

分散メモリ計算機における 局所非定常モデルを用いた楕円型格子生成法

Locally pseudo-unsteady model for elliptic-type grid generation
with distributed memory system

牛島 省*・藤田 学**・瀬津 家久***

Satoru Ushijima, Manabu Fujita and Iehisa Nezu

*正会員 工博 京都大学大学院助教授 社会基盤工学専攻 (〒 606-8501 京都市左京区吉田本町)

**学生員 京都大学大学院 社会基盤工学専攻 修士課程

***フェロー会員 工博 京都大学大学院教授 社会基盤工学専攻

In the multi-dimensional fluid computations for actual hydraulic problems, it is necessary to take account of the complicated boundary shapes to obtain suitable computational results. One of the effective methods is to employ the elliptic-type grid generation to transform the physical computational domain to the regular one. In this paper, a locally pseudo-unsteady model is proposed to make the grid generation much faster than the usual methods in the distributed-memory system. The non-linear equations for the grid generation are discretized by introducing pseudo-time and the numerical process is divided into a time-marching outer-loop and an inner-loop in which numerical solutions for linearized elliptic equations are derived. This model allows us to decrease message passings included in the inner-loop, which largely reduces the computational load in the parallel computations using distributed memory systems. The validity of the proposed model is demonstrated in some numerical experiments.

Key Words : *curvilinear coordinates, grid generation, distributed memory system, locally pseudo-unsteady model, parallel computation*

1. はじめに

水理現象を多次元の流体計算法を用いて予測する場合には、自然の地形あるいは人工的に造られた水理構造物などの境界形状を適切に扱うことが重要である。これらの境界面の形状は一般に複雑なものとなることが多い。境界付近だけでなく領域の広い範囲にわたる流況が境界形状の取り扱いに左右されることもあるため、この点に関する十分な配慮が必要である。複雑な境界形状を取り扱う手法の一つに、一般座標系を利用する流体計算法があり、各種の水理現象に適用されてその有効性が示されている^{1),2)}。

一般座標系に基づく格子生成法としては、これまで多くの方法が提案されている³⁾。このうち、楕円型方程式を利用する方法⁴⁾は領域内に偏りなく格子点を配置できること、また制御関数を適切に設定することにより格子間隔の粗密を調整できることから、一般に広く利用されているようである。しかしながら、この格子生成法では、本文中で述べるように、外部反復と内部反復という2重の反復計算が各座標成分に対して行われるため、計算対象が3次元で格子点数が増加する

と、多くの計算時間が必要となる。格子生成では、格子点分布の配置などを調整するため、複数回の試行が必要なものもある。流体計算に先立って格子生成に多大な計算時間が必要となることは全体の作業を進める上で効率的ではなく、高速化が望まれる。また、局所洗掘による砂面変形や自由水面の変動などの移動境界問題にこの格子生成法が用いられる場合^{5),6)}には、流体計算の過程で格子生成が何度も行われるため、高速の格子生成法が必要となる。

以上を考慮して、本研究では、並列演算を導入することにより、楕円型方程式を使用する場合の格子生成を高速化する検討を行う。並列計算環境としては、現在広く普及しつつあるPCクラスタに代表される分散メモリシステムを対象とする。既報^{7),8)}では、逐次計算環境における反復計算を領域分割により並列化する方法を検討しているが、本報では分割された小領域内で内部反復を行う際に、外部の領域で定常性を仮定する局所非定常モデルを導入することにより、プロセス間通信の負荷を軽減し、計算をさらに高速化する方法を考察する。数値実験を通じて、本報で示す解法は従来の並列格子生成法よりも高速であることを示す。

2. 格子生成法

2.1 座標生成のための関係式と一般的な解法

最初に, Thompson ら⁴⁾が提案した楕円型方程式を用いる格子生成法の関係式と, これに対して一般に行われている計算手順を示す. ここで扱う格子生成法は, 物理空間の3次元座標成分を x_i , 写像空間中の座標成分を ξ_m とするとき, 次式の関係を満たす x_i を求めるというものである ($i, m = 1, 2, 3$).

$$\nabla^2 \xi_m = P_m(\xi_1, \xi_2, \xi_3) \quad (1)$$

ここで, $P_m(\xi_1, \xi_2, \xi_3)$ は物理空間における格子間隔の粗密を制御するための任意の関数である. 実際に x_i を求めるためには, 式 (1) の独立変数と従属変数を交換した次式が用いられる.

$$\begin{aligned} & \alpha_1 \frac{\partial^2 x_i}{\partial \xi_1^2} + \alpha_2 \frac{\partial^2 x_i}{\partial \xi_2^2} + \alpha_3 \frac{\partial^2 x_i}{\partial \xi_3^2} \\ & + 2\beta_{12} \frac{\partial^2 x_i}{\partial \xi_1 \partial \xi_2} + 2\beta_{23} \frac{\partial^2 x_i}{\partial \xi_2 \partial \xi_3} + 2\beta_{31} \frac{\partial^2 x_i}{\partial \xi_3 \partial \xi_1} \\ & + P_1 \frac{\partial x_i}{\partial \xi_1} + P_2 \frac{\partial x_i}{\partial \xi_2} + P_3 \frac{\partial x_i}{\partial \xi_3} = 0 \end{aligned} \quad (2)$$

