

並列計算機による 3 次元流れ場の数値解析

Parallel Processing of SIMPLED Method with HXnet for 3D Cavity Flow

加藤 信介*・村上 周三**・金田 康正*
張 維*・岡本 理*・三浦 靖弘*
Shinsuke KATO, Shuzo MURAKAMI, Yasumasa KANADA,
Wei ZHANG, Tadashi OKAMOTO and Nobuhiro MIURA

1. 序

一般に流れのシミュレーションは莫大な記憶容量と演算能力を必要とする。現状の流れのシミュレーションでは使用する計算機能力の制約により不十分な解析を余儀なくされており、より大容量、高速の計算システムの開発が望まれる。分散メモリ方式の超並列計算システムはこのような流れのシミュレーションの計算機能力の向上に対する要望に充分対応するものと期待されている^{8),9)}。本報では、SIMPLE による流体の数値シミュレーション手法を HXnet 型並列計算機 ADENART^{注1)}に適用し、2次元および3次元の Cavity flow を解析した結果を示す。

2. 並列計算機 ADENART の概要

並列計算機 ADENART は分散メモリ方式であり、システムは Host computer (Work Station), 並列計算機本体, interface unit の三つの部分で構成される。各 PE (processor element) は独特の Hyper cross Network (HXnet)^{注2)}により結ばれている。

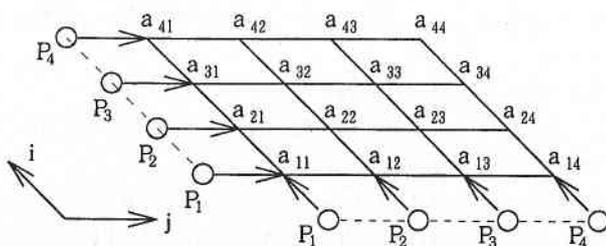


図1 ADENART の 2 次元配列並列処理概念

2.1 ADENART の並列処理構造

物理現象を記述する偏微分方程式の並列処理方針の一つに領域分割という考え方がある。すなわち、流れ場を各サブ領域に分割して各 PE に割り振る手法である。既報⁴⁾で報告した AP1000^{注3)}における並列処理はこの方針によるものである。一方、ADENART の並列処理は領域分割法でなく、偏微分方程式から得られた連立多元方程式の計算における配列データ処理に着目して並列化が行われる。すなわち、2次元や3次元配列をその各次元方向に DO 文などを用いて同じ計算処理する場合に関し並列処理が行われる。ADENART の 2 次元配列データの並列処理の概念を図 1 に示す。2次元問題では、2次元データ空間に対してプロセッサを一直線に配列する。たとえば $a_{i,j}$ に関し、 j 方向で並列化する場合 P_1 プロセッサは $a_{1,1}, a_{1,2}, a_{1,3}, a_{1,4}, \dots$ を処理し、 P_2 プロセッサは同じく $a_{2,1}, a_{2,2}, a_{2,3}, a_{2,4}, \dots$ を処理する。ここで、次に i 方向で並列化が必要になった場合 P_1 プロセッサは $a_{1,1}, a_{2,1}, a_{3,1}, a_{4,1}, \dots$ 、 P_2 プロセッサは $a_{1,2}, a_{2,2}, a_{3,2}, a_{4,2}, \dots$ と計算処理する。ADENART の最大の特徴はこの j 方向と i 方向の並列化の切り換えを行う際、個々のプロセッサに割り付けたデータの転送 (j 方向から i 方向への切り換えでは、 P_i プロセッサのデータは $a_{i,j} \rightarrow a_{j,i}$ と交換される) を効率的に行うようネットワークが組まれていることである (HXnet)。以下、行列 $A(i, j)$ と行列 $B(i, j)$ の和 $C(i, j) \{i=1, M, j=1, N\}$ を j 方向に並列化して求める例を示す。

```

pdo i=1, M
do 10 j=1, N
10 C(i/, j)=A(i/, j)+B(i/, j)
pend
    
```

pdo と pend (pdo パラグラフという) が一つの並列処理

*東京大学生産技術研究所 第5部

**東京大学生産技術研究所 付属計測技術開発センター

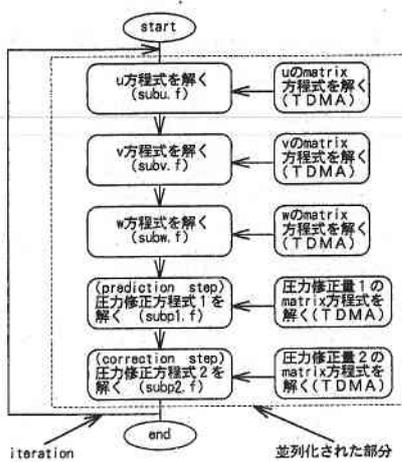


図2 SIMPLED法のフロー

単位になる。pdoパラグラフ内部では処理はjに関する次元のみであり、通常の逐次処理となる。'i/'はプロセッサ番号を示し、データが各プロセッサに割り当てられて処理されることを示す。3次元の場合の並列処理も2次元と同様に行うことができる。3次元の場合、空間に対してプロセッサを3次元データの各平面に配列し行う。3次元の単純な配列加算の例では、次のように表現される。

```

pdo i=1, M, j=1, N
do 10 k=1, L
10 C(i, j, k)=A(i, j, k)+B(i, j, k)
pend
    
```

