粒子シミュレーション法 概説

臼井英之, 杉山徹, 大村善治, 松本紘 粒子シミュレーション班

2003.8

目 次

第1章	テスト粒子解析	3
1.1	基本方程式と変数の規格化..............................	3
1.2	微分方程式を解くスキーム.............................	4
	1.2.1 Euler 法	4
	1.2.2 修正 Euler 法	4
	1.2.3 Runge-Kutta 法	5
	1.2.4 Buneman-Boris 法	5
	1.2.5 相対論効果を考慮した Buneman-Boris 法	7
	1.2.6 逆行列	7
第2章	1次元電磁粒子シミュレーション法	9
2.1	基礎方程式	9
2.2	粒子モデルの基本概念	9
2.3	電磁界定義のための空間格子点	11
2.4	時間更新チャート	12
2.5	電界及び磁界更新ルーチン	13
2.6	電流密度ルーチン	14
2.7	電荷密度ルーチンとポアソン方程式の解法 (補足)	16
2.8	クーラン条件 (補足)	17
2.9	マクスウェル分布を与える方法(補足)	18
第3章	ビームプラズマ不安定性に関する基本課題	21
3.1	背景プラズマと電子ビーム・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	21
3.2	背景プラズマとイオンビーム	23

第1章 テスト粒子解析

与えられた時間変化しない電磁場の中を、荷電粒子がどのような運動をするかを解析する 時に用いられる計算方法である。通常の粒子シミュレーションと違い、荷電粒子の運動に よる電磁場への影響を考えないという仮定は、軌道を追跡する粒子のエネルギーが、周囲 の電磁場のエネルギーよりも十分小さい場合に有効である。その仮定のもとでは、この方 法で近似的に解析可能である。

1.1 基本方程式と変数の規格化

ー般に宇宙プラズマは、十分稀薄であるため粒子間の衝突は無視できる。したがって、 相対論効果を含まない場合の個々の粒子の運動は、

$$\frac{d\boldsymbol{v}}{dt} = (\frac{Z}{M})(\boldsymbol{E} + \boldsymbol{v} \times \boldsymbol{B})$$
(1.1)

$$\frac{d\boldsymbol{x}}{dt} = \boldsymbol{v} \tag{1.2}$$

を解けば求まる。電磁場 E や B は一定値であるから、 $v \ge x$ を問題に合わせた初期値 から、時間積分するのみである。

さて、変数の値としては、実際の物理量を代入するということは行なわず、規格化した 値を用いる。規格化には、いくつかの方法があり、解析したい時空間のスケールに合わせ た規格化定数を用いれば良い。ここでは、イオンの運動を追跡する場合の変数を用いる。

速度	$oldsymbol{v}~ ightarrow$	$ ilde{m{v}_i} V_A$ アルフヴェン速
時間	$t \rightarrow$	$rac{ ilde{t}}{\Omega_i}$ ジャイロ周期
長さ	$x \rightarrow$	$\tilde{x_i} \frac{V_A}{\Omega_i}$
磁場	$oldsymbol{B}~ ightarrow$	$ ilde{m{B}} B_0$ 背景磁場強度
電場	$oldsymbol{E}~ ightarrow$	$ ilde{oldsymbol{E}} V_A B_0$
電荷	$Z \rightarrow$	$\tilde{Z}_i e$ 電荷素量
質量	$M \rightarrow$	$\tilde{M}_i M$ 陽子質量

規格化した結果の解くべき方程式は、

$$\frac{d\tilde{\boldsymbol{v}}_{\boldsymbol{i}}}{d\tilde{t}} = (\frac{\tilde{Z}_{i}}{\tilde{M}_{i}})(\tilde{\boldsymbol{E}} + \tilde{\boldsymbol{v}}_{\boldsymbol{i}} \times \tilde{\boldsymbol{B}})$$
(1.3)

$$\frac{d\tilde{\boldsymbol{x}_i}}{d\tilde{t}} = \tilde{\boldsymbol{v}_i} \tag{1.4}$$

規格化する前と形式的には同じであるが、意味合いが全く異なる。

1.2 微分方程式を解くスキーム

式 (1.3,1.4) を用いて粒子の速度と位置を時間ステップ Δt 毎に解き進めて行く数ある方 法の中のいくつかを以下に説明する。

1.2.1 Euler 法

変数 t、tの関数 v(t) として、v(t)の満たすべき微分方程式と初期条件

$$\frac{dv(t)}{dt} = f(v,t) \tag{1.5}$$

$$v(t=0) = v_0$$
 (1.6)

を与えて、v(t)を求める問題を考える。離散的に解くもっとも簡単な方法は、

$$\frac{dv}{dt} \sim \frac{v^{n+1} - v^n}{t^{n+1} - t^n} = \frac{v^{n+1} - v^n}{\Delta t}$$
(1.7)

から、

$$v^{n+1} = v^n + f(v^n, t^n) \Delta t$$
(1.8)

として計算するもので、Euler 法という。しかし、この方法は、後述の方法に比べ誤差が 大きい。

1.2.2 修正 Euler 法

 t^n から t^{n+1} まで積分するとき、 $f(v^n, t^n)$ だけを用いるのではなく、 $f(v^n, t^n)$ と予測した $f(v^{n+1}, t^{n+1})$ を用いると精度が良くなる。この方法は、修正 Euler 法という。予測の方法は、

$$k_1 = f(v^n, t^n) \tag{1.9}$$

$$v^{n+1} = v^n + k_1 \Delta t (1.10)$$

$$k_2 = f(v^{n+1}, t^n + \Delta t) \tag{1.11}$$

と予測値を求め、

$$v^{n+1} = v^n + \frac{1}{2}(k_1 + k_2)\Delta t$$
 (1.12)

と、 v^{n+1} を求めなおす。

1.2.3 Runge-Kutta 法

さらに、4つの予測値を使う方法が、ルンゲクッタ法である。

$$k_1 = f(v^n, t^n) \tag{1.13}$$

$$v_2 = v^n + k_1 \frac{\Delta t}{2} \tag{1.14}$$

$$k_2 = f(v_2, t^n + \frac{\Delta t}{2}) \tag{1.15}$$

$$v_3 = v^n + k_2 \frac{\Delta t}{2}$$
(1.16)

$$k_3 = f(v_3, t^n + \frac{\Delta t}{2}) \tag{1.17}$$

$$v_4 = v^n + k_3 \Delta t \tag{1.18}$$

$$k_4 = f(v_4, t^n + \Delta t) \tag{1.19}$$

と予測値を求め、

$$v^{n+1} = v^n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)\Delta t$$
 (1.20)

と、 v^{n+1} 求める。

1.2.4 Buneman-Boris 法

式(1.3)の差分表現は以下のようになる。

$$\frac{\boldsymbol{v}^{n+1/2} - \boldsymbol{v}^{n-1/2}}{\Delta t} = \frac{q}{m} (\boldsymbol{E}^n + \frac{\boldsymbol{v}^{n+1/2} + \boldsymbol{v}^{n-1/2}}{2} \times \boldsymbol{B}^n)$$
(1.21)

式を見てわかるように、時間 $n\Delta t$ における式展開になっている。ただし、E, B は粒子位置での電界値および磁界値である。この式から $v^{n+1/2}$ の値を計算するには以下のような方法を用いる。

まず、新しい変数として v⁻ と v⁺ を以下のように定義し導入する。

$$\boldsymbol{v}^{-} = \boldsymbol{v}^{n-1/2} + \frac{q}{m} \boldsymbol{E}^{n} \frac{\Delta t}{2}$$
(1.22)

$$\boldsymbol{v}^{+} = \boldsymbol{v}^{n+1/2} - \frac{q}{m} \boldsymbol{E}^{n} \frac{\Delta t}{2}$$
(1.23)

すなわち、 $v^{n-1/2}$ から電界 E^n で $\Delta t/2$ だけ加速を受けた後の速度、および $v^{n+1/2}$ から電 界 E^n で $\Delta t/2$ だけ加速を受ける前の速度を意味する。これらの変数を用いて式 (1.21) を書 き換えると以下のようになる。

$$\frac{\boldsymbol{v}^{+} - \boldsymbol{v}^{-}}{\Delta t} = \frac{1}{2} \frac{q}{m} (\boldsymbol{v}^{+} + \boldsymbol{v}^{-}) \times \boldsymbol{B}^{n}$$
(1.24)

この意味は、 v^+ から v^- へ変化する間に $\frac{q}{m}v \times B$ のローレンツ力によるサイクロトロン回転のみが作用するということである。



 \boxtimes 1.1: Vector relation in Byneman-Boris method

この式の両辺について ($v^+ + v^-$) との内積をとると (v^+)² = (v^-)² となる。すなわち、図 1.1 に示されたように、式 (1.24) は v^+ は v^- は大きさが同じで、角度 θ だけ回転させたも のであることを示す。つまり、式 (1.21) を $\Delta t/2$ 分の電界による加速 2 回と、 Δt 分のサイ クロトロン回転とに分離したことになる。

詳細は省くが、式(1.24)を整理すると、

$$\boldsymbol{v}^{+} = \boldsymbol{v}^{-} + \frac{2}{1+\boldsymbol{T}^{2}}(\boldsymbol{v}^{-} + \boldsymbol{v}^{-} \times \boldsymbol{T}) \times \boldsymbol{T}$$
(1.25)

となる。ただし、 $T = (q/m)\Delta t B^n/2$ と定義する。上の式で括弧内を v^o とすると、 v^- から v^+ への計算は

$$\boldsymbol{v}^o = \boldsymbol{v}^- + \boldsymbol{v}^- \times \boldsymbol{T} \tag{1.26}$$

$$\boldsymbol{v}^+ = \boldsymbol{v}^- + \boldsymbol{v}^o \times \boldsymbol{S} \tag{1.27}$$

ただし、 $S = 2T/(1 + T^2)$ とする。

以上をまとめると、粒子の速度更新には以下の4ステップの計算を行うことになる。

$$\boldsymbol{v}^{-} = \boldsymbol{v}^{n-1/2} + \frac{q}{m} \boldsymbol{E}^{n} \frac{\Delta t}{2}$$
(1.28)

$$\boldsymbol{v}^o = \boldsymbol{v}^- + \boldsymbol{v}^- \times \boldsymbol{T} \tag{1.29}$$

$$\boldsymbol{v}^+ = \boldsymbol{v}^- + \boldsymbol{v}^o \times \boldsymbol{S} \tag{1.30}$$

$$\boldsymbol{v}^{n+1/2} = \boldsymbol{v}^+ + \frac{q}{m} \boldsymbol{E}^n \frac{\Delta t}{2}$$
(1.31)

この計算方法を Buneman-Boris 法という。

速度がわかれば粒子位置 r は速度 v を時間的に積分することにより得られる。

$$x^{n+1} = x^n + v_x^{n+1/2} \Delta t \tag{1.32}$$

1.2.5 相対論効果を考慮した Buneman-Boris 法

相対論効果を考慮した場合の運動方程式は

$$\frac{d(m\boldsymbol{v})}{dt} = q(\boldsymbol{E} + \boldsymbol{v} \times \boldsymbol{B})$$
(1.33)

$$m = \gamma m_0 \tag{1.34}$$

$$\gamma = \frac{1}{\sqrt{1 - (v/c)^2}}$$
(1.35)

となる。ここで、 $u = \gamma v$ という変数を導入する。すなわち、

$$\boldsymbol{u} = \frac{c}{\sqrt{c^2 - |\boldsymbol{v}|^2}} \boldsymbol{v} \tag{1.36}$$

と定義される uを用意する。vに書き直すと、

$$\boldsymbol{v} = \frac{c}{\sqrt{c^2 + |\boldsymbol{u}|^2}} \boldsymbol{u} \tag{1.37}$$

となる。これを1.33に代入して整理すると、

$$\frac{d\boldsymbol{u}}{dt} = \frac{q}{m_0} (\boldsymbol{E} + \frac{c}{\sqrt{c^2 + |\boldsymbol{u}|^2}} \boldsymbol{u} \times \boldsymbol{B})$$
(1.38)

上式において、修正磁場を次のように定義する。

$$\boldsymbol{B}_{u} = \frac{c}{\sqrt{c^{2} + |\boldsymbol{u}|^{2}}} \boldsymbol{B}$$
(1.39)

そうすると、運動方程式は

$$\frac{d\boldsymbol{u}}{dt} = \frac{q}{m_0} (\boldsymbol{E} + \boldsymbol{u} \times \boldsymbol{B}_u)$$
(1.40)

となり、uについて前述の Bunema-Boris 法を用いて速度更新を行うことができる。つまり、事前に

$$\boldsymbol{u}^{n-1/2} = \frac{c}{\sqrt{c^2 - |\boldsymbol{v}^{n-1/2}|^2}} \boldsymbol{v}^{n-1/2}$$
(1.41)

としてvからuへ変換しておき、このuと修正磁場 B_u を用いてBuneman-Boris法でuを時間更新し、

$$\boldsymbol{v}^{n+1/2} = \frac{c}{\sqrt{c^2 + |\boldsymbol{u}^{n+1/2}|^2}} \boldsymbol{u}^{n+1/2}$$
(1.42)

で*u*から*v*に戻す。

1.2.6 逆行列

一般性を欠くが、以下のような方法もある。荷電粒子の運動の場合、

$$m\frac{v^{n+1/2} - v^{n-1/2}}{\Delta t} = q\mathbf{E}^n + q\frac{v^{n+1/2} + v^{n-1/2}}{2} \times \mathbf{B}^n$$
(1.43)

と、運動方程式は書かれる。成分で書くと、

$$m \begin{pmatrix} v_x^{n+1/2} - v_x^{n-1/2} \\ v_y^{n+1/2} - v_y^{n-1/2} \\ v_z^{n+1/2} - v_z^{n-1/2} \end{pmatrix} \Delta t = q \begin{pmatrix} E_x^n \\ E_y^n \\ E_z^n \end{pmatrix} + q \frac{1}{2} \begin{pmatrix} v_x^{n+1/2} + v_x^{n-1/2} \\ v_y^{n+1/2} + v_y^{n-1/2} \\ v_z^{n+1/2} + v_z^{n-1/2} \end{pmatrix} \times \begin{pmatrix} B_x^n \\ B_y^n \\ B_z^n \end{pmatrix}$$
(1.44)

これを $v^{n+1/2}$ について解くと、

$$v^{n+1/2} = A(\frac{q}{m}E\Delta t + 2v^{n-1/2}) - v^{n-1/2}$$
 (1.45)

$$\mathbf{A} = \frac{1}{D} \begin{pmatrix} 4 + B_x^2 & B_x B_y + 2B_z & B_z B_x + 2B_y \\ B_x B_y + 2B_z & 4 + B_y^2 & B_y B_z + 2B_x \\ B_z B_x + 2B_y & B_y B_z + 2B_x & 4 + B_z^2 \end{pmatrix}$$
(1.46)

$$D = 4 + B_x^2 + B_y^2 + B_z^2 (1.47)$$

$$D = 4 + B_x^2 + B_y^2 + B_z^2$$
(1.47)

$$B_x = \frac{q}{m} B_x^n \Delta t$$
(1.48)

$$B_x = -\frac{q}{m} B_x^n \Delta t$$
(1.49)

$$B_y = \frac{q}{m} B_y^n \Delta t \tag{1.49}$$

$$B_z = \frac{q}{m} B_z^n \Delta t \tag{1.50}$$

となり、時間を進めることができる。

第2章 1次元電磁粒子シミュレーション法

ここでは、宇宙プラズマを構成する電子、イオンを粒子として扱う電磁粒子シミュレーショ ン手法について概説する。

2.1 基礎方程式

本シミュレーションコードで用いる基礎方程式は衝突項を含まない粒子の運動方程式(式 (1.1)) および以下に示した Maxwell の電磁方程式である。

$$\nabla \times \boldsymbol{B} = \mu_o \boldsymbol{J} + \frac{1}{c^2} \frac{\partial \boldsymbol{E}}{\partial t}$$
(2.1)

$$\nabla \times \boldsymbol{E} = -\frac{\partial \boldsymbol{B}}{\partial t} \tag{2.2}$$

$$\nabla \cdot \boldsymbol{E} = \frac{\rho}{\epsilon_0} \tag{2.3}$$

$$\nabla \cdot \boldsymbol{B} = 0 \tag{2.4}$$

ここで、*c*, *J*, *ρ*はそれぞれ、光速、電流密度ベクトル、電荷密度を示す。

Maxwell 方程式のうち、式 (2.3),式 (2.4) は時間 t=0 で満足されておれば、それ以後の時間においては自動的に満足されるので、基本的には式 (2.1),式 (2.2) だけ解けばよい。すなわち、シミュレーションシステム内の電界の初期値を求めるためには式 (2.3) のポアソン式を解く必要がある。また、式 (2.4) より、1 次元シミュレーション方向での磁場の変動は常にゼロである。

