

# 直交格子積み上げ法による 大規模流体計算データの圧縮に関する研究

坂井玲太郎<sup>1</sup>, 佐々木大輔<sup>2</sup>, 中橋和博<sup>3</sup>  
<sup>123</sup> 東北大学大学院工学研究科航空宇宙工学専攻

## Large-Scale Data Compression Method for Building-Cube Method

by  
Ryotaro Sakai, Daisuke Sasaki, and Kazuhiro Nakahashi

### ABSTRACT

A compression method for large-scale flow field data has been developed. This method employs discrete wavelet transform, quantization, and entropy coding for compression. Discrete wavelet transform divides the data into approximation part and detail part, then quantization and entropy coding reduce the amount of data. Compressed data (Plot3D format) is computed by an incompressible Navier-Stokes flow solver using block-structured Cartesian mesh, Building-Cube Method. The quantization bit rate is variable according to the importance of flow field, which leads to efficient data compression because high quantization bit rate is assigned to the important region only. In compression case of two-dimensional cylinder, the effects of the discrete wavelet transform and the variable quantization bit rate were demonstrated. Then flow field data around sphere was compressed to define data quality needed for visualization. Following these compressions, instantaneous large-scale flow field data with  $O(10^8)$  mesh points was successfully compressed with root mean square error and maximum quantization error suppressed to  $O(10^{-4})$  and  $O(10^{-3})$ , respectively. This compression technique was then extended for time-series flow field data. By compressing the difference between the current flow variables and previous decompressed ones, higher compression ratio was achieved compared to the direct compression of each time-step flow separately.

### 1. はじめに

計算機性能は現在も向上を続けており、将来的にはより大規模で複雑な対象を CFD で取り扱うことができるようになるかと予想される。近年では格子生成の容易さ、計算効率の良さから直交格子法が見直されており、そうした中で中橋は直交格子積み上げ法を提案した<sup>1)</sup>。直交格子積み上げ法はマルチブロック構造を用いた直交格子ベースの計算法であり、大規模計算に対応した解適合格子細分化、並列化、高次精度化を容易にしているほか、前処理・後処理の負荷の軽減も目指している。実際、直交格子積み上げ法を用いての大規模流体計算では参考文献<sup>2)</sup>でも報告されている通り、高い計算効率を達成している。しかしながら、大規模計算より得られる膨大なデータにより計算の後処理が難しくなっているのが現状であり、データの保存や転送、可視化といった場面でデータを取り扱い易くするためのデータ圧縮手法が必要である。本研究では直交格子積み上げ法により得られる大規模計算結果データの後処理の一環として、離散ウェーブレット変換を用いたデータ圧縮手法を提案する。画像符号化規格の符号化手順に則り、離散ウェーブレット変換とともに量子化、エントロピー符号化を用いてデータ圧縮を行う。圧縮対象のは非圧縮性流れの計算結果データであり、瞬間データ、時系列データの両方について圧縮を実行した。

### 2. 直交格子積み上げ法

直交格子積み上げ法により表現した計算領域を図 1 に示す。本手法では、流れ場はキューブと呼ばれる小領域に分割される。このキューブ内にはさらにセルと呼ばれる等間隔直交格子が含まれている。すべてのキューブは同数のセルを含んでいることから、数値計算の解像度はキューブサイズによって決定される。キューブサイズは局所的な流れ場の特性長さにより決定される。

それぞれのキューブはキューブサイズレベルと呼ばれる大きさの指標となる値を持つ。その値は最小キューブにおいて 0 であり、キューブサイズが 2 倍になる毎にレベルは 1 ずつ増加していく。図 1(a) に示した計算領域中には物体に最も近いレベル 0 から、最も遠いレベル 5 まで、全部で 6 種類の大きさのキューブが存在している。

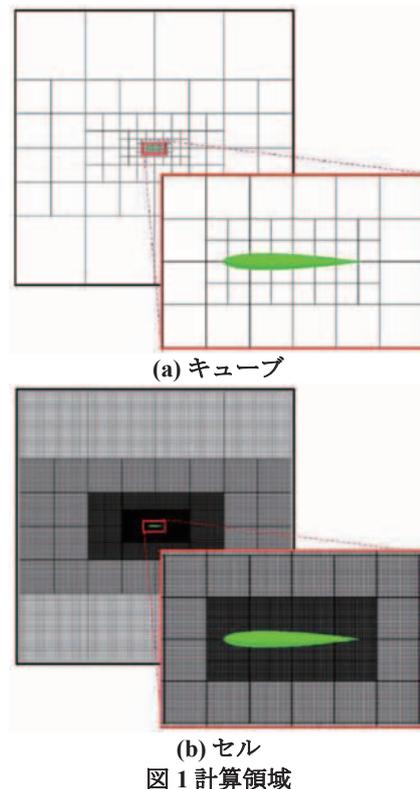


図 1 計算領域

結果データの出力フォーマットには Plot3D 形式を採用している。Plot3D 形式はマルチブロックの構造格子に用いられるフォーマットであり、直交格子積み上げ法において結果データは以下の情報を持つ。

