QUATERNIONIC APPROACH TO 8-CHANNEL GENERAL PARAUNITARY FILTER BANK*)

Marek Parfieniuk and Alexander Petrovsky

Department of Real-Time Systems, Bialystok Technical University
Wiejska 45A, 15-351, Bialystok, Poland
phone: (+48 85) 746-90-50, fax: (+48 85) 746-90-57, email: marekpk@ii.pb.bialystok.pl, palex@it.org.by
web: www.pb.bialystok.pl

ABSTRACT

This paper presents an alternative factorization for 8-channel general paraunitary filter bank. The utilization of quaternion multiplications leads to a lattice structure being lossless regardless of coefficient quantization. Other advantages are reduced memory requirements and good suitability for FPGA and VLSI implementations. The shown decompositions of 8 x 8 orthogonal matrices can find other applications unrelated to filter banks.

1. INTRODUCTION

Paraunitary filter banks (PUFBs) [1] are very important among all subband processing systems, because they are the basis for orthogonal M-band wavelets. The most evident area of their application is image coding. So, intensive research effort devoted to them is not surprising.

The commonly used approach to the design of PUFBs exploits the lattice structures based on Givens rotations [2]. An equivalent method, invented subsequently, utilizes Householder reflections and dyadic-based building blocks [1]. Both approaches are an effective design tool but their practical utilization in fixed-point computational platforms is not so easy because the abovementioned building blocks lose their orthogonality under coefficient quantization [3], and the compensation of the related distortions is not trivial. The only robust approach to obtain a perfectly invertible subband processing systems is to employ lifting schemes [4].

The authors have recently presented an alternative quaternionic building block insensitive to coefficient quantization [5], [6]. The use of the component was demonstrated in 4-channel general and linear phase PUFBs [7], and 8-channel linear phase PUFB [8], for whom appropriate factorizations were provided. The main results were: structurally guaranteed perfect reconstruction (up to scaling) under a rough coefficient quantization, reduced memory requirements, and good suitability for FPGA and VLSI implementations, mitigating the disadvantage of increased computational complexity. The only weakness of the solution is its limitation to the 4- and 8-channel filter banks being very practical, however.

The aim of this contribution is to show and evaluate an application of quaternionic building block in 8-channel general PUFB, and thus to complete the earlier published results. By the way, several interesting factorizations for 8×8 orthogonal matrices are derived, which can be useable if any algorithm involving such matrices is implemented in hardware.

Notations:

Quaternions are denoted by upper-case characters. Matrices are indicated by bold-faced upper-case characters. $\mathbf{I}_n$ denotes the identity matrix of size $n\times n$. The transposition is denoted by the superscript T. Without loss of generality, all our considerations are restricted to special orthogonal matrices with the determinant +1.

*) This work was supported by Bialystok Technical University under the grant W/WI/2/05.

2. PRELIMINARIES ON QUATERNIONS

2.1 Essential properties

Quaternions are hypercomplex numbers of the form

$$Q = q_1 + q_2 i + q_3 j + q_4 k \qquad q_{i=1..4} \text{ - real numbers} \tag{1}$$

Although they are based on the three imaginary units: $i, j,$ and k, they are very similar to ordinary complex numbers. For example, the conjugate quaternion is simply

$$Q^* = q_1 - q_2 i - q_3 j - q_4 k \ . \tag{2}$$

The modulus (norm) is defined as

$$|Q| = \sqrt{q_1^2 + q_2^2 + q_3^2 + q_4^2} \tag{3}$$

and it equals unity for the unit quaternions.