2.2 粒子モデルの基本概念

前述のテスト粒子解析とは違い、粒子シミュレーションでは、粒子更新による電流や電荷の変化が電磁界に影響を与え、その影響が粒子にフィードバックされる。すなわち、粒子シミュレーションにおいて、プラズマや電磁界の時間空間発展は粒子の運動方程式と Maxwell 方程式を互いに解き進めることによって得られるわけである。運動方程式と Maxwell 方程式を繋げる物理量は電流密度 j と電荷密度 ρ であり、これらはプラズマ各粒子の運動を決定する速度 v, r で表される。しかし、現実のプラズマは、たとえば、デバイ長を単位とする立方体内に非常に多数の粒子が含まれるため、(e.q., 磁気圏プラズマにおいては $n\lambda_D \sim 10^9$ 程度)、粒子一つ一の運動をすべて追跡することは計算機資源から見ても非現実的であ



図 2.1: 超粒子の形状および Shape factor

る。また、粒子の運動を追跡する場合、粒子軌跡は連続であるため任意の粒子位置において電磁界が必要となる。

これらの問題を解決するために PIC法 (Particle-In-Cell)を用いる。まず、多くの現実の プラズマ粒子を代表する大きい電荷、大きい質量をもつ超粒子 (superparticle)を考え、こ の超粒子を多数用いることによりプラズマ環境を再現する。この超粒子に空間的にある大 きさをもたせる方がより現実のプラズマの性質を模擬することができる。今回は、簡単の ため図 2.1 に示すような長方形形状を採用する。すなわち、大きさは空間格子点間隔 Δx を もつとする。

また、超粒子はシミュレーションシステム内で任意の位置を取ることができる。超粒子 に働く電磁界は粒子同士のクーロン力を直接計算するのではなく、シミュレーション空間 に離散点として定義された空間格子点にその粒子情報(電流、電荷)を一旦配分し、それ らの情報を用いて同じく格子点上で定義された電磁界成分について Maxwell 方程式を用い て更新する。

上述のように超粒子は点電荷ではなく広がりがあるため、格子点に電荷量を配分すると きにはqS(X' - x)に従って行う。この重み関数Sを shape factor と呼び、図 2.1 に示す。 ただしX'は格子点位置、Xは粒子の中心とする。各超粒子の運動を更新する際には、そ の隣接する格子点上の電磁界を超粒子位置に内挿しそれを用いて運動方程式を解く。すな わち、超粒子の情報を格子点を介して電磁界更新に用い、その電磁界を再び粒子更新に用 いるのが PIC 法である。粒子の速度vおよび位置r更新は上述したが、電磁界の格子点で の扱いについて以下に述べる。



図 2.2: 整数格子と半整数格子および各電磁界の定義点

2.3 電磁界定義のための空間格子点

テスト粒子シミュレーションでは、粒子ダイナミクスが電磁界に影響を及ぼさないため、 電磁界を Maxwell 方程式で解き進める必要がなかった。しかし、一般に、超粒子の運動は 電流と等価であり、これによりシステム内の電磁界は変動する。この超粒子プラズマダイ ナミクスによる電流の寄与を Maxwell 方程式に取り込み、電磁界を解き進めるには、それ らの各成分を時間的、空間的に離散点に定義する必要がある。これは、Maxwell 方程式を 差分形式で解くためである。ここでは、まず空間格子点について説明する。

Maxwell 方程式を中心差分で解き進めるため、シミュレーションシステム内に整数格子 群 $i\Delta x(i = 1, 2, 3, ..., N = x)$ と半整数格子群 $(i + 1/2)\Delta x$ を用意する。図 2.2 に示すよう に E_y, B_y, J_y, ρ は整数格子点に定義され、その他の電磁界成分は半整数格子点に定義され る。電界成分と磁界成分は互いに入れ子に配置されているが、これは、Maxwell 方程式を x 方向の空間変動のみを考慮した 1 次元空間で展開してみれば理解できる。また、電流密 度 J_x, J_y, J_z はそれぞれ電界成分 E_x, E_y, E_z と同じ格子点に定義する。ただし、整数格子、 半整数格子とは便宜上の呼び名であり、シミュレーションコード内では、電磁界の配列の 添え字は整数であるので混乱しないように注意する。

前章でBuneman-Boris法による粒子速度更新について述べたが、その際に、粒子位置で の電磁界の値が必要となる。粒子位置での電磁界は粒子位置に隣接する2つの格子点上の 電磁界成分を用いて内挿することにより求める。ただ、上述のように電磁界成分は整数格 子、半整数格子の二つの格子システムに定義されているため、各粒子位置への内挿法はそ れぞれ違う。また、粒子位置での電磁界成分を求める際には、静電的および静磁的ないわ ゆる"セルフフォース"に注意する必要がある。以下にそれについて簡単に述べる。

後述するが、静電界成分は電荷密度を用いてポアソン方程式を満たすように得られる。 すなわち、

$$\frac{\partial E_x}{\partial x} = \rho \tag{2.5}$$

が満たされ、 $E_{x,i+1/2} - E_{x,i-1/2} = \rho_i$ という関係がシミュレーション内で得られる(簡単の

ために $\epsilon_0 = 1$ としている。)。見てわかるように、 E_x は半整数格子点での定義であるため、 電荷密度 ρ は整数格子点で定義される。すなわち、シミュレーションシステム内に分布す るプラズマ粒子の各電荷量は各粒子に隣接する整数格子点に線形的に配分され電荷密度 ρ が得られ、その情報を元に式 (2.5) から半整数格子点において電界値 E_x が求められる。

さて、運動方程式を用いて粒子速度を更新する場合は粒子位置での電界値が必要である が、これを上に定義された半整数格子点から内挿して求めると正しい電界値が得られない。 たとえば、速度を持たない1粒子をある点におき、上のようにして求めた電界値を用いる と粒子位置での電界値は本来ゼロであるべきが、ゼロにはならない。これをセルフフォー スという。詳細は省くがこのセルフフォースを回避するには、Maxwell 方程式を解くため に半整数格子に定義されていた電界値 E_x を粒子更新の際にはもともとの電荷配分点であ る整数格子点に再配分し、その値を用いて粒子位置に電界値を内挿する。すなわち粒子速 度更新の際には $E_{x,i} = (E_{x,i-1/2} + E_{x,i+1/2})/2$ で得られた値を用いて粒子位置に電界値を内 挿するとうまくいく。つまり、セルフフォース回避の原則として、粒子電荷を配分した格 子点での電界成分を用いて粒子位置での電界成分を内挿することが言える。同じことが電 流密度 J と磁界 B についてもいえる。Maxwell 式より 1 次元では B_y , B_z は

$$B_{z,i+1/2} - B_{z,i-1/2} = -\frac{J_{y,i}}{c^2}$$
(2.6)

$$B_{y,i+1} - B_{y,i} = \frac{J_{z,i+1/2}}{c^2}$$
(2.7)

となり B_z と J_y および B_y と J_z についてもそれぞれ互いに入れ子になる必要があるため違う格子点で定義される。しかし、電流密度は粒子情報から直接得られるので、粒子速度更新の際に磁界を用いる場合は電流密度が定義された格子点に磁界を再配置する必要がある。

2.4 時間更新チャート

ここでは、電磁界と粒子の時間発展を解き進める手順を概説する。フローチャートを図 2.3 に示す。時間についても空間同様、整数格子群 $n\Delta t$ 、半整数格子群 $(n+1/2)\Delta t$ の 2 つ を用意し、時間的な中心差分に対応させる。

- 1. 初期のプラズマ分布より各格子点における電界密度 ρ を求め、ポアソン式を解き、初 期電界を求める。
- 2. 半ステップ、磁界**B**を解く。
- 3. $t = n\Delta t$ における電界 **E**、磁界 **B** が求まったので運動方程式より $t = (n+1/2)\Delta t$ における各超粒子の速度 v を求める。
- 4. $t = (n + 1/2)\Delta t$ の速度を用いて粒子位置 r を半ステップ分進める。
- 5. $t = (n + 1/2)\Delta t$ における各粒子の位置、速度が決まったので電流密度 J を各格子点で計算する。
- 6. 磁界 B を更に半ステップ進める。



図 2.3: 時間更新チャート

7. $t = (n+1)\Delta t$ における電界 **E** を求める。

8. $t = (n + 1/2)\Delta t$ の速度 v を用いて更に半ステップ各粒子の位置を進める。

以上、1ステップ (Δt)分、電磁界および粒子ダイナミクスが時間更新される。これらを繰り返し行うことにより、シミュレーションシステム内でのプラズマ現象の空間・時間発展 を解き進めることができる。

2.5 電界及び磁界更新ルーチン

電界の時間更新には式(2.1)を用いる。1次元モデルでは、解くべき式は以下の3式である。

$$\frac{\partial E_x}{\partial t} = -\frac{J_x}{\epsilon_0} \tag{2.8}$$

$$\frac{\partial E_y}{\partial t} = -c^2 \frac{\partial B_z}{\partial x} - \frac{J_y}{\epsilon_0}$$
(2.9)

第2章 1次元電磁粒子シミュレーション法

$$\frac{\partial E_z}{\partial t} = c^2 \frac{\partial B_y}{\partial x} - \frac{J_z}{\epsilon_0}$$
(2.10)

また、磁界の時間更新には式 (2.2) を用いる。1 次元モデルでは、解くべき式は以下の2 式である。式 (2.4) から B_x の変動はないので解かない。

$$\frac{\partial B_y}{\partial t} = \frac{\partial E_z}{\partial x} \tag{2.11}$$

$$\frac{\partial B_z}{\partial t} = -\frac{\partial E_y}{\partial x} \tag{2.12}$$

ただし、図 2.3 の時間更新チャートであるように、磁場は運動方程式を解く際に電界と時間ステップをあわせる必要があるため、 $\Delta t/2$ 分を2回解くことになる。

2.6 電流密度ルーチン

電流密度 J は各超粒子の速度と位置から計算する。Maxwell 方程式の差分形から、 J_x と J_z は半整数格子点、 J_y は整数格子点に定義されているが、簡単のため、一旦、半整数格子 点で電流密度 J を計算し、その後 J_y を整数格子に再配置することにする。

 $J_y \ge J_z$ については、後述する電荷密度の計算と同じ方法で超粒子のモーメントを隣接 する2つの格子点に線形重みで分配する。 J_x については電荷保存法を用いる。これはシス テム方向の電荷の連続式

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \boldsymbol{J} = 0 \tag{2.13}$$

を常に満たすように電流を求める方法である。差分形で書くと、

$$\rho_i^{n+1} - \rho_i^n = -(J_{i+1/2}^{n+1/2} - J_{i-1/2}^{n+1/2})\frac{\Delta t}{\Delta x}$$
(2.14)

となる。

超粒子が1タイムステップ Δt の時間に1空間格子 Δx 以上移動しない場合、 $J_x^{n+1/2}$ を計算するには、2つの場合が考えられる。まず第一の場合は、図2.4の上のパネルに示したように粒子が同じ格子内の移動であり、第二の場合は、同図の下のパネルにあるように2つの格子が関係する場合である。

第一の場合は、格子点 $X_{i+1/2}$ における電流値 $I_{i+1/2}$ は時間ステップ Δt 内に $X_{i+1/2}$ の点を通過した電荷量を計算することで得られる。すなわち、

$$I_{i+1/2} = \frac{q_A - q_B}{\Delta t} \tag{2.15}$$

となる。ただし、q_A, q_B は次のように与えられる。

$$q_A = q \frac{X_{i+1} - x_p(n\Delta t)}{\Delta x}, \qquad q_B = q \frac{X_{i+1} - x_p((n+1)\Delta t)}{\Delta x}$$
 (2.16)

また、第二の場合は、粒子運動が $X_{i+1/2}$ と $X_{i+3/2}$ の2つの格子点での電流に寄与する。 すなわち、

$$I_{i+1/2} = \frac{q_A}{\Delta t}, \qquad I_{i+3/2} = -\frac{q_B}{\Delta t}$$
 (2.17)



図 2.4: 電荷保存法による電流密度計算(ケース1(上)とケース2(下))



図 2.5: 領域比例配分による電荷密度計算

となる。ただし、図では、粒子速度が正を仮定したが、負の速度を持つ場合、式 (2.15) と 式 (2.17)の左辺に-1 をかける必要がある。

この電荷保存法では、粒子位置の情報が $n\Delta t \ge (n+1)\Delta t$ の時間において得られれば $(n+1/2)\Delta t$ での電流密度が求まる。そのため、前述の時間更新チャートにおいて、粒子位 置は $(n+1/2)\Delta t$ において必要ないように思われる。3 次元モデルではこれは正しいが、1 次元、2 次元モデルでは、空間グリッドが存在しない方向の電流密度成分は、後述する電荷 密度計算と同様の方法で超粒子モーメントを格子点に集めることによって計算する。この ため、前述の時間更新チャートのように、粒子を半時間ステップずつ動かし、 $(n+1/2)\Delta t$ における粒子位置と速度の情報が必要となる。

以上の方法は、既に公開されている KEMPO1 シミュレーションコードにおいて、オリジナ ルな電流ルーチンとして実装されている。(Omura, Y. and H. Matsumoto, KEMPO1: Technical Guide to One-Dimensional Electromagnetic Particle Code, *Computer Space Plasma Physics: Simulation Techniques and Softwares*, ed. by H. Matsumoto and Y. Omura, pages 21-65, Terra Scientific, Tokyo, 1993. (http://www.terrapub.co.jp/e-library/cspp/index.html))

2.7 電荷密度ルーチンとポアソン方程式の解法 (補足)

電界の初期値を得るためには、シミュレーションのはじめにポアソン方程式を解く必要 がある。これにより、初期の電荷密度分布による電界分布が得られ、この電界を用いて、 上述の電界更新を行う。ポアソン方程式にはシステム内の電荷密度 *ρ* が必要であるが、こ れを求めるにはシステム内に存在する超粒子の電荷を各格子点に配分する必要がある。基本的には図 2.1 に示した重み関数 *S* を用いて各格子点に電荷を配分する。その具体的な方法を図 2.5 に詳しく示す。超粒子の形は幅 Δx の長方形とし、その中心位置を x_p とすると、もし $X_i \leq x_p \leq X_{i+1}$ なら格子点 $X_i \geq X_{i+1}$ に粒子の電荷 q は配分される。すなわち、 $q(x_p - X_i)/\Delta x$ は $\rho(X_{i+1})$ に配分され、 $q(X_{i+1} - x_p)/\Delta x$ は $\rho(X_i)$ に配分される。

