

G-002

木構造フィルタバンクを用いた X 線 CT 再構成アルゴリズムの  
Cell Broadband Engine への実装  
An Implementation of the CT Reconstruction Algorithm  
using Tree-Structured Filter Bank on the Cell Broadband Engine

守本 弘明\*  
Hiroaki Morimoto

森川 良孝\*  
Yoshitaka Morikawa

## 1 はじめに

非破壊、非侵襲で物体の内部構造観察を可能にする X 線コンピュータ断層撮影装置 (Computerized Tomography: CT) の発達は、医療分野において大きな発展をもたらした。近年では、ヘリカルスキャン及び 2 次元検出器の開発により、3 次元データの高速収集が可能となり、3 次元 CT に関する研究も盛んに行われている [1]。しかし、一般的な CT 再構成アルゴリズムとして知られているフィルタ補正逆投影法 (Filtered Back Projection: FBP)[2], [3] は、X 線の照射から物体の内部構造画像を再構成するまでに莫大な計算量を必要とすることが問題視されている。

そこで FBP 法の高速化アルゴリズムとして、木構造フィルタバンク (Tree-Structured Filter Bank: TSFB) を用いた高速再構成法 [5],[6] が提案されている。TSFB 法とは、X 線照射により得られる Radon 変換と、各投影方向に Lowpass フィルタ処理後、ダウンサンプリングする分析処理を等価と見なし、CT 再構成をその逆操作、すなわち合成処理として行う手法である。合成処理では、ダウンサンプリングにより生じるイメージスペクトルをフィルタバンクの合成フィルタに相当する狭帯域平行四辺形フィルタにより抑圧し、最後に各投影方向の信号を足し合わせることで再構成画像を生成する。さらにこの合成処理過程において共通な処理を抽出し、木構造状に変形することで演算量を減少させ高速な処理を実現する。再構成画素数を  $N \times N$ 、投影回数を  $P$  とすると、FBP 法での計算量が  $N^2 P$  に比例するのに対し、TSFB 法での計算量は  $N^2 \log_2 P$  に比例する。

一方、近年 1 個の汎用的プロセッサコア (PowerPC Processor Element: PPE) と 8 個のマルチメディア処理に適したプロセッサコア (Synergistic Processor Element: SPE)、及びそれらのプロセッサコア間のデータ転送を行うためのデータ転送バスである Element Interconnect Bus (EIB) から構成される Cell Broadband Engine (CBE)[4] が注目されている。図 1 は CBE の構成概略図を示している。CBE は複数の演算コアを同時平行動作させることにより、膨大な計算量を持つ処理の高速化を可能にする。このため、数値解析や画像、動画、音声処理など多大な演算量が要求される処理においては CBE の持つ特徴を生かすことで高速な処理を実現することができる。そこで本研究では、その特徴に注目し CBE 上に TSFB 法を実装することで、より高速な再構成の実現を目的とする。CBE 上で処理を行う際には、PPE、SPE それぞれの特徴に基づく処理の割り当てや、各コア間でのデータ転送時に発生するオーバーヘッドを考慮した実装を行うことが必要となる。本論文では、それらを考慮し効果的な TSFB 法の並列化手法について検討する。以下、2 章では CT 再構成の基本原理について述べ、3 章では TSFB 法について要約する。続

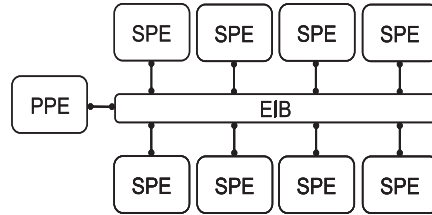


図 1: CBE の構成

いて 4 章にて CBE を構成する PPE、SPE 及び各コア間のデータ転送法について述べ、5 章にて TSFB 法の並列化について述べる。6 章では TSFB 法を CBE 上にて実装し、シミュレーションを行うことで、並列化の効果を明らかにする。

## 2 再構成の原理

X 線 CT 撮影では対象物体に対しあらゆる方向から X 線を照射し、検出器において物体通過により減衰した X 線の強度を測定する。その強度から Radon 変換の値を算出し、計算機を用いて任意の断面像を再構成する。本章では、Radon 変換と投影切断定理、FBP 法について述べる。

### 2.1 Radon 変換と投影切断定理

断層面内の位置ベクトルを  $\mathbf{x} = (x_0, x_1)^T$ 、 $x_0$  軸と角  $\theta$  をなす同面内の単位ベクトルを  $\Theta = (\cos \theta, \sin \theta)^T$  と記す。このとき、断層面内において原点からの距離  $r$ 、法線  $\Theta$  の直線に沿う分布関数  $f(\mathbf{x})$  の Radon 変換は次式で定義される。

$$\mathcal{R}f(r, \Theta) = \int \int f(\mathbf{x}) \delta(r - \mathbf{x} \cdot \Theta) dx \quad (1)$$

