

Cell BEによる流体シミュレーションの高速化に関する研究

M2007MM008 堀田健太郎

指導教員：杉浦洋

1 はじめに

数値流体シミュレーションの手法の一つであるMPS(Moving Particle Semi-implicit)法 [1][2] は、砕波や複雑な水面の変化や、飛沫のような分離や合体を安定して計算することができるため、広い範囲で適用されている。しかしMPS法のような粒子ベースシミュレーションでは、大規模なシミュレーションにおいて計算に膨大な時間がかかる。

Cell Broadband Engine(Cell BE) プロセッサはマルチメディアデータに対する高速な演算を目的として設計された分散処理を指向したプロセッサである。また、Cell BE プロセッサは家庭用ゲーム機であるPlay Station3にも搭載されており、入手が比較的容易である。

本論文では粒子ベースシミュレーションの一つであるMPS法をCell BEプロセッサを搭載したPlay Station3を用いて実装し高速化を目指す。

2 MPS法の概要

MPS法は非圧縮性流れを解く有力な数値解析法の一つであり、流体を有限個の粒子で近似し、粒子を計算点として流体の数値的計算を行う粒子法の一つである。

非圧縮性流れの支配方程式を以下に示す。

$$\frac{D\rho}{Dt} = 0 \quad (1)$$

$$\frac{D\vec{u}}{Dt} = \frac{1}{\rho}\nabla P + \nu\nabla^2\vec{u} + \frac{\vec{F}}{\rho} \quad (2)$$

ここで、 ρ は流体の密度、 ν は動粘性係数である。式(1)は質量保存則であり、密度の時間変化がゼロ、すなわち、密度が時間に対して一定であることを意味している。式(2)は運動量保存則で、ナビエ-ストークス方程式と呼ばれる。右辺第1項は圧力勾配項、第2項は粘性項、第3項は重力項である。

2.1 重み関数

MPS法では、各微分演算子の離散化にそれぞれ粒子間相互作用モデルを用いる。粒子間相互作用モデルの構成において、相互作用の強さの指標として、次式の重み関数 ω を導入する。

$$\omega(r) = \max\left\{\frac{r_e}{r} - 1, 0\right\} \quad 0 \leq r < r_e \quad (3)$$

ここで r は粒子間距離である。したがって、式(3)の重み関数を用いると、粒子間距離がパラメータ r_e より短い距離でのみ粒子間で相互作用することになる(図1)。また粒子同士の距離がゼロに近づくと ω は無限大に近づく。

粒子 i およびその近傍粒子 j の位置ベクトルをそれぞれ \vec{r}_i, \vec{r}_j とする。粒子 i の位置における重み関数の和を

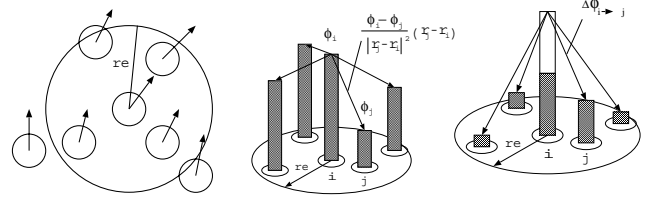


図1 粒子間相互作用

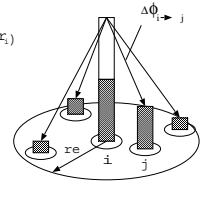
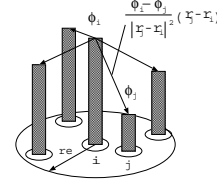


図2 勾配モデル
図3 ラプラシアンモデル

とったものを点 \vec{r}_i における粒子数密度として次式で定義する。

$$n_i = \sum_{j \neq i} \omega(|\vec{r}_j - \vec{r}_i|) \quad (4)$$

粒子1個の質量が一定で、かつパラメータ r_e が一定の場合、粒子数密度は流体の密度に比例する。非圧縮性流れでは流体の密度は一定であり、したがって粒子数密度も一定でなければならない。この一定値を n^0 とする。 n^0 は初期粒子配置において十分内側にある粒子、つまり重み関数の半径 r_e の中に自由表面がない粒子の粒子数密度を計算することで求める。解析中はこれを使いつづける。