ここで, α_i と β_{ij} は次式で与えられる.

$$\alpha_i = \sum_{k=1}^3 c_{ik}^2, \quad \beta_{ij} = \sum_{k=1}^3 c_{ki} c_{kj} \quad (3)$$

上式の c_{ij} は次式で定義される.

$$c_{ij} = \frac{\partial \xi_i}{\partial x_j} \quad (4)$$

この c_{ij} を求めるためには, 次の v_{ij} を計算し,

$$v_{ij} = \frac{\partial x_i}{\partial \xi_j} \quad (5)$$

これを要素とする行列 V と, c_{ij} を要素とする行列 C の間で成り立つ次の関係を利用する.

$$C = V^{-1} = \frac{1}{|V|} \tilde{V} \quad (6)$$

ここに, $\tilde{V}$ は V の余因子行列の転置行列であり, $|V|$ は V の行列式である.

これらの関係式を用いて格子生成を行うには, 一般に次のような方法が用いられる. まず, 初期値として, 適当な物理座標 x_i^0 を与え, 式 (5) より v_{ij} を計算し, 式 (6) の関係を利用して c_{ij} を得る. 次に, 式 (3) から α_i と β_{ij} を求める. これらを固定された係数と考えると, もともと非線形方程式であった式 (2) は, 線形な楕円型偏微分方程式と見なすことができるので, これを離散化して数値解が求められる. 数値解を求めるため

には, 多くの場合に反復計算が用いられる. このときの反復計算を本報では内部反復と呼ぶこととする. 内部反復で得られた数値解を x_i^1 とし, x_i^1 から上記と同様にして α_i と β_{ij} を求め, 再び内部反復を行い, x_i^2 を得る. このような x_i^k を求める外側の反復計算を外部反復とする. k 回の外部反復により得られる x_i^k と x_i^{k-1} との相違が微小となった時点で x_i が定まり, 格子分布が得られる.

このように, 1つの座標成分を定めるために2重の反復計算が行われ, 3次元の場合には3つの座標成分のそれぞれに対して同様の計算が必要となる. 実際の流体計算では, 境界付近で格子間隔を密に配置するなど, 適切な格子分布を得るためには何回かの試行が必要であることが多い. このような格子生成の計算を高速化するため, 既報^{7),8)}では, 並列計算を利用して格子生成を行う方法を示した. 本報では, 擬似的な時間項を導入して, 内部反復に局所非定常性を仮定するモデルを用いることにより, さらに高速に格子生成を行う方法を検討する.

2.2 擬似的な時間項の導入

楕円型方程式の数値解を得るために, 時間項を導入して, その定常解を求める方法が用いられている. 本報では, 式 (2) に時間項を導入して, 次のように時間発展型の式とする.

$$\frac{\partial x_i}{\partial \tau} - F(x_i) = 0 \quad (7)$$

ここで, τ は擬似的な時間で, 本報では以下 x_i と ξ_m は無次元量であるとし, τ も適当な無次元量と考える. $F(x_i)$ は式 (2) の左辺を表す. 式 (7) では, 外部反復は時間を更新する計算ステップに相当し, 定常解が得られた時点で格子点位置が定まる.

式 (7) を時間ステップ n と $n+1$ の間で重み ω を付けて次のように離散化する.

$$x_i^{n+1} = x_i^n + [\omega F(x_i^{n+1}) + (1-\omega)F(x_i^n)] \Delta \tau \quad (8)$$

ここで $0 \leq \omega \leq 1$ であり, $\omega = 0$ で陽解法, $\omega = 1$ で完全陰解法となる. 式 (8) の x_i に対して, Δ 型陰解法⁹⁾あるいはこれをコロケート格子に利用した C-ISMALC 法¹⁰⁾と同様に,

$$x_i^{n+1} = x_i^n + \phi_i \quad (9)$$

とおく. 定常解では ϕ_i は微小量となる. 式 (9) を式 (8) へ代入すると, 内部反復の過程では F は線形化されているので $F(x_i^n + \phi_i) = F(x_i^n) + F(\phi_i)$ が成り立ち, 次の結果を得る.

$$[1 - \omega \Delta \tau D] \phi_i = F(x_i^n) \Delta \tau \quad (10)$$

ここで、 D は次のような微分演算子である。

$$D = \sum_{k=1}^3 \left(\alpha_k \frac{\partial^2}{\partial \xi_k^2} + P_k \frac{\partial}{\partial \xi_k} \right) + 2 \left(\beta_{12} \frac{\partial^2}{\partial \xi_1 \partial \xi_2} + \beta_{23} \frac{\partial^2}{\partial \xi_2 \partial \xi_3} + \beta_{31} \frac{\partial^2}{\partial \xi_3 \partial \xi_1} \right) \quad (11)$$