電荷密度分布が得られれば、それを用いてポアソン方程式を解く。

$$\nabla^2 \phi = -\rho \tag{2.18}$$

差分形式では次にように表される。

$$\frac{\partial^2 \phi}{\partial x^2} = \frac{\phi(x_{i+1}) - 2\phi(x_i) + \phi(x_{i-1})}{(\Delta x)^2}$$
(2.19)

ポアソン式の解法はさまざまなものが提唱されているが、空間的に周期境界のもとでは、 高速フーリエ変換を用いた方法が一般的である。すなわち、電荷密度分布 $\rho(x_i)$ をフーリ エ変換し $\rho(k_m)$ を求め、次式から波数空間 k において電位 $\phi(k_m)$ を求める。

$$K_m^2 \phi(k_m) = \rho(k_m) \tag{2.20}$$

 $k_m = 2\pi m/L: m = 1, 2, ..., N_x/2$ であり、

$$K_m = \frac{\sin(\mathbf{k}_m \Delta \mathbf{x}/2)}{\Delta x/2} \tag{2.21}$$

得られた電位 $\phi(k_m)$ を逆フーリエ変換することにより各格子点での $\phi(x_i)$ 、すなわち ϕ_i を 求める。電界値 $E_{x,i+1/2}$ は次式から求まる。

$$E_{x,i+1/2} = \frac{\phi_i - \phi_{i+1}}{\Delta x}$$
(2.22)

上述の電荷保存法による電流計算を用いれば、基本的にポアソン方程式が満たされるの で電荷密度と静電界の関係は正しく解かれていることになる。しかし、電流密度を電荷密 度と同じ方法(各時間ステップにおいて粒子のモーメントを隣接する2つの格子点に配分 することにより電流密度を計算)で求めた場合、電荷の連続の式が常に満たされていると は限らない。この場合、ある一定時間間隔でポアソン方程式を陽に解くことにより静電界 を補正することが必要になる。

2.8 クーラン条件 (補足)

Maxwell 方程式を空間的に中心差分、時間的にリープフロッグで解き進めるには、上述 にあるように格子間隔 Δx と時間ステップ Δt が以下の関係を満足する必要がある。

$$\Delta x > c \Delta t \tag{2.23}$$

ここで c は光速である。これをクーラン条件という。このクーラン条件は光速波動の数値 分散関係から簡単に導かれる。 ϕ 、A(x,t)を波数 k 周波数 ω を持つ波の成分と仮定する。

$$A(x,t) = A_0 \exp(ikx - i\omega t) \tag{2.24}$$

空間的な中心差分を考えると

$$\frac{\Delta A}{\Delta x} = \frac{A(x_0 + \Delta x/2, t) - A(x_0 - \Delta x/2, t)}{\Delta x}$$
(2.25)

$$= \frac{\exp(ik\Delta x/2) - \exp(-ik\Delta x/2)}{\Delta x} A(x_0, t)$$
(2.26)

$$= i \frac{\sin(\mathbf{k}\Delta \mathbf{x}/2)}{\Delta x/2} A(x_0, t)$$
(2.27)

この $\Delta A/\Delta x$ を空間変微分 $\partial A/\partial x$ と比較すると、波数 k は以下のように変数 K に置き換えて考えることができる。

$$K = \frac{\sin(\mathbf{k}\Delta\mathbf{x}/2)}{\Delta x/2} \tag{2.28}$$

同じように周波数ωについても変数Ωに以下のように置き換えることができる。

$$\Omega = \frac{\sin(\omega\Delta t/2)}{\Delta t/2} \tag{2.29}$$

光速モードの波動分散関係は $\omega^2 = c^2 k^2$ であり、 $k \ge \omega$ を上の $K \ge \Omega$ で置き換える $\Omega^2 = c^2 K^2 \ge c^2 K^2 \le c^2 K^2 \ge c^2 K^2 \le c^2 K^2 \ge c^2 K^2 \le c^2 K^2$

$$\sin^2(\omega\Delta t/2) = (\frac{c\Delta t}{\Delta x})^2$$
(2.30)

という関係が得られる。もし $c\Delta t/\Delta x < 1$ なら、 ω は複素数となり数値不安定を示す。このため、最初に示したクーラン条件が満たされてないといけないことになる。 $c\Delta t/\Delta x = 1$ の場合、臨界安定である。

2.9 マクスウェル分布を与える方法(補足)

ここでは、0 < x < 1 の一様乱数から、ガウス乱数を生成する方法の一例を述べる。粒子シミュレーションでは、初期の粒子の速度分布をマクスウェル分布で与えることが多いためである。

一様乱数 x

$$p(x)dx = \begin{cases} dx & 0 < x < 1\\ 0 & \text{other} \end{cases}$$
(2.31)

を任意の関数で y(x) に変換した場合、y(x) の確立密度 P(y) は、

$$|P(y)dy| = |p(x)dx| \tag{2.32}$$

となり、

$$P(y) = p(x) \left| \frac{dx}{dy} \right| \tag{2.33}$$

と求まる。これを、2次元に拡張すれば、

$$P(y_1, y_2)dy_1dy_2 = p(x_1, x_2) \left| \frac{\partial(x_1, x_2)}{\partial(y_1, y_2)} \right| dy_1dy_2$$
(2.34)

となる。|...| は、xの y についてのヤコビアンである。 もし、P(y) がガウス分布を表す確立密度関数になれば、目的が達成される。 2次元のガウス分布は、

$$f(\boldsymbol{v}) = \frac{1}{\pi \langle v \rangle^2} \exp(-\frac{v_x^2 + v_y^2}{\langle v \rangle^2})$$
(2.35)

であるから、式(2.34)のヤコビアンが、

$$\left|\frac{\partial(x_1, x_2)}{\partial(y_1, y_2)}\right| = \left[\frac{1}{\sqrt{\pi}\langle v \rangle} \exp(-\frac{y_1^2}{\langle v \rangle^2})\right] \left[\frac{1}{\sqrt{\pi}\langle v \rangle} \exp(-\frac{y_2^2}{\langle v \rangle^2})\right]$$
(2.36)

となる変換関数があれば良い。 それは、

$$y_1 = V_x = \langle v \rangle \sqrt{-\ln(x_1)\cos(2\pi x_2)}$$
 (2.37)

$$y_2 = V_y = \langle v \rangle \sqrt{-\ln(x_1)\sin(2\pi x_2)}$$
 (2.38)

である。よって、2つの(0,1)の範囲の一様乱数 x_1 、 x_2 が求まれば、式(2.37,2.38)の変換により、1組の2次元ガウス分布が求まる。

第3章 ビームプラズマ不安定性に関する 基本課題

本シミュレーションスクールでは実習課題を用意している。本来なら、宇宙空間またはチャンバー装置内でしか再現できないようなプラズマ現象を計算機内でバーチャルに実現することができ、その解析を自ら行うことにより、シミュレーションの有用性について実感していただけると幸いである。粒子モデルには、ビーム-プラズマ不安定性や衝撃波現象に関する基本問題があるが、今回は、特にビームプラズマ不安定性に着目する。

近年、衛星を用いた地球磁気圏プラズマ波動観測により様々なプラズマ波動が観測され ている。これらのプラズマ波動の形成要因として考えられているもの1つにビームプラズ マ不安定性がある。ビームとは、背景プラズマに対して相対速度を持ったイオンや電子の 流れのことを示す。ビーム成分と背景プラズマとの相互作用は非常に古典的な問題であり、 理論的には1940年代から研究されてきている。その後、計算機シミュレーションにより その線形過程および初期の非線形過程について様々な研究が行われてきた。近年では、静 電孤立波の衛星観測に伴い、ビーム-プラズマ相互作用の非線形発展に関する計算機シミュ レーション研究が注目を浴びている。

このような研究背景の中、本基本課題では、特に不安定性の線形発展から非線形過程に いたるプロセスに着目し、背景プラズマにビームとなる成分を加えたモデルを用いて、計 算機シミュレーション実習を行う。実際のビームプラズマ不安定性では背景プラズマとビー ムの熱速度、質量比、密度比またはビームの相対速度などにその発展は大きく依存する。 しかし基本的には電子-電子間、電子-イオン間そしてイオン-イオン間での生じる不安定性 の組み合わによりある程度理解することができる。そこで今回、基本課題として、電子ま たはイオンのビームが1つだけ存在する場合に生じる電子-電子間、電子-イオン間そしてイ オン-イオン間の不安定性の特徴的な場合について計算機シミュレーションを行ってみる。

3.1 背景プラズマと電子ビーム

静止系において背景プラズマ(電子とイオン)が存在し、加えてビーム電子が存在する場 合について考える。この時起りうる不安定性としては、(1)背景電子-ビーム電子間の電子 二流体不安定性と(2)背景イオン-ビーム電子間のBuneman不安定性の2種類が考えられ る。どちらの不安定性が支配的になるかは、各成分の熱速度、質量比、密度比またビーム の相対速度などにより決まるが、本基本課題では背景電子の熱速度にのみ注目することに する。

Case (1):背景電子の温度が低い場合(電子二流体不安定性)



図 3.1: 電子及びイオンの速度分布とビーム不安定性

3.2. 背景プラズマとイオンビーム

背景電子の温度が低い場合、背景電子-ビーム電子間の不安定性である電子二流体不 安定性が支配的である。電子二流対不安定性とは電子-電子間で相対速度を持つ場合 に生じる静電的な不安定性で、各電子が自らの運動エネルギーを電場に与えより安定 した速度分布に落ち着こうとして生じる。背景電子の温度が低い場合の代表的な速 度分布を図に示す。図をみて分かるように背景電子の他に背景イオンも存在するが、 電子の方がイオンより軽く動きやすいためにイオンより先にビーム電子と電子二流 対不安定性を起し安定な状態に落ち着こうとする。課題としては、線形分散解析を予 め行いどのような波数の波動が最も成長しやすいかを確認後、その波動を再現でき るようなシミュレーションシステム・パラメータを選択し実際にシミュレーションを 行う。シミュレーション結果を解析して、最大成長率をもつ波数、時間的な波動成長 率などを理論値と比較する。

• Case (2): 背景電子の温度が高い場合 (Buneman 不安定性)

背景電子の温度が高い場合、電子二流体不安定性は小くなり背景イオン-ビーム電子 間のBuneman不安定性が支配的になる。Buneman不安定性とは電子-イオン間で相 対速度を持つ場合に生じる静電的な不安定性で、電流駆動型 (Current-driven)不安定 とも言われる。基本的にはイオン音波モードから静電波が成長するが、不安定性を起 すイオンと電子の温度がその波の成長に関係する。すなわち、イオン温度と電子温度 より決まるイオン音速がイオンの熱速度より小さい場合、イオンによるランダウ減 衰が起り波の成長は妨げられる。背景電子の温度が高い場合の代表的な速度分布を 図に示す。図をみて分かるように背景電子は熱速度が高く速度分布が広がっていて安 定しているので背景電子-ビーム電子間の不安定性は小さい、しかし、背景イオンは 熱速度が小さく、系全体から見た場合、背景イオンとビーム電子は不安定な状態にあ る。よって Buneman 不安定性が起こり安定な状態に落ち着こうとする。

この場合についても前のケース同様の解析を行い、シミュレーション結果の解析を 行う。

3.2 背景プラズマとイオンビーム

次に、静止系において背景プラズマとして電子とイオンが存在し、加えてビームイオンが存在する場合について考える。この時は上に述べた Buneman 不安定性と背景イオン-ビームイオン間のイオン二流体不安定性の2種類が考えられる。ここでも背景電子の熱速度にのみ注目して、それぞれの不安定性が支配的になる2つの場合について計算機シミュレーションを行ってみる。

• Case (3): 背景電子の温度が低い場合 (Buneman 不安定性)

背景電子の温度が低い場合、背景電子-ビームイオン間の Buneman 不安定性が支配 的である。この時の代表的な速度分布を図に示す。この場合、系全体で見た場合、背 景電子,イオンそしてビームイオンのすべてが不安定な状態にある。しかし電子はイ オンより軽く動きやすいためよ早く安定な状態に移行しやすい、よって、背景イオ ン-ビームイオン間の不安定性より背景電子-ビームイオン間の Buneman 不安定性の 方が早く起こり安定な状態に落ち着いてしまう。

この場合についても前のケース同様の解析を行い、シミュレーション結果の解析を 行う。

● Case (4): 背景電子の温度が高い場合 (イオン二流体不安定性)

次に背景電子の温度が高い場合は、Buneman 不安定性が弱まり背景イオン-ビームイ オン間の不安定性であるイオン二流体不安定性が支配的となる。イオン二流体不安 定性とは、イオン-イオン間で相対速度を持つ場合に生じる静電的な不安定性で、各 イオンが自らの運動エネルギーを電場に与えより安定した速度分布に落ち着こうと して起る不安定性である。この場合の代表的な速度分布を図に示す。見て分かるよう に背景電子は熱速度が大きく速度分布が広がっており、系全体見た場合ビームイオン と背景電子の不安定性は小さい。よって背景イオンとビームイオン間のイオン二流体 不安定性が一番強くなる。イオン二流体不安定性は他の2つの場合と違い両方イオン なので最も成長率が小さく、波の成長に時間がかかり長い計算時間を要する。

以上の基本課題は、非常に古典的な問題であるにもかかわらず、それぞれの現象の時空 間発展はプラズマパラメータに大きく依存する。このため、観測、理論、シミュレーション を含めて現在でも非常にホットで最先端の研究テーマでもある。また、限られたパラメー タ範囲ではあるが、オンライン上でもWebベースの基本課題シミュレーションが実行でき る。http://www-netlab.kurasc.kyoto-u.ac.jp をご覧頂きたい。ID が必要な場合は連絡をい ただければ幸いである。

関連図書

- Birdsall, C. K, and A. B. Langdon, *Plasma physics via computer simulation*, McCgraw-Hill, 1985.
- [2] Hockney, R. W., and J. W. Eastwood, Computer simulation using particles, McGraw-Hill, 1981.
- [3] Matsumoto, H. and Y. Omura, Particle Simulations of Electromagnetic Waves and its Applications to Space Plasmas, *Computer Simulations of Space Plasmas*, ed. by H. Matsumoto and T. Sato, Terra Pub. and Reidel Co., pages 43-102, 1984.
- [4] Omura, Y. and H. Matsumoto, KEMPO1: Technical Guide to One-Dimensional Electromagnetic Particle Code, Computer Space Plasma Physics: Simulation Techniques and Softwares, ed. by H. Matsumoto and Y. Omura, pages 21-65, Terra Scientific, Tokyo, 1993. (http://www.terrapub.co.jp/e-library/cspp/index.html)
- [5] Villasenor, J. and O. Buneman, Rigorous charge conservation for local electromagnetic field solvers, *Computer Physics Communication*, 69, 306-316, 1992.
- [6] William H. Press, Saul A. Teukolsky, William T. Vetterling, Brian P. Flannery Numerical Recipes in C, 技術評論社、1998年
 - 粒子モデルの基本実習課題に関連する参考図書も以下にあげる。
 - 1. Nicholson, D. R., Introduction to plasma theory, John Wiley & Sons, 1983.
 - Melrose, D. B., Instabilities in space and laboratory plasmas, Cambridge Univ. Press, 1986.