2.2 Multiplication

In many cases it is very convenient to identify quaternions with 4-element vectors, i.e.

$$Q \Leftrightarrow \begin{bmatrix} q_1 & q_2 & q_3 & q_4 \end{bmatrix}^T . \tag{4}$$

In this notation, quaternion addition is equivalent to vector summation. In turn, quaternion product is written as

$$R = PQ \Leftrightarrow \tag{5}$$

$$\underbrace{\begin{bmatrix} p_1 & -p_2 & -p_3 & -p_4 \\ p_2 & p_1 & -p_4 & p_3 \\ p_3 & p_4 & p_1 & -p_2 \\ p_4 & -p_3 & p_2 & p_1 \end{bmatrix}}_{\mathbf{M}^+(P)} \times \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} = \underbrace{\begin{bmatrix} q_1 & -q_2 & -q_3 & -q_4 \\ q_2 & q_1 & q_4 & -q_3 \\ q_3 & -q_4 & q_1 & q_2 \\ q_4 & q_3 & -q_2 & q_1 \end{bmatrix}}_{\mathbf{M}^-(Q)} \times \begin{bmatrix} p_1 \\ p_2 \\ p_3 \\ p_4 \end{bmatrix}$$

It is obviously non-commutative as the multiplication matrix $\mathbf{M}^+(P)$ for the left operand differs from $\mathbf{M}^-(Q)$ for the right one.

Owing to the specific structures of the matrices, this operation requires only 8 real multiplications [9].

3. QUARTERNIONS FOR ORTOGONAL MATRICES

3.1 4 x 4 orthogonal matrix

Although, 8×8 orthogonal matrices are of interest when 8-channel PUFB is considered, we start with a result concerning the 4×4 case, being crucial for further investigations.

Theorem [10]:

For an arbitrary 4×4 orthogonal matrix $\mathbf{R}$, there exist a unique pair of unit quaternions P and Q such that

$$\mathbf{R} = \mathbf{M}^{+}(P) \cdot \mathbf{M}^{-}(Q) \tag{6}$$

The proof of this little known fact can be found in [10]. It should be noted that the above product is commutative, and it is satisfied by the negative quaternions $-P$ and $-Q$ as well.

3.2 8 x 8 block diagonal orthogonal matrices

Given (6), we can deal with 8×8 block diagonal matrices.

Theorem:

Any block diagonal matrix composed from two arbitrary 4×4 orthogonal matrices U and V, can be modelled with four unit quaternions in the following manner

$$\begin{bmatrix} \mathbf{U} & \mathbf{0} \\ \mathbf{0} & \mathbf{V} \end{bmatrix} = \left(\operatorname{diag}\left\{ \mathbf{M}^{-}(P), \mathbf{M}^{-}(P) \right\} \cdot \operatorname{diag}\left\{ \mathbf{M}^{-}\left(Q^{*}\right), \mathbf{M}^{-}(Q) \right\} \right) \cdot \\ \cdot \left(\operatorname{diag}\left\{ \mathbf{M}^{+}\left(R^{*}\right), \mathbf{M}^{+}(R) \right\} \cdot \operatorname{diag}\left\{ \mathbf{M}^{+}(S), \mathbf{M}^{+}(S) \right\} \right). \tag{7}$$

The first step of the proof of (7) directly utilizes (6).

$$\begin{bmatrix} \mathbf{U} & \mathbf{0} \\ \mathbf{0} & \mathbf{V} \end{bmatrix} = \begin{bmatrix} \mathbf{M}^{-}(A) & \mathbf{0} \\ \mathbf{0} & \mathbf{M}^{-}(B) \end{bmatrix} \cdot \begin{bmatrix} \mathbf{M}^{+}(C) & \mathbf{0} \\ \mathbf{0} & \mathbf{M}^{+}(D) \end{bmatrix} \tag{8}$$

Then, exploiting the fact that both multiplication matrices form algebraic groups, we can assume the expansion

$$\begin{bmatrix} \mathbf{M}^{\pm}(A) & \mathbf{0} \\ \mathbf{0} & \mathbf{M}^{\pm}(B) \end{bmatrix} = \begin{bmatrix} \mathbf{M}^{\pm}\left(E^{*}\right) & \mathbf{0} \\ \mathbf{0} & \mathbf{M}^{\pm}(E) \end{bmatrix} \cdot \begin{bmatrix} \mathbf{M}^{\pm}(F) & \mathbf{0} \\ \mathbf{0} & \mathbf{M}^{\pm}(F) \end{bmatrix} \tag{9}$$

for both the left and right factors of the product in (8). The matrix equation written in this way, has an easily obtainable, unique solution.

