

静的電磁場中における荷電粒子の運動

Motion of charged particle in static electro-magnetic field

Author : 志多 友史 (Yuji Shida)

Date : 2017/3/4

Keywords : 荷電粒子(charged particle), クーロン力(Coulomb force), ローレンツ力(Lorentz force), E×B ドリフト(E-cross-B drift), C 言語(C programming language), 数値計算(numerical calculation), 電子銃(electron gun), サイクロトロン(cyclotron), マグネトロン(magnetron), 四重極電磁石(quadrupole electromagnet), トロイダルコイル(toroidal coil), ヴァン・アレン帯(Van Allen radiation belt)

Abstract:=====

本稿では時間変化の無い電場・磁場中における非相対論的荷電粒子の運動について検討したものである。内容としては高校物理で扱うレベルのクーロン力やローレンツ力の知識を基に、荷電粒子の運動を計算するというものである。また、電場と磁場が直交した場における荷電粒子の運動方程式の理論解（E×B ドリフト）とシミュレーション結果の比較も行った。

In this report, I examined the motion of non-relativistic charged particle under electro-magnetic field which has no time change. The content is the orbital simulation of charged particle which used HS level knowledge of Coulomb force and Lorentz force. In addition, I compared theory solution (E-cross-B drift) and simulation result under crossed electric and magnetic field.

=====

1. 序論(Introduction)

荷電粒子の運動を制御または利用する技術は、ブラウン管テレビや電子レンジのような身近なものから X 線発生装置や粒子線ガン治療、加速器など幅広い分野で用いられている。

本稿では、比較的エネルギーの低い、言い換えると相対論的影響を考慮せずに済む程度の速度で運動する荷電粒子のシミュレーションを行った。また直観に反する挙動をする E×B ドリフトについてもシミュレーションを行い、理論解とその結果を比較した。(※E×B ドリフトについて、直感的には電場の向きに沿って螺旋軌道を描きそうに見えるが実際は電場に対して直交した方向に運動する。)

2. 理論(Theory)

本シミュレーションプログラムに用いる基礎方程式は次の 2 式である。

$$m \frac{d^2 \mathbf{r}}{dt^2} = q \mathbf{E}(\mathbf{r}) \quad : \text{クーロン力}$$

$$m \frac{d^2 \mathbf{r}}{dt^2} = q \mathbf{v} \times \mathbf{B}(\mathbf{r}) \quad : \text{ローレンツ力}$$

ここで、 m は荷電粒子の質量(kg)、 q は荷電粒子の電荷量(C)、 \mathbf{r} は位置ベクトル(m)、 \mathbf{v} は速度ベクトル(m/s)、 $\mathbf{E}(\mathbf{r}), \mathbf{B}(\mathbf{r})$ は位置 \mathbf{r} における電場・磁場である。従って、荷電粒子の運動は

$$m \frac{d^2 \mathbf{r}}{dt^2} = q \{ \mathbf{E}(\mathbf{r}) + \mathbf{v} \times \mathbf{B}(\mathbf{r}) \}$$

に従う。ここで加速度を $\mathbf{a} = \frac{d^2 \mathbf{r}}{dt^2}$ とおいて、タイムステップ毎の速度、位置、加速度の関係式を作ると

$$\mathbf{v}(t+\Delta t) = \mathbf{v}(t) + \mathbf{a}(t) \Delta t$$

$$\mathbf{r}(t+\Delta t) = \mathbf{r}(t) + \mathbf{v}(t+\Delta t) \Delta t$$

$$\mathbf{a}(t+\Delta t) = \frac{q}{m} \{ \mathbf{E}(\mathbf{r}(t+\Delta t)) + \mathbf{v}(t+\Delta t) \times \mathbf{B}(\mathbf{r}(t+\Delta t)) \}$$

となる。これにより荷電粒子の初期条件が与えられれば逐次演算によって粒子の運動シミュレーションが可能となる。ここで微小時間 Δt を速度に対して十分に小さく取らなければシミュレーション結果は全く役に立たないものとなるので注意が必要である。

次に上記のシミュレーション結果と理論解の比較を考える。この比較は本シミュレーションの計算精度を知る上で重要であるが、任意の状態の電場及び磁場に対する理論解を得ることは非常に難しいので、特定の条件下での運動について考える。

用いる条件は下記の通りである。このような条件下での荷電粒子の運動は、色々な教科書や Web ページに詳細が記されているが一応確認のため記す。

$$\mathbf{E}(\mathbf{r}) = (E_x, E_y, 0), \mathbf{B}(\mathbf{r}) = (0, 0, B_z)$$

この条件下での運動方程式を成分毎に記すと次のようになる。

$$m \frac{d^2 x}{dt^2} = q \left\{ E_x + B_z \frac{dy}{dt} \right\}$$

$$m \frac{d^2 y}{dt^2} = q \left\{ E_y - B_z \frac{dx}{dt} \right\}$$

$$m \frac{d^2 z}{dt^2} = 0$$

ここで、 $v_x = \frac{dx}{dt}, v_y = \frac{dy}{dt}$ と置くと、x,y 成分に関する運動方程式は $m \frac{dv_x}{dt} = q \{ E_x + B_z v_y \}$ 及び

$$m \frac{dv_y}{dt} = q \{ E_y - B_z v_x \} \text{ と書き換えられ、それぞれ変形すると } v_y = \frac{1}{B_z} \left(\frac{m}{q} \frac{dv_x}{dt} - E_x \right) \text{ 及び}$$

$$v_x = -\frac{1}{B_z} \left(\frac{m}{q} \frac{dv_y}{dt} - E_y \right) \text{ となる。さらにこの 2 式を時間微分すると、}$$

$$\frac{dv_y}{dt} = \frac{d}{dt} \left\{ \frac{1}{B_z} \left(\frac{m}{q} \frac{dv_x}{dt} - E_x \right) \right\} = \frac{1}{B_z} \left(\frac{m}{q} \frac{d^2v_x}{dt^2} - \frac{dE_x}{dt} \right) = \frac{m}{qB_z} \frac{d^2v_x}{dt^2}$$

$$\frac{dv_x}{dt} = \frac{d}{dt} \left\{ -\frac{1}{B_z} \left(\frac{m}{q} \frac{dv_y}{dt} - E_y \right) \right\} = -\frac{1}{B_z} \left(\frac{m}{q} \frac{d^2v_y}{dt^2} - \frac{dE_y}{dt} \right) = -\frac{m}{qB_z} \frac{d^2v_y}{dt^2}$$

が得られ、先の $m \frac{dv_x}{dt} = q \{ E_x + B_z v_y \}$ 及び $m \frac{dv_y}{dt} = q \{ E_y - B_z v_x \}$ を用いて式を整理すると次の常微分方程式が得られる。

$$\frac{d^2v_x}{dt^2} + \left(\frac{qB_z}{m} \right)^2 v_x = \left(\frac{qB_z}{m} \right)^2 \frac{E_y}{B_z}$$

$$\frac{d^2v_y}{dt^2} + \left(\frac{qB_z}{m} \right)^2 v_y = - \left(\frac{qB_z}{m} \right)^2 \frac{E_x}{B_z}$$

この二階の線形常微分方程式の解法には色々な方法があるが、まず右辺を0とした同次線形微分方程式の場合の解を求めた後、特殊解を求める。

まず解の形を $f(t) = A \cos(\omega t + \theta)$ とおくと $\frac{d^2f(t)}{dt^2} + \omega^2 f(t) = 0$ となるので、これを v_x, v_y に当てはめると $\omega = \frac{qB_z}{m}$ となる。次に特殊解について考えるが、これは右辺が定数であることからそのまま右辺の値を特殊解として求める式に含めれば方程式が成立することが分かる。従って速度成分の一般解は次のように表される。

$$v_x(t) = A_x \cos \left(\frac{qB_z}{m} t + \theta_x \right) + \frac{E_y}{B_z}$$

$$v_y(t) = A_y \cos \left(\frac{qB_z}{m} t + \theta_y \right) - \frac{E_x}{B_z}$$

$$v_z(t) = v_z(0)$$

上記の方程式にはそれぞれ未知定数が4つ含まれているが、 $t=0$ の時の運動方程式及び初期条件

$$v_x(0) = A_x \cos \theta_x + \frac{E_y}{B_z}, \quad v_y(0) = A_y \cos \theta_y - \frac{E_x}{B_z}, \quad A_x \cos \theta_x = A_y \sin \theta_y$$

を用いることによって、それぞれ次のように求めることができる。

$$A_x = A_y = \sqrt{\left(v_x(0) - \frac{E_y}{B_z} \right)^2 + \left(v_y(0) + \frac{E_x}{B_z} \right)^2}, \quad \theta_x = \cos^{-1} \left(\frac{v_x(0) - E_y/B_z}{A_x} \right), \quad \theta_y = \theta_x + \frac{\pi}{2}$$

以上から任意の時刻における速度成分が求められたので、これらの積分及び初期条件から座標を求めると次のようになる。

$$x(t) = \frac{mA}{qB_z} \sin \left(\frac{qB_z}{m} t + \theta \right) + \frac{E_y}{B_z} t + x(0) - \frac{mA}{qB_z} \sin \theta$$

$$y(t) = \frac{mA}{qB_z} \cos \left(\frac{qB_z}{m} t + \theta \right) - \frac{E_x}{B_z} t + y(0) - \frac{mA}{qB_z} \cos \theta$$

$$z(t) = v_z(0)t + z(0)$$

$$\ast A = \sqrt{\left(v_x(0) - \frac{E_y}{B_z} \right)^2 + \left(v_y(0) + \frac{E_x}{B_z} \right)^2}, \quad \theta = \cos^{-1} \left(\frac{v_x(0) - E_y/B_z}{A} \right)$$

この理論解とシミュレーション結果の比較には、同時刻における各粒子の距離を用いる。

3. 方法(Method)

前章で示した理論を基にシミュレーションプログラムで具体的に作り上げる。

3. 1. プログラムの仕様

本プログラムを作成する上で軸となる事柄を下記に示す。

- 任意の数の荷電粒子が設定可能である事。
- 計算対象となっている荷電粒子同士の相互作用は考慮しない。
- 電磁場の設定・保守が容易である事。
- 粒子の運動計算において、タイムステップ毎の粒子の移動距離を一定にする機能（可変タイムステップ機能）を作る。
- 点電荷及び電流片を用いた電磁場の設定が可能である事。
- 点電荷及び電流片によって電磁場を設定する場合にはクーロンの法則及びビオ・サバルの法則を用いる。
- エラー発生時の対処が容易になるようログファイルの出力を行う。

3. 2. 各関数の関係

作成したプログラムを構成する関数を表3. 2. 1、グローバル変数を表3. 2. 2に示す。また、各関数の関係を図3. 2. 1に示す。

表3. 2. 1 荷電粒子の運動計算プログラムを構成する関数群

番号	関数宣言
1	int main(void)
基本関数	
2	void init_setting(void)
3	void particle_init(int PTM)
4	void motion_calc(int FMN,int VTS,double TS,int PTN,int ETS)
5	void EB_field(int FMN,double brx,double bry,double brz,double *bex,double *bey, double *bez,double *bbx,double *bby,double *bbz)
6	void EB_source_calc(int etotal,int bttotal,double brx,double bry,double brz,double **bex, double **bey,double **bez,double **bbx,double **bby,double **bbz)
計算結果出力用関数	
7	void source_element_output(void)
8	void EB_field_output(void)
9	void single_particle_trace(int PTN)
10	void particle_animation(int ETS)
11	void comparison_of_solutions(void)
点電荷・電流片作成モデル関数	
12	void quadrupole_electromagnet_SM(int *total)
13	void solenoidal_coil_01(int max,int *total,double BAX[],double BAY[],double BAZ[])
14	void toroidal_coil_SM(int *total)
15	void toroidal_coil(int max,int *total,double BAX[],double BAY[],double BAZ[])
16	void annular_current_01_SM(int *total)
17	void regular_icosahedron_SM(int *total)

