

並列リフティング構造を用いた M 分割完全再構成フィルタバンク

鈴木 大三^{†a)} 池原 雅章[†]

M-Channel Parallel Lifting-Based Perfect Reconstruction Filter Banks

Taizo SUZUKI^{†a)} and Masaaki IKEHARA[†]

あらまし ロッシー・ロスレス統合画像符号化のために、リフティング構造をもつ様々なフィルタバンクが提案されている。しかし、ラウンディング数の増加によるフィルタ特性の劣化や、逆行列を含むリフティング係数による設計の複雑さなどの問題があり、どれも実用的であるとはいいがたい。そこで本論文では、 M 分割完全再構成フィルタバンク (PRFB) のより実用的なリフティング構造を提案する。PRFB のポリフェーズ行列に条件を課して、二つの PRFB のラティス構造における各ビルディングブロックを並列に結合し、これらを同時に考慮する並列システム (ブロック並列システム) を用いる。これにより、各ブロック間でリフティングし合う構造 (並列リフティング構造) を適用することができる。すると PRFB の各ビルディングブロックは、同じリフティング係数行列を用いた三つのブロックリフティング構造で表現できる。最後にロッシー・ロスレス統合画像符号化におけるシミュレーションを行い従来法と比較して、提案法の優位性を示す。

キーワード 完全再構成フィルタバンク (PRFB), ブロック並列システム, 並列リフティング構造, ロッシー・ロスレス統合画像符号化

1. ま え が き

JPEG [1] や MPEG [2] のような静止及び動画像、音響符号化といったデータ符号化には、ロスレス符号化とロッシー符号化の二つがある。ロスレス符号化は原信号を完全に復元できるが、ロッシー符号化は原信号を完全には復元できない。そのため、ロッシー符号化はロスレス符号化よりも高い圧縮率を実現することができ、広く用いられている。しかし、医療や美術などの分野では、情報損失の少ないより高品質なデータが必要とされ、近年、ロスレス符号化が注目されている。一方、携帯電話やインターネットコンテンツなどの品質よりも圧縮率が重視される分野では、ロッシー符号化がよく用いられる。

本論文では、データ符号化の中でも特に画像符号化に焦点を当てることとする。画像符号化では、離散コサイン変換 (Discrete Cosine Transform : DCT) [3] や離散ウェーブレット変換 (Discrete Wavelet Transform : DWT) [4] を含むフィルタバンクが広く用い

られており、その有用性が証明されている。一般的に、フィルタバンクはロッシー画像符号化に用いられ、JPEG-LS に採用されている LOCO-I [5] のような予測符号化がロスレス画像符号化に用いられる。しかし近年、ブロードバンドの普及やハードウェアの発達によって、膨大なデータがあふれ返り、圧縮率の向上がより重要となる一方で、高品質データのニーズも増えてきており、JPEG2000 [4] のようなロッシー・ロスレス統合画像符号化が注目されている。予測符号化を用いるとロッシー・ロスレス統合画像符号化の実現は難しく、リフティング構造 [6] を用いた完全再構成フィルタバンク (Lifting-Based Perfect Reconstruction Filter Bank : LBPRFB) が重要な解決法となる [7] ~ [10]。この構造を用いてビットプレーン符号化を行うことでロスレス画像符号化を実現でき、符号化されたビットストリームを復号側で設定したビットレートで打ち切ることで、ロッシー画像符号化にも対応できる。しかし [7]、[8] 及び [9] の構造はラウンディング操作をまとめるには適さないために、ロスレス画像符号化に最適なフィルタとはいえない。ブロックリフティング分解を用いた [10] はラウンディング操作をまとめるのに効率的なフィルタであるが、フリーパラメータが多い上に、リフティング係数に逆行列を含むため、係数

[†] 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

のダイナミックレンジが広がる可能性があり、実用的ではない。

本論文では、以上のような背景の下、ロッキー・ロスレス統合画像符号化にとって、より実用的なLBPRFBを提案する。まず、PRFBのポリフェーズ行列に条件を課して、二つのPRFBのラティス構造における各ビルディングブロックを並列に結合し、これらを同時に考慮する並列システム（ブロック並列システム）を用いる。これにより、各ブロック間でリフティングし合う構造（並列リフティング構造）を適用することができる。このようにして得られる提案フィルタバンクをParallel Lifting-Based PRFB (PLPRFB)と呼ぶ。結果的に、PLPRFBの各ビルディングブロックは、同じリフティング係数行列を用いた三つのブロックリフティング構造で表現できる。また、提案法はリフティング分解の制約にもかかわらず、ロッキー・ロスレス統合画像符号化において、良好な性能を実現できることを示す。

表記： \mathbf{I} , M^T , $M^{[M]}$ はそれぞれ単位行列、行列 M の転置、 $M \times M$ 行列 M である。

2. 基礎理論

2.1 完全再構成フィルタバンク (Perfect Reconstruction Filter Bank: PRFB)

分割数 M のフィルタバンクの構造を図 1 に示す。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 [11] (図 2)。

$$\begin{aligned} \begin{bmatrix} H_0(z) \\ H_1(z) \\ \vdots \\ H_{M-1}(z) \end{bmatrix}^T &= \mathbf{E}(z^M) \mathbf{e}(z)^T \\ \begin{bmatrix} F_0(z) \\ F_1(z) \\ \vdots \\ F_{M-1}(z) \end{bmatrix} &= \mathbf{e}(z) \mathbf{R}(z^M) \end{aligned} \quad (1)$$

ただし、 $\mathbf{e}(z) = [1 \ z^{-1} \ \dots \ z^{-(M-1)}]$ である。

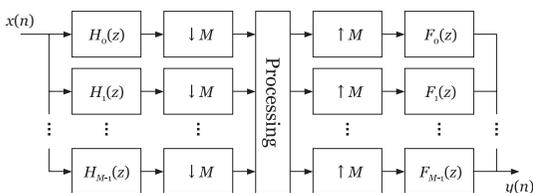


図 1 M 分割フィルタバンク
Fig. 1 An M -channel filter bank.