It should be noted that the factors in (8) as well as in (9) can be reordered. Hence, any reordering of (7) is allowed, not disturbing the parentheses. The above factorizations are also unique up to sign.

The factorization (7) corresponds to the lattice structure depicted in Fig. 1.

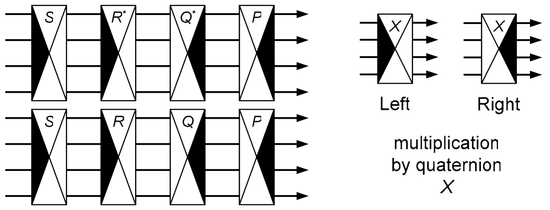

Figure 1. Lattice structure implementing (7).

It simplifies significantly if $\mathbf{U}$ is the identity matrix.

$$\begin{bmatrix} \mathbf{I}_4 & \mathbf{0} \\ \mathbf{0} & \mathbf{V} \end{bmatrix} = \operatorname{diag}\left\{ \mathbf{M}^{-}(P), \mathbf{M}^{-}(P) \right\} \cdot \operatorname{diag}\left\{ \mathbf{M}^{-}\left(P^{*}\right), \mathbf{M}^{-}(P) \right\} \cdot \\ \cdot \operatorname{diag}\left\{ \mathbf{M}^{+}\left(Q^{*}\right), \mathbf{M}^{+}(Q) \right\} \cdot \operatorname{diag}\left\{ \mathbf{M}^{+}(Q), \mathbf{M}^{+}(Q) \right\} \tag{10}$$

as only two different unit quaternions are involved in this case.

The permutation matrix

$$\mathbf{P} = \begin{bmatrix} \mathbf{0} & \mathbf{I}_6 \\ \mathbf{I}_2 & \mathbf{0} \end{bmatrix} \tag{11}$$

allow us to apply (10) to a matrix of a slightly different structure as

$$\operatorname{diag}\left\{ \mathbf{I}_2, \mathbf{V}, \mathbf{I}_2 \right\} = \mathbf{P} \cdot \operatorname{diag}\left\{ \mathbf{I}_4, \mathbf{V} \right\} \cdot \mathbf{P}^{T}, \tag{12}$$

The corresponding schema transformation is shown in Fig. 2.

As $\mathbf{P}$ is orthogonal, it is trivial to show that

$$\mathbf{X} \cdot \mathbf{P} \cdot \mathbf{Y} = \mathbf{P} \cdot \mathbf{P}^{T} \cdot \mathbf{X} \cdot \mathbf{P} \cdot \mathbf{Y} \tag{13}$$

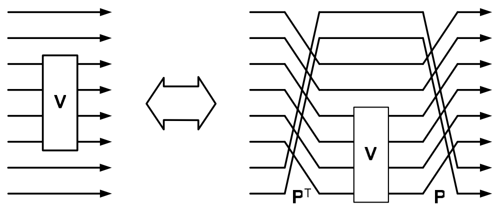

Figure 2. Structural conversion corresponding to (12).

for arbitrary $\mathbf{X}$ and $\mathbf{Y}$. But if we take the block diagonal matrices from (7), the following equivalence

$$\mathbf{P}^{T} \cdot \operatorname{diag}\left\{ \mathbf{M}^{+}(P), \mathbf{M}^{+}(P) \right\} \cdot \mathbf{P} \cdot \operatorname{diag}\left\{ \mathbf{M}^{+}(Q), \mathbf{M}^{+}(Q) \right\} = \\ = \begin{bmatrix} \mathbf{M}^{+}(P') & \mathbf{M}^{+}(Q') \\ \mathbf{M}^{+}(Q') & \mathbf{M}^{+}(P') \end{bmatrix} \qquad \begin{matrix} P' = (p_1 + ip_2)Q \\ Q' = -(jp_3 + kp_4)Q \end{matrix} \tag{14}$$

surprisingly turns out to be valid (also for $\mathbf{M}^{-}$ replaced with $\mathbf{M}^{+}$ or/and $\mathbf{P}$ replaced with $\mathbf{P}^{T}$). Its structural meaning is explained in Fig. 3.

