

# アクセラレータを用いた ルジャンドル変換の高速化

窪田 昌史<sup>†1</sup> 齊藤 彰久<sup>†1</sup> 北村 俊明<sup>†1</sup>

ルジャンドル変換は気候変動や気象予報のシミュレーションなどに用いられる重要なアルゴリズムであり、その高速化が求められている。各種アクセラレータ (Cell B.E., GPU(Tesla), ClearSpeed) および Intel x86-64 のマルチコアプロセッサ向けにルジャンドル変換を実装し、その性能評価を行った。その結果、GPU を用いることで、比較的プログラム作成が容易、かつ、高い性能が得られることを確認した。

## Performance Improvement of Legendre Transforms by Accelerators

ATSUSHI KUBOTA,<sup>†1</sup> AKIHISA SAITO<sup>†1</sup>  
and TOSHIKI KITAMURA<sup>†1</sup>

Legendre transforms are import algorithms which are used for simulations such as global climate change and weather forecasting. Therefore, performance improvement of the transforms has been investigated. We implemented the Legendre transforms on several accelerators including Cell B.E., GPU(Tesla) and ClearSpeed and x86-64 multicore processor and evaluated their performance. As a result, we confirmed that high performance was achieved by using GPU.

### 1. はじめに

天気予報、気候変動、地球温暖化の解明などのため、全球大気モデルが広く用いられている。多くの研究機関でモデルの構築、改良が行われており、米国 NCAR の CCM<sup>1)</sup>、地

<sup>†1</sup> 広島市立大学  
Hiroshima City University

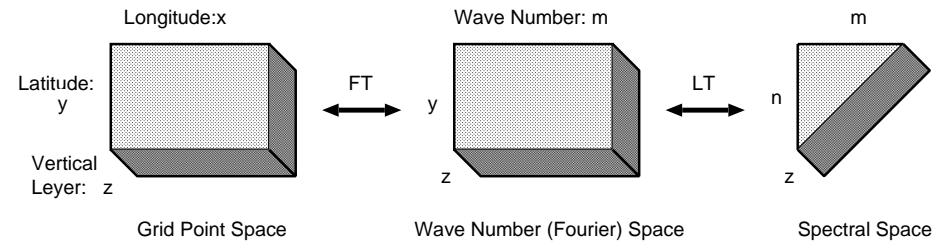


図1 スペクトル変換法  
Fig.1 Spectral Transform Method

球シミュレータをターゲットとして構築された AFES<sup>2)</sup> などの例がある。全球大気モデルにおいては、スペクトル変換法が広く用いられている。その中で、ルジャンドル変換の実行に多くの時間を要するため、須田<sup>3)</sup> など、その高速化が試みられている。

本報告では、各種アクセラレータ (Cell B.E., GPU(Tesla), ClearSpeed) および Intel x86-64 のマルチコアプロセッサを用いたルジャンドル変換の高速化について述べる。特に、高解像度のモデルを用いて大量のデータをクラスタ上で扱う際に、そのクラスタのノードのアクセラレータを活用することを考慮した高速化手法を検討する。

以下、2章でルジャンドル変換とその並列化について説明し、3章でルジャンドル変換の各種アクセラレータ向けの高速化手法について述べる。4章で性能評価結果を示し、最後に5章でまとめとする。

## 2. スペクトル変換法とルジャンドル変換

### 2.1 スペクトル変換法

球面上の大気の偏微分方程式を解く手法としてはスペクトル変換法 (Spectral Transform Method) が広く用いられている。気温などの各種物理変数は球面調和関数展開されることで高い精度を保って近似される。次式は経度  $\lambda_i$ 、緯度  $\mu_j$ 、高度  $z_k$  の地点の物理変数  $\xi(\lambda_i, \mu_j, z_k)$  を表している。

$$\xi(\lambda_i, \mu_j, z_k) = \sum_{m=-M_j}^{M_j} \sum_{n=|m|}^{N_m} \xi_n^m(z_k) P_n^m(\mu_j) e^{im\lambda_i} \quad (1)$$