表 3. 2. 2 グローバル変数

番号	変数名	役割
1	MAX_TS	逐次演算結果の最大管理数 (#define により指定)
2	MAX_PN	計算粒子の最大管理数 (#define により指定)
3	MAX_SE	点電荷・電流片の最大管理数 (#define により指定)
4	*fp	ログファイル用ファイルポインタ
5	cal_mod	計算を行う電磁場のモデル番号
6	variable_ts	可変タイムステップ有無の指定 ON(1)/OFF(0)
7	particle_num	計算対象の粒子数
8	se_Eflag	点電荷の設定数 (ログ書き込み時のフラグも兼ねる)
9	se_Bflag	電流片の設定数 (ログ書き込み時のフラグも兼ねる)
10	error_flag	エラーフラグ
11	ts_val	微小時間(variable_ts=0)または移動距離(variable_ts=1)
12	ts[][]	各タイムステップ毎の微小時間
13	pop[][]	荷電粒子のパラメータ ([X][0]: 質量, [X][1]: 電荷量)
14	rx[], ry[], rz[]	荷電粒子の位置
15	vx[], vy[], vz[]	荷電粒子の速度
16	ax[], ay[], az[]	荷電粒子の加速度
17	seev[]	電場を構成する点電荷の電荷量
18	seep[][]	電場を構成する点電荷の位置
19	sebv[][]	磁場を構成する電流片の電流ベクトル
20	sebp[][]	磁場を構成する電流片の位置

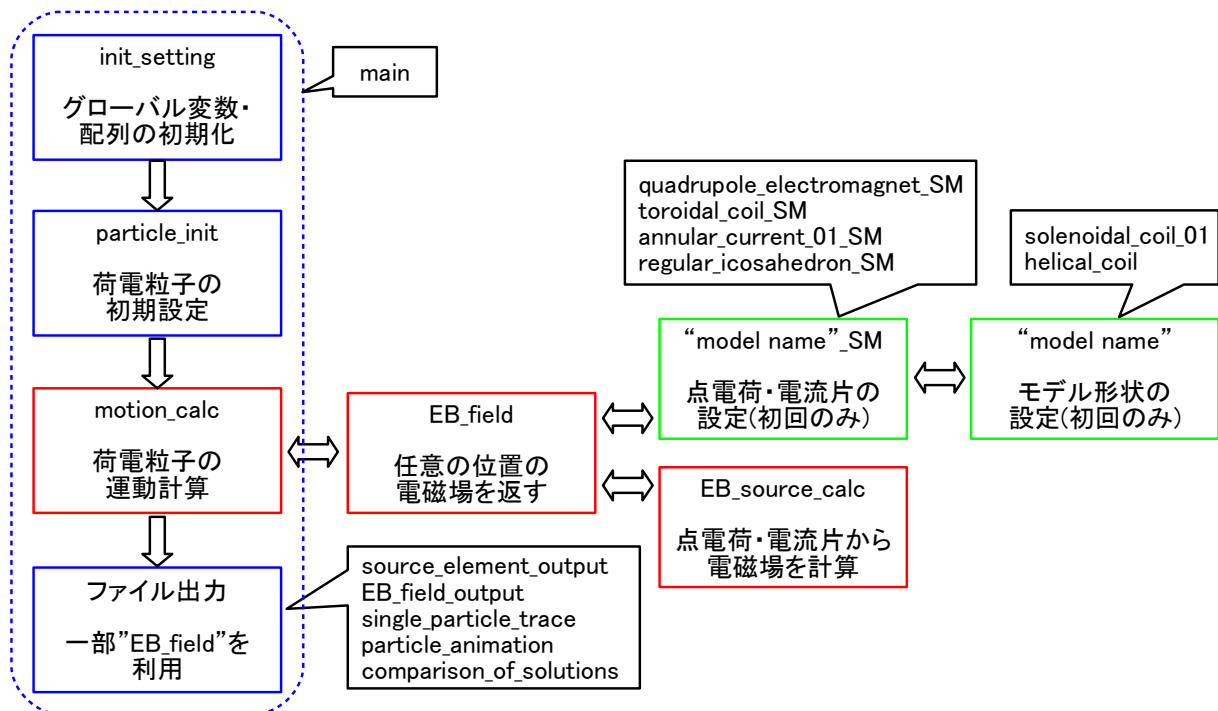


図 3. 2. 1 各関数の関係

本プログラムは、開始と同時にログファイルの作成とグローバル変数・配列の初期化(`init_setting`)を行い、その後 `main` 関数内で設定されたモデル設定値(`cal_mod`)を基に荷電粒子、電磁場の初期設定 (`particle_init, EB_field`)が実施される。電磁場の初期設定について、計算モデルによっては図 3. 2. 1 中の緑色枠のモデル関数によって点電荷・電流片の作成が行われる。

荷電粒子の計算は `motion_calc` によって逐次行われ、その都度 `EB_field` が呼び出され、指定された座標に対する電磁場が計算される。なお、点電荷・電流片によって電磁場が表現される場合は `EB_source_calc` が `EB_field` を経由して呼び出される。(図 3. 2. 1 中の赤色枠が何度も実行される事により荷電粒子の運動計算が行われる。)

運動計算を行う `motion_calc` では、`main` 関数で設定された `variable_ts` が "0" の時はタイムステップ毎の微小時間を等しいものとして計算し、`variable_ts` が "1" の時にはタイムステップ毎の荷電粒子の移動間隔がほぼ等しくなるように計算が行われる。`ts_val` には微小時間または微小距離の値を設定しなければならない。その後、運動計算が終了すると各種出力関数によって、粒子軌道等が `csv` ファイルとして出力される。

本プログラムにおいて新しく荷電粒子の設定を追加する場合は、`particle_init` 関数内の `switch-case` 文に新しい `case` 文を付け加え、新しく電磁場モデルを追加する場合には `EB_field` 関数内にも同様に新しい `case` 文を付け加える。なお、この時各関数の `case` 文の番号とグローバル変数の `cal_mod` を対応させると後々便利である。

3. 3. 荷電粒子の初期設定 (作成済みモデルの説明)

荷電粒子の初期設定ではタイムステップが 0 の時の位置・速度及び荷電粒子の質量・電荷の値を設定する。荷電粒子の質量の設定には予め設定された電子、陽子、中性子の質量(`me, mp, mn`)を用いる事ができ、電荷量の設定には素電荷(`ec`)を用いる事ができる。

本プログラムで作成した荷電粒子の初期設定は以下の 4 つである。

- ・テスト用粒子(`case 0`) : 任意の位置・速度の荷電粒子を 1 つ作る。
- ・チューブ型粒子群(`case 1`) : x 軸に対して垂直な円板上に荷電粒子を同心円状に配置する。
(次節 `case 1, 4~7` 用)
- ・整列型粒子群(`case 2`) : y 軸上に荷電粒子を配置する。(サイクロトロン用)
- ・円形放射型粒子群(`case 3`) : 原点を中心とした x-y 平面上の円周上に荷電粒子を配置する。
(マグネトロン用)

(1) テスト用粒子(`case 0`)

テスト用粒子(`case 0`)はテスト用の荷電粒子を 1 つ作成する。`particle_num` は "1" となる。

`pop[0][0]` : 質量、`pop[0][1]` : 電荷、
`rx[0][0], ry[0][0], rz[0][0]` : 初期位置、
`vx[0][0], vy[0][0], vz[0][0]` : 初速

(2) チューブ型粒子群(case 1)

チューブ型粒子群(case 1)はx軸に垂直な円板上に荷電粒子を配し、x軸方向に同じ初速を与えた粒子群を作成する。「チューブ型」の由来は電磁場が存在しない場合の粒子群の軌跡がチューブ状になることから、このように命名した。

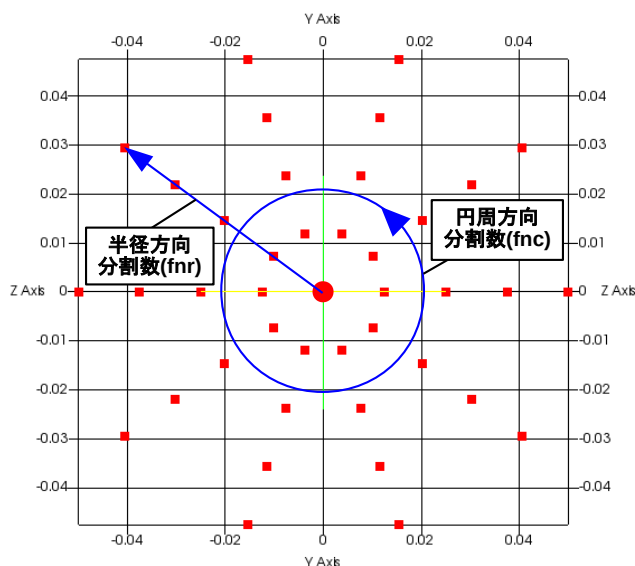


図3. 3. 1 荷電粒子の配置(case 0,fnr=4,fnc=10,set6=0.05)

設定パラメータは次の通りである。なお、円板の中心に電荷は配置されないため `particle_num` は"`fnr×fnc`"となる。

`fnr` : 円板の半径方向の分割数、`fnc` : 円周方向の分割数、
`set0` : 質量、`set1` : 電荷、`set2~set4` : 円板の中心位置(x,y,z)、`set5` : x軸方向の初速、
`set6` : 円板の半径

(3) 整列型粒子群(case 2)

整列型粒子群(case 2)はサイクロトロン用の荷電粒子群で、y軸上に等間隔に粒子を配置するものである。次節で紹介するサイクロトロンモデルではy軸上に加速電場領域が存在するため、運動計算の開始と同時に粒子の加速が行われる。設定パラメータについては下記の通りで `particle_num` は"`fnr`"となる。

`fnr` : `set2,set3` によって設定されるy軸上の閉区間の分割数、
`set0` : 質量、`set1` : 電荷、
`set2` : 内円筒の半径(<`set3`)、`set3` : 外円筒の半径(>`set2`)

(4) 円形放射型粒子群(case 3)

円形放射型粒子群(case 3)はマグネトロン用の荷電粒子群で、原点を中心としたx-y平面の円周上に粒子を配置したものである。設定パラメータについては下記の通りで `particle_num` は"`fnc`"となる。

`fnc` : 円周の分割数、
`set0` : 質量、`set1` : 電荷、
`set2` : 円(内円筒)の半径

3. 4. 電磁場の初期設定（作成済みモデルの説明）

電磁場の初期設定では main 関数で設定された変数(cal_mod)に対応する電磁場モデル構成される。特に点電荷・電流片の集合によって電磁場が構成される場合は、対応するモデル関数によって初回実行時にグローバル変数(seep[[[]],seev[],sebp[[[]],sebp[[[]]])に必要な値が設定される。