上式において  $\delta(\cdot)$  はそれぞれディラックのデルタ関数、内積を示す。さらに 2 次元角周波数を  $\omega = (\omega_0, \omega_1)^T$  とすると、 $f(\mathbf{x})$  の 2 次元フーリエ変換  $\mathcal{F}_2 f(\omega)$  と  $\mathcal{R}f(r, \Theta)$  の  $r$  に関する 1 次元フーリエ変換  $\mathcal{F} \mathcal{R}f(\omega)$  の間に以下の関係が成立する。

$$\mathcal{F}_2 f(\omega, \Theta) = \mathcal{F}_1 \mathcal{R}f(\omega, \Theta) \quad (2)$$

上式は、方向  $\Theta$  の Radon 変換の 1 次元スペクトルが原信号スペクトルの方向  $\Theta$  の切断を与えることを示しており、「投影切断定理」と呼ばれる。

\* 岡山大学大学院自然科学研究科, 〒 700-8530 岡山県岡山市北区津島中 3-1-1

2.2 フィルタ補正逆投影法

$\mathcal{F}_2 f(\omega)$  の逆変換を極座標積分として表し、式 (2) を用いれば、逆 Radon 変換は次式で示される。

$$f(x) = \frac{1}{2\pi} \int_0^\pi q(x \cdot \Theta, \Theta) d\Theta \quad (3a)$$

$$q(r, \Theta) = \frac{1}{2\pi} \int_{-\infty}^\infty \mathcal{F}_1 \mathcal{R} f(\omega, \Theta) |\omega| e^{j\omega r} d\omega \quad (3b)$$

上式において式 (3a) の操作は逆投影と呼ばれる。また式 (3b) は、 $|\omega|$  の周波数領域において補正関数 [3],[7],[8] を施す操作である。これより、この再構成法は「フィルタ補正逆投影法 (Filtered backprojection: FBP)」と呼ばれる。以上の計算は、実際には離散化されて実行される。

3 木構造フィルタバンクによる再構成

本章では FBP 法と比べ高速な再構成を実現する木構造フィルタバンク (TSFB) 法について説明する。まず、ある方向の Radon 変換が 2 次元フィルタバンク系の分析処理と等価であり、CT 再構成はその逆操作、つまり合成処理であることを述べる。次に、合成処理に用いる平行四辺形フィルタバンクを木構造状に変形した TSFB について説明する。

3.1 Radon 変換と分析フィルタ

フィルタバンク系 [9] においては多段接続されたフィルタにより入力信号を異なる帯域信号に分割し、各帯域に相応したダウンサンプリングを行う。この処理を分析処理という。一方、式 (1) の積分操作は法線が  $\Theta$  である直線に沿う線積分であり、この操作は  $\Theta$  方向に狭帯域方向性フィルタを施す操作に相当する。 $\Theta$  方向に沿う直線上の値を取り出すと式 (1) の Radon 変換を得る。この操作は直線状にダウンサンプリングすることに対応する。TSFB 法ではこのように Radon 変換をフィルタバンク系における分析処理とみなし、 $\Theta$  に相応する方向性を有する合成フィルタにより、CT 再構成を実行する。またダウンサンプリングは、 $N \times N$  画素の再構成領域に対して図 2 に示すような  $n_0$  方向の標本ライン上で行う。これらの標本値は次節にて述べる。