- (1) キューブ数と各キューブ内のセル頂点数  
(4 byte 整数型)
- (2) 各セル頂点が持つ物理量  
(密度, 速度, 圧力) (4 byte 実数型)

なおデータ圧縮にあたっては密度一定とした非圧縮性流れの結果データのみ取り扱っているため、密度を除く速度と圧力の物理量情報を圧縮している。

### 3. 離散ウェーブレット変換

離散ウェーブレット変換 (Discrete Wavelet Transform: DWT)は圧縮や解析を目的として信号処理、画像処理に広く用いられている。離散ウェーブレット変換は信号を近似成分 (低周波成分) と詳細成分 (高周波成分) に分離することから、信号中の支配的な周波数成分とそうでない成分とを特定することができる。一般的に、様々な周波数成分のうち重要でない周波数成分を省くことで非可逆的にデータを圧縮できる。図 2 に、円柱周りの主流方向速度分布に離散ウェーブレット変換を実行した様子を示す。図 2(a)が変換前、(b)が変換後のデータである。ここで各領域に対応する AA, DA, AD, DD は  $x, y$  各軸方向の変換での近似/詳細成分の別を表している。AA 以外の領域は成分の絶対値が小さく、適切に省くことでデータ圧縮が可能な領域といえる。

データ圧縮は画像符号化規格 JPEG 2000 の符号化手順に則り、離散ウェーブレット変換とともに量子化、エントロピー符号化を用いて行う。以下にその手順を示す。

1. 離散ウェーブレット変換を物理量データに適用する。変換には Cohen-Daubechies-Feauveau (CDF) 9/7 ウェーブレット<sup>3)</sup>を用いている。
2. 量子化を行う。量子化は各キューブの各成分別に実行し、量子化ビットレートもそれぞれで別々に定める。
3. エントロピー符号化を行う。符号化にはレンジコーダー<sup>4)</sup>を用いている。エントロピー符号化も、各キューブの各成分別に実行する。

上記圧縮手順 2 において、量子化ビットレートを各キューブ、各成分で変化させている。これにより流れ場の重要性に応じた柔軟な圧縮が可能となっている。物体近傍のキューブでは量子化ビットレートを高くして細かく量子化し、量子化誤差を小さくする。一方で、物体から離れたところにあるキューブでは量子化ビットレートを低くして粗く量子化することでデータ量の削減を図る。また 1 キューブ内においても、流れの物理量の詳細成分は近似成分に比べその影響が小さいことから、量子化ビットレートを低くして粗く量子化することでデータ量の削減を図る。

