

## Cellを用いたクラスタシステムによる 計算機合成ホログラムの高速化

荒井 大輔

千葉大学工学部電子機械工学科

Cellの大きな特徴として、PPEが1つとSPEが8つのヘテロジニアス型のマルチコア・プロセッサであることがあげられる。Cellの内部でPPEとSPEをうまく使って並列処理を行うことにより、計算処理の高速化が期待できる。特に、PCを何台も接続して分散処理させるクラスタシステムで高速化できるプログラムは、Cellにおいても実装しやすいものと言える。また、Cellで並列処理を行うことにより高速化できるプログラムは、そのクラスタシステムにおいても高速化が期待できると考えられる。そこで、今回はソニー・コンピュータエンターテイメントより発売されているCellを搭載したPLAYSTATION3(PS3)を用いて、並列処理に適した計算機合成ホログラム(CGH: Computer Generated Hologram)のアルゴリズムを実装し、さらにPS3でクラスタシステムを構築し、実際の挙動や処理時間を計測、検証した。結果、Cellとクラスタ両方の並列性を利用したCGH計算システムの構築に成功し、PS3を5台用いたクラスタシステムで、SPEの複数利用、SIMD演算などの高速化手法を用いることによりCPU(Pentium4)に対して約130倍速い計算速度が得られた。

### 1 研究背景

ホログラフィは、究極の立体映像技術とも言われるもので、三次元物体をそのまま記録・再生できる技術である。ホログラフィをリアルタイムで処理できるようになれば究極の立体テレビを作ることができるかと期待されている。このような技術は電子ホログラフィと呼ばれ、1990年ごろから研究され始めている。また今年の1月に総務省があたかも実物が目の前にあるかのように見える立体映像を映し出す「立体テレビ」の開発に民間企業と共同で乗り出す方針を明らかにした。特殊な眼鏡などを使わなくても臨場感あふれる鮮明な映像が楽しめる技術を開発し、2025年の家庭向け放送の開始を目指すとしている。

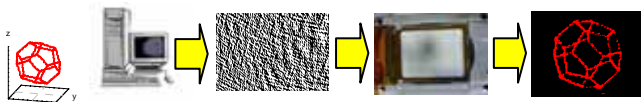


図1.電子ホログラフィの再生手順

電子ホログラフィの再生手順の例を図.1に示す。まず、コンピュータ内に、三次元座標データを用意する。このデータは、コンピュータグラフィックソフトや数値シミュレーションなどさまざまな方法で得ることができる。これを専用の計

算式によってホログラムに変換する。このホログラムをLCD(液晶ディスプレイ)などに表示して、そこに参照光を照射すると空中に元の三次元像が再生される。ホログラムは、もともと光学系で作られていたが、このように計算で作成するものをCGHと呼ぶ。CGHを用いると、逐次データを更新していくことで動画再生も可能となる。

また、参照光に光の三原色を用い、それを重ね合わせることでカラーの三次元像の再生も可能である。図2にカラー化した再生像を示す。



図2.カラーホログラフィ

ホログラムに変換する計算式は、次に示すようなコサイン関数値の重ね合わせである。物体点の座標を入力していくとホログラム上の点の値が求まる。

$$I(x_\alpha, y_\alpha) = \sum_{j=1}^N A_j \cos \left[ \frac{2\pi}{\lambda} \left( z_j + \frac{x_{\alpha j}^2 + y_{\alpha j}^2}{2z_j} \right) \right] \quad (1)$$

ここで、添え字  $\alpha$  はホログラム上の点を、 $j$  は物体を構成する点を表わす。 $x, y, z$  は座標を表し、 $x_{\alpha j} = x_\alpha - x_j$ 、

$y_{aj} = y_{\alpha} - y_j$  である。  $A_j$  は物体光の強度，  $\lambda$  は参照光の波長，  $N$  は物体点の総数である。

ホログラムの画素数を  $M$  とすると，1 枚の CGH を作成するためには(1)式のコサイン計算を  $M \times N$  回行う必要があり，リアルタイムの動画を再生するためには毎秒 30 フレーム程度のビデオレートが要求される。そのための計算量が膨大であり，CGH を用いた電子ホログラフィによる三次元テレビの実用化には今後 10~20 年を要するともいわれている。