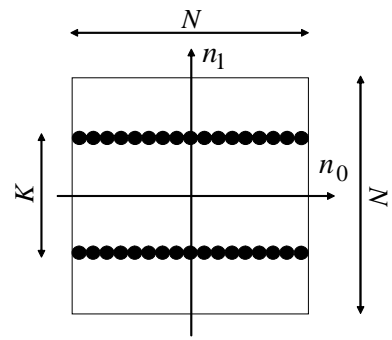


図 2: ダウンサンプリング

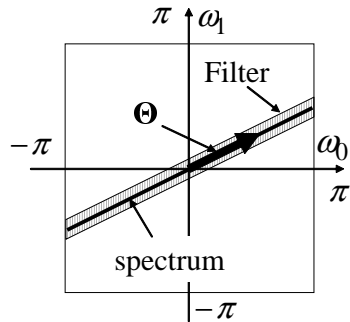


図 3: 合成フィルタ

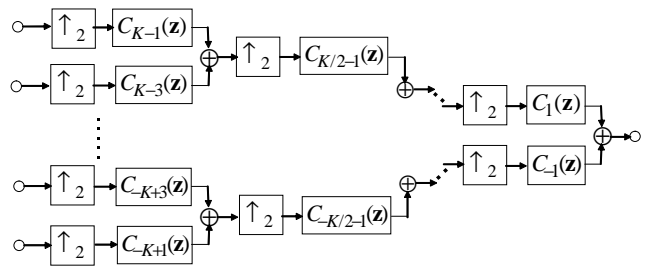


図 4: 木構造フィルタバンク

3.2 逆投影信号とそのスペクトル

以降の議論では式 (3a) において、方向  $\Theta$  の逆投影信号  $q(x \cdot \Theta, \Theta)$  を  $x$  に関して間隔  $\tau$  の正方形格子  $\{\tau \mathbf{n} | \mathbf{n} = (n_0, n_1)^T\}$  で標本化した信号を  $b(\mathbf{n}, \Theta)$  と記し、これを逆投影と呼ぶ。 $b(\mathbf{n}, \Theta)$  は  $\Theta$  方向にのみ変化し、 $\Theta$  に直交する方向には変化が無い。そのため逆投影信号のスペクトル  $\mathcal{F}_2 b(\omega, \Theta)$  は、図 3 に示すように、 $\Theta$  方向上の輝線スペクトルとなる。 $b(\mathbf{n}, \Theta)$  を  $n_1$  方向に因子  $K$  でダウンサンプリングし、その直後に同因子  $K$  でアップサンプリングすると、このとき得られる 2 次元信号  $g(\mathbf{n}, \Theta)$  のスペクトルは次式により表現される [9]。

$$\mathcal{F}_2 g(\omega, \Theta) = \frac{1}{K} \sum_{k=0}^{K-1} \mathcal{F}_2 b(\omega - \frac{2\pi k}{K} \mathbf{u}_1, \Theta) \quad (4)$$

ここで  $\mathbf{u}_1$  は  $\omega_1$  方向の単位スペクトルである。上式は元のスペクトルの平行な余分のイメージを  $K-1$  個含んでいることを示している。このイメージを抑圧するために、図 3 中で陰影で示す部分を通過させる狭帯域平行四辺形フィルタが用いられる。

3.3 平行四辺形フィルタ

図 3 のような平行四辺形通過帯域を持つフィルタは、次式で示す全域通過フィルタ  $A(z)$  を用いた直線位相フィルタ  $H(z)$  により構成される。

$$A(z) = \frac{1 + az^{-1}}{1 + az} \quad (5)$$

$$H(z) = \frac{zA(z) + A(z^{-1})}{2} \quad (6)$$

ただし  $a$  は定数で、ここでは  $a = 0.5$  としている。平行四辺形フィルタ  $C_k(z)$  は  $H(z)$  を用いて次のように構成される。

$$C_k(z) = 0.5 * [1 + z_0^{-\frac{1-k}{2}} z_1^{-1} H(z_0) H(z_0^{-k} z_1^2)] \quad (7)$$

上式において  $\mathbf{z} = (z_0, z_1)^T$  であり、奇変数  $k (= -K+1, -K+3, \dots, K-3, K-1)$  は平行四辺形状の通過帯域の傾きを表す。角度は  $\omega_0$  方向を基準に  $\theta_k = \arctan(k/K)$  となる。

ダウンサンプリングにより生ずるイメージスペクトルを抑制

する狭帯域平行四辺形フィルタは、アップサンプリングと平行四辺形フィルタの多段接続により実現される。合成フィルタバンクを構成する場合、異なる方向の共通処理を抽出し、木構造状に変形することにより、計算量の削減が可能である。図4にTSFBを示す。図中 $\uparrow_2$ は $n_1$ 方向のアップサンプリングである。TSFBの第 $i$ ( $= 1, 2, \dots, \log_2 K$ )段、上から $k$ ( $= -K_i + 1, -K_i + 3, \dots, K_i - 3, K_i - 1$ ; ただし、 $K_i = K/2^{i-1}$ )番目の処理では、像面で $\theta_{i,k} = \tan^{-1}(k/K_i)$ の方向の平行四辺形フィルタを用いており、次式を満たすような方向角の投影信号の合成処理を行っている。

$$\tan^{-1}\left(\frac{k-1}{K_i}\right) \leq \theta < \tan^{-1}\left(\frac{k+1}{K_i}\right) \quad (8)$$

