

MPS法による3次元の水柱崩壊シミュレーションの高速化とその評価 Performance Improvement and Evaluation of collapse simulation of a water column by MPS method

北山 雅邦[†] 黒田 久泰[†]
Masakuni Kitayama Hisayasu Kuroda

1. はじめに

本研究ではMPS法により3次元の水柱崩壊シミュレーションを行うプログラムを作成し、そのプログラムをOpenMPを用いて並列化することで高速化した。そしてそのプログラムの性能評価を行った。

2. MPS法

MPS法は、粒子を用いた連続体の数値解析手法である粒子法の1つである。

粒子法では格子を用いずに微分方程式を離散化する必要がある。MPS法では、粒子間相互作用モデルを用いて、勾配、発散、回転、ラプラシアンといった微分方程式を離散化する[1][2]。

粒子間相互作用モデルには次の重み関数 ω を利用する。

$$\omega(r) = \begin{cases} \frac{r_e}{r} - 1 & 0 \leq r \leq r_e \\ 0 & r_e \leq r \end{cases} \quad (1)$$

ここで r は粒子間距離であり、図1のように粒子間距離がパラメータ r_e より短い場合に相互作用するようになる。

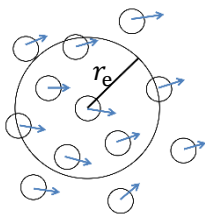


図1 粒子間相互作用モデル

3. 近傍粒子探索

3.1 領域分割

MPS法では、注目粒子の相互作用半径の内側にいる粒子を全ての粒子の中から探さなければならない。これを注目粒子と他の全ての粒子との距離を調べながら行うのでは時間がかかってしまうので、図2のように粒子のある領域を格子状に分割し注目粒子のあるグリッドの周りのグリッド内にある粒子のみを調べることにより探索時間を短縮することができる。

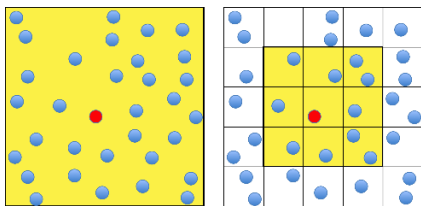


図2 近傍粒子探索

3.2 近傍粒子探索の並列化

近傍粒子探索の並列化を行うとき、粒子がどの領域にあるかを配列に書き込むときに複数のスレッドが同時にその配列にアクセスすることになる。そのときデータの不整合が起きないようにそれぞれのスレッドが配列に書き込むときにその配列にロックをかける必要があるが、そうした場合処理時間が大幅に低下してしまう。そこで図3のようにスレッドごとに配列を用意し、近傍粒子を探索する際には全ての配列を調べるようにすることで並列化効率を上げることができる。

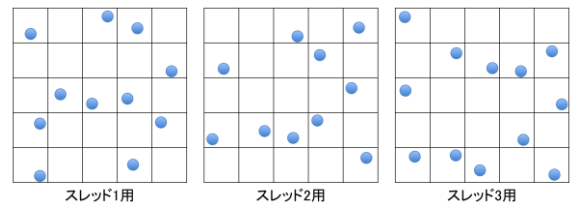


図3 近傍粒子探索の並列化

4. 実験

3次元の水柱崩壊シミュレーションを行うプログラムを作成し、粒子数を11508, 16525, 22032, 29868, 38340, 48108, 59244, 71820, 85908と変えながらそれぞれについてスレッド数を1, 2, 4, 6, 8, 10, 12と変えながら0.2秒間のシミュレーションを行った。

MPS法のプログラムでは、連立一次方程式を解く必要があるが今回作成したプログラムでは反復解法の一つである共役勾配法を用いた[3]。

4.1 計算機環境

本実験に使用した計算機の環境を表1に示す。

表1 計算機環境

CPU	Intel Xeon E5-2630×2
コア数	6×2
動作周波数	2.3 GHz - 2.8 GHz
キャッシュサイズ	15 MB
メインメモリ	64 GB
OS	Ubuntu 13.04
gcc	ver 4.7.3
コンパイルオプション	-O3 -fopenmp -lm

