European Signal Processing Conference*, Tampere, Finland, September 2000.

[9] J. Liang and T. D. Tran, "Fast multiplierless approximation of the DCT with the lifting scheme," in *Proc. SPIE Applications of Digital Image Processing XXIII*, San Diego, CA, August 2000.

[10] A. Mertins and T. Karp, "Perfect reconstruction integer–modulated filter banks," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Salt Lake City, UT, May 2001.

[11] T. Karp, A. Mertins, and G. Schuller, "Recent trends in the design of biorthogonal modulated filter banks," in *Proc. TICSP Workshop on Transforms and Filter Banks*, Tampere, Finland, February 1998.

[12] T. Karp, A. Mertins, and G. Schuller, "Efficient biorthogonal cosine-modulated filter banks," *EURASIP Signal Processing*, vol. 81, pp. 997–1016, May 2001.

[13] R. Calderbank, I. Daubechies, W. Sweldens, and B.-L. Yeo, "Wavelet transforms that map integers to integers," *Appl. Comput. Harmon. Anal.*, vol. 5, no. 3, pp. 332–369, 1998.

[14] G. Strang and T. Nguyen, *Wavelets and Filter Banks.* Wellesley, MA: Wellesley-Cambridge Press, 1996.