式 (10) の ϕ_i は、外部反復により定常解が得られた段階では 0 に近い微小な値となる。このため、 Δ 型陰解法の考え方と同様に、空間方向の離散化に式 (10) 左辺には低次スキームを用い、右辺には必要であれば高次スキームを利用することができる。このようにすると、陰的に扱われる ϕ_i には低次スキームが用いられているので離散化式が簡単になり、しかも高次精度の定常解が得られるという利点がある。このように高・低次のスキームを使い分ける解法の有効性は、流体計算において確認されている¹⁰⁾。ただし、本報の以下の部分では、格子生成に必要な計算の速度を向上させることに重点を置き、式 (10) に対しては、両辺に同じ 2 次精度の中心差分を用いることとする。

式 (10) を空間的に離散化すると、 ϕ_i に関する連立 1 次方程式

$$A\phi_i = b_i \quad (12)$$

が得られる。ここで、 ϕ_i は格子点 (p, q, r) 上の ϕ_i 、すなわち $\phi_{i,pqr}$ から構成される列ベクトルで、 A は係数行列、右辺の b_i は式 (10) 右辺を離散化した値である。 b_i は x_i^n より計算できるので既知である。式 (12) に適当な反復解法を用いて数値解 $\phi_{i,pqr}$ を求める (内部反復)。式 (9) より、得られた $\phi_{i,pqr}$ を $x_{i,pqr}^n$ に加えることで、 $x_{i,pqr}^{n+1}$ が求められる。以上の演算で外部反復の 1 サイクルが終了する。 x_i の収束解が得られていなければ次の外部反復が繰り返される。

2.3 分散メモリ計算機上での並列演算

写像空間中に変換された計算領域は、格子点が規則的に配置する直方体の形状となるので、図-1 に概略的に示すようにこれを複数の小領域に区分し、対応するプロセスを規則的に配置する。MPI ではこのような Cartesian トポロジーを設定するルーチンも用意されているが、ここではユーザ側で rank 番号とプロセスの配置番号 (図-1 中の P_{IJK}) の対応を明示的に定めた。この配置に基づいて領域外の配置番号に MPIPROC_NULL という rank 番号を設定することにより、通信時に境界近傍で場合分けをせずに簡潔にプログラムを書くことができる。

式 (10) 左辺に 2 次の中心差分を用いる場合には、図-2 に示す格子点を使用され、中央の番号 0 の格子点を中心として周囲の 18 点の値が用いられる。このため、

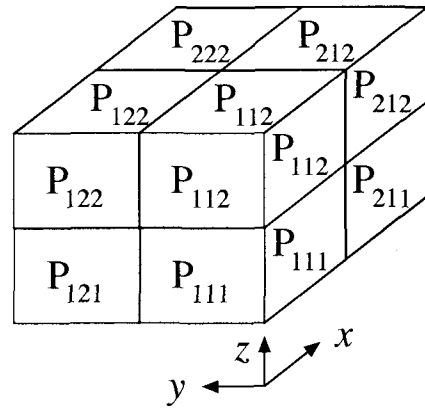


図-1 写像空間中での計算領域の分割 (P_{IJK} はプロセスの配置番号)

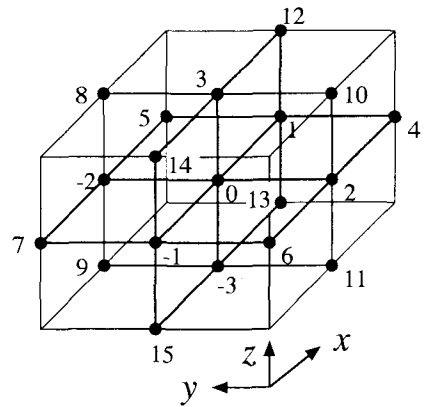


図-2 離散化式で用いられる 19 個の格子点 (格子点番号は-3 から 15)

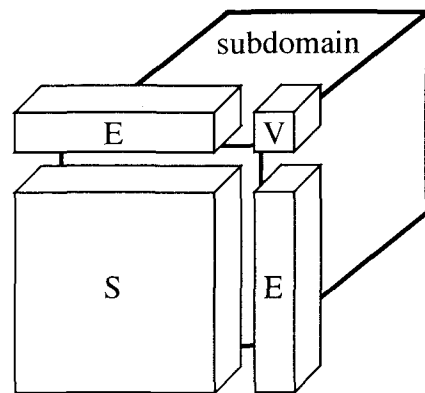


図-3 重合格子点の分布

式 (12) の係数行列は対角要素も含めて 1 行に最大 19 の非ゼロ要素を持つ帯行列となる。このような大規模な疎行列を係数行列とする方程式に対しては、非定常反復解法により比較的高速に解が得られることが示されているので、本報ではいくつかの代表的な解法を選択して分散メモリシステム上での並列演算を行うこととする。

