

## [共同研究成果]

## 大規模非圧縮性流体シミュレーションの工学問題への応用

高橋 俊<sup>\*1</sup> 石田 崇<sup>\*1</sup> 中橋 和博<sup>\*1</sup>小林 広明<sup>\*2</sup> 岡部 公起<sup>\*2</sup>下村 陽一<sup>\*3</sup> 曾我 隆<sup>\*3\*4</sup> 撫佐 昭裕<sup>\*3\*4</sup><sup>\*1</sup> 東北大学大学院 工学研究科 航空宇宙工学専攻<sup>\*2</sup> 東北大学 サイバーサイエンスセンター<sup>\*3</sup> 日本電気<sup>\*4</sup> 東北大学大学院 情報科学研究科

## 1. 緒言

計算機を用いた流体シミュレーションが始められてから現在までを俯瞰すると、そこには幾度かの変遷があったように見受けられる。シミュレーションが行なわれ始めた当初は最も単純な等間隔の直交格子が広く用いられた[1]。それからさらに実用的な工学問題を扱う際に重要になったのが境界層を正確に解像することである。等間隔の直交格子を用いた場合、流れの Reynolds 数の約 4 分の 9 乗に比例して格子点が増加するが、大多数の実工学問題における Reynolds 数は 100,000~10,000,000 程度となるため、当初の流体シミュレーションでは計算資源上不可能であった。

そこで遠方で格子点を節約して解析することができる構造格子(境界適合格子)が直交格子に代わって用いられるようになった。これにより流体シミュレーションの実用性は大きく向上したが、さらに実用的な応用計算へと研究者たちの興味が移っていくと再度問題が生じた。それはより実際の工学製品に近い 3 次元複雑形状周りの格子生成にかかるコストである。実際の流体シミュレーションに必要なのは数時間から数日であるにも関わらず、その際に用いられる計算格子の生成に数週間から数ヶ月を要することもあった。対応策として重合格子法などにより多数の格子を重ね合わせて複雑形状を表現する手法も考案されたが、計算アルゴリズムが多くの内挿計算により煩雑化するなどの難点もあり、複雑形状周りのシミュレーションに構造格子法が向かないということは本質的に変わらなかった。

そこで注目されたのが非構造格子法である。形状融通性に富むこの手法はほぼ 3 次元の任意形状周りに格子を生成することができ、また格子の粗密制御も比較的容易なことから現在では世界中で用いられている。だがこの非構造格子法にも難点はある。まず、非構造格子は確かに形状融通性に富んでいるのだが、複雑形状周りの格子生成には熟練した技術が必要とされることである。さらに格子点の並びが不規則であるため高い空間解像度を実現することが難しく、解析プログラムも複雑になりがちである。

ここまで流体シミュレーションの中でも特に工学的なアプリケーションに使われる

格子の変遷を追ってきたが、今後果たしてどのように変化していくかを考えてみる。上に述べたように、今まで流体シミュレーションがより実用的な解析を目指して研究が進められてきた結果として、計算格子は直交格子から構造格子を経て非構造格子へと変わってきた。今後さらに高度な応用計算へと発展するとすれば、現在よりも複雑な形状周りの流れ場において、さらに空間解像度の高い解析が求められるであろう。それに対して、前述したように非構造格子法も個人のスキルによって格子の質が変化してしまう点と空間高解像度を達成することの難しさから、今後のより実用的な計算手法として採用するには若干の不安が残る。そこで一度は応用計算の舞台から降りた直交格子に目を向けてみると、前述の要求をほぼ満たしていることが分かる。直交格子法は格子生成が高速かつ容易で、容易に空間高解像度スキームを構築することができ、等間隔の直交格子を活かすことでプログラム構造も簡略化することができる。ただ一つ、格子点数が膨大になりやすいという難点がこの方法の大きな弱点であったが、今後さらに強力になっていく計算機資源を有効に活用すればこの点は解決することができる。このようなコンセプトに基づいて考え出されたのが本研究室で開発している **Building-Cube** 法(直交格子積み上げ法)である[2]。本研究では最先端のベクトル並列計算機である NEC SX-9 を用いて、現状の **Building-Cube** 法の性能を評価、検討し今後の開発に向けた課題を明らかにすることを目的としている。