2.2 勾配

MPS法では、粒子 i の位置における物理量 ϕ の勾配ベクトルに対して、次式の計算モデルを用いる(図2)。

$$\langle \nabla \phi \rangle_i = \frac{d}{n^0} \sum_{j \neq i} \left[\frac{\phi_j - \phi_i}{|\vec{r}_j - \vec{r}_i|^2} (\vec{r}_j - \vec{r}_i) \omega(|\vec{r}_j - \vec{r}_i|) \right] \quad (5)$$

ここで ϕ_i は粒子 i の有する物理量、 d は空間の次元数、 n^0 は初期粒子数密度、 ω は重み関数である。

2.3 ラプラシアン

MPS法では、粒子 i の位置における物理量 ϕ_i のラプラシアンのモデルを次式で与える。

$$\langle \nabla^2 \phi \rangle_i = \frac{2d}{\lambda n^0} \sum_{j \neq i} [(\phi_j - \phi_i) \omega(|\vec{r}_j - \vec{r}_i|)] \quad (6)$$

これは図3に示すように、粒子 i の物理量 ϕ_i の一部を近傍粒子 j に重み関数の分布で分配することを意味している。式(6)の λ は統計的な分散の増加を解析解と一致させるために導入する係数で、次式で与える。

$$\lambda = \frac{\sum_{j \neq i} |\vec{r}_j - \vec{r}_i|^2 \omega(|\vec{r}_j - \vec{r}_i|)}{\sum_{j \neq i} \omega(|\vec{r}_j - \vec{r}_i|)} \quad (7)$$

2.4 計算アルゴリズム

MPS法の計算アルゴリズムは半陰解法アルゴリズムであり、同時間ステップ内に陽解法と陰解法を含む。

全体の計算アルゴリズム

- 1: 初期パラメータ入力
- 2: 近傍粒子探索を含む粒子数密度計算
- 3: while シミュレーションを続行 do
- 4: 時間ステップを進める
- 5: 粒子の仮の速度, 仮の位置の計算
- 6: 近傍粒子探索を含む粒子数密度計算
- 7: 共役勾配法用の係数行列などの作成
- 8: 共役勾配法による圧力の連立方程式の計算
- 9: 粒子の速度の修正量の計算
- 10: 粒子の位置の修正
- 11: 粒子の位置の出力
- 12: end while

3 Cell BE プロセッサの概要

Cell BE プロセッサは, 汎用プロセッサコアである PowerPC Processor Element(PPE)1 つとマルチメディア処理に適した Synergistic Processor Element(SPE)8 つを Element Interconnect Bus(EIB) で接続した非対称なマルチコアプロセッサである [3].

3.1 PowerPC Processor Element

PPE は 64bit の PowerPC Processor Unit(PPU) に Cache を付属させたものである. PPE は汎用の処理ユニットであり, 主に SPE へのタスクの割り当てや制御を行う.

3.2 Synergistic Processor Element

SPE は Synergistic Processor Unit(SPU) と Memory Flow Controller(MFC) で構成されている. SPU には Single Instruction/Multiple Data(SIMD) 演算能力があり, 高密度な演算を行うことができる. SPU 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. また, SPU は LS 以外のメモリに直接アクセスできない. そのため, SPU がメインメモリのデータのロード/ストアを行う際は, MFC によるデータ転送を行う必要がある. メインメモリと LS とのデータ転送は Direct Memory Access(DMA) 転送によって行われる.

3.3 Direct Memory Access 転送

通常データの転送は, プロセッサ (PPE, SPE) が作業管理する. DMA 転送は MFC が作業管理するデータ転送であり, その間プロセッサに他の作業を割り当てることができる. DMA 転送が可能なデータサイズは, 1, 2, 4, 8byte 及び 16 の倍数 byte であり, 一度に最大で 16KB まで転送することができる. また, DMA 転送ではデータが 16 の倍数 byte 境界に整列されている必要があり, 128byte 境界で整列されているとき最大効率を発揮する.

3.4 Mailbox

MFC が提供する DMA 転送以外の PPE と SPE の通信機能として Mailbox によるデータの受け渡しがある. DMA 転送が最大 16Kbyte であるのに対し, Mailbox では 32bit のデータの受け渡しを行うことができる. その

ため, PPE や SPE の状態やパラメータなどの比較的サイズの小さいデータを転送するのに適している.