式 (1) において、PRFB は、完全再構成条件 $\mathbf{R}(z)\mathbf{E}(z) = \mathbf{I}$ 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 PRFB を考える。 $L = K - 1$ とすると、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 [11]。

$$\mathbf{E}(z) = \mathbf{W}_L \mathbf{\Lambda}_L(z) \cdots \mathbf{W}_1 \mathbf{\Lambda}_1(z) \mathbf{W}_0 \quad (2)$$

ここでビルディングブロック \mathbf{W}_k ($0 \leq k \leq L$) は $\mathbf{W}_k^{-1} \mathbf{W}_k = \mathbf{I}$ を満たす $M \times M$ の任意の正則行列であり、

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

である。 γ_k は $1 \leq \gamma_k \leq M$ の任意の整数をとり得るが、ここでは M を偶数とし、最も効率的な実現が可能である $\gamma_k = M/2$ とする。以下簡単のため、 $\mathbf{\Lambda}(z) \triangleq \mathbf{\Lambda}_k(z)$ とする。このような PRFB のラティス構造は、図 5 の上のブロックのように表される。また、 \mathbf{W}_k が直交行列、つまり $\mathbf{W}_k^{-1} = \mathbf{W}_k^T$ であるとき、この PRFB をパラユニタリフィルタバンク (Paraunitary Filter Bank: PUFB) と呼ぶ。そして、それ以外の PRFB を双直交フィルタバンク (Biorthogonal Filter Bank: BOFB) と呼ぶ。

2.2 リフティング構造

本節では、フィルタバンクをロスレス画像符号化に適用するために必要な、リフティング構造 [6] について述べる。まず、図 3(a) にリフティングの基本構造を示す。分割側入力信号を x_i, x_j , 分割側出力信号かつ合成側入力信号を y_i, y_j , 合成側出力信号を z_i, z_j , リフティング係数を T とすると、

$$\begin{aligned} y_i &= x_i + R[Tx_j] \\ y_j &= x_j \\ z_i &= y_i - R[Ty_j] = x_i \\ z_j &= y_j = x_j \end{aligned}$$

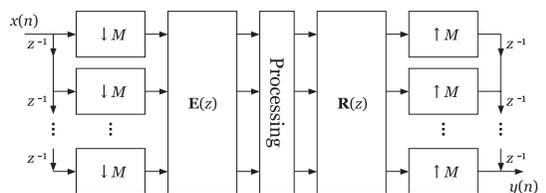


図 2 フィルタバンクのポリフェーズ構造
Fig. 2 A polyphase structure of filter bank.

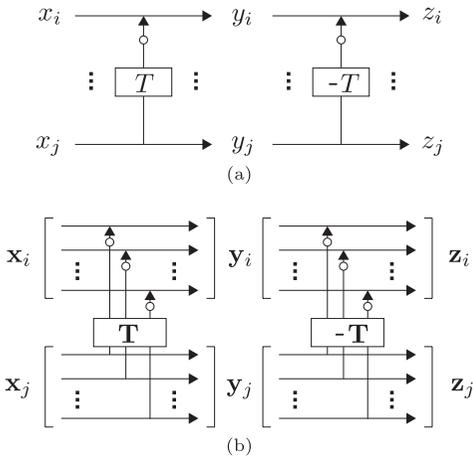


図 3 リフティング構造：(a) 基本構造，(b) ブロックリフティング構造（白丸：ラウンディング操作）

Fig. 3 Lifting structure: (a) basic structure, (b) block lifting structure (White circles: rounding operators).

と表すことができる．ただし， $R[\cdot]$ は最も近い整数へのラウンディング操作を表す．このとき， x_i, x_j が整数である限り， y_i, y_j も常に整数になるため，符号化の際の量子化誤差を生じず，完全再構成 $z_i = x_i, z_j = x_j$ が構造的に成り立つ．つまりフィルタバンクをロスレス画像符号化に応用するためには，その構造自体にこのリフティング構造を用いればよい，ということになる．

2.3 ブロックリフティング構造

ロスレス画像符号化のためには，ラウンディング操作をリフティングステップごとに行う必要がある．しかし，ラウンディング誤差がリフティングステップごとに同時に生成されることになる．システム全体を通してのラウンディング誤差が大きいと圧縮率が低下し，したがってラウンディング数を削減することが圧縮率向上につながる．そこで，ブロックリフティング構造という，ラウンディング数を大幅に削減し，効率的にラウンディング操作を行うことのできるリフティング構造が提案されている [10]．図 3 (b) にブロックリフティング構造を示す．分割側入力信号ベクトルを x_i, x_j ，分割側入力かつ合成側出力信号ベクトルを y_i, y_j ，合成側出力信号ベクトルを z_i, z_j ， $N \times N$ リフティング係数行列を T とすると，

$$y_i = x_i + R[Tx_j]$$

$$y_j = x_j$$

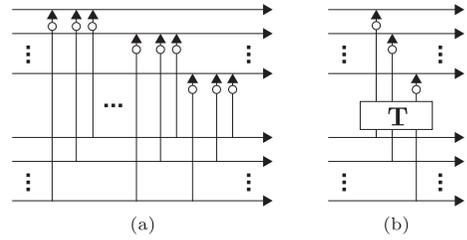


図 4 ラウンディング数の削減：(a) まとめ操作前，(b) まとめ操作後（白丸：ラウンディング操作）

Fig. 4 Reducing the number of rounding operators: (a) before merged, (b) after merged (White circles: rounding operators).

