# 2009年度 卒業論文

液体金属熱対流シミュレーションとその可視化

# 神戸大学工学部情報知能工学科

## 古田敦哉

## 指導教員 陰山聡

2010年2月23日

#### 古田敦哉

### 要旨

外部磁場や回転の影響の下での磁気流体 (MHD 流体) の熱対流運動を解明する 目的で、液体金属(ガリウム)を用いた熱対流実験プロジェクトが北海道大学と海 洋研究開発機構で始まった。そして、この実験に対応する計算機シミュレーション がスーパーコンピュータ「地球シミュレータ」で行われており、その計算結果とし て膨大な数値データが得られつつある。本研究ではそのシミュレーションデータ を可視化し、解析する為の専用の可視化ツール "GFV" (Galium Field Visualizer) を開発した。GFV は速度場、磁場、圧力、エンストロフィ、ヘリシティ密度等の シミュレーションデータを読み込み、スカラー場の断面分布やベクトル場の矢印、 パーティクルトレーサーと呼ばれる流れ場の動きを見るための可視化オブジェク トを OpenGL を用いて表示する。GFV ではマウスを用いて視点を自由に変える ことができ、複数の可視化機能を同時に使用することができる。この GFV を使 用して、液体ガリウムに水平方向の磁場をかけて行った熱対流実験に対応するシ ミュレーションのデータを可視化し、解析を行った。その結果、実験では直接得 ることのできない3次元的な速度分布や、磁場の変化を明らかにすることができ た。今回 GFV によって初めて見つかった構造や効果として、対流ロール中の螺 旋状流線と、対流による磁束の集中効果がある。

## 目 次

1	序論	1
<b>2</b>	コンピュータグラフィックスと数値計算手法の基礎	2
	2.1 コンピューターグラフィックス	2
	2.2 補間法	2
	2.3 4次精度ルンゲ=クッタ法	3
3	可視化プログラムの概要とユーザーインターフェイス	6
	3.1 可視化プログラム GFV の概要	6
	3.2 ユーザーインターフェイス	8
	3.3 データ読み込み	9
	3.3.1 配列の確保	9
	3.3.2 エンディアン変換	9
4	スカラー場の可視化手法	10
<b>5</b>	ベクトル場の可視化手法	13
	5.1 平面の矢印表示	13
	5.2 3次元の矢印表示	13
	5.3 パーティクルトレーサー	14
6	GFV プログラムの応用	18
	6.1    実際の実験の概要	18
	6.2 シミュレーションの概要	18
	6.3 GFV による可視化結果	19
-	キレル	91

## 1 序論

熱対流運動は、古くから知られた興味深い流体現象である。水のような通常の 流体の熱対流は詳しく研究されているが、それと比較すると電気電導性の流体、 すなわち MHD (Magnetohydrodynamics) 流体の熱対流運動に関する研究の歴史 は浅い。特に外部から印加された強い磁場の下での MHD 対流や、高速回転の影 響を受けた MHD 対流については、未解明な点が多く残されている。

そこで現在、北海道大学と海洋研究開発機構において、回転や外部磁場影響下 での磁気流体(MHD流体)の熱対流運動を解明するために、液体ガリウムを使っ た熱対流実験のプロジェクトが進められている。液体金属は不透明であり、光学 的な測定をすることはできない。そこで、この実験では超音波流速分布測定技術 (Ultrasonic Velocity Profiler, UVP)を用いて液体ガリウム内部の対流速度分布 を直接計測している。しかし UVP は1次元的な速度分布しか得られず、実験で は3次元的な速度分布を直接見ることができない。また、磁場の構造も得ること ができない。

そこで実験と相補的な役割を果たす計算機シミュレーションが重要となる。計 算機シミュレーションでは速度分布や磁場の3次元的構造を調べることができる という利点がある。また、シミュレーションであらかじめ観測するべきポイント を調べることができれば、実験を効率的に進めることができる。

ただし、計算機シミュレーションによって得られる結果は数値データの羅列で あり、膨大な数値データから直接シミュレーションの結果を理解することは難し い。その為、コンピューターを用いてデータから CG(コンピューターグラフィッ クス)を作り、画像を見ることで計算結果を定性的、定量的に理解するプロセス が不可欠である。これをデータの可視化 (Visualization) と呼ぶ。可視化を行う際 には単に CG を作るだけでなく、データの抽出方法や可視化手段を変えて様々な 視点からデータを見て解析する必要がある。また、インタラクティブな可視化の 為には、高速に可視化処理を行うことが重要となる。そこで、MHD 熱対流運動 の解析という目的に特化した、専用の可視化ツールが必要とされている。本研究 の目的は MHD の熱流体運動の解析のためのインタラクティブな可視化プログラ ムを開発し、実際に MHD 対流のデータに適用し、その有効性を確認することで ある。

## 2 コンピュータグラフィックスと数値計算手法の基礎

#### 2.1 コンピューターグラフィックス

計算機シミュレーションで得られるデータは数値の羅列であり、人間が直感的 に理解するのは困難である。そこで数値データに潜む情報を2次元の画像として 抽出し、人間の優れた視覚能力を使ってそのデータを解釈するプロセス(可視化) が必要となる。可視化にはコンピューターグラフィックス(CG)の技術が用いら れる。