この場合、'i, j/'はプロセッサ番号を示し、おのおののプロセッサが'i, j/'を省略したkに関する次元のデータについて、k方向に並列処理を行う。

2.2 ADENARTのnetwork

ADENARTでは、任意のPE間を2回の転送で通信可能なHXnetを採用している。注2]に他のnetworkとの通信必要時間の比較を示す。

2.3 並列処理言語

ADENARTの並列プログラミング言語ADETRANはFORTRANに並列性を加味した言語である。次の3つ点に関し、FORTRANが拡張されている。(1) 並列記述動作 (pdo文), (2) データ転送文 (pass文), (3) プロセッサ同期文 (ifall文)。

3. SIMPLED法の並列処理

3.1 SIMPLEDの概要

SIMPLED法は時間的に陰的な離散化を行うものであり、速度成分や圧力修正量に関し、おのおのの連立多元方程式を解く必要が生じる。各連立方程式は一般に反復法による

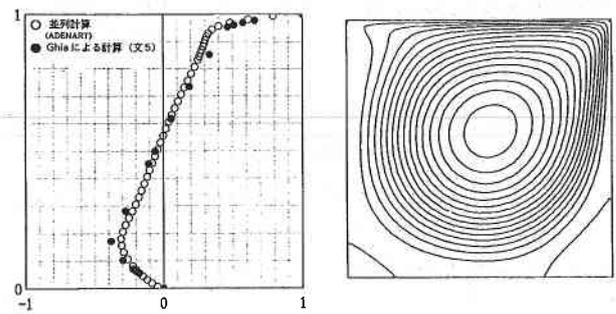


図3 2次元Cavity flow (Re=1000)

収束計算により解かれる。図2に解法のフローを示す。

3.2 Matrixの解法の並列性

一般に反復法は逐次加速緩和 (SOR) のように収束を加速するため逐次性を持たす。すなわち各反復計算においてk+1次点の近似値を得るためにはすでに計算されたk+1次の隣接格子の値を必要とする。並列処理は逐次処理と異なり、異なるPEは独立して演算が行われるため、この逐次性を十分とり込めない。ADENARTは行列の一方方向に関して並列処理を行い、この方向のみ逐次性を持たすことができる。本研究では行列解法にLine法 (TDMA) を用いている。2次元の場合、TDMAではy方向 (j方向) の解を固定して定数扱いし、x方向 (i方向) の一次元方程式 (代数方程式の係数行列は3重対角行列となる) を解く。これをy方向に関し逐次的に処理すればLine-SORとなるが、y方向に関しては同時に並列処理するため逐次性を持たすことができない。Line法はRed and Black法により逐次性を持たして並列化することも可能であるが、ADENARTはこの場合PEの利用率が1/2となるため、今回は用いてない。

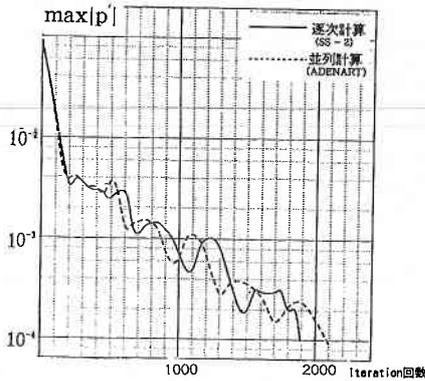
4. 2次元Cavity flow解析例

4.1 解析対象

正方形のCavity flow。Re数は1000。移流項は1次精度のpower lawスキームを利用。格子分割は80x80。Cavity固定壁はno-slip条件を課し、移動壁の無次元速度は1とする。

4.2 計算結果および考察

- 1) 流れ場 計算した結果は図3に示すようにGhiaらの結果とほぼ一致している。
- 2) 加速率 表1にiteration 100回 (図4参照) 当たりのSunSS-2, ADENART64 (PE), 同256 (PE) の実行時間を示す。SS-2のCPUとADENARTのPEは性能が異なるので、単純な比較はできないが、今回のADENARTの実行性能は次に述べる理由で期待された程高く