3.5 Signal Notification Register

MFC が提供するもう 1 つの通信機能として, Signal Notification Register(SNR) がある. SNR では, バッファ完了フラグなどのシグナルを任意の SPU へ PPE や他の SPU から送信する. SNR は SNR1 と SNR2 の独立した 2 つのレジスタが提供されている. 送信データは 32bit のシグナルである. SNR の受信モードには, 上書きモードと論理 OR モードの 2 種類がある. 1 対 1 の通信では上書きモード, 多対 1 の通信では論理 OR モードを利用すると効果的である.

Mailbox は PPE と SPE の通信に利用されるのに対して, SNR は SPE 同士の通信にも利用される. そのため SNR は複数の SPE 間での同期などに効果がある.

4 高速化手法

4.1 近傍粒子探索

MPS 法では粒子間相互作用モデルを用いる際に, 重み関数を計算する必要がある. しかし単純に重み関数を全粒子同士に用いると, 粒子数 n に対して $O(n^2)$ の計算コストがかかる.

式 (3) より, 重み関数は粒子間距離がパラメータ r_e 以上離れるとゼロとなる. このため, r_e 以上離れた粒子同士の相互作用は無視することができ, 計算コストを削減することができる. 本論文では領域をセルと呼ばれる格子状の小領域に区切る手法を用いる. セルの一辺は粒子間相互作用の影響範囲 r_e とする. 粒子は座標によりいずれかのセルに対応づけられ, 各セルの隣接セル情報を用いることで, そのセルに含まれる粒子の近傍粒子を限定することができる (図 4). この手法では, 粒子同士の平均隣接数 m に対して $O(mn)$ の計算コストとなる.

4.1.1 ハッシュの導入

時間ステップごとに粒子は移動するため, 一つのセルに対応づけられる粒子の数は必ずしも一定にはならない. 解決法として, 粒子の座標をハッシュに変換し格納する方法がある [4]. この方法は粒子とセルの座標からハッシュ値を計算し, いくつかの一次配列によってセルと粒子の関係を表現するものである. ハッシュテーブルの容量は増減がなく一定であるため, 無駄なメモリを保持することもなく途中でのメモリ確保する必要もないため効率がいい. 本論文ではこの手法を用いる.

4.1.2 処理の分割単位

近傍粒子探索を行う際, 処理は隣接したセル同士に限られるため, それを処理単位として考える. 図 5 のように処理単位を考えた場合, 周囲のセルは中心のセルの計算のために読み込まれる. しかし, 周囲のセルの計算は行われず, 周囲のセルの読み込み回数が重複するため効率が悪い. 重み関数は近傍粒子のペアに対して対称だから, 計算結果を両方の粒子に適用することが出来る (図 6). 計算に必要なセルは 2 次元では周囲 8 マスのうち 4

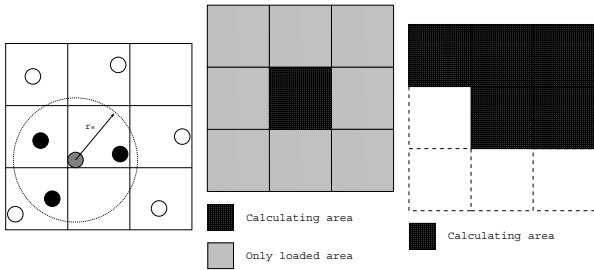


図4 セルを用いた近傍粒子探索

図5 分割の処理単位1

図6 分割の処理単位2

マスであり、計5マスを処理単位とし、これをワークブロックと呼ぶ。

4.1.3 タスク割り当て

一つのワークブロック内の近傍粒子探索にかかる計算コストは、中心セル内の粒子と隣接セル内の粒子の個数の関数である。これを利用して、PPEは各SPEに計算コストが均等になるようにワークブロックに分配する。

ワークブロックに含まれる粒子の情報はメインメモリにそれぞれ離れて格納されている。それらを個別にDMA転送を行うと通信のレイテンシの影響が大きくなる。そのため、DMAリスト転送を使い、データを収集転送する。DMAリスト転送は、転送データのアドレスリストにより、データを収集し一括転送するCell BEの機能である。

4.2 連立方程式の解法

MPS法では、圧力勾配項を求めるために連立方程式を解く必要がある。圧力計算において、ある一定距離離れた粒子から受ける圧力は重み関数によって0になる。よって係数行列はその要素の多くが0である、つまり疎行列である。疎行列に効果的な手法として共役勾配法がある。

共役勾配法の計算で計算量が多いのは行列とベクトルの積である。この部分を並列化していくこととなる。またベクトルの内積計算も並列化することで計算時間を短縮することが出来る。