CG プログラムの基本 API として標準的に使われているものの1つに OpenGL (Open Graphics Library) がある。OpenGL はハードウェアに依存せず、また高精 度な3D画像を専用のハードウェア(GPU)を用いて高速に描写できるため、広く 普及している。OpenGL は点や線の描画、シェーディング処理、幾何変換や投影変 換など CG の作成に必要となる基本的な機能を持つ。本研究ではこの OpenGL に 加えて、GLU と GLUT という 2つの補助ライブラリを活用した。GLU (OpenGL Utility Library) は OpenGL の上位ライブラリであり、球体や円柱などの基本立 体や透視変換、ビューイング変換などの機能を持つ。これら 2 つのライブラリは CG の描画機能しか持たないので、ウィンドウ制御やイベント処理のために GLUT (OpenGL Utility Toolkit)を用いる。GLUT は OpenGL と同様に様々なプラット フォームに対応しており、これによりプラットフォームに依存しないシステム構築 を実現できる。これらのライブラリにより使用できるようになる関数は glTranslate (座標を移動する)、gluPerspective (透視投影を行う)、glutMouseFunc(マウス からの入力があった時に呼び出す関数を設定する)など、関数の名前からどのラ イブラリから呼び出されたかを判断できる [1]。

#### 2.2 補間法

本研究で開発する可視化ソフトウェアが対象とするシミュレーションデータは、 有限差分法 [2] で計算されたデータである。従って、シミュレーションデータは離 散格子点上に定義されている。そのため、格子点上に位置しない点における値を 求めるには、近傍の格子点から値を補間して求める必要がある。本研究では、3 次元線形補間を採用した。

例として、あるスカラー場 $\phi$ に対する補間を考える。点 $\mathbb{X}(x,y,z)$ が、Fig. 1のように、8点の格子点に囲まれた直方体の内部に位置しているとする。この直方体は点 $\mathbb{X}$ を通り、x、y、z軸に垂直な3つの平面によって8つの直方体に分割できる。こうして作られる8つの直方体の体積を、各格子点に対する重みとして利用する。ある格子点[i][j][k]における座標を $(x_i, y_j, z_k)$ 、スカラー量を $\phi_{ijk}$ とする。

 $\phi$ の値を補間して求める際、この格子点に対してかかる重み $w_{iik}$ を

 $w_{ijk} = f_i^x \times f_j^y \times f_k^z$ 

と定義する。ここで、

$$f_{i}^{x} = \frac{x_{i+1} - x_{i}}{x_{i+1} - x_{i}} = \frac{x_{i+1} - x}{\Delta x}$$

$$f_{i+1}^{x} = 1 - f_{i}^{x}$$

$$f_{j}^{y} = \frac{y_{j+1} - y}{y_{j+1} - y_{j}} = \frac{y_{j+1} - y}{\Delta y}$$

$$f_{j+1}^{y} = 1 - f_{j}^{y}$$

$$f_{k}^{z} = \frac{z_{k+1} - z}{z_{k+1} - z_{k}} = \frac{z_{k+1} - z}{\Delta z}$$

$$f_{k+1}^{z} = 1 - f_{k}^{z}$$

である。粒子から近い位置の格子点に対する重みが大きい。 以上の重みを使うと、Fig. 1 において点 X における φ の補間値は、

$$\phi(x, y, z) = w_{000} \times \phi_{000} + w_{001} \times \phi_{001}$$
$$+ w_{010} \times \phi_{010} + w_{011} \times \phi_{011}$$
$$+ w_{100} \times \phi_{100} + w_{101} \times \phi_{101}$$
$$+ w_{110} \times \phi_{110} + w_{111} \times \phi_{111}$$

と書ける。

## 2.3 4次精度ルンゲ=クッタ法

パーティクルトレーサーや流線等の可視化では積分計算が必要とされる。数値 積分法として、本可視化プログラムでは4次精度のルンゲ=クッタ法[3]を採用し た。ここでは4次精度のルンゲ=クッタ法について、簡単に説明する。例として、

$$\frac{dx}{dt} = v(x)$$

という常微分方程式を積分する問題を考える。



Fig. 1: Location of  $\phi$ .

はじめの x 座標を  $x_n$  とし、1 ステップ  $\Delta t$  だけ進んだ後の x 座標  $x_{n+1}$  を 4 次精 度ルンゲ=クッタ法を用いて求めるアルゴリズムは以下の通りである。

$$x_0 = x_n,$$
  

$$k_1 = \Delta t v(x_0),$$
  

$$k_2 = \Delta t v(x_0 + \frac{k_1}{2}),$$
  

$$k_3 = \Delta t v(x_0 + \frac{k_2}{2}),$$
  

$$k_4 = \Delta t v(x_0 + k_3),$$

$$x_{n+1} = x_0 + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4).$$

3次元の方程式を解く場合、例えば

$$\frac{dx}{dt} = v_x(x, y, z)$$
$$\frac{dy}{dt} = v_y(x, y, z)$$
$$\frac{dz}{dt} = v_z(x, y, z)$$

