論　文

リフティング構造に基づくパラユニタリフィルタバンクの設計とロッシー及びロスレス画像符号化への応用

鈴木　大三[†a)]　田中　雄一[†]　池原　雅章[†]

Lifting-Based Paraunitary Filterbanks for Lossy/Lossless Image Coding

Taizo SUZUKI[†a)], Yuichi TANAKA[†], and Masaaki IKEHARA[†]

あらまし　本論文では，ロスレス画像符号化に適したリフティング構造に基づく M 分割パラユニタリフィルタバンクの設計法を提案する．近年，パラユニタリフィルタバンクのリフティング因数分解が提案され，ロスレス画像符号化への応用が検討されているが，ラウンディング総数が多いため，ロスレス画像符号化に適用しても十分な性能が得られない．そこで，Householder 分解を用いたパラユニタリフィルタバンクに注目する．まず，Householder 分解と Givens 分解における対応関係を示し，Householder 分解を用いた場合におけるパラメータの冗長性を除去した新しい構造を提案する．この冗長性の除去されたパラユニタリフィルタバンクをリフティング構造に因数分解すると，ラウンディングの総数を大幅に削減することができる．最後にロッシー及びロスレス画像符号化に適用することで従来法と比較し，提案法の優位性を示す．

キーワード　パラユニタリフィルタバンク，Householder 分解，パラメータの冗長性，リフティング構造，ロスレス画像符号化

1. まえがき

ディジタル信号処理技術における静止及び動画像符号化では，DCT [2] やウェーブレット変換 [9] を含むフィルタバンクが広く用いられている．典型的な分割数 M のフィルタバンクの構造を図 1 に示す．一般的にフィルタバンクはフィルタ，デシメータ，インターポレータで構成され，分割側，合成側のフィルタは，それぞれのポリフェーズ行列を $\mathbf{E}(z)$，$\mathbf{R}(z)$ とすると，以下の式で表すことができる [3]（図 2）．

$$\begin{aligned}\left[H_0(z)\,H_1(z)\cdots\,H_{M-1}(z)\right]^T &= \mathbf{E}(z^M)\mathbf{e}(z)^T\\ \left[F_0(z)\,F_1(z)\cdots\,F_{M-1}(z)\right] &= \mathbf{e}(z)\mathbf{R}(z^M)\end{aligned} \tag{1}$$

ただし，$\mathbf{e}(z)=\left[1\ z^{-1}\ \cdots\ z^{-(M-1)}\right]$ である．

従来，画像符号化において優れた性能を示している重複直交変換（LOT）[1]，その一般化である一般化重複直交変換（GenLOT）[5] は，フィルタバンクという視点から考えると，線形位相パラユニタリフィルタバンク（LPPUFB）と等価であるといえる [4]．しかし，LPPUFB では線形位相性という制限があるために，フィルタとしての自由度を有効に利用していないとも

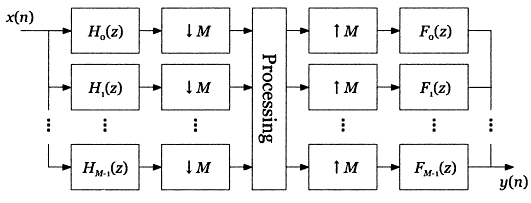


図 1　M 分割フィルタバンク
Fig. 1　An M-channel filterbank.

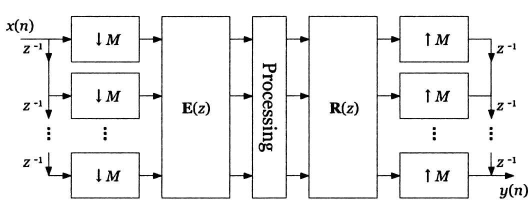


図 2　フィルタバンクのポリフェーズ構造
Fig. 2　A polyphase structure of filterbank.

[†] 慶應義塾大学理工学部電子工学科，横浜市
Department of Electronics and Electrical Engineering, Keio University, Yokohama-shi, 223–8522 Japan
a) E-mail: suzuki@tkhm.elec.keio.ac.jp

考えられる．そこで線形位相性をもたない，つまり非線形位相パラユニタリフィルタバンク（PUFB）の研究が進んでいる [8], [11]．PUFB は LPPUFB に比べてより高い阻止域減衰量と符号化利得が得られることが示されている．

一方，画像符号化には，完全に復元できるロスレス（可逆）画像符号化と，復元誤差を許容し，圧縮率の高いロッシー（非可逆）画像符号化の二つがある．インターネットやマルチメディア信号などの分野では画質よりも圧縮率が重視されるために，ロッシー画像符号化がよく用いられる．それに対し，医療や美術などの分野では高画質な画像が必要とされ，このような分野においてはロスレス画像符号化がよく用いられる．JPEG2000 では，同一の構造でロッシー・ロスレス統合形の画像符号化を実現することができ，この際リフティング構造という技術が重要になる [9]．リフティング構造を用いたロスレス画像符号化においてはラウンディング処理を用いて少数値を整数値に丸める操作が行われる．この際誤差が生じ，圧縮率が低下する可能性がある．すなわちロッシー画像符号化と異なる点はリフティング部で誤差が生じる点である．そこで実現規模の面からも，各々のラウンディング操作によって誤差が累積するのを防ぐためにも，ラウンディング総数ができるだけ少ないことが望ましい．

近年, [10] において，M 分割 PUFB のリフティング因数分解が示され，PUFB をロッシー画像符号化のみならず，ロスレス画像符号化にも応用することが可能となった．しかし, [10] で述べられた方法では，パラメータとラウンディング総数が冗長であり，ロスレス画像符号化に適用しても十分な性能は得られない．また, [11] において示された PUFB も，Givens 回転行列に対しリフティング因数分解が可能であるのでパラメータの冗長性を除去できるが，ラウンディング操作をまとめるには適しておらず，これもロスレス画像符号化に適しているとはいえない．