5 実験結果と考察

本論文では、水柱の崩壊の問題をCell BEを搭載したPlayStation3に実装した。水柱の崩壊の問題はダム崩壊問題とも呼ばれ、自由表面流れ解析ではもっとも代表的なベンチマーク問題である。出力は、描画は行わずに各粒子の持つ座標などの変数をファイルに出力することとする。総粒子数は1122とし、シミュレーション時間が1.0秒を越えた時点(約3000ステップ)で計算を終了する。なお各パラメータや粒子の持つ変数の初期値は外部ファイルから読み込むものとする。

出力データを描画した結果、実際の物理現象に近い結果となった(図7)。ここで図内の数値はシミュレーション開始からの時間(sec)である。

実験の結果、全体として約3.5倍の処理向上を得ることができた。ここで高速化を行う前のオリジナル、つまり

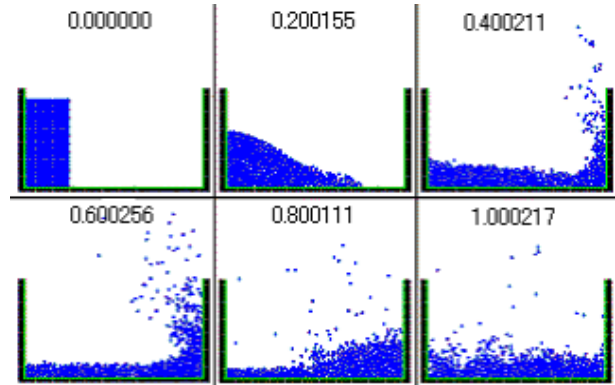


図7 水柱の崩壊の問題 描画結果

近傍粒子探索は全探索で行い、SIMD演算および並列化を行わずにPPEのみで実行した場合と高速化後の各タスクの占める処理時間の割合を比較したものを表1に示す。なお、この時の1サイクルの平均実行時間はオリジナルが237.113msec、高速化後(SPE6個)が68.284msecだった。このようにオリジナルでは支配的であった近傍粒子探索と共役勾配法の割合が高速化後には小さくなっていることがわかる。

表1 オリジナルと高速化後のプログラム処理時間の割合

タスク内容	処理時間の割合 (%)	
	オリジナル	高速化後
近傍粒子探索を含む 粒子数密度計算	64.01	6.47
共役勾配法用の 係数行列などの作成	6.15	31.24
共役勾配法による 圧力の連立方程式の計算	18.85	6.62
粒子の速度の修正量の計算	5.03	25.56
その他	5.96	30.11

次に1サイクルの処理時間をSPE数によって比較したものを図8に示す。ここで横軸はオリジナルとSPE数、縦軸は処理時間(msec)である。表1からもわかるように、SPE数が増えるにつれて並列化を行っていない部分の処理時間が支配的になってくるため、SPE数を増やしても大きな処理向上を得られていない。そのため、他の部分についても並列化を行うことが不可欠であると考えられる。また、全体のうち並列化した部分の並列効率はSPE数6個で約70%となった。より高い並列効率を得るために、近傍粒子探索と共役勾配法それぞれの処理時間について考察していく。

5.1 近傍粒子探索

本論文では近傍粒子リストの作成を粒子数密度の計算において同時に行っているため、粒子数密度の計算を含めた処理時間を高速化を行う前のオリジナルの処理時間と比較した(図9)。ここで横軸はSPE数、縦軸は処理時間(msec)であり、実線の折れ線グラフは実験においての