という問題に対しては、以下のような手順をとる。

$$x_0 = x_n, y_0 = y_n, z_0 = z_n$$

とすると、

$$k_1^x = \Delta t \, v_x(x_0, y_0, z_0)$$
  

$$k_1^y = \Delta t \, v_y(x_0, y_0, z_0)$$
  

$$k_1^z = \Delta t \, v_z(x_0, y_0, z_0)$$

$k_2^x$	=	$\Delta t v_x(x_0$	+	$\frac{k_1^x}{2}, y_0 +$	$\frac{k_1^y}{2}, z_0$	+	$\frac{k_1^z}{2})$
$k_2^y$	=	$\Delta t v_y(x_0$	+	$\frac{k_1^x}{2}, y_0 +$	$\frac{k_1^y}{2}, z_0$	+	$\frac{k_1^z}{2})$
$k_2^z$	=	$\Delta t v_z(x_0$	+	$\frac{k_1^x}{2}, y_0 +$	$\frac{k_1^y}{2}, z_0$	+	$\frac{k_1^z}{2})$

$$k_3^x = \Delta t \, v_x \left(x_0 + \frac{k_2^x}{2}, y_0 + \frac{k_2^y}{2}, z_0 + \frac{k_2^z}{2}\right)$$
  

$$k_3^y = \Delta t \, v_y \left(x_0 + \frac{k_2^x}{2}, y_0 + \frac{k_2^y}{2}, z_0 + \frac{k_2^z}{2}\right)$$
  

$$k_3^z = \Delta t \, v_z \left(x_0 + \frac{k_2^x}{2}, y_0 + \frac{k_2^y}{2}, z_0 + \frac{k_2^z}{2}\right)$$

$$k_4^x = \Delta t \, v_x(x_0 + k_3^x, y_0 + k_3^y, z_0 + k_3^z)$$
  

$$k_4^y = \Delta t \, v_y(x_0 + k_3^x, y_0 + k_3^y, z_0 + k_3^z)$$
  

$$k_4^z = \Delta t \, v_z(x_0 + k_3^x, y_0 + k_3^y, z_0 + k_3^z)$$

$$x_{n+1} = x_0 + \frac{1}{6}(k_1^x + 2k_2^x + 2k_3^x + k_4^x)$$
  

$$y_{n+1} = y_0 + \frac{1}{6}(k_1^y + 2k_2^y + 2k_3^y + k_4^y)$$
  

$$z_{n+1} = z_0 + \frac{1}{6}(k_1^z + 2k_2^z + 2k_3^z + k_4^z)$$

となる。

## 3 可視化プログラムの概要とユーザーインターフェイス

本研究では MHD 流体の流れ場と磁場を可視化するプログラムを開発した。プ ログラミングには C 言語を用い、3次元グラフィックスの作成には OpenGL を用 いた。このプログラムは計算機シミュレーションで得られた結果を元に、3次元 モデルを作成する。また、インタラクティブな操作を可能にし、モデルの解析の 利便性にも考慮した。このプログラムを GFV (Galium Field Visualizer) と名付 けた。

#### **3.1** 可視化プログラム GFV の概要

シミュレーションでは空間を格子点上で区切り、各格子点における速度場、磁 場、圧力、渦度(流れ場の回転)、ヘリシティ(速度場と渦度の内積)を計算する。 速度場と磁場はベクトルであり、各格子点においてx、y、z方向の3成分を持つ。 計算結果は各ベクトル成分、またはスカラー毎にバイナリファイルに出力される。 GFV ではこのファイルを読み込み、各格子点上の数値データを可視化する。

プログラムを実行するとデータがプログラムに読み込まる。その後、Fig. 2の ようにシミュレーション領域の境界を表す直方体の辺(以下、バウンディングボッ クスと呼ぶ)が表示される。赤い線が x 軸、緑色の線が y 軸、青色の線が z 軸に 平行な線を表す。GFV はこのバウンディングボックス内部での流体の動きを可視 化する。

読み込んだデータの可視化手法としてスカラー場の断面表示、ベクトル場の矢 印表示、速度場或いは磁場に体するパーティクルトレーサーを実装した。メニュー からそれぞれの機能を選択すると、可視化が行われる。

また、このソフトを利用する為には、ユーザーはあらかじめ以下の値を設定し ておく必要がある。

- NX,NY,NZ X,Y,Z 軸方向における格子点の数
- XMIN,XMAX X軸方向におけるシミュレーションデータ定義域の最大と 最小位置
- YMIN,YMAX Y軸方向におけるシミュレーションデータ定義域の最大と 最小位置
- ZMIN,ZMAX Z軸方向におけるシミュレーションデータ定義域の最大と 最小位置
- PARTICLEMAX パーティクルトレーサーで一度に置ける粒子の最大数



Fig. 2: Bounding box. 赤い線が x 軸、緑色の線が y 軸、青色の線が z 軸に平行 な線を表す。 ● MESH 4次精度ルンゲ=クッタ法により数値積分を行う際の積分刻み幅

#### 3.2 ユーザーインターフェイス

バウンディングボックスを含め、可視化されるオブジェクトはマウスによる操作 で、様々な角度から観察することができるように設計した。マウスの左ボタンを 押しながら左右にドラッグさせると視点の位置を変えることができる。また、マ ウスの右ボタンを押しながら上下にドラッグさせると物体の拡大、縮小ができる。

マウスの中央のボタンをクリックすると Fig. 3のようなメニューが表示される。 [Slice] は断面表示、[Arrow] は矢印表示、[ParticleTracer] はパーティクルトレー サーを実行する。これらの機能はそれぞれ同時に実行することが可能である。断 面表示で選ぶスカラー場は [SelectSliceData] から、矢印表示で選ぶベクトル場は [SelectArrowData] から選択できる。これらの機能の詳細は後述する。また、Reset View を選ぶと視点を初期位置に戻し、End を選ぶとプログラムが終了する。



Fig. 3: Example of the menu of GFV.