本論文では，以上のような背景のもと，ロスレス画像符号化に適したリフティング構造に基づく M 分割 PUFB（Lifting-Based PUFB：LBPUFB）を提案する．まず, [11] において示されたラティス構造をもとに，冗長性を除去した Householder 分解を用いた新しいラティス構造を設計する．これをリフティング構造に因数分解すると，ラウンディング総数も大幅に削減することができる．最後に，提案リフティング構造をもつ LBPUFB をロッシー画像符号化とロスレス画像符号化へと実際に応用し，従来法と比較することにより，その優位性を示す．

表記：$\mathbf{I}$，$\mathbf{M}^{(N)}$，$\mathbf{M}^{\dagger}$ はそれぞれ単位行列，$N \times N$ 正方行列，行列の共役転置である．また，$\mathbf{m} = [m_1, \cdots, m_N]^T$ としたとき，$\| \mathbf{m} \| = \sqrt{\sum_{i=1}^{N} | m_i |^2}$ である．

2. パラユニタリフィルタバンク（PUFB）

2.1 ラティス構造

式 (1) において，PUFB は，

$$\mathbf{E}^{\dagger}(z^{-1})\mathbf{E}(z) = \mathbf{I}, \quad \mathbf{R}(z) = \mathbf{E}^{\dagger}(z^{-1}) \tag{2}$$

を満たすフィルタバンクとして表すことができる．本論文では，分割数 M（偶数），フィルタ長 MK ($K \in \mathbb{N}$) の実係数 PUFB を考える．その際のポリフェーズ行列 $\mathbf{E}(z)$ は次式のようなラティス構造で表現できる [8]．

$$\mathbf{E}(z) = \mathbf{X}_{K-1}\mathbf{\Lambda}_{K-1}(z)\cdots\mathbf{X}_1(z)\mathbf{\Lambda}_1(z)\mathbf{X}_0 \tag{3}$$

ここで $\mathbf{X}_k$ は $\mathbf{X}_k^{\dagger}\mathbf{X}_k = \mathbf{I}$ を満たす $M \times M$ の任意のユニタリ行列であり，また $\mathbf{\Lambda}_k(z)$ は，

$$\mathbf{\Lambda}_k(z) = \begin{bmatrix} \mathbf{I}^{(M-\gamma_k)} & \mathbf{0} \\ \mathbf{0} & z^{-1}\mathbf{I}^{(\gamma_k)} \end{bmatrix} \tag{4}$$

で表され，γ_k は $1 \leq \gamma_k < M$ の任意の整数である．本論文では簡単のため，$\gamma_k = M/2$ とする．以下，簡単のため $\mathbf{\Lambda}_k(z)$ を $\mathbf{\Lambda}(z)$ と表記する．

ユニタリ行列を構成する方法として，Givens 分解に基づく方法や Householder 分解に基づく方法が提案されている [3]．

2.2 Givens 分解

$M \times M$ の任意のユニタリ行列 $\mathbf{X}$ は，以下の式のように $M(M-1)/2$ 個の回転角 $\theta_{i,j}$ の Givens 回転行列によって表すことができる [3]．

$$\mathbf{X} = \prod_{i=1}^{M-1} \prod_{j=i+1}^{M} \mathbf{\Theta}_{i,j} \tag{5}$$

ただし，

$$[\mathbf{\Theta}_{i,j}]_{k,l} = \begin{cases} 1 & : k = l \neq i \text{ or } j \\ \cos\theta_{i,j} & : k = l = i \text{ or } j \\ -\sin\theta_{i,j} & : k = i \text{ and } l = j \\ \sin\theta_{i,j} & : k = j \text{ and } l = i \\ 0 & : \text{otherwise} \end{cases} \tag{6}$$

である．なお，回転角の順序は任意である．これは同

一のユニタリ行列を Givens 回転行列の順序の異なる様々な構造で実現できることを意味している．この際，行列の交換則は成り立たないから，当然各構造の回転角は異なることに注意されたい．Givens 回転行列を用いると，式 (3) で表される PUFB のパラメータ数は，$KM(M-1)/2$ 個となる．しかし，このパラメータは冗長であり，冗長性を除去した一般的なラティス構造が提案されている [8]．

また, [8] とは別のラティス構造が示されており，以下の式で表すことができる [11]．

$$\mathbf{E}(z) = \left(\prod_{k=K-1}^{1} \widetilde{\mathbf{X}}_k \mathbf{\Lambda}(z) \right) \mathbf{X}_0 \tag{7}$$

この構造を図 3 に示す．ここで，$\widetilde{\mathbf{X}}_k$ ($k = 1, 2, \cdots, K-1$) は余剰なパラメータを除去した $M \times M$ の任意のユニタリ行列であり，

$$\widetilde{\mathbf{X}}_k = \prod_{i=1}^{M/2} \prod_{j=M/2+1}^{M} \mathbf{\Theta}_{i,j} \tag{8}$$

で表される．式 (8) は，M 本のパスを，遅延器を含む $M/2$ 本のパスとそれ以外の $M/2$ 本のパスの二つの部分に分離し，異なるパス間で Givens 回転行列を構成することを意味している．ここで，行列 $\mathbf{X}_0$ は $M \times M$ の任意のユニタリ行列であり，$M(M-1)/2$ 個のパラメータを含んでいる．また，$\widetilde{\mathbf{X}}_k$ は $(M/2)^2$ 個のパラメータを含んでいるから，フィルタバンク全体のパラメータ数は $(K-1)M^2/4 + M(M-1)/2$ となり, [8] と [11] のラティス構造のパラメータ数は全く同じとなる．

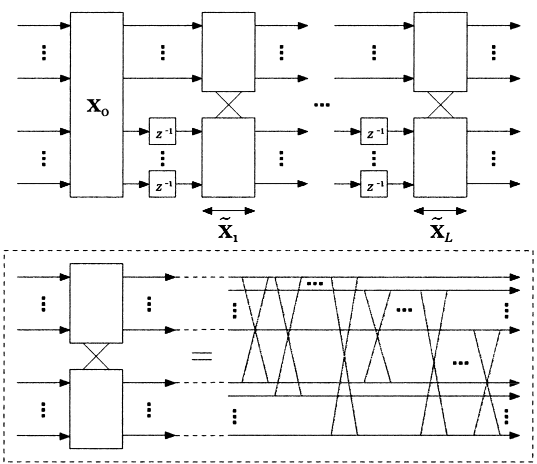


図 3　Givens 分解を用いた PUFB
Fig. 3　PUFB based on Givens factorization.