以下、この各項の係数を求めるために用いられるフーリエ変換とルジャンドル変換について説明する。

図1は変換される過程を表したものである。左端の直方体は球面上のグリッドを直方体上に並べ直したものであり、横が東西方向  $x$ 、縦が南北方向  $y$ 、高さが鉛直方向  $z$  を表す。この3次元グリッド空間 (Grid Point Space) 上の各点は、2次元の  $xy$  平面ごとに変換される。

東西方向に同一の緯度上に並ぶ  $N_I$  個のグリッド上の点の物理変数  $\xi(\lambda_i, \mu_j, z_k)$  はフーリエ変換 (FT; Fourier Transform) によって図1の2番目の直方体のように波数  $m$  の順に並べられる波数空間 (Wave Number Space) 上の点  $\xi^m(\mu_j, z_k)$  へと変換される。

$$\xi^m(\mu_j, z_k) = \frac{1}{N_I} \sum_{i=1}^{N_I} \xi(\lambda_i, \mu_j, z_k) e^{-im\lambda_i} \quad (2)$$

ここで、実数  $\xi(\lambda_i, \mu_j, z_k)$  をフーリエ変換と、 $\xi^m(\mu_j, z_k)$  と  $\xi^{-m}(\mu_j, z_k)$  は共役複素となる。そのため、波数  $m$  は正負の整数であるが、 $m \geq 0$  のみ計算すればよい。

次に、波数空間で南北方向のすべての緯度について同一の波数  $m$  の  $N_J$  個の値  $\xi^m(\mu_j, z_k)$  をルジャンドル変換 (LT; Legendre Transform) することで図1の右端のような  $m \times n$  のスペクトル空間 (Spectral Space) 上の点  $\xi_n^m(z_k)$  が得られる。

$$\xi_n^m(z_k) = \sum_{j=1}^{N_J} \xi^m(\mu_j, z_k) P_n^m(\mu_j) w(\mu_j) \quad (3)$$

ここで  $P_n^m(\mu_j)$  はルジャンドル陪関数と呼ばれる実数、 $w(\mu_j)$  はガウス積分の重みである。

偏微分方程式の線形項については、スペクトル空間上で時間微分係数を求め、それを用いて微小時間後の値を求める。非線形項については、元の物理空間上の値を使って求めるため、スペクトル空間上の値  $\xi_n^m(z_k)$  から物理空間上の値  $\xi(\lambda_i, \mu_j, z_k)$  を逆ルジャンドル変換と逆フーリエ変換によって求める必要がある。

スペクトル空間で同一の波数  $m$  の  $N_m$  個の値  $\xi_n^m(z_k)$  を逆ルジャンドル変換 (ILT; Inverse Legendre Transform) することで図1の右端のスペクトル空間上の点  $\xi_n^m(z_k)$  から2番目の波数空間上の点  $\xi^m(\mu_j, z_k)$  が得られる。

$$\xi^m(\mu_j, z_k) = \sum_{n=|m|}^{N_m} \xi_n^m(z_k) P_n^m(\mu_j) \quad (4)$$

波数空間上で同一の緯度  $\mu_j$  上に並ぶ  $M_j$  個の値  $\xi^m(\mu_j, z_k)$  を逆フーリエ変換 (IFT; Inverse Fourier Transform) することで図1の波数空間上の点の値から左端の物理空間上の点の値  $\xi(\lambda_i, \mu_j, z_k)$  が得られる。

$$\xi(\lambda_i, \mu_j, z_k) = \sum_{m=-M_j}^{M_j} \xi^m(\mu_j, z_k) e^{im\lambda_i} \quad (5)$$

## 2.2 グリッドの形状

もとのグリッド空間の東西方向のグリッド数を  $N_I$ 、南北方向のグリッド数を  $N_J$  とすると、 $N_I = 2 \times N_J$  である。 $N_I \geq 3M + 1$  を満たす切断周波数  $M$  で球面調和関数の展開を打ち切っても数値の精度を十分に保つことができる<sup>1)</sup>。グリッドの解像度は切断周波数  $M$ 、鉛直方向のレベル数  $K$  を用いて TMLK と表される。たとえば、T1279L96 は地球シミュレータで気候変動予測に用いられた  $N_I = 3840$ 、 $N_J = 1920$  の大規模なモデル<sup>2)</sup> である。