なお複号は上記圧縮手順の 1. から 3. を逆の順序で実行することで行う。

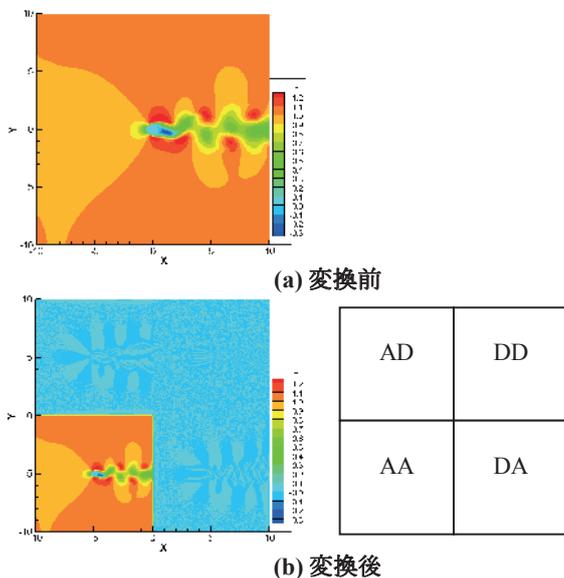


図 2 2次元円柱周りの主流方向速度分布

### 4. 結果および考察

#### 4.1 予備計算

離散ウェーブレット変換がデータ圧縮に及ぼす影響を調べるため、2次元等間隔直交格子による計算結果データを圧縮した。圧縮対象は2次元円柱周りの流れの計算結果の主流方向速度である。計算領域の1辺の格子点数は  $x, y$  軸ともに 250 である。離散ウェーブレット変換回数を 0 回もしくは 1 回とし、量子化ビットレートは 12 ビットを基準とし、近似成分と詳細成分の量子化ビットレートを別々に変化させた。表 1 中の量子化ビットレート (Q. bit rate) の A は近似成分 (図 2 (b) の AA に相当)、D は詳細成分 (図 2 (b) の DA, AD, DD に相当) に対応する。その後圧縮データを復元し、復元データより誤差の二乗平均平方根 (Root Mean Square Error: RMSE)、誤差の最大値、圧縮率 (Comp. ratio) を計算した。圧縮率は圧縮後のデータサイズを 1 とし、元データサイズとの比の形で表している。

圧縮の結果得られた数値を表 1 に、可視化図を図 3 に示す。表 1 の 1 列目のアルファベットと、図 3 の (a) から (d) のアルファベットは対応している。離散ウェーブレット変換回数が 0 回の際には、(a) と (b) の結果より、量子化ビットレートを小さくすると圧縮率は向上するものの、データ品質は低下する。一方、(c) の場合には (a) と同程度の品質を保ちつつも、圧縮率は (a) の場合よりもやや高くなっている。そして、近似成分を 12 ビット量子化、詳細成分を 11 ビット量子化としたときには、データ品質を (a) の程度に保ちつつ、圧縮率は (b) や (c) の場合よりも高くなっている。このように離散ウェーブレット変換と、近似/詳細成分に対応した量子化ビットレートをを用いることで、データ品質を保ちつつ圧縮率を高くすることができる。

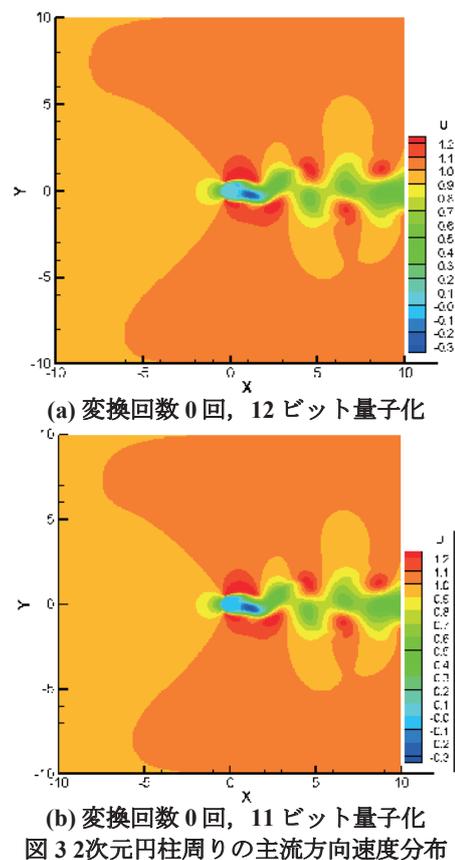
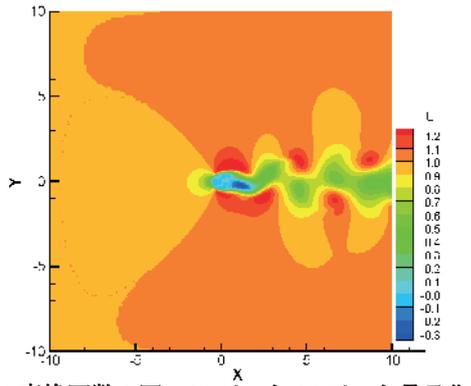
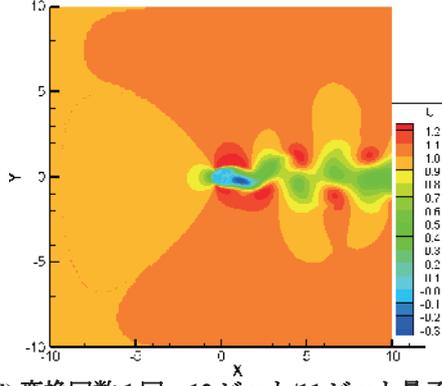