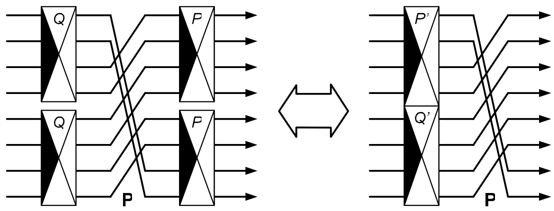

Figure 3. Schema transformation corresponding to (14).

This result clearly offers some structure simplification. It should be noted that the new quaternions are not unit, but the norm of their combination is obviously unity.

3.3 Reduction of 8 x 8 orthogonal matrix

Other arrangements of quaternion multiplication matrices, similar to that from the previous section, can be used to factorize 8×8 general orthogonal matrices.

Theorem:

An arbitrary 8×8 orthogonal matrix $\mathbf{R}$ can be represented as the product

$$\mathbf{R} = \underbrace{\begin{bmatrix} \mathbf{M}^{+}\left(P^{*}\right) & -\mathbf{M}^{-}\left(Q^{*}\right) \\ \mathbf{M}^{-}(Q) & \mathbf{M}^{+}(P) \end{bmatrix}}_{\mathbf{G}} \begin{bmatrix} 1 & \mathbf{0} \\ \mathbf{0} & \mathbf{R}' \end{bmatrix} \\ P = r_{11} - ir_{21} - jr_{31} - kr_{41} \quad Q = -r_{51} - ir_{61} - jr_{71} - kr_{81} \tag{15}$$

where $\mathbf{R}'$ is a 7×7 orthogonal matrix, and r_{ij} refers to the (i, j) entry of $\mathbf{R}$.

The orthogonality of $\mathbf{G}$ can be easily verified. Thus the product $\mathbf{G}^{T}\mathbf{R}$ of orthogonal matrices must also be orthogonal. Moreover, P and Q are selected to make the dot product of the first columns of $\mathbf{R}$ and $\mathbf{G}$ equal unity. This is the value of the $(1,1)$ entry of $\mathbf{G}^{T}\mathbf{R}$, so all the remaining entries in the first row and column of this orthogonal matrix must be zeros. Hence, the rest of the entries forms an orthogonal matrix $\mathbf{R}'$ of size 7×7.

The corresponding structural conversion is shown in Fig. 4.

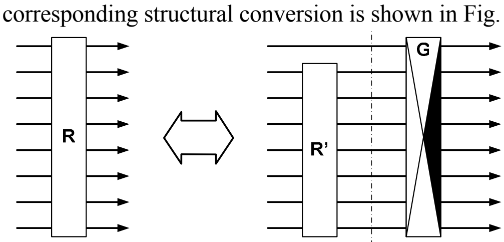

Figure 4. Lattice conversion corresponding to (15)

It should be noted that such a reduction can be done using a sequence of Givens rotations or a single Householder reflection, but those matrices lose orthogonality under quantization of their entries.

3.4 8 x 8 orthogonal matrix

To completely decompose an arbitrary 8×8 orthogonal matrix $\mathbf{R}$ into quaternion multiplications, let us look at its QR factorization using Givens rotations. The corresponding lattice structure is depicted in Fig. 5(a), which also explains how to group the rotations into five groups representing 4×4 orthogonal matrices. Moreover, these matrices are embedded into 8×8 identity matrices, in the manners considered in Sec. 3.2. Thus, using (12), the matrix $\mathbf{R}$ can be represented as the product

$$\mathbf{R}=\begin{bmatrix}\mathbf{I}_4 & \mathbf{0}\\ \mathbf{0} & \mathbf{V}_4\end{bmatrix}\mathbf{P}\begin{bmatrix}\mathbf{I}_4 & \mathbf{0}\\ \mathbf{0} & \mathbf{V}_3\end{bmatrix}\mathbf{P}^T\begin{bmatrix}\mathbf{U}_2 & \mathbf{0}\\ \mathbf{0} & \mathbf{V}_2\end{bmatrix}\mathbf{P}\begin{bmatrix}\mathbf{I}_4 & \mathbf{0}\\ \mathbf{0} & \mathbf{V}_1\end{bmatrix}\mathbf{P}^T\begin{bmatrix}\mathbf{I}_4 & \mathbf{0}\\ \mathbf{0} & \mathbf{V}_0\end{bmatrix} \quad (16)$$

The decompositions (7) and (10) can be employed here, utilizing quaternionic building blocks. This leads to the structure depicted in Fig. 5(b). The reorderings of (7) and the transformation (14) can subsequently be applied to obtain the equivalent lattice in Fig. 5(c).