上式より図4で示すTSFBでは $\pm\pi/2$ に近い角度まで有限の $k$ に対応することは不可能である。そこで方向 $\theta$ を $[-\pi/4, \pi/4]$ と $[\pi/4, 3\pi/4]$ の2範囲に分割し、それぞれの範囲で合成処理を行った後に行列転置し加え合わせることで $-\pi/4 \leq \theta < 3\pi/4$ の範囲の投影方向に対応している。

#### 4 CBEの概要 [4]

本章ではCBEを構成するPPE, SPE, またPPE-SPE間のデータ転送方法それぞれについて説明する。まずCBE内に含まれる2種類のプロセッサコアの内、汎用的なプロセッサコアであるPPEについて述べる。次いでマルチメディア処理に適したプロセッサコアであるSPEについて述べ、その後、各プロセッサ間のデータ転送法であるDirect Memory Access 転送について説明する。

##### 4.1 PowerPC Processor Element (PPE)

図5にPPEの構成を示す。PPEは演算処理を行う核となるユニットであるPowerPC Processor Unit (PPU)とメインメモリへのデータアクセスを制御するユニットであるPowerPC Processor Storage Subsystem (PPSS)から構成され、その役割は、メインメモリや外部デバイスへの入出力制御に加え、後述するSPEに対するタスク割り当てが主で、CBEに1基搭載されている。PPUには1次キャッシュとして32KBの命令キャッシュと32KBのデータキャッシュを有しており、PPSSはPPUからのメモリアクセスを高速化させるために512KBの2次キャッシュを搭載している。

##### 4.2 Synergistic Processor Element (SPE)

図6にSPEの構成を示す。SPEはSynergistic Processor Unit (SPU)とMemory Flow Controller (MFC)で構成されている。SPUはSingle Instruction Multiple Data(SIMD)演算能力を有し、高密度な演算を行うことが可能である。SPUにはデータを格納するためのLocal Storage (LS)と呼ばれる256KBのメモリを持つ。しかし、SPUは自身のLS以外のメモリへの直接アクセスは不可能である。そのため、SPUがメインメモリ上または他のSPUのLSにデータのロード/ストアを行う際は、MFCによるデータ転送が必要となる。メインメモリとLSまたは各SPUのLS間でのデータ転送は後述するDirect Memory Access (DMA) 転送により実現される。

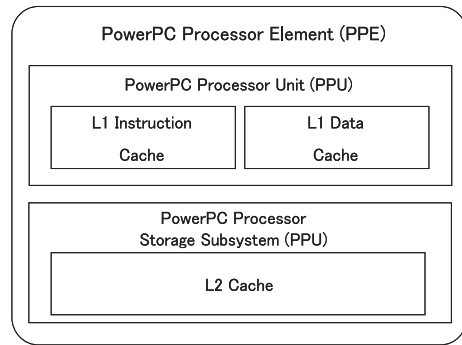


図5: PPEの構成

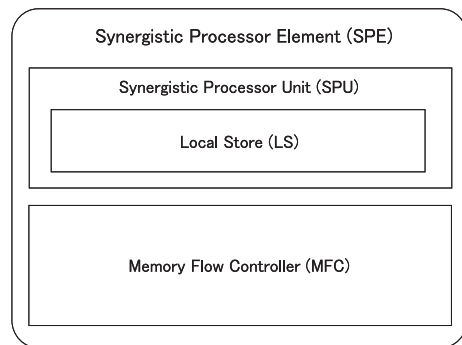


図6: SPEの構成

#### 4.3 Direct Memory Access(DMA) 転送

DMA 転送はMFCが管理するデータ転送であり、メインメモリと各SPEのLS間のデータ転送を行うために用いられる。DMA 転送は一度の転送で最大16KBまでのデータを転送することができる。また転送されるデータは16の倍数byte境界に整列されている必要があり、128byteで整列されているときに最大効率を発揮する。

#### 5 TSFBのCBEへの実装

本章ではTSFBのCBEへの実装法について述べる。CBE上での実装に際し、その性能を十分に発揮させるためには以下の2点を考慮する必要がある[10]。

1. 複数のSPEを並列に利用し、より少ない時間でより多くの演算を行うこと
2. SPE上でSIMD演算を利用し、より少ない命令数でより多くの演算を行うこと

以降では上記項目を考慮し、TSFB法の合成処理過程の並列化及びSIMD演算を用いたフィルタ処理を提案する。

##### 5.1 TSFBの並列化

例として、3段構成のTSFBに対し4個のSPEを用いた合成処理過程の並列化について検討する。3章で述べたように、TSFBは計算量を削減するため木構造状の形体を成しており、第 $i$ 段のフィルタ処理を行うためには第 $i-1$ 段のフィルタ処理の出