図1の右端のスペクトル空間上には、ルジャンドル変換によって  $n = |m|$  から  $N_m$  までの係数が並ぶ。 $N_m$  は一般には波数  $m$  の関数となる。

$N_m = M$  とすれば、波数  $m$  について展開される項の個数は  $M - |m|$  である。各高度のスペクトル空間は、図1の右端のように三角形になる。このとき、三角形に並ぶデータの格納方法を工夫することで記憶領域を削減することができる。

一方、 $N_m = M + |m|$  とすれば、波数  $m$  について展開される項の個数は切断周波数と同じ定数  $M$  となる。このとき、図1の一番右のスペクトル空間は、各高度について菱形 (rhomboid) になる。各高度のスペクトル空間は、 $M \times M$  の記憶領域に格納できる。本報告では、この菱形のスペクトル空間を扱うものとする。

## 2.3 ルジャンドル変換の計算量の削減

ルジャンドル変換では、球面上の点の南北の対称性を利用して乗算回数を削減することができる。ルジャンドル陪関数  $P_n^m(\mu_j)$  とガウス積分の重み  $w(\mu_j)$  は、南北の対称な点  $\mu_j$  と  $-\mu_j$  について、

$$P_n^m(-\mu_j) = (-1)^{(n-m)} P_n^m(\mu_j) \quad (6)$$

$$w(-\mu_j) = w(\mu_j) \quad (7)$$

という関係がある。よって、式(3)のルジャンドル変換は、以下のように書き換えられる。

$$\xi_n^m(z_k) = \sum_{j=1}^{N_J/2} (\xi^m(\mu_j, z_k) + (-1)^{(n-m)} \xi^m(-\mu_j, z_k)) P_n^m(\mu_j) w(\mu_j) \quad (8)$$

よって、和  $\xi^m(\mu_j, z_k) + \xi^m(-\mu_j, z_k)$  と差  $\xi^m(\mu_j, z_k) - \xi^m(-\mu_j, z_k)$  を用意しておき、 $n - m$  が偶数の場合は和を、奇数の場合は差を使って計算する。これによって、乗算の回数を約

1/2 に削減することができる。

同様に、式 (4) の逆ルジャンドル変換においても、南北の対称性を利用した乗算回数の削減が可能である。逆ルジャンドル変換の式 (4) は、北緯  $\mu_j$  と南緯  $\mu_j$  について表すと以下のようになる。

$$\xi^m(\mu_j, z_k) = \sum_{n=|m|}^{N_m} \xi_n^m(z_k) P_n^m(\mu_j) \quad (9)$$

$$\xi^m(-\mu_j, z_k) = \sum_{n=|m|}^{N_m} \xi_n^m(z_k) P_n^m(-\mu_j) \quad (10)$$

ルジャンドル陪関数  $P_n^m$  の性質

$$\xi^{m+}(\mu_j, z_k) = \sum_{n-m=0(mod 2)} \xi_n^m(z_k) P_n^m(\mu_j) \quad (11)$$

$$\xi^{m-}(\mu_j, z_k) = \sum_{n-m=1(mod 2)} \xi_n^m(z_k) P_n^m(\mu_j) \quad (12)$$

を利用すると、波数空間上で北半球、南半球上の変数はそれぞれ

$$\xi^m(\mu_j, z_k) = \xi^{m+}(\mu_j, z_k) + \xi^{m-}(\mu_j, z_k) \quad (13)$$

$$\xi^m(-\mu_j, z_k) = \xi^{m+}(\mu_j, z_k) - \xi^{m-}(\mu_j, z_k) \quad (14)$$

と求めることができる。ゆえに、ルジャンドル変換と同様、乗算の回数を約 1/2 に削減することができる。

#### 2.4 モデルの並列性

ここまでで述べたように、各種物理変数の値は、フーリエ変換、ルジャンドル変換によってスペクトル空間上の値に変換される。次のタイムステップ後のスペクトル空間上の値を求め、逆ルジャンドル変換、逆フーリエ変換によって次のタイムステップ後の物理変数の値へと変換される。実際には、多くの処理を精度を高く保つことのできるスペクトル空間上で行い、非線形項の計算のために一時的に逆ルジャンドル変換、逆フーリエ変換によってグリッド空間へと変換し、非線形項の計算結果をフーリエ変換、ルジャンドル変換によってスペクトル空間へと変換する。