$$z_i = y_i - R[Ty_j] = x_i$$

$$z_j = y_j = x_j$$

と表すことができる．ただし， $R[\cdot]$ は最も近い整数へのラウンディング操作を表す．リフティングの基本構造と同様に， x_i, x_j が整数ベクトルである限り， y_i, y_j も常に整数ベクトルになるため，符号化の際の量子化誤差を生じず，完全再構成 $z_i = x_i, z_j = x_j$ が構造的に成り立つ．また，重要な点として，リフティングステップの数は $N^2/4$ 個存在するが，それを $N/2$ 個のラウンディング操作にまとめる，つまりラウンディング数を大幅に削減できる点である．図 4 にラウンディング操作のまとめ操作前とまとめ操作後の図を示す．

3. 並列リフティング構造を用いた完全再構成フィルタバンク (PLPRFB: Parallel Lifting-Based Perfect Reconstruction Filter Bank)

本章では，式 (2) における PRFB のラティス構造を用いた各ビルディングブロックにおける並列システム（ブロック並列システム）に注目し，それを利用したより効率的なリフティング構造を提案する．

3.1 ブロック並列システム

ブロック並列システムには，並列処理する場合と，時系列処理する場合とがあり，それぞれの処理について述べる．

3.1.1 並列処理

まず，二つの入力信号 $x(z), \tilde{x}(z)$ を二つのフィルタバンクで並列処理する場合を考える．それぞれの出力信号を $y(z), \hat{y}(z)$ とすると，PRFB のラティス構造は，

$$\begin{bmatrix} \mathbf{y}(z) \\ \tilde{\mathbf{y}}(z) \end{bmatrix} = \begin{bmatrix} \mathbf{E}(z) & \mathbf{0} \\ \mathbf{0} & \mathbf{E}(z) \end{bmatrix} \begin{bmatrix} \mathbf{x}(z) \\ \tilde{\mathbf{x}}(z) \end{bmatrix}$$

となり、

$$\mathbf{X}_k = \begin{bmatrix} \mathbf{W}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{W}_k \end{bmatrix}, \Gamma(z) = \begin{bmatrix} \Lambda(z) & \mathbf{0} \\ \mathbf{0} & \Lambda(z) \end{bmatrix} \quad (3)$$

としたとき、

$$\begin{bmatrix} \mathbf{E}(z) & \mathbf{0} \\ \mathbf{0} & \mathbf{E}(z) \end{bmatrix} = \mathbf{X}_L \Gamma(z) \cdots \mathbf{X}_1 \Gamma(z) \mathbf{X}_0$$

である。結局、このような並列処理は図5のように表すことができる。

実際の画像符号化において、並列処理する場合は、入力信号 $\mathbf{x}(z)$, $\tilde{\mathbf{x}}(z)$ をそれぞれ別の画像として処理することもできるし、1枚の画像を扱う場合、偶数番目の行(列)と奇数番目の行(列)に分割するように、全行(列)を半分ずつに分割した二つの画像として行(列)を処理することもできる。もちろん、全列(行)を半分ずつに分割した二つの画像として行(列)を処理することもできるが、この場合は低ビットレートで圧縮した際に継ぎ目が目立つ可能性がある。

式(3)のように二つのブロックを同時に考慮するようなシステムをブロック並列システムと呼ぶ。この \mathbf{X}_k がリフティング構造で表現できれば、PRFB全体がリフティング構造と遅延器ブロック $\Lambda(z)$ のみで表現できるので、ロッシー・ロスレス統合符号化に応用できる。

3.1.2 時系列処理

次に、一つの入力信号を時系列処理する場合を考える。式(2)のポリフェーズ表現を時間域に変換し、実際の時系列信号の処理は図6で表現できる[12]。入力信号 \mathbf{x} は画像の行または列の一次元信号であり、 \mathbf{y} はその変換係数である。これは信号をブロックに分割し、ブロック単位で処理することを意味し、図中のクロスラインは $\Lambda(z)$ の遅延を表している。ここで破線に囲まれた二つのビルディングブロック \mathbf{W}_k に注目すると、

$$\mathbf{X}_k = \begin{bmatrix} \mathbf{W}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{W}_k \end{bmatrix} \quad (4)$$

を得る。

このとき、式(3)における \mathbf{X}_k と式(4)における \mathbf{X}_k が、完全に同じブロック並列システムになっているこ

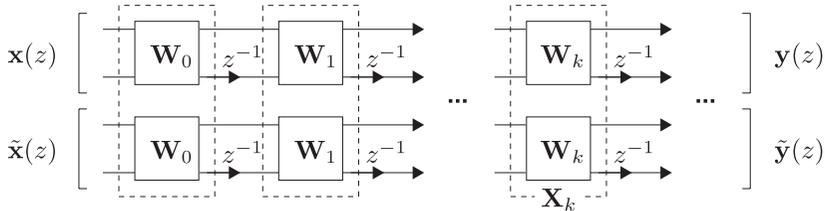


図5 PRFBのラティス構造(並列処理)
Fig.5 Lattice structure of PRFB (parallel processing).