## 2. Building-Cube 法

### 2.1. 計算格子

**Building-Cube** 法という名前は積み木を意味する **Building-Block** から付けられた。積み木を構成するパーツは直方体や三角柱だが、**Building-Cube** 法におけるパーツは大きさの異なる多数の立方体であり、これを本手法では **cube** と呼んでいる。そしてそれらの **cube** の中にはデカルト座標系の **XYZ** 軸に関して完全に等間隔の直交格子が切られており、これを本手法では **cell** と呼んでいる。

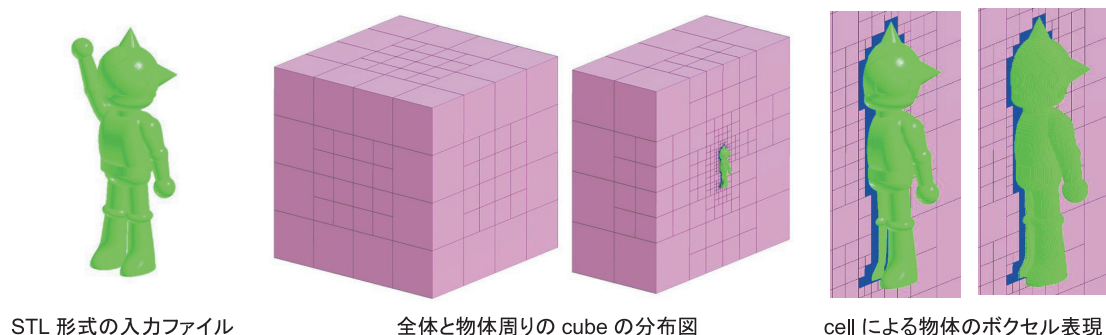


図 1. Building-Cube 法における計算格子

図 1 に Building-Cube 法における計算格子の例を示す。cube の分布は非等間隔の直交格子に似ており，内部には等間隔の直交格子である cell が存在し，物体はこの cell を用いて図のようにボクセル的に表現されている。この計算格子生成手法に関する詳細は石田の文献[3]に譲るが，cube は八分木法により生成され，その際の交差判定には Tomas の方法が用いられている。この格子生成法を 64 ビットのマルチコアのパーソナルコンピュータ上で OpenMP を用いて並列実行すれば，数億 cell という巨大な計算格子であっても僅か数分で生成することができ，非常に高速かつロバストに格子生成を行なうことができる。またその際にユーザが指定するのは主に領域の大きさと最小格子幅のみであり，格子生成の熟練度に関わらず同質の計算格子を容易に生成することができる。

## 2.2. 流体計算手法

本計算では支配方程式に式(1)に示される非圧縮性 Navier-Stokes 方程式を用い，スタガード格子配置を用いて離散化を行なう。

$$\begin{cases} \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} = -\nabla p + \frac{1}{\text{Re}} \nabla^2 \mathbf{u} \\ \nabla \cdot \mathbf{u} = 0 \end{cases} \quad (1)$$

非圧縮性 Navier-Stokes 方程式を解くために本研究では MAC 型解法の一つである Fractional-step 法[4-6]を採用した。この手法では，非圧縮性 Navier-Stokes 方程式を仮の速度場である  $\mathbf{u}^*$  の式，真の速度場  $\mathbf{u}^{n+1}$  の式，さらにそれら 2 つの式を，運動量方程式の両辺の発散をとることで求められた圧力の Poisson 方程式で関連付けて段階的に解く。

仮の速度場の式は以下のように記述される。ここで対流項を  $\mathbf{A}$ ，粘性項を  $\mathbf{B}$  とおき，本研究ではどちらの項も Adams-Bashforth の 2 次精度陽解法で扱っている。対流項と粘性項はそれぞれ 3 次精度風上差分法[7]と 2 次精度中心差分法で離散化されている。

$$\frac{\mathbf{u}^* - \mathbf{u}^n}{\Delta t} = -\left(\frac{3}{2} \mathbf{A}^n - \frac{1}{2} \mathbf{A}^{n-1}\right) + \left(\frac{3}{2} \mathbf{B}^n - \frac{1}{2} \mathbf{B}^{n-1}\right) \quad (2)$$

