

非均質 PC クラスタによる 3 次元流体解析プログラムの並列化

鬼澤 亮[†]石黒 美佐子[‡]茨城大学大学院理工学研究科[†]茨城大学工学部[‡]

1. はじめに

本研究では、安価な PC を複数台用いて流体計算を行う。スーパーコンピュータは利用できる人が限られてしまうが、研究室等にある各種 PC を利用し並列計算を行えば、多くの人により高性能な演算能力を持った環境を得ることができる。流体解析プログラムでは 3 次元非圧縮粘性流れの計算を対象としている。ネットワークはふだん研究室で使用している 1000BASE-T の Ethernet を使用し、通信には MPICH for Windows を用いる。ここでは計算の高速化に着目して、非均質 PC を最大 8 台まで用いて数値実験を行い性能評価する。さらに、PC がワープロ、音楽・動画の再生などで利用中の場合の並列化も試す。

2. 3 次元非圧縮粘性流れ

計算対象は 3 次元 cavity 流れとする。基礎方程式は連続の式と Navier-Stokes 方程式から成る。

$$\nabla \cdot \mathbf{v} = 0 \quad (1)$$

$$\rho \frac{d\mathbf{v}}{dt} + (\mathbf{v} \cdot \nabla) \mathbf{v} = -\nabla p + (1/Re) \nabla^2 \mathbf{v} \quad (2)$$

ここで $\mathbf{v}$, p , Re はそれぞれ流速、圧力、レイノルズ数を表す。

計算スキームには擬似圧縮法を用いる。擬似圧縮法は定常状態を解く時に有効である。連続の式に関しては、擬似的に時間発展型方程式とする：

$$\partial p / \partial t = \delta \nabla \cdot \mathbf{v} \quad \delta: \text{擬似圧縮性係数} \quad (3)$$

$\partial p / \partial t$ が十分小さくなるまで反復計算する。方程式はスタガードメッシュを用いて差分化される。2×1×1 の計算領域を 60×30×30 の格子に分割し、 $Re=1000$ 、時間刻み $\Delta t=0.001$ として計算を行う。境界条件は上面がスリップ、他の面はノンスリップとする。

3. 計算領域の分割による負荷分散

計算領域は x 軸で分割される。非均質 PC クラスタであることを考慮し、計算速度に応じて分担させることにする。また、CPU クロック数のみでは計算速度を測りきれないので、実際の計算速度から計算領域の割り当てを決める。

使用する PC の仕様を表 1 に示す。

表 1 PC の仕様

PC のタイプ	A	B	C	D
CPU[GHz]	Pentium4	Pentium4	Athlon	Athlon
	2.8(1.00)	2.0(0.71)	1.1(0.39)	0.85(0.30)
キャッシュメモリ [KB]	512	512	256	512
メモリ[MB]	512	384	384	256
台数	3	3	1	1
実測計算速度比	1.00	0.74	0.51	0.40

3 通りの計算領域の分割法を試みる。

CPU クロック数による静的領域分割：開始時に各 PC の CPU クロック数に応じて分割する。

$$\text{担当格子数} = l_{\max} \times c_i / \sum_{i=0}^n c_i \quad (4)$$

c_i ：CPU クロック数， n ：使用 PC 数，

$l_{\max}$ ： x 軸の全格子数

実測計算時間による静的領域分割：CPU クロック数で分割し、計算を開始する。1000 反復ステップ経過後に実測計算時間から領域の再分割を行う。

動的領域分割：実測計算時間から 1000 ステップ毎に領域の再分割を行う。

表 1 に示すタイプ A の PC を基準とし (PC₀ とおく)，合計 8 台まで並列に計算する。

通信を考慮しない場合の理想的な速度向上を理論値と呼び、実測計算速度比より算出する。本数値実験ではこれを性能評価の基準とする。

4. 通信時間の削減

並列計算を行うと、1 回の反復計算の中でデータ交換のための通信と収束判定を行うための通信が必要となる。ここでは反復回数が増加し過ぎない範囲で通信の削減を行う。数値実験の結果、この問題の場合には 5 ステップに 1 回通信を行えば良いことがわかった。

CPU クロック数による領域分割と実測計算時間による領域分割での並列化効率について図 1 に示す。このときの PC クラスタの構成は非均質性を重視し、A, A+B, A+B+C, A+B+C+D, 2A+B+C+D, 2A+2B+C+D, 3A+2B+C+D,

Parallelization of a 3D CFD Program Based on the PC Cluster Composed of Heterogeneous PC

[†] Graduate School of Science and Engineering, Ibaraki University

[‡] Faculty of Engineering, Ibaraki University

3A+3B+C+D の順に接続台数を上げた．4，5 台の所で差が顕著に現れているのは Athlon の PC が加わっており，この PC の計算速度が CPU クロック数に比べて大きいことによる．

