

MPI を用いたハードディスクドライブシミュレータプログラムの並列化

高橋 聡† 藤井 昭宏† 吉田 和悦† 小柳 義夫†

工学院大学†

1. はじめに

近年, ブロードバンド通信の普及やデジタルコンテンツの増加で, 記録媒体の取り扱うデータ量が増加している. それに伴い, ハードディスクドライブ (以下, HDD) に求められるデータ容量も増加している. HDD はデータ容量を増加させるために高記録密度化していった. しかし, HDD の記録密度の上昇によって, 熱揺らぎ問題やヘッドの低浮上化などさまざまな課題が発生している. 課題を解決するためには, ナノメートル領域における HDD の物理シミュレーションが必要である. しかし, 厳密なシミュレーションには多大な計算時間がかかる.

本研究の HDD シミュレーションプログラムは, 静磁界計算の部分で畳み込み計算が行われており, プログラム実行時間の 80% を占めている. HDD シミュレータプログラムの畳み込み計算について, データアクセスの効率化と, MPI を用いた並列化による高速化の結果を報告する.

2. 計算モデル

本研究の HDD シミュレーションプログラムの主要な部分は, マイクロマグネティクスに基づく計算である. マイクロマグネティクスとは, ナノメートル領域の磁気的相互作用を物理法則に従い計算する原理である. このプログラムで求める有効磁界は, 外部磁界, 交換磁界, 異方性磁界, 静磁界の 4 種類の磁界の和によって成る. 有効磁界の計算領域を図. 1 のような立方体型のセルで等間隔に離散化する. 外部磁界, 交換磁界, 異方性磁界が磁化に与えるエネルギーを求めるときの計算回数がセル数 N に比例して増えるのに対して, 静磁界が与えるエネルギーは, セル数 N の 2 乗に比例して増える. そのため, 計算時間の大部分は, 静磁界計算に費やされる.

なお, 本プログラムでは畳み込み計算の計算量を

減らすために, 高速フーリエ変換 (以下, FFT) を用いている. x 軸方向に FFT を行った後, y 軸方向に FFT を行うことで計算量を大幅に減らし, 計算時間の短縮に成功している. しかし, z 軸方向はセルの厚さが不均一の場合や空のセルが入っている場合などのプログラムも考慮して FFT を用いていない.

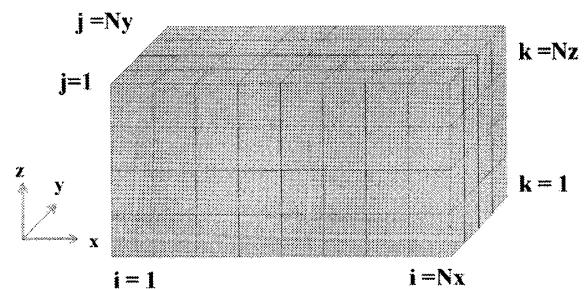


図 1. 静磁界の離散化モデル

3. 高速化手法

藤沢らの研究 (1) では, MPI と OpenMP による並列化を行っているが, データ構造の変更を行っていない. 本研究では効率的なデータアクセスを行い, そのうえで並列化を行うことにより性能改善を目指す.

3-1. データ構造の改良

プログラムの問題点

- A, 1 つのセルに付き 6 種類の配列を用いて畳み込み計算を行っている.
- B, y 軸方向に連続になっている配列データを用いて z 軸方向の畳み込み計算を行っている. そのため, 配列データにアクセスするとき, データに連続アクセスが出来ず計算時間が伸びていると考えた.

解決方法

- A, 複数の配列を 1 つにまとめる 1 配列化を行う. 6 種類の配列を 1 つにまとめ, アクセスする配列を少なくすることで, データアクセスが効率的になる. また, 配列をまとめることで 4 次元配列になるので, セルの座標を表す 4 次元配列を 1 次元配列に変更する.
- B, A に加えて, z 軸方向のセルが連続に並ぶように配列の順番を変え, それに合わせて