図 3 2次元円柱周りの主流方向速度分布



(c) 変換回数 1 回, 12 ビット/12 ビット量子化



(d) 変換回数 1 回, 12 ビット/11 ビット量子化

図 3 2次元円柱周りの主流方向速度分布 (続き)

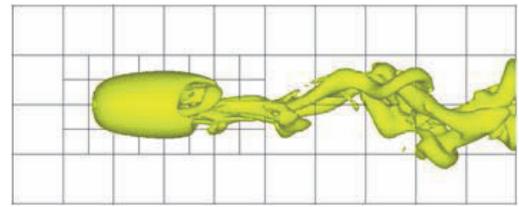
表 1 復元データ品質の比較

	# of DWT	Q. bit rate		RMSE	Max. Error	Comp. ratio
		A	D			
(a)	0	12	-	1.18E-04	2.09E-04	3.46:1
(b)	0	11	-	2.35E-04	4.17E-04	4.40:1
(c)	1	12	12	1.14E-04	3.08E-04	3.72:1
(d)	1	12	11	1.15E-04	3.18E-04	5.00:1

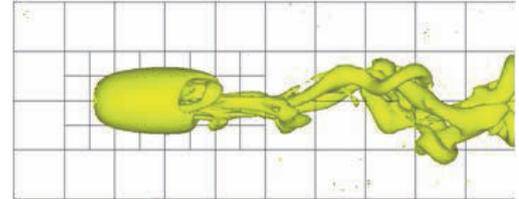
続いて直交格子積み上げ法による 3 次元計算結果を圧縮・復元し, 可視化図を利用して復元データ品質の評価を行った. 圧縮対象は球周りの流れの計算結果データであり, データの詳細を表 2 に示す. 離散ウェーブレット変換回数を 1 回, 量子化ビットレートを近似成分, 詳細成分ともに 8, 9, 10 ビットと変化させて圧縮を行った. その後圧縮データを復元し, 復元データより速度勾配テンソルの第二不変量  $Q^2$  を計算して等値面を可視化した. 図 4 に各ビットレートでの  $Q = 0$  の等値面を示す. また, 表 3 に各ビットレートでの誤差の二乗平均平方根, 誤差の最大値, 圧縮率を示す. 可視化の観点から, 10 ビット圧縮時に相当する程度のデータ品質が必要と判断し, RMSE が  $O(10^{-4})$ , 誤差の最大値が  $O(10^{-3})$ であることを復元データの品質の指標とした.

表 2 圧縮対象データの詳細

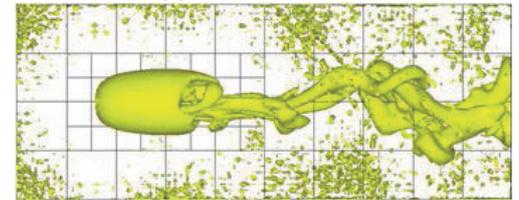
Object	Sphere
# of cells in cube	$16^3$
# of cubes	400
Original data size [byte]	39,328,020
Minimum grid spacing	3.91E-02
Reynolds number	1.00E+03



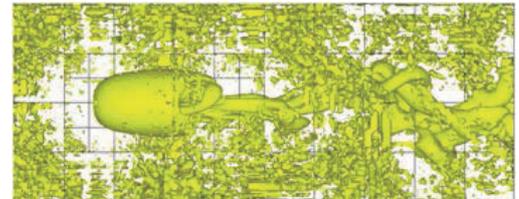
(a) 元データ



(b) 10 ビット量子化



(c) 9 ビット量子化



(d) 8 ビット量子化

図 4 球周りの  $Q = 0$  等値面

表 3 復元データ品質の比較

Q. bit rate	RMSE	Max. Error	Comp. ratio
10	6.62E-04	5.66E-03	9.04:1
9	1.36E-03	1.30E-02	11.6:1
8	4.14E-03	2.64E-02	15.2:1

#### 4.2 大規模流体計算結果データの圧縮

圧縮対象は円柱, Ahmed Body, F1 レーシングカーモデル周りの流れの計算結果データであり, データの詳細を表 4~表 6 に示す. いずれのケースも離散ウェーブレット変換回数は 1 回とし, 量子化ビットレートは表 7 に示す通りとした. 量子化ビットレートは最小キューブの近似成分で最大とし, また最大キューブの詳細成分で最小となるようにした.

