非理想磁気流体力学に関する実習

高棹 真介（大阪大学）

ここで学ぶこと

- オーム散逸が含まれた磁気流体現象を扱う基本手順を学ぶ
  - 基礎方程式と規格化された方程式の確認
  - オーム散逸が関わる具体的な天体現象の話（磁気リコネクション）
  - 実習で使う pgen file (reconnection.cpp) の説明
  - 計算の実行と可視化の目標の説明

非理想磁気流体方程式：今回は磁気拡散

以下、cgs 単位系。オーム散逸によるの電気伝導度を σ とする。

$$\boldsymbol{E} = -\frac{\boldsymbol{v}}{c} \times \boldsymbol{B} + \frac{1}{\sigma}\boldsymbol{J} \tag{6.1}$$

だから、誘導方程式は次のようになる：

$$\frac{\partial \boldsymbol{B}}{\partial t} = -c\nabla \times \boldsymbol{E} = \nabla \times (\boldsymbol{v} \times \boldsymbol{B}) + \nabla \times \left(\frac{c}{\sigma}\boldsymbol{J}\right) \tag{6.2}$$

ここで $\boldsymbol{J} = \frac{c}{4\pi}\nabla \times \boldsymbol{B}$ を使うと

$$\frac{\partial \boldsymbol{B}}{\partial t} = \nabla \times (\boldsymbol{v} \times \boldsymbol{B}) + \nabla \times \left(\frac{c^2}{4\pi\sigma}\nabla \times \boldsymbol{B}\right) \tag{6.3}$$

この式の意味を見やすくするため、電気伝導度の空間変化が小さいとすると

$$\frac{\partial \boldsymbol{B}}{\partial t} \approx \nabla \times (\boldsymbol{v} \times \boldsymbol{B}) + \frac{c^2}{4\pi\sigma}\Delta \boldsymbol{B} \tag{6.4}$$

となり、有限の電気伝導度がある時、磁場の拡散項が現れる。これはもともと電流の項から現れており、電流の散逸に対応する項である。つまり、電流の散逸を通じて磁場が拡散する（電流が磁場を作ることを考えれば自然）。以下では磁気拡散係数 η を

$$\eta \equiv \frac{c^2}{4\pi\sigma} \tag{6.5}$$

として定義する。

Athena++ で規格化された誘導方程式

cgs ガウス単位系

$$\frac{\partial \boldsymbol{B}}{\partial t} = -c\nabla \times \boldsymbol{E}$$

$$\boldsymbol{E} = -\frac{\boldsymbol{v}}{c} \times \boldsymbol{B} + \frac{4\pi\eta}{c^2}\boldsymbol{J}$$

$$\boldsymbol{J} = \frac{c}{4\pi}\nabla \times \boldsymbol{B}$$

規格化されたもの

$$\frac{\partial \boldsymbol{B}}{\partial t} = -\nabla \times \boldsymbol{E}$$

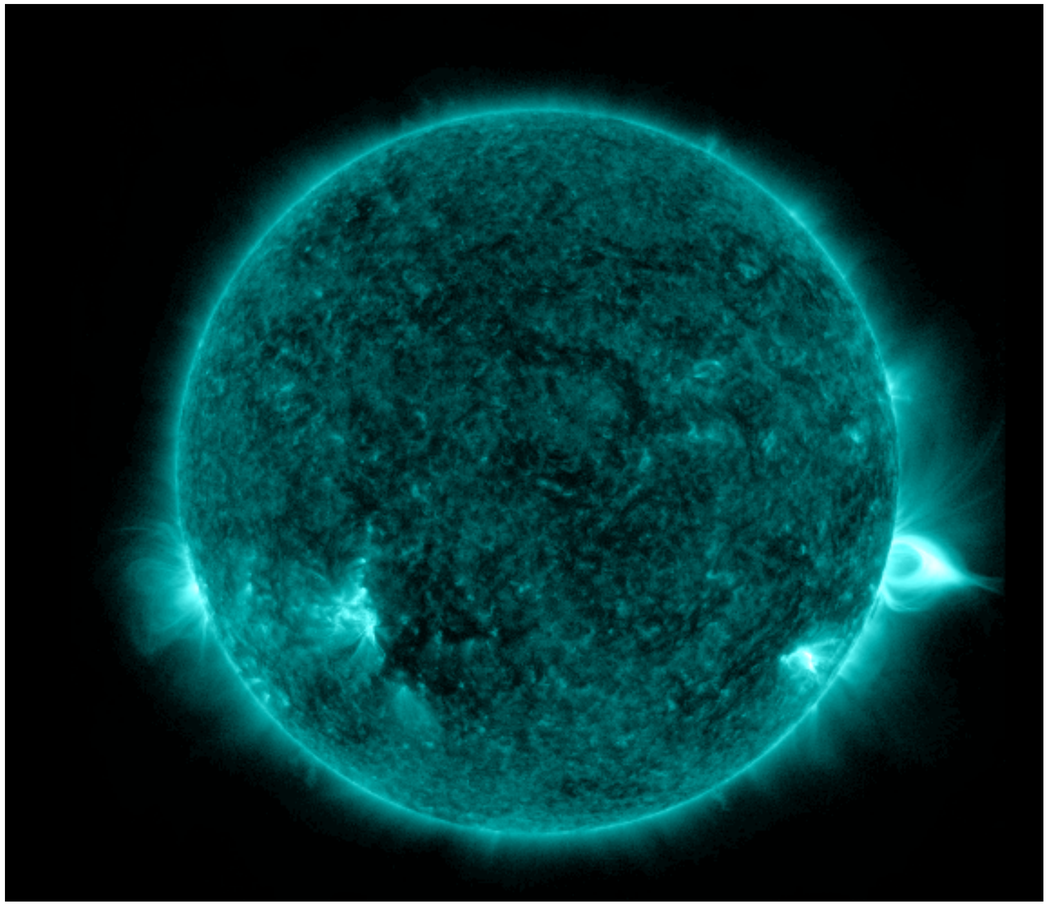
$$\boldsymbol{E} = -\boldsymbol{v} \times \boldsymbol{B} + \eta\boldsymbol{J}$$