具体的な解法については後述することとし、ここでは並列演算の概要のみを示す。分散メモリシステム上では、ベクトルと係数行列は、次式のように行ごとにブロック分割されることが多い。

$$A\phi_i = \begin{bmatrix} \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots \\ \cdots & a_{pqr}^0 & \cdots \\ \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} \vdots \\ \vdots \\ \phi_{i,pqr} \\ \vdots \\ \vdots \end{bmatrix} \\ = b_i = \begin{bmatrix} \vdots \\ \vdots \\ b_{i,pqr} \\ \vdots \\ \vdots \end{bmatrix} \quad (13)$$

a_{pqr}^0 は i 行の対角要素である。各プロセスは、ベクトル ϕ_i の要素のうち、割り当てられた小領域内の要素のみを持つ。

非定常反復計算では、計算量が比較的大きい演算は行列とベクトルの積の計算とベクトルの内積計算である。前者に関しては、 A には非対角要素が含まれるので、ブロック内の $A\phi_i$ が全体の $A\phi_i$ の対応する行と一致するためには、ブロック内の係数行列の非対角要素とブロック外の ϕ_i の要素の積を含める必要がある。このときの通信対象となる格子点の配置は、図-3 に示す重合格子点のうち、S および E という記号が付された部分に相当する。これは、離散化式には隣接する領域の面のデータ (S) だけでなく、辺上のデータ (E) が含まれているためである。これらのデータを通信する際には、MPI の派生データ型 (derived data type) を使用して、S および E の格子配置形状を定めることにより、プログラムを簡潔に記述した。なお、派生データ型を利用することによる通信速度の低下は認められないことが報告されている¹¹⁾。

一方、内積計算に関しては、ブロック内の内積値を通信して総和を取ることになるので、やはりプロセス間の通信が必要であるが、mpi_allreduce を一度呼ぶだ

けなので通信量は上記の行列・ベクトル積の場合と比較して少ない。

上記で述べた分散メモリシステム上での全体の計算過程の概略は、次のように表される。

```
do n = 0, 1, 2, ... ; until  $|r_x| < \epsilon_x$ 
  do k = 0, 1, 2, ... ; until  $|r_\phi| < \epsilon_\phi$ 
    ...
```

```
    mpi_sendrecv(w) (14)
```

```
    mpi_allreduce(v) (15)
```

```
    ... (16)
```

```
    mpi_allreduce( $r_\phi^{k+1}$ ) (17)
```

```
  enddo (18)
```

```
   $x_i^{n+1} = x_i^n + \phi_i$  (19)
```

```
  mpi_sendrecv( $x_i^{n+1}$ ) (20)
```

```
  mpi_allreduce( $r_x^{n+1}$ ) (21)
```

```
  calculate local  $A, b_i$  (22)
```

```
enddo
```

上記の計算過程のうち、 n および k の反復がそれぞれ外部および内部反復である。後者は式 (12) の解を求める非定常反復計算に相当する。式 (15) の mpi_sendrecv は、非定常反復計算の行列・ベクトル積の計算に必要な通信を表し、式 (16) の mpi_allreduce は、内積計算の際に必要な通信を表す。解法により詳細は異なるが、1 回の反復計算で行列・ベクトル積の演算は 1 回ないし 2 回行われ、内積計算は通常複数回行われるので、これらの通信はその度に必要となる。式 (18) の mpi_allreduce は、小領域の誤差を全体で合計して、領域全体の残差を求める処理を表す。

外部反復では、式 (19) により、内部反復で得られた ϕ_i から x_i^{n+1} が求められ、係数行列と右辺ベクトルの形成に他の小領域の x^{n+1} が必要となるので、式 (20) により通信が行われる。本報では、式 (10) 右辺の離散化にも 2 次の中心差分を用いるので、式 (20) の通信は式 (15) のそれと同じルーチンで行われる。式 (21) は外部反復の収束判定のための通信である。

2.4 局所非定常モデル

分散メモリシステム上の計算では、プロセス間の通信量を軽減することが計算効率向上のために重要である。上記の計算過程のうち、負荷が大きい通信処理は、2 重ループの内側にある式 (15) の部分である。ここでは、局所的な非定常性を仮定して、この通信処理を軽減するモデルを提案する。このモデルでは、計算対象

となる小領域の外部の空間ではすでに x_i^n の定常解が得られていると仮定し、小領域内部の空間でのみ定常解からの残差 ϕ_i が存在するとして内部反復を行う。この局所的な非定常状態を仮定したモデルを用いると、領域外部の空間では $\phi_i = 0$ であるので、内部反復の対象は境界値 0 のディリクレ問題と同等となる。このため、初期ベクトルを $\phi_i^0 = \mathbf{0}$ とすれば式 (15) の通信は不要となり、負担が比較的小さい内積計算に関する通信のみを行えばよいことになる。しかしながら、局所的な非定常性を仮定する場合には、当然予想されるように、より多くの外部反復が必要となる。したがって、全体の計算効率化は通信の削減と外部反復の増加のどちらの影響が支配的となるかに依存する。