図 5, 図 6 に, 復元データより得られた円柱周りの主流方向速度分布の全体図と物体近傍図を示す. 元データと復元データの分布はほぼ一致している. また図 7 に, Ahmed Body 先頭部分での  $Q = 0$  の等値面可視化結果を示す. 図 8 に, Ahmed Body 周りの主流方向速度プロファイルを示す. 図 7 と図 8 はともに復元データは元データの結果と良く一致しており, とくに図 8 の結果から復元データは定量的な検証にも使うことができるといえる. 図 9 に, 復元データより得られた F1 レーシングカーモデル周りの主流方向速度分布の物体近傍図を示す. この場合においても, 復元データの結果は元データとほぼ一致している.

表 8 にそれぞれの場合の誤差の二乗平均平方根, 誤差の最大値, 圧縮率を示す. いずれの場合でも RMSE は  $O(10^{-4})$  のオーダー, 誤差の最大値は  $O(10^{-3})$  のオーダーとなっており, 品質を保ちつつ圧縮することができた. 圧縮率はレイ

ノルズ数が高くなるにつれて低くなっている。この原因として、高レイノルズ数流れでは速度変動が大きくなり、流れ場における速度の偏りが小さくなったことが考えられる。

表 4 圧縮対象データの詳細 (円柱)

Object	Cylinder
# of cells in cube	16 <sup>3</sup>
# of cubes	19,536
Original data size [byte]	1,920,779,540
Minimum grid spacing	3.05E-03
Reynolds number	1.00E+04

表 5 圧縮対象データの詳細 (Ahmed Body)

Object	Ahmed body
# of cells in cube	32 <sup>3</sup>
# of cubes	2,838
Original data size [byte]	2,039,954,420
Minimum grid spacing	1.22E-03
Reynolds number	2.80E+06

表 6 圧縮対象データの詳細 (F1レーシングカー)

Object	F1 racing car model
# of cells in cube	32 <sup>3</sup>
# of cubes	5,930
Original data size [byte]	4,262,484,020
Minimum grid spacing	7.32E-04
Reynolds number	1.49E+07

表 7 量子化ビットレート

Cube size level	Approximation / Detail	Q. bit rate
0	Approximation	16
	Detail	12
1	Approximation	14
	Detail	10
2	Approximation	12
	Detail	8
Greater than or equal to 3	Approximation	10
	Detail	8

表 8 復元データ品質の比較

	RMSE	MaxError	Comp. ratio
Cylinder	9.82E-04	6.99E-03	11.4:1
Ahmed body	5.21E-04	5.99E-03	9.26:1
F1 racing car model	5.19E-04	3.70E-03	6.70:1

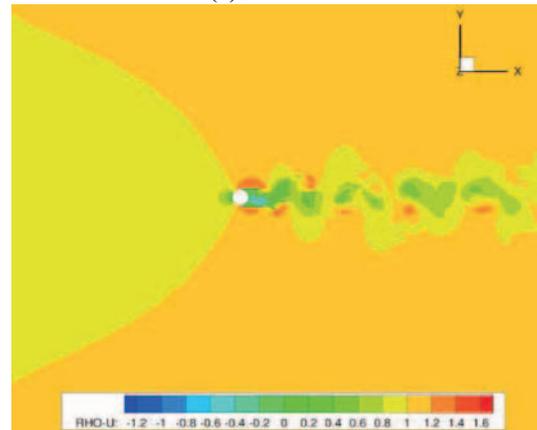
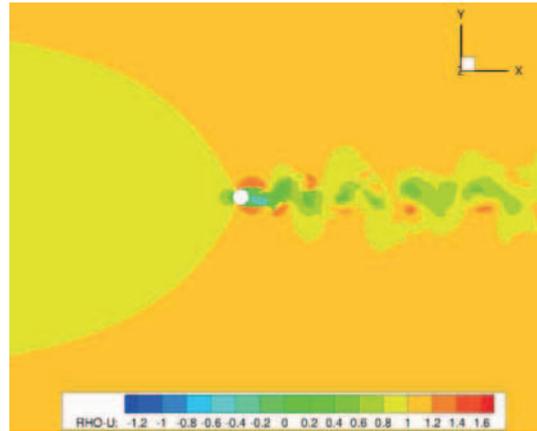


図 5 円柱周りの主流方向速度分布 (全体図)