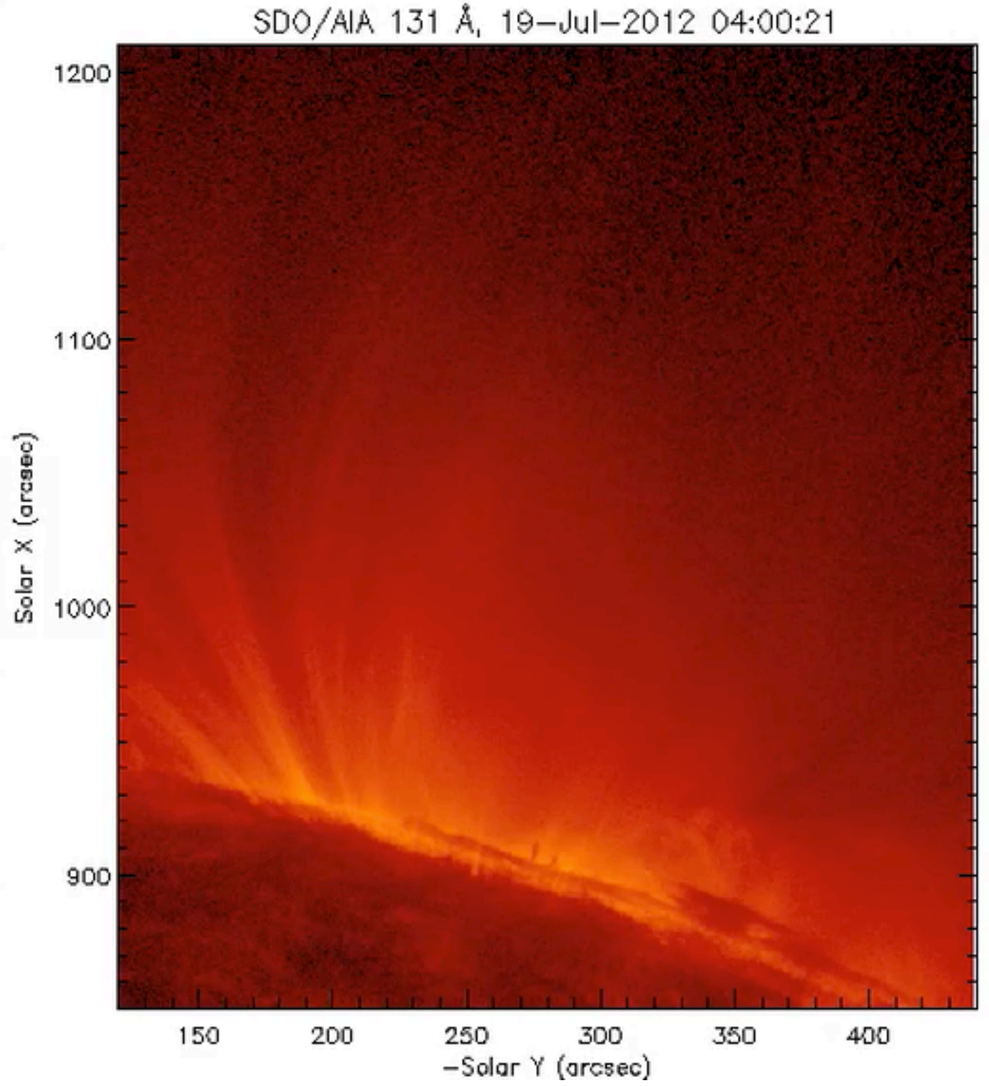
$$\boldsymbol{J} = \nabla \times \boldsymbol{B}$$

今回はこれを与える方法をみていきます

磁気拡散が関わる天体現象：太陽フレア

実際の太陽の画像（極端紫外線）

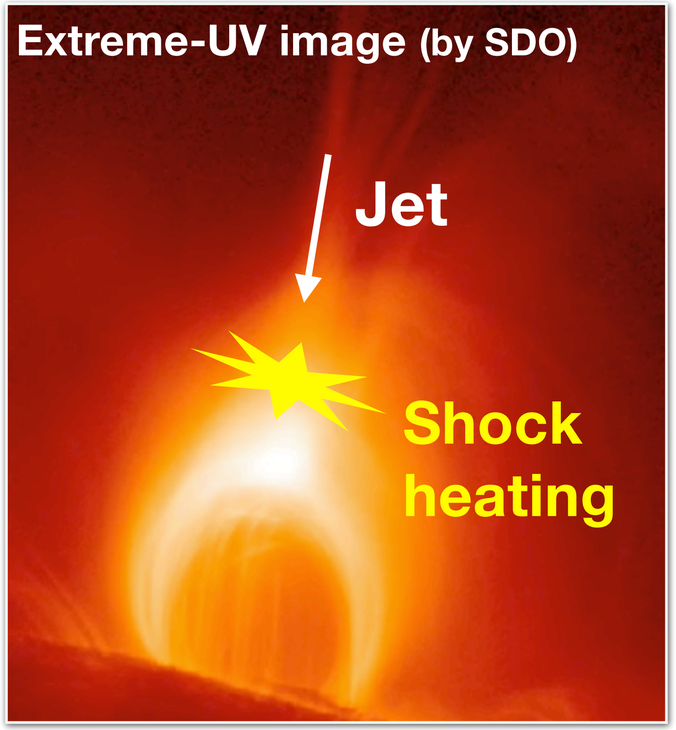




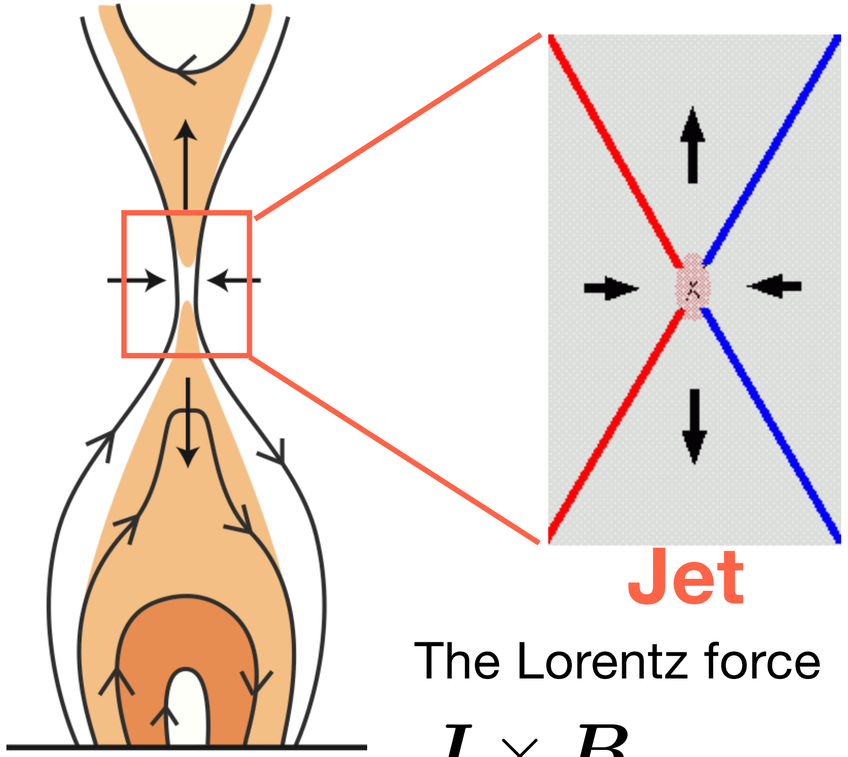
サイズ ~1-10 万 km の爆発が数時間のタイムスケールで起きる

Courtesy: Wei Liu

磁気リコネクション



Magnetic energy
—> kinetic & thermal energies



磁気拡散で
磁場の
反平行成分が散逸

The Lorentz force

$$\boldsymbol{J} \times \boldsymbol{B}$$

$$= -\nabla\left(\frac{B^2}{8\pi}\right) + \frac{1}{4\pi}(\boldsymbol{B}\cdot\nabla)\boldsymbol{B}$$

Magnetic pressure gradient

Magnetic tension

磁気リコネクションについてもう少し

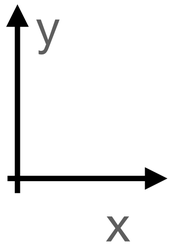
いきなり磁力線のつなぎ変えを
考えるのではなく、一旦磁場を
ベクトル場にしてから考えると
わかりやすい