2. 3　Householder 分解

ユニタリ行列 $\mathbf{X}$ は，Householder 行列に因数分解することもできる [3]．ここで，

$$\mathbf{H}[\mathbf{p}] = \mathbf{I} - 2\mathbf{p}\mathbf{p}^{\dagger} \text{ where } \| \mathbf{p} \| = 1 \tag{9}$$

と定義される行列 $\mathbf{H}[\mathbf{p}]$ を Householder 行列と呼ぶ．また $\mathbf{p} = [p_0 \ p_1 \ \cdots \ p_{M-1}]^T$，$(\mathbf{H}[\mathbf{p}])^{-1} = \mathbf{H}[\mathbf{p}]$ である．以下に因数分解の手順を示す．

まず Householder 行列を用いて，$M \times M$ の任意のユニタリ行列 $\mathbf{X}^{(M)}$ のサイズを 1 小さくするようなパラメータ $\mathbf{p}_0$ が存在する．

$$\mathbf{H}[\mathbf{p}_0]\,\mathbf{X}^{(M)} = \left[\begin{array}{c|c} \mathbf{I}^{(1)} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{X}^{(M-1)} \end{array} \right] \tag{10}$$

同様に Householder 行列 $\mathbf{H}[\mathbf{p}_1]$ を用いて，以下のように $(M-1) \times (M-1)$ の任意のユニタリ行列 $\mathbf{X}^{(M-1)}$ のサイズを 1 小さくするような $\mathbf{p}_1$ が存在する．

$$\mathbf{H}[\mathbf{p}_1]\,\mathbf{H}[\mathbf{p}_0]\,\mathbf{X}^{(M)} = \left[\begin{array}{c|c} \mathbf{I}^{(2)} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{X}^{(M-2)} \end{array} \right] \tag{11}$$

これを繰り返していくと，

$$\mathbf{H}[\mathbf{p}_{M-2}] \cdots \mathbf{H}[\mathbf{p}_1]\,\mathbf{H}[\mathbf{p}_0]\,\mathbf{X}^{(M)} = \mathbf{I} \tag{12}$$

となり，最終的に単位行列になることが分かる．ここで $(\mathbf{H}[\mathbf{p}])^{-1} = \mathbf{H}[\mathbf{p}]$ より，$M \times M$ の任意のユニタリ行列 $\mathbf{X}^{(M)}$ は，以下のように $(M-1)$ 個の Householder 行列を用いて因数分解できることが分かる．

$$\mathbf{X}^{(M)} = \mathbf{H}[\mathbf{p}_0]\,\mathbf{H}[\mathbf{p}_1] \cdots \mathbf{H}[\mathbf{p}_{M-2}] \tag{13}$$

このときのパラメータ $\mathbf{p}_i$ は，

$$\begin{bmatrix} | & & | \\ \mathbf{p}_0 & \cdots & \mathbf{p}_{M-2} \\ | & & | \end{bmatrix} = \begin{bmatrix} \times & 0 & \cdots & 0 \\ \times & \times & \ddots & \vdots \\ \vdots & \vdots & \ddots & 0 \\ \times & \times & \cdots & \times \\ \times & \times & \cdots & \times \end{bmatrix} \tag{14}$$

となる．$\times$ は非ゼロの値をとり得ることを示す．よって，式 (3) の PUFB のポリフェーズ行列は，Householder 行列の積へと因数分解が可能である．

ここで，$\mathbf{p}_0$，$\mathbf{p}_1$，$\cdots$，$\mathbf{p}_{M-2}$ の各パラメータ数は，

それぞれ $M-1$，$M-2$，$\cdots$，1 個となる（$\| \mathbf{p}_j \| = 1$ の制約から自由度は 1 減少する）ので，PUFB の総パラメータ数は，$KM(M-1)/2$ 個となる．しかし，この構造のままではパラメータ数は冗長である．そこで，Givens 分解の冗長性除去を基本として，Householder 分解における冗長性除去について次章で述べ，冗長性を除去した LBPUFB を示す．

3. 冗長性を除去したリフティング構造に基づく PUFB（LBPUFB）

3.1 Householder 分解と Givens 分解における対応関係

まずはじめに Householder 分解と Givens 分解の構造を比較し，二つの対応関係を調べることにする．前述した Householder 分解を最後の段階（$\mathbf{H}[\mathbf{p}_{M-2}]$）まで行わずに，一つ前の段階（$\mathbf{H}[\mathbf{p}_{M-3}]$）でとどめ，これを移項すると，

$$\mathbf{X}^{(M)} = \mathbf{H}[\mathbf{p}_0] \cdots \mathbf{H}[\mathbf{p}_{M-3}] \left[\begin{array}{c|c} \mathbf{I}^{(M-2)} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{X}^{(2)} \end{array} \right] \tag{15}$$

となる．式 (13)，式 (15)，更に 2×2 の任意のユニタリ行列 $\mathbf{X}^{(2)}$ の Givens 分解を考えると，

$$\mathbf{H}[\mathbf{p}_{M-2}] = \left[\begin{array}{c|cc} \mathbf{I}^{(M-2)} & \multicolumn{2}{c}{\mathbf{0}} \\ \hline \mathbf{0} & \cos\theta_1 & -\sin\theta_1 \\ & \sin\theta_1 & \cos\theta_1 \end{array} \right] \tag{16}$$

と表すことができる．

次に，Householder 分解を最後の段階（$\mathbf{H}[\mathbf{p}_{M-2}]$）まで行わずに，二つ前の段階（$\mathbf{H}[\mathbf{p}_{M-4}]$）でとどめ，これを移項すると，

$$\mathbf{X}^{(M)} = \mathbf{H}[\mathbf{p}_0] \cdots \mathbf{H}[\mathbf{p}_{M-4}] \left[\begin{array}{c|c} \mathbf{I}^{(M-3)} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{X}^{(3)} \end{array} \right] \tag{17}$$