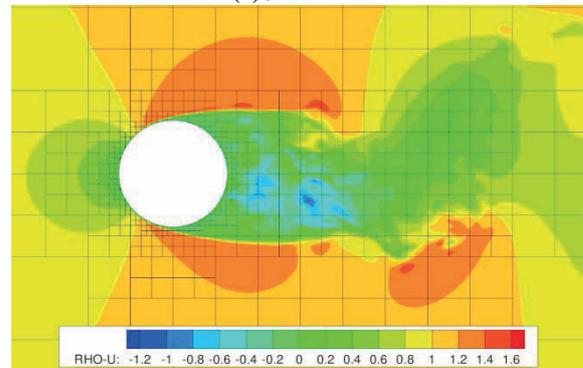
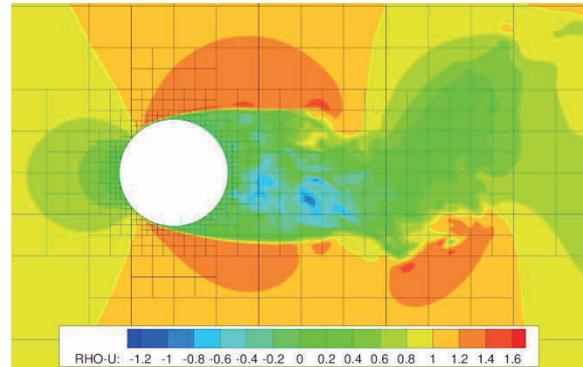


図 6 円柱周りの主流方向速度分布

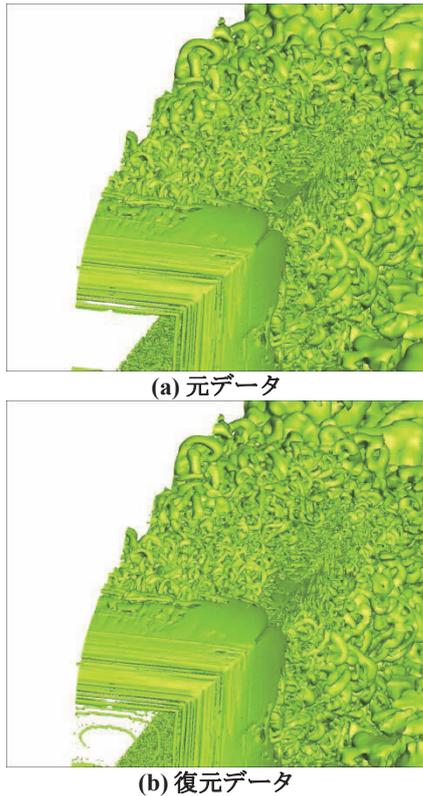


図 7 Ahmed Body先頭部の  $Q = 0$  等値面  
 ● : Original — : Decompressed

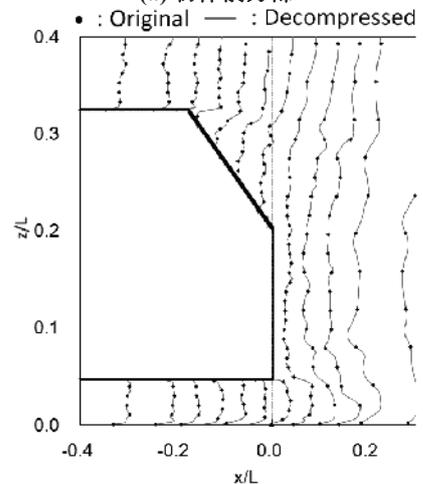
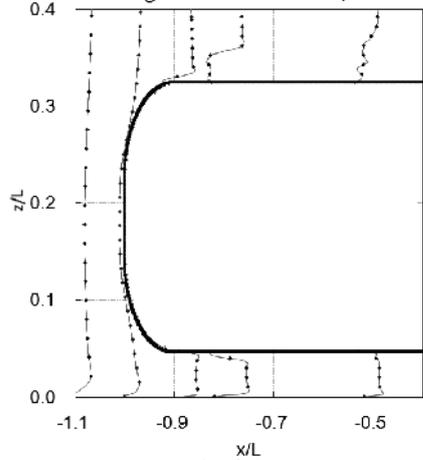