ただし，CGH の計算は，ホログラム上の各点で独立に行うことができるという特徴を持っている。そのため，SIMD 演算を用いた並列計算，計算領域を分割した並列処理に向いており，Cell を用いたクラスタシステムで高速化ができると考えられる。そこで，本研究では CGH 計算を Cell のクラスタシステムに実装し，有効性を検証した。

## 2 システムの概要

今回構築したシステムと CGH 計算の流れを以下に示す。

### 2.1 全体の流れ

今回構築したシステムでは，各ノードの PPE でファイルやデータの管理，他ノードとの通信を行い，SPE で CGH 計算の計算式を実装している。

CGH の計算においては，ホログラム上の各点の計算結果は他の座標の計算結果と互いに独立であるので，計算の並列化が可能である。そこで，並列化の方法としては，計算領域であるホログラム上の点数を分割する手法をとった。

まず，CGH の領域を分割し各ノードに割り当てる。次に，サーバとなる一つの PS3 で物体点データを読み出し，その他の PS3 に MPI (Message Passing Interface) を用いて転送する。ここでは通信の並列化も行われ，できるだけ通信時間による遅延を無くすよう工夫している。

次に，各 PS3 では割り当てられた計算領域をさらに分割し，SPE に割り当て，それぞれの SPE で CGH を計算して求める。その後，メモリに結果を集める。

最後に，MPI を用いて各ノードから一つの PS3 に計算結果を集めて 1 枚の CGH が完成する。クラスタでの並列処理と Cell 内での並列処理，二重の並列化を行っている。

今回は，PS3 を 5 台，各 PS3 内部の Cell で SPE を 6 つ使用した。本来 Cell には 8 つの SPE が搭載されているが，PS3 では使用できる最大数が 6 つとなっている。PS3 クラスタシ

ステムの概観を図 3 に示す。

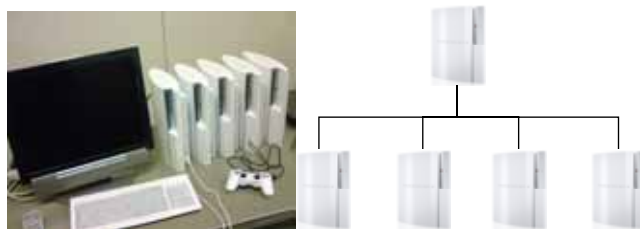


図 3.PS3 クラスタシステム

### 2.2 PPE プログラム(MPI の実装)

PPE では，まず MPI を用いて，全ノード数と自ノードの識別番号を取得する。1 台が物体点データをメモリ上に読み込み，他のノードへ転送する。各ノードで物体点データを DMA 転送に適した形に加工する。

さらに，各 SPE に渡す情報(物体点のデータや計算結果の格納場所のアドレス，担当する計算領域や各種パラメータなどの情報)を編集して，これも DMA 転送用に加工する。

そして，SPE を起動して CGH 計算を行い，その計算結果を一つのノード上に集める。

最後に表示媒体や，記録媒体に出力する。

### 2.3 SPE プログラム

SPE では，PPE で加工されたデータや情報を自身のローカルストアに DMA 転送し，そのデータをもとに CGH 計算を行う。

高速化の手法として，コサイン演算を 1 周期 256 階調の配列としてとらえ，その値を先に計算しておき，実際のコサイン演算ではその配列を参照する方法や，データを float 型のベクトルとしてとらえることで 4 つのデータを同時に取り扱う SIMD 化，計算と DMA 転送の並列化などを行った。

### 2.4 システムの特徴

CGH 計算は，物体点の座標，ホログラムを表示する LCD などの電子表示デバイスのサイズや画素間隔，参照光として使用する光の波長などの共通のパラメータと，計算するホログラム面の座標によって行う。そのため，はじめに各ノード，各 SPE に共通のパラメータを与えておけば，その後は再送信する必要がなく，計算領域を割り振るだけでよい。

また，クラスタシステムを構築するにあたり，通常は通信時間などの遅延も生じる。しかし，Cell においては PPE で通

信, SPE で演算と完全に分離し, SPE で計算している間にデータを転送することで, 通信時間の隠蔽も可能である.

### 3 結果と考察

シングルコアの CPU を搭載した PC と Cell を搭載した PS3 において同様の CGH 計算を行った計算時間を比較する. 各場合の動作環境を表 1 に示す.