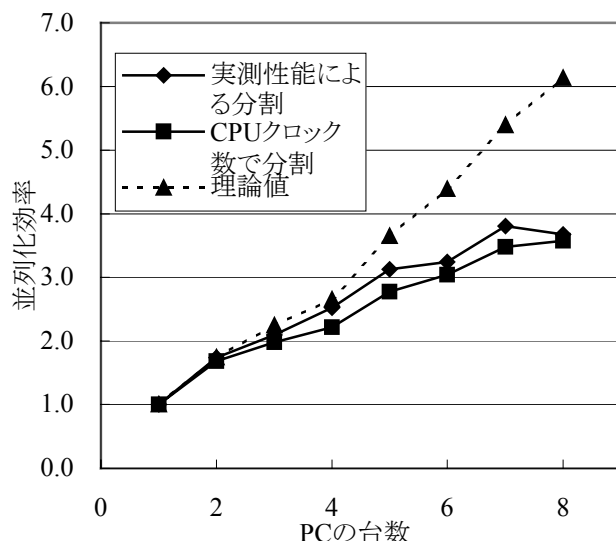


図1 領域分割法による並列化効率

5. 空き時間を利用した並列計算

PC にワープロなどの負荷がかかっている状態で，流体解析プログラムをバックグラウンドで実行する場合の並列化効率について評価する．CPU ビジー率はタイプ A の PC においてワープロで約 3%，音楽ファイル再生で約 5%，動画ファイル再生で約 15%である．この空き時間を流体解析プログラムに利用する．ここでは図 1 と同じ PC クラスターの構成を用いるが，タイプ A の PC 3 台にワープロ，音楽ファイル再生，動画ファイル再生の負荷をかけた状態での並列計算を行う．このとき音楽ファイル再生と動画ファイル再生では静的分割法，ワープロでは動的分割法を採用する．

負荷の違いによる並列化効率を図 2 に示す．音楽再生では負荷を考慮した理論値の約 70%，動画再生では約 62%の速度向上率が得られた．それぞれ 6 台，7 台の辺りから並列化効率の伸びが無くなっている．これは擬似圧縮法では 1 回の通信で u, v, w, p の 4 種類のデータの扱うため，通信がボトルネックとなっているからである．そのため，あまり台数を増やしても効果は望めない．また，MPICH for Windows では PC 固有の負荷を妨げないように実行優先度が低く設定されている．空き時間を利用した並列計算において，PC 固有の負荷の割合が大きくなると，流体解析プログラムへの CPU 割り当てが抑制され CPU に空きが生じることがある．音楽再生の

バックグラウンドでの並列計算は，負荷が無い状態での並列化効率に近い結果が得られた．音楽再生のバックグラウンドでの並列計算は，負荷が無い状態での並列化効率に近い結果が得られた．これは 5%程度の負荷では，MPICH の通信オーバーヘッドにより CPU の空き時間内に収まってしまうためである．

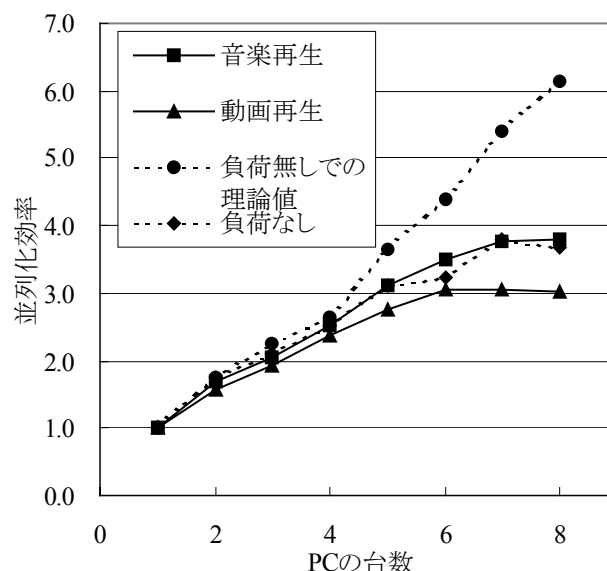


図2 負荷の違いによる並列化効率

6. まとめ

MPICH for Windows を用いた 3 次元非圧縮粘性流れの並列計算を行うことができた．8 台まで並列化し，負荷無しの状態では理論値の約 74%の並列化効率を得られた．計算領域を PC の実測計算速度に応じて分割することにより，PC ごとの負荷バランスのとれた並列計算が行えた．

CPU の空き時間を利用した並列計算を行い，音楽再生では負荷を考慮した理論値の約 70%，動画再生では約 62%の並列化効率を得られた．

今後の課題としては，SMAC 系のスキームによる計算を行う．このスキームでは 1 回の反復の中で圧力についてのみ通信を行えばよいため，通信時間の削減が見込める．また，通常の優先度で実行できる他の通信ライブラリによる計算などが挙げられる．

参考文献

- [1] Argonne National Laboratory:
<http://www-unix.mcs.anl.gov/mpi/mpich/>
- [2] 水野明哲：流れの数値解析入門，朝倉書店，1995
- [3] 日本機械学会：流れの数値シミュレーション，コロナ社，1988
- [4] 保原，大宮編：数値流体力学，東大出版会，1992
- [5] 繪幡，石黒：空き時間を利用した WindowsPC クラスタ上での流体解析プログラムの並列化，第 16 回数値流体力学シンポジウム，