

大規模 LES における圧力反復計算の並列処理効率について

神戸大学工学部建設学科
神戸大学大学院自然科学研究科

学生員 ○竹田 広希
正会員 中山 昭彦

1. はじめに

実験的手法では時間や費用が問題となり、また再現困難な相似則も容易に設定できることから CFD が流体力工学で用いられてきた。実際の現場では流体の計算を行う場合、一定の期間内に計算を終わらせる必要があるが、一方で複雑な形状の対象物を解像するために多くの格子点を取り、大規模な演算が必要となる。その大規模演算の計算負荷に対しコンピュータの進化が追いついておらず、1つの CPU を用いた計算では計算時間がかかりすぎてしまい大きな計算領域を計算できていないのが実情である。そこで複数の CPU を同時に使用して1つの計算を行う並列化と呼ばれる手法が考えられる。並列計算において、計算領域は各 CPU が計算を担当する領域に分割され、CPU は自分の担当する領域のデータのみを持ち計算を行う。領域の境界面では MPI と呼ばれる関数を用いて必要なデータのやりとりが行われる。

水工学の分野では牛島ら¹⁾²⁾が単純な境界内の基礎的な流れの計算を行いその効果を検証している。近年計算機は更に発展し、本格的乱流シミュレーションが以前に比べ低コストで行えるようになってきている。そこで本研究では実用的乱流シミュレーション法である LES 計算の並列化を行い、複雑な形状の例として立体トラスを対象として計算を行った。全体の計算時間の中で多くの割合を占める部分を並列化するほうが並列化の効率がより改善されることから、今回計算時間のほとんどの割合を占める、流速と圧力の計算部分である HSMAC 法の部分を並列化し、その検証を行う。

2. 解析方法の概要

本研究では、乱流シミュレーション法として LES を使い、数値計算を Staggered 格子で行った。時間進行は 2 次精度の Adams-Bashforth 法により行い、圧力解法に HSMAC 法を用いている。また HSMAC 法において通信時間短縮を目的として計算領域間における通信を、流速・圧力の修正を n 回行った後に 1 回通信を行うというように変化した。

3. 計算条件の概要

本研究における計算領域を図-1 に示す。計算領域は、トラスの総高を D とすると、 $5.3D \times 2.3D \times 3.3D$ である。計算格子は、 $128 \times 64 \times 96$ の直線直交不等間隔格子である。xy 平面の境界条件はすべり条件、zx 平面は周期境界条件とした。流入条件は、流速 1 の一様流を与え、流出条件は自由流出とする。この領域に図 1 に見られるように、ワーレントラスを設置し計算を行う。トラス総高 D と流速 U で定義される Reynolds 数は 4.0×10^5 である。

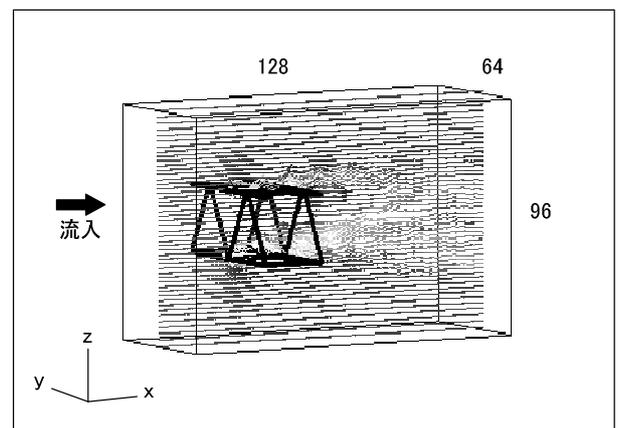


図-1 トラスを過ぎる流れの LES 計算領域

計算領域は、データ交換における通信量を減らすことを目的とし、断面が小さくなるよう x 方向で領域分割を行った。

計算に用いた計算機は CPU が AMD Opteron Processor 252 2.6GHz で OS が Linux2.6.11 である。この CPU の数を 1,2,4 と変化させて計算時間の計測を行った。

4. 計算結果

本研究で時間ステップを 8000 ステップ計算する間に、HSMAC 法によって圧力および流速を修正する部分
キーワード 数値計算, 並列化, LES, トラス

〒658-0044 神戸市灘区六甲台町 1-1 神戸大学工学研究科市民工学専攻 078-803-6280

でかかった累積時間を示す. 図-2には, 使用 CPU 数を変化させた結果を, 図-3には使用 CPU 数を 4CPU に固定し, HSMAC 法において圧力・流速の修正回数に対する通信の割合を 1/2, 1/4, 1/6 と変化させた場合の計時結果を示す. また, 図-4, 図-5 はそれぞれの条件で 1CPU での実行時間を標準とした何倍計算が速くなったかを表すスピードアップである.