本プログラムで作成した電磁場モデルは以下の 8 つである。

- ・ 一様電磁場(case 0) : 場所に依存しない一様電磁場モデル。
- ・ 簡易電子銃(case 1) : x 軸上を運動する荷電粒子を加速した後、y,z 軸方向の磁場をかけて粒子軌道を曲げるモデル。
- ・ サイクロトロン(case 2) : 円板状の領域に一様磁場と加速用電場領域を持つモデル。
- ・ マグネトロン(case 3) : 円柱状の領域に一様磁場と同心円状に広がる電場を持つモデル。
- ・ 四重極電磁石(case 4) : ソレノイドコイルを 4 つ x 軸に対して対称に配置したモデル。
- ・ トロイダルコイル(case 5) : トロイダルコイルを配置したモデル。
- ・ 模擬地磁気(case 6) : トーラス状の電流と原点に点電荷を配置したモデル。
- ・ 球面配置点電荷群(case 7) : 正二十面体の 12 箇所の頂点に点電荷を配置したモデル。

(1) 一様電磁場(case 0)

一様電磁場(case 0)では与えられた座標に関わらず、設定された電磁場を返却するもので、設定パラメータの set0~set2 は $E_x \sim E_z$ 、set3~set5 は $B_x \sim B_z$ に対応する。

(2) 簡易電子銃(case 1)

簡易電子銃(case 1)は円柱状の領域に x 軸方向の電場と y,z 軸方向の磁場を持つモデルである。

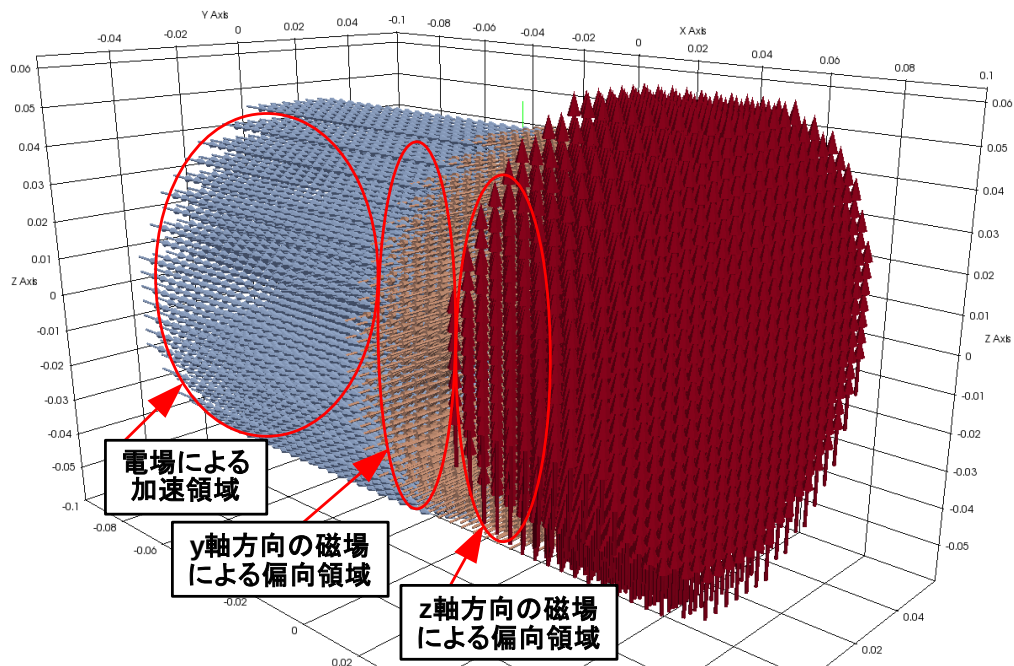


図 3. 4. 1 簡易電子銃モデル

設定するパラメータは次の 6 つである。

- ・ 印加電圧(set0) : 電場の算出に使用。($E_x = \text{set0}/\text{set4}$)
- ・ y 軸方向の磁場(set1) : 磁場の設定。(B_y)
- ・ z 軸方向の磁場(set2) : 磁場の設定。(B_z)
- ・ 電子銃の管の半径(set3) : 電子銃モデルの大きさの設定。
- ・ 加速領域の長さ(set4) : 電子銃モデルの大きさの設定及び電場の算出に使用。
- ・ 磁場領域の長さ(set5) : 電子銃モデルの大きさの設定。

(3) サイクロトロン(case 2)

サイクロトロン(case 2)は円形領域に z 軸方向の一様磁場と y 軸近傍に加速電場領域を持つモデルである。

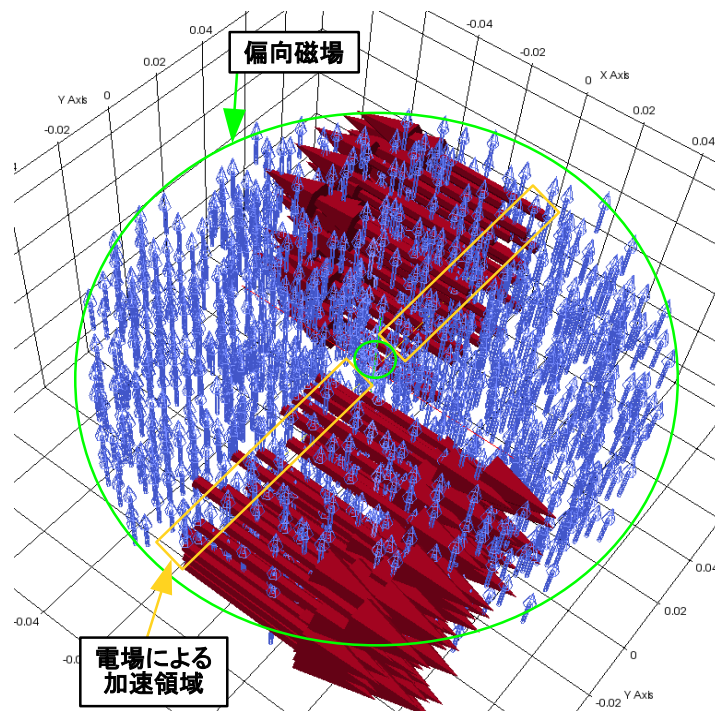


図 3. 4. 2 サイクロトロンモデル

設定するパラメータは次の 6 つである。

- ・印加電圧(set0) : 電場の算出に使用。(E_x=set0/set4)
- ・内円筒の半径(set1) : サイクロトロンモデルの大きさの設定。
- ・外円筒の半径(set2) : サイクロトロンモデルの大きさの設定。
- ・厚み(set3) : サイクロトロンモデルの大きさの設定。
- ・加速領域の長さ(set4) : 電場の算出に使用。
- ・z 軸方向の磁場(set5) : 磁場の設定。(B_z)

(4) マグネトロン(case 3)

マグネトロン(case 3)は電圧が印加された無限に長い円筒の一部を切り出したものにz軸方向の磁場を加えたモデルである。

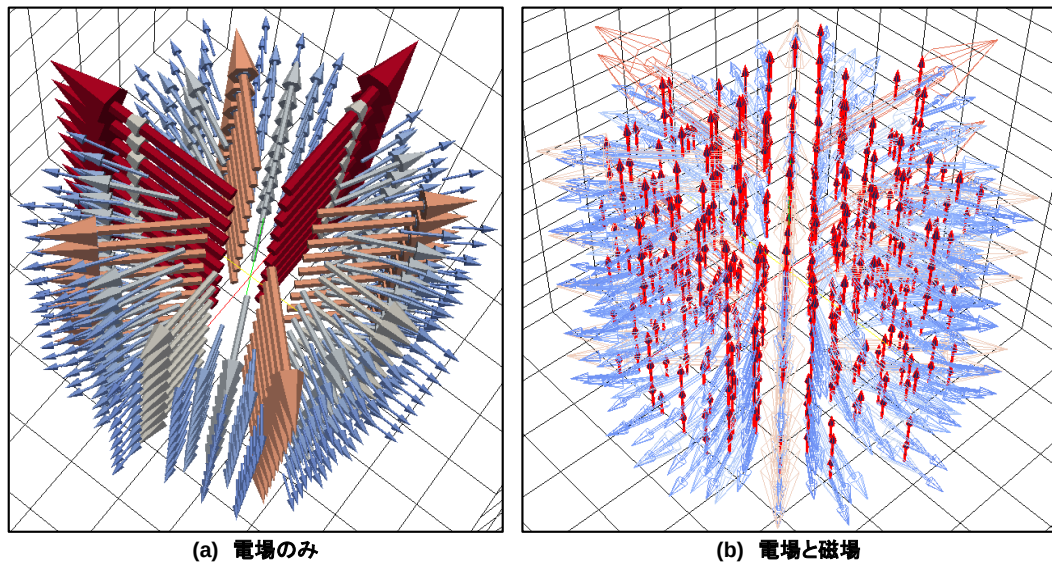


図3. 4. 3 マグネトロンモデル

設定するパラメータは次の5つである。

- ・ 印加電圧(set0) : 電場の算出に使用。
- ・ 内筒の半径(set1) : マグネトロンモデルの大きさ及び電場の算出に使用。
- ・ 外筒の半径(set2) : マグネトロンモデルの大きさ及び電場の算出に使用。
- ・ 高さ(set3) : マグネトロンモデルの大きさの設定。
- ・ z軸方向の磁場(set4) : 磁場の設定。(B_z)

このモデルでは電磁気学の演習問題によく出る二次元軸対称電場を用いているので、座標(x,y)における電場は次のように表される。なお、Vは印加電圧(set0)、 r_i, r_o は内筒、外筒の半径である。

$$E_x = \frac{V}{\ln(r_o/r_i)} \frac{x}{x^2+y^2} \quad E_y = \frac{V}{\ln(r_o/r_i)} \frac{y}{x^2+y^2}$$

(5) 四重極電磁石(case 4)

四重極電磁石(case 4)はソレノイドコイル（※別関数で作成される）をx軸に対して対称になるように配置したモデルで、4つのコイルそれぞれに異なる電流値の設定が可能である。