そのため、タイムステップを進めるたびに以下の処理が繰り返される。

- (1) ILT (Inverse Legendre Transform; 逆ルジャンドル変換)
- (2) IFT (Inverse Fourier Transform; 逆フーリエ変換)
- (3) NONLIN (非線形項 (Non-linear terms) の計算)

表 1 モデルの並列性

Table 1 Parallelism in the model

処理	鉛直	東西/波数	南北
ILT	p	p	p
IFT	p		p
NONLIN		p	p
FT	p		p
LT	p	p	
LIN	p	p	p
TSTEP	p	p	p

(4) FT (Fourier Transform; フーリエ変換)

(5) LT (Legendre Transform; ルジャンドル変換)

(6) LIN (線形項 (Linear terms) の計算)

(7) TSTEP (タイムステップを進める)

FT, IFT を行う高速フーリエ変換と、LT, ILT を行う正/逆ルジャンドル変換は、それぞれ  $\mathcal{O}(M^2 \log M)$ ,  $\mathcal{O}(M^3)$  という、他の処理に比べて多大な計算量が必要となる。そのため、これらの変換を並列化によって高速化することが重要となる。

上記の各処理は、3次元のデータに対して処理を行うものであり、その各次元は、鉛直方向 (z)、東西方向 (x、処理によっては波数 m)、南北方向 (y) である。

スペクトル変換法の並列化では、これらの各次元方向の並列性を利用することになる<sup>4)1)2)</sup>。表 1 は、各処理が並列化可能な次元に p と示して一覧にしたものである。どの次元をとっても、すべての処理で並列化可能とはならないこと、鉛直方向は東西、南北方向に比べて点数が少なく、十分に並列性を引き出すことができないことを考慮する必要がある。

表 2 にグリッドサイズと切断周波数の例を示す。この例では、地球の地表上の東西および南北方向に 2 倍ずつグリッド数を増加させている。モデルのデータサイズが大きくなるにつれ、1 台の計算機だけではデータが格納できなくなる。本報告では、クラスタ上でこれらのモデルを並列実行し、1 ノードに格納できるサイズにデータが分割される状況を想定する。各ノードには、4GB のメモリが搭載され、ノード内でデータが処理できるのに十分なノード数が確保できているものとする。以上の方針に基づき、表 2 のとおり、各モデルのノード数と、1 ノードあたりの記憶領域を見積もっている。

物理変数には、風速、気圧、気温などがあり、いずれも同様の変換が行われる。そのため、複数の物理変数を格納する配列を結合し、ILT, IFT, FT, LT などの変換において、1 つの

表 2 データサイズ

	東西方向	南北方向	切断周波数	垂直方向	ノード数	メモリサイズ
	$N_I$	$N_J$	M	H		
T159L29	480	240	159	29	1	308MB
T319L44	960	480	319	44	1	2.0GB
T639L64	1920	960	639	64	16	1.3GB
T1279L96	3840	1920	1279	96	64	1.3GB
T2559L152	7680	3840	2559	152	256	3.5GB

表 3 GPU の仕様

型番	NVIDIA Tesla C1060
テクノロジー	65nm
ストリーミングプロセッサコア数	240
動作周波数	1.296GHz
ピーク性能 (単精度)	933GFlops
ピーク性能 (倍精度)	78GFlops
メモリサイズ	4GB
インタフェース	PCI Express x16 Generation 2

配列として扱うことができる．このとき、鉛直方向のデータ数が物理変数の種類だけ増加するものとして扱う．

### 2.5 行列積を利用したルジャンドル変換

図 1 に示す物理空間のサイズは東西方向のグリッド数  $N_I$ 、南北方向のグリッド数  $N_J$ 、鉛直方向の高度  $H$  である．フーリエ変換によって波数  $M$  の空間に変換され、ルジャンドル変換によって縦方向に  $N_m$  個のデータをもつ空間に変換される．

ルジャンドル変換の式 (3) より、各波数  $m$  について、波数空間の行列  $F(N_J, H)$  と、ルジャンドル陪関数とガウス積分重みの積の行列  $A_W(N_m, N_J)$  からスペクトル空間の行列  $S(N_m, H)$  を行列積  $S = A_W F$  によって求めることができることがわかる．