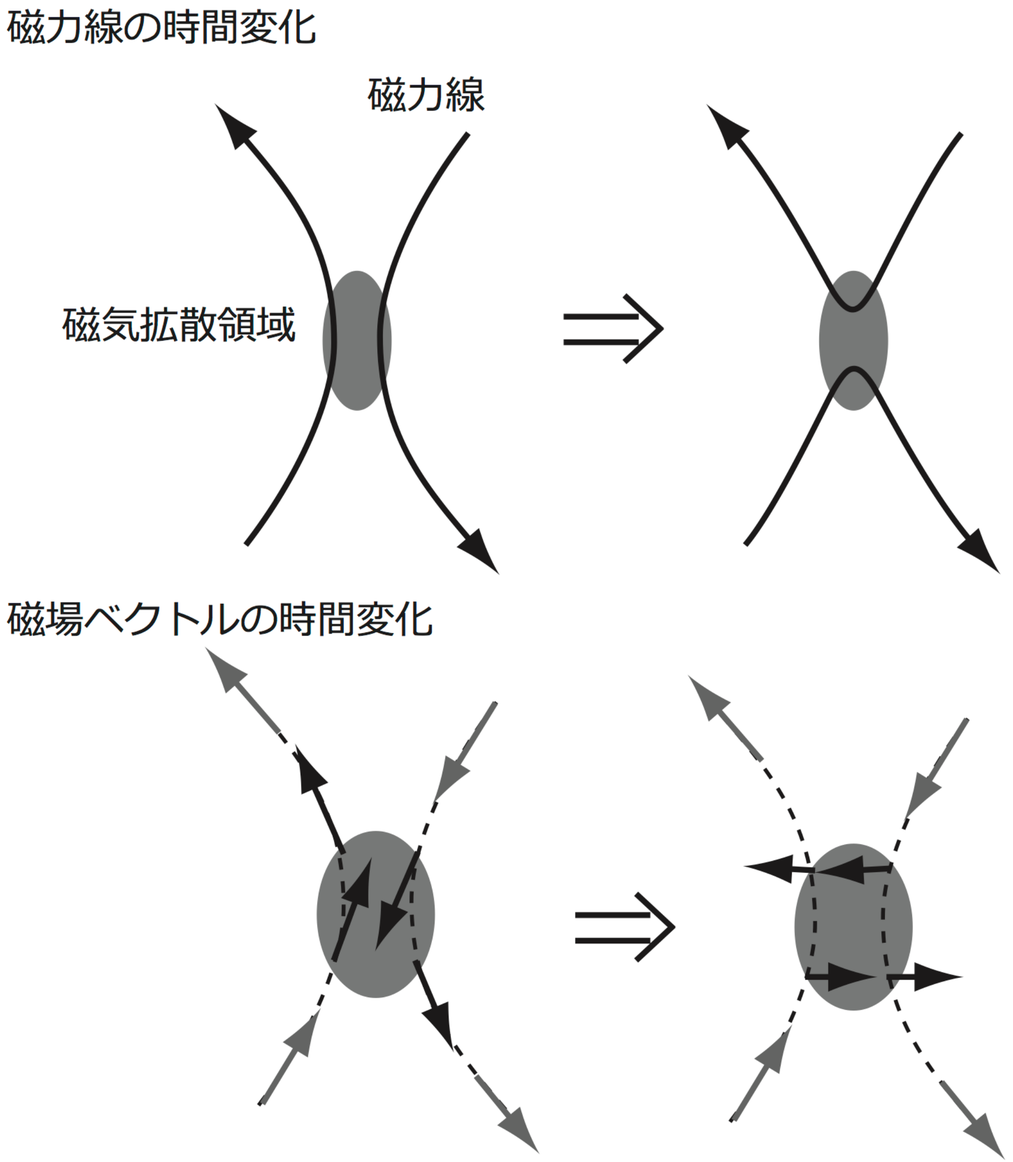
$$\frac{\partial \boldsymbol{B}}{\partial t} = \nabla \times (\boldsymbol{v} \times \boldsymbol{B} - \eta \boldsymbol{J})$$

リコネクション領域では $\boldsymbol{v} \approx 0$

y 成分 $\quad \dfrac{\partial B_y}{\partial t} \approx \eta \dfrac{\partial^2 B_y}{\partial x^2}$

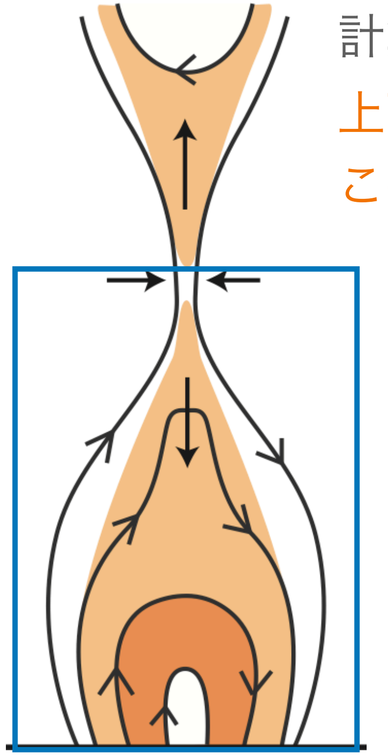
散逸するのは By だけ





reconnection.cpp について

太陽フレアなどの爆発現象を駆動する**磁気リコネクション**という現象を扱うモデル。



計算量削減のため、
上下対称性を仮定し
この下側だけを計算する

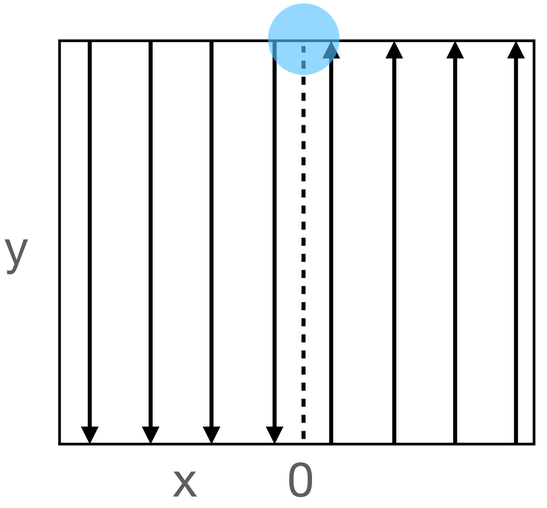
初期 & 境界条件

磁場

$$B_x(x, y) = 0,$$
$$B_y(x, y) = B_0 \tanh(x/w),$$
$$B_z(x, y) = B_0/\cosh(x/w),$$

