

AMR 法による複雑せん断乱流の LES 解析

松尾裕一 (JAXA), 桑原匠人, 中森一郎 (アドバンスソフト)

LES analysis of complex turbulent shear flows with an AMR method

by

Yuichi MATSUO, Takuhito KUWABARA, and Ichiro NAKAMORI

ABSTRACT

In this paper, an LES analysis methodology with an AMR approach are described for complex turbulent shear flows like jets, mixing layers, and shear layers in separated regions, and numerical examples are presented. In the present AMR approach, for numerical simplicity and practical use, we adopt a block-based method where a structured mesh in each block, a body-fitted coordinate system and a self-similar tree-based hierarchical data structure are used. The numerical issues are discussed to apply the present AMR approach to large-scale parallel computations.

1. はじめに

実用問題の LES 解析においては、高解像度スキームの適用が容易なことから、格子としては構造格子の利用が未だ主流と言って良いであろう。非構造格子による LES 解析の事例も増えて来ているようだが、精度を上げるのに工夫が要するため、未だ発展途上[1]である。

構造格子を用いる際、図 1 に例示したように、格子点が本来必要のないところに集中してしまったり、逆に必要ないところうまく集められないといった場合がある。特に、LES 解析では、格子サイズ自体が Subgrid Scale (SGS) 渦粘性のパラメータとなるため、捉えるべき現象に対し空間格子解像度をできるだけ一定に保つことが重要であると言われている[2]。また、LES のような非定常解析においては、ある程度多くの計算ステップ数が要求されるため、無用な計算時間の増加を招かないためにも格子点数は有効に使いたいところである。

こうした場合に、近年、解適合格子細分化法 (Adaptive Mesh Refinement; AMR) と呼ばれる方法が注目されている。AMR 法は、必要な領域に対してのみ格子を細分化する方法である。解適合法は一般に、 r 法、 h 法、 p 法の 3 種類の方法に分類される。 r 法 (r -refinement) とは、格子点を移動させる方法、 h 法 (h -refinement) とは、AMR のように格子を局所的に細分化する方法、 p 法 (p -refinement または p -enrichment) とは、局所的にスキームの精度を高くする方法のことを指す。 h 法と p 法を組み合わせると hp -adaptation として使われることもある[3]。 h や p は、 h を格子間隔、 p を空間精度とすると、打ち切り誤差は $O(h^p)$ と表されることに由来する。

他方、格子点配置や計算時間の問題は、いずれ計算機の進歩が解決してくれるだろうという考え方もある。しかし、3 次元問題・同一の計算時間を仮定したとき、解像度を 2 倍にするためには、空間各方向に格子点数 2 倍、時間刻み

半分で $2^4=16$ 倍、解像度を 4 倍にするためには $4^4=256$ 倍の性能向上が必要となる。しかし一方で、計算機の性能の伸びは、(ムーアの法則によれば) 5 年で 10 倍、技術改善要素を入れても 5 年でせいぜい数 10 倍程度であり、計算機の性能向上に頼り過ぎるのは必ずしも現実的でない。

本報告では、剥離乱流境界層、伴流 (ウェーク)、噴流 (ジェット) 等の複雑せん断乱流を伴う実用問題の LES 解析を、AMR 法により効率的に行う方法・解析事例について述べ、その大規模並列計算への適用の可能性について考察する。

2. AMR (解適合格子細分化) 法によるアプローチ

AMR 法は、1980 年代から現在に至るまで様々な手法が提案されているが、これらは主に

- A) 直交デカルト格子系を基盤として、セル (格子) 単位で細分化する方法...セルベース AMR
 - B) 構造格子上にブロック (領域) を定義し、ブロック単位で細分化する方法...ブロックベース AMR
- の 2 種類に大別される。

前者の AMR は、Berger と Oliger による先行研究[4]まで遡ることができ、諸量の空間変化や時間変化に応じてセル単位で随時、分割 (Refinement) と結合 (De-refinement または Coarsening) を繰り返すものである (図 2(a))。主に火炎や爆轟、噴流、自由界面のような物理変化の激しい部分が時々刻々移り変わって行くような場合の解析に有効な手法であり、Aftosmis[5]や Wang[6]により航空宇宙分野にも応用されている。しかし、物体壁面を含む解析には、カットセルや境界層専用格子を使うなどの工夫が必要であり、データ構造や前後処理についても独自の開発が必要となる。