同様に、逆ルジャンドル変換の式 (4) より、各波数  $m$  について、スペクトル空間  $S(N_m, H)$ 、ルジャンドル陪関数  $A(N_m, N_J)$  から波数空間  $F(N_J, H)$  を行列積  $F = A^T S$  によって求めることができることがわかる．

そこで、我々は各アクセラレータ上で行列積を利用してルジャンドル変換の実装を進めることとした．ここで、スペクトル空間と波数空間は複素数であるのに対し、ルジャンドル陪関数は実数であるため、単に BLAS などの行列積のルーチン呼び出すわけにはいかないことに注意する必要がある．

## 3. アクセラレータ上の実装

本章では、本研究で使用したアクセラレータやマルチコアプロセッサの特徴と、それらへの順逆ルジャンドル変換を実装について述べる．実装は扱うデータはすべて倍精度浮動小数点数とする．

### 3.1 GPU

GPU は、グラフィックス表示用の LSI として開発されたが、General Purpose GPU (GPGPU) として使用するためのプログラミング環境である CUDA などが提供され、HPC

分野でも広く使われるようになってきている．今回使用した NVIDIA Tesla は表 3 に示すとおり、単精度浮動小数点演算のピーク性能は 933GFlops と高いものの、倍精度のピーク性能は 78GFlops となっている．

ルジャンドル変換の実装には、NVIDIA から提供されている CUDA 3.1 を用い、ライブラリ CUBLAS の行列積ルーチンを使用している．

この行列積ルーチンを実行するには、データを GPU 内のデバイス・メモリに格納する必要がある．このデバイス・メモリとホスト PC 側のメモリとの間で PCI Express を介してデータを転送する必要がある．

### 3.2 ClearSpeed

ClearSpeed は SIMD 演算器を搭載したアクセラレータである．使用した CSX600 のボードは、表 4 に示すように 2 チップを搭載し、ホスト PC と PCI-X で接続される．

ルジャンドル変換の実装には ClearSpeed 用の行列積ライブラリを用いている．このライブラリを用いるとピーク性能の 96GFlops に近い性能が得られるが、性能は行列サイズに大きく依存する．行列積  $C = A \times B$  を、行列サイズ  $A(M, K)$ 、 $B(K, N)$ 、 $C(M, N)$  で実行する場合、SIMD 演算器数の制約から、 $M, N$  は 192 の倍数、 $K$  は 288 の倍数であることが推奨されている．そのため、ルジャンドル変換の実装では、この制約を満たすように行列のサイズを大きくとっている．

### 3.3 Cell Broadband Engine

Cell Broadband Engine (Cell B.E.) は、Power Processing Unit (PPU) と Synergistic Power Unit (SPU) からなるヘテロジニアスマルチコアプロセッサである．今回使用した PowerXCell 8i では、1 個の PPU と 8 個の SPU、それらを結合する高速なバスが 1 チップ内に搭載されている．

多数の SPU 上に搭載されている SIMD 演算器を稼働させて性能を引き出す点に特徴が

表 4 ClearSpeed の仕様

型番	CSX600
SIMD PE 数	96
テクノロジー	130nm
チップ数/ボード	2
ピーク性能	96GFlops
インタフェース	PCI-X 10.6GB/s

表 5 Cell B.E. の仕様

型番	PowerXCell 8i
動作周波数	2.8GHz
テクノロジー	65nm
ピーク性能 (単精度)	180GFlops
ピーク性能 (倍精度)	90GFlops
主記憶	4GB
SPE 数	8
SPE Local Store	256KB
バス EIB 帯域幅	

表 6 Intel Xeon の仕様

型番	Xeon 5365
動作周波数	3.0GHz
テクノロジー	65nm
ピーク性能 (倍精度)	48GFlops
コア数	4
2nd Cache	8MB(コア間共有)
拡張命令セット	SSSE3