プラズマ線形分散関係ソルバー emdisp について

羽田 亨⁽¹⁾、白石岳雄⁽²⁾、関 光弘⁽²⁾

1:九州大学大学院総合理工学研究院流体環境理工学専攻 2:九州大学大学院総合理工学府大気海洋環境システム学専攻

August 10, 2002

要旨:無衝突プラズマ中のプラズマ波動の線形分散関係を数値計算するプログラム emdisp の使 用方法および計算の背景となる理論の概要について紹介する。このプログラムでは、プラズマに 関する各種パラメータ(分布関数、イオン種数など)と波数ベクトル(の組)を入力することに より、対話型操作を通して、複素角周波数(実角周波数および線形成長率) 波動の偏波指数、波 動が電磁的/静電的かを表す指数、などの出力を得ることができる。プラズマ分布関数は、ドリフ ト速度を持ち得るマックスウェル分布の重ね合わせとして表現する。操作の実例を、いくつかの 例題を用いて解説する。なお、ここで紹介するプログラムは、「宇宙シミュレーション・ネットラ ボラトリーシステム開発」(科学技術振興事業団)の一部として整備が行われているものの簡約化 されたバージョンである。

1 はじめに

宇宙・天体で起きている様々な興味深い現象を議論する際、これらの環境を満たす媒質である 無衝突プラズマ中に、どの様な性質をもつどのような線形波動が存在し得るかを知る、つまり線 形分散関係を得る、ことは極めて重要な意味を持つ。これは、議論の対象が宇宙・天体プラズマ 波動である場合には勿論のこと、それ以外の場合でも、例えば衝撃波、磁気再結合過程、といっ た一見、線形波動とは関係のなさそうなダイナミックな物理現象においても、その核心的部分で (異常)散逸をまかなっているのがプラズマ波動を媒介とした効果的な粒子拡散であるように、数 多くの宇宙・天体現象において、プラズマ波動が本質的な役割を果たしているからである。さら に一般的な立場から見ても、ある物理系の性質を理解するためには、まずその定常状態を議論し、 その上で系の微少擾乱に対する反応(線形応答)を求めるのが筋道であるが、これは媒質中の線 形波動を求めることに他ならない。

この様に無衝突プラズマ中の線形分散関係を計算する機会は、宇宙・天体プラズマの研究に携 わっている限り、かなり頻繁にあるものと思う。しかしその一方、他の数値計算コードの場合と同

1

様、線形分散関係を計算するプログラム(線形分散関係ソルバー)の汎用バージョンは、少なくと も筆者達の知る限りにおいては、手軽に使える形では世の中に出回っていないようである(実際、 過去数年の間に筆者らが複数の研究グループから、我々の使用している線形分散関係ソルバーを 使いたいとの依頼があった)。大きな研究組織では、計算機ライブラリーの一部としてプログラム が整備されているところも多くあるのだろうが、宇宙・天体研究者の多くがそうであるように、ス タッフ数が数人程度の小さな研究グループでは、むしろ小回りのきくプラグラムの方が使い勝手 がよいであろう。本稿で紹介するプログラム emdisp は、分散関係ソルバーのなかでは必要最小 限の機能のみを備える基本的なものだが、今回のワークショップのように、とりあえず数値シミュ レーションの結果と照合したい、というような目的には、十分役にたつはずである。また、必要 に応じて新しい問題に適応できるよう、プログラムに修正・変更を逐次加えていくことも可能であ る(利用者からのフィードバックを期待したい)。以下、必要最低限の計算の背景の説明に続き、 与えられた条件のプラズマの中に存在する波動の線形分散関係を、具体的な題材を例にとって説 明する。

2 計算の背景

ここでは、各種物理量の定義と物理的意味を明確にすることも兼ねて、emdispを使用する際に 必要と思われる最低限の理論的なバックグラウンドとして、線形分散関係式を求める過程をごく 簡単に振り返っておく。詳細は多くのプラズマ物理の教科書に解説してある(例えば[1],[2],[3]) のでこれらを参照してほしい。

2.1 線形分散関係の導出

プラズマ波動の線形分散関係式を導くために使用する式は、ヴラソフ方程式

$$\frac{\partial f_{\sigma}}{\partial t} + \mathbf{v} \cdot \nabla f_{\sigma} + \frac{q_{\sigma}}{m_{\sigma}} (\mathbf{E} + \frac{1}{c} \mathbf{v} \times \mathbf{B}) \cdot \frac{\partial f_{\sigma}}{\partial \mathbf{v}} = 0$$
(1)

およびマックスウェルの方程式

$$\frac{1}{c}\frac{\partial \mathbf{E}}{\partial t} + \frac{4\pi \mathbf{j}}{c} = \nabla \times \mathbf{B} \quad , \quad \nabla \cdot \mathbf{E} = 4\pi\rho \quad , \tag{2a, 2b}$$

$$\frac{1}{c}\frac{\partial \mathbf{B}}{\partial t} = -\nabla \times \mathbf{E} \quad , \quad \nabla \cdot \mathbf{B} = 0 \tag{3a, 3b}$$

である。ここに、 $f_{\sigma}(\mathbf{r}, \mathbf{v}, t)$ はプラズマの分布関数、 σ は粒子種、その他の記号は標準的なものである。単位系はガウス単位系を用いている。

3次元空間内に一様にひろがるプラズマを考え(0次量)ここに微少擾乱(1次量)を与える。 2次以降のオーダー量は無視できると仮定し(線形近似) さらに1次の擾乱を、空間・時間に関 してフーリエ変換する ($\partial/\partial t \rightarrow -i\omega$ 、 $\nabla \rightarrow i\mathbf{k}$)。ここに、 $\omega = \omega_r + i\gamma$ は波動の複素角周波数、 ω_r は実角周波数、 γ は線形成長率(正が不安定)、k は波数ベクトルである。

これらの操作により、マックスウェルの方程式 (2a) と (3a) は、屈折率ベクトル $\mathbf{n} = c\mathbf{k}/\omega$ を用いて

$$\mathbf{n} \times (\mathbf{n} \times \mathbf{E}_1) + \mathbf{E}_1 = -\frac{4\pi i}{\omega} \mathbf{j}_1 \tag{4}$$

と書ける。ただし磁場の摂動は、 $\mathbf{B}_1 = \mathbf{n} \times \mathbf{E}_1$ によって電場の摂動と関係づけられている。式(4) の右辺、つまり電流が、電場 \mathbf{E}_1 の関数として書ければ式は閉じ、分散関係式が得られる。((4)を 導くにあたって (2b),(3b)を使わなくてすんだのは、 $\omega \neq 0$ の波動を考える場合には、(2a)、(3a) と電荷保存の式が(2b)と(3b)を含んでいるからである。 $\omega = 0$ の波動(構造)を考える場合には 注意が必要である(例えばエントロピー波))

一方、ヴラソフ方程式を線形化し、分布関数の1次の摂動を粒子の軌道に沿った積分としてあ らわすと

$$f_{\sigma 1} = \frac{-q_{\sigma}}{m_{\sigma}} \int_{-\infty}^{t} (\mathbf{E}_{1} + \frac{1}{\omega} \mathbf{v} \times (\mathbf{k} \times \mathbf{E}_{1})) \cdot \frac{\partial f_{\sigma 0}}{\partial \mathbf{v}} dt$$
(5)

となるが、この積分を実行するためには、ゼロ次オーダーでの粒子の軌道を指定する必要がある。 一様・定常な磁場のもとでは、粒子はスパイラル軌道をとるから、これを用いて実際に(5)を評 価することができる(ランダウの積分)。それ以外の軌道の場合にでも f_{o1} が計算できる場合があ る。また数値的には、一般的なゼロ次オーダー軌道に対して(5)の積分が計算でき、例えばニュー トラルシート近傍でのテアリング不安定性の解析に応用されている)。

分布関数の1次摂動がわかれば、電流の1次摂動は

$$\mathbf{j}_1 = \sum_{\sigma} q_{\sigma} \int f_{\sigma 1} d\mathbf{v} \equiv \sigma \cdot \mathbf{E}_1 \tag{6}$$

により求められる。ここに σ は電気伝導度テンソルである。線形波動に対するプラズマの効果は、 すべてこの σ の中に集約されている。実際、(6)を(フーリエ空間でなく)実空間で書けば、与 えられた摂動電場に対する、非局所的かつ時間の遅れをともなうプラズマの線形応答が摂動電流 を作ることを示す畳み込み積分となり、直感的である。プラズマの応答(6)を用いて(4)を書き直 せば、

$$\mathbf{D} \cdot \mathbf{E}_1 \equiv \mathbf{n} \times (\mathbf{n} \times \mathbf{E}_1) + (1 + \frac{4\pi i}{\omega}\sigma)\mathbf{E}_1 = 0$$
(7)

となり、これが E1 に関して自明でない解を持つ条件、

$$\det(\mathbf{D}) \equiv D(\omega, \mathbf{k}) = 0 \tag{8}$$

が線形分散関係である。ここに、D は線形分散関係テンソルであり、また (7) にあらわれる $\epsilon \equiv 1 + 4\pi i \sigma / \omega$ を誘電率テンソルと呼ぶ。

2.2 0次の分布関数

線形分散関係を解くためには、プラズマの0次の分布関数を指定する必要がある。人工衛星に よって観測された分布関数や、数値実験により得られた分布関数は、任意の形をしているが、こ れを(5)の積分が評価しやすい関数の重ね合わせとして表わすのが、分布関数の標準的な表現方 法である。emdispでは分布関数を、ドリフトと温度異方性を持つマックスウェル分布の重ね合わ せとして書く。

$$f_{\sigma 0}(v_{\parallel}, v_{\perp}) = \sum_{j} f_{\sigma 0, j} = \sum_{j} \frac{n_{\sigma j}}{(2\pi)^{3/2} a_{\sigma \parallel} a_{\sigma \perp}^2} \exp(-\frac{(v_{\parallel} - u_{\sigma})^2}{2a_{\sigma \parallel}^2} - \frac{v_{\perp}^2}{2a_{\sigma \perp}^2})$$
(9a)

ここに、 $n_{\sigma j}$ は密度、 $a_{\sigma \parallel}^2 = T_{\parallel}/m_{\sigma} \ge a_{\sigma \perp}^2 = T_{\perp}/m_{\sigma}$ は磁力線に平行および垂直方向の熱速度 の2乗、 $u_{\sigma j}$ は沿磁力線方向のドリフト速度である。マックスウェル分布の線形重ね合わせとして 0次分布関数を表現することの利点は、性質がよくわかっていて数値計算の比較的容易な「プラ ズマ分散関数」を用いて線形分散関係を評価することができるからである。

上式の温度が低い場合の極限として、冷たいプラズマの分布関数、

$$f_{\sigma 0,j} = \frac{n_{\sigma j}}{\pi v_{\perp}} \delta(v_{\parallel} - u_{\sigma}) \delta(v_{\perp})$$
(9b)

および、冷たいリング分布、

$$f_{\sigma 0,j} = \frac{n_{\sigma j}}{\pi v_{\perp}} \delta(v_{\parallel} - u_{\sigma \parallel}) \delta(v_{\perp} - u_{\sigma \perp}))$$
(9c)

も扱うことができる(但し、夏の学校バージョンではサポートしていない)。

2.3 計算出力

入力パラメータを与え、与えられた波数ベクトルkに対して線形分散関係(8)を解けば、解と してωが得られ、実周波数および線形成長率が求められる。一方、ひとたびωが求められると、 (7)の固有ベクトルとして、E₁(の方向)が定まるが、これを用いて波動の固有の性質をあらわ す物理量を計算することができる。

いま、一般性を失うことなく、磁場の 0 次成分が z軸に平行であり、k が x - z 平面に含まれる ように座標系をとり、求められた E_1 から以下の量を評価する。

- **(** \mathbf{B} (Polarization) : $P = iE_{1,x}/E_{1,y}$
- 電磁静電指数 (EM/ES index) : $S = |\mathbf{k} \cdot \mathbf{E}_1 / \mathbf{k} \times \mathbf{E}_1|$

偏波はその定義からわかるように、E1の回転の向きを表している。磁力線のまわりを、電子と 同じ向きにまわるか、陽子と同じ向きにまわるか、をそれぞれ「右回り」および「左回り」と定 義する。ここで「まわる」と言っているのは、ある空間上の点で観測したベクトルが時間的にど のように回転するか、のことである。(一方、例えば真空中の光の伝搬に対しては、電子も陽子も 関与しないから磁力線の方向に対して定義するのは無意味であるため、波動の伝搬方向に対する 電場の回転方向(波動が進むにつれて右ネジの方向に電場が回転するか、あるいは左ネジの方向 か)で定義することが普通である)。

また、電磁静電指数を、上の様に定義した。定義の分子および分母をそれぞれゼロとした場合 が、純粋な電磁波(k \perp E₁)および純粋な静電波(k \parallel E₁)に対応している。したがって、波動 が静電的(例えば音波、プラズマ振動)なほど S は 1 を越えて大きな値をとり、波動が電磁的で あるほど(例えば平行伝搬アルフヴェン波)S は 0 に近い値をとる。なお静電波の場合、磁場摂動 は B₁ = n × E₁ = 0 であり、また線形分散関係式 (7) は、 $\epsilon \cdot$ E₁ = 0 と書ける。さらに k \parallel E₁ を もちいてこれを書き直すと静電波の線形分散関係、 $\epsilon_{es}(\omega, \mathbf{k}) \equiv \mathbf{k} \cdot \epsilon \cdot \mathbf{k} = 0$ が得られる。このスカ ラー方程式は (7) の一部分であり、したがって emdisp にも含まれている。

3 計算の手順

3.1 計算手順の概略

与えられたプラズマ条件のもとでの線形分散関係を求めるためには、以下の手続きを踏む。 Step 1. プラズマのパラメータ指定

プラズマに関する情報を粒子種ごとに、分布関数をマックスウェル分布 (9a) で表現する際のパ ラメータの組(密度、ドリフト速度など)として与え、また磁場の強さをプラズマ周波数とサイ クロトロン周波数の比として与える。これにより、線形分散関係 (8) の具体的な表式が決まる。 Step 2. シード解の探索

与えられた波数ベクトル k に対して、これを満たす ω を見つけることが目標であるが、線形分 散関係 (8) は ω に関して著しく非線形なので、解を見つけるための一般的な方法は存在しない。 本プログラムでは、期待される解が存在する領域を指定し、D(ω, k)の「地図」を描く、あるいは この領域内でのトライアル・アンド・エラーによる方法により、まずシード解(波数の組に対して 解を求めていくための元になる解)を見つける。

Step 3. 分散関係を求める

シード解が見つかれば、それを起点として外挿を繰り返すことにより、 *ω* と k の組を比較的効率よく求めていくことができる。この際、偏波、電磁静電指数、計算誤差も評価する。

以下、実際に emdisp を走らせながら、各手続きを解説する。

3.2 Step 1. プラズマのパラメータ指定

プラズマのパラメータは、データセットを作成し、その中で namelist 文を用いて指定する。例 えば、データセット data1 を以下のように作る。

```
cat data1
&input
```

```
ns = 3,
pc = 1.0000d+04,
ds = 0.9900d-00, 1.0000d-00, 0.0100d-00,
el = 1.0000d+00, -1.0000d+00, 1.0000d-00,
rm = 1.8360d+03, 1.0000d+00, 1.8360d+03,
aba = 0.2200d+00, 0.5400d+00, 0.2200d+00,
abe = 0.2200d+00, 0.5400d+00, 0.2200d+00,
ud = 0.0000d+00, 0.0000d+00, 1.0000d+01,
```

&end

ここでは、電子、プロトン、およびプロトンのビームから構成されるプラズマを考えている。使われているパラメータは、以下の通りである。

変数	:	意味	規格化
ns	:	マックスウェル分布の数	
pc	:	規格化定数	
ds(i)	:	数密度	ds(i)の総和=1
el(i)	:	電荷	素電荷 = 1
rm(i)	:	質量	電子 = 1
aba(i)	:	平行熱速度	c / sqrt(pc)
abe(i)	:	垂直熱速度	c / sqrt(pc)
ud(i)	:	ドリフト速度	c / sqrt(pc)