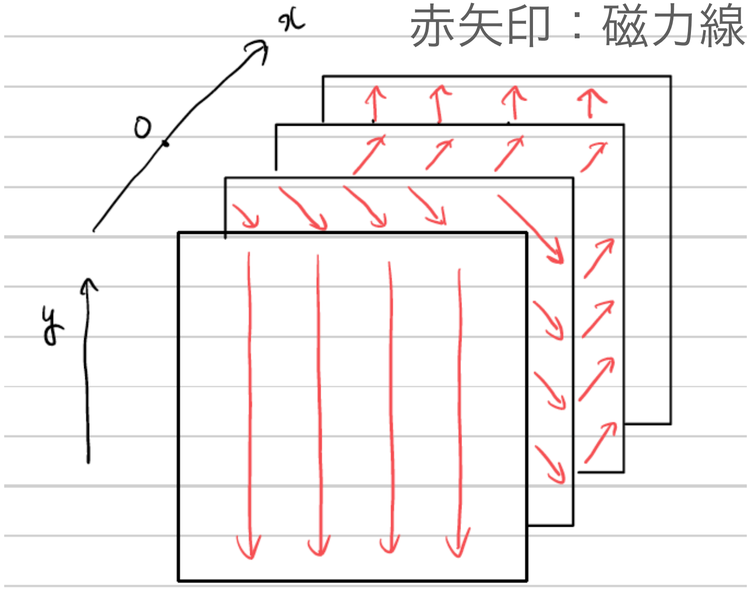
密度と圧力は空間的に一様

ここにだけ局在化した磁気拡散
（リコネクション場所を固定）



x方向: reflecting,

y方向: user定義の reflecting



赤矢印：磁力線

磁気圧がどこでも一定になっている

reconnection.cpp の中身

```
43  void LocalizedDiffusivity(FieldDiffusion *pfdif, MeshBlock *pmb, const AthenaArray<Real> &w,
44      const AthenaArray<Real> &bmag, const int is, const int ie, const int js,
45                            const int je, const int ks, const int ke) ;
46
47  namespace {
48  Real d0, beta0, eta0, w_eta;
49  } // namespace
50
51  void Mesh::InitUserMeshData(ParameterInput *pin) {
52    d0     = pin->GetReal("problem", "d0");
53    beta0  = pin->GetReal("problem", "beta0");
54    eta0   = pin->GetReal("problem", "eta0");
55    w_eta  = pin->GetReal("problem", "w_eta");
56
57    EnrollUserBoundaryFunction(BoundaryFace::inner_x2, MyBoundaryInnerX2);
58    EnrollUserBoundaryFunction(BoundaryFace::outer_x2, MyBoundaryOuterX2);
59
60    EnrollFieldDiffusivity(LocalizedDiffusivity); // User-defined diffusivity
61    return;
62  }
```

これがユーザ定義の
オーム散逸を計算する関数
（関数の宣言）

InitUserMeshData 内で
EnrollFieldDiffusivity を使い
Athena++ にユーザ定義関数
を渡す（必須）

注意：オーム散逸を使うように Athena++ に教えるには、input file の <problem> で
eta_ohm に non-zero の値を渡さなければいけない（実際に eta_ohm の値を使うかは別として）

reconnection.cpp の中身

```
135  void LocalizedDiffusivity(FieldDiffusion *pfdif, MeshBlock *pmb,
136       const AthenaArray<Real> &w,
137       const AthenaArray<Real> &bmag, const int is, const int ie, const int js,
138                               const int je, const int ks, const int ke)
139  {
140    Coordinates *pcoord = pmb->pcoord;
141    Real h = pmb->pmy_mesh->mesh_size.x2max;
142    Real w_eta2_i = 1.0/SQR(w_eta);
143
144    for(int k=ks; k<=ke; k++) {
145      for(int j=js; j<=je; j++) {
146        Real y = pcoord->x2v(j);
147  #pragma omp simd
148        for(int i=is; i<=ie; i++){
149          Real &etaB = pfdif->etaB(FieldDiffusion::DiffProcess::ohmic,k,j,i);
150          Real x = pcoord->x1v(i);
151          Real r2 = x*x + SQR(y-h);
152          Real tmp = r2*w_eta2_i;
153          etaB = eta0 * exp(-tmp);
154        }
155      }
156    }
157
158    return;
159  }
```

ユーザ定義の
オーム散逸を計算する関数の
具体的な中身の記述

オーム散逸の磁気拡散係数は
このように呼び出す

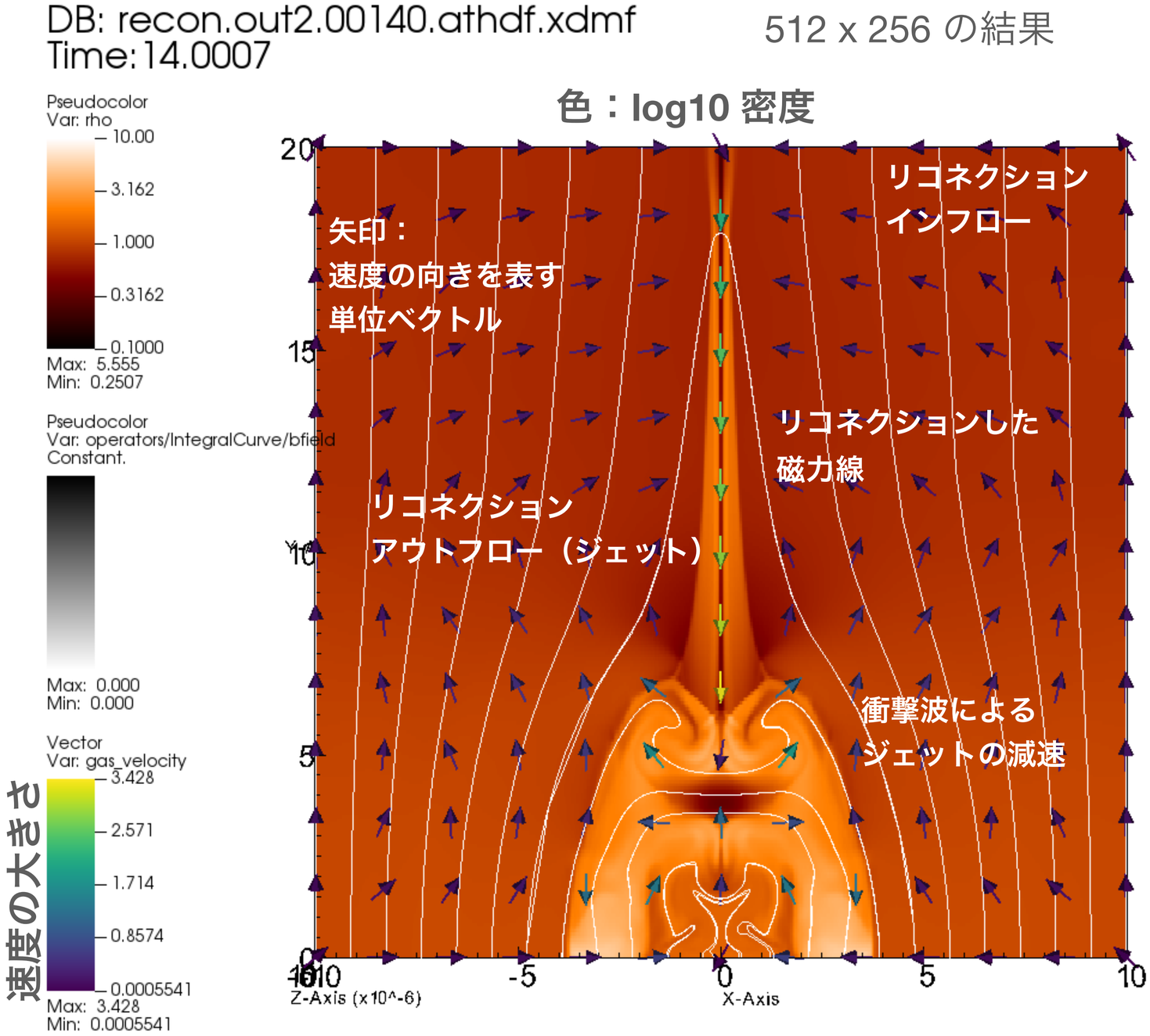
（詳細は src/field/field_diffusion/field_diffusion.hpp）

$$\eta = \eta_0 \exp\left(-[(x^2 + (y-h)^2)]/w_\eta^2\right)$$

という、広がりが w_eta の
ガウシアンの形にしている

目標

- 第一目標
  - xc50 で計算を流し、データを手元に持ってくる
  - こんな感じの図を作って、磁力線やガスの流れ場の構造と背景の物理量（密度など）の関係をわかりやすく表示してみる
- 発展編
  - 今回のモデルでは dt が磁気拡散によって制限されている（ログを見ると、dtが一定になっているのがわかる）。STSを使うとどの程度 dt が大きくなるか調べてみよう。
  - history データを使い、磁気エネルギーの解放の結果として運動・熱エネルギーが増加していく時間変化の様子をグラフにしてみよう
  - 今回のモデルでは境界条件を reflecting ではなくユーザ定義のものを利用している。こうしなければいけない理由を考えてみよう。