ある。SIMD 演算には 128 ビット長のベクトル・レジスタが用いられ、1 レジスタあたり倍精度浮動小数点数なら 2 要素、単精度なら 4 要素が格納される。各 SPU では、倍精度浮動小数点演算を同時に 4 演算 (加減算 2 + 乗算 2)、単精度 8 演算 (加減算 4 + 乗算 4) 実行可能である。SPU から主記憶には直接アクセスできず、各 SPU 内の Local Store のみ、load/store 命令による直接アクセスが可能である。主記憶と Local Store との間は明示的なデータ転送が必要となる。

SPU の命令パイプラインは、Even と Odd に別れている。SIMD による浮動小数点演算などは Even パイプラインに投入され、ベクトル・レジスタ内のバイト毎のシャッフル操作、ロード/ストア、分岐などは Odd パイプラインに投入される。また、ベクトルレジスタ内の個々の要素を SIMD 演算以外で扱うと性能が低下する。

これらを考慮して、浮動小数点演算とシャッフル操作などを同時に実行するように演算順序を考慮することで高い性能が得られることが報告されている<sup>5)</sup>。我々もこの方針にしたがってルジャンドル変換の実装を行っている。

### 3.4 Intel Xeon

各アクセラレータとの比較のため、広く用いられている x86-64 アーキテクチャのマルチプロセッサに対してルジャンドル変換を実装した。使用したプロセッサは表 6 に示す quad core の Xeon である。ルジャンドル変換の実装には、Intel MKL 10.0.2.018 の BLAS ライ

ブラリを使用している。

なお、この Xeon プロセッサが稼動している PC は ClearSpeed のホストでもある。ClearSpeed に対して実装したルジャンドル変換では、行列積以外の処理はこの Xeon プロセッサで実行される。

## 4. 性能評価

本章では、前章で述べた各アクセラレータとマルチコアプロセッサに対して実装されたルジャンドル変換の実行結果について述べる。

表 7 から表 11 に各モデルを用いた実行結果を示す。それぞれ順変換と逆変換の実行時間と実行性能である。変換全体の実行時間を (1)+(2) としているが、その内訳 (1) と (2) はそれぞれ以下ようになる。

2.3 節で述べた乗算回数の削減のため、順変換の前、逆変換の後に加減算が必要となる。これが (1) 加減算・コピーの実行時間である。また、順逆ルジャンドル変換の前後に、順逆高速フーリエ変換や線形項の計算処理が行われるが、これらの処理間で受け渡される配列は、それぞれの処理に応じて次元の順番を入れ替える必要がある。この次元の入れ替えと、上記の乗算回数の削減のための加減算が同時に行われるため、まとめて (1) 加減算・コピーの実行時間としている。

行列積ルーチンなどを用いたルジャンドル変換のカーネル部分は (2) カーネルの実行時間としている。カーネルに含まれる実行時間は、各アーキテクチャで以下のように異なっていることに注意されたい。

- GPU: dgemm の実行時間であり、デバイス・メモリとホストとのデータ転送時間は含まれない。
- ClearSpeed: dgemm の実行時間であるが、ホスト PC のメインメモリとボードとの間の PCI-X を介したデータ転送時間が含まれる。

表 7 モデル T159L29 の実行結果

		GPU	ClearSpeed	Xeon	Cell B.E.
順変換	(1)+(2) 実行時間 (秒)	0.144	2.364	0.525	0.382
	(1) 加減算・コピー (秒)	0.080	0.269	0.430	0.032
	(2) カーネル (秒)	0.064	2.364	0.095	0.350
	(1)+(2) 性能 (GFlops)	7.447	5.748	2.035	2.799
	(2) 性能 (GFlops)	16.748	6.484	11.201	3.054
逆変換	(1)+(2) 実行時間 (秒)	0.638	2.156	0.833	0.765
	(1) 加減算・コピー (秒)	0.166	0.049	0.713	0.202
	(2) カーネル (秒)	0.472	2.107	0.120	0.563
	(1)+(2) 性能 (GFlops)	1.675	6.303	1.283	2.795
	(2) 性能 (GFlops)	2.265	6.448	8.897	3.798