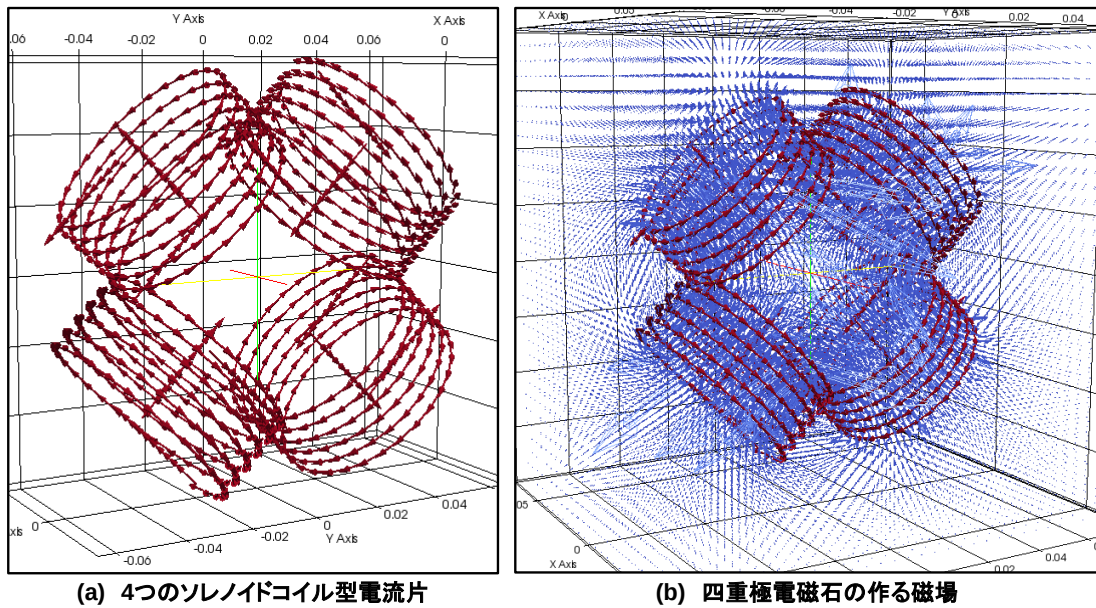


図 3. 4. 4 四重極電磁石モデル

設定するパラメータは次の5つである。

- ・内接円筒の半径(flt) : モデルの大きさの設定。
- ・コイル1の電流(fcv0) : コイル1($y < 0, z > 0$)の電流値の設定に使用。
- ・コイル2の電流(fcv1) : コイル2($y < 0, z < 0$)の電流値の設定に使用。
- ・コイル3の電流(fcv2) : コイル3($y > 0, z < 0$)の電流値の設定に使用。
- ・コイル4の電流(fcv3) : コイル4($y > 0, z > 0$)の電流値の設定に使用。

また、ソレノイドコイル(solenoidal_coil_01)の設定パラメータは次の6つである。

- ・給電線の半径方向の分割数(fnr) : 電流片の作成に使用。
- ・コイル一周の分割数(fnc) : 電流片の作成に使用。
- ・コイルの巻数(fnt) : モデルの大きさの設定。
- ・戻り線の分割数(fnb) : 電流片の作成に使用。
- ・コイルの半径(flrr) : モデルの大きさの設定。
- ・コイルのピッチ(fdl) : モデルの大きさの設定。

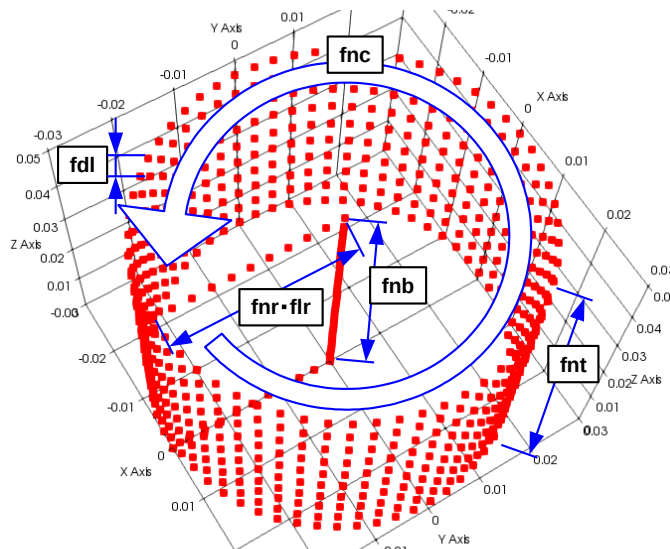


図 3. 4. 5 ソレノイドコイル形状モデル

(6) トロイダルコイル(case 5)

トロイダルコイル(case 5)は別関数(`toroidal_coil`)で作成されたトロイダルコイルに定電流を流したモデルである。

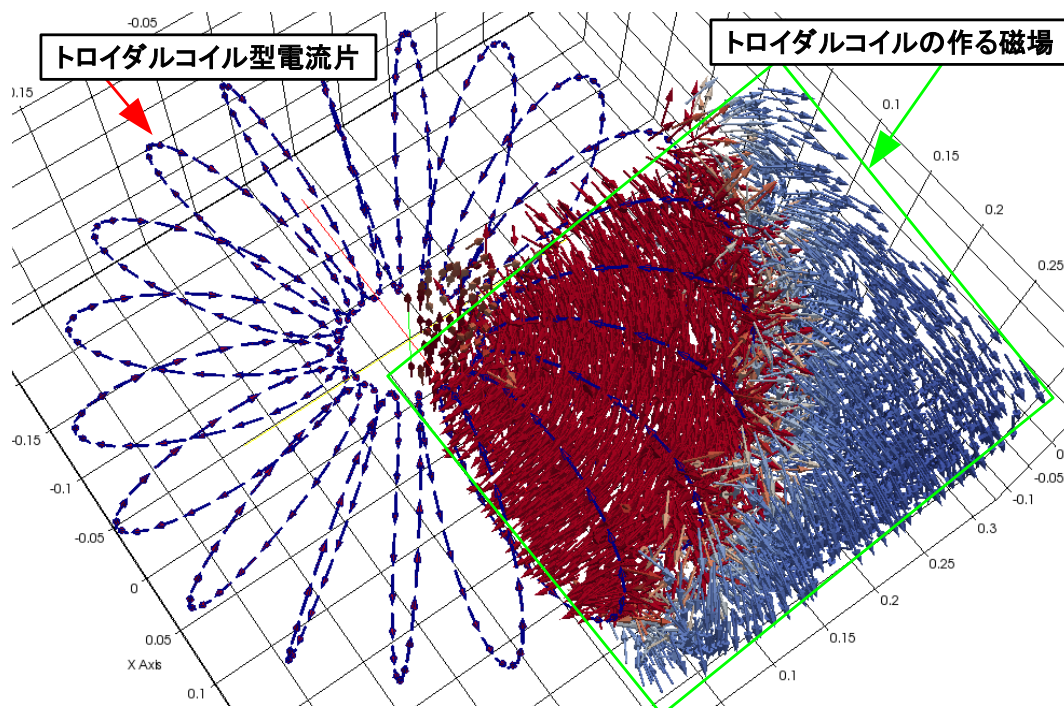


図 3. 4. 6 トロイダルコイルモデル

設定するパラメータは次の電流値のみである。

- ・コイルの電流(`fcv`) : トロイダルコイルの電流値。

また、トロイダルコイル(`toroidal_coil`)の設定パラメータは次の4つである。

- ・コイル一周の分割数(`fnc`) : 電流片の作成に使用。
- ・コイルの巻数(`fnt`) : 電流片の作成に使用。
- ・大半径(`flt`) : モデルの大きさの設定。
- ・小半径(`fld`) : モデルの大きさの設定。

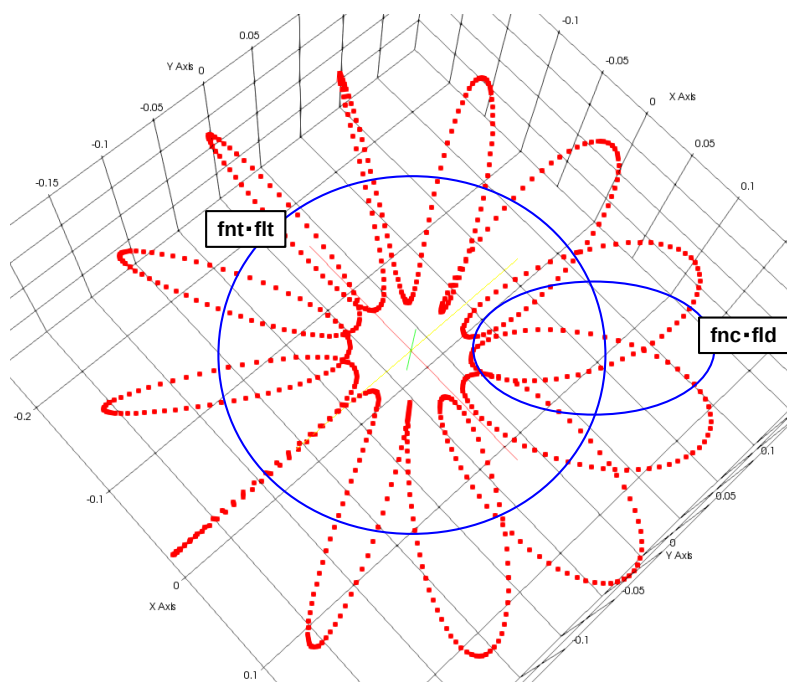


図 3. 4. 7 トロイダルコイル形状モデル

(7) 模擬地磁気(case 6)

模擬地磁気(case 6)はトーラス状の領域の内部に一様電流が流れる環流電流モデルと重力を模擬するための点電荷で構成される。

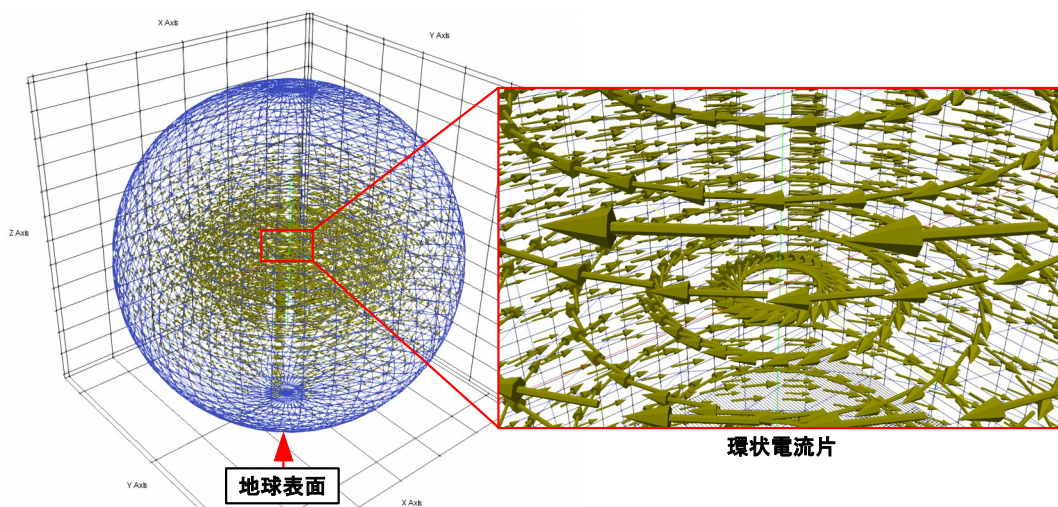


図 3. 4. 8 環流電流モデル

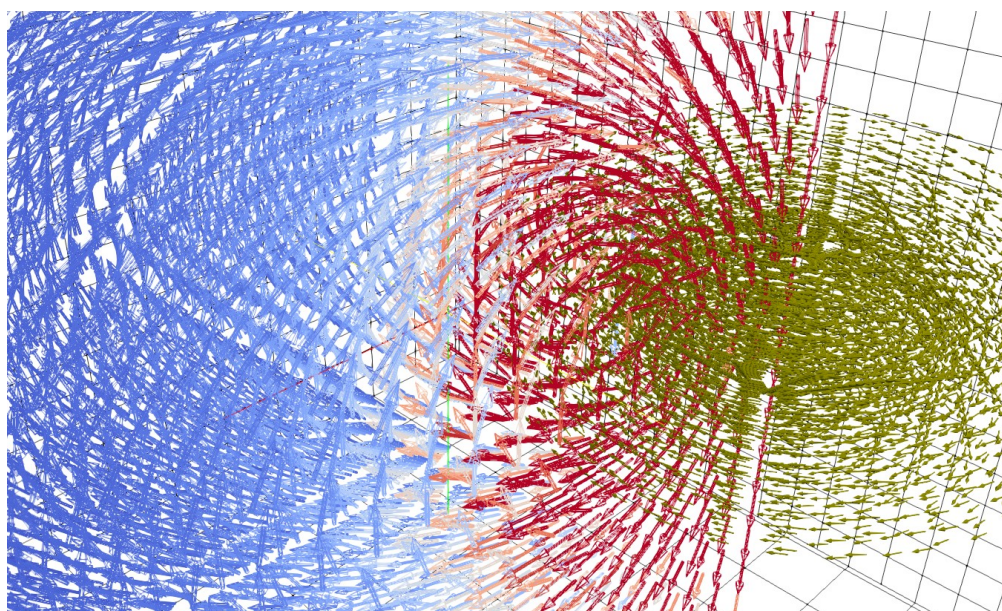


図 3. 4. 9 環流電流モデルの作る磁場

設定するパラメータは次の6つである。

- ・小円の半径の分割数(fnr) : 電流片の作成に使用。
- ・小円の角度の分割数(fnc) : 電流片の作成に使用。
- ・大円の分割数(fnt) : 電流片の作成に使用。
- ・電流片の電流値(fcv) : 電流片の作成に使用。
- ・大半径(flt) : モデルの大きさの設定。
- ・小半径(fld) : モデルの大きさの設定。