Parallelization of Hard Disk Simulator Program on MPI

†Satoshi Takahashi

†Akihiro Fuji

†Kazuetsu Yoshida

†Yoshio Oyanagi

†Kogakuin University

for 文を変更する。

以上の 2 つデータ構造の改良を行うことで、データアクセスの無駄を無くし計算時間を短縮する。

3-2. 並列化

C, データ構造の改良 A, B を行ったプログラムに加えて, 計算時間の最もかかる静磁界の畳み込み計算部分を MPI を用いて並列化をする。データアクセスの効率を上げる改良を行った後に, 並列化を行うことで並列化の効率も上げる。並列化の方法は, 静磁界の畳み込み計算の計算領域を y 軸方向に分割する。分割した領域を各プロセッサが計算し, その結果を MPI_Allgather 関数を用いてデータをまとめる。計算部分以外は重複して実行する。HDD シミュレーションプログラムの規模が大きくなった時, メモリ不足の問題が発生するので, それを考慮してプロセッサの数を増やすことで大規模化に対応させる。

4. 計測結果

高速化手法で示した解決方法を行ったプログラムをそれぞれ A, B, C, 元のプログラムを 0 とする。プログラム 0 とプログラム A, B, C の畳み込み計算部分の実行時間とプログラム全体の実行時間を比較する。0 の実行時間を 100%とした時の A, B, C の実行時間の短縮率を図 2. のグラフに示す。

開発・実行環境は以下の通りである
Intel Xeon プロセッサ 2.93GHz
メモリ: 1GB
コンパイラ: GNU Compiler 3.2.3-54
並列計算時は同機を 16 台並列で実行

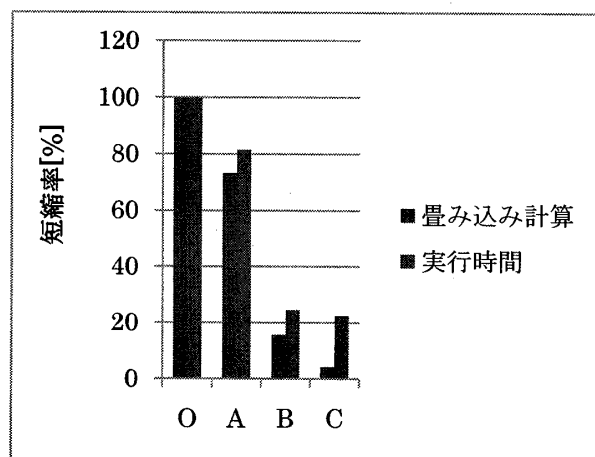


図 2. 計測時間の比較

1 配列化と 1 次元配列化のデータ構造の変更を行った A では約 80%, データに連続アクセスさせる改善を行った B で 25%まで実行時間を短縮した。主要な計算部分である静磁界の畳み込み計算も A で 73%, B で 15%, C に至っては 5%まで短縮した。

5. 考察

計測結果より, データ構造の改良とデータアクセスの改良を行った A, B のプログラムは, 実行時間の大幅な短縮に成功した。また, 1 配列化と 1 次元配列化により容易に MPI による並列化を行えた。MPI による並列化を行った C のプログラムの静磁界の畳み込み計算部分の計測時間も大きく短縮した。しかし, C の実行時間は, 畳み込み計算部分ほど短縮率は大きくなかった。その原因として, 畳み込み計算の実行時間は減ったが MPI_Allgather によるデータの通信に時間がかかりプログラム全体で見ると実行時間が短縮出来なかったと考えられる。

データ構造の改良とデータアクセスの効率化を行うことで, 性能改善に至った。また, データに連続アクセスさせるようにデータ構造を改良することで, 大幅な実行時間の短縮になる。

6. おわりに

本研究では HDD シミュレーションプログラムを高速化するために, データ構造の改良と MPI による並列化を行った。その結果, 大幅な計算時間の短縮を行うことに成功した。

今後の課題として, データの転送を工夫することで実行時間の短縮を図る。また, 静磁界の畳み込み計算部分だけでなく, 他の部分で計算領域の分割を行えるところに並列化を施すことで, 実行時間の短縮を目指す。

7. 参考文献

- (1) 藤澤明信「マイクロマグネティックシミュレータの高速化」, 日本磁気学会学会誌 Vol. 33 (2009) No. 3 pp. 189-192
- (2) 中田敏幸, 青木健一郎, 古屋篤「HDD におけるナノシミュレーション技術」
<http://img.jp.fujitsu.com/downloads/jp/jmag/vol58-1/paper05.pdf>