真の速度場の式は以下の式のように書かれる。この真の速度場の式と仮の速度場の式を足し合わせると元の運動量方程式が導かれる。

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^*}{\Delta t} = -\nabla p^{n+1} \quad (3)$$

上記の 2 つの式を，運動量方程式の両辺の発散をとって導かれる以下の圧力の Poisson 方程式によって関連付ける。ここで右辺に表れる次の時間ステップにおける質量保存の

式の項が満たされると仮定する( $\nabla \cdot \mathbf{u}^{n+1} = 0$ )ことで、質量保存則と運動量保存則を関連付けている。この方程式は2次精度中心差分法で離散化されており、ボクセル的に壁面に Neumann 境界条件を与えている[8].

$$\nabla^2 p^{n+1} = \frac{1}{\Delta t} \nabla \cdot \mathbf{u}^* \tag{4}$$

また、Building-Cube 法における格子は前述したように等間隔の直交格子によるマルチブロック格子であるので、各 cube 間での情報交換が必要になる。そこで本手法では全ての cube に数 cell のオーバーラップ領域を持たせて、また隣り合う cube の幾何学的サイズが2倍以上の差にならないという制約を課している。これにより物体遠方などのサイズの異なる cube 同士間では1次精度の内挿が必要になるが、物体近傍の同じ幾何学的サイズの cube 同士の場合には完全に等間隔の直交格子上で高精度の計算を行なうことができる。

現在のプログラムの構造を図2のフローチャートに示す。各手続きはそれぞれ全ての cube に対して実行される。図2の中で interface と記されているのが cube のオーバーラップ領域の値を更新する手続きである。

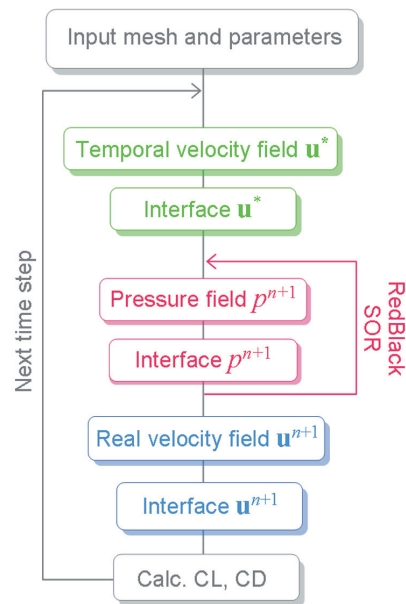


図2. 計算プログラムのフローチャート

### 2.4. プログラムの最適化

本手法では今後さらに大規模化する計算機を効率的に活用するために、各 cube の中に配置されている等間隔の直交格子である cell の数を完全に同じにすることで、データ構造を簡略化している。これにより cube の数と同数のプロセスを使用すれば完全に均一の計算負荷で計算を行なうことができる特長を備えている。現在はまだそれだけのプロセスを使用することはできないが、等間隔の直交格子である cell のループに対して次元化によるベクトル長の伸展など細かな最適化が施されている。また本研究のような MAC 型の解法においては圧力の Poisson 方程式を解く手続きに計算負荷の大部分が費やされるため、この部分のベクトル化が不可欠となる。そこで本手法では超平面法や Red-Black のカラーリングを適用した SOR 法でベクトル演算を活用している。結果として、現在ではシミュレーション全体で 99.9% のベクトル化率を達成するまで最適化が施されている。

### 3. 解析結果

#### 3.1. Ahmed body 周りのシミュレーション

まず、自動車の簡易モデルとして実験的にも計算的にも広く研究が行なわれている Ahmed body 周りのシミュレーションを行なった[9-14]。計算格子は  $32^3$  の cell が含まれる 3,048 の cube から構成されており、総 cell 数は 99,876,864 である。Reynolds 数は主流方向の長さを基準長として  $2.8 \times 10^6$  で、最小格子幅は  $1.22 \times 10^{-3}$  である。計算格子の全体像と物体近くを図 3 に示すが、これは効率的に可視化するために実際に計算に用いられた格子を 1/8 に圧縮したものである。外部の計算条件としては、床面に滑りなし境界条件を与えて他の外部境界では線形外挿を用いている。