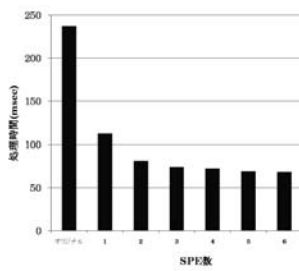


図 8 プログラムの 1 サイクルの処理時間

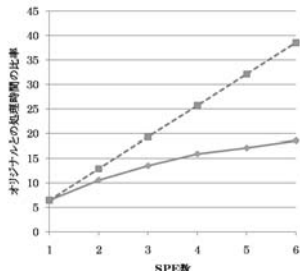


図 9 近傍粒子探索を含む粒子数密度計算の処理時間

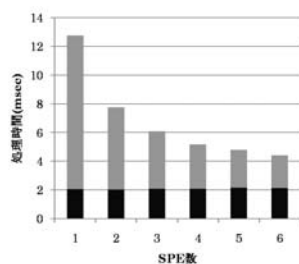


図 10 ハッシュテーブル作成と粒子数密度計算の処理時間

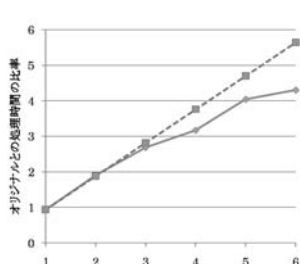


図 11 共役勾配法の処理時間

処理時間を示し、破線の折れ線グラフは並列効率 100%の場合の処理時間である。図 9 のように全探索の手法と比べてハッシュテーブルを用いた手法は約 18.6 倍の処理向上が見られた。また SPE 数を最大数の 6 個用いて並列化することで SPE 数 1 個と比べて約 2.9 倍の処理向上が見られた。並列効率があまり高くない理由として次のことが考えられる。ハッシュテーブルの作成と粒子数密度の計算それぞれの処理時間を図 10 に示す。ここで横軸は SPE 数、縦軸は処理時間 (msec) の積み上げグラフであり、下の棒グラフがハッシュテーブル作成時間を示し、上の棒グラフが粒子数密度計算の処理時間である。

このように粒子数密度の計算では約 5 倍の処理向上がみられ比較的高い並列効率を得ることができているが、ハッシュテーブルの作成部分では SPE 数にかかわらずほぼ一定となっている。これはハッシュテーブルの作成において、ハッシュ関数の実行は SPE で並列処理し、ハッシュテーブルの作成 (ソート部分) は PPE で処理しているので、PPE での処理時間の占める割合が大きく、並列効率が落ちたものと考えられる。対策としてハッシュテーブルの作成にマージソートなどを用いて並列化することが考えられるが、必要なデータの転送や同期などの関係から工夫が必要であると考えられる。

5.2 共役勾配法

共役勾配法部分の処理時間は図 11 のようになった。ここで横軸は SPE 数、縦軸は処理時間 (msec) であり、実線の折れ線グラフは実験における処理時間を示し、破線の折れ線グラフは並列効率 100%の場合の処理時間である。このように SPE 数 1 個と比べて約 4.6 倍の処理向上を得ることが出来た。ここで並列化なしの処理時間よ

りも SPE 数 1 個の処理時間の方が長くなってしまっている。これは PPE と SPE 間で同期を行う部分でのコストが原因であると考えられる。本実験では 1 ステップにおける共役勾配法の反復数は平均 21 回であり、1 回の反復で 2 回の同期を行う。そのため ICCG 法などの前処理付き共役勾配法を用いることで反復回数を減らすことでより高い並列効率を得ることが出来ると考えられる。しかし、ICCG 法などは前処理部分においてデータの依存性が高く並列化に向いていないので、前処理部分が支配的になり並列効率が逆に落ちる可能性がある。そのため並列効率の高い前処理を行う必要がある。

6 今後の課題

本論文では粒子法の一つである MPS 法を用いて水柱の崩壊の問題を Cell BE プロセッサを搭載した PlayStation3 に実装することで高速化を目指し、実際に近傍粒子探索部において約 18.5 倍、共役勾配法部において約 4.4 倍、プログラム全体として約 3.5 倍の速度向上を得ることが出来た。

しかし、並列効率は SPE 数 6 個で 70%であり十分とはいえない。特にハッシュテーブルの作成におけるコストを削減する工夫が必要である。また PPE と SPE 間の同期による待ち時間を削減するために、より並列化に適した前処理付きの共役勾配法を用いることで、さらに並列効率を高めることができるものと考えられる。

また支配的であった近傍粒子探索と共役勾配法の計算時間が並列化などにより短縮されたので、他の部分の計算時間が支配的となった。さらなる高速化のために、共役勾配法用の係数行列の作成や速度の修正量の計算の並列化についての研究も進めていきたい。

参考文献

- [1] 越塚誠一：数値流体力学。培風館 1997。
- [2] 越塚誠一 日本計算工学会：粒子法。丸善 2005。
- [3] Sony Computer Entertainment Inc: Cell Broadband Engine Architecture. <http://cell.scei.co.jp/>。
- [4] Nils Hjelte: Smoothed Particle Hydrodynamics on the Cell Broadband Engine. Master's Thesis in Computing Science, 2006。