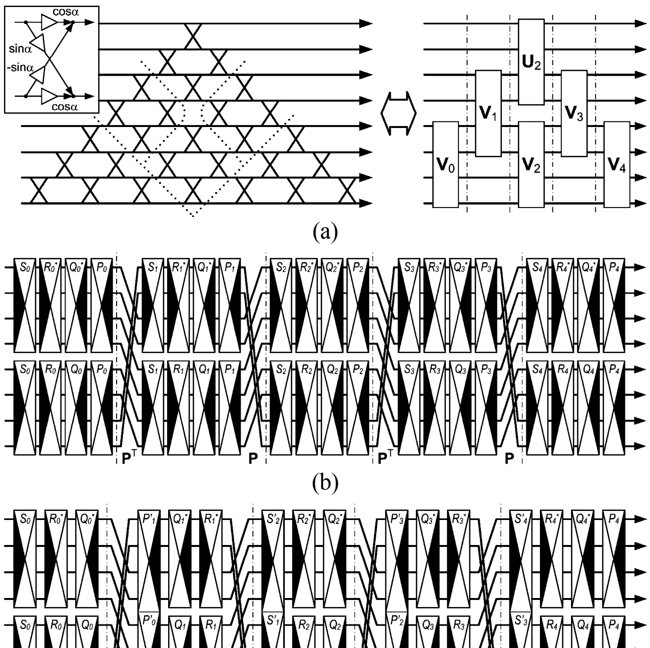

Figure 5. Conversion of lattice for 8 x 8 matrix from Givens rotations to quaternionic building blocks.

3.5 Orthogonality and losslessness under quantization

The factorizations considered in Sections 3.2 through 3.4 were designed to utilize orthogonal matrices having constant column, as this is necessary for quantization robustness [3]. Such matrices were composed of the quaternion multiplication matrices (5) to have the same set of absolute values in every row and column. This structural property guarantees that the matrix remains orthogonal (up to scaling) after its quantization. Moreover, the product of quantized matrices is orthogonal too. In turn, the corresponding lattice structures are lossless.

This unique property can not be obtained by factorizing 8×8 matrices with Givens rotations, Householder reflections, or mixing Givens rotations with quaternionic building block [6].

3.6 Complexity considerations

The matrices used in the above factorizations, arranged from quaternion multiplication matrices, have specific structures offering computational and memory benefits.

First, it is sufficient to store in memory only 4 numbers instead of each block diagonal matrix on the left side of (7). The product of such a matrix with a vector can be computed with 16 real multiplications. The same computational complexity is related to the matrix on the left side of (14). In this case, 8 numbers must be stored, however. 8 coefficients is also sufficient to describe the matrix $\mathbf{G}$ in (15), but 32 multiplications are necessary in this case.

Unfortunately, the lossless property comes at the cost of superfluous computations in all the decompositions. The computational complexity related to the factorization (7) is $4\times 16=64$ real multiplications, though the factorized matrix has only 32 non-zero entries. But only $4\times 4=16$ numbers must be stored in memory.

Considering the quaternionic factorization of a 8×8 general matrix from Fig. 5(b), the required number of multiplications is $20\times 16=320$, 5 times more than using the matrix directly. The small savings are in memory, as only $16+4\times 8=48$ coefficients must be stored.

A moderate reduction of computational complexity can be noticed in the lattice in Fig. 5(c), needing $16\times 16=256$ real multiplications. However, 72 numbers must be stored in this case.