#### 3.3 データ読み込み

#### 3.3.1 配列の確保

可視化を行うにはまずデータをメモリ上に読み込む必要がある。本研究で可視 化するデータは地球シミュレーターで計算されたもので、そのシミュレーション プログラムは Fortran90 言語で書かれており、データの配列はそのままバイナリ データで出力される。しかし本可視化ソフトはCで開発するため、読み込む時に Cと Fortran では配列の並び方の扱いが異なることを考慮しなければならない。

データは各格子点において値を持っている。格子点は x 軸、y 軸、z 軸における 座標を持つので、その格子点における値は 3 次元配列に格納する。バイナリファ イルへ出力された配列は (NX,NY,NZ) の大きさの 3 次元配列である。しかし C で は 3 次元配列の順序が [0][0][0]、[0][0][1]、[0][0][2] ・・・と並ぶのに対して、Fortran では (1,1,1)、(2,1,1)、(3,1,1) ・・・という並び方になっている。従って、Fortran で 作られた並び方そのままに C の配列へ移してしまうと、格子の座標がバラバラに なってしまう。そのため C プログラムにおいては、データの配列は [NZ][NY][NX] のように設定する。この配列を使うことでデータの順序を変えることなく、C の 配列へデータを移すことができる。

また、C 言語の3次元配列では Fortran と異なり、NX\*NY\*NZ 個のデータがメ モリ内で連続しているとは限らない、ということにも注意する必要がある。



Fig. 4: Multidimensional arrays in Fortran and C.

#### 3.3.2 エンディアン変換

本研究では地球シミュレーターにより計算・出力されたシミュレーションデー タを可視化した。 ここで問題となるのは計算機同士のエンディアンの違いである。現在は主にビッ グエンディアンとリトルエンディアンの2つが使われている。例えば0x123456AB という16進数の値があったとき、ビッグエンディアンでは上位のバイトから順に [12 34 56 AB] という順番でメモリ上に配置する。一方、リトルエンディアンでは [AB 56 34 12] と下位のバイトから配置される。

本研究で可視化するシミュレーションデータは地球シミュレーターで計算されたものである。地球シミュレーターはビッグエンディアンが採用されているのに対して、本研究で可視化ソフトを開発した CPU はリトルエンディアンが採用されているため、エンディアンを変換する必要が生じた。

エンディアンを変換するにはメモリの配置を逆に並び替えればよい。これは共 用体 (union) を用いて実現した。エンディアン変換用の関数に引き渡された値は 以下のような共用体に代入する。

union {

float f;

unsigned char adr[sizeof(float)];

}data;

共用体の要素 adr は値の1バイト毎のアドレスにあたる。data.adr[n] という 形でこの共用体にアクセスすることで、値の1バイト毎にアクセスできる。float のバイト数は4であるので、これを利用して1バイト目と4バイト目、2バイト目 と3バイト目をそれぞれ交換すればエンディアン変換が実現される。可視化プロ グラムではデータを読み込んだ後に、すべての要素に対してこの方法でエンディ アン変換を施して、値を格納し直した。

## 4 スカラー場の可視化手法

本プログラム GFV では、スカラー場の可視化手法の一つとして、断面表示の 機能を実装した。この機能ではバウンディングボックスのx、y、z 軸それぞれに 直行する3つの平面上に分布したスカラー場を、値が最も大きい部分を赤く、最 も小さい部分を青く表示する。最大、最小値は指定した面上でのスカラー値分布 から決定する。スカラー場を表示させるには、まずメニューより [Slice] から表示 する断面を選ぶ。次に、表示するスカラー値はメニューの [SelectSliceData] より Velocity(速度ベクトルの絶対値)、Pressure(圧力)、Enstrophy(エンストロフィ)、 Helicity(ヘリシティ密度) の中から選択する。選択されたデータの配列へのポイン タが表示する為の関数へ引き渡される。xy 平面は Shift キー+左右キー、yz 平面 は Ctrl キー+左右キー、zx 平面は Shift キー+上下キーで1 格子間隔ずつ移動で きる。 断面上のスカラーデータの分布を可視化する為には、各格子点を頂点とした微小 な三角形を敷き詰めるように描いている。OpenGLにおいてglBegin(GL\_TRIANGLE\_STRIP) ~glEnd()で囲まれた範囲では、点を取る毎に指定された直近の3点が三角形を 描画される。例えばxy平面の断面表示を行う場合(Fig. 5)を考える。左下の点 を始点とし、そこから右、左上、右、左上・・・という順番で格子点を辿っていく。 こうして直近に取られた3点から成る三角形は、高さが断面のy軸方向の長さに 等しい長方形の形に敷き詰められる。1つの長方形を描き終えたら格子点を右へ 1つ進め、再び左下の点から上へ格子点を辿っていく。これをx軸方向の長さが 満たされるまで続け、断面を描写する。また、各頂点ではその格子点におけるス カラー量より、色情報が設定される。OpenGLでは頂点の間の色は周りの色情報 から補間されるので、隣り合う格子点の色が違っていても格子点の間には自動的 にグラデーションがかかり、滑らかに色が変化して見える。