3次元Cavity flow (Re=400)
図4 逐次と並列処理の収束過程

はない。3次元のコードを用いて、2次元計算を行ったためデータ配列が計算セルの他に仮想セルを含み $82 \times 82 \times 3$ となる。このため、k方向は、3セル（計算は1セル）しかないが、並列計算は、この3次元データに関して並列化されている。この場合、PEの利用率が全体の約 $1/8^{[4]}$ となり、低い性能比を示す。PE数を64から4倍の256と増加させた場合、性能比は2倍に留まった。

3) 逐次計算と並列計算の収束性の比較 並列処理では、各Line間の逐次性が確保されず、異なるLineの隣接値は前次点の値が参照される。しかし、今回の2次元計算においては逐次更新値を使えないのは一方向のみであることおよび流れ性状等の特性により逐次処理した場合と同様の収束性が得られた。

4) データの通信時間 2次元Cavity flowの場合のPEの利用率が約 $1/8$ と低いため正確な通信時間の見積りはできないが、全計算時間に占めるデータの通信時間の割合は64PEの場合8.6%、256PEの場合10.6%となる。すなわち、データの転送時間は全時間の10%程度に留まる。

5. 3次元Cavity flow 解析例

5.1 計算概要

3次元立方体のCavity flow. Re数は400. 移流項は1次精度のpower lawスキームを利用。格子分割は仮想セルを含みCASE 1 ($42 \times 42 \times 42$), およびADENARTの並列処理上都合がよいCASE 2 ^[5] ($48 \times 48 \times 48$). Cavityの固定壁はno-slip条件を課し、移動壁の無次元速度は1とする。

5.2 計算結果および検討

1) 流れ場 並列計算した結果を既往の計算結果¹⁰⁾とともに図5 (Re=400) に示す。並列計算機による解析結果は逐次計算機(SS-2)および既往の計算結果と一致して

いる。

2) 加速率 表2に、CASE 1の並列処理(64PEおよび256PE)と逐次処理(SS-2)のiteration 100回当たりの実行時間を示す。3次元シミュレーションの場合、並列計算では3次元データに関して有効に並列化され、2次元で問題となった低いPEの利用率は生じない。加速率は2次元の場合の性能に比べて4倍程度の増加であり、期待された8倍までは到達していない。表3に、格子分割数を8の倍数とし並列計算機の処理特性に合わせたCASE 2の場合の並列処理と逐次処理のiteration 100回当たりの実行時間を示す。加速率はCASE 1に比べ向上している。この場合、PE数を64から4倍の256と増加させた場合、性能比の向上は3.9倍を示し、PE数の増加にほぼ比例した性能の向上を示すよい結果を得た。

3) 逐次計算と並列計算の収束性の比較 図4にCASE 1の場合の並列処理と逐次処理の収束過程を示す。2次元計算と比べ3次元の場合、圧力修正量が 10^{-4} に到達するまでのiteration回数は並列処理の方が逐次処理に比べ多く、収束性がわずかに劣っている。

4) データの通信時間 CASE 1の場合、通信時間が全計算時間に占める割合は64PEの場合10.8%、256PEの場合9.1%となり、2次元の場合とほぼ同様である。プロセッサ数が増加しても通信時間はあまり多くなりず、並列処理の効率はかなり高い。

6. ま と め

(1) 並列計算機ADENARTの概要および並列処理方法

表1 2次元Cavity flowの100回iterationの計算時間

機種	SS-2	ADENART 64PE	同 256PE
実行時間(秒)	375.0	74.5	37.9
性能比	1	5.03	9.9
データ転送時間(秒)		6.4	4.0

表2 3次元Cavity flow(CASE 1)の100回iterationの計算時間

機種	SS-2	ADENART 64PE	同 256PE
実行時間(秒)	2770.7	132.2	46.1
性能比	1	21.0	60.1
データ転送時間(秒)		14.3	4.2

表3 3次元Cavity flow(CASE 2)の100回iterationの計算時間

機種	SS-2	ADENART 64PE	同 256PE
実行時間(秒)	5100.5	205.8	52.4
性能比	1	24.8	97.3
データ転送時間(秒)		14.3	4.2

研究速報

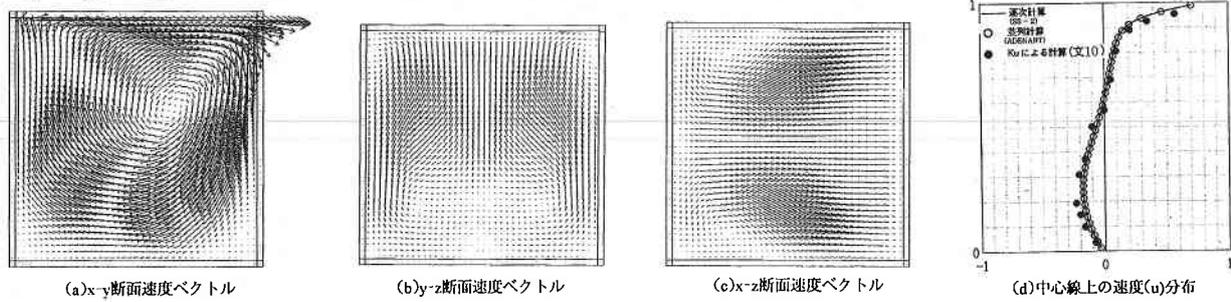


図5 3次元 Cavity flow (Re=400)

を述べた. これにより SIMPLED 解法を並列処理し, 2次元および3次元の Cavity flow を計算し, 並列処理の効率を確認した.