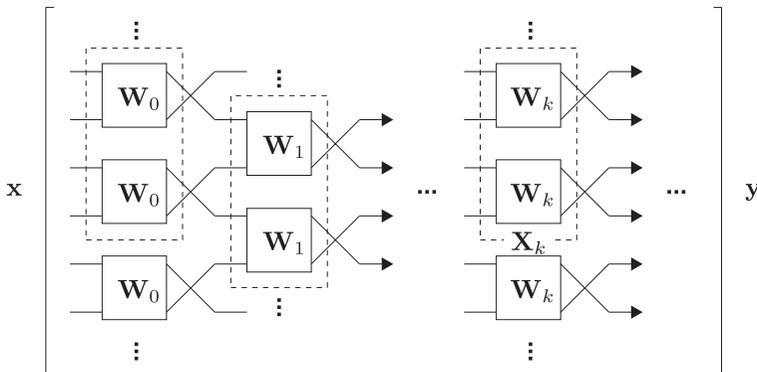


図6 PRFBのラティス構造(時系列処理)
Fig.6 Lattice structure of PRFB (time-series processing).

とが分かる．そして，3.1.1 同様， \mathbf{X}_k がリフティング構造で表現できれば，PRFB 全体がリフティング構造と遅延器ブロック $\Lambda(z)$ のみで表現できるので，ロッキー・ロスレス統合符号化に適用できるようになる．要するにロッキー・ロスレス統合符号化実現問題は，並列処理においても，時系列処理においても，最終的に \mathbf{X}_k のリフティング分解に帰着することに注意されたい．

3.2 並列リフティング構造

前節で述べたとおり，式 (3) 及び式 (4) におけるブロック並列システム \mathbf{X}_k がリフティング分解できるとき，ロッキー・ロスレス統合画像符号化に適用できるようになる．よって，以下， \mathbf{X}_k のリフティング分解について述べる．まず，

$$\begin{bmatrix} \mathbf{0} & \mathbf{I} \\ \mathbf{I} & \mathbf{0} \end{bmatrix} \mathbf{X}_k = \begin{bmatrix} \mathbf{0} & \mathbf{W}_k \\ \mathbf{W}_k & \mathbf{0} \end{bmatrix} \quad (5)$$

とすると，式 (5) の右側から，リフティング係数に \mathbf{W}_k^{-1} をもつブロックリフティング行列を乗算すると，行列の左上の要素が \mathbf{I} である以下の行列を得る．

$$\begin{bmatrix} \mathbf{0} & \mathbf{W}_k \\ \mathbf{W}_k & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{W}_k^{-1} & \mathbf{I} \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{W}_k \\ \mathbf{W}_k & \mathbf{0} \end{bmatrix} \quad (6)$$

更に式 (6) の右側から，リフティング係数に $-\mathbf{W}_k$ をもつブロックリフティング行列を乗算し，行列の右上の要素が $\mathbf{0}$ である以下の行列を得る．

$$\begin{bmatrix} \mathbf{I} & \mathbf{W}_k \\ \mathbf{W}_k & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{I} & -\mathbf{W}_k \\ \mathbf{0} & \mathbf{I} \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{W}_k & -\mathbf{W}_k^2 \end{bmatrix} \quad (7)$$

最後に式 (7) の右側から，式 (6) と同じブロックリフティング行列を乗算すると，行列の左下の要素が $\mathbf{0}$ である以下の行列を得る．

$$\begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{W}_k & -\mathbf{W}_k^2 \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{W}_k^{-1} & \mathbf{I} \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & -\mathbf{W}_k^2 \end{bmatrix} \quad (8)$$

つまり，式 (5)~(8) を用いて，

$$\begin{aligned} \mathbf{X}_k &= \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ \mathbf{I} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & -\mathbf{W}_k^2 \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ -\mathbf{W}_k^{-1} & \mathbf{I} \end{bmatrix} \\ &\quad \times \begin{bmatrix} \mathbf{I} & \mathbf{W}_k \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ -\mathbf{W}_k^{-1} & \mathbf{I} \end{bmatrix} \quad (9) \end{aligned}$$

を得ることができる．しかし，このままでは 2 番目の行列の対角ブロックに \mathbf{W}_k^2 を含むため，式 (9) は完

全なりフティング分解ではない．そこで $\mathbf{W}_k^{-1} = \mathbf{W}_k$ という制約を設けることとする．PUFB の場合は， \mathbf{W}_k が $M \times M$ の任意の直交行列であり，かつ $\mathbf{W}_k^{-1} = \mathbf{W}_k^T = \mathbf{W}_k$ を満たす行列，つまり対称直交行列にならなければならない．対称直交行列は以下のように設計することができる [13]．

$$\mathbf{W}_k = \begin{bmatrix} \mathbf{U}_{k,0} & \mathbf{0} \\ \mathbf{0} & \mathbf{U}_{k,1} \end{bmatrix} \begin{bmatrix} \mathbf{C}_k & \mathbf{S}_k \\ \mathbf{S}_k & -\mathbf{C}_k \end{bmatrix} \begin{bmatrix} \mathbf{U}_{k,0}^T & \mathbf{0} \\ \mathbf{0} & \mathbf{U}_{k,1}^T \end{bmatrix} \quad (10)$$

このとき， $[\mathbf{C}_k]_{ii} = \cos \alpha_{ki}$ ， $[\mathbf{S}_k]_{ii} = \sin \alpha_{ki}$ であり， $\mathbf{U}_{k,0}$ ， $\mathbf{U}_{k,1}$ は $M/2 \times M/2$ の任意の直交行列である．BOFB の場合は，式 (10) を以下のように拡張して設計する．

$$\mathbf{W}_k = \begin{bmatrix} \mathbf{V}_{k,0} & \mathbf{0} \\ \mathbf{0} & \mathbf{V}_{k,1} \end{bmatrix} \begin{bmatrix} \mathbf{C}_k & \mathbf{S}_k \\ \mathbf{S}_k & -\mathbf{C}_k \end{bmatrix} \begin{bmatrix} \mathbf{V}_{k,0}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{V}_{k,1}^{-1} \end{bmatrix} \quad (11)$$