一方、後者の AMR は、AMR をセル単位ではなくブロック単位 (図 2(b)) で行うものであり、ブロックの中では既存の構造格子ソルバーが使える、ブロック間通信は境界値のみで済む、データ構造がシンプル、並列化が容易など、比較的簡単に AMR のメリットを享受できる。ただ、セル単位ほどの形状適合性はない。翼周りの遷音速非粘性流れに適用した構造格子の例として Dudek ら[7]の計算や、NS 方程式を支配方程式として翼まわりと鈍頭物体まわりの圧縮性流れを計算した Steinhilber ら[8]の例がある。最近では、SAMRAI[9]、AMROC[10]といった汎用 AMR ライブラリも開発されている。

ここで、AMR を適用する際によく使われる技法や一般原則について述べる[10]。AMR で利用されるデータ構造に「木構造 (Tree data structure)」がある。2 次元では「四分木 (Quadtree)」、3 次元では「八分木 (Octree)」と呼ばれるデータ構造が良く使われる。1 つのセルの各辺を半分にすると 1 つの親セル (Parent) に対して、2 次元では 4 つの子セル

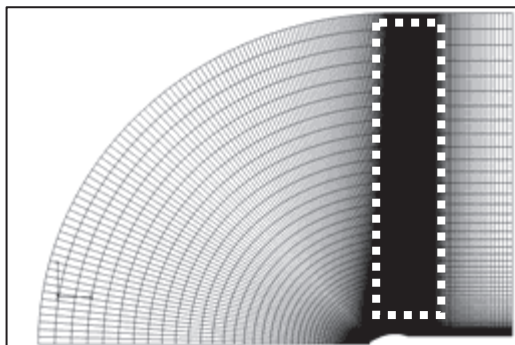
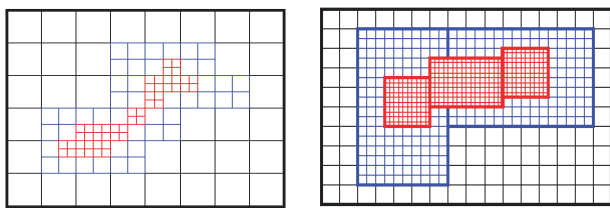


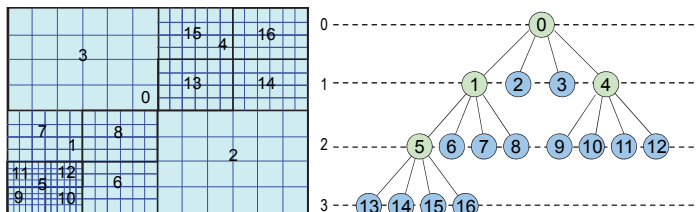
図 1 構造格子使用時に格子が遠方に集中した例

(Children), 3次元では8つの子セルが生まれる. CやFortran90のポインタによる連結リスト(Linked list)を使えば木構造は容易に実装できる. 単位がセルではなくブロックの場合でも考え方は同じである. 「適切な入れ子(Proper nesting)」とは, AMR境界での精度を確保するためにAMRの格子サイズの変更は必ず1段階に留めるという原則である. AMR境界は, ハングングノードになる. 「空間充填曲線(Space filling curve)」は, 木構造データを並列計算のために1次元配列に並べ替えるために使われ, Z曲線, Morton曲線, Hilbert曲線などがある. 「ガードセル(Guard cell)」とは, ブロックベースAMRでブロック境界のデータ交換のために設けられる数点のステンシルから成る領域を指す. AMRでは, RefinementやDerefinementの度に親から子, 子から親への格子間のデータ補間が必要になる. このうち親(粗い格子)から子(細かい格子)への補間を「延長(Prolongation)」, その逆を「制限(Restriction)」と呼ぶことがある.