(2) 3次元 SIMPLED 法の並列処理コードを利用して, 2次元計算した場合, PE の利用率が1/8 (ADENART 64PE) となり, 十分な加速率が得られない. しかし, 同コードを用いた3次元計算では, 加速率は2次元計算の加速率の4倍となり, 効率的に計算が行われる.

(3) 2次元および3次元計算における通信時間のオーバーヘッドは全計算時間の10%程度である.

(4) Line 法による逐次性を犠牲にした並列処理は今回の流れ場では2次元計算, 3次元計算とも, LineSOR による逐次処理とほぼ同様の収束性を示す.

謝 辞

本研究は松下電器(株)との共同研究によるものである. 研究実施にあたり松下電器(株)超 LSI デバイス研究所各位のご協力を得た. 記して, 謝意を表する.

(1993年12月9日受理)

注

注1] 使用した ADENART64の性能表 (松下電器製)

PE	64 bit Floating Point Processor (16MFLOPS)
Memory	8(MB) × 64PE = 512 (MB)
Performance	16(MFLOPS) × 64PE = 1.0 (GFLOPS)
Data Transfer	16(MB/s) × 64PE = 1.0 (MB/s)

注2] Network の通信比較

Netname	複数度(ブランチ数)	実行転送時間	応用分野
Mesh 型	2N	N ^{1/2}	PDE陽解法
Hyper cube 型	N/2 log ₂ N	log ₂ N	PDE陽解法
Hxnet(ADENART)型	N ^{3/2}	2	PDE陰解法

注3] 富士通製

注4] 現時点で ADENART コンパイラが並列処理 PE の割り当てを自動的に行うため, このような現象が生じる. コンパイラは i 方向に並列処理する時 j, k/面に対し8(j) × 8(k) プロセッサ平面を割り当てて並列化する. 実際の計算は j が2~81, k が2~2になるので, 64プロセッサが64(j) × 1

(k)と割り当てるのが合理的となるが, そのような最適化は今回されていない. すなわち今回, 配列(82 × 82 × 3)と宣言された場合, i 方向への並列処理は全体の1/8の PE しか動作しない. この点は mesh 分割の最適化も含め, 今後改善する予定である.

注5] ADENART は, 64PE の場合, プロセッサが8 × 8平面に割り当てられて並列化されるので, 8の倍数となる mesh 分割は利用率が最も高いと考えられる.

参 考 文 献

- 1) 松下電器: "ADENART マニュアル" 1992
- 2) K. Tsuboi, K. Miyakosi and K. Kuwahara: "An Approach to in Compressible Flow Problems with a Massively Parallel Computer", Parallel Computational Fluid Dynamics, 1991
- 3) 岡本理, 富田貞文, 小田中紳二, 材木幸治, 鎌田浩: "並列計算機 ADENA の性能評価", 日本機械学会第68期全国大会講演論文集, Vol. B, 1990.9.23, 仙台
- 4) 村上周三, 加藤信介, 内海康雄, 水谷国男: "室内気流解析の並列演算処理に関する検討", 生産研究, 1991.2
- 5) Ghia, U, Ghia, K, N and Shin, C.J, J. Comput. Phys. 48, p 387, 1982
- 6) 張維, 村上周三, 持田灯: "定常解法による建物周辺気流の数値シミュレーション (その1, その2)", 日本建築学会関東支部論文報告集, PP 76~84, 1992.1, 東京
- 7) 張維, 村上周三, 持田灯: "定常解法による建物周辺気流の数値シミュレーション (その3)", 日本建築学会論文梗概集, 1992.8, 東京
- 8) 内海康雄, 村上周三, 加藤信介, 張維: "並列処理による流体の数値解析に関する研究 (その1) - 並列処理の概説 -", 日本建築学会関東支部論文報告集, 1993.3, 東京
- 9) 張維, 村上周三, 加藤信介, 内海康雄: "(同題その2) - 流体の数値シミュレーションの並列アルゴリズム -", 日本建築学会関東支部論文報告集, 1993.3, 東京
- 10) H C, Ku, Richard S, Thomas D, Taylor: "A Pseudospectral Method for Solution of the Three-Dimensional Incompressible Navire-Stokes Equations", J. Comput. Phys. 70, 1987