となる．同じように，式 (13) と 3×3 の任意のユニタリ行列 $\mathbf{X}^{(2)}$ の Givens 分解を考え，更に式 (16) より，

$$\mathbf{H}[\mathbf{p}_{M-3}] = \prod_{j=M-1}^{M} \mathbf{\Theta}_{M-2,j} \tag{18}$$

を得る．

以下同じようにして，

$$\mathbf{H}[\mathbf{p}_i] = \prod_{j=i+2}^{M} \mathbf{\Theta}_{i+1,j} \tag{19}$$

を導くことができる．

3.2 冗長性除去

次に，前述した Householder 分解と Givens 分解における対応関係を用いて，Householder 分解を用いた PUFB の場合でのパラメータの冗長性を除去する．

簡単のために，まずは $M = 4$ の場合の PUFB で考えていくことにする．Givens 分解を用いた場合の $\widetilde{\mathbf{X}}_k$ $(k = 1, 2, \cdots, K-1)$ は図 4 のように表すことができる．一方，これに対応させた Householder 分解を用いた $\widetilde{\mathbf{X}}_k$ $(k = 1, 2, \cdots, K-1)$ も，図 4 に点線の四角形で表す．式 (8) で示した冗長性を除去した PUFB のビルディングブロックの構造及び図 3 より，点線で表した Givens 回転行列は冗長であり，除去することができる．この冗長分を含む Householder 行列 $\mathbf{H}[\mathbf{p}_0]$ と $\mathbf{H}[\mathbf{p}_2]$ のパラメータも冗長なものを含んでおり，これを除去することができる．以下にその過程を示す．

（i）$\mathbf{H}[\mathbf{p}_2]$ の項は，Givens 回転行列は一つも含まれていないため，$\mathbf{H}[\mathbf{p}_2] = \mathbf{I}$ とおくことができる．つまり，$\mathbf{p}_2 = [0\ 0\ 0\ 0]^T$ となる．

（ii）$\mathbf{H}[\mathbf{p}_0]$ の項では，信号 $x(1)$ を用いない．よって，$\mathbf{p}_0 = [p_{0,1}\ 0\ p_{0,3}\ p_{0,4}]^T$ となる．

以上のように前節の対応関係から冗長性を除去することができる．この手順は分割数 M の場合に容易に拡張することができ，冗長性を除去したユニタリ行列の Householder 分解のパラメータは，

$$\begin{bmatrix} | & & | \\ \mathbf{p}_0 & \cdots & \mathbf{p}_{M/2-1} \\ | & & | \end{bmatrix} = \left[\begin{array}{cccc} \times & 0 & \cdots & 0 \\ 0 & \times & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \times \\ \hline \times & \times & \cdots & \times \\ \vdots & \vdots & \ddots & \vdots \\ \times & \times & \cdots & \times \end{array} \right] \tag{20}$$

と表すことができる．ただし，ファーストブロックを構成する $\mathbf{p}_i$ は式 (14) のままである．ゆえに，一般的な PUFB と同数のパラメータ（$(K-1)M^2/4 + M(M-1)/2$）で設計できることが分かる．これは，**2.2** の Givens 分解における自由度と同じである．

3.3 リフティング因数分解

Householder 行列 $\mathbf{H}[\mathbf{p}]$ はリフティング因数分解することができる [10]．ここで式 (14) を用いて，ユニタリ行列 $\mathbf{X}$ を構成する Householder 行列 $\mathbf{H}[\mathbf{p}_i]$ $(i = 0,$

$1, \cdots, M-2)$ をリフティング因数分解すると以下のように表される．

$$\mathbf{H}\left[\mathbf{p}_i\right] = \mathbf{A}_i \mathbf{B}_i \overline{\mathbf{A}}_i \tag{21}$$

ただし，

$$\mathbf{A}_i = \left[\begin{array}{c|cccc} \mathbf{I}^{(i)} & & & \mathbf{0} & \\ \hline & 1 & \alpha_{i+2} & \cdots & \alpha_M \\ \mathbf{0} & & 1 & & \\ & & & \ddots & \\ & & & & 1 \end{array}\right]$$

$$\mathbf{B}_i = \left[\begin{array}{c|cccc} \mathbf{I}^{(i)} & & & \mathbf{0} & \\ \hline & -1 & & & \\ & \beta_{i+2} & 1 & & \\ \mathbf{0} & \vdots & & \ddots & \\ & \beta_M & & & 1 \end{array}\right]$$

であり，α_k は設計変数，β_k はパラユニタリ性を満たすための変数

$$\alpha_k = -\frac{p_k}{p_{i+1}}, \quad \beta_k = -2p_k p_{i+1}$$

である．つまり，$\mathbf{A}_i$，$\mathbf{B}_i$ はそれぞれ $i+1$ 行目，$i+1$ 列目にのみ成分をもつ行列となっており，また，$\overline{\mathbf{A}}_i$ は $\mathbf{A}_i$ のパラメータ α_k が $-\alpha_k$ になったものである．図 5 (a) に Householder 分解を用いたユニタリ行列（$M=4$）を示す．

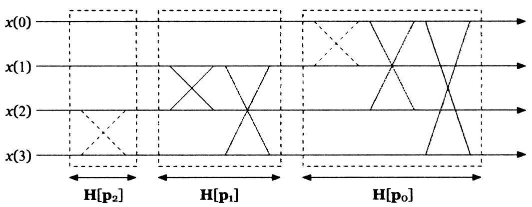


図 4　Householder 分解と Givens 分解における対応関係（$M=4$）

Fig. 4　Relationship between Householder and Givens factorization ($M=4$).