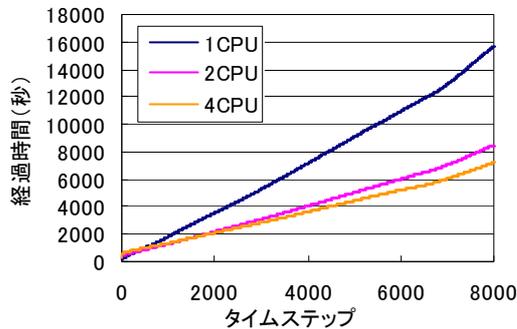


図-2 使用 CPU 数を変化させた計時結果

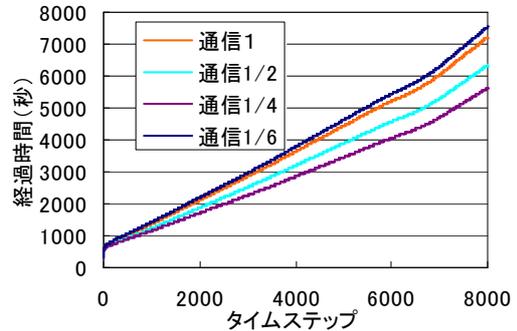


図-3 通信回数を変化させた計時結果

	経過時間	スピードアップ
1CPU	15701.7	1.00
2CPU	8488.9	1.85
4CPU	7229.8	2.17

図-4 使用 CPU 数を変化させたスピードアップ

	経過時間	スピードアップ
通信1	7229.8	2.17
通信1/2	6345.7	2.47
通信1/4	5635.1	2.79
通信1/6	7550.9	2.07

図-5 通信回数を変化させたスピードアップ

使用 CPU 数を変化させると, 1CPU から 2CPU に使用する CPU 数を変化させた場合スピードアップが 1.85 倍と変化している. N 台の CPU を利用した場合のスピードアップの理想値が N 倍であることから, 良好なスピードアップを得られていることがわかる. ところが, 使用 CPU 数を 4CPU に増やしても 2.17 倍とそれほど変化がない. これは領域を更に細かく分割したため, 各計算領域間のデータ通信回数が増え, それに伴いアイドルリング中である CPU が多く存在していることが原因であると考えられる.

そこで, 各反復計算における通信の割合を減らし計算を行った. 図-5 をみると, 反復計算を 4 回行った後に 1 回の通信を行うという割合まで通信を減らすと計算時間が短縮されることがわかる. スピードアップについても, 2.79 倍と理想値に近づけることができた. しかし, 更に通信割合を減らし, 反復計算 6 回行った後に 1 回の通信を行うようにしたところ逆に計算時間がかかってしまった.

5. まとめ

本研究では, LES において多くの時間を費やす HSMAC 法の部分の並列化を行い, どれほどのスピードアップが実現できるのか検討した. その結果, 通信を行う回数を減らすことで計算時間の短縮を実現することができた. また, 反復計算において, 全体の計算時間を最短にする最適な通信の割合が存在することがわかった.

参考文献

- 1) 牛島省, 奥山洋平, 禰津家久: 領域分割法を用いた QSI スキームによる移流拡散方程式の並列計算法, 水工学論文集, 第 46 巻, pp.415-420, 2002
- 2) 牛島省, 奥山洋平, 禰津家久: コロケート格子配置に基づく 3 次元非圧縮流体の陰的計算アルゴリズムの構築とその並列化, 応用力学論文集, vol.6, pp.185-192, 2003s
- 3) 青山幸也: 並列プログラミング入門 MPI 版