(a) セルベース (b) ブロックベース

図2 AMR格子のタイプ



(a) 自己相似的AMR格子 (b) ツリー型データ構造

図3 2次元におけるブロックAMR格子例

我々は, LES解析において格子を有効利用するとともに必要な場所に動的に集めることを視野に, ブロックAMR法を用いた並列化LESコードを開発してきた[11]. 実用LES解析に適用することを目指し, 八分木のブロック化アルゴリズムに基づくAMR法を開発・検証するとともに, MPI並列化及び初期マルチブロックへの適用やメモリの削減などの実用に向けた改善に取り組んできた[12].

データ構造として, 並列化に合わせて負荷バランス維持に都合良い均等ブロック再分割が可能な自己相似的AMRと八分木構造(Octree)を採用した. 図3は, 2次元の自己相似の木構造(四分木)を例示したものである. 例えばBlockIDという番号のブロックは,

親ブロック Parent(BlockID)
 子ブロック Child(LocalID, BlockID), $1 \leq \text{LocalID} \leq 8$
 隣接ブロック Neigh(FaceID, BlockID), $1 \leq \text{FaceID} \leq 6$

を持ち, そのブロックが

Rlevel(BlockID) AMRレベル
 Lbtype(BlockID) 最末端かどうか
 Flag_refine(BlockID) AMRを施すかどうか

というフラグを持つようにすれば, AMRの基本アルゴリズムを構成できる. 図4は, ある状況下での配列間や番号付けの関係を示したものである. AMR適用の際には図5に示したように, ガードセル充填の作業が必要になるが, 補間に際して単調性が保持されるような工夫を施している. すなわち, 図6のような2次元のAMR境界を考えたとき,

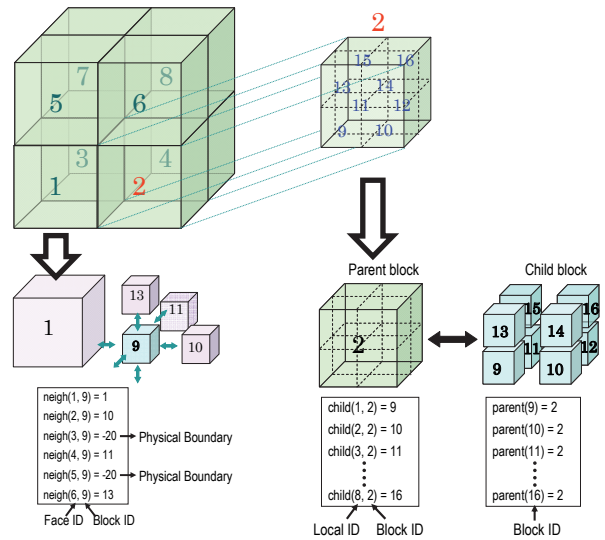


図4 AMRに係る配列間の関係事例

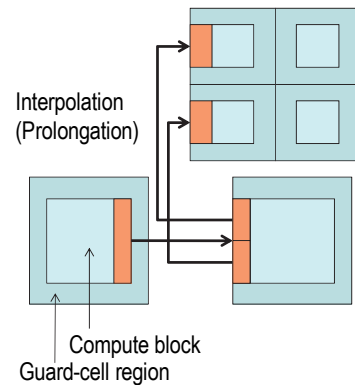


図5 ガードセル充填のイメージ(2次元)