このとき， $[\mathbf{C}_k]_{ii} = \cos \alpha_{ki}$ ， $[\mathbf{S}_k]_{ii} = \sin \alpha_{ki}$ であり， $\mathbf{V}_{k,0}$ ， $\mathbf{V}_{k,1}$ は $M/2 \times M/2$ の任意の正則行列である．結果，以下のようなリフティング分解を得ることができる．

$$\mathbf{X}_k = \begin{bmatrix} \mathbf{0} & -\mathbf{I} \\ \mathbf{I} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ -\mathbf{W}_k & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{W}_k \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ -\mathbf{W}_k & \mathbf{I} \end{bmatrix} \quad (12)$$

以上により得られる並列リフティング構造を用いた PRFB を Parallel Lifting-Based PRFB (PLPRFB) と呼ぶ．リフティング分解前の PRFB の各ビルディングブロックにおけるブロック並列システムを図 7 (a)，それを並列リフティング分解したものを図 7 (b) に示す．本論文では，PLPRFB を更に分類し，式 (10) を用いるときを Parallel Lifting-Based PUFB (PLPUFB)，式 (11) を用いるときを Parallel Lifting-Based BOFB (PLBOFB) とし，従来法とこれらのフリーパラメータ数の比較を表 1 に示す．フリーパラメータ数は， $\mathbf{U}_{k,0}$ ， $\mathbf{U}_{k,1}$ がそれぞれ $M(M-2)/8$ 個， $\mathbf{V}_{k,0}$ ， $\mathbf{V}_{k,1}$ がそれぞれ $M^2/4$ 個， \mathbf{C}_k ， \mathbf{S}_k がそれぞれ $M/2$ 個となるので，結果として，PLPUFB のときは $KM^2/4$ 個，PLBOFB のときは $KM(M+1)/2$ 個となる．

また，式 (12) から分かるとおり，各ビルディングブロックは，逆行列を含まない同じリフティング係数行列をもつ三つのブロックリフティング構造から構成される．

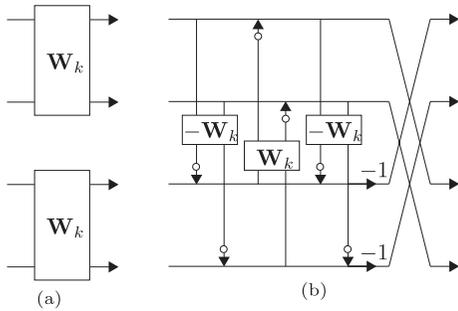


図7 PRFBのブロック並列システム：(a) 基本構造，(b) 並列リフティング構造（白丸：ラウンディング操作）

Fig. 7 Block parallel system of PRFB: (a) basic structure, (b) parallel lifting structure (White circles: rounding operators).

表1 フリーパラメータ数比較

Table 1 Comparison of the number of free parameters.

	LBPUBF [9]	LBBOFB [10]	PLPUBF	PLBOFB
8 × 16	44	96	32	72
8 × 24	60	144	48	108

表2 削減後のラウンディング数比較

Table 2 Comparison of the number of reduced rounding operators.

	LBPUBF [9]	LBBOFB [10]	PLPUBF	PLBOFB
8 × 16	62 (132)	20 (112)	24 (192)	24 (192)
8 × 24	82 (180)	28 (176)	36 (288)	36 (288)

3.3 ラウンディング数削減

ラウンディング数の削減による圧縮率向上は、2.3で述べた。PLPRFBの構造は、2.3で述べたブロックリフティング構造と同じように、リフティング係数が行列（ブロック）として表現できるため、多くのラウンディング操作をまとめて行うことができる。削減後のラウンディング数比較を表2に示す。このとき、表2の（）内は削減前のラウンディング数を表す。PLPRFBは、通常システムを二つ同時に考慮したシステムであるので、ラウンディング数をシステム一つ分、つまり半分の値で表示した。表2から分かるとおり、削減前のラウンディング数、削減後のラウンディング数ともに[10]が一番少ないが、PLPRFBも[10]と同程度のラウンディング数まで削減できることが分かる。具体的には、ラウンディング操作をまとめる前は、リフティング係数行列ごとに M^2 個のラウンディング操作が必要であるが、ブロックリフティング構造を用いると、リフティング係数行列ごとに M 個のラウンディング操作にまとめることができ、一つのシス

テム当りの総ラウンディング数は、 $3KM^2/2$ 個から $3KM/2$ 個に削減できる。

4. シミュレーション結果

本章では実際にPLPRFBを設計し、ロッキー・ロスレス統合画像符号化に適用し、従来法と比較することで優位性を示す。

4.1 設計

本論文では、 8×16 （分割数8，フィルタ長16）， 8×24 （分割数8，フィルタ長24）PLPUBF及びPLBOFBの計4種類のPLPRFBを設計した。

画像符号化に適したPLPRFB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} を評価関数とした[11]。ここで各評価関数の線形結合，

$$C = -w_1 C_{CG} + w_2 C_{STOP} + w_3 C_{DC}$$

が最小になるように、MatlabのOptimization Toolbox関数fminunc.mを用いてパラメータの最適化を行った。なお、 w_i は重みであり、 $\{w_1, w_2, w_3\} = \{1, 0.1, 0.1\}$ を基準に、経験的に決定した。設計したPLPRFBの周波数特性を図8に示す。