この点をより明確にするため、局所非定常モデルの極限の状態として、小領域内に単一の格子点のみが含まれる場合を考える。このときには、式 (10) より

$$\phi_{i,pqr} = \frac{F(x_i^n)\Delta\tau}{1 - \omega a_{pqr}^0 \Delta\tau} \quad (23)$$

となる。ここで、係数行列の対角成分 a_{pqr}^0 は 2 次の中心差分を用いているので、

$$a_{pqr}^0 = -2 \sum_{k=1}^3 \alpha_k \quad (24)$$

であり、 $a_{pqr}^0 \leq 0$ となる。なお、ここでは写像空間中の格子幅は $\Delta\xi_m = 1$ としている ($m = 1, 2, 3$)。したがって、式 (23) の分母は 1 以上の正の値となり、時間刻み $\Delta\tau'$ を改めて次のようにおく。

$$\Delta\tau' = \frac{\Delta\tau}{1 - \omega a_{pqr}^0 \Delta\tau} \quad (25)$$

そして、式 (23) を離散化された式 (9) に用いると、外部反復の 1 ステップの計算後には、

$$x_{i,pqr}^{n+1} = x_{i,pqr}^n + F(x_i^n)\Delta\tau' \quad (26)$$

という結果が得られる。これは、式 (7) を時間刻み $\Delta\tau'$ で Euler 陽解法により離散化したものとなる。すなわち、この場合の計算は内部反復を行わずに外部反復のみで収束解を得る計算に相当する。

以上より、小領域内に含まれる格子点数が少ない、すなわち粒度が小さい条件では、局所非定常モデルは有効でない可能性がある。しかし、流体計算で用いられる格子点数は、一般に数万から数百万以上であるので、一般の分散メモリシステムのプロセッサ数と格子点数の関係を考慮すると、ある程度粒度の大きい計算が行われる場合がほとんどであると考えられる。このため、少ない格子点数に対して超並列計算機を利用するような場合を除けば、本モデルは有効であると期待される。

3. 適用例

3.1 内部反復に用いる解法

内部反復に用いる非定常反復計算法としては、GCR(k) 法¹²⁾を主として使用することとした。GCR 法は、収束解を得る際の安定性の観点から、GMRES 法と比較してあまり広く使用されることはないようであるが、すべての境界面の境界条件がディリクレ条件で与えられる 3 次元ラプラス方程式の差分式など、本報の内部反復で扱われるものと同様の方程式に対しては比較的高速かつ安定に解が得られる場合が報告されているので¹¹⁾、これを選択した。

GCR 法では、方向ベクトルは $A^T A$ 直交、すなわち

$$(A\mathbf{p}_i, A\mathbf{p}_j) = 0, \quad (i \neq j) \quad (27)$$

となるように取られ、 $\mathbf{p}_{i+1}$ は次式のように反復過程で得られたすべての方向ベクトルを用いて定められる。

$$\mathbf{p}_{i+1} = \mathbf{r}_{i+1} + \sum_{j=0}^i \beta_j^{(i)} \mathbf{p}_j \quad (28)$$

ここで、 $\beta_j^{(i)}$ を以下のように設定することにより、 $\mathbf{x}_{i+1}$ の残差ノルムは空間 $\mathbf{x}_0 + \{\mathbf{p}_0, \dots, \mathbf{p}_i\}$ において最小となる。

$$\beta_j^{(i)} = -\frac{(A\mathbf{r}_{i+1}, A\mathbf{p}_j)}{(A\mathbf{p}_j, A\mathbf{p}_j)}, \quad (j \leq i) \quad (29)$$

GCR 法では A が $n \times n$ 行列の場合に高々 n 回の反復計算で解が得られることが示されている¹²⁾。しかし、上記の手順からわかるように、GCR 法では、反復過程で得られたすべての方向ベクトルを保存しておく必要があり、記憶容量の面で実用的ではない。このため、GMRES 法と同様に適当な反復回数 k を定めて、 k 回の反復計算で数値解が得られない場合には、これを初期値としてリスタートする GCR(k) 法が提案された。一方、過去 k 回の反復計算で得られた最新の k 個の方向ベクトルのみを保存し、これを用いて方向ベクトルを定める方法は Orthomin(k) 法といわれる。GCR(k) 法と Orthomin(k) 法は、 $(A + A^T)/2$ が正定値のときに収束することが示されている¹²⁾。