また、断面表示は xy 平面、yz 平面、zx 平面それぞれを同時に可視化すること ができる (Fig. 6)。



Fig. 5: A slice plane by OpenGL's triangle strip. xy平面に対して描く場合。頂 点を取る度に、最後に取られた3点で三角形を作る。紫色の矢印の順に下から上 に向かって描ききると、格子点を1つ右へ進む。断面の端まで同じように点を辿 り、結果的に三角形が敷き詰められた形で長方形を描く。



Fig. 6: A snapshot of slice planes that visualize flow velocity amplitude. 赤い部 分が、バウンディングボックス内で流体が最も速く動いている場所であることが 分かる。一方、青い部分では速度が最も小さく、流体はほとんど動いていない。

## 5 ベクトル場の可視化手法

ベクトル場を可視化する手法は幾つか提案されているが、その中で GFV プロ グラムでは、矢印表示(2次元と3次元)及びパーティクルトレーサーを実装し た。本章ではそれらの機能と実装方法について述べる。

#### **5.1 平面の矢印表示**

この機能では格子点におけるベクトル成分より、2次元の矢印を表示する(Fig. 7)。 xy 平面、yz 平面、zx 平面に平行な面においてそれぞれx 成分とy 成分、y 成分と z 成分、z 成分とx 成分が成す方向を向いた矢印が表示される。また、これらの矢 印の長さはベクトルの強さに比例する。

例えば、メニューの [Arrow] より Slicexy を選ぶと、選択されている xy 平面に おけるベクトルを矢印で表示する。同様に Sliceyz を選ぶと yz 平面、Slicezx を選 ぶと zx 平面で表示する。この矢印は格子点を中心にして、その点に置けるベクト ル値の絶対値に比例した大きさで表示される。しかし全ての格子点におけるベク トルを表示すると、格子間隔を細かく取るほど描画時間がかかってしまう。また、 細かな多くの矢印が表示される為に視覚的にも見づらくなってしまう。そのため 全ての矢印を表示するのでなく、間引きを行い、一部の矢印のみを表示させるよ うにした。すなわち、全ての格子点においてではなく、ある一定の間隔での格子 点上にのみ矢印を表示した。また、値が小さく、ほとんど長さがない微小な矢印 は描画させないようにした。

矢印を表示させる関数には、規格化されたその格子点の各方向におけるベクト ル値と、その格子点の位置が引き渡される。引き渡された格子点の位置は矢印の 中心位置にあたる。例えば xy 平面の矢印の場合、x 成分、y 成分のみ考慮する。 引き渡された中心の位置からベクトルの正の方向へ 50%, 負の方向へ 50%の大き さだけそれぞれ点の座標を取り、ベクトルの始点、終点とする。矢印の先端もベ クトルの大きさに合わせて一定の位置に取る。

#### 5.2 **3**次元の矢印表示

この機能では格子点において x、y、z 軸方向の全てのベクトル成分より、3次 元の矢印を表示する (Fig.8 と Fig. 9)。平面の矢印表示と同様に、矢印の長さはベ クトルの強さに比例する。

メニューの [Arrow] から print3DArrow を選ぶと3次元ベクトルを矢印で表示する。平面の矢印表示とは異なり、立体的なポリゴンで表示される。平面の矢印は 頂点を結んだ線であったのに対して、この表示方法では陰がつくため奥行きがよ り直感的に理解できる。しかし立体を描写する分、平面の矢印表示に比べて表示 と位置の計算に時間がかかる。そのため平面の矢印表示よりも間引きの割合を多 くしている。

3次元の矢印を表示させる関数には、平面の矢印を表示させる関数と同様に、 規格化されたその格子点の各方向におけるベクトル値と、矢印の中心としてその 格子点の位置が引き渡される。関数の内部では、まずx成分、y成分、z成分それ ぞれについて中心の位置からベクトルの正の方向へ50%、負の方向へ50%の位置 に始点と終点を取る。次に軸の回転方向を得るため、逆正接を求める atan2 関数 を使いx 軸から y 軸への回転角度、xy 平面から z 軸への回転角度を求める。最後 に求めた角の分だけ回転した方向に、始点からはベクトルの70%の大きさで円柱 を、その先からは円錐を表示する。

#### 5.3 パーティクルトレーサー

パーティクルトレーサーは、流れの速度場や磁場など、ベクトル場に対して一 般的に適用できる可視化手法である。シミュレーション領域内に粒子の初期位置 を設定すると、時間積分の計算が開始され、計算し終わった部分までリアルタイ ムに粒子の位置が描写される。ここでは説明の為、速度場に対してパーティクル トレーサーを適用した場合を例として説明する。

メニューの [Particle Tracer] より set を選ぶと粒子のスタート位置を設置する モードに入る。マウスの左ボタンを押しながらドラッグするとxy 平面に黄色い枠 が表示され、これを動かすことができる。任意の地点でクリックすると枠が固定 される。この固定された位置が、粒子のスタート位置のz 成分となる。続いて緑 色の枠が zx 平面に表示されるため、同様に操作してy 成分を決める。最後に桃色 の枠が yz 平面に表示され、x 成分を決めることができる。このようにして粒子の スタート位置が決まると、その場所に粒子が表示される。粒子はあらかじめ決め た数まで置くことができ、[Particle Tracer] より start を選ぶと動きの追跡(計算) が始まる。

また、粒子のスタート位置はテキストファイルからの読み込みでも設定できる。 テキストファイルは particle.txt というファイル名で、粒子の初期位置のx、y、z をスペースを空けて記述する。複数個の設定をしたいときは、改行して次の粒子 の初期位置を書けば良い。[ParticleTracer] より、input を選ぶとテキストファイ ルからデータが呼び出され粒子がセットされる。そのまま続けて set を選べば粒 子を増やすことができる。また、パーティクルトレーサーが実行されている時に、 [ParticleTracer] より output を選択すれば粒子の初期位置が particle.txt に出力さ れる。



Fig. 7: A snapshot of velocity vector visualization by 2D arrows.