また重力を模擬するために用いられる点電荷はEB_field関数内のcase文の中で設定される。

以上の模擬地磁気及び模擬重力を用い、ヴァン・アレン帯が形成されるかを確認するための重要なパラメータはおおよそ次の通りである。

電流片の電流値(fcv) : $-3.0 \times 10^9 / (\text{fnr} \times \text{fnc})$

模擬重力用点電荷 (自動計算) : $Q = -4 \pi \epsilon_0 G M_E m_p / q_p$

大半径(flt)・小半径(fld) : 3.0×10^6

(8) 球面配置点電荷群(case 7)

球面配置点電荷群(case 7)は正二十面体の12個の頂点に点電荷を配置したモデルである。

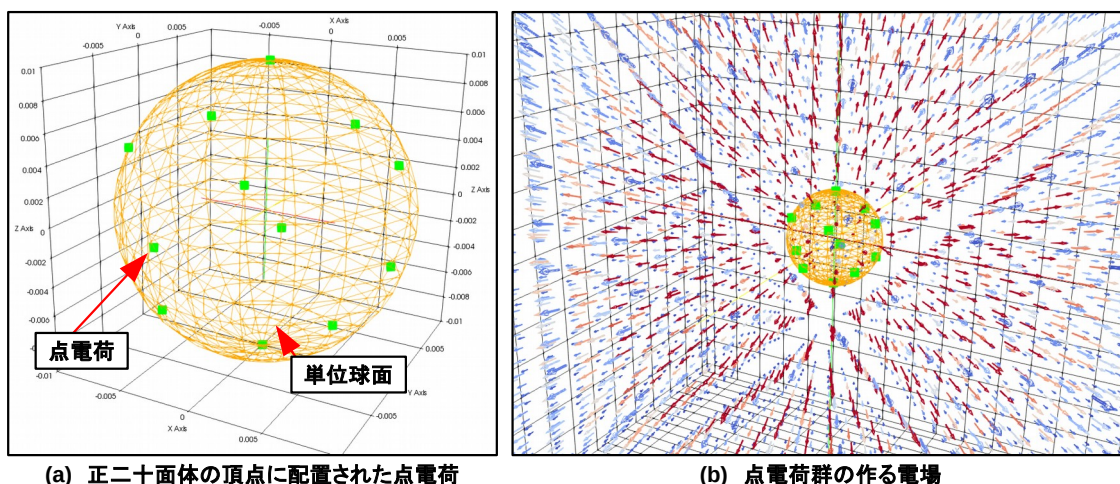


図 3. 4. 1 0 球面配置点電荷群の作る電場

設定するパラメータは次の2つである。

- ・点電荷の電荷量(fpc) : 電荷の設定。
- ・外接球の半径(fsr) : モデルの大きさの設定。

3. 5. 各関数の詳細説明

各関数の仕様を以下に示す。

関数名	main
役割	プログラムの全体の流れを制御する。
引数	なし
返却値	"0"
参照する変数	error_flag, particle_num, MAX_TS
書換える変数	fp, cal_mod, variable_ts, ts_val
備考	ログファイルの作成・終了を行う。

関数名	init_setting
役割	グローバル変数の初期化を行う。
引数	なし
返却値	なし
参照する変数	MAX_PN, MAX_TS, MAX_SE
書換える変数	cal_mod, particle_num, variable_ts, se_Eflag, se_Bflag, error_flag, ts_val, ts[], rx[], ry[], rz[], vx[], vy[], vz[], ax[], ay[], az[], pop[], seev[], seep[], sebv[], sebp[]
備考	ログファイルへの書き込みあり。

関数名	particle_init
役割	荷電粒子の初期条件（初期位置、初速及び質量、電荷量）の設定を行う。
引数	PTM (ParTicle Mode) : 通常 main 関数で設定された cal_mod が渡される。
返却値	なし
参照する変数	MAX_PN
書換える変数	particle_num, rx[], ry[], rz[], vx[], vy[], vz[], pop[], error_flag
備考	ログファイルへの書き込みあり。

関数名	motion_calc
役割	荷電粒子の位置、速度及び加速度に対して逐次演算を行う。
引数	FMN (Field Model Number) : 通常 main 関数で設定された cal_mod が渡される。 VTS (Variable Time Step) : 可変タイムステップ機能、0:off, 1:on 通常 main 関数で設定された variable_ts が渡される。 TS (Time Step) : VTS=0 ならタイムステップ(s)、VTS=1 なら距離(m) 通常 main 関数で設定された ts_val が渡される。 PTN (ParTicle Number) : 計算対象の粒子番号。 ETS (End Time Step) : 計算終了のタイムステップ。
返却値	なし
参照する変数	MAX_PN, MAX_TS, pop[][]
書換える変数	ts[], rx[], ry[], rz[], vx[], vy[], vz[], ax[], ay[], az[], error_flag
備考	EB_field を逐次呼び出す。 VTS=1 で速度の大きさが 0 の時は、仮の微小時間として dt=1.0e-15(s)が設定されている。

関数名	EB_field
役割	与えられた座標に対応する電磁場の計算を行う。
引数	FMN (Field Model Number) : 通常 main 関数で設定された cal_mod が渡される。 brx, bry, brz : 座標の x,y,z 成分。 &bex, &bey, &bez : 電場の各成分の格納先のアドレス。 &bbx, &bbz : 磁場の各成分の格納先のアドレス。
返却値	なし
参照する変数	se_Eflag, se_Bflag, pop[][] (特例 case6)
書換える変数	se_Eflag, se_Bflag, seev[] (特例 case6) , error_flag
備考	初回実行時にログファイルへの書き込みあり。 計算モデルによっては EB_source_calc 関数を呼び出す。 点電荷・電流片を用いたモデルを新しく作成する場合は、別途モデル関数を作成する事が推奨される。

関数名	EB_source_calc
役割	点電荷・電流片により構成される電磁場の計算を行う。
引数	etotal : 点電荷の数。通常 EB_field からの呼び出し時に設定された se_Eflag が渡される。 btotal : 電流片の数。通常 EB_field からの呼び出し時に設定された se_Bflag が渡される。 brx, bry, brz : 座標の x,y,z 成分。 &bex, &bey, &bez : 電場の各成分の格納先のアドレス。 &bbx, &bby, &bbz : 磁場の各成分の格納先のアドレス。
返却値	なし
参照する変数	seev[], seep[[]], sebv[[]], sebp[[]]
書換える変数	なし
備考	点電荷についてはクーロンの法則、電流片についてはビオ・サバールの法則を用いる。 与えられた座標と点電荷・電流片の座標が等しい場合は、その要素については計算に含めないものとする。

関数名	source_element_output
役割	設定された点電荷・電流片の位置と大きさを入力する。
引数	なし
返却値	なし
参照する変数	se_Eflag, se_Bflag, seev[], seep[[]], sebv[[]], sebp[[]]
書換える変数	なし
備考	ログファイルへの書き込みあり。

関数名	EB_field_output
役割	この関数の中で設定された範囲(fs*,fe*)・分割幅(fd*)で作られる格子点上の電磁場を出力する。
引数	なし
返却値	なし
参照する変数	cal_mod
書換える変数	なし
備考	ログファイルへの書き込みあり。 EB_field 関数の逐次呼び出し。

関数名	single_particle_trace
役割	引数で指定された荷電粒子の軌道を入力する。
引数	PTN (ParTicle Number) : 出力対象の粒子番号。
返却値	なし
参照する変数	MAX_PN, MAX_TS, ts[[]], rx[[]], ry[[]], rz[[]]
書換える変数	error_flag
備考	ログファイルへの書き込みあり。

関数名	particle_animation
役割	全荷電粒子の位置をタイムステップ毎にファイルを分けて出力する。
引数	ETS (End Time Step) : 出力終了時のタイムステップ。
返却値	なし
参照する変数	MAX_TS, particle_num, variable_ts, rx[[[]], ry[[[]], rz[[[]]
書換える変数	error_flag
備考	ログファイルへの書き込みあり。

関数名	comparison_of_solutions
役割	条件付きの電磁場中（電場と磁場が直交）におけるシミュレーション結果と理論解の比較を行う。基本的に cal_mod が 0 の時、すなわち単一荷電粒子の運動シミュレーションで、電場の z 成分が 0 かつ磁場の z 成分のみが値をもつ電磁場のみで有効。
引数	なし
返却値	なし
参照する変数	MAX_TS, cal_mod, variable_ts, ts_val, rx[[[]], ry[[[]], rz[[[]], vx[[[]], vy[[[]], vz[[[]], pop[[[]]
書換える変数	error_flag
備考	ログファイルへの書き込みあり。

関数名	quadrupole_electromagnet_SM
役割	ソレノイドコイルの形状データを用い、四重極電磁石の電流片モデルを作成する。
引数	&total : 電流片の総数の格納先のアドレス。
返却値	なし
参照する変数	MAX_SE
書換える変数	sebv[[[]], sebp[[[]], error_flag
備考	ログファイルへの書き込みあり。 関数内でソレノイドコイルの形状データを受け取るための配列を宣言・使用。

関数名	solenoidal_coil_01
役割	ソレノイドコイルの形状データを作成する。
引数	max : 上位の関数で用意された配列の数。 &total : 点データの総数の格納先のアドレス。 BAX[], BAY[], BAZ[] : 点データの格納先のアドレス。
返却値	なし
参照する変数	なし
書換える変数	error_flag
備考	ログファイルへの書き込みあり。

関数名	toroidal_coil_SM
役割	トロイダルコイルの形状データを用い、同コイル型の電流片モデルを作成する。
引数	&total：電流片の総数の格納先のアドレス。
返却値	なし
参照する変数	MAX_SE
書換える変数	sebv[[[]], sebp[[[]], error_flag
備考	ログファイルへの書き込みあり。 関数内でトロイダルコイルの形状データを受け取るための配列を宣言・使用。