力結果を用いる必要がある．そのため図7に示すように，2つのSPEに1~3段目の処理，残りのSPEに1,2段目の処理を割り当てると，3段目の処理を行うためにSPE間で前段のフィルタ処理結果をデータ転送する必要が生じる．このためデータ転送が完了するまでは3段目の処理を行うことができず，その結果オーバーヘッドが生じ処理時間の増加を引き起こす．図7においてはSPE2からSPE1へ，SPE3からSPE4へデータ転送を行っており，ここでオーバーヘッドが発生する．TSFB法の合成処理をCBE上で効果的に並列動作させるためには，この問題を解決することが不可欠である．

そこでTSFBを改良することで，SPE間のデータ転送を不要にしオーバーヘッドの発生を回避する．合成処理過程では，2つの隣接するフィルタ処理結果を加算(図7中 $\oplus$ で示す)し，その結果を次段のフィルタ処理の入力としている．すなわち，加算される2つのフィルタ処理結果が異なるSPEに保持されている場合において，該当するSPE間でのデータ転送が必要となる．そこで，合成処理過程の一部をSPEの個数に応じて図8に示すように変形する．図8では2段目のフィルタ処理後に隣接する2信号を加算せず，それぞれの信号に対し3段目のフィルタ処理( $C_1$  or  $C_{-1}$ )を施し，その結果を各SPEからPPEに転送し加え合わせている．この変形により各SPE内で処理が完結し，SPE間でのデータ転送が不必要となるためオーバーヘッドの発生を回避することができる．

### 5.2 SIMD 演算を用いたフィルタ処理

1命令で1つのデータに対して逐次的に処理を行うスカラ演算方式に対し，SIMD演算は1命令で複数の処理を行う演算方式である．例として図9にそれぞれの演算方式による4個のデータに対する加算処理を示す．スカラ演算では，図9(a)のように4個の演算結果を求めるためには4回の加算を逐次的に実行する必要がある．一方，SIMD演算は図9(b)のように1回の加算により4個の演算結果を求めることが可能である．このため，SIMD演算はスカラ演算と比較し，少ない命令数で同様の処理を効果的に実行することができる．

本章の冒頭で述べたように，Cellの性能を十分に発揮させるためにはSPE上でSIMD演算を利用し，少ない命令数で演算を行うことが必要である．そこで本節では，SIMD演算を用いたフィルタ処理法を提案する．式(7)で表現される平行四辺形フィルタを2次元入力信号 $X(z)$ に施す場合には， $X(z)$ の $z_0^{-(1-k)/2} z_1^{-1}$ 遅延要素に対して1次元フィルタ $H(z_0)$ と $H(z_0^{-k} z_1^2)$ を縦続して施し，元の信号に加えて0.5を乗じることでフィルタ出力 $Y(z)$ を得る．図10ではフィルタ $C_k$ の処理ブロック図を示している．またフィルタ $H(z)$ は式(5)の全域通過フィルタ $A(z)$ により構成され，平行四辺形フィルタの実装にあたっては実空間領域において $A(z)$ を施すことにより計算を行う．実空間領域において $A(z)$ は， $x(n), y(n)$ をそれぞれ入力，出力信号とすると $n$ を減少させながら次式により計算する．

$$y(n) = x(n) + 0.5 * [x(n-1) - y(n+1)] \quad (9)$$

上式を $H(z_0)$ では入力信号に対し $n_0$ 方向， $H(z_0^{-k} z_1^2)$ では $n_0$ 方向に $-k$ 進み， $n_1$ 方向には2進む斜め方向に施す．この処理に対してSIMD演算を適用するために，入力信号の $n_0$ または $n_1$ 方向に連続する信号を1つのベクトル信号として扱う．図11にSIMD演算を用いた $H(z_0), H(z_0^{-1} z_1^2)$ それぞれの処理を示す．図11(a)では， $n_1$ 方向に連続する4つの信号を1つの入

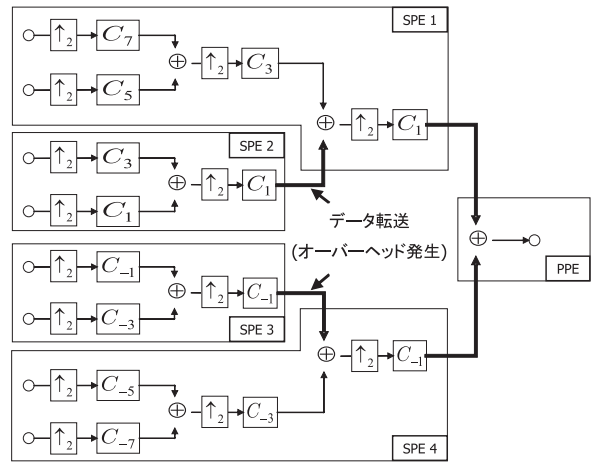


図7: 3段構成のTSFBに対し4個のSPEを用いた並列化．2段目の処理終了後，SPE間の転送が完了するまで3段目の処理を行うことができず，オーバーヘッドが発生する．