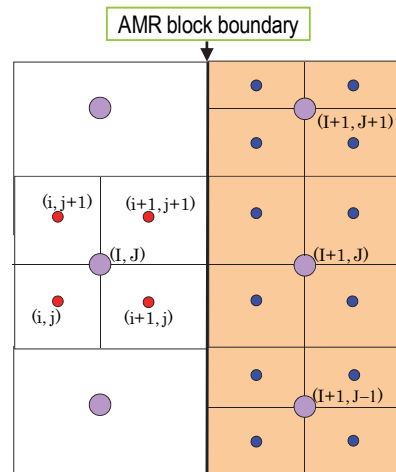


図6 AMR境界における格子点の関係

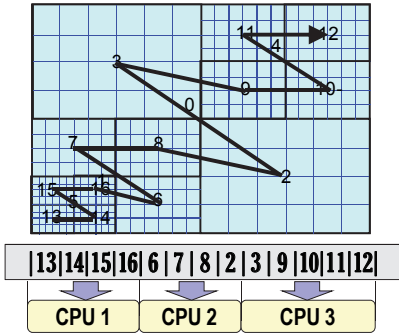


図 7 Z 曲線によるオーダリング

$$\begin{aligned}
 Q_{i+1,j+1}^{l+1} &= Q_{i,j}^l + 0.25\Delta_\xi + 0.25\Delta_\eta \\
 Q_{i+1,j}^{l+1} &= Q_{i,j}^l + 0.25\Delta_\xi - 0.25\Delta_\eta \\
 Q_{i,j+1}^{l+1} &= Q_{i,j}^l - 0.25\Delta_\xi + 0.25\Delta_\eta \\
 Q_{i,j}^{l+1} &= Q_{i,j}^l - 0.25\Delta_\xi - 0.25\Delta_\eta
 \end{aligned}$$

とする。ここに、

$$\begin{aligned}
 \Delta_\xi &= 0.5(\text{sign}(\Delta_\xi^+) + \text{sign}(\Delta_\xi^-)) \min(|\Delta_\xi^+|, |\Delta_\xi^-|) \\
 \Delta_\eta &= 0.5(\text{sign}(\Delta_\eta^+) + \text{sign}(\Delta_\eta^-)) \min(|\Delta_\eta^+|, |\Delta_\eta^-|) \\
 \Delta_\xi^+ &= Q_{i+1,j}^l - Q_{i,j}^l, \Delta_\xi^- = Q_{i,j}^l - Q_{i-1,j}^l \\
 \Delta_\eta^+ &= Q_{i,j+1}^l - Q_{i,j}^l, \Delta_\eta^- = Q_{i,j}^l - Q_{i,j-1}^l
 \end{aligned}$$

であり、 $Q_{i,j}^{l+1}$ は、 $\{l+1\}$ AMR レベル (細かい格子) における格子点 (i,j) の物理量を、 $Q_{i,j}^l$ は、 $\{l\}$ レベル (粗い格子) における格子点 (i,j) の物理量をあらわす。自己相似木構造の下では、各ブロックの格子点数は同一数になるので、「領域分割」の考え方で並列化を行うことができる。ただし、ここでは複数ブロック→1CPU という割付を可能とするとともに、再分割 Refinement によってブロックが新たに生成される際、各 CPU の負荷バランスを一定に保つように、各 CPU にできるだけ均等に、かつ、物理的に近いブロックを配置する Z 曲線による並べ替え (オーダリング) を行っている (図 7)。また、各ブロック周囲のガードセル充填において、各面の転送データ量は同じでなく、よってブロック毎に通信すると効率が悪いので、面 ID が 1 と 2, 3 と 4, 5 と 6 の 3 グループに分けて一度に通信を行うようにしている。並列化コードは、MPI と Fortran90 で作成し、流体ソルバーの部分をカセット式に交換可能なように、ソルバー部と AMR 部は分離したプログラム構造としている。図 8 に、並列化コードの処理の流れを示した。

さらに、実用問題への適用を考慮し、親格子を一個ではなくマルチブロック化し、必要な場所に確実に AMR が適用できるようにしている。また、曲線座標で定式化し、複雑形状や境界層の扱いを可能としている。時間積分は、基本は陽解法であるが、ブロック毎に陽解法と陰解法を選択可能とし、AMR により格子が細分化された場合や境界層内での時間刻みが小さく成り過ぎるのを回避している。また、Rewind と呼ばれる機能やエラーによる AMR 判定もオプションにより選択可能としている [12]。