上の例では、プラズマは3つのマックスウェル分布の重ね合わせ(プロトン、電子、プロトン ビーム)として表現しているので、ns=3である。nsは20以下とする。分布数nsと、規格化定数 pc(詳細は後述)のあとにならんでいるのが、各マックスウェル分布(添字iで表すことにする) を特徴づけるパラメータ群である。数密度ds(i)は、iについての総和が1となるように指定す る。電荷el(i)は、例えば陽子ならば1、電子ならば-1である。これらの2つに対して、プラズ マの中性条件、つまりds(i)*el(i)の総和が0となるようにする。但し、パラメータ入力時の 数値誤差のため、この値が厳密に0になるとは限らないので、プログラム内部で、最も数密度の 大きい粒子種の数密度を調整することで、中性条件を満たすようにしている。質量 rm(i) は、電 子を1とし、(現実の)陽子ならば1836、あるいは計算機実験で使った値を指定すればよい。平 行熱速度 aba(i)、垂直熱速度 abe(i)、およびドリフト速度 ud(i) は、(9a) で定義している。

*変数の規格化について

線形分散関係の計算に限らず、計算はなるべく無次元量を用いて行った方が効率も、見通しもよい。しかし、プログラムにデータを渡す際に、少なくとも周波数と速度(つまり時間と空間)をどのように規格化しているか、を指定する必要がある。例えば、時間は必ず電子のサイクロトロン 角周波数、速度は光速を用いる、と決めておけば簡単であるが、実際にはプロトンの時間スケー ルやMHD波動の速度スケールなど、指定するパラメータの値が余りに大きくなったり小さくなっ たりすることを避けるために、emdispでは以下のように変数の規格化を行っている。

・時間は、データを入力する際の、マックスウェル分布の初めの列の粒子種のサイクロトロン角 周波数を基準として、これを用いて規格化する。以下、これをΩ₀とする。

・空間は、考えている問題に都合のよい任意の長さを基準とする。これを L₀ とする。

・上の2つにより、速度スケールは、 $L_0\Omega_0$ で規格化される。

・規格化定数 PC = $c^2/L_0^2\Omega_0^2$ と指定する。

例えば、 Ω_0 としてプロトンのサイクロトロン角周波数 Ω_i を選び、 L_0 として V_A/Ω_i (V_A はア ルフヴェン速度)を選ぶと、PC の値は

$$\frac{c^2}{V_A^2} = \frac{\omega_i^2}{\Omega_i^2} (1+\mu) \sim \frac{\omega_i^2}{\Omega_i^2} \tag{10}$$

となり (ω_i はプロトンプラズマ角周波数、 $\mu = m_e/m_i$ はプロトンと電子の質量比) また、 Ω_0 として電子のサイクロトロン角周波数 Ω_e 、 L_0 として電子の慣性長 c/ω_e を選ぶと、PC の値は

$$\frac{c^2}{L_0^2 \Omega_e^2} = \frac{\omega_e^2}{\Omega_e^2} \tag{11}$$

となる。実際には、考えている現象がMHDスケールに近いか、電子スケールに近いかを見極めて、上の2つのどちらかを選んでおけば良い。

上の入力データを用いて、実際にプログラムを走らせてみる。端末から、

./emdi817.exe

と入力すると(プログラム名は変更になっている可能性あり)

Step 1: Input plasma parameters

Input the dataset name

と聞いてくるので、先ほどのデータセット名

data1

を入力する。すると、以下の表が表示されるはずである。

new ds(1) = 0.990000D+00

of species 3

pc 0.10000D+05

species #	1	2	3
density	0.990000D+00	0.100000D+01	0.10000D-01
el charge	0.10000D+01	-0.100000D+01	0.10000D+01
mass	0.183600D+04	0.100000D+01	0.183600D+04
para thml vel	0.333300D+00	0.222600D+02	0.331600D+01
perp thml vel	0.333300D+00	0.222600D+02	0.331600D+01
drift vel	0.00000D+00	0.00000D+00	0.10000D+02

cycl freq	0.10000D+01	-0.183600D+04	0.10000D+01
plsm freq **2	0.10000D+05	0.185455D+08	0.101010D+03
para beta	0.222178D+00	0.545221D+00	0.222139D+00
perp beta	0.222178D+00	0.545221D+00	0.222139D+00
larmor radius	0.333300D+00	0.121242D-01	0.331600D+01

一番上の行は、中性条件に合うように、数密度の調整を(表示されていない桁に対して)行った 結果である。以下、入力されたパラメータのエコーが続き、最後の4行は、サイクロトロン角周 波数、プラズマ角周波数、平行、垂直方向のベータ値、およびラーマ半径である。

3.3 Step 2. シード解の探索

前節までで、分散方程式(8)の具体的な関数が定まったので、次にこれを、与えられた k に対し て解かなければならない。前に述べたように、まずシード 解をもとめ、これをもとにして外挿に より(k,ω)の組を計算していく方法をとる。 分散方程式(8)は、実際には次の2つの式からなる連立方程式である。

$$D_r(\omega, \mathbf{k}) = 0 \quad ; \quad D_i(\omega, \mathbf{k}) = 0 \tag{12a, b}$$

ここに、 D_r 、 D_i はそれぞれ D の実数部、虚数部である。解を確実に探すためには、 ω 複素平面 上で、 $D_r = 0$ の曲線と $D_i = 0$ の曲線を描き、これらの交点を求めればよい。同じことであるが、 D_r と D_i のそれぞれの正負に応じて、 ω 平面を4通りに「色分け」し、4色が集まるところを探 してもよい。関数 D が ω に関して正則であること(例えば $|\omega|$ は D に含まれない)等の性質を うまく利用して、シード解を効率よく見つけることができる。(夏の学校バージョンでは、この機 能は省略している)

しかし、高速の計算機が身近に使える今日では、上の方法よりもむしろトライアル・アンド・エ ラー(もぐらたたき)によって、力まかせでシード解を探すほうが効率がよいようである。これ は、解があると推測される ω の領域内のランダムに選んだ点を起点として、逐次近似(ニュート ン法など)により、点を移動させ、解に収束させる、というものである。多くの場合、点は領域 内にとどまらず、遠くに発散してしまうが、シード解に収束する場合もある。以下、例の続きを 示す。

Step 2: find seed root

Input wkk(=wave number) and wth(=prop angle)

.1,.1

シード解を見つける際の、波数ベクトルを入力している。wkk は波数の絶対値 ($|\mathbf{k}| = k$)、wth は k と定磁場とのなす角度 (θ) で、単位は度である。

Specify the frequency domain for the seed roots.

Input real(01), real(02), imag(01), imag(02)

0,1,-.02,.1

シード解を探す $\omega = \omega_r + i\gamma$ の領域を、real(o1) < ω_r < real(o2)、imag(o1) < γ < imag(o2) と指定している。

注意: imag(o1)、imag(o2)を負で、とても小さい値にすると、強減衰モードが多数現れ、計算が困難になることがある(プラズマ粒子の数だけ波動モードの自由度もある!)。

多少の時間経過の後、シード解が表示される。

... now searching the roots.

```
10 root(s) are found
```

rt#	real w	imag w	residue	iter	itry
1	-0.200970D+00	-0.880320D-01	0.69349D-14	83	121
2	-0.126941D+00	-0.232688D-08	0.78023D-14	33	4
3	-0.832901D-01	-0.175863D-01	0.56708D-13	21	2
4	0.832902D-01	-0.175856D-01	0.13754D-13	23	1
5	0.116475D+00	-0.800347D-01	0.26313D-11	26	36
6	0.118361D+00	0.105746D-07	0.46397D-14	37	5
7	0.125838D+00	0.111404D+00	0.22495D-14	31	3
8	0.264957D+00	-0.247404D+00	0.10834D-11	68	27
9	0.544976D+00	-0.444636D+00	0.13168D-09	72	28
10	0.846275D+00	-0.131343D+00	0.75697D-11	34	26

3.4 Step 3. 分散関係を求める

上に引き続いて、正の成長率を持つ7番目の根について、分散関係を求めることにしよう。

Step 3: computation mode ?

1: vary k, fix theta / 2: fix k, var theta

```
0: back to previous step / 999: end
```

```
1
```

key in the root number for tracing

```
7
```

Input wkk1, wkk2, nkk

0.01,2,50

emdispでは、分散関係としては、(1) θ を固定しておき、kを変化させる、(2) kを固定しておき、 θ を変化させる、の2つの計算をするようになっている。上の例では、まず(1)を選択する。 次に、先のステップで求めたシード解の表の中の、どの解について計算を行うか、を指定する。 今の場合 7 を入力する。

最後に、変化させる k の範囲(wkk1< k < wkk2)と、計算する回数(nkk)を指定する。 すると、結果が出力される(印刷できるように有効数字をおとして表示している)。

1 0.0100 0.1000 -0.0314 -0.0000 0.0000 -1.0000 0.0012 17 2 0.0506 0.1000 0.0168 0.0903 0.0000 1.0000 0.0012 13 3 0.0912 0.1000 0.1066 0.1154 0.0000 1.0000 0.0012 5 0.1318 0.1000 0.1842 0.0883 0.0000 1.0000 0.0012 4 7 5 0.1724 0.1000 0.2391 0.0588 0.0000 1.0000 0.0012 9 6 0.2131 0.1000 0.2850 0.0386 0.0000 1.0000 0.0012 7 7 0.2537 0.1000 0.3303 0.0263 0.0000 1.0000 0.0012 7 (以下省略)

左から順に、番号、k、 θ 、 ω_r 、 γ 、誤差 (|D|)、偏波指数(2.3 章参照)、電磁静電指数(2.3 章参 照)、イテレーション数、である。計算結果は、データセット'vary-k.txt' に保存される。 次に、伝搬角 θ に対する依存性を調べよう。

Step 3: computation mode ?

1: vary k, fix theta / 2: fix k, var theta

0: back to previous step / 999: end

2

key in the root number for tracing

```
7
```

Input wth1, wth2, nth

.1,60,50

1	0.1000	0.1000	0.1258	0.1114	0.0000	1.0000	0.0012	0
2	0.1000	1.3222	0.1258	0.1114	0.0000	0.9995	0.0164	3
3	0.1000	2.5456	0.1256	0.1114	0.0000	0.9980	0.0316	3
4	0.1000	3.7671	0.1254	0.1114	0.0000	0.9957	0.0469	3
5	0.1000	4.9900	0.1251	0.1114	0.0000	0.9925	0.0621	3
6	0.1000	6.2125	0.1247	0.1114	0.0000	0.9883	0.0774	3
7	0.1000	7.4358	0.1242	0.1114	0.0000	0.9833	0.0928	3

今度は、(2)を選び、同じ解 (=7)を選択し、そして伝搬角の範囲(wth1< θ <wth2)と計算数(nth)を指定すると、先ほどと同様に結果の表が出力され、また計算結果はデータセット'vary-th.txt'に保存される。



図 1: 計算結果のグラフ。上段はkに対する依存性($\theta = 0.1$)、下段は θ に対する依存性(k = 0.1)、 また左側と右側にそれぞれ複素角周波数、偏波指数と電磁静電指数をプロットしている。

4 終わりに:WEB公開版との相違

ここで紹介したプログラムは、「宇宙シミュレーション・ネットラボラトリーシステム開発」(科学技術振興事業団)の一部として WEB 上での公開を目指して整備が行われているものを、今回の宇宙・天体プラズマ夏の学校での使用のために準備したバージョンである。WEB 公開バージョンには、ここで紹介した機能の他に以下の項目を追加する予定である。

・マックスウェル分布以外に、冷たいプラズマ、リング・プラズマなどの分布関数の指定。

・トライアル・アンド・エラーによる方法に加え、複素角周波数平面でのマップを描いて効率よ くシード解を見つけるルーチン。

- ・WEBページ上での入出力。
- ・特に、WEB上で、結果のグラフ出力。

その他、機能の強化、使い勝手の向上をはかっていきたいと考えている。ご意見、ご要望、そ の他何でも感じたことがあれば、ご連絡いただきたい。

参考文献

- Ichimaru, S., Basic Principles of Plasma Physics: A Statistical Approach, Benjamin/Cummings Pub. Co., 1973.
- [2] Akhiezer, A. I., I. A. Akhiezer, R. V. Polovin, A. G. Sitenko, and K. N. Stepanov, Plasma Electrodynamics, Vol. I. Linear Theory, Pergamon Press, 1975.
- [3] Miyamoto, K., Plasma physics for nuclear fusion, The M.I.T. Press, 1976.

プラズマ粒子シミュレーションで使用する境界条件

国立極地研究所 岡田雅樹

プラズマのシミュレーションは一般に、偏微分方程式の初期値境界値問題を数値的に解く ことに相当する。物理的に正しいシミュレーションを行うことができるどうかは、これら の初期値と境界条件によって決まるといっても過言ではない。本章では、プラズマ粒子シ ミュレーションにおいて使用される境界条件とその物理的意味について述べる。

1. 概要

シミュレーションでは、無限の物理領域の一部を計算機上に再現するためになんらかの仮 定を必要とする。例えば、一様無限の空間内に周期性のみを持つ物理現象を再現する場合 には、周期境界条件を使用することができる。あるいは、空間内に局所的に擾乱の発生領 域が存在し、その影響が空間的に拡散または放射されるような現象をシミュレートする場 合には、擾乱の発生領域を取り囲むように擾乱の吸収領域を設ける方法が有効である。ま た、宇宙飛翔体等をシミュレーション空間内にモデル化するためには、導体または誘電体 をシステムないに置く必要がある。本節では、これらの境界条件を与える手法について詳 述する。

2. 周期境界条件

ー様無限の空間を再現する手法として最も一般的な手法が、周期境界条件である。システム長Lの1次元電磁粒子コードの場合、全ての物理量F(x)に対して、F(x=L)=F(0)が成り立つことを意味する。これは、左右のシステム境界を越えて伝播する波動は、反対側のシステム境界からシステム内に伝播することを意味している。このことを電位 にあてはめると、システムの両端における電位 は同じでなければならず、ガウスの発散定理からシステム内の全電荷は零でなければならないことになる。さらにシミュレーションの全ての時間ステップにおいてこの条件が満たされるために、ひとつの境界を越えたプラズマ粒子は、そのまま速度を変えることなく反対側の境界から注入されなければならない。このような周期境界条件を与えることによって、実空間においてシステム長Lを周期とする現象をシミュレーション空間に再現することが可能になる。この周期性によってポアソン方程式の解法として高速なFFT法が有効である。

3.拘束プラズマ境界条件

実験室プラズマ等の有限空間に閉じ込められたプラズマ環境を再現するためには、 F(x=L)≠F(X=0)である拘束プラズマ境界条件を使用する必要がある。ここで紹介する拘束 プラズマ境界条件は、シミュレーション空間以外の領域はプラズマの存在しない自由空間 である状況に似ている。つまり、放射電磁波が境界に達するとシステムには戻らず、シミ ュレーション空間から取り除かれ、外部からレーザー等の電磁波を入射することも可能と なる。シミュレーション空間外(x<0, L<x)の領域では静電波は存在しないため、Ex(x)=0 と なる。このようなシステムでポアソン方程式を解くためには FFT 法をそのまま使用するこ とはできないが、少々工夫をすることで FFT 法と同程度の計算量でポアソン方程式を解く ことができる。非周期的なシステムに対して FFT 法を適用するためには、電荷密度 が、 $\rho(x=L) = \rho(x=0)$ を仮定する必要がある。非周期的でかつ電荷中性なシステムは

$$\rho(x = L) = 0 = \rho(x = 0)$$
 (1)