It should be mentioned that the above analysis is rather inadequate in the cases of hardware (FPGA or VLSI) implementations. Namely, quaternionic building block is well suited to these technologies [6], as it can be realized using only shift-and-add operations. Simultaneously, there are wide parallelization and pipelining possibilities. Moreover, the related lattice structures have very regular layouts simplifying circuit synthesis and allowing area minimization. These advantages extend to the factorizations considered here.

4. 8-CHANNEL GENERAL PUFB

4.1 Traditional approach

An adaptation of the classical PUFB design method [2] to the 8-channel case, leads to the following factorization

$$\mathbf{E}(z)=\mathbf{\Theta}_{N-1}\mathbf{\Lambda}(z)\mathbf{\Theta}_{N-2}\cdots\mathbf{\Theta}_1\mathbf{\Lambda}(z)\mathbf{E}_0 \qquad \mathbf{\Lambda}(z)=\begin{bmatrix}z^{-1} & 0\\ 0 & \mathbf{I}_7\end{bmatrix} \quad (17)$$

of the polyphase transfer matrix $\mathbf{E}(z)$. $\mathbf{E}_0$ is an arbitrary orthogonal matrix (possessing 28 degrees of freedom) and each $\mathbf{\Theta}_i$ is a sequence of 7 Givens rotations. Since all these matrices are sensitive to coefficient quantization, the transfer matrix losses the paraunitary property.

4.2 Alternative quaternionic lattice

Given the results from Section 4, one can convert (17) to an alternative form guarantying the lossless property even under a severe coefficient quantization.

Namely, $\mathbf{E}_0$ can be modeled as in Section 3.4. In turn, the matrix $\mathbf{G}$ from (15) can replace a sequence of Givens rotations in $\mathbf{\Theta}_i$. This is possible because both $\mathbf{G}$ and Givens rotations are kind of a remainder resulting from the reduction of an arbitrary orthogonal $\mathbf{\Theta}_i$ to a 7×7 submatrix which can be moved to the adjacent stage $\mathbf{\Theta}_{i-1}$ and finally included in $\mathbf{E}_0$.

The synthesis filter bank is constructed simply by multiplying the inverses of the factors of (17) in reverse order.

5. DESIGN EXAMPLE

A practical evaluation of the presented approach was performed on 8-channel filter bank with filters of the length $L = 16$, designed to maximize the stopband attenuation. The coefficient optimization was done using the standard MATLAB function *fminunc*. The infinite precision coefficients determined for the traditional lattice based on Givens rotations, were then converted to the coefficients of the equivalent quaternionic factorization.

Given two equivalent systems, the quantization of their coefficients was performed using the CSD representation with wordlength 8 and the number of non-zeros equal 2. It should be explained that, in the traditional approach, the matrices $\mathbf{\Theta}_1$ and $\mathbf{E}_0$ were directly quantized instead of particular elementary rotations. On the contrary, the matrices obtained from the factorization of $\mathbf{\Theta}_1$ and $\mathbf{E}_0$ were quantized for the quaternionic lattice - their coefficients are shown in Table 1.