Fig. 8: A snapshot of velocity vector visualization by 3D arrows. Fig. 7 と Fig. 8 は同じ流体を、同じ視点から可視化している。



Fig. 9: Another snapshot of velocity vector visualization by 3D arrows.

時間積分には 2.3 節で述べた 4 次精度ルンゲ=クッタ法を適用する。速度場は x、y、z 方向の成分をそれぞれ持つベクトルであるので、各成分毎にルンゲ=クッ タ法による積分を行う。また、粒子位置における速度場は 2.2 節で述べた線形補間 により求める。時間積分を行う関数は、start が選ばれた時から 1 ステップ毎に、 GLUT 関数である glutIdleFunc 関数によって呼び出される。glutIdleFunc 関数は、 描画などの他の処理が行われない時に呼び出される関数である。時間積分の結果 の粒子の位置はグローバルな表示用の配列に格納されるので、周期的に呼ばれて いるパーティクルトレーサーを表示する関数もすぐに 1 ステップ分の動きを反映 させることができる。これにより時間経つにつれて粒子が動き、アニメーション として粒子の軌跡を視認することが可能となる。

しかし、全ての積分した点を表示すると描画処理に時間がかかってしまうという問題がある。この問題を解決する為に、矢印表示と同様に間引きを行う。すな わち、ある時刻に表示されている位置からの粒子の移動量が格子間隔を超えたと きのみ、結果を表示用の配列に格納する。



Fig. 10: A snapshot of velocity vector visualization by animated particle tracers.

## 6 GFV プログラムの応用

#### 6.1 実際の実験の概要

海洋研究開発機構と北海道大学では、液体ガリウムを使い、回転や磁場が液体 金属の熱対流運動にどのような影響を与えるかを実験的に調べるプロジェクトを 進めている。本節では液体ガリウムを使ったこの熱対流運動実験の概要をまとめ る。液体ガリウムは不透明であり、外部からその対流の様子を視認できないため、 超音波流速分布測定法 (UVP)を使って対流の様子を測定している。

実験ではFig. 11のように、容器を液体ガリウムで満たし、蓋をする。容器は高 さ40mm、幅200mm、奥行き200mmの直方体である。この容器の上面と下面に 異なる温度を与え、温度差により内部の液体金属に対流を発生させる。容器の上 面の温度は固定し、下面の温度は上面よりも高い温度にする。温度差の変化によ る対流の変化を測定する場合は下面の温度を調整する。更にFig. 12に示したヘ ルムホルツコイルを使い、磁場が熱対流運動に与える影響を見る。この装置によ り、容器に対して水平方向(x軸方向)に磁場がかけられる。

#### 6.2 シミュレーションの概要

上に述べた実験に対応して、同じ形状と設定の下での液体ガリウムの熱対流運動 の実験シミュレーションが九州大学の大学院生、井上によって行われた [5]。Fig. 13 にそのシミュレーションモデルを示す。シミュレーション領域は実験装置に合わ せて幅:奥行き:高さの比を 5:5:1 とした。原点は図のような下面の頂点の1つ で、幅方向に x 軸、奥行き方向に y 軸、高さ方向に z 軸を取った。また、境界面 においてはすべて滑り無しの条件であり、側面(xz 平面または yz 平面に平行な境 界面)の4 面は断熱の境界条件を設定した。また、上面と下面 (xy 平面に平行な 境界面)は等温の境界条件とした。

このシミュレーションで重要となる無次元パラメーターとして、レイリー数 Ra とエクマン数 Ek が挙げられる。それぞれ、

$$Ra = \frac{(g/T_0)\beta d^4}{(\kappa/\rho_0 C_p)(\nu/\rho_0)}$$
$$Ek = \frac{\nu}{\Omega H^2}$$

と定義される [4]。

シミュレーションは地球シミュレーターを使って行われた。格子点数はNX=251、NY=251、NZ=51 である。また Ra =  $9.84 \times 10^4$ 、Ek =  $\infty$ 、外部磁場の規格化 された強さは 0.1 である。



Fig. 11: Liquid galium contained in a convection vessel. この上に蓋をして、磁場をかける装置の中へ置く。実験時にはガリウムの上下の蓋にはそれぞれ一定の温度に保たれた水が常に流される。そうして上下の温度を調整することで、容器内に対流を生み出す。

#### 6.3 GFV による可視化結果

計算機シミュレーションのデータを本研究で開発した GFV プログラムで可視 化した。格子点数は NX=251、NY=251、NZ=51 で、外部磁場が x 軸方向にかけ られた場合である。また、バウンディングボックスの位置は原点が中心となるよ うに、実験容器の比に合わせて XMIN を-1.25、XMAX を 1.25、YMIN を-1.25、 YMAX を 1.25、ZMIN を-0.25、ZMAX を 0.25 と設定した。これらの値を設定し た後、プログラムを起動して 6.2 節のシミュレーション結果を読み込んだ。