表 1.動作環境の比較

環境	PC	PS3
プロセッサ	Pentium4	Cell Broadband Engine
動作周波数	3.2GHz	PPU SPU とともに 3.2GHz
メモリ	DDR SDRAM 1.5GB	XDR RAM256MB
OS	Linux (Fedora 7)	Linux (Fedora 7)
コンパイラ	gcc 4.1.2	gcc 4.1.2

PS3 においては, 開発環境として Cell SDK3.0 を用いている. また, クラスタシステム構築の際に MPI を用いるために MPICH2 を導入した.

PC, PS3 両方で, コンパイル時には O3 オプションを用いて最適化している.

#### 3.1 CGH 計算時間の比較

Cell のクラスタシステムと PC で CGH 作成にかかった時間を表 2 に示す. ここで計測したのは, サーバで物体点データを処理した後, 各ノードにデータを転送する前から, 各ノードから一つのノードにデータを集めるまでにかかる時間である. また, CGH 計算を行うホログラムのサイズを 1920 × 1080 ピクセル(これは再生実験に用いた LCD のサイズである)とし, 物体点数は 11646 点(3.2 章で示す再生実験用に作成した三次元像の物体点数)とした.

表 2.PC と PS3 クラスタシステムでの処理時間(秒)

物体点数	PS3 のノード数					PC
	1	2	3	4	5	
128	0.30	0.20	0.18	0.16	0.16	6.98
512	1.12	0.61	0.45	0.36	0.32	27.85
2048	4.41	2.25	1.54	1.18	0.98	109.82
8192	17.55	8.82	5.92	4.48	3.62	455.94
11646	24.95	12.52	8.40	6.33	5.10	666.07

表 2 より, 物体点数が 11646 点の場合, 1 枚の CGH を生成するのに PC では約 11 分かかっていたのが, PS3 単体では約 25 秒. PS3 を 5 台用いたクラスタシステムでは約 5 秒で生成することに成功した. PS3 単体で PC に対して約 26 倍, PS3 を 5 台用いたクラスタシステムでは PC に対して約 130 倍高速化されている.

また, クラスタに用いた PS3 のノード数で見ると, PS3 単体に対して 2 台で 1.99 倍, 3 台で 2.97 倍, 4 台で 3.94 倍, 5 台で 4.89 倍高速化されている.

図 4 に PC に対する高速化比と物体点数, ノード数のグラフを示す.

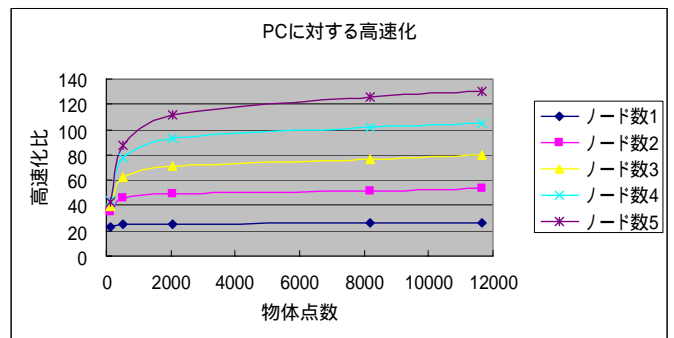


図 4.PC に対する高速化

図 4 から, 物体点数が多くなるほど高速化できていることがわかる. これは, 計算時間に対して通信時間の占める割合が非常に小さくなっているためである.

また, 今回のシステムでの通信時間は物体点データを各ノードに送るときと, 計算結果を一つのノードに集めるときの 2 回だけである. 図 5 に物体点データを各ノードに送るときにかかる時間のグラフを示す.

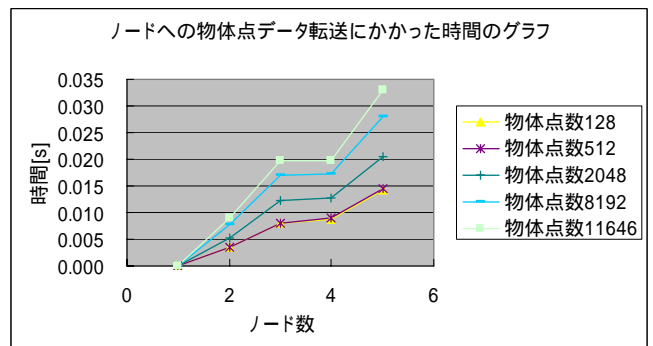


図 5.物体点データ転送時間

図 5 からわかるように, ノード数 3 と 4 では通信時間が変わらない. これは, ノード数が 2 の累乗間では通信の並列化により効率的に送れるためである.