athinput.reconnection

```
<comment>
problem   = reconnection
reference = Yokoyama and Shibata 2001, Takasao et al. 2015
configure = -b --prob=reconnection

<job>
problem_id   = recon   # problem ID: basename of output filenames

<output1>
file_type  = hst       # History data dump
dt         = 0.01      # time increment between outputs

<output2>
file_type  = hdf5     # hdf5
variable   = prim     # variables to be output
dt         = 0.5      # time increment between outputs

<time>
cfl_number = 0.4       # The Courant, Friedrichs, & Lewy (CFL) Number
nlim       = -1        # cycle limit
tlim       = 15.       # time limit
integrator = vl2       # time integration algorithm
xorder     = 2         # order of spatial reconstruction
ncycle_out = 1         # interval for stdout summary info
```

```
nx2         = 512       # Number of zones in X2-direction
x2min       = 0.      # minimum value of X2
x2max       = 20.       # maximum value of X2
ix2_bc      = user  # inner-X2 boundary flag
ox2_bc      = user  # outer-X2 boundary flag

nx3         = 1         # Number of zones in X3-direction
x3min       = -0.5      # minimum value of X3
x3max       = 0.5       # maximum value of X3
ix3_bc      = periodic  # inner-X3 boundary flag
ox3_bc      = periodic  # outer-X3 boundary flag

num_threads = 1         # maximum number of OMP threads

<meshblock>
nx1 = 64
nx2 = 64
nx3 = 1

<hydro>
iso_sound_speed = 1.0         # isothermal sound speed
gamma      = 1.666666667     # gamma = C_p/C_v

<problem>
```

<output2> の dt を 0.5 くらいにすると
ファイル数が 30 になって転送が楽

<meshblock> で nx1=nx2=64, nx3=1
とすると 64 の meshblock に領域分割
（64並列にすると、1 meshblock / 1コア）

Job script

2ノードを使って64並列計算をする場合の job script 例

```
#PBS -q R6755316
#PBS -l nodes=2
#PBS -l walltime=0:20:00
#PBS -N recon

cd ${PBS_O_WORKDIR}
aprun -n 64 -N 32 -S 16 ./athena -i athinput.reconnection -t 00:18:00 > log
```

あとは qsub コマンドでジョブ投入

> qsub job_filename.job

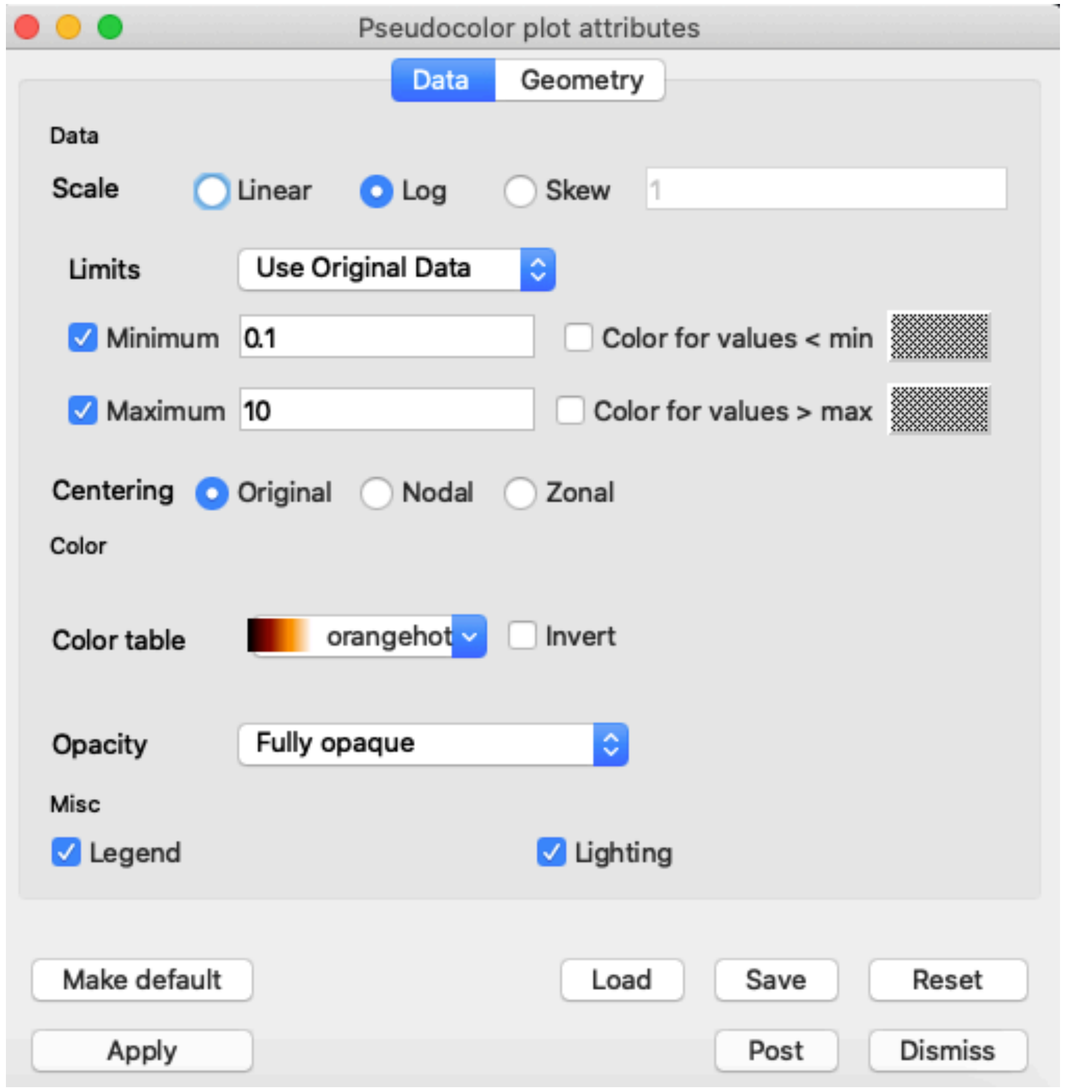
この場合、XC50 だと 1 分弱で計算が終わります

注意

- Mac の人への注意点（Windowsでも？）
  - Visit で磁力線を描くとき、一度 Bcc1, Bcc2, Bcc3 のどれかに対して Pseudocolor プロットして磁場データを読み込ませておかないと、エラーが出る