一方，冗長性を除去した PUFB のビルディングブロック $\widetilde{\mathbf{X}}_k$ $(k = 1, 2, \cdots, K-1)$ を構成する Householder 行列 $\mathbf{H}\left[\widetilde{\mathbf{p}}_i\right]$ $(i = 0, 1, \cdots, M/2-1)$ の場合，式 (20) を用いて以下のようにリフティング因数分解することができる．

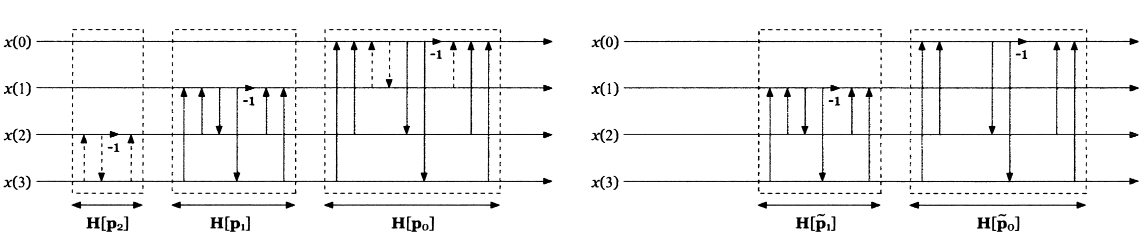
$$\mathbf{H}\left[\widetilde{\mathbf{p}}_i\right] = \widetilde{\mathbf{A}}_i \widetilde{\mathbf{B}}_i \overline{\widetilde{\mathbf{A}}}_i \tag{22}$$

ただし，

$$\widetilde{\mathbf{A}}_i = \left[\begin{array}{c|ccccccc} \mathbf{I}^{(i)} & & & & \mathbf{0} & & & \\ \hline & 1 & 0 & \cdots & 0 & \alpha_{M/2+1} & \cdots & \alpha_M \\ & & 1 & & & & & \\ & & & \ddots & & & & \\ \mathbf{0} & & & & 1 & & & \\ & & & & & 1 & & \\ & & & & & & \ddots & \\ & & & & & & & 1 \end{array}\right]$$

$$\widetilde{\mathbf{B}}_i = \left[\begin{array}{c|ccccccc} \mathbf{I}^{(i)} & & & & \mathbf{0} & & & \\ \hline & -1 & & & & & & \\ & 0 & 1 & & & & & \\ & \vdots & & \ddots & & & & \\ \mathbf{0} & 0 & & & 1 & & & \\ & \beta_{M/2+1} & & & & 1 & & \\ & \vdots & & & & & \ddots & \\ & \beta_M & & & & & & 1 \end{array}\right]$$

と表され，$\widetilde{\mathbf{A}}_i$，$\widetilde{\mathbf{B}}_i$ はそれぞれ $i+1$ 行目，$i+1$ 列目



(a)　(b)

図 5　Householder 分解を用いたユニタリ行列のリフティング因数分解（$M=4$）．(a) 冗長性除去前，(b) 冗長性除去後

Fig. 5　Lifting factorization of a unitary matrix based on Householder factorization ($M=4$). (a) Redundant structure. (b) Nonredundant structure.

$$
\mathbf{E}(z) = \widetilde{\mathbf{X}}_1 \mathbf{\Lambda}(z) \mathbf{X}_0 = \begin{bmatrix} 1 & & \alpha_9 & \alpha_{10} \\ & 1 & & \\ & & 1 & \\ & & & 1 \end{bmatrix} \begin{bmatrix} -1 & & & \\ & 1 & & \\ \beta_9 & & 1 & \\ \beta_{10} & & & 1 \end{bmatrix} \begin{bmatrix} 1 & & \overline{\alpha}_9 & \overline{\alpha}_{10} \\ & 1 & & \\ & & 1 & \\ & & & 1 \end{bmatrix} \begin{bmatrix} 1 & & & \\ & 1 & \alpha_7 & \alpha_8 \\ & & 1 & \\ & & & 1 \end{bmatrix} \begin{bmatrix} 1 & & & \\ & -1 & & \\ & \beta_7 & 1 & \\ & \beta_8 & & 1 \end{bmatrix} \begin{bmatrix} 1 & & & \\ & 1 & \overline{\alpha}_7 & \overline{\alpha}_8 \\ & & 1 & \\ & & & 1 \end{bmatrix} \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & z^{-1} & \\ & & & z^{-1} \end{bmatrix}
$$

$$
\begin{bmatrix} 1 & \alpha_4 & \alpha_5 & \alpha_6 \\ & 1 & & \\ & & 1 & \\ & & & 1 \end{bmatrix} \begin{bmatrix} -1 & & & \\ \beta_4 & 1 & & \\ \beta_5 & & 1 & \\ \beta_6 & & & 1 \end{bmatrix} \begin{bmatrix} 1 & \overline{\alpha}_4 & \overline{\alpha}_5 & \overline{\alpha}_6 \\ & 1 & & \\ & & 1 & \\ & & & 1 \end{bmatrix} \begin{bmatrix} 1 & & & \\ & 1 & \alpha_2 & \alpha_3 \\ & & 1 & \\ & & & 1 \end{bmatrix} \begin{bmatrix} 1 & & & \\ & -1 & & \\ & \beta_2 & 1 & \\ & \beta_3 & & 1 \end{bmatrix} \begin{bmatrix} 1 & & & \\ & 1 & \overline{\alpha}_2 & \overline{\alpha}_3 \\ & & 1 & \\ & & & 1 \end{bmatrix} \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & 1 & \alpha_1 \\ & & & 1 \end{bmatrix} \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & -1 & \\ & & \beta_1 & 1 \end{bmatrix} \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & 1 & \overline{\alpha}_1 \\ & & & 1 \end{bmatrix} \quad (23)
$$

にのみ成分をもち，かつ $M/2$ 個の成分しか含まれていない行列である．また，$\widetilde{\overline{\mathbf{A}}}_i$ は $\widetilde{\mathbf{A}}_i$ のパラメータ α_k が $-\alpha_k$ になったものである．図 5 (b) に図 5 (a) の構造から冗長性を除去した構造を示す．

PUFB をこのように冗長性を除去しながら Hoseholder 因数分解し，リフティング因数分解を適用することで，冗長性を除去した LBPUFB を設計することが可能である．図 6 (a) に Householder 分解を用いた PUFB のリフティング因数分解（$M = 4$）を，具体例として式 (23) に 4×8 LBPUFB を示す．