$\mathbf{E}_0$

	P₀ = Q₀			***R₀ = S₀***	
0.8542159	0.8750000	$+2^{-0}-2^{-3}$	0.6269166	0.6250000	$+2^{-1}+2^{-3}$
0.2986928	0.3125000	$+2^{-2}+2^{-4}$	0.5112678	0.5078125	$+2^{-1}+2^{-7}$
-0.3555262	-0.3750000	$-2^{-1}+2^{-3}$	-0.5475108	-0.5625000	$-2^{-1}-2^{-4}$
0.2338779	0.2343750	$+2^{-2}-2^{-6}$	0.2140390	0.2187500	$+2^{-2}-2^{-5}$
	P₁ = Q₁			***R₁ = S₁***	
0.5646208	0.5625000	$+2^{-1}+2^{-4}$	0.5237621	0.5312500	$+2^{-1}+2^{-5}$
0.5810943	0.5625000	$+2^{-1}+2^{-4}$	-0.4394914	-0.4375000	$-2^{-1}+2^{-4}$
0.2654356	0.2656250	$+2^{-2}+2^{-6}$	-0.2007534	-0.1875000	$-2^{-2}+2^{-4}$
0.5225674	0.5156250	$+2^{-1}+2^{-6}$	-0.7015830	-0.7500000	$-2^{-0}+2^{-2}$
	P₂			***Q₂***	
-0.1091407	-0.1093750	$-2^{-3}+2^{-6}$	0.9492411	0.9375000	$+2^{-0}-2^{-4}$
0.9270083	0.9375000	$+2^{-0}-2^{-4}$	-0.1055311	-0.1093750	$-2^{-3}+2^{-6}$
-0.2690411	-0.2656250	$-2^{-2}-2^{-6}$	0.2673831	0.2656250	$+2^{-2}+2^{-6}$
-0.2374043	-0.2343750	$-2^{-2}+2^{-6}$	-0.1277138	-0.1250000	-2^{-3}
	R₂			***S₂***	
0.8534744	0.8750000	$+2^{-0}-2^{-3}$	0.7985220	0.7500000	$+2^{-0}-2^{-2}$
-0.3314303	-0.3125000	$-2^{-2}-2^{-4}$	0.1188791	0.1171875	$+2^{-3}-2^{-7}$
-0.0336804	-0.0312500	-2^{-5}	-0.1011158	-0.0937500	$-2^{-3}+2^{-5}$
0.4007506	0.3750000	$+2^{-1}-2^{-3}$	-0.5813828	-0.5625000	$-2^{-1}-2^{-4}$
	P₃ = Q₃			***R₃ = S₃***	
0.7065719	0.7500000	$+2^{-0}-2^{-2}$	0.7596650	0.7500000	$+2^{-0}-2^{-2}$
0.6743503	0.6250000	$+2^{-1}+2^{-3}$	0.0150397	0.0156250	$+2^{-6}$
0.2143848	0.2187500	$+2^{-2}-2^{-5}$	0.0047813	0.0078125	$+2^{-7}$
-0.0068558	-0.0078125	-2^{-7}	-0.6501231	-0.6250000	$-2^{-1}-2^{-3}$
	P₄ = Q₄			***R₄ = S₄***	
0.5636663	0.5625000	$+2^{-1}+2^{-4}$	0.9033501	0.8750000	$+2^{-0}-2^{-3}$
0.5889034	0.5625000	$+2^{-1}+2^{-4}$	-0.2286368	-0.2343750	$-2^{-2}+2^{-6}$
-0.4492186	-0.4375000	$-2^{-1}+2^{-4}$	-0.2494317	-0.2500000	-2^{-2}
0.3656169	0.3750000	$+2^{-1}-2^{-3}$	-0.2635671	-0.2656250	$-2^{-2}-2^{-6}$

$\mathbf{\Theta}_1$

	P			***Q***	
-0.1331993	-0.1328125	$-2^{-3}-2^{-7}$	0.5390552	0.5312500	$+2^{-1}+2^{-5}$
0.2383718	0.2421875	$+2^{-2}-2^{-7}$	-0.3512911	-0.3750000	$-2^{-1}+2^{-3}$
-0.3627984	-0.3750000	$-2^{-1}+2^{-3}$	0.2322585	0.2343750	$+2^{-2}-2^{-6}$
0.5557832	0.5625000	$+2^{-1}+2^{-4}$	-0.1303431	-0.1328125	$-2^{-3}-2^{-7}$

Table 1. Quaternion lattice coefficients for designed PUFB (from left to right: precise value, quantized value, CSD expansion).

The effects of quantization were then analyzed in terms of the channel responses $H_k(z), k = 0..7$, as well as the total magnitude response of the analysis/synthesis system.

$$\left|T\left(e^{j\omega}\right)\right| = \sum_{k=0}^{7}\left|H_k\left(e^{j\omega}\right)\right|^2 \tag{18}$$

Unlike traditional lattice with Givens rotations, the transfer function of the quaternionic lattice does not introduce any distortions, as $T(z) = c^2 z^{-15}$. The signal level is only decreased by ~1dB and this can be easily corrected as

$$c^2 = \sum_{k=0}^{7}\left|H_k\left(e^{j\omega}\right)\right|^2\Bigg|_{\omega=0} = \sum_{k=0}^{7}\left|\sum_{l=0}^{L-1} h_k\left(l\right)\right|^2 \tag{19}$$

where $h_k(l)$ denotes the l th coefficient of the k th filter.