- **今回の visit を用いた磁力線の描き方だと、実は磁力線の時間変化を正しく追えないことに注意。各時刻の磁力線は同一ではない。**

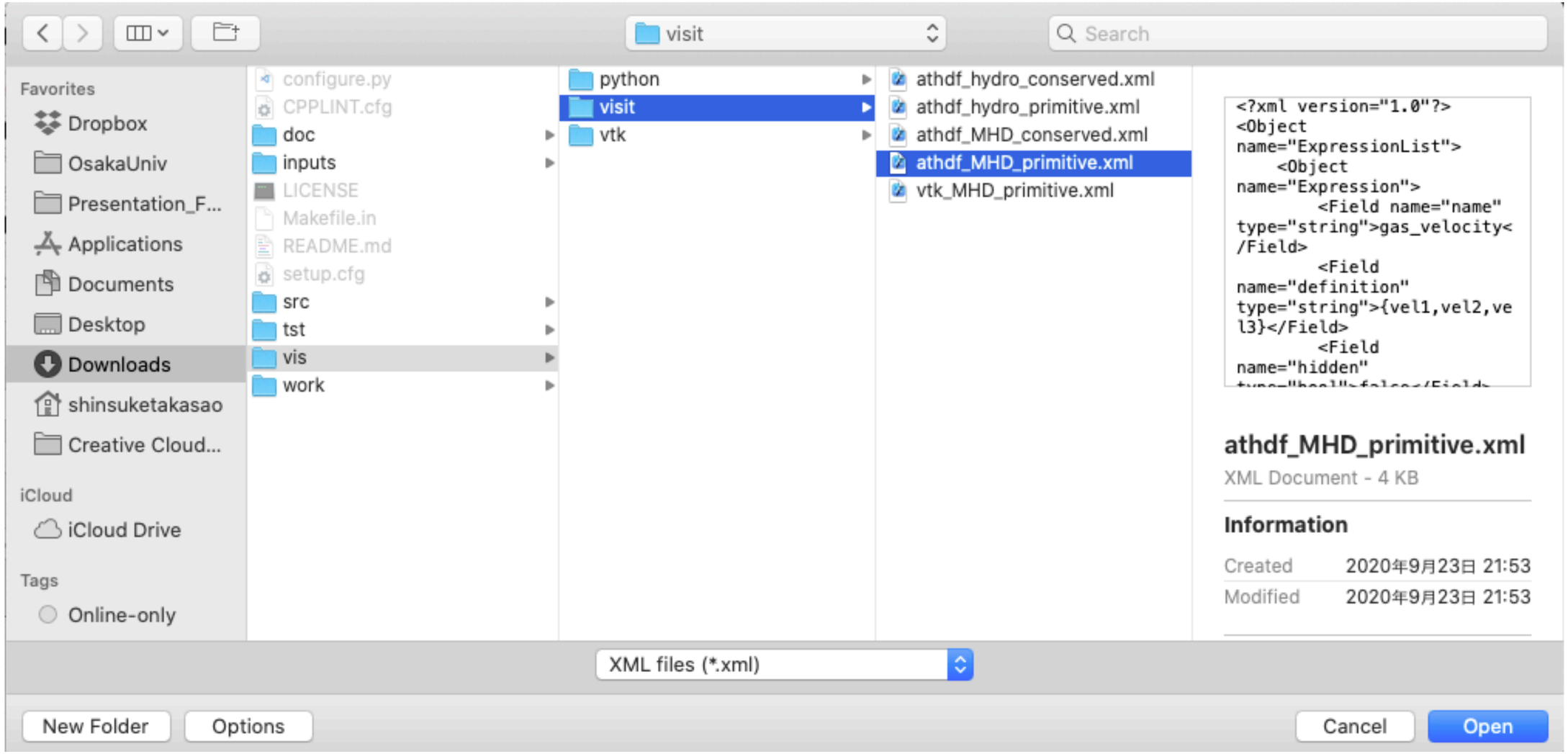
密度の表示：Pseudocolor



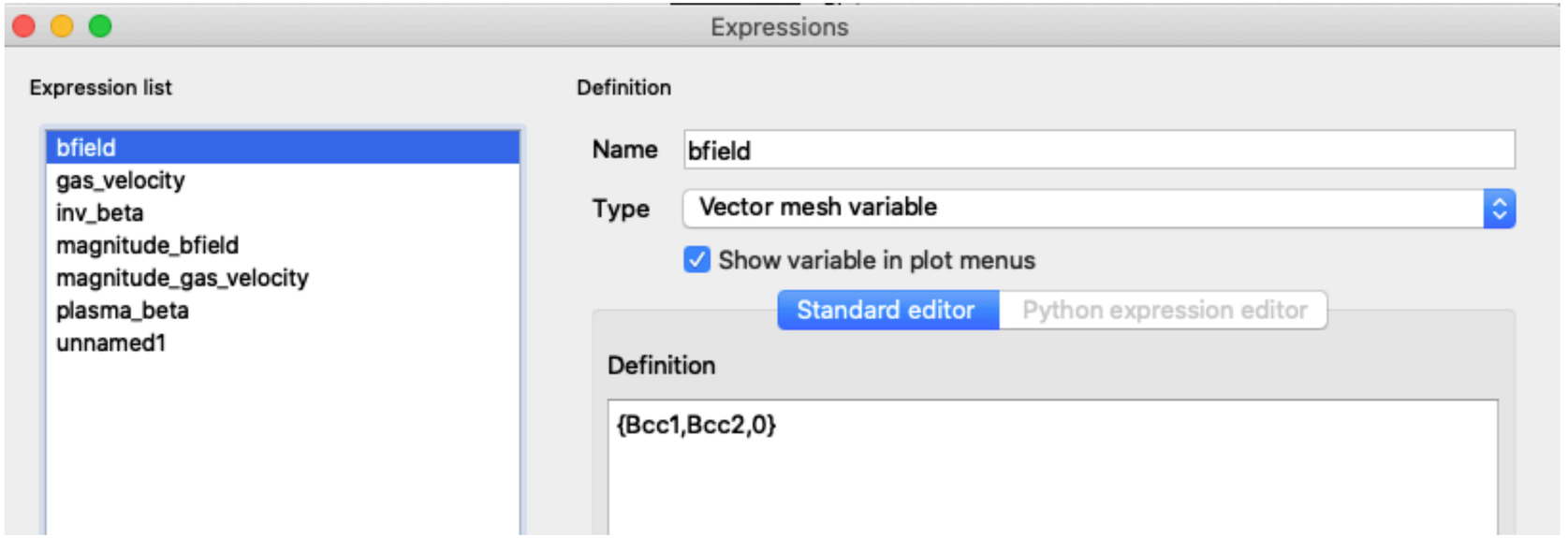
rho を可視化

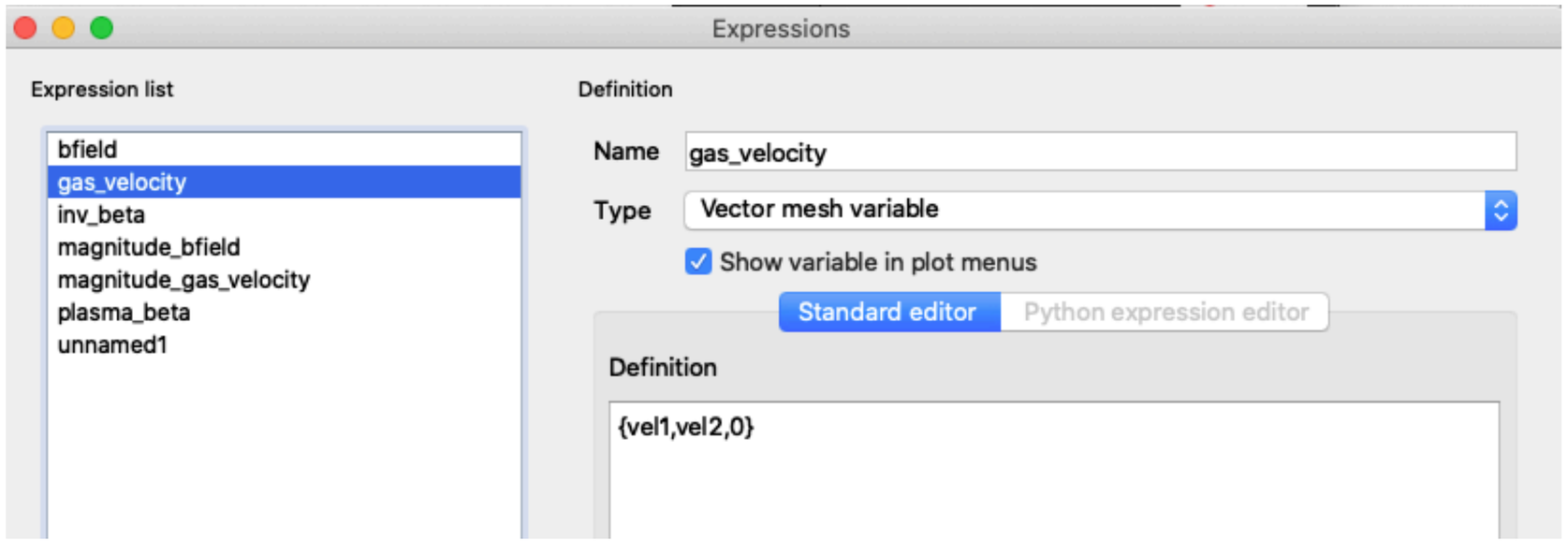
ユーザ定義の変数の準備

visit の Controls -> Expressions —> load で Athena/vis/visit の中にある .xmlファイル を読み込むと変数定義がラク