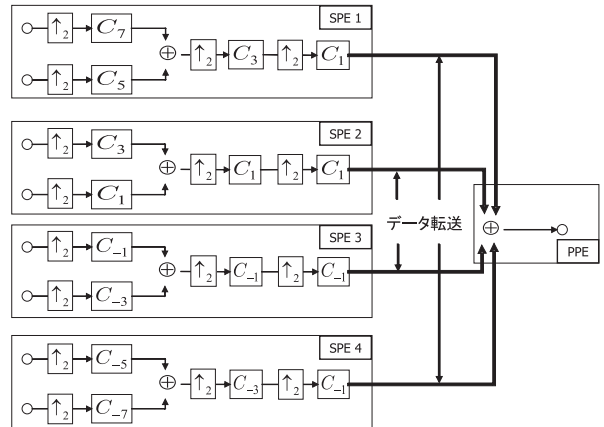
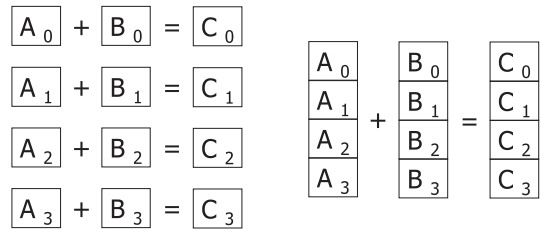


図8: SPEの個数に応じたTSFBの変形．SPE間でのデータ転送が必要なく，オーバーヘッドが発生しない．



(a) スカラ演算 (b) SIMD 演算

図9: スカラ演算とSIMD演算による加算処理

力ベクトル信号 $v(n)$ として扱っている<sup>1</sup>．SIMD演算を用いたフィルタ処理の出力ベクトル信号を $w(n)$ とすると，式(9)の

<sup>1</sup> 1つのベクトル信号を構成する信号の要素数は，要素となる信号のデータ型により異なる．

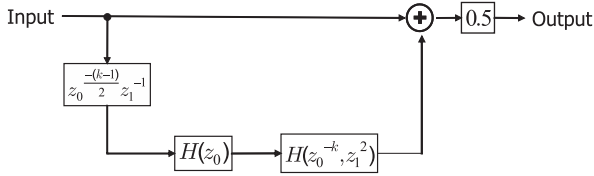
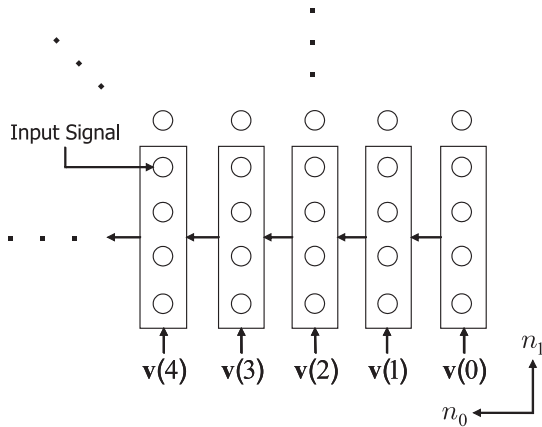
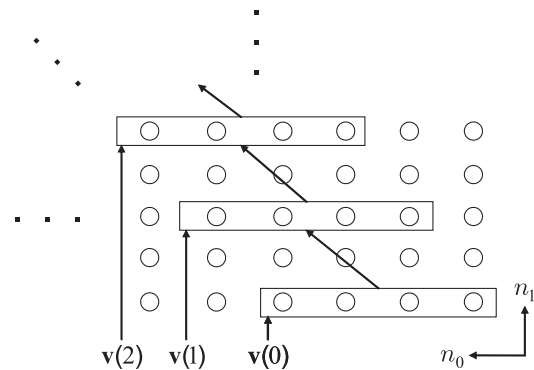