図 8 Ahmed Body主流方向速度プロファイル

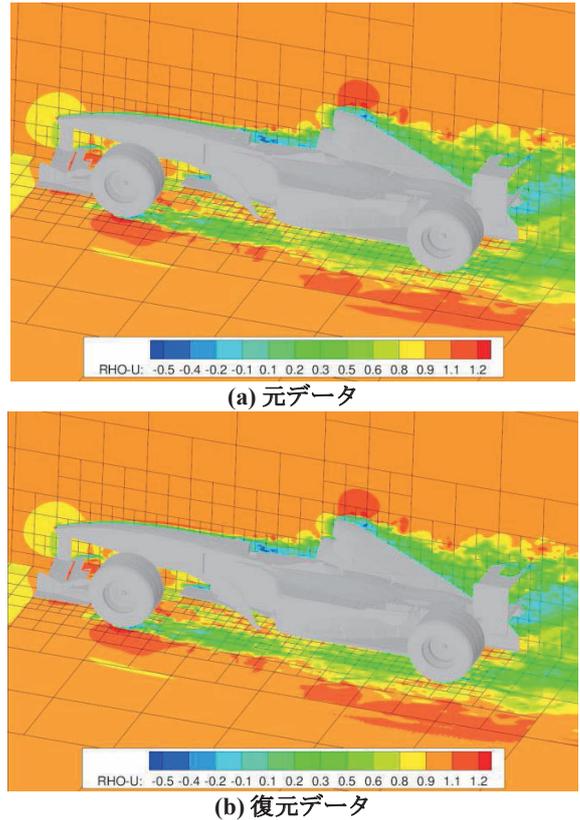


図 9 F1モデル周りの主流方向速度分布

### 4.3 時系列結果データの圧縮

本節では、多ステップにわたる連続データを圧縮した。連続データの圧縮に際し、前節と同じやり方で各ステップのデータを個別に圧縮する手法（個別圧縮）と、動画圧縮の手法を応用した差分を圧縮する手法（差分圧縮）の、2通りの方法を用いた。差分圧縮では、1ステップ目のみ個別圧縮を行い、2ステップ目以降は元データの代わりに、元データと前ステップの復元データとの差分をとり、その差分を圧縮する。差分は絶対値が小さいことから、小さな量子化ビットレートを割り当てることができ、データサイズを小さくすることができる。

圧縮対象は球周りの流れの結果データであり、4.1節で用いたものと同じである。時系列データとして 200 ステップ分のデータを用いた。データ出力間隔は無次元時間で 0.1 である。離散ウェーブレット変換回数を 1 回とし、量子化ビットレートは表 9 に示す通りとした。近似/詳細成分に対する量子化ビットレートの配分は前節と同様であるが、差分圧縮では個別圧縮よりも全体的にビットレートを小さくしている。

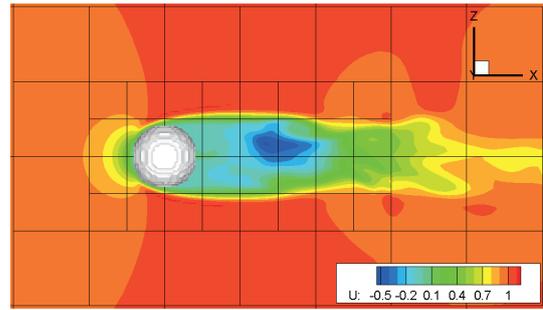
図 10 に元データと差分圧縮での主流方向速度分布の物体近傍図を示す。復元データは元データと良く一致している。表 10 にそれぞれの場合の誤差の二乗平均平方根、誤差の最大値、圧縮率を示す。各値は全 200 ステップでの値を平均したものである。両圧縮手法においてデータ品質は保たれており、とくに差分圧縮のほうがデータ品質が高い。そして圧縮率も差分圧縮のほうが高くなっている。このことから、差分圧縮はデータ品質を保ちつつ圧縮率を高くすることにより有効な手法といえる。

表 9 量子化ビットレート

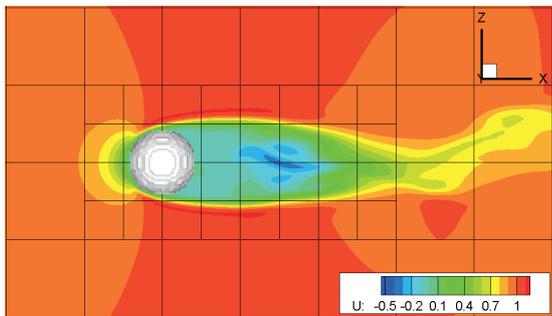
Cube size level	Approximation / Detail	Quantization bit rate	
		個別圧縮	差分圧縮
0	Approximation	14	11
	Detail	11	8
1	Approximation	14	11
	Detail	11	8
2	Approximation	8	6
	Detail	6	4
3	Approximation	6	4
	Detail	4	4