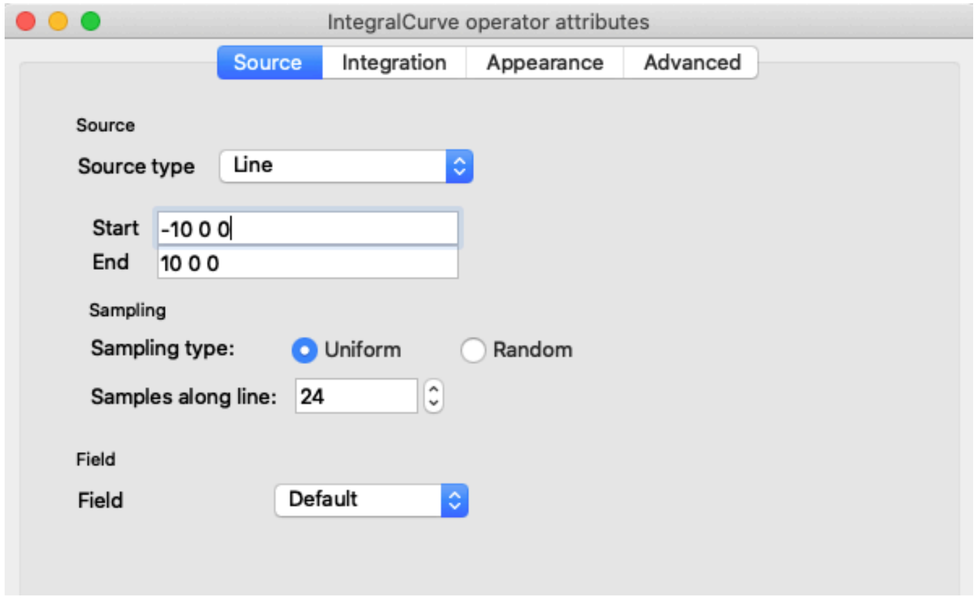
Expression での磁場・速度ベクトルの設定

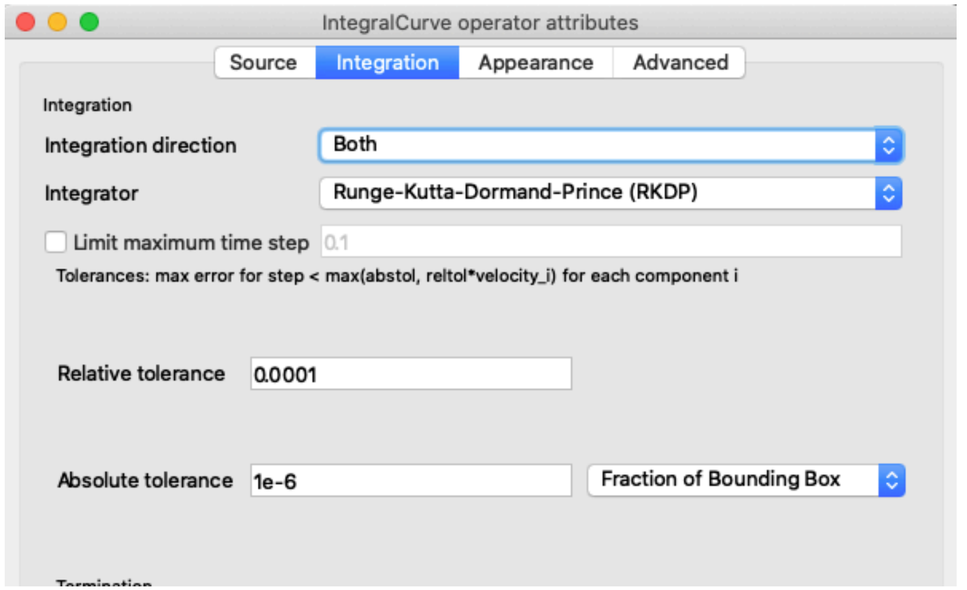


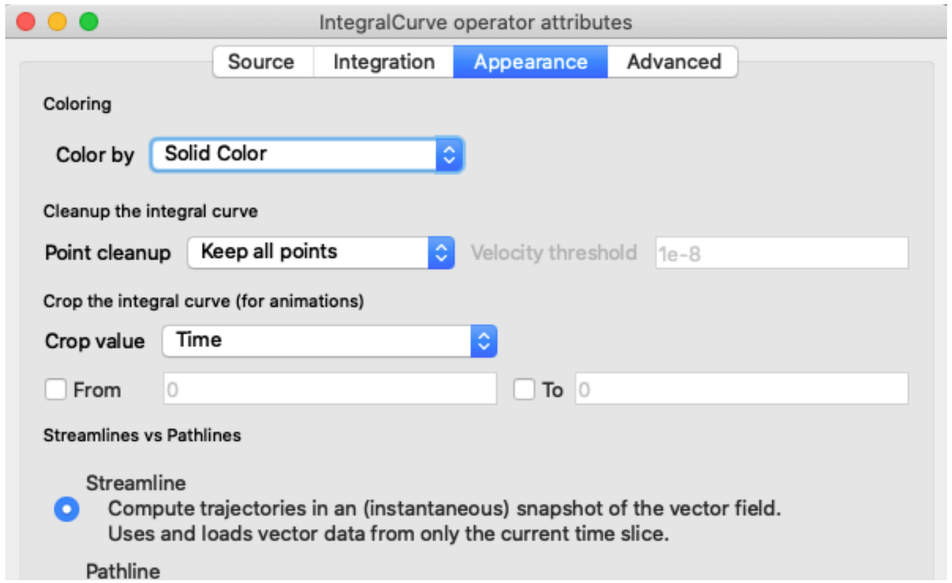


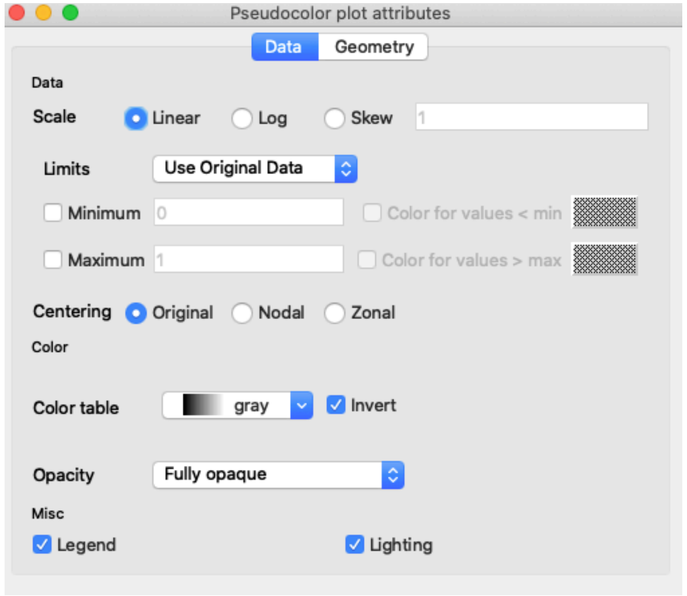
- Type は Vector
- x, y 成分にだけベクトル成分を与え、z成分は 0 にしておくこと（今2次元面でのベクトル成分だけ使いたいので。こうしないと磁力線の計算で失敗する）