4.2 パラメーター設定

本実験に使用したパラメーター一覧を表2に示す。

表2 パラメーター設定

重力加速度	9.8 m/s ²
圧力計算の収束条件	1.0E-9
初期配置の粒子間距離	8.0E-3

[†] 愛媛大学大学院理工学研究科,
Graduate School of Science and Engineering
Ehime University

5. 結果

スレッド数ごとの粒子数と実行時間の関係を図4に示す。

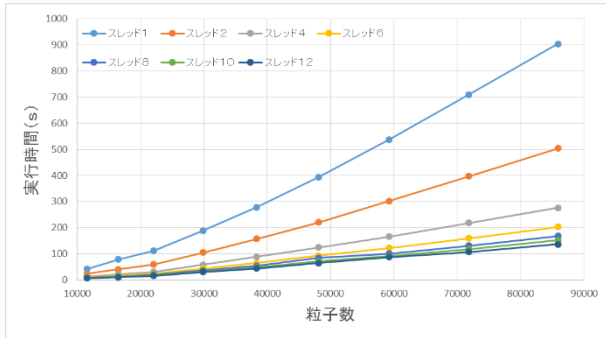


図4 スレッド数ごとの実行時間

図5に逐次で実行したときと比べて並列で実行したときの全体の速度向上率を示し、図6, 7, 8に近傍粒子探索、共役勾配法、行列作成の部分の速度向上率を示す。

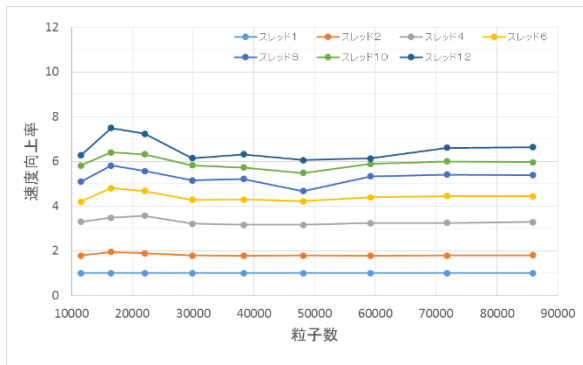


図5 速度向上率 (全体)

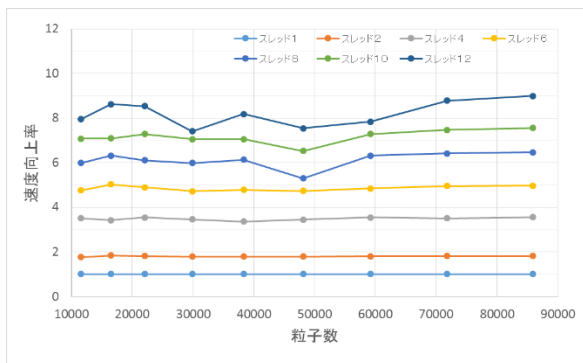


図6 速度向上率 (近傍粒子探索)

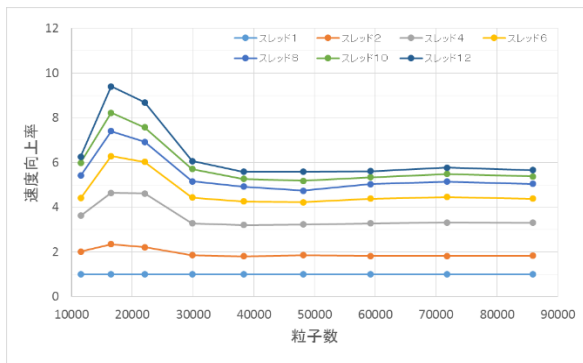


図7 速度向上率 (共役勾配法)

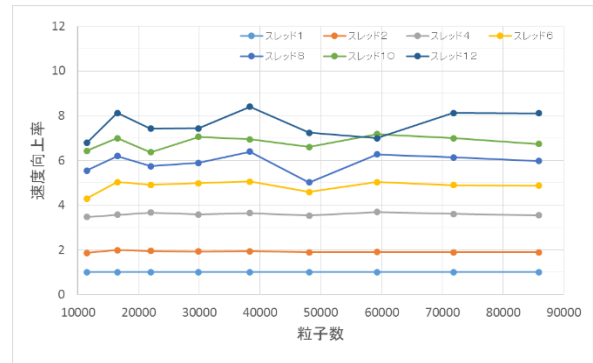


図8 速度向上率 (行列作成)

最後に粒子数を 85908 としたときの詳細な実行時間を図9に示す。

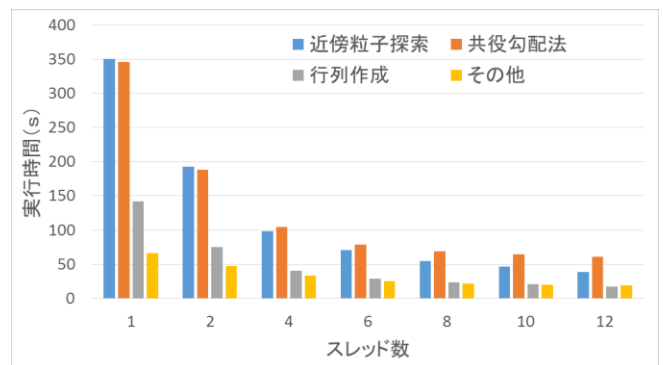


図9 実行時間 (粒子数: 85908)

6. まとめ

図4よりどのスレッド数で実行しても実行時間は粒子数に比例して増えていっていることが分かる。またスレッド数が多いほど実行時間が短いことから並列化により高速にシミュレーションが行えていることが分かる。図5から逐次で実行した場合と比べてスレッド数12で実行した場合には平均で6.5倍の速度で計算が終了していることが分かる。その内訳を見てみると、図6, 7, 8から分かるように近傍粒子探索では平均して8.2倍、共役勾配法では6.4倍、行列作成では7.6倍速度が向上している。図7において粒子数16524のときに速度向上率のピークがある。これは共役勾配法はある程度までは粒子数が多いほど並列化効率が高がること、また粒子数16524より多くなってくると行列のサイズが大きくなりキャッシュに収まりきらなくなったことが原因だと考えられる。

図9の粒子数85908の実行時間を見ると分かるようにこの3つの部分が全体の大部分を占めておりこの部分を高速化できたことが全体の高速化につながった。

参考文献

- [1] 越塚 誠一 “粒子法の流れの数値解析”, 日本流体力学会誌 21(3), pp.230-239 (2002).
- [2] 越塚 誠一 “粒子法”, 丸善出版 (2005)
- [3] 戸川 隼人, “共役勾配法”, 教育出版 (1977)