4.2 ロッキー・ロスレス統合画像符号化

本節では、前節で設計したPLPRFBを、3.1.2及び3.1.1で述べた並列及び時系列処理の2通りを用いて、ロッキー・ロスレス統合画像符号化に応用する。並列処理については別の画像を扱う場合と1枚の画像を扱う場合があるが、本論文では1枚の画像を扱う場合を考え、奇数番目の行（列）と偶数番目の行（列）に分割して行（列）の処理を行う。当然、列（行）を処理する場合は、信号を並べ換えて、同様の処理を行う。そして、ロッキー・ロスレス統合画像符号化は以下の2ステップで実現する。

(1) リフティングステップごとにラウンディング操作を行ってからビットプレーン符号化をすることで、ロスレス画像符号化を実現する。

(2) ステップ1で得たビットストリームを、復号側で設定したビットレートで打ち切ることで、ロッキー画像符号化に対応する。

また、画像端での拡張法は周期拡張法、ビットプレーン符号化方式にはEZW-IP[14]、テスト画像には‘Barbara’や‘Lena’などの代表的な10枚を用いた。ロスレス画像符号化では画質による評価ができないため、ビットレート[bpp]（＝総ビット数[bit]/総ピクセ

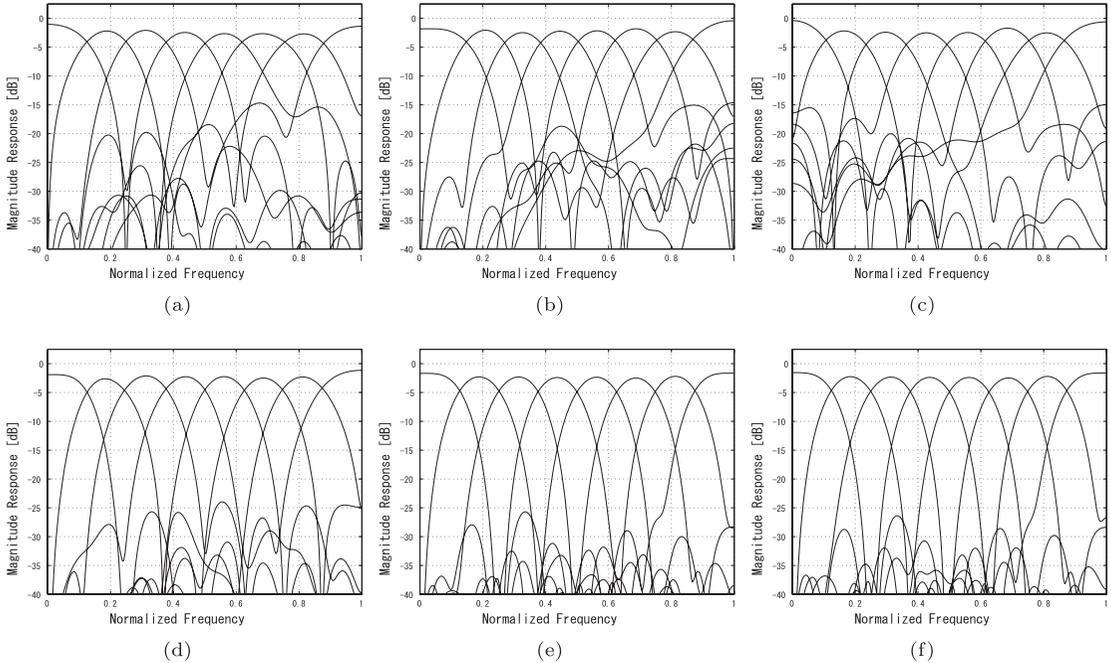


図 8 PLPRFB の周波数特性 : (a)~(c) 8×16 PLPUFB 及び PLBOFB (分割側, 合成側) (d)~(f) 8×24 PLPUFB 及び PLBOFB (分割側, 合成側)

Fig. 8 Frequency responses of PLPRFBs: (a)~(c) 8×16 PLPUFB and PLBOFB (analysis, synthesis), (d)~(f) 8×24 PLPUFB and PLBOFB (analysis, synthesis).

表 3 ロスレス画像符号化比較 (ビットレート [bpp])
Table 3 Comparison of lossless image coding (Bit rate [bpp]).

テスト 画像 (512×512)	従来法			提案法 (並列処理)		提案法 (時系列処理)	
	5/3-tap DWT [4]	8 × 24 LBPUBF [9]	8 × 24 LBBOFB [10]	8 × 24 PLPUFB	8 × 24 PLBOFB	8 × 24 PLPUFB	8 × 24 PLBOFB
Baboon	6.17	6.23	6.22	6.22	6.22	6.23	6.22
Barbara	4.87	4.82	4.77	4.77	4.76	4.77	4.76
Boat	5.10	5.13	5.12	5.10	5.10	5.10	5.10
Elaine	5.11	5.06	5.04	5.05	5.03	5.05	5.03
Finger1	5.84	5.68	5.65	5.66	5.66	5.66	5.66
Finger2	5.60	5.43	5.39	5.40	5.39	5.40	5.39
Goldhill	5.01	5.14	5.11	5.11	5.10	5.11	5.10
Grass	6.06	6.05	6.04	6.04	6.04	6.05	6.04
Lena	4.49	4.68	4.65	4.63	4.63	4.63	4.63
Pepper	4.85	4.95	4.94	4.92	4.93	4.92	4.93
平均	5.31	5.32	5.29	5.29	5.29	5.29	5.29