表 10 復元データ品質の比較

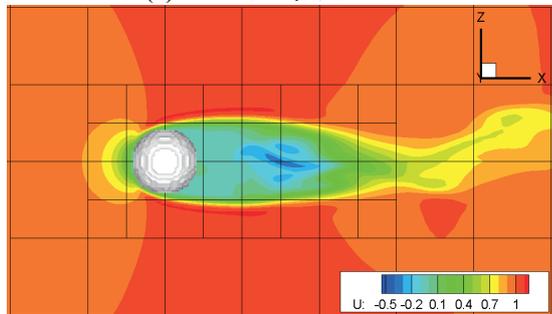
	RMSE	Max. Error	Comp. ratio
差分圧縮	2.64E-05	1.56E-04	20.9:1
個別圧縮	1.50E-04	1.27E-03	9.37:1



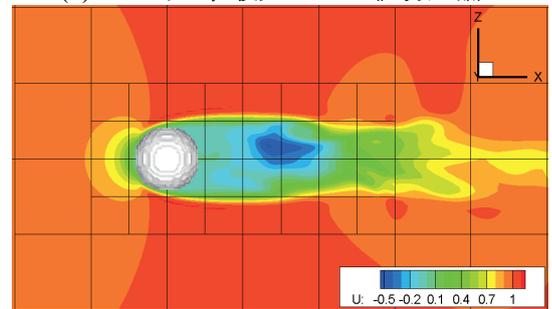
(d) 200 ステップ, 復元データ (差分圧縮)  
図 10 球周りの主流方向速度分布 (続き)



(a) 1 ステップ, 元データ



(b) 1 ステップ, 復元データ (差分圧縮)



(c) 200 ステップ, 元データ

図 10 球周りの主流方向速度分布

5. 結論

直交格子積み上げ法での大規模計算に伴う大規模出力データの後処理の一環として、離散ウェーブレット変換を用いたデータ圧縮法を提案した。データ圧縮には離散ウェーブレット変換のほか、量子化、エントロピー符号化を利用した。離散ウェーブレット変換により得られる近似/詳細成分に応じて、また直交格子積み上げ法のキューブサイズに応じて量子化ビットレートを変化させることで効率的なデータ圧縮を可能にしている。

円柱周りの 2 次元計算に対するデータ圧縮の結果では、離散ウェーブレット変換を実行し、かつ成分に応じてビットレートを変化させることで、データ品質を保ちつつ圧縮率を向上できることを確認した。

球周りの 3 次元計算に対するデータ圧縮の結果では、渦の可視化により復元データに必要なデータ品質の指標を定めた。その結果、可視化のためには RMSE で  $O(10^{-4})$ 、誤差の最大値で  $O(10^{-3})$  程度のデータ品質が必要であるとの結果を得た。

大規模瞬間データの圧縮では、データ品質を保ったままデータサイズを約 10 分の 1 程度に圧縮することができた。しかしレイノルズ数の増加とともに圧縮率が低下する傾向が見られた。

時系列データの圧縮では、元データを圧縮する代わりに前ステップとの差分データを圧縮することで、高品質かつ高圧縮率とすることができた。この結果より、時系列データに対する差分圧縮は有効な手法といえる。

参考文献

- 1) Nakahashi, K., "Building-Cube Method for Flow Problems with Broadband Characteristic Length," *Computational Fluid Dynamics 2002*, edited by S. Armfield, R. Morgan, K. Srinivas, Springer, pp. 77-81, 2003.
- 2) Takahashi, S., Ishida, T., Nakahashi, K., Kobayashi, H., Okabe, K., Shimomura, Y., Soga, T., and Musa, A., "Large Scaled Computation of Incompressible Flows on Cartesian Mesh Using a Vector-Parallel Supercomputer," *Parallel Computational Fluid Dynamics 2008*, Lyon, France, 2008.
- 3) Cohen, A., Daubechies, I., and Feauveau, J., "Biorthogonal Bases of Compactly Supported Wavelets," *Communication on Pure and Applied Mathematics*, 45(5), pp. 485-560, 1992.
- 4) Martin, G. N. N., "Range encoding: an algorithm for removing redundancy from a digitized message," *Video and Data Recording Conference*, 1979.
- 5) Jeong, J., and Hussain, F., "On the identification of a vortex," *J. Fluid Mech*, pp. 69-94, 1995