内部反復に対する試計算を行った結果、 $k = 1$ 、すなわちリスタートまでの反復回数を最小とする場合に計算時間が最も短く、また安定に解が得られることが分かったので、GCR(1) 法を使用することとした。GCR(1) 法は Orthomin(1) 法と同一で、方向ベクトルは

$$\mathbf{p}_{i+1} = \mathbf{r}_{i+1} + \beta_i^{(i)} \mathbf{p}_i \quad (30)$$

$$\beta_i^{(i)} = -\frac{(A\mathbf{r}_{i+1}, A\mathbf{p}_i)}{(A\mathbf{p}_i, A\mathbf{p}_i)} \quad (31)$$

から計算されることになり、 p_{i+1} は p_i に対してのみ直交となる。

内部反復計算では、前処理として打ち切りノイマン展開¹³⁾を共通に使用した。この前処理法は、係数行列 A の対角要素が 1 であるようにスケーリングされているとき、これを $A = I - E$ の形に分解し、 E のスペクトル半径が 1 より小さいときに次のように表されることを利用するものである。

$$A^{-1} = (I - G)^{-1} = \sum_{j=0}^{\infty} E^j \quad (32)$$

式 (32) 右辺は、多くの場合 2 次か 3 次程度で打ち切られるが、試計算の結果、計算負荷の軽い 1 次の打ち切りで比較的良好な結果を得たので、本報では 1 次打ち切りを利用する。この前処理は行列とベクトルの積の計算により行われるので、不完全 LU 分解系統の前処理法と比較して分散メモリシステムでは扱いやすい。

3.2 座標生成への適用

本報で提案された局所非定常モデルの有効性を調べるため、実際に格子生成を行い、計算時間等を比較する。計算時間の比較を適切に行うためには、各方向に境界形状の相違や格子間隔の偏りが無いことが望ましいと考え、格子生成の対象として単純な立方体領域を選択した。立方体の一辺の無次元長さは 1 である。式 (1) の P_m は、 $P_m = 10 (\xi_m - \xi_{mc})$ とし、各境界面近傍で格子点を均等に密に分布させる。ここに、 ξ_{mc} は ξ_m 方向の領域中心点である。50 × 50 × 50 の格子点の分布例を図-4 に示す。なお、この格子数は、各方向の両端に位置する境界条件を設定する格子を含んだもので、実際に計算対象となる格子数は 48 × 48 × 48 である。後述する並列計算では、各プロセスが受け持つ格子数は等しく設定されている。

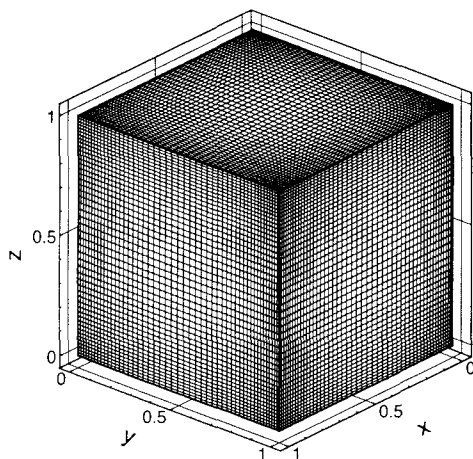


図-4 格子点の分布 (格子点数 50 × 50 × 50)

格子生成の際の初期値としては、等間隔格子を用いた。立方体の境界面においては、法線方向の座標値はディリクレ条件、接線方向にはノイマン条件を用いた。このため、境界上の格子点は計算の進行とともに接線方向に移動することになる。

計算では $\Delta\tau = 1.0$ および $\omega = 1$ とし、外部および内部反復の収束判定は、残差ベクトルの内積に対してそれぞれ $\epsilon_x = 10^{-10}$ および $\epsilon_\phi = 10^{-10}$ に統一した。計算には京都大学学術情報メディアセンターの HPC2500 を利用した。HPC2500 は、SMP クラスシステムであるが、本報では分散メモリシステムに対する格子生成法の特長を見るため、スレッド並列機能は使わず、プロセス並列による並列計算のみを行うこととした。

図-5 に x 方向の格子点間隔の分布を示す。同図では、隣り合う格子点との間隔がその中央にプロットされている。1CPU による逐次計算と、局所非定常モデルを用いた並列計算の結果は一致しており、同一の格子分布が生成されていることを確認した。

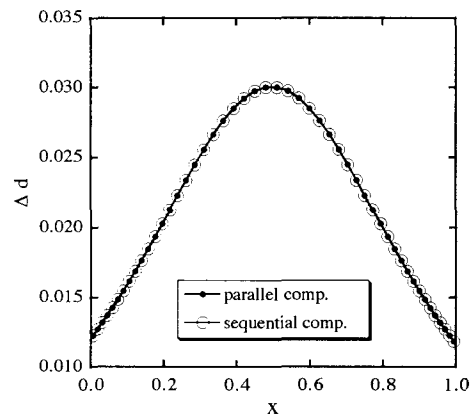


図-5 格子点間隔の分布 (逐次計算と並列計算の結果)