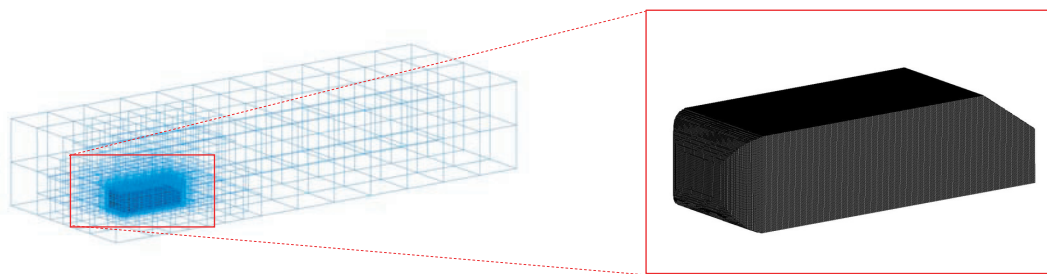


図 3. Ahmed body 周りの計算格子(1/8 に粗視化)

Ahmed body は後部の斜めの傾斜の角度によって様々な流れの様相を検討するために用いられるモデルであるが、一般的によく用いられるのは  $25^\circ$  と  $35^\circ$  のケースで、本計算では  $35^\circ$  のケースを用いて計算を行った。このケースでは後部で大規模な剥離が生じるのが特徴である。図 3 に流れが発達した後で得られた時間平均の結果を示すが、本計算でもその特徴が可視化図のようによく捉えられている。またモデルの中心の平面上における速度プロファイルに関して実験値との比較を行った。赤の線が本計算から得られた結果で、黒の円が実験結果を示しているが、比較的良好な一致が見てとれる。これらの結果から、現在のプログラムによって妥当な結果が得られることを確認した。

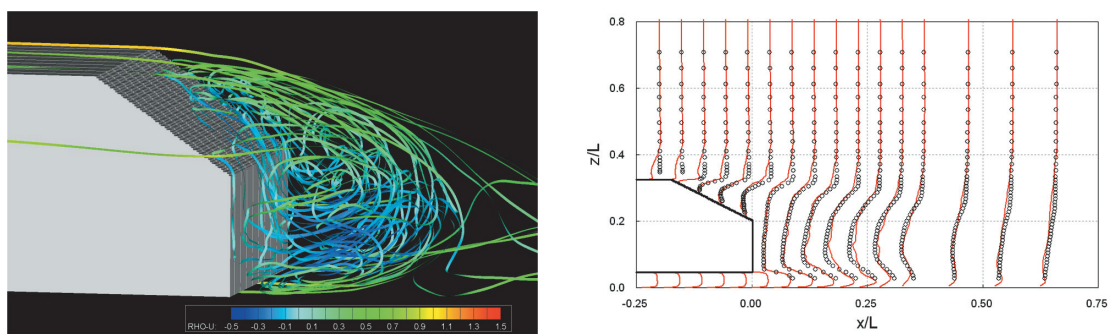


図 4. Ahmed body 周りの可視化結果と速度プロファイルの比較



### 3.2. フォーミュラカー周りのシミュレーション

前節で記した Ahmed body は比較的単純な形状であったが、次により高度な応用を目指してフォーミュラカー周りのシミュレーションを行なった[15]. 計算格子は  $32^3$  の cell が含まれる 5,930 の cube から構成され、総 cell 数は 194,314,240 である. Reynolds 数は主流方向の長さを基準長として  $14.9 \times 10^6$ , 最小格子幅は  $7.32 \times 10^{-4}$  である. ちなみに総 cell 数が約 2 億という膨大な格子であるが、この格子も Windows のマルチコア PC により僅か数分で作成された. 外部の計算条件としては、先ほどと同様に床面に滑りなし境界条件を与えて他の外部境界では線形外挿を用いている. 図 5 に物体表面の cell による近似と、時間平均した結果の流線を示す. より現実的なシミュレーションにはタイヤの回転や床面を考慮する必要があるが、今後はそれらに取り組む予定である.