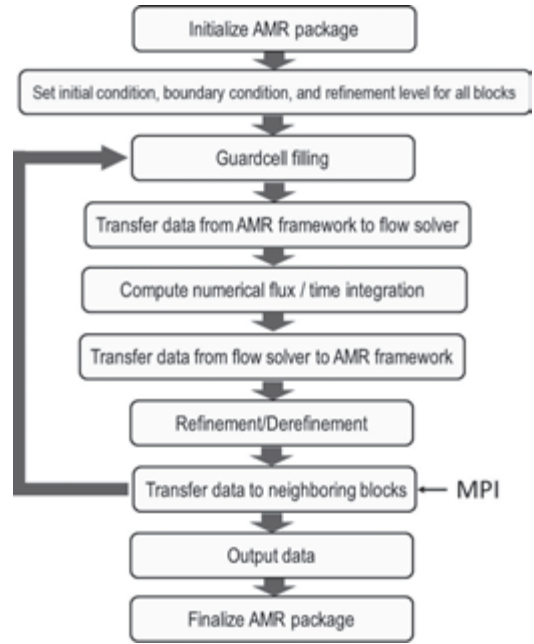
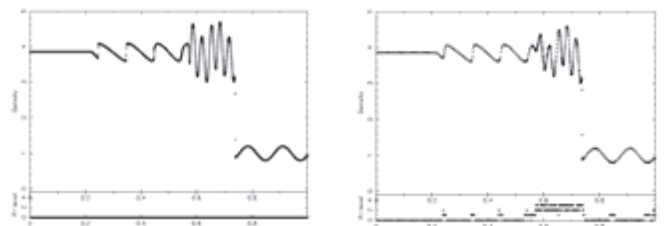


図 8 ブロックベース AMR 法のフローチャート

3. AMR 法による解析事例

図 9 は、AMR 法の基本特性を調べるために、1 次元の Shu-Osher のエントロピー波 [13] を解いた結果であり、 $T=1.8$ における密度分布を示している。図 9(a) は、2,048 点の一樣格子で解いたもの、図 9(b) は、256 点の一樣格子から出発して 3 段階の AMR を施した結果である。AMR では、トータルで 505 点しか使っていないにもかかわらず、衝撃波付近の波構造が一樣格子と同程度の解像度で捉えられているのがわかる。

図 10 に、2 次元問題への適用事例を示す。図 10(a) は、ステップのある風洞内の超音速非粘性流れ ($M_\infty=3$) [14] の解析結果である。240×80 の初期格子を 5×5 のブロック (1 ブロックあたり 48×16) に分割し、ブロックあたり $\Delta M_{\max} > 0.15$ の条件下で 2 段階の AMR を適用したもので、 $T=4$ の密度等高線を示している。衝撃波の細かい構造や 3 重点から出るせん断層の様子が捉えられている。図 10(b) は、二重マッハ反射問題 ($M_\infty=10$) [14] を解いたものである。120×30 の初期格子を 3×3 のブロック (1 ブロックあたり 40×10) に分割し、ブロックあたり $\Delta M_{\max} > 0.2$ の条件下で 3 段階の AMR を適用したもので、 $T=0.2$ の密度等高線を示している。三重点やせん断層が不安定になっている様子が捉えられている。図 10(c) は、上下に対向する混合層 ($M_{\text{upper}}=0.5$, $\rho_{\text{upper}}=1$, $M_{\text{lower}}=0.5$, $\rho_{\text{lower}}=2$) の混合過程を解いたものである。64×64 の初期格子を 8×8 のブロック (1 ブロックあたり 8×8) に分割し、ブロックあたり $\Delta Q_{\max} > 2.0$ の条件下で 2 段階の AMR を適用したもので、ある瞬間の渦度の等高線を示しており、混合層における複雑な渦の様子が捉えられている。



(a) 一樣格子 (2,048 点) (b) AMR 格子 (3 段階)