表-1 に内部反復に GCR(1) を用いた場合の外部反復回数と計算時間を示す。局所非定常モデルを用いているため、当初の予想通りプロセッサ数が増加すると外部反復回数が増加する。また、数値実験の条件とした格子数 (50^3) では、プロセッサ数が 64 となると効率が若干低下した。表-2 は、内部反復に GMRES(10) を用いたときの結果である。わずかな差ではあるが、CGR(1) を用いたときの方が計算短くなったので、以降の計算では CGR(1) を内部反復に使用する。

図-6 に局所非定常モデルを用いた場合と通常モデル (図中では usual model) を使用した場合の計算時間の比較を示す。通常モデルでは、内部反復の際に重合格子点上のデータ通信が行われるため、並列計算ではより多くの計算時間を要する結果となった。図-7 に両モデルによる計算時間の比 r を示した。

一方、外部反復回数を局所非定常モデルと通常モデル

デルで比較した結果を図-8に示す。通常のモデルでは、外部反復回数は大きく変化しないが、局所非定常モデルではプロセッサ数の増加とともに外部反復回数も増加する。このような外部反復回数の増加に対して、内部反復における通信負荷の低減の効果がより高かったために、局所非定常モデルが計算時間の点で有利となったと考えられる。

表-1 プロセッサ数と外部反復回数および計算時間
(局所非定常モデル, GCR(1) 使用)

プロセッサ数	外部反復回数	計算時間 (秒)
1	207	858.2
2	249	452.8
8	279	74.8
16	342	27.7
27	362	18.9
36	397	16.0
64	456	17.6

表-2 プロセッサ数と外部反復回数および計算時間
(局所非定常モデル, GMRES(10) 使用)

プロセッサ数	外部反復回数	計算時間 (秒)
1	200	947.8
2	235	546.0
8	290	76.8
16	353	30.5
27	369	20.5
36	403	19.8
64	461	23.0

図-9は逐次計算の計算時間を並列計算のそれを除した計算速度向上比 (speedup) とプロセッサ数の関係を示すものである。この結果からも局所非定常モデルの有効性が明らかである。なお、局所非定常モデルおよび通常のモデルのいずれでも super linear となる領域が見られた。この原因は明らかでないが、単一プロセスが受け持つ格子数が減少すると、2次キャッシュメモリのヒット率が向上することが確認されたため、この効果が super linear となった1つの要因と考えられる。その他の要因として、SMP クラスタという計算機の特徴なども関係していると思われる。