を仮定することによって実現できる。この仮定によって、ポアソン方程式

$$\frac{\partial^2 \phi}{\partial x^2} = -\rho(x) \tag{2}$$

は周期的な特解 ϕ_{p} を持つことができる。この特解 ϕ_{p} に $\phi = a + bx$ なる解を加えることによって、非周期的な解をえることができる。ここで、 b は境界条件から求められ

$$Ex(0) = -\frac{\partial \phi}{\partial x}\Big|_{0} = -\frac{\partial \phi_{p}}{\partial x}\Big|_{0} - b = -\frac{\partial \phi_{p}}{\partial x}\Big|_{L} - b = Ex(L) = 0$$
(3)

さらに中心差分の原理と周期性 p(x+L)= p(x)を利用すると

$$b = -\frac{\partial \phi_p}{\partial x}\Big|_0 = \frac{\phi_p(-\Delta x) - \phi_p(+\Delta x)}{2\Delta x} = \frac{\phi_p(L - \Delta x) - \phi_p(+\Delta x)}{2\Delta x}$$
(4)

$$\phi(X_j) = \phi_P(X_j) + bX_j \qquad \text{for} \qquad 0 \le X_j \le L \tag{5}$$

$$E_x(X_j) = \frac{\phi(X_j - \Delta x) - \phi(X_j + \Delta x)}{2\Delta x} \quad \text{for } \Delta x \le X_j \le L - \Delta x \tag{6}$$

このような条件 $\rho(x = L) = 0 = \rho(x = 0)$ を実現するためには、システムの両端のグリッド に荷電粒子が入らないようにする必要がある。システム両端における粒子の取り扱いとし ては、 $x = \Delta x \ge x = L - \Delta x$ において粒子を反射する壁があるように取り扱うことによって 実現できる。



図1.拘束プラズマ境界条件を扱う場合の境界近傍での粒子の取り扱い

4 . 完全導体境界条件

完全導体表面では、電界ベクトルの接線成分がゼロになるため、 x = 0(j = 0) と $x = L_x(j = N_x)$ における境界条件は次のように書くことができる。ここで、j および k は x,y 方向のグリッドの添え字である。

$$E_y = 0$$
 $E_{y,0,k+1/2} = 0$ (7)

$$E_z = 0 \qquad \frac{E_{z,-1/2, k+1/2} + E_{z,1/2, k+1/2}}{2} = 0 \qquad (8)$$

Ez 成分の境界条件は(8)式のような平均操作を行うことにより2次の精度を持つことができる。境界面上において、Ey, Ez 成分が常にゼロであることをマックスウェル方程式の磁界成分の時間変化の式に適用すると

$$\frac{\partial B}{\partial t} = -c\nabla \times E \tag{9}$$

$$Bx \qquad \frac{B_{x,-1/2,k} + B_{x,1/2,k}}{2} = B_{x0}(y) \qquad (1\ 0\)$$

が得られる。この条件は方程式を閉じるためには必要ではないが、導体境界面上において 初期値として与えた磁場成分が後の時間ステップにおいて保存されるということを示して いるという意味で重要である。



図2.導体境界条件を使用する場合の境界(x=0)近傍のグリッド配置

6. 開放境界条件

無限に広がる空間をシミュレーション空間に再現する方法として必要な境界条件が開放境 界条件である。この条件でもっとも重要なのは、実際にシミュレーション空間の外に格子 点が存在しないにもかかわらず、あたかも無限遠まで続いているかのように"みせかける" 必要があるという点にある。つまり、シミュレーション空間内で発生した静電的、電磁的 波動は外部に向かって伝播するが、これらの波動はシミュレーション空間外の無限遠に伝 播し、シミュレーション空間内にはもどってこないように見せかける必要がある。これを 実現するためには、大きくわけると二つの方法がある。ひとつは、外向きの波動を数値的 に設けた吸収領域において減衰させる方法である。この方法は、電波無響室の原理と同様 で、有効な方法であるが計算量が増加することが欠点である。二つ目の方法は、自由空間 と同じインピーダンスを持つ媒質を境界面上に置く方法である。この方法は、計算量の増 加はわずかですむ反面、プラズマ波のような自由空間と異なる屈折率を持った媒質中を伝 播する波動に対しては適用が困難である。この説では、幅広い適用が可能な吸収境界条件 について述べることにする。

吸収境界条件の設定法はいくつかあるが、基本となる考え方は以下のファラデーの法則に 磁流項を加える方法である。

$$-c\nabla \times E = \frac{\partial B}{\partial t} + J_m$$
 (11)

ここで、 $J_m = \sigma_m(x) B$ であり、仮想的に磁気単極子の流れを作ることに相当する。このような操作と同様の手法としては、1以下の定数を磁場成分に毎ステップ乗じることによっても可能である。

マスキング法は、電界、磁界などの物理量を直接的に減衰させる方法である。使用する減 衰係数 f_M(x)の空間分布は図5に示すように、x に関して2次関数で与えると比較的効率的 に減衰できることがわかっている。

$$E(x) \to f_M(x)E(x) \tag{13}$$

$$E(x) \to f_M(x)E(x) \tag{14}$$

ここに、 $f_M(x)$ は

$$f_{M}(x) = \begin{cases} 1 & (|x| \le L) \\ 1 - \left(\frac{|x| - L}{L_{D}}\right)^{2} & (|x| > L) \end{cases}$$
(15)

となる減衰関数である。この手法に改良を加えた方法なども開発されている。

さらに純粋に横伝播の電磁波に対してのみ吸収効果を持つ方法として、電場の回転から電 流成分を求める方法が使用される。

 $J = c\nabla \times M = -\nabla \times [\alpha(x)\nabla \times E]$ (12) この方法は、静電波成分にまったく影響せずに、電磁波成分のみを減衰させることができ る点において、興味深い方法であるといえる。

いづれの方法においても、減衰させる波の波長と同程度の長さの減衰領域がシステムに必要となる。十分な減衰を得るためには、シミュレーション空間の1方向のみを吸収境界とする場合でも、システム全体の半分、2方向を吸収境界とした場合はシステム全体のおよそ 1/4を吸収領域に設定する必要があり、計算量が非常に大きくなるが、現実的なシミュレーションを行うためには、現段階ではもっとも実用的な方法といえる。



図3.2次元のシミュレーション空間の x<0,x>Lx の領域に吸収境界を取り付け、y 方向は 周期境界を設定した例。



図4.吸収境界を物理領域の両側に設け、この領域では電界、磁界を減衰させることによって反射を抑える。

6. 非構造格子電磁粒子コード

以上のような境界条件を組み合わせることによって、宇宙空間を飛翔する衛星などの飛翔 体環境を模擬することも可能となる。飛翔体などの複雑な形状をシミュレーションのモデ ルとして取り入れるためには、直行格子によって空間を離散化するのみでは不十分で、構 造格子あるいは非構造格子を使った空間格子が必要となる。非構造格子電磁粒子コードで は、空間を三角要素に分割して電磁界の離散化を行う。図5は、一つの三角要素上におい て離散化された電界、磁界の配置を示したずである。電界、磁界のXY面内の成分は、三角 要素の辺上で定義され、Z 成分は三角要素の重心で定義されている。これらの空間格子はシ ミュレーションに適した Delauney-Voronoi Mesh を形成する。

この離散化により、 E_z 成分の時間発展を計算するために必要な $\nabla \times B$ は以下のようにして 計算できる。

$$(\nabla \times B_{xy})_z \cong \frac{B_{xy1} \cdot l_1 + B_{xy2} \cdot l_2 + B_{xy3} \cdot l_3}{A_e}$$
 (13)

同様に、*Bxy*成分の時間発展は隣り合う三角形要素で定義された*E*z 成分の差分をとることによって計算することができる。



図5.三角形要素上における電界、磁界の離散化。

図6(a)は、非構造格子によってシャトル型飛翔体近傍の電磁環境をモデル化した例である。 (b)は電磁波の伝播の様子をシミュレートした例で、ほぼ任意の形状の飛翔体環境に適用す ることが可能となる。図7は、3次元空間を4面体要素によって離散化した例である。飛 翔体の表面物性に合わせて、導体境界条件、誘電体境界条件、吸収境界条件を組み合わせ ることにより、このような宇宙飛翔体環境のシミュレーションが実用になりつつある。



図6.2次元のシミュレーション空間に導体境界条件を内部境界として使用して飛翔体を モデル化(a)を行い、電磁波の伝播実験を行った例(b)



図7.3次元の飛翔体環境シミュレーションコードの例。4面体によってシミュレーション空間を離散化をおこなっている。

参考文献

[1] Birdsall, C. K. and A. B. Langdon, Plasma Physics via Computer Simulation, McGraw-Hill, 1985

[2] Okada, M., Study on spacecraft-plasma interaction in fast plasma flow via computer experiments, Ph.D thesis, 1994

[3] Okada, M. and H. Matsumoto, Simulation techniques of electromagnetic environment in the vicinity of a spacecraft, ISTS 2002 at Matsue, 2002

MPI によるプラズマ粒子コードの並列化

¹寺田 直樹、²村田 健史

- 1. 名古屋大学太陽地球環境研究所
- 2. 愛媛大学総合情報メディアセンター (兼:工学部情報工学科)

目次

1.	はじる	めに2
2.	ハイ:	ブリッド分割法の概要3
2	.1.	プラズマ粒子コードにおける各サブルーチンの計算量の比較
2	.2.	プラズマ粒子コードの並列化手法4
	2.2.1	9. 動的領域分割法
	2.2.2	?. ハイブリッド分割法
2	.3	ハイブリッド分割法の手順5
3.	ハイ:	ブリッド分割法の適用例5
3	.1.	プラズマ粒子流体混成コードについて5
3	.2.	ハイブリッド分割法による計算の流れ
3	. 3 . I	MPI による並列化
3	.4.	ハイブリッド分割法で用いる MPI 関数10
4.	並列	化による速度向上率の評価14
4	.1. 🔅	速度向上率の評価式14
4	.2.	速度向上率の予測と実測評価1 ϵ
	4.2.1	1. 全てのサブルーチンで並列化数が等しい場合10
	4.2.2	?. サブルーチンごとに並列化数を設定する場合17
5.	まと	め18
6.	参考	文献18

Abstract

本講義では、プラズマ粒子シミュレーションコードの並列化手法の一つであるハイブリッド分割法の解説を行う。 ハイブリッド分割法とは、粒子計算サブルーチンに粒子分割法を用いる一方で、電磁場および流体計算サブルー チンには領域分割法を用いて並列化を行う手法である。プロセッサ間の通信には MPI (Message-Passing Interface) 関数を用いたコード並列を行い、その並列化の手続きを、プラズマ粒子流体混成コードへの適用例を 交えて見ていく。MPI 関数の通信速度特性と通信ロジックに基づいて、粒子計算、電磁場計算、流体計算、およ び電流計算の各サブルーチンの並列化数を決定し、最大速度向上率を得る方法についても言及する。

1. はじめに

プラズマ粒子コードは、宇宙プラズマや核融合プラズマなどの様々なプラズマ研究分野で利用され、特 に、プラズマの粒子性が重要となるパラメタ領域でのプラズマ現象の解明に重要な役割を果たしてきた。 近年においては計算の大規模化が進み、大容量のメモリや高速演算を必要とする大規模プラズマ粒子シミ ュレーションが数多く行われている。一般に、プラズマ粒子コードでは、同一格子点数を用いた MHD コ ードと比較して数十倍から数百倍の主記憶容量および演算量を必要とする。近年の大規模粒子シミュレー ションでは数十 GB 以上の主記憶容量を要するものも珍しくなく、そのような計算の実行は、単一プロセ ッサから成る計算機ではもはや実現が困難である。主記憶容量の大規模化を実現し、かつ、膨大な演算時 間の短縮を図る並列プログラミングは、今やプラズマ粒子シミュレーションに不可欠である。

このように、並列プログラミングの必要性は近年ますます高まってきている。しかし我々科学者のよう に応用プログラムを手がける者には、出来る限りプログラムの並列化に労力をかけたくないという本音も またあるのではなかろうか。残念ながら現状の並列プログラミング言語にはユーザが何の労力も払わずに 並列化を行えるものは存在しておらず、「何もしなくても並列化されて速くなる」といういわゆる完全自動 並列化コンパイラは実現されていない。決定的な並列プログラミング言語が存在しない今、我々は並列化 に対して大なり小なり労力を払わなければならないが、無駄な労力を減らすためにも、我々は適切な(記 述が容易、ポータビリティが良い等々) 並列プログラミング言語を選択すべきである。特定のアーキテク チャやメーカに依存する並列プログラムを書いたが為に、プラットフォームの変更に伴うプログラムの書 き換えに数ヶ月を要したというのはよく聞く話であり、そのような事態は避けなければならない。近年、 並列プログラミング言語の標準化は盛んに行われており、並列プログラムライブラリとしては、MPI (Message-Passing Interface), PVM (Parallel Virtual Machine), pthread 等が存在し、高級言語としては、 HPF (High Performance Fortran), OpenMP 等が存在している。本講義では、メッセージ交換ライブラリ の事実上の標準である MPI-2 を用いた並列プログラミングについて解説を行う。MPI はフリーで入手可能 であり、世界中のほとんどの並列マシン上で実行可能である上に、わりと本格的な並列プログラミングを 可能とする。MPIのようなメッセージ通信による並列プログラミングは、分散メモリ型並列計算機のハー ドウェア構成に即し高い実行性能が得られる一方で、共有メモリ型並列計算機においても容易に実行可能 である。本講義の MPI の解説は、講義の都合上、ハイブリッド分割法に関係する MPI 関数の解説を中心 としたものとなっている。MPI に関する一般的な事項の詳細をさらに希望する人は参考文献¹⁾⁻³⁾を参照さ れたい。

プラズマ粒子コードの並列化手法には、大別して粒子分割法と領域分割法が存在する。本講義で紹介す る並列化手法は、この2つの手法を組み合わせたハイブリッド分割法⁴⁾である。ハイブリッド分割法では、 粒子計算部は粒子分割法で、電磁場および流体計算部は領域分割法を用いて並列化を行う(図1)。ハイブ リッド分割法は実装が比較的容易ながら、高い速度向上率を得ることが可能という利点を持つ。

2. ハイブリッド分割法の概要

2.1. プラズマ粒子コードにおける各サブルーチンの計算量の比較

一般に、プラズマ粒子コードは、粒子計算と電磁場計算(および流体計算)から成る。格子点上で定義 された電磁場は、空間を自由に移動する多量のプラズマ粒子と PIC (Particle-In-Cell)法によって結び付け られ、コードのスキームに従ってその時間発展が解き進められる。この様な現在一般的に用いられている PIC プラズマ粒子コードにおいては、粒子計算部が計算全体に占める割合は、通常大きなものとなる。

プラズマ粒子シミュレーションでは、速度分布関数は位置空間および速度空間において十分に滑らかで ある必要がある。図2の速度分布関数は (a) 64 個, (b) 160 個, (c) 320 個, (d) 640 個, (e) 1000 個の粒子が作 る速度分布関数である。図2より、速度分布が十分に滑らかである為には、格子点あたりの粒子数が数百 個以上必要であるといえる。一般には、プラズマ粒子コードでは格子点あたり数十から数百個の粒子が用 いられており、その粒子数に応じて粒子計算量も電磁場計算量(および流体計算量)の数十倍から数百倍 になる。このことから、プラズマ粒子コードの計算効率の向上には、まず、計算量の多い粒子計算部を効 率的に計算する必要があることがわかる(後述するがこれは効率の良い並列計算にとっては不十分な条件 である)。

(a) Dynamic Region Division Method



	°°°°°°°°°°°°°°°°°°°°°			● PE0 PE1 ● ■ PE2 ● ■ PE3
- 0 -	0 ° °	0000	0 0	



図 1 (a) 動的領域分割法と (b) ハイブ リッド分割法

図 2 異なる粒子数による速度分布関数。(a) 64 個, (b) 160 個, (c) 320 個, (d) 640 個, (e) 1000 個の粒子によ り構成されている。速度空間は 50 分割して表示して いる。