図 6 に計算結果を一つのノードに集めるのにかかる時間の

グラフを示す。

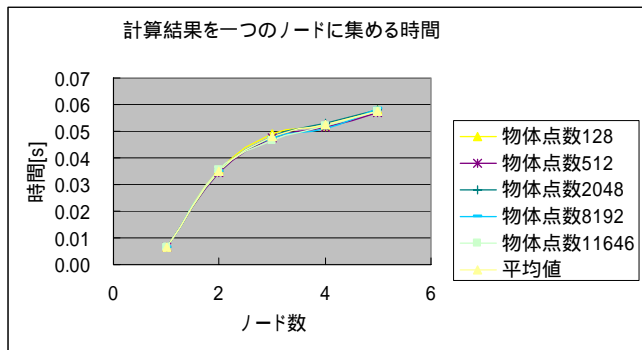


図 6. 計算結果を一つのノードに集める時間

計算結果を一つのノードに集める時間は、作成するホログラムのサイズとノード数に依存している。

ノード数が  $N$  台の時には、1 台あたりから転送するデータ量は  $X \times Y \div N$  byte ( $X$  は画像データの横、 $Y$  は画像データの縦のピクセルサイズ) となる。そして、サーバとなるノードからの転送は行われないので、システム全体としての画像データの転送量は  $(N - 1) \div N \times X \times Y$  byte となる。このことより、図 6 は理論値にほぼ等しいものといえる。

### 3.2 ホログラフィ再生

Cell のクラスタシステムを用いて計算した CGH を、実際に  $1920 \times 1080$  ピクセルの LCD を用いて再生した結果を図 7 に示す。



図 7. 物体点数 11646 点の再生像

図 7 は物体点数が 11646 点で構成された恐竜の再生像である。参照光源に赤色レーザーを用いて、デジタルカメラで撮影した。

CGH1 枚を生成するのに 5 秒程度の時間を要したが複雑な立体像に対しても、PC での計算と同じ再生像が得られ、精度のうえでも電子ホログラフィのシステムとして十分に活用できることを確認した。

## 4 まとめと今後の展望

PS3 単体で PC に対して約 26 倍高速化することができた。これは、Cell において 6 つの SPE の使用、SIMD 演算などの高速化手法を用いたためであると考えられる。

また、PS3 を 5 台用いたクラスタシステムでは PC に対して約 130 倍高速化することに成功した。

Cell に適したプログラムは、クラスタシステムにも非常に親和性が良いと言える。

今回構築したシステムでは、最後に計算結果を一つのノードに集めて表示させていたが、それぞれのノードから直接表示させるようなシステムを構成すれば通信時間の遅延はほぼ無視でき、PPE が通信、SPE が演算と完全に分離できるので、純粋にノード数に比例して高速化することも可能であると考えられる。

立体テレビを実現するためには、CGH をビデオレート (30fps) 程度で生成する必要があるため、今回のシステムをそのまま適用すると、物体点数 11646 点の恐竜の動画をリアルタイムで動かすためには PS3 を 800 台程度用いる必要がある。

実際には、さらなる最適化と開発環境の充実によって Cell の性能をより引き出すことが可能になると考えられる。

Cell を用いたクラスタシステムでは、今回実装した CGH の計算や、各種の解析システムなど膨大な計算量で並列処理可能なものに応用することにより高速化が期待できる。

特に既にクラスタシステムが構成されているものに関しては、より一層の高速化が可能であると言える。

### 参考文献

- [1] 株式会社ソニー・コンピュータエンタテインメント：“Cell Broadband Engine™アーキテクチャ version 1.01”
- [2] 株式会社ソニー・コンピュータエンタテインメント：“Cell Broadband Engine™アーキテクチャ用 C/C++ 言語拡張 version 2.3”
- [3] インプレスジャパン：“PLAYSTATION3 Linux 完全ガイド”
- [4] 共立出版 石川裕 佐藤三久 堀敦史 住元真司 原田浩 高橋俊行 著：“Linux で並列処理をしよう”
- [5] 培風館 P・パチエコ著 秋葉博記：“MPI 並列プログラミング”
- [6] フィックスターズ PLAYSTATION®3：