速度場の断面表示を行うと、Fig. 14のような x 軸方向に軸を持ったロールが5 つ確認できた。これらは yz 平面を x 軸の正の方向へ移動させても、ほぼ一様な速 度分布をしていた。

続いて速度場のベクトルを見る為に、断面表示から矢印表示へ切り替えた。 Fig. 15のように、ロールの回転方向が隣り合うロール同士で異なっていることが 分かる。xy 平面の矢印(黄色の矢印)からも、x 軸方向には一様な動きをしてい ることが見られる。また、x 軸方向へ向かう成分がほとんど無い為、Fig. 16のよ うに3次元の矢印表示は yz 平面の矢印(緑色の矢印)と同じ形となった。なお、 3次元の矢印は yz 平面の格子点上では全て同じ形をしていたので、描画時間の高 速化を考えて x=0 の面上でのみ表示させている。

次に、磁場の断面表示を行ったところ、Fig. 17が得られた。また、磁場の矢印



Fig. 12: Helmholtz coils for horizontal magnetic field applied to the convection vessel. 左端の円形の装置がヘルムホルツコイルであり、電流を流すと磁場が発生する。2つのコイルの中央に容器を置き、磁場の影響を与える。



Fig. 13: Simulation model.

表示を行うとFig. 18のようになった。Fig. 18の矢印表示から、領域全体に x 軸方 向のほぼ一様な磁場が存在するように見える。しかし、Fig. 17 で示したように、 正確には磁場強度に空間分布が見られる。磁場にも5つのロールの形が見え、速 度場との関連が考えられる為、速度場と磁場の表示を比較したのが Fig. 19 であ る。Fig. 19の上下の図を比較すると、磁場が強い(赤く表示されている)場所は ロール同士が隣り合っていて、かつ速度場が最も小さい(青く表示されている) 場所に対応していることが分かる。また、Fig. 20 は磁場の断面表示と同時に、速 度場の3次元矢印表示を行ったものである。磁場が強い場所は、隣り合うロール の流れが合流する場所であることが分かる。これらの結果から、対流によって磁 場が寄せ集められていることが分かった。これは GFV の可視化によって初めて 見出された効果である。

続いて速度場に対してパーティクルトレーサーを実行した。5つのロールそれ ぞれに対して粒子の動きを見たのがFig. 21である。粒子はロールに沿って、x軸 の正の方向へ螺旋を描きながら進む。手前から1、3、5番目のロールでは右巻 きの螺旋を描いているのに対して、2、4番目のロールでは左巻きの螺旋を描き ながら進んでいる。Fig. 22のように螺旋の半径は進むにつれてだんだんと大きく なっていき、それに伴って x 軸方向へ進む距離が減っていく。また、各ロールに おいて軸の中心から離れた位置からパーティクルトレーサーを開始した方が円を 大きく描いていた。やがて x 軸方向へはほとんど進まなくなり、螺旋の半径があ る長さを超えると、Fig. 22の中心のロールのように、今度は x 軸の負の方向へ進 み始めた。

パーティクルトレーサーと、エンストロフィーの断面表示を同時に行ったのが Fig. 23 である。エンストロフィーは、速度場の回転(渦度ベクトル) $\omega = \nabla \times v$ の自乗 ( $\omega \cdot \omega$ )を取った値である [6]。すなわち、エンストロフィーが大きい場所と は、流体の渦度が強い場所である。螺旋が描かれている場所ではエンストロフィー の値が大きく、流体の回転が起こっていることが Fig. 23 からも分かる。

Fig. 24 ではパーティクルトレーサーと、ヘリシティ密度の断面表示を同時に 行った。ヘリシティ密度は速度場と渦度ベクトルの内積 *v*・ω で表され、渦の巻き 方を表す [7]。ヘリシティ密度が0 であれば回転はその場で留まっていて、正であ れば渦は右巻きの螺旋を、負であれば左巻きの螺旋を描く。Fig. 24 の螺旋の巻き 方と断面表示の色を比較すると、赤い部分はヘリシティ密度が正、緑色の部分は ヘリシティ密度が負であることが分かる。y 軸を挟んで反対側では赤色と緑色の 部分が反転していることから、y 軸に対して線対称な形で渦を描くと考えられる。 実際に反対側でもパーティクルトレーサーを行うと、Fig.25 のように螺旋の巻き 方が反対になっていることが確認できた。対流ロールの内部にこのような螺旋型 の流線構造があることは、GFV による可視化で初めて気がついたものである。

最後に圧力の断面表示を行ったのが Fig. 26 である。容器の下面の方が上面より



Fig. 14: A snapshot of slice planes for velocity field amplitude of liquid galium convection. x軸方向に軸を持つロールが5つあることが分かる。

も圧力が高いことが分かる。これは実験において、容器の下面の温度を上面より も高くしているので当然である。圧力に関しては、速度場や磁場との関連は見ら れなかった。



Fig. 15: Velocity vectors by 2D arrows. 黄色の矢印が xy 平面、緑色の矢印が yz 平面の速度場を表している。



Fig. 16: Velocity vectors visualized by 2D and 3D arrows. 速度場の yz 平面の矢 印表示と3次元矢印表示を行った。両者は同じ形をしており、x 軸方向の成分は ほとんど無いことが分かる。



Fig. 17: Distribution of magnetic field amplitude in the liquid galium convection.



Fig. 18: Magnetic field vectors visualized by 2D and 3D arrows.