図 9 Shu-Osher エントロピー波の 1 次元解析結果

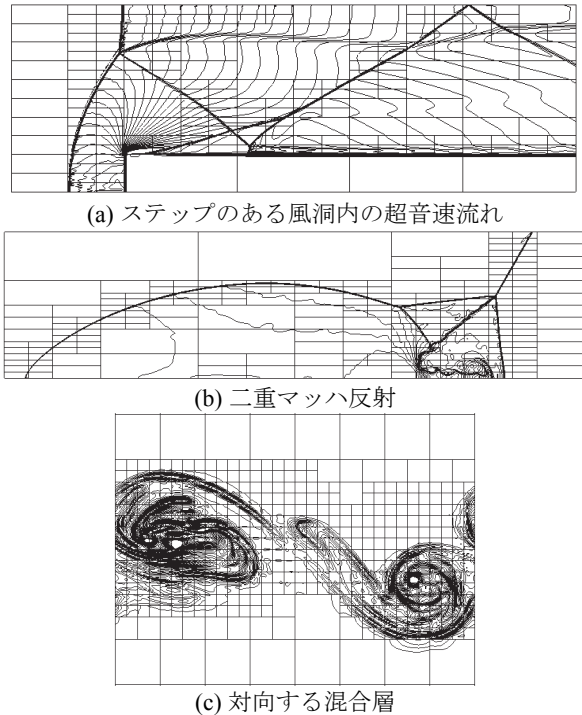


図 10 2次元問題のAMRによる解析結果

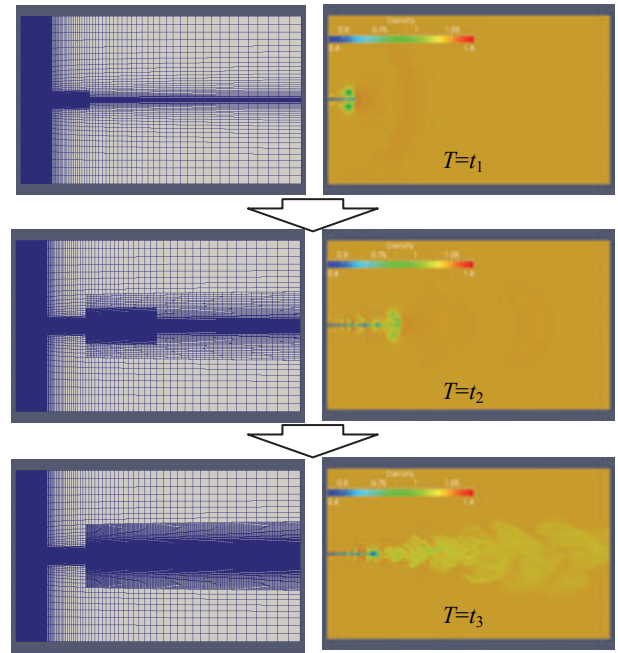


図 13 同軸噴流の3次元解析結果

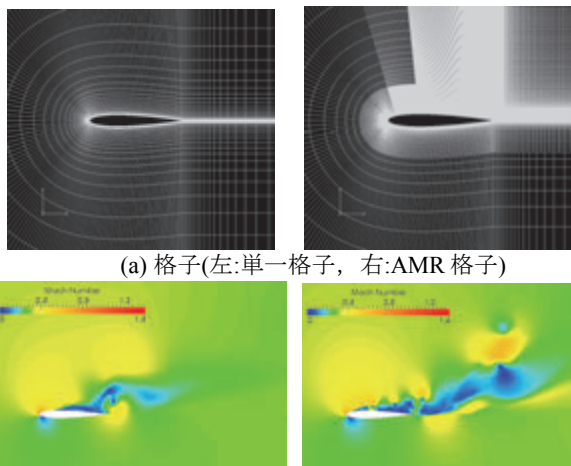


図 11 翼まわり剥離流れの2次元解析結果

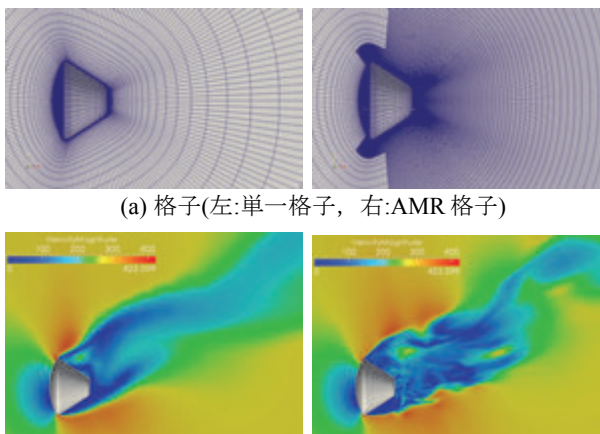


図 12 大気圏再突入物体の遷音速流3次元解析結果

図 11 は、NACA0012 翼型の高迎角剥離流を、 $M_\infty=0.3$, $\alpha=20^\circ$, $Re=10^6$ の条件で解いたものである。252×64 の初期格子を 14×4 のブロック (1 ブロックあたり 18×16) に分割し、3 段階の AMR を適用しており、AMR 細分化後は 518 ブロックになっている。図では、ある瞬間のマッハ数分布を単一格子(左側)と AMR 格子(右側)で比較しているが、AMR 格子では細かい剥離渦が鮮明に捉えられている。