関数名	toroidal_coil
役割	トロイダルコイルの形状データを作成する。
引数	max：上位の関数で用意された配列の数。 &total：点データの総数の格納先のアドレス。 BAX[], BAY[], BAZ[]：点データの格納先のアドレス。
返却値	なし
参照する変数	なし
書換える変数	error_flag
備考	ログファイルへの書き込みあり。

関数名	annular_current_01_SM
役割	環流電流の電流片モデルを作成する。
引数	&total：電流片の総数の格納先のアドレス。
返却値	なし
参照する変数	MAX_SE
書換える変数	sebv[[[]], sebp[[[]], error_flag
備考	ログファイルへの書き込みあり。

関数名	regular_icosahedron_SM
役割	正二十面体の頂点に点電荷を配置したモデルを作成する。
引数	&total：点電荷の総数の格納先のアドレス。
返却値	なし
参照する変数	MAX_SE
書換える変数	seev[[[]], seep[[[]], error_flag
備考	ログファイルへの書き込みあり。

4. 結果(Results)

前章で作成したプログラムの実行結果を示す。なお、計算結果の数値的な比較はE×Bドリフトに対してのみ行い、他のモデルについては結果のみを示すことに留める。

4. 1. E×Bドリフト(case0)

まず始めに電場が無くz軸方向の磁場のみの空間で荷電粒子を運動させ、どのくらい計算結果が発散するのかを評価する。シミュレーションの条件は次の通りである。なお、この条件下における円運動の半径は約1.04(cm)である。

- 荷電粒子：陽子（質量：mp、電荷：ec） ※いずれもプログラム内で定義済み
- 初期位置：(0.0,0.0,0.0)、初速：(1.0×10⁶,0.0,0.0)(m/s)
- 磁場：1.0(T)
- タイムステップ(dt)：1.0×10⁻¹⁰(s), 1.0×10⁻⁹(s), 1.0×10⁻⁸(s) ※可変タイムステップ機能：OFF
- 実計算時間：1.0(s)

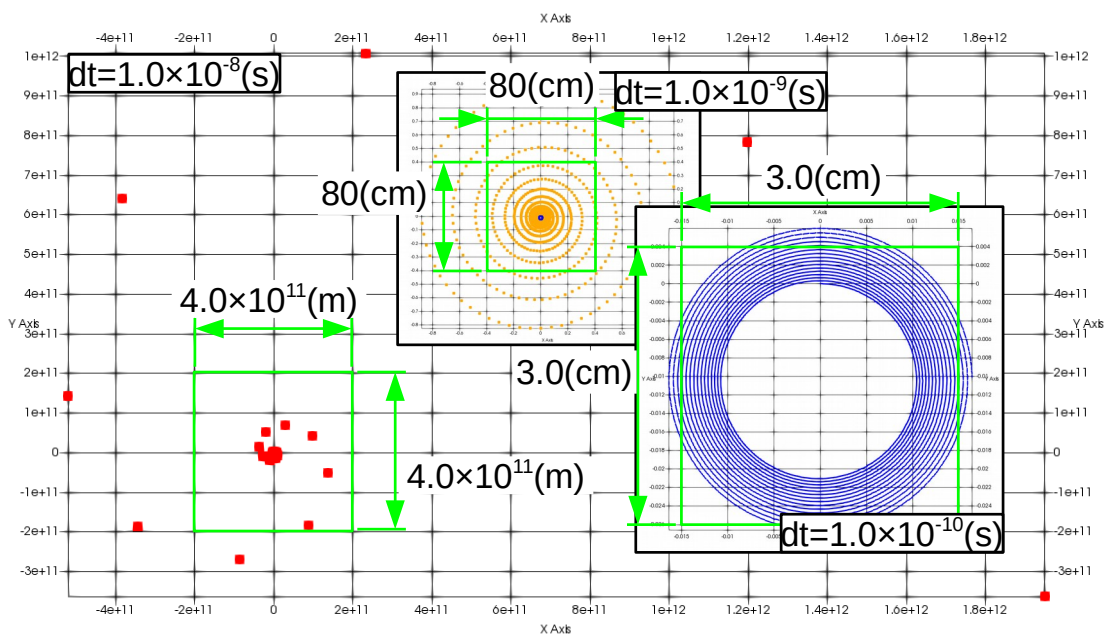


図4. 1. 1 タイムステップ毎の荷電粒子の軌跡

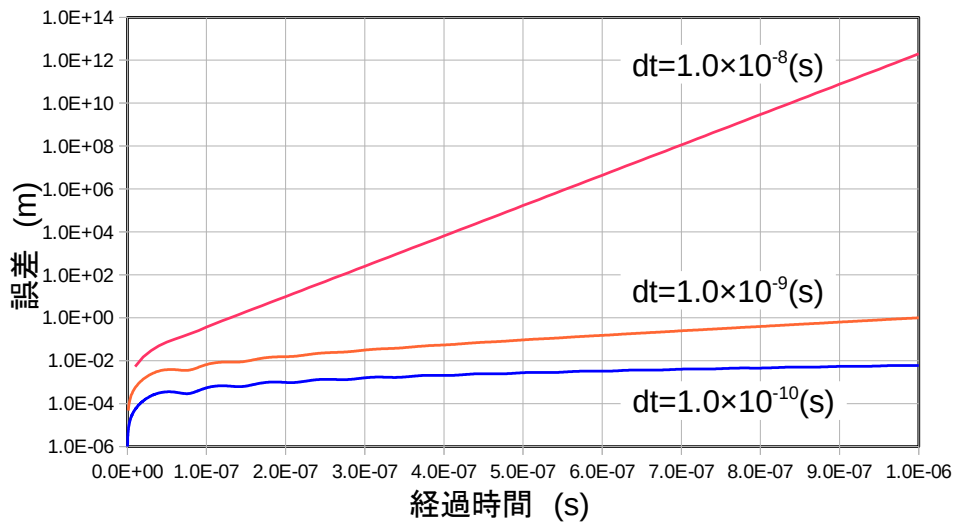


図4. 1. 2 タイムステップ毎の誤差の推移

図4. 1. 1に荷電粒子の軌跡を図4. 1. 2に横軸を実時間、縦軸を誤差としてグラフにしたものを示す。当然であるが、タイムステップ毎の時間幅が粒子の速度に対して十分小さくない場合、シミュレーション結果が発散していく事が分かる。

次にy軸方向に電場を加えた場合のシミュレーションを行う。タイムステップは $dt=1.0 \times 10^{-10}(s)$ とし他の条件は先のシミュレーションと同じである。また電場の条件は図のタイトルに示す通りである。

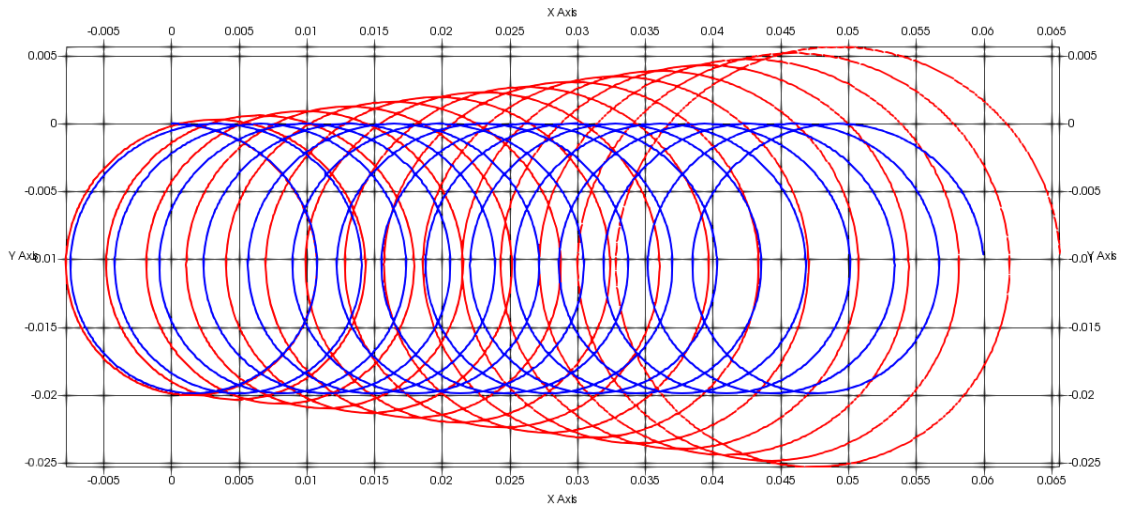


図4. 1. 3 $E \times B$ ドリフト $E=(0.0, 5.0 \times 10^4, 0.0)$

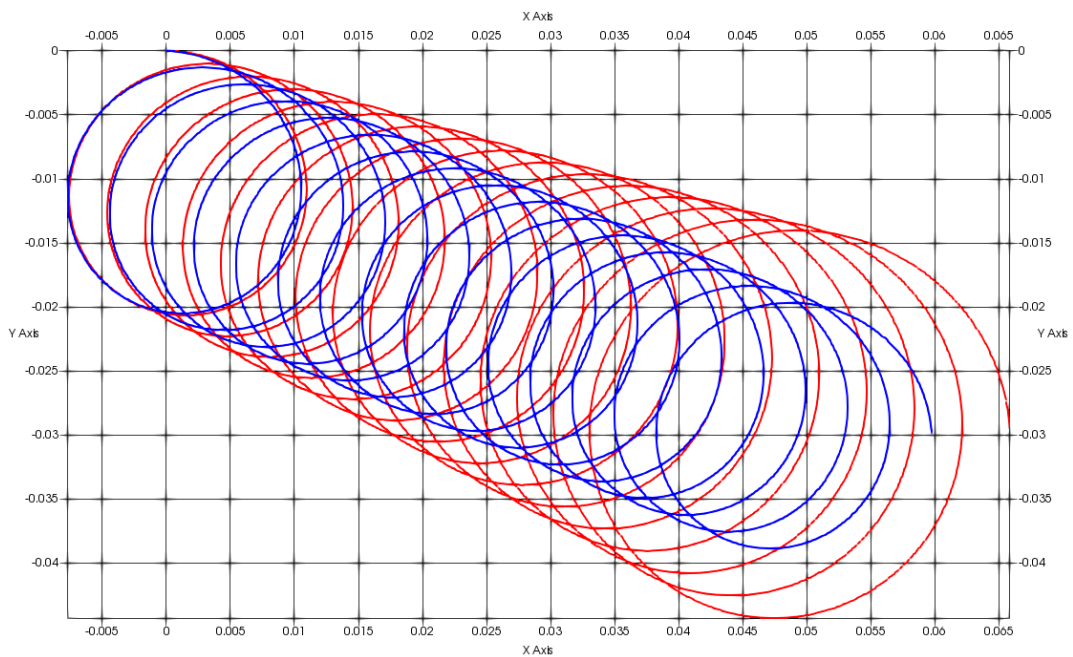


図4. 1. 4 $E \times B$ ドリフト $E=(2.0 \times 10^4, 5.0 \times 10^4, 0.0)$

図4. 1. 3及び図4. 1. 4の結果からシミュレーション結果が徐々に発散する傾向が見取れるが定性的には理論解とほぼ同じ傾向が得られた。 $E \times B$ ドリフトの運動では、直感的には電場に沿って螺旋運動を行いそうであるが、逐次演算によっても確かに電場に対して直交した方向に運動する事が分かる。