3.4 ラウンディング総数削減

ロスレス画像符号化においては，ラウンディング誤差が大きいと圧縮率の低下につながるため，ラウンディング操作をできるだけまとめて行うことが重要となる．また，Householder 行列はユニタリ行列内における Householder 行列の並べ換えが可能（ただし，並べ換え前とパラメータは同値でない）である．そこでラウンディング操作をできるだけまとめられるように Householder 行列を並べ換えて設計する（図 6 (b)）．図 6 (b) において黒丸は加算を，白丸はラウンディング操作を表す．図から分かるように，実際に信号処理に用いる際には 1 回のラウンディングに対して複数のリフティング処理された信号の和が対応しているため，結果的にラウンディング総数を削減することができる．また，本論文で考えているのは画像圧縮であり，対象がリアルタイム処理を必要としない．であるから，遅延器 z^{-1} の左右においてもラウンディング操作を共有できる．

このようにして削減したラウンディングの総数を表 1 に示す．ここでは，Givens 分解に以下のリフティング因数分解 [6] を適用した PUFB と比較した．

$$
\begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} = \begin{bmatrix} 1 & \alpha \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ \beta & 1 \end{bmatrix} \begin{bmatrix} 1 & \alpha \\ 0 & 1 \end{bmatrix} \quad (24)
$$

ただし，$\alpha = (\cos\theta - 1)/\sin\theta$，$\beta = \sin\theta$ である．

表 1 リフティング因数分解した PUFB のラウンディング総数削減 (a) 削減前，(b) Givens 分解を用いた場合，(c) Householder 分解を用いた場合

Table 1 The number of reduced rounding operators in PUFBs using lifting factorization. (a) Before reducing, (b) Givens factorization, (c) Householder factorization.

	(a)	(b)		(c)	
4×8 PUFB	30	23	(-7)	18	(-12)
4×12 PUFB	42	31	(-11)	24	(-18)
4×16 PUFB	54	39	(-15)	30	(-24)
8×16 PUFB	132	95	(-37)	62	(-70)
8×24 PUFB	180	127	(-53)	82	(-98)
8×32 PUFB	229	159	(-70)	102	(-127)

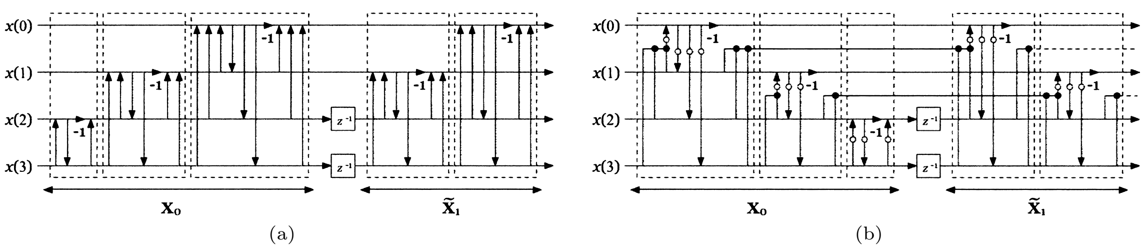


図 6 LBPUFB のラティス構造（$M = 4$）．(a) 基本構造，(b) ラウンディング総数削減．(b) において黒丸は加算操作を，白丸はラウンディング操作を表す

Fig. 6 Lattice structure of LBPUFB ($M = 4$). (a) Basic structure. (b) Reducing the rounding operators; a black and white dot denote an addition and a rounding operator, respectively.

この表を見ても分かるとおり，Householder 分解を用いた場合の方が，Givens 分解を用いた場合よりもラウンディング総数は大幅に減少し，ロスレス画像符号化に適しているといえる．

4. 結　　果

本章では実際に LBPUFB を設計し，ロッシー及びロスレス画像符号化に適用し，従来法と比較することでその優位性を示す．

4.1 設　　計

本論文では，4×8，4×12，4×16，8×16，8×24，8×32 LBPUFB を設計した．

まず，符号化利得 C_{CG} [3] のみで最適化した場合の符号化利得比較を表 2 に示す．この表より，従来法と同等の符号化利得が得られており，本提案法で除去したパラメータが冗長であったことが分かる．

次に画像符号化に適した PUFB を設計するために，符号化利得 C_{CG}，阻止域減衰量 C_{STOP}，DC（直流）漏れ C_{DC}，パラメータの絶対値の総計 C_{abs} を評価関数とした．前者三つは既知の評価関数 [3] である．また，C_{abs} はリフティングステップごとのパラメータの絶対値の和であり，リフティングステップにおけるラウンディング誤差を低減させるために導入した．ここで重み w_i を用いて，各評価関数の線形結合，

$$C = -w_1C_{CG} + w_2C_{STOP} + w_3C_{DC} + w_4C_{abs} \tag{25}$$

が最小になるように最適化をした．なお，重みは $\{w_1, w_2, w_3, w_4\} = \{1, 0.1, 0.1, 3\}$ を基準に経験的に決定した．設計した 6 種類の LBPUFB の周波数特性を図 7 に示す．

表 2　符号化利得比較
Table 2　Comparison of coding gain.

	PUFB [8]	PUFB [11]	LBPUFB
4×8	8.19	8.19	8.19
4×12	8.40	8.40	8.40
4×16	8.48	8.48	8.48
8×16	9.36	9.36	9.36
8×24	9.50	9.50	9.50
8×32	9.55	9.55	9.55