図 12 は、大気圏再突入物体モデル(ORION CM)[15]を過ぎる遷音速流を、 $M_\infty=0.8$, $\alpha=20^\circ$, $Re=10^7$ の条件で解いたものである。96 (流れ方向)×60 (半径方向)×40 (周方向)の格子を 3×3×2 のブロック 1 ブロックあたり 32×20×20) に分割し、後流部分に 2 段階の AMR を適用し、トータル 1,308 ブロックを用いている。図は、中央断面におけるある瞬間の速度分布を単一格子(左側)と AMR 格子(右側)で比較しているが、AMR 格子では後流の細かな渦構造が捉えられている。

図 13 は、コア流 $M_\infty=0.58$, ファン流 $M_\infty=0.76$, 単位長さ $Re=5 \times 10^6$ のエンジン出口を模擬した同軸噴流を $100D \times 30D \times 30D$ (ただし、 D はコア流の直径)の計算領域で解いたものである。128 (流れ方向)×40 (半径方向)×32 (周方向)の格子を $4 \times 2 \times 4$ のブロック (1 ブロックあたり $32 \times 20 \times 8$) に分割し、ブロックあたり $\Delta M_{\max} > 0.01$ の条件下で 2 段階の AMR を適用したもので、図は、中央断面における瞬間的な密度分布の時間による変化を示している。時間が進むにつれ、格子が分割 (最終的に 752 ブロック) され、噴流の細かい渦構造が捉えられている。

上記では、AMR 法が有効な事例をいくつか示したが、AMR 法が比較的有効なのは、流体運動や渦構造を捉えるような場合であることに注意する。音の伝播を捉えるような場合は、AMR 格子境界 (ハンギングノード) から振動が出やすいので注意が必要である。また、ソルバーとしては、通常の圧縮性 NS ソルバー (空間: Roe/MUSCL or WENO, 時間: RK/LUADI) を用いていることを付記する。

4. HPC 計算への適用に関して

ここでは、本 AMR 法の大規模並列計算への適用の可能性について考察する。データ構造として本研究のような自己相似木構造を用い、領域の外側だけ通信を行うようにすれば、凝ったことをしない限り並列計算には比較的向いていると思われる。しかし、並列化率が高くなればアムダールの法則により、並列度を上げて高速化は図れない。図 14 は、図 13 の同軸噴流問題の 2 レベルの AMR を施した状態において、問題規模を変えずに並列度によるスケール性（強スケール特性）を、JAXA スパコンにおいて測定した結果である。横軸にプロセス数、縦軸に各処理の計算時間を取っている。ここに、Total : 全処理時間、Delta_T : 時間ステップの計算時間、Data_Copy : データコピーの時間、Solver : ソルバーで解いている時間、Guardcell : ガードセル充填に要する時間、BC : 境界条件設定の時間、Refinement : 細分化に要する時間をあわす。直線（線形）からのずれが認められるのは、ガードセル充填、時間ステップ計算、細分化の部分であり、その他の部分は 256 並列まではほぼ直線である。このことから、直線でない部分の処理量は全体の数%程度であり、従って 95% 以上の並列化率があるものと思われる。よって、プロセスあたりの負荷を一定にして規模を大きくしていく弱スケールリング特性に関しては、おそらく 1000 並列以上でもスケールする（並列性能がサチらない）と予想される。こういう特性は、大規模並列計算を行う上で望ましい性質である。