ル数 [pixel]) で圧縮率を評価する。一方, ロッシー画像符号化では, 一般的な評価関数である PSNR [dB] ($= 10 \log_{10}(255^2/\text{MSE})$) を用いて, 同ビットレート時の画質で評価する。ただし, MSE (Mean Square Error) は誤差エネルギーである。ロスレス画像符号化の比較対象には JPEG2000 [4] における 5/3-tap DWT, 同分割数及び同フィルタ長である [9], [10] を, ロッシー画像符号化の比較対象には同様に 9/7-tap

DWT, 同分割数及び同フィルタ長である [9], [10] を選んだ。その際, 周波数の分解能を統一するため, 提案法である PLPRFB と従来法である [9] 及び [10] については 2 レベルに, 5/3-及び 9/7-tap DWT については 6 レベルに, それぞれ低周波領域のみを階層的に変換した。画像端拡張法には, DWT には対称拡張法 [9], [10] には周期拡張法を適用した。更に, 変換による符号化性能を公平に比較するために, 符号化方

表 4 ロッシー画像符号化比較 (PSNR [dB])
Table 4 Comparison of lossy image coding (PSNR [dB]).

ビット レート [bpp]	テスト 画像 (512 × 512)	従来法			提案法 (並列処理)		提案法 (時系列処理)	
		9/7-tap DWT [4]	8 × 24 LBPUFB [9]	8 × 24 LBBOFB [10]	8 × 24 PLPUFB	8 × 24 PLBOFB	8 × 24 PLPUFB	8 × 24 PLBOFB
0.25	Baboon	22.82	22.77	22.77	22.73	22.77	22.72	22.77
	Barbara	27.25	28.38	28.40	28.32	28.36	28.33	28.36
	Boat	28.46	28.41	28.34	28.46	28.42	28.46	28.42
	Elaine	31.51	31.34	31.26	31.32	31.33	31.33	31.33
	Finger1	23.51	23.79	23.77	23.70	23.90	23.71	23.91
	Finger2	24.18	24.55	24.68	24.06	24.65	24.06	24.65
	Goldhill	29.67	29.14	29.11	29.12	29.12	29.12	29.12
	Grass	24.36	24.57	24.57	24.55	24.55	24.56	24.55
	Lena	32.53	32.23	32.09	32.27	32.26	32.26	32.26
	Pepper	31.94	31.67	31.50	31.66	31.62	31.66	31.63
	平均	27.62	27.68	27.65	27.62	27.70	27.62	27.70
0.50	Baboon	24.60	24.73	24.69	24.71	24.71	24.71	24.72
	Barbara	30.49	31.98	32.03	31.91	31.98	31.92	31.99
	Boat	31.39	31.38	31.34	31.41	31.38	31.42	31.39
	Elaine	32.94	33.29	33.24	33.28	33.34	33.28	33.33
	Finger1	25.99	26.59	26.62	26.58	26.62	26.58	26.62
	Finger2	27.53	27.62	27.73	27.35	27.74	27.32	27.73
	Goldhill	31.53	31.67	31.66	31.67	31.70	31.69	31.70
	Grass	26.09	26.46	26.47	26.49	26.47	26.49	26.47
	Lena	35.52	35.39	35.27	35.53	35.50	35.51	35.51
	Pepper	34.48	34.17	34.03	34.26	34.20	34.25	34.19
	平均	30.06	30.33	30.31	30.32	30.37	30.32	30.37
1.00	Baboon	27.61	27.71	27.71	27.71	27.72	27.70	27.72
	Barbara	34.88	36.07	36.18	36.13	36.20	36.14	36.21
	Boat	34.51	34.65	34.69	34.77	34.75	34.77	34.76
	Elaine	34.61	35.24	35.32	35.30	35.38	35.31	35.38
	Finger1	29.07	30.21	30.25	30.25	30.21	30.26	30.21
	Finger2	30.90	32.40	32.50	32.43	32.46	32.44	32.46
	Goldhill	34.51	34.69	34.76	34.79	34.77	34.80	34.77
	Grass	28.68	29.22	29.27	29.26	29.25	29.26	29.24
	Lena	38.65	38.54	38.69	38.86	38.84	38.85	38.86
	Pepper	36.09	35.91	35.96	36.08	36.03	36.08	36.03
	平均	32.95	33.46	33.53	33.56	33.56	33.56	33.56

式は提案法と同様の EZW-IP [14] に統一して行った。8 × 24 PLPRFB 及び従来法のシミュレーション結果を表 3, 表 4 に示す。8 × 24 PLPRFB の方が 8 × 16 PLPRFB よりもフィルタ特性が優れるため、8 × 16 PLPRFB に関しては表 3, 表 4 から割愛した。最後に、図 9 に、9/7-tap DWT 及び提案法によって圧縮された 'Barbara' の拡大画像を示す。

4.3 考 察

表 3, 表 4 より、ロッシー・ロスレス統合画像符号化において、画像やビットレートによる差はあるものの、JPEG2000 [4] に採用されている 5/3-tap DWT 及び 9/7-tap DWT と同等以上の性能を示していることが分かる。図 9 より、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 PLPRFB ではその形状を残したまま符号化できることが分かる。つまり、PLPRFB は高

周波成分を多く含む画像に、特に適しているといえる。一方、低周波成分を多く含む画像群においては、ロスレス画像符号化やロッシー画像符号化の 0.25 [bpp] では、DWT の方が良好な値を示している。しかし、JPEG2000 における DWT は、ロスレス画像符号化及びロッシー画像符号化でそれぞれ 5/3-tap DWT 及び 9/7-tap DWT という異なる変換を用いて異なるビット列を作成する必要がある。