図 10: 平行四辺形フィルタ  $C_k$  の処理ブロック図



(a)  $H(z_0)$ :  $n_1$  方向に連続する信号を 1 つのベクトル信号として扱う



(b)  $H(z_0^{-1} z_1^2)$ :  $n_0$  方向に連続する信号を 1 つのベクトル信号として扱う

図 11: SIMD 演算を用いたフィルタ処理

フィルタ処理は次式により表現される

$$\mathbf{w}(n) = \mathbf{v}(n) + 0.5 * [\mathbf{v}(n-1) - \mathbf{w}(n+1)] \quad (10)$$

一方、図 11(b) では  $n_0$  方向に連続する 4 つの信号を 1 つのベクトル信号として扱い、 $n_0$  方向に  $-1$  進み、 $n_1$  方向に 2 進む斜め方向に上式のフィルタ処理を施している。このように複数の信号を 1 つの信号として扱い SIMD 演算を用いることで、スカラー演算による処理と比較少ない演算量でフィルタ処理を行うことが可能となる。

TSFB 法の合成処理では、入力された 2 次元信号内で一定方向に連続する信号に対しフィルタ処理を行うため、容易に SIMD 演算を用いて処理を行うことが可能である。一方、逆投影は 1 つの画素値を求めるために式 (3a) 中の 2 変数  $x, \Theta$  により定まる 2 次元信号  $q(x \cdot \Theta, \Theta)$  内で、連続性のない信号を足し合わせる処理であるため、SIMD 演算を用いた逆投影処理を実現することは困難である。そのため合成処理に SIMD 演算を用いることが可能である TSFB 法は逆投影処理のみで再構成を行う FBP 法に比べ、CBE への実装に適したアルゴリズムであるといえる。

## 6 再構成シミュレーション

本章では、最初に通常の TSFB 法 (図 7) と変形を施した TSFB 法 (図 8) をそれぞれ CBE 上に実装し、それらの処理時間を比較することで提案手法の有効性を示す。次に FBP 法の逆投影処理を CBE 上に実装し並列処理を行った場合の処理時間と比較することで、TSFB 法の並列化の効果を示す。また以降の実験では CBE の実行環境として IBM Full-System Simulator [11] を用いた。

### 6.1 TSFB の変形の有無による処理時間の比較

本実験では、 $256 \times 256$  画素の画像再構成実験を行った。投影回数は  $P = 256$  とし、Radon 変換を求め、続いて補正関数 (Shepp-Logan フィルタ [8]) を施した。その後フィルタリングした投影データを各 SPE に転送し、それぞれの SPE において TSFB の入力信号となる逆投影信号を作成する。逆投影信号の作成においては、3.3 で述べた投影角  $\theta$  の 2 つの範囲  $[-\pi/4, \pi/4]$ 、 $[\pi/4, 3\pi/4]$  をそれぞれ 4 区間に等分し、分割した投影角の範囲を各 SPE に割り当て、その範囲内でダウンサンプリング/アップサンプリング因子である  $K = 64$  画素ごとに逆投影を行った。作成した逆投影信号は各 SPE において  $\log_2 K = 6$  段の TSFB に入力した。提案法では TSFB の 5.6 段目に図 8 に示す変形を施した。またフィルタ処理  $H(z_0)$  と  $H(z_0^{-k} z_1^2)$  は両手法とも SIMD 演算を用いて処理を行った。本実験では再構成画像の画素値として 2byte のデータ型を用いたため、SIMD 演算に用いられるベクトル信号の要素数は 8 個となる。また PPE では各 SPE の制御、PPE-SPE および SPE-SPE 間のデータ転送の制御を行った。表 1 に通常の TSFB 法 (Type A と示す) と変形を施した TSFB 法 (Type B と示す) の TSFB による合成処理を CBE 上にて実行した場合の処理時間をそれぞれ示す。

表 1: Type A と Type B の合成処理時間の比較

	Processing time[ms]
Type A	10
Type B	8

表 1 より、TSFB に変形を施した Type B の合成処理に要する時間は無変形の Type A の結果と比べ高速に動作していることがわかる。これより、TSFB 法を並列に実装する際には使用するコアの個数に応じて 5.1 に示す変形を施すことが効果的であるといえる。

## 6.2 FBP 法との比較

前述の実験同様に  $256 \times 256$  画素の画像再構成実験を行った。補正関数を施す処理では、両手法とも式 (3b) 中の  $\Theta$  で表す区間を 8 等分し、各 SPE で分割した区間ごとに補正処理を行うことで並列化を行った。FBP 法の逆投影処理は、再構成画像を  $n_1$  方向に 8 等分し、各 SPE において部分的に画像再構成を行うことで並列化を行った。TSFB 法については前節 6.1 で述べた Type B を実装し処理時間を測定した。表 2 に並列化した TSFB 法、FBP 法に要する処理時間をそれぞれ示す。また参考として 1 スレッドによる FBP 法と TSFB 法の処理時間 (表 2 中 ( ) 内の数値) も併せて示す。