2.2. プラズマ粒子コードの並列化手法

並列計算を効率よく行うためには、(1) 各 PE (Processing Element) に掛かる負荷バランスを均等にす る、(2) PE 間の通信時間を出来る限り減らす、(3) 並列化可能部分を出来る限り増やす、ことが必要であ る。コードの並列化に用いる並列化手法には、これらの条件をなるべく満たすような手法を用いることが 望ましい。プラズマ粒子コードの並列化手法としては、動的領域分割法や粒子分割法などがある。

2.2.1. 動的領域分割法

動的領域分割法では図1(a) に示すように、各領域に含まれる粒子数が均等になるように領域の分割を動 的に行う。その結果、粒子計算においては均等分割が実現できるが、電磁場(および流体)計算で各 PE の負荷バランスにばらつきが生じてしまい、(1)の均等な負荷バランスを満たさなくなる。また、粒子が領 域をまたいで移動する際に、(2) に該当する粒子データの通信が発生する。

2.2.2. ハイブリッド分割法

一方、本講義で紹介するハイブリッド分割法では、図1(b)に示すように粒子計算を均等データ分割し、 電磁場(および流体)計算を均等領域分割する。その結果、ハイブリッド分割では、ほぼ完全に均等な負 荷バランスが実現でき、(1)の条件を満足する。

次に、(2) PE 間データ通信量の低減および(3) 並列化不可能部分の低減について考える。プラズマ粒子 コードでは前節で述べたように粒子計算部が全体に占める割合が大きいので、粒子計算を効率的に並列化 することが「まず」全計算効率の向上に重要となる。ハイブリッド分割法では、粒子量をデータ分割(粒 子分割)しているので、プラズマ粒子の運動計算部においては PE 間の通信が発生しない。その結果、運 動計算部に要する時間は並列化数が増えるごとに反比例して短縮される。一方、各粒子の持つ速度や電荷 は、モーメント計算部において、粒子近傍の格子点に足し込まれる。PE 間の通信は、この各 PE の格子点 上に足しこまれたモーメント量を全 PE で合計する際に発生する。したがって、粒子計算部においては、 各 PE で求めたモーメント値を合計する計算においてのみ、(2) に該当する PE 間通信が発生する。また、 粒子計算部ではほとんど全ての計算がループ処理になっているので、(3) の並列化不可能部分はほとんど存 在しない。

続いて電磁場計算部および流体計算部での、PE 間通信と並列化不可能部分の低減を考える。プラズマ粒 子の運動計算部では粒子位置における電磁場の値が必要となるが、ハイブリッド分割法では、ある PE に 所属する粒子はシミュレーション空間全体を自由に移動することが可能であるので、全ての PE が全空間 の電磁場データを保持する必要が生じる。ここで、プラズマ粒子コードにおいては、電磁場および流体計 算に必要な計算量は全体の数%程度にしか過ぎないことを考慮すると、最も簡単な方法としては、全ての PE が全空間の電磁場データを保持して同じ電磁場計算を並行して行う(逐次処理し、PE 間の電磁場デー タ通信を全く行わない)というものが考えられる。しかし、有名なアムダ - ルの法則:

n 台の PE を用いて得られる速度向上率 =
$$\frac{1}{\alpha + \frac{1 - \alpha}{n}}$$
 (α : 逐次処理部分の割合) (1)

4

によると、少量の逐次処理でも全体の並列処理の効率を妨げる原因となってしまう(図3)。例えば、単一 PE で実行したときに逐次計算が全体の5%あったとすると、図3より、たとえ10,000 台の PE を用いた としても得られる速度向上は高々20 倍にしかならない。このように、たとえ少量である電磁場や流体の計 算に関しても、並列化を行わなければ高い並列処理効率を得ることは期待できない。ハイプリッド分割法 では、この並列処理効率の低下を避けるために、電磁場および流体計算部も並列化を行う。ハイプリッド 分割法では電磁場および流体計算の並列化に、計算する領域を実空間上で均等に分割し、各領域の計算を 各 PE に割り当てる均等領域分割法を採用している。ただし、電磁場計算部と流体計算部では、PE 間のデ ータ通信は異なる通信となる。

電磁場計算部においては、各 PE で の粒子計算において全空間の電磁場値 が必要となることから、各 PE は担当 する領域での電磁場を計算した後、担 当領域のデータを全ての PE 間で交換 する。一方、流体量は粒子計算に不要 であるので、分割したまま計算を進め ることができる。したがって流体計算 では、差分計算に必要となる計算担当 領域の境界部(のりしろ部)のデータ のみ交換すればよい。



図 3 並列処理の速度向上率

この議論からわかるように、最大速度向上率を生む各計算部の並列化数は一概には決定できない。特に 電磁場計算部では PE 間のデータ通信量が多いので、(2)の PE 間通信量の低減と(3)の並列処理を出来る 限り増やす、の二つの要請を同時に満たすことは困難となる。それぞれの計算部での並列化数の決定方法 については4章で議論する。

2.3. ハイブリッド分割法の手順

前節で述べたハイブリッド分割法では、一般的には以下の手順で、各計算部サブルーチンの実装を行う。 タイムチャートを作成し、各サブルーチンの計算順序を決定する。

粒子計算サブルーチンを粒子分割法により実装する。電磁場計算サブルーチン(および流体計算サブ ルーチン)を領域分割法により実装する。

各サブルーチン間の関係を調べ、それに応じた通信関数をサブルーチンに実装する。

各サブルーチンの計算時間と通信時間を、測定および予測により求める。

各サブルーチンの並列化数を決定する。

- 3. ハイブリッド分割法の適用例
- 3.1. プラズマ粒子流体混成コードについて

本章では、ハイブリッド分割法のプラズマ粒子コードへの適用を具体的に見る。ここでは例として、プ

ラズマ粒子コードの一つであるプラズマ粒子流体混成コード(プラズマ電磁ハイブリッドコード)への適 用を考える。プラズマ粒子流体混成コードは、イオンは超粒子として個々の軌道を追跡する一方で、電子 は(無質量の)流体として扱うもので、イオンスケールの低周波現象の解析に適している。格子点上で定 義された電磁場および流体量と、シミュレーション領域をラグランジュ的に移動する粒子量とのやりとり は、PIC 法を用いて行われる。

プラズマ粒子流体混成コードで用いられる基本方程式は以下の通りである。

◆ *s* 種イオンの運動方程式(超粒子として扱う)

$$m_s \frac{dv}{dt} = q_s (E + v \times B) - q_s \eta J$$
⁽²⁾

$$\frac{dx}{dt} = v \tag{3}$$

◆ 電荷の準中性条件式

$$-en_e + \sum_s q_s n_s = 0 \tag{4}$$

◆ 電子の運動方程式(無質量の流体として扱う)

$$n_e m_e \frac{du_e}{dt} = 0 = -en_e (E + u_e \times B) - \nabla P_e + en_e \eta J$$
(5)

◆ 電子のエネルギーの式

$$\left(\frac{\partial}{\partial t} + u_e \cdot \nabla\right) P_e = -\gamma_e P_e \nabla \cdot u_e + (\gamma_e - 1)\eta J^2$$
(6)

◆ マクスウェル方程式(ダーウィン近似)

$$\frac{\partial B}{\partial t} = -\nabla \times E \tag{7}$$

$$I = \frac{1}{\mu_0} \nabla \times B \tag{8}$$

◆ 電流密度の定義式

$$J = -en_e u_e + \sum_s q_s n_s u_s \tag{9}$$

プラズマ粒子流体混成コードでは、これらの基本式を差分化し、各変数 (x,v:粒子量、B,E:電磁場量、 n_e, u_e, P_e, J :電子流体量および電流)の時間発展を解き進めていく。

3.2. ハイブリッド分割法による計算の流れ

プラズマ粒子流体混成コードでは、式(2)~(9)をサブルーチン化して解き進めるが、その処理順序はコードが採用するスキームに依存している。以下では、図4の流れ図にある処理順序を考える。

図4の(1),(5),(9),(14),(15)は電磁場計算である。(2),(6),(7),(8),(10),(13)は電子流体および電流計算、 (3)(4),(11)(12)は粒子計算を表している。図4の例では、粒子計算サブルーチンは粒子分割法を用いて、電 磁場計算サブルーチンおよび流体計算サブルーチンは領域分割法を用いて、4つの PE で並列計算を行っている。ここで考える並列化手続きは、2.3.の手順 と手順 にあたる。



図 4 プラズマ粒子流体混成コード(プラズマ電磁ハイブリッドコード) の並列処理の流れ(4PEの場合)

プラズマ粒子流体混成コードのハイブリッド分割法による計算の流れは、

- I. 図 4 の (1) で各 PE は担当する領域の磁場計算を行う。(3)(4)の粒子計算のために全ての PE が全 領域の磁場データを保持している必要があるので、各 PE は担当領域の磁場計算結果を全ての PE にコピーする(通信に用いる MPI 関数: MPI_BCAST)。
- II. 次に (2) で各 PE が担当する領域の電流の計算を行う。

- III. (3)(4) では、粒子の位置、速度を時間的に進める。
- IV.(5) で再度、磁場の値を求め、計算結果を全ての PE にコピーする(通信に用いる MPI 関数:
MPI_BCAST)。
- V. (6) で再度、電流の値を求める。
- VI. (7) では、各 PE が担当する粒子のモーメント量を求める。粒子イオンの粒子量を PIC 法によって 格子点上に足し合わせてモーメントを各 PE で得た後、全領域のモーメント量を全ての PE で足し 合わせる(通信に用いる MPI 関数: MPI_ALLREDUCE)。イオンのモーメント量 u_s, n_s をも とに、電子流体の u_e, ρ を求める。
- VII. (8) で各 PE は、担当する領域の電子流体計算を行う。差分計算で必要となる計算担当領域の境界部(のりしろ部)のデータ交換を行う(通信に用いる MPI 関数: MPI_SENDRECV)。
- VIII. (9) では、各 PE が担当する領域の電場の計算を行う。(1), (5) の磁場の計算と同様に、各 PE は担当領域の計算を行った後、計算結果を全ての PE にコピーする(通信に用いる MPI 関数: MPI_BCAST)。
- IX. (10) 以降は (1)~(9) と同様に行う。

である。この並列処理においては、2.3.の手順 に相当する「データおよびループ処理の分割」と、手順 に相当する「PE 間の通信」の2つの並列化手続きが含まれている。次節では、MPI を用いてこれらの並 列化手続きを記述する方法について説明する。 3.3. MPI による並列化

MPI-2 (Message-Passing Interface) は現在多くの並列計算機で標準化されている代表的なメッセージ 交換ライブラリである。ここでは、MPI の説明を行う前に、まず、1(=is0) から 10(=ie0) までの和を求め る MPI サンプルプログラムを見ることにする。以下ではプログラム実行時に、並列数を3と指定した場合 を考える。

program sample_sum include 'mpif.h' !! おまじない1 call MPI_INIT(ier) !! おまじない2 call MPI_COMM_SIZE(MPI_COMM_WORLD, nproc, ier) !! おまじない3 call MPI_COMM_RANK(MPI_COMM_WORLD, irank, ier) !! おまじない4(プロセスのランクの取得) !! PE1: irank=0 !! PE2: irank=1 !! PE3: irank=2 is0=1 ; ie0=10 is=irank*((ie0-is0+1)/nproc)+is0+min(irank,mod(ie0-is0+1,nproc)) ie=is+(ie0-is0+1)/nproc-1 if(mod(ie0-is0+1,nproc).gt.irank) ie=ie+1 !! 各プロセスでのループ始値、終値を求める !! PE1: is=1,ie=4 !! PE2: is=5,ie=7 !! PE3: is=8.ie=10 isum=0 ! do i=is0,ie0 !! ループ始値、終値を is0, ie0 から do i=is.ie !! is. ie に置き換える isum=isum+i enddo !! PE1は isum=10を出力 print *, irank, isum !! PE2 は isum=18 を出力 !! PE3 は isum=27 を出力 call MPI_REDUCE(isum,itemp,1,MPI_INTEGER,MPI_SUM,0,MPI_COMM_WORLD,ier) !! isum の値を PE1 に集める isum=itemp if(irank.eq.0) print *, irank, isum !! PE1は isum=10+18+27=55を出力 !! おまじない5 call MPI_FINALIZE(ier) end program sample_sum

MPIでは、基本的に、各 PE が同一の実行プログラムを並行して実行する(図5)。そして並列処理部分 に来ると、ループ処理等を分割して並列計算する。一つのプログラムでありながらそれぞれの PE で異な る動作を実現させる根源は、プロセスのランク(ここでは irank)にある。サンプルプログラムでは、irank を用いて、do ループの始値 is と終値 ie にそれぞれの PE で異なる値を与えている。MPI ではこのように して並列処理を行っていくが、ここでもし並列化に伴う"矛盾"が計算に生じたら、矛盾を解消するため に(ここで初めて!)MPI 関数を用いた通信を行い、矛盾の修正を行う。効率の良い並列計算には MPI 関数による通信を必要最小限にすることが不可欠である。



図 5 サンプルプログラムの並列処理の流れ

サンプルプログラムにおいて、後半の MPI_REDUCE を用いた PE 間通信が無ければ PE1 は小計 10 を 出力するのみである。しかし今は小計ではなく「合計」が必要なので、MPI_REDUCE を用いて各 PE の 小計を足し合わせ、合計 55 を求める必要がある(並列化に伴う矛盾の解消)。

ここで、サンプルプログラムの irank を用いた「ループ処理の分割」は 2.3.の手順 に対応し、 MPI_REDUCE を用いた「PE 間の通信」は 2.3.の手順 に対応していることに注意されたい。

3.4. ハイブリッド分割法で用いる MPI 関数

3.2. で見たように、ハイブリッド分割法のプラズマ粒子流体混成コードへの適用においては、(1) MPI_ALLREDUCE, (2) MPI_BCAST, (3) MPI_SENDRECVの3つのMPI 関数が用いられる。ここでは、 この3つの MPI 関数の機能や利用法について説明を行う。

(1) MPI_ALLREDUCE

機能

コミュニケータ(comm)内の全プロセスの送信バッファ(sendbuff)のメッセージに対して、通信 しながら演算(op)を行い、結果を全プロセスの受信バッファ(recvbuff)に返す。

<u>使用方法 (Fortran)</u>

call MPI_ALLREDUCE(sendbuff, recvbuff, count, datatype, op, comm, ierror)

- ・ sendbuff(recvbuff)には、送信(受信)バッファの先頭アドレスを指定する。送信バッファと 受信バッファは、実際に使用する部分では重ねてはいけない。
- ・ count には、送信メッセージの要素数を指定する。
- datatype や op には、MPI であらかじめ用意されているデータ型や演算(MPI_INTEGER、 MPI_SUM など)を指定する。または自分で定義したデータ型や演算を指定する。

ハイブリッド分割法では、MPI_ALLREDUCE は各 PE で格子点上に集められたモーメント量を全 PE で足し合わせる際に用いられる。以下のサンプルプログラムに使用例を示す。