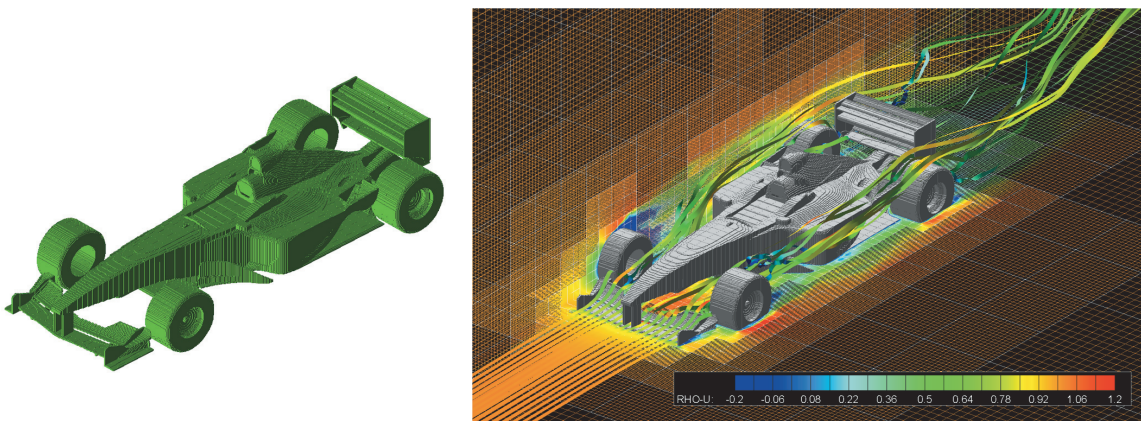


図 5. フォーミュラカー周りの格子と可視化結果(1/8 に粗視化)

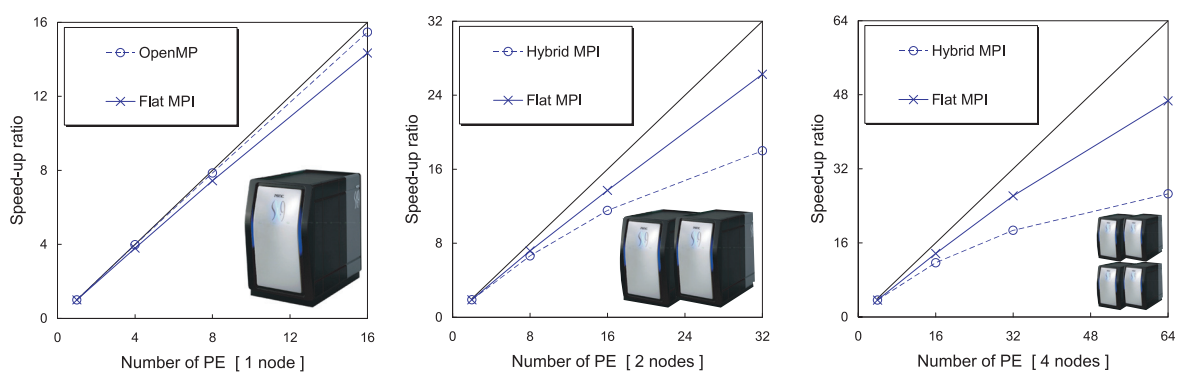


図 6. フォーミュラカー周りの並列シミュレーションにおける速度向上率

図 6 にこのシミュレーションで得られた速度向上率を示した. Flat MPI とは、計算格子を PE と等しい数の領域に分割し、MPI のみで並列計算を行なう処理を指し、一方で