4. 2. 電子銃(case1)

電子銃モデルのシミュレーション結果とログファイルの一部を下記に示す。

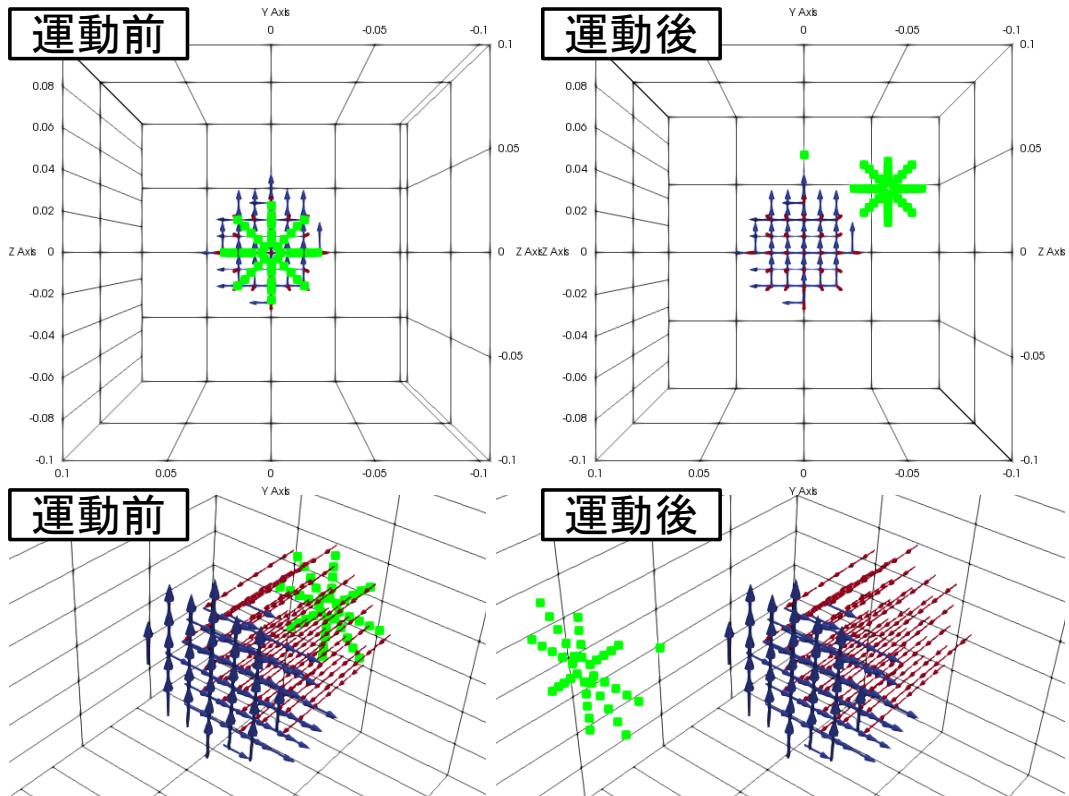


図4. 2. 1 電子銃モデルのシミュレーション結果

ログファイルの一部 (電子銃モデル)

```

=== calculation model is No.1 (main) ===
=== variable time step OFF : 1.0000e-010 (s) (main) ===

=== initialization of particles start (particle_init) ===
= tube type test particle =
= particle_num      :      40 (-) =
= radius partition number :      5 (-) =
= angle partition number :      8 (-) =
= particle mass     :  9.1094e-031 (kg) =
= particle charge   :  1.6022e-019 (C) =
= start position (rx) : -5.0000e-002 (m) =
= start position (ry) :  0.0000e+000 (m) =
= start position (rz) :  0.0000e+000 (m) =
= start velocity (vx) :  1.0000e+002 (m/s) =
= tube radius       :  2.5000e-002 (m) =
=== initialization of particles completion (particle_init) ===

=== electron gun (EB_field) ===
= acceleration voltage : 8.0000e+004 (V) =
= acceleration length  : 5.0000e-002 (m) =
= electric field (Ex)  : 1.6000e+006 (V/m) =
= deflection tube 1 length : 2.0000e-002 (m) =
= magnetic field 1 (By)  : 2.0000e-002 (T) =
= deflection tube 2 length : 2.0000e-002 (m) =
= magnetic field 2 (Bz)  : 3.0000e-002 (T) =
= tube radius          : 3.0000e-002 (m) =
=== electron gun (EB_field) ===

```

4. 3. サイクロトロン(case2)

サイクロトロンモデルのシミュレーション結果とログファイルの一部を下記に示す。

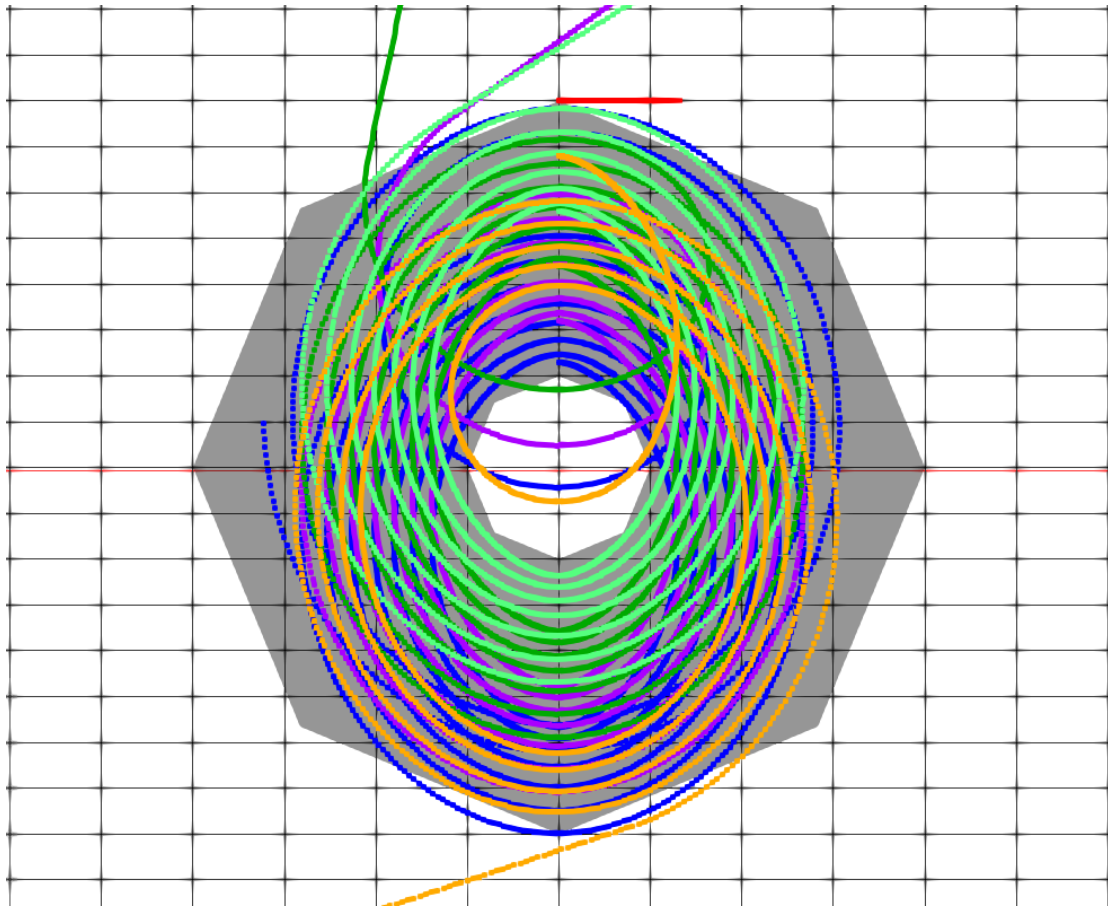


図4. 3. 1 サイクロトロン内の荷電粒子の軌跡

ログファイルの一部 (サイクロトロンモデル)

```
=== calculation model is No.2 (main) ===  
=== variable time step OFF : 5.0000e-011 (s) (main) ===  
  
=== initialization of particles start (particle_init) ===  
= test particle for cyclotron =  
= particle_num          :          20 (-) =  
= radius partition number :          20 (-) =  
= particle mass         : 9.1094e-031 (kg) =  
= particle charge       : -1.6022e-019 (C) =  
= inner case radius (ry) : 2.0000e-002 (m) =  
= outer case radius (ry) : 8.0000e-002 (m) =  
=== initialization of particles completion (particle_init) ===  
  
=== cyclotron (EB_field) ===  
= acceleration voltage : 2.0000e+001 (V) =  
= acceleration length  : 1.0000e-003 (m) =  
= electric field (Ex)  : 2.0000e+004 (V/m) =  
= magnetic field (Bz)  : -3.0000e-003 (T) =  
= thickness            : 1.0000e-003 (m) =  
= inner case radius    : 2.0000e-002 (m) =  
= outer case radius    : 8.0000e-002 (m) =  
=== cyclotron (EB_field) ===
```

4. 4. マグネトロン(case3)