表 8 モデル T319L44 の実行結果

		GPU	ClearSpeed	Xeon
順変換	(1)+(2) 実行時間 (秒)	0.955	6.898	4.513
	(1) 加減算・コピー (秒)	0.495	1.696	3.663
	(2) カーネル (秒)	0.460	5.202	0.850
	(1)+(2) 性能 (GFlops)	13.588	5.910	2.875
	(2) 性能 (GFlops)	28.224	7.838	15.260
逆変換	(1)+(2) 実行時間 (秒)	1.916	7.120	7.554
	(1) 加減算・コピー (秒)	1.040	0.162	6.662
	(2) カーネル (秒)	0.876	6.958	0.891
	(1)+(2) 性能 (GFlops)	6.774	11.452	1.718
	(2) 性能 (GFlops)	14.817	17.578	14.556

表 9 モデル T639L64 の実行結果

		GPU	ClearSpeed	Xeon
順変換	(1)+(2) 実行時間 (秒)	0.412	2.287	1.375
	(1) 加減算・コピー (秒)	0.064	0.387	0.848
	(2) カーネル (秒)	0.348	1.900	0.527
	(1)+(2) 性能 (GFlops)	45.839	17.830	13.731
	(2) 性能 (GFlops)	54.246	21.460	35.837
逆変換	(1)+(2) 実行時間 (秒)	1.085	2.168	2.011
	(1) 加減算・コピー (秒)	0.342	0.117	1.481
	(2) カーネル (秒)	0.743	2.051	0.530
	(1)+(2) 性能 (GFlops)	17.398	18.801	9.385
	(2) 性能 (GFlops)	25.414	24.844	35.590

• Cell B.E.: PPU から SPU を起動 (オフロード) してから, SPU 上での処理が終了するまでの時間を (1)+(2) の全体の実行時間とし, そのうち, SPU からのメインメモリに対する DMA 転送時間と上記の加減算を (1) とし, その残り時間を (2) カーネルの実行時間とみなしている.

• Xeon: dgemm の実行時間である.

実行性能については, (1)+(2) の全体の実行時間に対する性能と, (2) カーネルのみの実行時間に対する性能を示している.

アーキテクチャ間の性能の比較は, 実行時間を比べることで行うため, ここで求めている性能は, 各アーキテクチャ上でピーク性能にどれだけ近づいているかを確認する目的で算出している. そのため, ClearSpeed については 3.2 節で述べたように実際のデータサイズ以上のデータを用いて計算していることを考慮し, 本来不要な演算を含めた演算数を実行時間で割った値を性能としている.

各アーキテクチャの 1 ノードで表 2 のモデルを実行しているが, 記憶領域の制約から, Cell B.E. のみ T159L29 で実行を行っている.

#### 4.1 考 察

表 7 から表 11 の実行結果より, GPU が高速であることが確認できた. 使用した Tesla の次世代の GPU である Fermi では, 倍精度演算性能が大きく向上していることから, Fermi ではさらに良好な結果が得られることが予想される.

表 7 の T159L29 や表 8 の T319L44 など, グリッドサイズが小さい場合は, いずれのアーキテクチャでもコピーのオーバーヘッドが大きい, グリッドサイズが大きくなるにつれ, カーネルである行列積の実行時間が占める割合が大きくなる.

Xeon の場合は, 一番サイズの大きい表 11 の T2559L152 でも, データコピーおよび加減算の実行時間に占める割合が大きい. GPU の場合, GPU 内のデバイス・メモリと, ホストとの間の転送時間の割合が小さい. これは, PCI Express で結合される GPU とホストとの間のデータ転送が高速であることを示している. ClearSpeed では, ホスト上のメモリと ClearSpeed のボードとを結合する PCI-X バス上のデータ転送時間は (2) カーネルの実行時間に含まれているため, ボード上での行列積のみの実行時間は表の (2) カーネルの実行時間よりさらに短い.

(2) カーネルのみの性能は, グリッドサイズが大きい場合はそれぞれのハードウェアのピークに近い性能も得られている. いずれも, もっとも大きいサイズで表 11 の T2559L152 で最大の性能が得られている.