Fig. 19: Combined visualization of 3D arrows and slice planes. 上図が速度場の 断面表示と速度場の3次元矢印表示、下図が磁場の断面表示と磁場の3次元表示 を同時に行ったものである。共に同じ視点から、同じ断面を表示している。速度 場の値が最も小さい場所で、且つロールが隣り合っている場所が磁場の強い場所 であることが分かる。





Fig. 20: Magnetic field concentration that is visualized by the red stripes in the horizontal slice planes. 磁場の断面表示と速度場の3次元矢印表示を同時に行っている。上図は箱の上部、下図は箱の下部にあたる。ロールの回転が合流する場所で磁場が強くなっていることが分かる。



Fig. 21: Helical flows structure visualized by particle tracers.



Fig. 22: Torus structure of the flow in convection rolls.



Fig. 23: The enstrophy distribution on slice planes with helical flow lines.



Fig. 24: Two Different kinds of helical flow lines; right-winding and left winding. The color shows the helicity density distribution; positive in red and negative in blue.



Fig. 25: A pair of opposite windings of helical flow, or a pair of opposite signs of helicity density, in each convection roll.



Fig. 26: Pressure on slice planes.

## 7 まとめ

本研究では、MHD 流体の熱対流シミュレーションを可視化するツール GFV を 開発し、地球シミュレーターで行われたシミュレーション結果に対して適用して、 その有効性を確認した。シミュレーション領域内には、x 軸方向に軸を持つ5本 のロールの存在が確認することができた。速度場の矢印表示を行うことで、隣り 合うロールはそれぞれ逆向きに回転していることが分かった。また、磁場の断面 表示を同時に行うことで、ロールの流れにより磁場が寄せ集められていることが わかった。磁場の構造は実験では直接測ることができない為、GFV によって見出 されたこの現象は実験家にも重要な示唆を与えるであろう。更にパーティクルト レーサーによって速度場を見ると、粒子は螺旋を描きながらロールに沿って、外 部磁場のかけられた x 軸に平行な方向へ移動することが分かった。Fig. 16 の速度 場の 3 次元矢印を見ても、x 軸方向に対して速度を持っているようには見えなかっ たが、それにも関わらず螺旋構造があることは、速度場の x 成分が微小であるた めである。速度場が x 軸方向に対しても成分を持っていたことは、矢印表示の機 能だけでは発見できなかったであろう。これは可視化に際して、様々な手段を持 つことの重要性を示している。

螺旋には右巻きと左巻きがあり、隣り合うロール同士、更にロールの両側で巻き 方が逆になっていることが分かった。このことは、ヘリシティ密度の断面表示でも 確認できた。パーティクルトレーサーではこの螺旋の半径がだんだん大きくなり、 半径が一定の長さを超えると x 軸の進行方向が逆向きに変わることが分かった。

GFVによって、今まで知られていなかった幾つかの事象を発見することができ た。しかし今回実装した機能だけで分かることは、ほんの一部でしかないだろう。 可視化においてどのデータをどのように可視化するのが最適であるかという問題 は、その時々によって答えは異なり、使用者が調整しなければならない。例えば 断面表示の最大、最小値は全てのスカラー値において、データの最大、最小値を そのまま適用すれば良いというわけでは無かった。最大値が飛び抜けて大きな値 を持ち、その他の点で変化の幅が小さかった場合、断面表示はほとんどの点で同 じ色になってしまい変化が詳しく見れなかった。加えて全体像だけでなく、場所 に応じて細かい範囲の変化を切り取って見たい時には、現在のプログラムは適し ていない。多くの研究の要望に応えられる機能の考案、実装は大きな課題となる。

また、GFVでは実装されている機能は同時に実行できるが、多くの機能を一度 に実行すると動作が目に見えて遅くなってしまった。プログラムの処理の並列化 などによる高速化も必要であろう。

## 参考文献

- [1] 林武史・加藤清敬, "OpenGLによる3次元CGプログラミング," コロナ社 (2003)
- [2] 桑原邦郎・川村哲也, "流体計算と差分法," 朝倉書店 (2005)
- [3] W.H.Press, B.P.Flannery, S.A.Teukolsky and W.T.Vetterling, "Numerical Recipes in C : The Art of Scientific Computing," Second Edition, *Cambridge Press*, pp.710-714(1992)
- [4] D.O.Gough, D.R.Moore, E.A.Spiegel, N.O.Weiss, "CONVECTIVE INSTABIL-ITY IN A COMPRESSIBLE ATOMOSPHERE. II," THE ASTROPHYSI-CAL JOURNAL, 206, pp.536-542(1976)
- [5] 井上真志,"液体金属熱対流実験の計算機シミュレーション,"九州大学総合理工 学府修士論文 (2010)
- [6] 巽友正,"流体力学," 培風館, pp.407-408(1982)
- [7] 日本流体力学会編,"流体力学ハンドブック,"丸善,p.162(1998)

### 謝辞

本研究に際して、初めての研究で分からないことも多い中、いつも熱心かつ丁 寧な指導を賜りました指導教官である陰山聡教授には深く感謝いたします。九州 大学理工学府の井上真志氏には MHD 流体のシミュレーションデータを快く提供 して頂きました。また、海洋研究開発機構の柳澤孝寿博士、山岸保子博士には液 体ガリウム実験の詳しい解説をして頂きましたことを感謝いたします。

最後に1年間同じ研究室で過ごした、陰山研究室の学部生である村田歌織さん、 吉田真人君に感謝します。