磁力線の可視化：IntegralCurve の設定



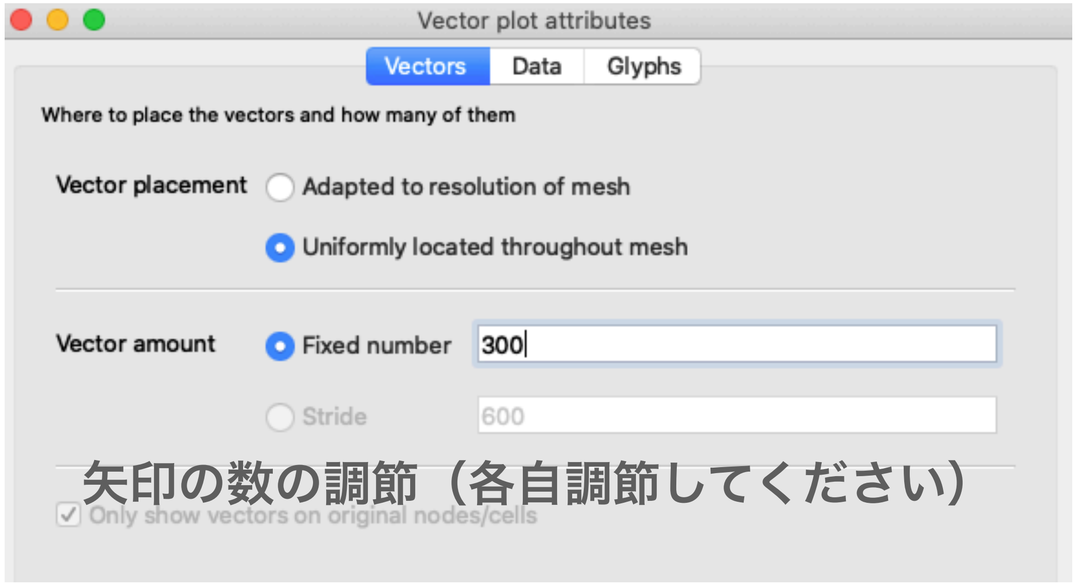


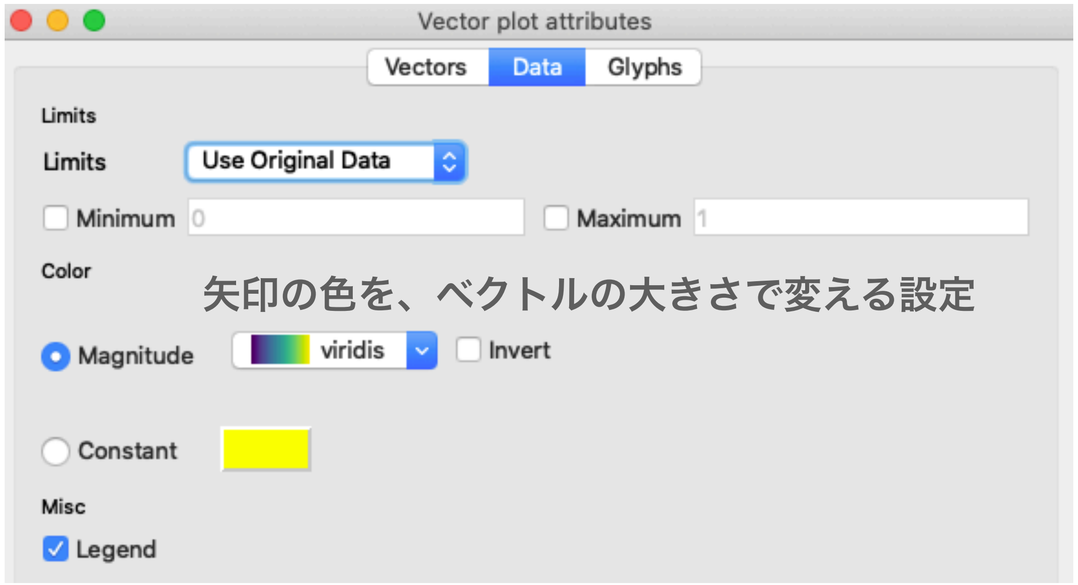


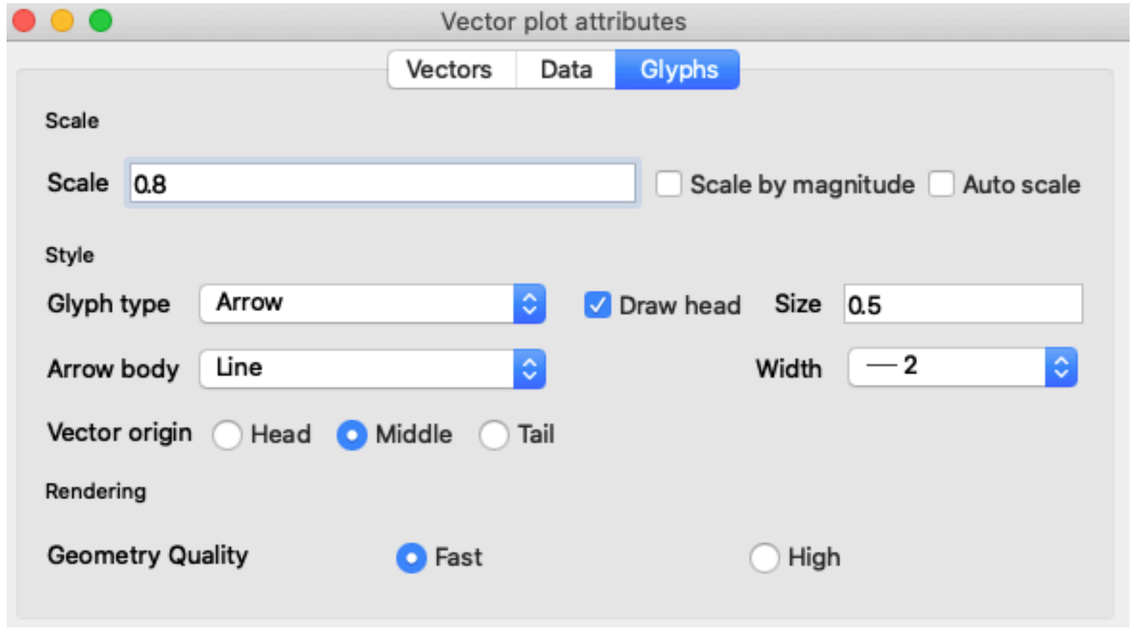


IntegralCurve の
Pseudocolor にて
磁力線の色を白にする
設定

速度ベクトルの可視化：Vector の設定







矢印の大きさ (Scale) や、矢印の頭の大きさ (head size)、
太さ（Width） の設定
各自調節してください