並列処理と時系列処理の結果に大きな差異が見られないのは、どちらの処理においても使っているフィルタが同じであることと、極小のラウンディング誤差はあるものの、互いのブロック W_k 間で干渉することがほぼないためと考えられる。また、表 1 より、フリーパラメータ数が従来法 [9], [10] より少ないにもかかわらず、それらと同等以上の性能を示していることが分かる。リフティング分解のための制約 $W_k^{-1} = W_k$ が

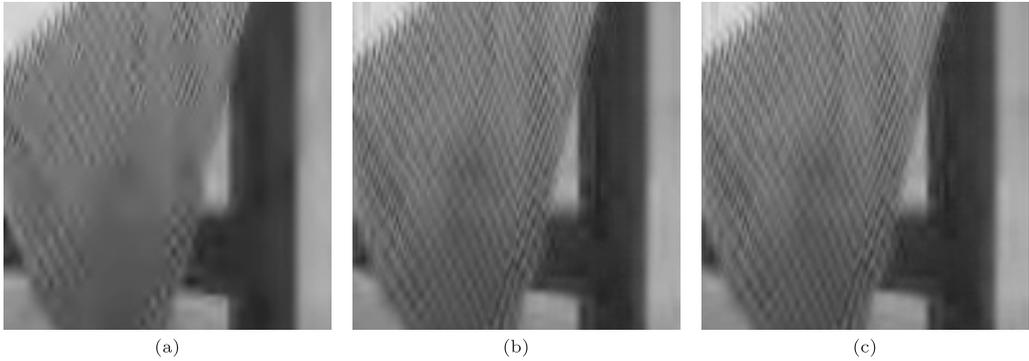


図9 'Barbara' の拡大画像 (ビットレート: 0.25 [bpp]): (a) 9/7-tap DWT, (b) 8×24 PLPUFB (並列処理), (c) 8×24 PLPUFB (時系列処理)

Fig. 9 The enlarged image of 'Barbara' (Bit rate: 0.25 [bpp]): (a) 9/7-tap DWT, (b) 8×24 PLPUFB (parallel processing), (c) 8×24 PLPUFB (time-series processing).

設計の自由度を下げているとも考えられるが, 3.3 で述べたとおり, それ以上に [10] と同様に大幅なラウンディング数削減 (表 2) と, リフティング係数の単純さによるフィルタ設計の簡易さが, ロッシー・ロスレス統合画像符号化にとってより効果的であったと考えられる.

5. むすび

本論文では, M 分割完全再構成フィルタバンク (PRFB) のラティス構造を用いた各ビルディングブロックにおける並列システム (ブロック並列システム) に注目し, それを利用したより実用的なリフティング構造 (並列リフティング構造) を提案した. 並列リフティング構造を用いた PRFB (Parallel Lifting-Based PRFB: PLPRFB) を二つ用いることで, 各ビルディングブロックは同じリフティング係数行列を用いた三つのブロックリフティング構造で表現できる. また, リフティング分解のための制約を課したものの, 設計した PLPRFB を用いたロッシー・ロスレス統合画像符号化において, 同等以上の性能を示した. 今後の課題として, より簡単な構造かつ優れたフィルタ特性をもった, 更に実用的なフィルタの設計の研究が望まれる.

謝辞 本研究の一部はグローバル COE プログラム「アクセス空間支援基盤技術の高度国際連携」により行われました.

文 献

[1] W. Pennebaker and J. Mitchell, JPEG, Still Image Data Compression Standard, Van Nostrand, NY,

1993.

[2] T. Sikora, "MPEG digital video-coding standards," IEEE Signal Process. Mag., vol.14, no.5, pp.82–100, Sept. 1997.

[3] K.R. Rao and P. Yip, Discrete Cosine Transform Algorithms, Academic Press, 1990.

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

[5] M.J. Weinberger, G. Seroussi, and G. Sapiro, "The LOCO-I lossless image compression algorithm: Principles and standardization into JPEG-LS," IEEE Trans. Image Process., vol.9, no.8, pp.1309–1324, Aug. 2000.

[6] W. Sweldens, "The lifting scheme: A custom-design construction of biorthogonal wavelets," Appl. Comput. Harmon. Anal., vol.3, no.2, pp.186–200, 1996.

[7] T.D. Tran, " M -channel linear phase perfect reconstruction filter bank with rational coefficients," IEEE Trans. Circuits Syst. I, vol.49, no.7, pp.914–927, July 2002.

[8] 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, vol.50, no.12, pp.963–976, Dec. 2003.

[9] T. Suzuki, Y. Tanaka, and M. Ikehara, "Lifting-based paraunitary filterbanks for lossy/lossless image coding," Proc. EUSIPCO'06, Florence, Italy, Sept. 2006.

[10] S. Iwamura, Y. Tanaka, and M. Ikehara, "An efficient lifting structure of biorthogonal filter banks for lossless image coding," Proc. ICIP'07, San Antonio, TX, Sept. 2007.

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

- [12] T.D. Tran, J. Liang, and C. Tu, "Lapped transform via time-domain pre- and post-filtering," *IEEE Trans. Signal Process.*, vol.6, no.6, pp.1557-1571, June 2003.
- [13] 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.
- [14] Z. Liu and L.J. Karam, "An efficient embedded zerotree wavelet image codec based on intraband partitioning," *Proc. ICIP'00, Vancouver, British Columbia, Canada, Sept. 2000.*

(平成 21 年 2 月 2 日受付, 5 月 14 日再受付)



鈴木 大三 (学生員)

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。現在,慶大大学院博士課程在学中。デジタル信号処理,特にフィルタバンクを用いた静止,動画像及び音響符号化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,デジタル信号処理に関する研究に従事。