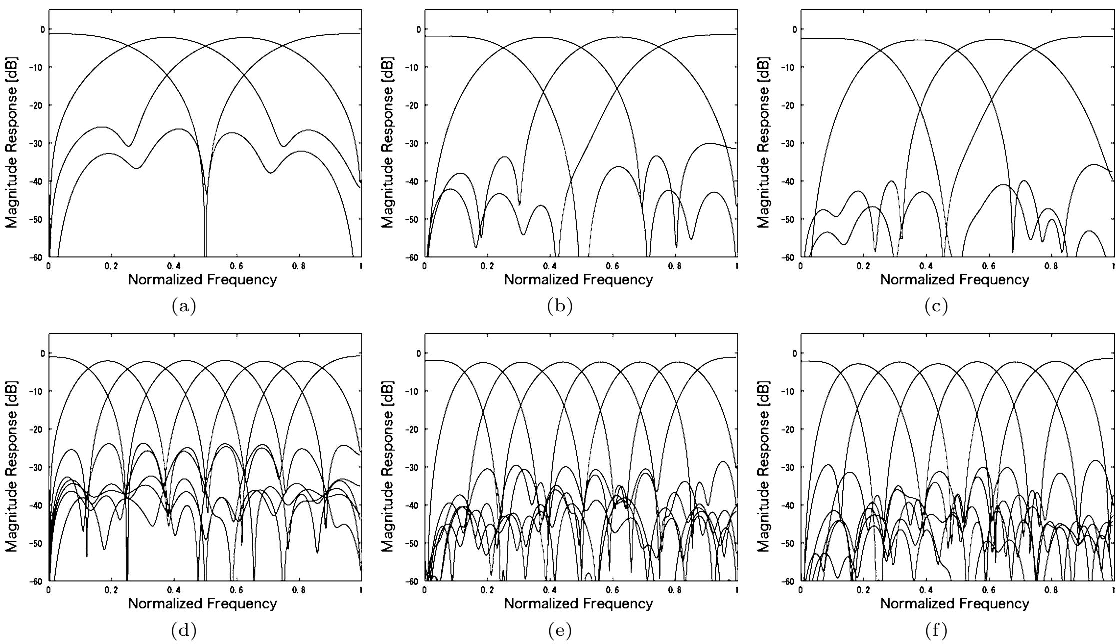


図 7　LBPUFB の周波数特性．(a) 4×8，(b) 4×12，(c) 4×16，(d) 8×16，(e) 8×24，(f) 8×32 LBPUFB
Fig. 7　Frequency responses of LBPUFBs. (a) 4×8, (b) 4×12, (c) 4×16, (d) 8×16, (e) 8×24, (f) 8×32 LBPUFB.

表 3　ロスレス画像符号化比較（エントロピー [bpp]）
Table 3　Comparison with lossless image coding (Entropy [bpp]).

テスト画像 (512×512)	5/3-tap DWT	4 × 8 LBPUFB	4 × 12 LBPUFB	4 × 16 LBPUFB	8 × 16 LBPUFB	8 × 24 LBPUFB	8 × 32 LBPUFB
Barbara	4.87	4.88	**4.79**	**4.79**	4.88	4.82	4.83
Boat	5.10	5.14	5.10	**5.09**	5.16	5.13	5.15
Elaine	5.11	5.17	5.12	5.11	5.12	**5.06**	5.07
Finger	5.84	5.82	5.74	5.72	5.70	**5.68**	**5.68**
Finger2	5.60	5.63	5.50	5.47	5.48	**5.43**	**5.43**
Grass	6.06	6.11	6.07	6.06	6.07	**6.05**	6.06

表 4　ロスレス画像符号化に対してラウンディング総数削減が与える影響（エントロピー [bpp]）
Table 4　The effect of reducing rounding operators in lossless image coding (Entropy [bpp]).

テスト画像	8 × 24 LBPUFB	
(512 × 512)	基本構造	ラウンディング削減
Barbara	4.91	**4.82**
Boat	5.19	**5.13**
Elaine	5.11	**5.06**
Finger	5.71	**5.68**
Finger2	5.48	**5.43**
Grass	6.06	**6.05**

図 8　Barbara（ビットレート：0.25 [bpp]）：（左）9/7-tap DWT,（右）8 × 32 LBPUFB
Fig. 8　Barbara (Bit rate: 0.25 [bpp]): (left) 9/7-tap DWT, (right) 8 × 32 LBPUFB.

4. 2　ロスレス画像符号化

本節では，リフティングステップごとにラウンディング操作を行うことで，ロスレス画像符号化に応用する．画像端での拡張法は周期拡張法，符号化には EZW-IP [7] を用い，その際のエントロピー [bpp]（= 総ビット数 [bit]/総ピクセル数 [pixel]）で圧縮率を評価する．また，5/3-tap 離散ウェーブレット変換（DWT）[9] と符号化性能を比較し，表 3 にその結果を示す．また，図 6 で示したラウンディング総数削減の効果を検証するために，8×24 LBPUFB の係数はそのままでラウンディング総数を削減したものとしていないものについて画像符号化後のエントロピーを比較したものを表 4 に示す．なお，表において太字は最良の数値であり，以下の表も同様とする．

以上より，ロスレス画像符号化において，5/3-tap DWT と比べ，同等若しくはより低いエントロピーを得られたことが分かる．また，ラウンディングをまとめることによるラウンディング誤差の軽減が圧縮率の向上の一つの要因であることが分かる．

4. 3　ロッシー画像符号化

本節では，ロスレス画像符号化に対して，各リフティングステップごとのラウンディング操作を外すことにより，ロスレス画像符号化と同じパラメータを用いて，ロッシー画像符号化に応用した例を示す．ロッシー画像符号化では，一般的な評価関数である $\mathrm{PSNR} = 10\log_{10}\frac{255^2}{\mathrm{MSE}}$ [dB] を用いて性能を評価する．ただし，MSE は誤差エネルギーである．

ロスレス画像符号化の際と同様，画像端拡張法は周期拡張法，符号化には EZW-IP を用いた．また，比較対象として 9/7-tap DWT [9]，LOT [1]，GenLOT [5] を用いた．ビットレートが 1.0 [bpp]，0.5 [bpp]，0.25 [bpp] の場合について，それぞれ表 5 に示す．更に，ビットレートが 0.25 [bpp] のときの画像（Barbara）の一部を拡大したものを図 8 に示す．特に画像中のしま模様に注意されたい．9/7-tap DWT では，このような空間周波数が高い部分は残らずにぼけてしまうが，本論文における LBPUFB ではその形状を残したまま符号化できることが分かる．これは DWT が 2 分割フィルタを用いて信号の低周波数部分を再帰的に分割するトリー構造をとっているのに対して，LBPUFB は分割数が多いため高周波数帯域を DWT と比較し多く取り込めることによる．

以上より，9/7-tap DWT，LOT，GenLOT と比べ，同等若しくはより高い PSNR を得ることができたことが分かる．これによりロスレスのみならずロッシー画像符号化でも，従来法に比べより良い圧縮率が得られる．

表 5　ロッシー画像符号化比較（PSNR [dB]）
Table 5　Comparison with lossy image coding (PSNR [dB]).