Hybrid MPI とは、計算格子をノードの数に領域分割して各ノード間では MPI による並列計算を行ない、同時に各ノード内では OpenMP により並列計算を行なう処理を指している。Flat MPI と比較して Hybrid MPI の効率が低いのは、格子の構造上本手法では領域分割を cube を基準として容易に行なうことができるのだが、現在はその領域分割において飛び地が発生することを許容していることと、Hybrid MPI の方が一つの領域に属する cube が多くなり MPI を用いた情報交換の量が Flat MPI よりも多くなっているためであると考えられる。これを解消するために、現在は各領域に飛び地を発生させない手法や通信の隠蔽など、より効率の良いアルゴリズムを模索中である。

#### 4. 結言

今後の大規模化する計算機とより複雑化する工学問題への応用を踏まえて、等間隔の直交格子法とボクセル法を基礎とした Building-Cube 法による非圧縮流体シミュレーションのコードを開発した。またそれをベクトル並列型のスーパーコンピュータである NEC SX-9 において実行し、妥当な解の精度と性能を確認することができた。今後、更に並列効率を高めて実アプリケーションに応用していく。

#### 謝辞

本文中のシミュレーションは、東北大学サイバーサイエンスセンターにより、コードのベクトル化や並列化、そして様々なデータの測定などの多大な支援を頂いた。

#### 参考文献

- [1] Harlow, F. H. and Welch, E., "Numerical Calculation of Time-Dependent Viscous Incompressible Flow of Fluids with Free Surface," *Physics of Fluids*, Vol. 8, pp. 2182-2189, 1965
- [2] Nakahashi, K., "High-Density Mesh Flow Computations with Pre-/Post-Data Compressions," AIAA paper, 2005-4876, 2005
- [3] Ishida, T., Takahashi, S. and Nakahashi, K., "Efficient and Robust Cartesian Mesh Generation for Building-Cube Method," *Journal of Computational Science and Technology*, Vol. 2, pp. 4, 2008
- [4] Kim, J. and Moin, P., "Application of a Fractional-Step Method to Incompressible Navier-Stokes Equations," *Journal of Computational Physics*, Vol. 59, pp. 308-323, 1985
- [5] Perot, J. B., "An Analysis of the Fractional Step Method," *Journal of Computational Physics*, Vol. 108, pp. 51-58, 1993
- [6] Dukowicz, J. K. and Dvinsky, A., "Approximate Factorization as a High Order Splitting for the Implicit Incompressible Flow Equations," *Journal of Computational Physics*, Vol. 102, pp. 336-347, 1992

- [7] Kawamura, T. and Kuwahara, K., “Computation of High Reynolds Number Flow around Circular Cylinder with Surface Roughness,” AIAA paper, 84-0340, 1984
- [8] Akasaka, K. and Ono, K., “An Implementation of Boundary Condition of Incompressible Flow Solver for Complex Geometries on Voxel Method,” *Transactions of JSCEs*, No. 20060024, 2006
- [9] Ahmed, S. R., Ramm, G. and Faltin, G., “Some Salient Features of the Time Averaged Ground Vehicle Wake,” SAE paper, No. 840300, 1984
- [10] Lienhart, H., Stoots, C. and Becker, S., “Flow and Turbulence Structures in the Wake of a Simplified Car Model (Ahmed Model),” In DGLR Fach Symposium der AGSTAB, 2000
- [11] Fares, E., “Unsteady Flow Simulation of the Ahmed Reference Body using a Lattice Boltzmann Approach,” *Computers and Fluids*, Vol. 35, pp. 940-950, 2006
- [12] Hinterberger, C., Garcia-Villalba, M. and Rodi, W., “Large Eddy Simulation of Flow around the Ahmed Body,” *Lecture Notes in Applied and Computational Mechanics / The Aerodynamics of Heavy Vehicles: Trucks, Buses, and Trains*, edited by McCallen, R., Browand, F. and Ross J., Springer Verlag, ISBN: 3-540-22088-7, 2004
- [13] Krajnovic, S. and Davidson, L., “Large Eddy Simulation of the Flow around an Ahmed Body,” *Proceedings of HTFED04, ASME Heat Transfer/Fluids Engineering Summer Conference*, July 11-15, Charlotte, North Carolina, USA, 2004
- [14] Howard, R. J. A. and Pourquie, M., “Large Eddy Simulation of an Ahmed Reference Model,” *Journal of Turbulence*, Vol. 3, No. 12, 2002