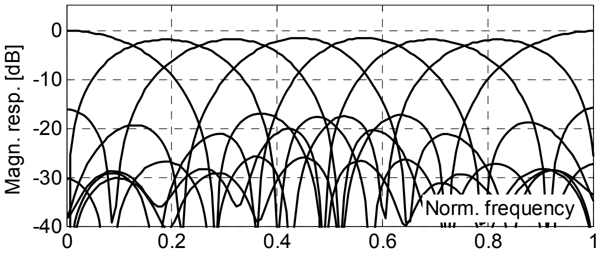

Figure 6. Magnitude response.

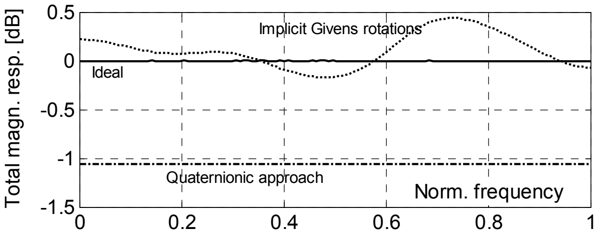

Figure 7. Total magnitude response.

6. CONCLUSIONS

The application of quaternionic building block in 8-channel general PUFB was shown, leading to the lattice maintaining the lossless property in spite of coefficient quantization. Theoretical deliberations were supported by one, but quite detailed design example.

The developed quaternionic factorizations of several classes of 8×8 orthogonal matrices seem to be quite interesting and applicable in other areas of digital signal processing, unrelated to PUFBs.

REFERENCES

[1] P. P. Vaidyanathan, *Multirate Systems and Filter Banks*. Englewood Cliffs, NJ: Prentice-Hall, 1993.

[2] Z. Doganata, P. P. Vaidyanathan, T. Q. Nguyen, "General synthesis procedure for FIR lossless transfer matrices, for perfect-reconstruction multirate filterbank applications," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 36, pp. 1561–1574, Oct. 1988.

[3] P. P. Vaidyanathan, "On coefficient-quantization and computational roundoff effects in lossless multirate filter banks," *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. 39, pp. 1006-1008, April 1991.

[4] Y.-J. Chen, K.S. Amaratunga, “M-Channel Lifting Factorization of Perfect Reconstruction Filter Banks and Reversible M-Band Wavelet Transforms,”, IEEE Trans. Circuits Syst. II, vol. 50, pp. 963-976, December 2003.

[5] M. Parfieniuk, A. Petrovsky, "Quaternionic Building Block For Paraunitary Filter Banks," *in Proc. EUSIPCO 2004*, Vienna, Austria, Sept. 6-10. 2004, pp. 1237-1240.

[6] M. Parfieniuk, A. Petrovsky, "Implementation Perspectives of Quaternionic Component for Paraunitary Filter Banks," in *Proc. SMMSP 2004*, Vienna, Austria, Sept. 11-12. 2004, pp. 151-158.

[7] M. Parfieniuk, A. Petrovsky, "Linear phase paraunitary filter banks based on quaternionic component," in *Proc. ICSES’04*, Poznan, Poland, Sept. 13-15. 2004, pp. 203-206.

[8] M. Parfieniuk, A. Petrovsky, "Hypercomplex Factorizations for 8-Channel Linear Phase Paraunitary Filter Banks," in *Proc. DSPA 2005*, Moscow, Russia, March 16-18. 2005, (accepted).

[9] T. D. Howell, J.-C. Lafon, "The complexity of the quaternion product," TR 75-245, Cornell Univ., http://citeseer.nj.nec.com/-howell75complexity.html, June 1975.

[10] H. G. Baker, "Quaternions and Orthogonal 4x4 Real Matrices," http://www.whisqu.se/per/docs/math33.htm, June 1996.