• GPU の逆変換で 74.456GFlops; ピーク性能 78GFlops

表 10 モデル T1279L96 の実行結果

		GPU	ClearSpeed	Xeon
順変換	(1)+(2) 実行時間 (秒)	0.477	2.145	1.493
	(1) 加減算・コピー (秒)	0.056	0.477	0.725
	(2) カーネル (秒)	0.421	1.667	0.768
	(1)+(2) 性能 (GFlops)	59.307	22.178	18.957
	(2) 性能 (GFlops)	67.179	28.525	36.864
逆変換	(1)+(2) 実行時間 (秒)	1.222	1.913	1.934
	(1) 加減算・コピー (秒)	0.320	0.033	1.201
	(2) カーネル (秒)	0.902	1.880	2.257
	(1)+(2) 性能 (GFlops)	23.175	23.980	14.642
	(2) 性能 (GFlops)	31.389	40.660	38.662

表 11 モデル T2559L152 の実行結果

		GPU	ClearSpeed	Xeon
順変換	(1)+(2) 実行時間 (秒)	2.741	5.967	6.357
	(1) 加減算・コピー (秒)	0.215	1.177	2.038
	(2) カーネル (秒)	2.526	4.791	4.319
	(1)+(2) 性能 (GFlops)	65.849	34.873	28.392
	(2) 性能 (GFlops)	71.456	43.438	41.787
逆変換	(1)+(2) 実行時間 (秒)	4.729	4.627	7.524
	(1) 加減算・コピー (秒)	0.895	0.034	3.316
	(2) カーネル (秒)	3.834	4.593	4.208
	(1)+(2) 性能 (GFlops)	38.169	45.891	23.989
	(2) 性能 (GFlops)	47.077	83.208	42.890

- ClearSpeed の逆変換で 83.208GFlops; ピーク性能 96GFlops
- Xeon の逆変換で 42.890GFlops; ピーク性能 48GFlops

## 5. おわりに

本報告では、気候変動や気象予報のシミュレーションなどに用いられるルジャンドル変換の各種アクセラレータ (Cell B.E., GPU, ClearSpeed) および Intel x86-64 の Xeon マルチコアプロセッサ向けの高速化について述べた。行列積ライブラリを用いることにより、Tesla, ClearSpeed, Xeon, 特に Tesla で高い性能が得られることを確認した。また、ライブラリを活用した実装により、各アーキテクチャ向けのプログラム作成が容易になることを確認した。

今後の課題として、Cell B.E. 向けのルジャンドル変換アルゴリズムの改良、順逆ルジャ

ンドル変換以外のフーリエ変換などを含めた全球大気モデルのアクセラレータ向け高速化、各種アクセラレータを用いる際に、ホストとアクセラレータを並列実行させるアルゴリズムの考案などが挙げられる。

ホストとアクセラレータの並列実行については、

- GPU の場合は、ホストと GPU の並列実行
- ClearSpeed の場合は、ホストと ClearSpeed の並列実行
- Cell B.E. の場合は、PPU と SPU の並列実行

を可能にする手段は提供されているが、アクセラレータ上のプログラムの実行時間とホスト上のプログラムの実行時間を均衡させることが困難である。そのため、実行プロファイルの活用などが必要であると考えられる。

## 参 考 文 献

- 1) Foster, I.T. and Worley, P.H.: Parallel Algorithms for the Spectral Transform Method, *SIAM J. Sci. Comput.*, Vol.18, No.3, pp.806–837 (1997).
- 2) Shingu, S., Takahara, H., Fuchigami, H., Yamada, M., Tsuda, Y., Ohfuchi, W., Sasaki, Y., Kobayashi, K., TakashiHagiwara, ichi Habata, S., Yokokawa, M., Itoh, H. and Otsuka, K.: A 26.58 Tflops Global Atmospheric Simulation with the Spectral Transform Method on the Earth Simulator, *Proc. of Supercomputing 02* (2002).
- 3) 須田礼二, 高見雅保: 高速球面調和関数変換法の誤差の解析と制御, 情報処理学会論文誌ハイパフォーマンスコンピューティングシステム, Vol.42, No.SIG12, pp.49–59 (2001).
- 4) Kubota, A., Naughton, M. and Bourke, W.: Parallelisation of a Global Atmospheric Model with MPI, Technical Report 2002-HPC-91-11 (2002).
- 5) Kurzak, J., Alvaro, W. and Dongarra, J.: Optimizing Matrix Multiplication for a Short-Vector SIMD Architecture – CELL Processor, *Parallel Computing*, Vol.35, No.3, pp.138–150 (2009).