テスト画像 (512×512)	9/7-tap DWT	8 × 16 LOT	8 × 24 GenLOT	4 × 8 LBPUFB	4 × 12 LBPUFB	4 × 16 LBPUFB	8 × 16 LBPUFB	8 × 24 LBPUFB	8 × 32 LBPUFB
ビットレート：1.0 [bpp]									
Barbara	34.91	35.92	36.26	35.15	35.71	35.91	35.84	**36.33**	**36.33**
Boat	34.63	34.68	34.85	34.57	34.79	**34.89**	34.65	34.84	34.82
Elaine	34.89	35.16	35.45	34.59	34.94	35.01	35.10	**35.52**	35.50
Finger	29.04	30.04	30.12	29.30	29.74	29.83	30.13	30.36	**30.60**
Finger2	31.26	32.23	32.45	29.70	30.83	31.51	32.09	32.52	**32.56**
Grass	28.68	29.18	29.22	28.85	29.04	29.11	29.14	**29.23**	**29.23**
ビットレート：0.5 [bpp]									
Barbara	30.47	31.68	31.98	30.66	31.21	31.45	31.56	32.03	**32.07**
Boat	31.44	31.33	**31.48**	31.15	31.34	31.45	31.27	31.44	31.44
Elaine	33.05	33.10	33.38	32.71	33.02	33.08	33.04	**33.43**	33.41
Finger	25.96	26.46	26.52	25.68	26.08	26.22	26.50	26.60	**26.68**
Finger2	27.55	27.45	27.62	25.68	26.58	26.91	27.15	27.72	**27.90**
Grass	26.11	26.41	26.45	26.11	26.26	26.34	26.37	**26.47**	**26.47**
ビットレート：0.25 [bpp]									
Barbara	27.23	28.09	28.31	27.10	27.62	27.85	27.97	28.40	**28.42**
Boat	28.47	28.36	**28.50**	28.09	28.32	28.39	28.32	28.44	28.43
Elaine	**31.57**	31.18	31.45	30.98	31.32	31.39	31.11	31.40	31.37
Finger	23.50	23.65	23.70	22.87	23.19	23.32	23.65	**23.84**	23.75
Finger2	24.17	24.18	24.25	22.51	23.32	23.67	23.84	**24.65**	24.57
Grass	24.36	24.50	24.53	24.17	24.34	24.41	24.48	**24.58**	24.57

5. む す び

本論文では, [11] において提案された PUFB を基盤に，Householder 分解を用いてパラメータの冗長性を除去した新しいリフティング構造に基づく PUFB を提案した．この構造は, [10] において提案された PUFB や，Givens 分解を用いた場合よりもラウンディング操作の数を大幅に削減することができるため，ロスレス画像符号化に適した構造であるといえる．最後にロッシー及びロスレス画像符号化に応用し，PSNR とエントロピーにおいて従来法と比較し，それらと同等の性能，若しくはそれ以上の成果を実現し，その優位性を示した．またロッシー・ロスレス統合型の画像符号化においてラウンディングによる効果と圧縮率に関する相関は重要であり，更なる研究が望まれる．

文　　献

[1] E.S. Malvar and D.H. Staelin, "The LOT: Transform coding without blocking effects," IEEE Trans. Acoust. Speech Signal Process., vol.37, no.4, pp.553–559, April 1989.

[2] K.R. Rao and P. Yip, Discrete Cosine Transform: Algorithms, Advantages, Applications, Academic Press, 1990.

[3] P.P. Vaidyanathan, Multirate Systems and Filter Banks, Prentice Hall, Englewood Cliffs, NJ, 1992.

[4] A.K. Soman, P.P. Vaidyanathan, and T.Q. Nguyen, "Linear phase paraunitary filter banks: Theory, factorizations and designs," IEEE Trans. Signal Process., vol.41, no.12, pp.3480–3496, Dec. 1993.

[5] R.L. De Queiroz, T.Q. Nguyen, and K.R. Rao, "The GenLOT: Generalized linear-phase lapped orthogonal transform," IEEE Trans. Signal Process., vol.44, no.3, pp.497–507, March 1996.

[6] Y.-J. Chen, S. Oraintara, and T. Nguyen, "Integer Discrete Cosine Transform (IntDCT)," Proc. 2nd Int. Conf. Inform. Commun. Signal Process., Invited paper, Singapore, Dec. 1999.

[7] Z. Liu and L.J. Karam, "An efficient embedded zerotree wavelet image codec based on intraband partitioning," Proc. Int. Conf. Image Process., vol.3, pp.162–165, Sept. 2000.

[8] X. Gao, T.Q. Nguyen, and G. Strang, "On factorization of M-channel paraunitary filterbanks," IEEE Trans. Signal Process., vol.49, no.7, pp.1433–1446, July 2001.

[9] A. Skodras, C. Christopoulis, and T. Ebrahimi, "The JPEG2000 still image compression standard," IEEE Signal Process. Mag., vol.18, no.5, pp.36–58, Sept. 2001.

[10] Y.-J. Chen and K.S. Amaratunga, "M-channel lifting factorization of perfect reconstruction filter banks and reversible M-band wavelet transforms," IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process., vol.50, no.12, pp.963–976, Dec. 2003.

[11] M. Ikehara and Y. Kobayashi, "A novel lattice structure of M-channel paraunitary filter banks," IEEE

Int. Symp. Circuits Syst., pp.4293–4296, 2005.

（平成 18 年 3 月 22 日受付，6 月 9 日再受付，
6 月 26 日最終原稿受付）

鈴木　大三　（正員）

平 16 慶大・理工・電子卒．平 18 同大大学院修士課程了．同年凸版印刷（株）入社．在学中，ディジタル信号処理に関する研究に従事．

田中　雄一　（学生員）

平 15 慶大・理工・電子卒．平 17 同大大学院修士課程了．現在，同大学院博士課程在学中．ディジタル信号処理に関する研究に従事．

池原　雅章　（正員）

昭 59 慶大・工・電気卒．平元同大大学院博士課程了．工博．平元長崎大・工・講師．平 10 慶大・理工・助教授．平 16 同大・理工・教授．回路理論，ディジタル信号処理に関する研究に従事．