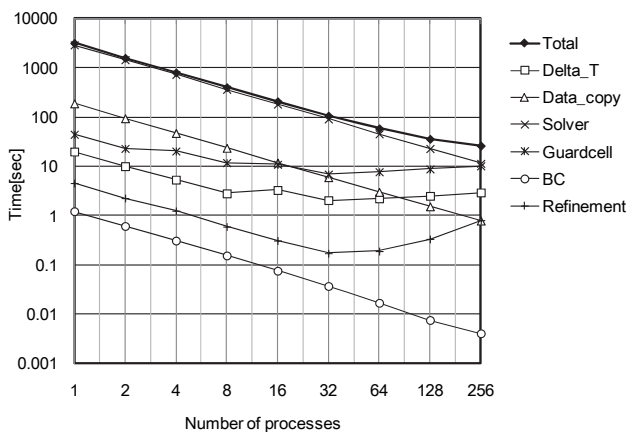


図 14 同軸噴流問題における強スケール特性

5. おわりに

本稿では、剥離乱流境界層、伴流（ウエーク）、噴流（ジェット）等の複雑せん断乱流を伴う実用問題の LES 解析を AMR 法により効率的に行う方法とそれを用いた解析事例について述べた。自己相似木構造等を用いることにより、解適合 AMR 法のメリットを活かしつつも並列計算にも適応できるアルゴリズムを構築できることを示した。また、その大規模並列計算への適用可能性について考察し、スケール性については望ましい性質を有していることを示した。

今後は、実測によるスケール性の検証を進めるとともに、流体の現象を解明するツールとしての本手法の可能性を検証する。具体的には、実験との比較検討を通じて定量的な予測精度の向上の確認、ならびにその向上に寄与している流れの機構を AMR によってどの程度抽出することが可能であるかの確認を実施する予定である。

参考文献

- [1] Winkler, C. M., Dorgan, A. J. and Mani, M.: A Reduced dissipation Approach for Unsteady Flows on Unstructured Grid, AIAA Paper 2012-0570, 2012.
- [2] Spalart, P. R.: Strategies for turbulence modeling and simulations, *International Journal of Heat and Fluid Flow* **21**, 2000, pp.252-263.
- [3] Burgess, N. K. and Mavriplis, D. J.: An *hp*-Adaptive Discontinuous Galerkin Solver for Aerodynamic Flows on Mixed-element Meshes, AIAA Paper 2011-490, 2011.
- [4] Berger, M. J. and Olinger, J.: Adaptive Mesh Refinement for Hyperbolic Partial Differential Equations, *Journal of Computational Physics* **53**, 1984, pp.484-512.
- [5] Aftosmis, M. J., Berger, M. J. and Adomavicius, G.: A Parallel Multilevel Method for Adaptively Refined Cartesian Grids with Embedded Boundaries, AIAA Paper 2000-0808, 2000.
- [6] Wang, Z.J. and Chen, R.F.: Anisotropic Cartesian Grid Method for Viscous Turbulent Flow: AIAA Paper 2000-0395, 2000.
- [7] Dudek, S. A. and Colella, P.: Steady-State Solution-Adaptive Euler Computations on Structured Grids, AIAA Paper 98-0543, 1998.
- [8] Steinthorsson, E., Modiano, D. and Colella, P.: Computations of Unsteady Viscous Flows Using Solution-Adaptive Mesh Refinement in Curvilinear Body-Fitted Grid Systems, AIAA Paper 94-2330, 1994.
- [9] Wissink, A. M., Hornung, R. D., Kohn, S. R., Smith, S. S., and Elliott, N. S.: Large Scale Structured AMR Calculations Using the SAMRAI Framework, SC01 Proceedings, 2001.
- [10] Deiterding, R.: Block-Structured Adaptive Mesh Refinement - Theory, Implementation and Application, ESAIM: Proceedings, **34** (2011), pp.97-150.
- [11] 松尾裕一, 池知直子, 中森一郎: LES 解析のための並列化 BAMR コードの開発, 第 21 回数値流体力学シンポジウム講演集, E1-6, 2007.
- [12] 松尾裕一, 桑原匠史, 池知直子, 中森一郎: BAMR 法に基づく並列 LES コードにおける計算効率の改善, 第 24 回数値流体力学シンポジウム講演集, E4-5, 2010.
- [13] Shu, C.-W., and Osher, S.: Efficient Implementation of Essentially Non-Oscillatory Shock Capturing Schemes, II, *Journal of Computational Physics* **83** (1989), pp.32-78.
- [14] Woodward, P., and Colella, P.: The Numerical Simulation of Two-Dimensional Fluid Flow with Strong Shocks, *Journal of Computational Physics* **54**, 1984, pp.115-173.
- [15] Murphy, K. J., Bibb, K. L., Brauckmann, G. J., Rhode, M. N., Owens, B., Chan, D. T., Walker, E. L., Bell, J. H., and Wilson, T. M., "Orion Crew Module Aerodynamic Testing," AIAA Paper 2011-3502, 2011.