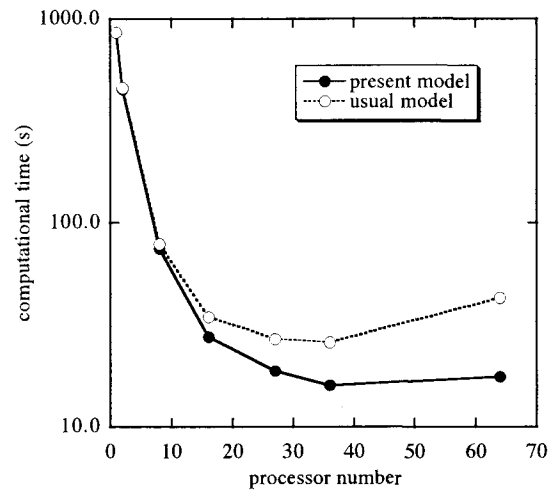


図-6 計算時間とプロセッサ数の関係

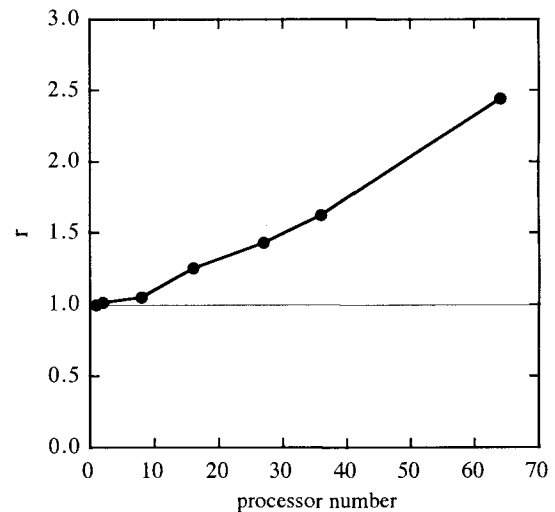


図-7 標準解法と局所非定常モデルの計算時間の比 r とプロセッサ数の関係

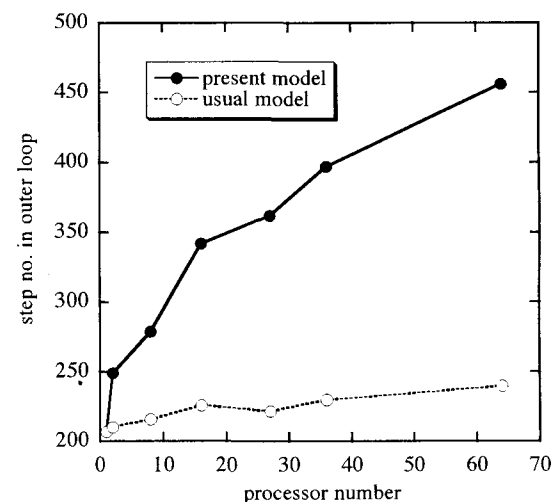


図-8 外部反復回数とプロセッサ数の関係

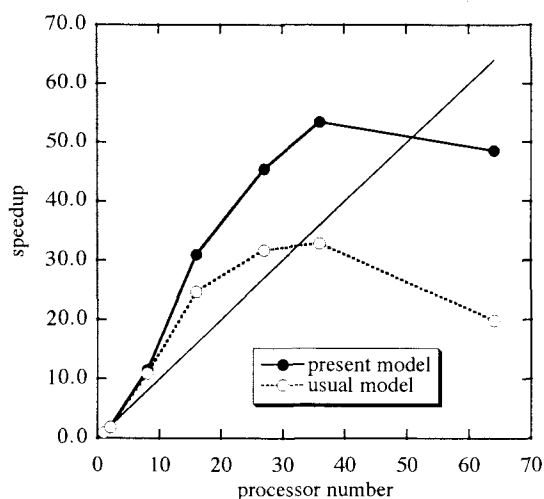


図-9 speedup とプロセッサ数の関係

4. おわりに

本報では、楕円型格子生成法による計算格子の生成を分散メモリ計算機上で高速に行う方法について検討した。分割された領域内で各プロセスが内部反復を行う際に、外部の領域で定常性を仮定する局所非定常モデルを導入し、通信負荷の軽減を図った。京都大学学術情報メディアセンターの HPC2500 を利用してプロセス並列計算を行ったところ、内部反復に通常の解法を用いる場合よりも本報の解法は高速であることが確認された。提案された格子生成法は、境界形状の扱いが重要となる流体計算等に有用と考えられる。

参考文献

- 1) 木村一郎, 細田尚, 音田慎一郎, 富永晃宏. 越流型水制周辺の三次元乱流構造に及ぼす水理パラメータの効果. 水工学論文集, Vol. 48, pp. 661-666, 2004.
- 2) 音田慎一郎, 細田尚, 三浦倫秀. 移動一般座標形での水深積分モデルを用いた水槽内流体振動解析. 応用力学論文集, Vol. 7, pp. 1023-1032, 2004.
- 3) J. F. Thompson, Z.U.A. Qarsi, and C.W. Mastin. (小国力, 川村哲也訳) 数値格子生成の基礎と応用. 丸善株式会社, 1994.
- 4) J. F. Thompson, F. C. Thames, and C. W. Mastin. Automatic numerical generation of body-fitted curvilinear coordinate system for field containing any number of arbitrary two-dimensional bodies. *J. Comput. Phys.*, Vol. 15, No. 1, pp. 299-319, 1974.
- 5) S. Ushijima. Arbitrary Lagrangian-Eulerian numerical prediction for local scour caused by turbulent flows. *Journal of Computational Physics*, Vol. 125, pp. 71-82, 1996.
- 6) 牛島省, 襦津家久. 移動一般座標系を用いたコロケート格子による自由水面流れの数値解析手法. 土木学会論文集, No. 698/II-58, pp. 11-19, 2002.
- 7) 牛島省, 奥山洋平, 竹村雅樹, 襦津家久. 3次元一般座標系における圧力場の並列計算法の提案. 水工学論文集, Vol. 47, pp. 385-390, 2003.
- 8) 牛島省, 奥山洋平, 襦津家久. コロケート格子配置に基づく3次元非圧縮性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, 2003.
- 9) T. Ikohagi and B. R. Shin. Finite-difference schemes for steady incompressible Navier-Stokes equations in general curvilinear coordinates. *Computers and Fluids*, Vol. 19, No. 3/4, pp. 479-488, 1991.
- 10) 牛島省, 竹村雅樹, 襦津家久. 3次元定常流れ場に対するコロケート格子を用いた高次精度陰的計算法. 水工学論文集, Vol. 47, pp. 391-396, 2003.
- 11) 檜山和男 (研究代表者). 国際フォーラム「土木工学における計算力学手法の新展開」の企画調査. 平成16年度科学研究費補助金研究成果報告書, 課題番号: 16636010, pp. 211-250, 平成17年.
- 12) S. C. Eisenstat, H. C. Elaman, and M. H. Schultz. Variational iterative methods for nonsymmetric systems of linear equations. *SIAM J. Numer. Anal.*, Vol. 20, No. 2, pp. 345-357, 1983.
- 13) P. F. Dubois, A. Greenbaum, and G. H. Rodrigue. Approximating the inverse of a matrix for use in iterative algorithms on vector processors. *Computing*, Vol. 22, , 1979.

(2005年4月15日 受付)