表 2: 並列化した FBP 法と TSFB 法の処理時間の比較

	Processing time[ms]			
	補正関数	逆投影処理	合成処理	合計
TSFB	29 (257)	21 (255*)	8 (22*)	58 (534*)
FBP	29 (257)	224 (4028*)	-	253 (4285*)

\* CPU: Intel Core 2 CPU 2.13GHz

表 2 中の処理時間の合計に注目する。並列化した TSFB 法と 1 スレッドによる TSFB 法の結果を比較すると、並列化により約 9.2 倍高速に処理されていることがわかる。また並列化した FBP 法と比較しても約 4.4 倍高速に処理されていることから、TSFB 法を CBE 上に実装し並列に動作させることで、より高速な再構成を実現することが可能であることがわかった。

## 7 むすび

本論文では CBE の持つ優れた演算能力に注目し、より高速な再構成を実現するため、高速画像再構成法である TSFB 法の CBE への実装法について述べた。実装に際し、SPE 間でのデータ転送を起因とするオーバーヘッドを回避し合成処理を効率良く並列動作させるため、TSFB の変形手法を提案した。さらにフィルタ処理の演算量を減少させるために 1 命令で複数の処理を行う SIMD 演算を用いたフィルタ処理を示した。シミュレーション結果より、これらの改良を施し CBE 上で並列動作させた TSFB 法は、1 スレッドによる TSFB 法の処理時間と比較し 7 倍以上高速に処理される結果を得た。

最後に CBE を用いた 3 次元画像再構成の可能性について述べる。TSFB 法は、Feldkamp ら [12] により提案された近似的 3 次元画像再構成法 (Feldkamp 法) に対しても適用可能であることが知られている [13]。Feldkamp 法はコーンビーム投影から複数枚の 2 次元画像を再構成することで近似的に 3 次元再構成を行う手法であり、[13] では 2 次元画像の再構成手段として TSFB 法を利用している。そのため、TSFB 法を用いた 2 次元画像再構成を各 SPE で同時に行うことで、高速な 3 次元画像再構成が可能であると考えられる。

## 参考文献

[1] 遠藤真広, “医用イメージング技術の最近の発展,” 信学論 (D-II), vol.J87-DII, no.1, pp.3-18, Jan. 2004.

- [2] A. Rosenfeld and A.C. Kak, Digital Picture Processing, Academic Press, 1982.
- [3] A.K. Jain, Fundamentals of Digital Image Processing, Prentice Hall, 1989.
- [4] Sony Computer Entertainment Inc. : Cell Broadband Engine Architecture. <http://cell.scei.co.jp/>.
- [5] 村上純一, 溝脇一成, 森川良孝, “木構造フィルタバンクを用いた CT の再構成アルゴリズム,” 信学論 (D-II), vol.J84-DII, no.3, pp.580-589, March 2001.
- [6] 植田和弘, 守本弘明, 村上純一, 森川良孝, “木構造フィルタバンクを用いた CT 再構成の高精度化,” 信学論 (D-II), vol.J92-DII, no.7, July 2009, 採録決定
- [7] G.N. Ramachandran and A.V. Lakshminarayanan, “Three-dimensional reconstruction from radiographs and electron micrographs: Application of convolutions instead of Fourier transactions,” Proc. Nat. Acad. Sci. USA, vol.68, no.9, pp.2236-2240, 1971.
- [8] L.A. Shepp and B.F. Logan, “The Fourier reconstruction of a head section,” IEEE Trans. Nucl. Sci., vol.NS-21, no.3, pp.21-43, 1974.
- [9] P.P. Vaidyanathan, Multirate Systems and Filter Banks, Prentice Hall, 1993.
- [10] Fixstars Corporation: PLAYSTATION3® Linux Information Site. <http://cell.fixstars.com/ps3linux>.
- [11] IBM Co. : IBM Full-System Simulator for the Cell Broadband Engine Processor . <http://www.alphaworks.ibm.com/tech/cellsystemsimm>.
- [12] L.A. Feldkamp, L. C. Davis, and J. W. Kress, “Practical cone-beam algorithm,” Journal of Optical Society America A, vol.1, no.6, pp.612-619, 1984.
- [13] 村上純一, 溝脇一成, 森川良孝, “木構造フィルタバンクを用いた円錐ビームヘリカル CT の再構成アルゴリズム,” 信学論 (D-II), vol.J84-DII, no.10, pp.2346-2349, Oct. 2001.