サンブルブログラム2	スタート	
program sample_allreduce		
include 'mpif.h'		!! *1
integer(kind=4),parameter :: nx=20,np=100		
real(kind=4),dimension(nx) :: den,vel,vtemp		
! real(kind=4),dimension(np) :: r,v		
real(kind=4),dimension(:),allocatable :: r,v		
call MPI_INIT(ier)		!! *2
call MPI_COMM_SIZE(MPI_COMM_WORLD,nproc,	ier)	!! *3, *4
call MPI_COMM_RANK(MPI_COMM_WORLD,irank	,ier)	!! *5

ns0=1; ne0=np

ns=irank*((ne0-ns0+1)/nproc)+ns0+min(irank,mod(ne0-ns0+1,nproc))



v(n)=3.0				
end do				
vel(1:nx)=0.0				
den(1:nx)=0.0				
! do n=ns0,ne0 1 フロセスで実行時のループ始値、終値から				
do n=ns,ne	複数ノロセスで実行時のルー	フ始値、終値におきかえる。		
i=int(r(n),kind=4)				
den(i)=den(i)+1.0				
vel(i)=vel(i)+v(n)				
enddo				
if(irank.eq.0) print *, irank,'d	en_before=',den			
call MPI_ALLREDUCE(den,v	/temp,nx,MPI_REAL,MPI_SUM,	MPI_COMM_WORLD,ier)		
den(1:nx)=vtemp(1:nx)				
call MPI_ALLREDUCE(vel,v	temp,nx,MPI_REAL,MPI_SUM,M	MPI_COMM_WORLD,ier)		
vel(1:nx)=vtemp(1:nx)				
vel(1:nx)=vel(1:nx)/(den(1:nx)	+1.0e-20)	_ 各プロセスの格子点上の値 den, vel }		
if(irank.eq.0) print *, irank,'d	en_after=',den	を全プロセスで足し合わせる。		
if(irank.eq.0) print *, irank,'v	el_after=',vel			
call MPI_FINALIZE(ier)		!! *6		
end program sample_allreduc	:e			
	サンプルプログラム2 エンド			
*1 インクルードファイルの	指定。mpif.h では MPI で使用する	る変数の宣言等がなされている。		
*2 MPI 環境の初期化。				
*3 (指定したコミュニケー	タの)プロセス数の取得。			
*4 MPI_COMM_WORLD	は mpif.h で定義済みのコミュニケ	ータ。		
*5 (指定したコミュニケー	タ内での)プロセスのランクの取行	得。		
*6 MPI 環境の終了処理。				
サンプルプログラム 2 ではル	レープ処理の分割以外に、配列の分	}割も行っている。配列の分割は、プラズ		
マ粒子コードのように大きな	∶主記憶要領を必要とする計算に	おいてしばしば利用される。ここでは		

Fortran90の allocatable 宣言の機能を利用して、各PEへの配列の分割を行っている。

(2) MPI_BCAST

12

機能

コミュニケータ(comm)内の一つの送信元プロセス(root)の送信バッファ(buff)のメッセージを、 その他の全プロセスの受信バッファ(buff)に送信する。

<u>使用方法 (Fortran)</u>

call MPI_BCAST(buff, count, datatype, root, comm, ierror)

buffには、送信プロセスでは送信バッファの先頭アドレスを指定する。宛先プロセスでは受信バッファの先頭アドレスを指定する。

ハイブリッド分割法では、MPI_BCAST は各 PE が計算を担当した領域の磁場や電場データを、他の全 ての PE にコピーする際に用いられる。送信するデータ量が多く、ハイブリッド分割法では最も通信時間 がかかる通信の一つである。

(3) MPI_SENDRECV

機能

MPI_SENDとMPI_RECVをまとめて行う。MPI_SENDでは、送信バッファ(sendbuff)のsendcount 個のメッセージをタグ(sendtag)をつけて宛先プロセス(コミュニケータ comm 内でランクが dest) に送信する。MPI_RECVでは、送信元プロセス(コミュニケータ comm 内でランクが source)から 送信されたタグ(recvtag)のメッセージを recvcount 個、受信バッファ(recvbuff)に受信する。送 受信が完了するまで待ちの状態に入る(プロッキング通信)。

<u>使用方法 (Fortran)</u>

call MPI_SENDRECV(sendbuff, sendcount, sendtype, dest, sendtag, recvbuff, recvcount, recvtype, source, recvtag, comm, status, ierror)

- sendbuff(recvbuff)には、送信(受信)バッファの先頭アドレスを指定する。送信バッファと
 受信バッファは、実際に使用する部分では重ねてはいけない。
- sendtype (recvtype)には、送信メッセージ(受信メッセージ)のデータ型(MPI_INTEGER など)を指定する。
- dest には、宛先プロセスの comm 内でのランクを、source には、受信したいメッセージを送信 するプロセスの comm 内でのランクを指定するが、メッセージを送信(受信)したくないときに はワイルドカード MPI_PROC_NULL を指定する。
- status には、状況オブジェクトの配列を指定する。大きさは MPI_STATUS_SIZE である。include 'mpif.h' より後で、たとえば「integer(kind=4),dimension(MPI_STATUS_SIZE)::istatus」のようにして配列宣言を行う必要がある。

ハイブリッド分割法では、MPI_SENDRECV は電子流体量の差分計算において用いられる。すなわち、 各 PE が担当する領域の電子運動方程式と電子エネルギーの式の差分計算に必要となる領域の境界部(の りしろ部)のデータを、隣り合う領域を担当する PE に送受信する際に用いられる。送受信するデータは 領域の境界部のデータだけなので、送信データ量は通常少なく、ハイブリッド分割法ではあまり時間がか からない通信である(流体・MHD コードでは重要な通信であるが)。

差分計算でシミュレーション領域の最も端(シミュレーション領域の境界)の領域を担当する PE の MPI_SENDRECV の利用においては、通常 if 文などによる条件分岐が必要となるが、そこではワイルド カード MPI_PROC_NULL を利用すると便利である。たとえば、

iup=irank+1

idown=irank-1

if(irank.eq.nproc-1) iup=MPI_PROC_NULL

if(irank.eq.0) idown=MPI_PROC_NULL

call MPI_SENDRECV(a(1,1,ke), nx*ny, MPI_REAL, iup, 100 &

, a(1,1,ks-1), nx*ny, MPI_REAL, idown, 100, MPI_COMM_WORLD, istatus, ier) call MPI_SENDRECV(a(1,1,ks), nx*ny, MPI_REAL, idown, 110 &

, a(1,1,ke+1), nx*ny, MPI_REAL, iup, 110, MPI_COMM_WORLD, istatus, ier) のように利用する。

4. 並列化による速度向上率の評価

本章では 2.3.の手順 と手順 、すなわち、各サブルーチンの計算時間と通信時間の評価、および各サ ブルーチンの並列化数を決定する方法について言及する。以下で示すように、ハイブリッド分割法では全 てのサブルーチンを最大の PE 数で並列化しないほうが速度向上率が良い場合がある。したがって、各サ ブルーチンそれぞれの並列化数を決定するための、速度向上率の評価式が必要となる。ここでは例として、 3.1.のプラズマ粒子流体混成コード(プラズマ電磁ハイブリッドコード)における定式化を行う。速度向上 率の評価式は、参考文献4に従い、富士通 AP3000, 16PE の実測結果と比較し議論する。

4.1. 速度向上率の評価式

ハイブリッド分割法における各サブルーチンの並列化数を n_i 、各サブルーチンの全処理時間を $T_i(n_i)$ とする。ここでiは、サブルーチンを識別する引数であり、P(粒子計算) E(電場計算) B(磁場計算) e(電子流体計算) およびJ(電流計算)をとる。この時、最大並列化数nの場合の、速度向上率S(n)は

$$S(n) = \sum_{i}^{P,E,B,e,J} T_{i}(1) / \sum_{i}^{P,E,B,e,J} T_{i}(n_{i})$$
(10)

で与えられる。

ここで、粒子の全計算時間 $T_p(n_p)$ は

$$T_{P}(n_{P}) = \frac{T_{P}^{cal}(1)}{n_{P}} + T_{P}^{com}(2) \cdot \log_{2} n_{P}$$
(11)

である。右辺第一項目は、*n_p*並列時の実計算時間である。プラズマ粒子シミュレーションでは各サブルー チン内での実計算のほとんどがループ計算になっているので、*n_i*並列時のサブルーチン*i*の実計算時間は $T_i^{cal}(n_i) = T_i^{cal}(1)/n_i$ となる。右辺第二項目は、 n_p 並列時に MPI 通信に要する時間である。粒子計算における通信には MPI_ALLREDUCE が用いられる(図4参照)。ここで、図6(c)に示すように MPI_ALLREDUCE の通信では、n並列(ただしnは2のべき乗)の場合、 $\log_2 n$ 回の通信が必要となる。従って、通信に要する時間は式(11)の右辺第二項のようになる。

電場計算サブルーチンの全処理時間は $T_E(n_E)$

$$T_E(n_E) = \frac{T_E^{cal}(1)}{n_E} + T_E^{com}(2) \cdot \log_2 n_E$$
(12)

で与えられる。右辺第二項は、式(11)の右辺第二項と同様の形をとる。電場計算結果の通信に用いる MPI_BCAST(図4参照)の内部通信ロジックを示す図6(b)にあるように、MPI_BCASTでは MPI_ALLREDUCEと同数の通信を必要とする。一回の通信量は、2並列の場合と比較して、2/n倍とな り、通信時間はn/2倍となる。従って、通信に要する時間は式(12)右辺第二項のようになる。磁場計算の 全処理時間も同様であるので

$$T_B(n_B) = \frac{T_B^{cal}(1)}{n_B} + T_B^{com}(2) \cdot \log_2 n_B$$
(13)

となる。

電子流体の MPI 通信には MPI_SENDRECV を用いる(図4参照)ので、

$$T_e(n_e) = \frac{T_e^{cal}(1)}{n_e} + T_e^{com}(2)$$
(14)

となる。ここで、MPI_SENDRECVの内部通信ロジックは図6(a)であるので、通信に要する時間は右辺 第二項の形で与えられる。最後に、電流計算はMPI通信が不要であるので

$$T_{J}(n_{J}) = \frac{T_{J}^{cal}(1)}{n_{J}}$$
(15)

で与えられる。



図 6 MPI 関数の内部通信ロジック (4 並列および 8 並列の場合): (a) MPI_SENDRECV, (b) MPI_BCAST, (c) MPI_ALLREDUCE.

4.2. 速度向上率の予測と実測評価

本節では、プラズマ粒子流体混成コード(プラズマ電磁ハイブリッドコード)を用いた速度向上率の予 測値と実測値の比較を行う。4.2.1.では、全てのサブルーチンを、最大並列数で並列化した場合の予測と実 測評価をみる。4.2.2.では、各サブルーチンごとの処理時間を調べ、最も高い速度向上率を得るために、サ ブルーチンごとに並列化数を設定する。

4.2.1. 全てのサブルーチンで並列化数が等しい場合

図7に、式(10)より得られた速度向上率の予測値と、AP3000で速度向上率を測定した結果を示す。図7 の (a), (b), (c), (d) の値は、全てのサブルーチンで並列化数を等しくした場合の値である。(a), (b), (c) では 格子点数を一定として粒子数を変化させ、(d) では格子点数を変化させている。



図 7 プラズマ粒子流体混成コードの速度 向上率の予測値と実測値 (AP3000 によ る)。粒子数は、(a) 640/grid, (b) 320/grid, (c) 160/gird, (d) 40/grid, (e) 640/grid。格 子点数は、(a), (b), (c), (e) が128×128、 (d) が 512×512。電磁場サブルーチンの 並列化数は、(a), (b), (c), (d) が最大並列化 数、(e) は逐次計算。(a)', (b)', (c)', (d)', (e)' は予測値、(a), (b), (c), (d), (e) は実測値。

図7(a), (b), (c) より速度向上率は、格子点数が等しい場合には、粒子数が多いほど向上することがわか る。これは、粒子計算サブルーチンと比較して、電磁場計算サブルーチンの並列化効率が悪いことに起因 している。図8に、図7(a)の計算の、サブルーチンごとの実計算時間と通信時間を示す。図8(c)粒子計 算、(d)電子流体計算、および(e)電流計算サブルーチンは、通信時間に比して実計算時間が大きいため、 スケーラビリティーが高い。一方、図8(a), (b)の電磁場計算サブルーチンは、計算量が小さく、また、ス ケーラビリティーの低い MPI_BCAST を通信関数として用いているため、並列化数が大きいほど全処理時 間が増加する。図8(c)の粒子計算の全処理時間が短くなると、相対的に、図8(a)や(b)の効果が大きく なる。

速度向上率の低下は、粒子数が等しい場合には、格子点数が多いほど顕著になる。比較のため、粒子数 が図7(a)の計算と等しく、格子点数を多くした場合の速度向上率の値を図7(d)に示す。図7(a)と比較 して、速度向上率の低下が顕著にみられる。



図 8 AP3000 によるプラズマ粒子流体混成コードの各サブルーチンの実計算時間とデータ通信時間:(a) 電場計算サブルーチン、(b) 磁場計算サブルーチン、(c) 粒子計算サブルーチン、(d) 電子流体計算サブルーチン、(e) 電流計算サブルーチン。

4.2.2. サブルーチンごとに並列化数を設定する場合

図8では、電磁場計算サブルーチンにおいて、並列化数が大きいほど、全処理時間が増加する傾向が見 られた。この結果は、サブルーチンごとに並列化数を変更することにより、さらに速度向上率を向上でき ることを示唆している。ここでは、サブルーチンごとの並列化数を個別に設定した場合の、速度向上率の 向上を見る。

図8より、16PE システムで全処理時間が最小となるのは、粒子、電子流体、および電流サブルーチン が、それぞれ、16,16,15 並列の場合である。一方、電磁場サブルーチンでは、1PE(逐次計算)の場合が 最も全処理時間が小さい。このように並列化数をサブルーチンごとに個別に設定して、速度向上率の予測 値および実測値を求めたものを、図7の(e)に示す。図7(e)では、図7(a)の場合と比べて、速度向上率 が約12%改善されている(16 並列時の実測値が12.96 14.45)。

ここで、図7(e) において電流計算サブルーチンの並列化数を 15 ではなく 16 とした。これはサブルー チンの並列化数が他のサブルーチンとの関係に依存するためである。たとえば、電流計算と電子流体計算 のサブルーチンの並列化数を異なる値で設定すると、両者の間に新しい通信が発生する。このとき速度向 上率は逆に低下することになるので、並列化数はここでは独立に設定できない。

5. まとめ

本講義では、プラズマ粒子コードの並列化手法である領域分割法と粒子分割法を組み合わせた、ハイブ リッド分割法の解説を行った。ハイブリッド分割法では、粒子計算サブルーチンを粒子分割し、電磁場お よび流体計算サブルーチンを領域分割して並列計算を行う。

ハイブリッド分割法は、動的粒子分割法と比較して、各 PE の負荷バランスを均等にでき、粒子データ の通信を必要としないという利点がある。しかし一方で、粒子計算に必要となる全空間領域の電磁場デー タを、全ての PE が保持していなければならないという主記憶容量上の不利点を持つ。1PE あたりのメモ リが少ないシステムで大規模粒子シミュレーションの超並列計算を行う際にこの不利点は問題となりうる ので注意が必要である。

本講義では、ハイブリッド分割法のプラズマ粒子流体混成コード(プラズマ電磁ハイブリッドコード) への適用例を紹介したが、ハイブリッド分割法は流体計算部を持たないプラズマ粒子コード(例えばフル 粒子コード)にも適用可能である。すなわち、ほとんどの標準的なプラズマ粒子コードに適用できる汎用 性の高い手法であると言える。

6. 参考文献

1) P. パチェコ、「MPI 並列プログラミング」、 培風館、 2001.

- 2) 青木幸也、「並列プログラミング虎の巻 MPI 版」、日本 IBM 株式会社、2001.
- 3) 湯浅太一ら、「初めての並列プログラミング」、共立出版、1999.
- 4) 村田健史ら、「プラズマ電磁粒子コードの並列化手法と速度向上率の評価」、情報処理学会論文誌、2002.