マグネトロンモデルのシミュレーション結果とログファイルの一部を下記に示す。

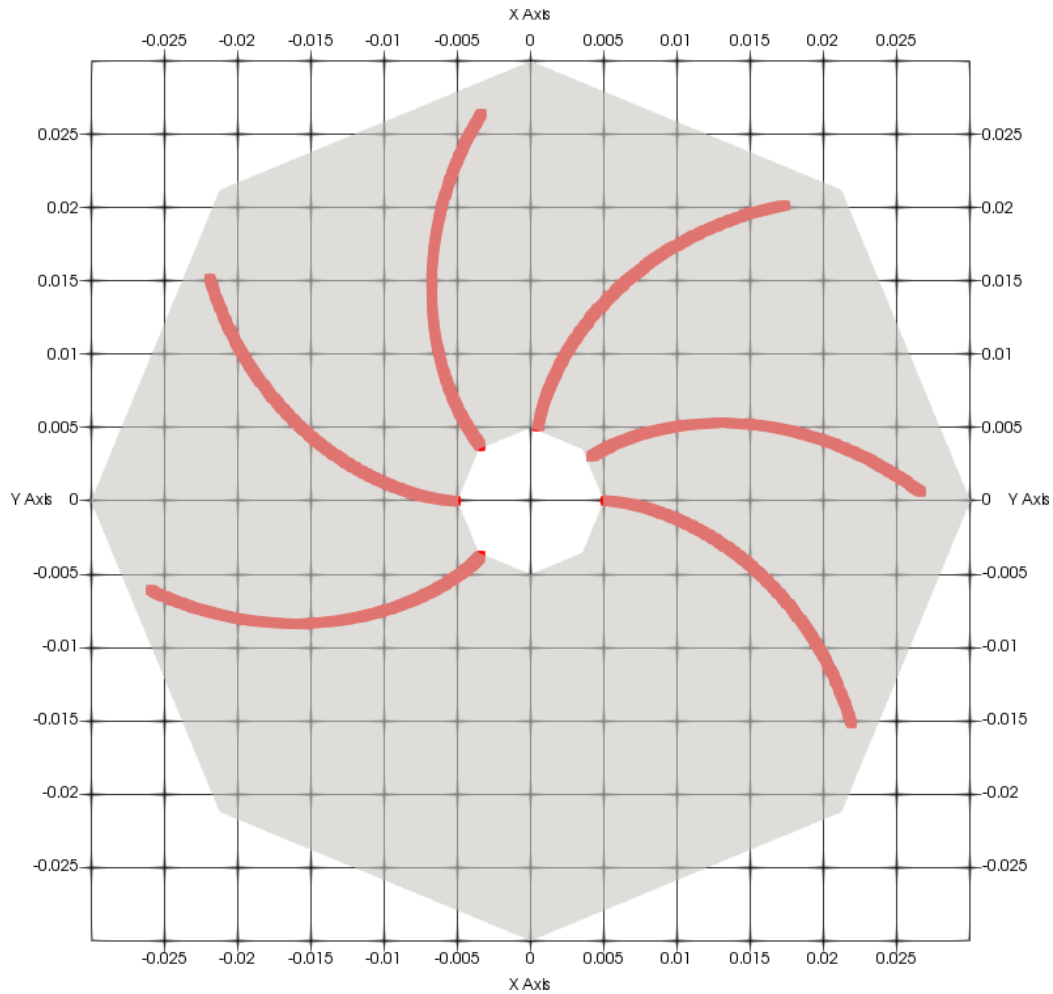


図 4. 4. 1 マグネトロン内の荷電粒子の軌跡

ログファイルの一部 (マグネトロンモデル)

```
=== calculation model is No.3 (main) ===  
=== variable time step OFF : 1.0000e-011 (s) (main) ===  
  
=== initialization of particles start (particle_init) ===  
= test particle for magnetron =  
= particle_num      :          30 (-) =  
= angle partition number :          30 (-) =  
= particle mass     : 9.1094e-031 (kg) =  
= particle charge   : 1.6022e-019 (C) =  
= inner case radius : 5.1000e-003 (m) =  
=== initialization of particles completion (particle_init) ===  
  
=== magnetron (EB_field) ===  
= voltage          : 1.0000e+003 (V) =  
= magnetic field (Bz) : 5.0000e-003 (T) =  
= height           : 2.0000e-002 (m) =  
= inner case radius : 5.0000e-003 (m) =  
= outer case radius : 3.0000e-002 (m) =  
=== magnetron (EB_field) ===
```

4. 5. 四重極電磁石(case4)

四重極電磁石モデルのシミュレーション結果とログファイルの一部を下記に示す。

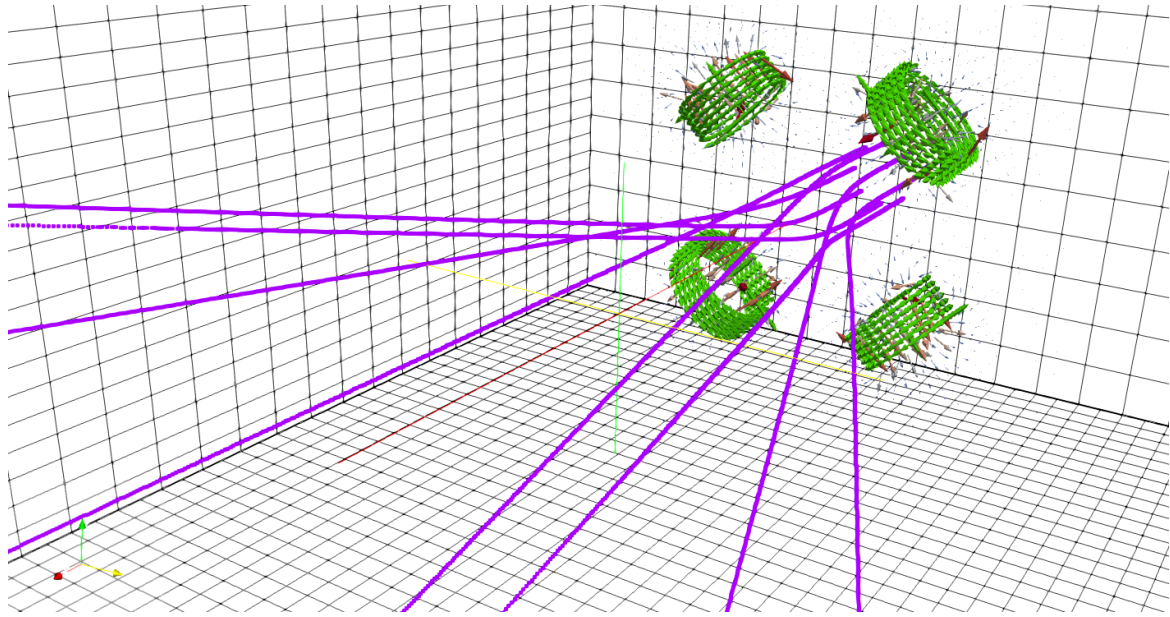


図 4. 5. 1 四重極電磁石通過後の荷電粒子の軌跡 (全景)

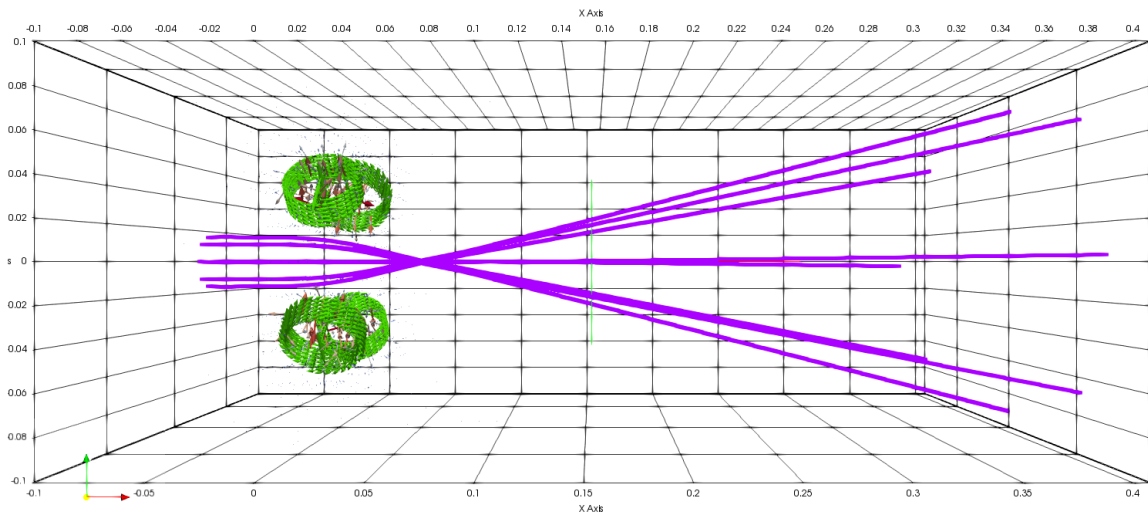


図 4. 5. 2 四重極電磁石通過後の荷電粒子の軌跡(X-Z)

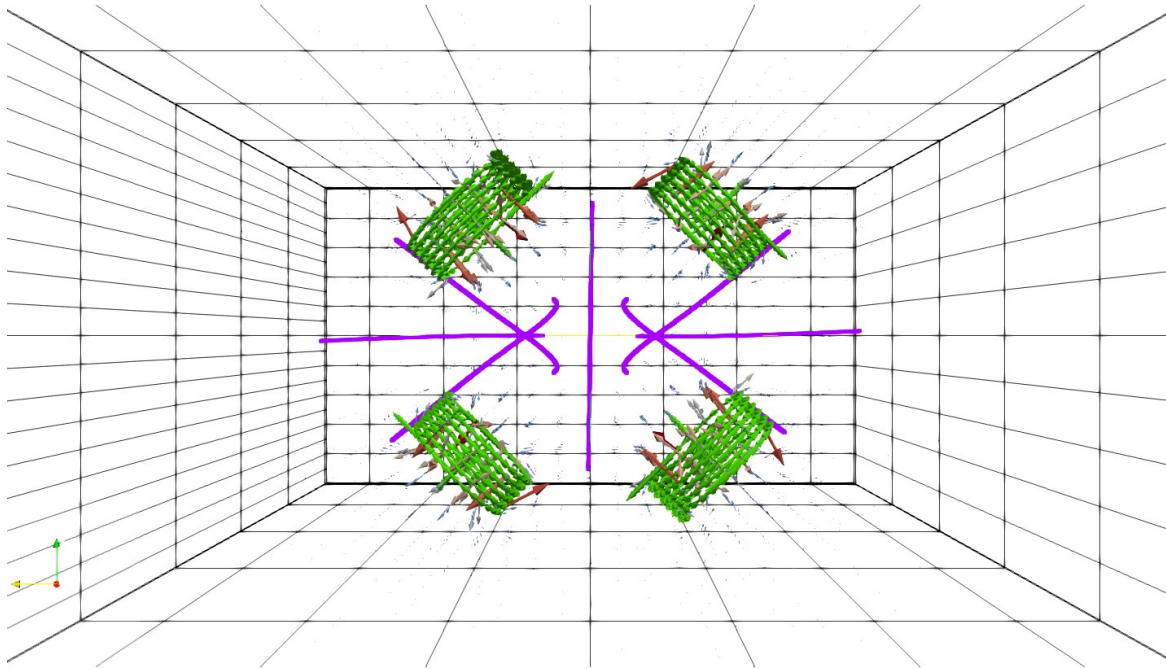


図 4. 5. 3 四重極電磁石通過後の荷電粒子の軌跡(Y-Z)

ログファイルの一部 (四重極電磁石モデル)

```

=== calculation model is No.4 (main) ===
=== variable time step OFF : 1.0000e-006 (s) (main) ===

=== initialization of particles start (particle_init) ===
= tube type test particle =
= particle_num          :          40 (-) =
= radius partition number :          5 (-) =
= angle partition number :           8 (-) =
= particle mass         : 9.1094e-031 (kg) =
= particle charge       : 1.6022e-019 (C) =
= start position (rx)   : -8.0000e-002 (m) =
= start position (ry)   : 0.0000e+000 (m) =
= start position (rz)   : 0.0000e+000 (m) =
= start velocity (vx)   : 1.0000e+003 (m/s) =
= tube radius           : 2.5000e-002 (m) =
=== initialization of particles completion (particle_init) ===

=== quadrupole electromagnet (quadrupole_electromagnet_SM) ===
= tube radius      : 5.0000e-002 (m) =
= coil current 1  : 1.0000e-005 (A) =
= coil current 2  : -1.0000e-005 (A) =
= coil current 3  : 1.0000e-005 (A) =
= coil current 4  : -1.0000e-005 (A) =
=== quadrupole electromagnet (quadrupole_electromagnet_SM) ===

=== solenoidal coil 01 (solenoidal_coil_01) ===
= power supply wiring partition number (radius) :          10 (-) =
= power supply wiring partition number (return) :          10 (-) =
= angle partition number                        :           30 (-) =
= winding number                               :              6 (-) =
= coil radius                                  : 2.0000e-002 (m) =
= coil pitch                                   : 3.0000e-003 (m) =
=== solenoidal coil 01 (solenoidal_coil_01) ===

```

4. 6. トロイダルコイル(case5)

トロイダルコイルモデルのシミュレーション結果とログファイルの一部を下記に示す。

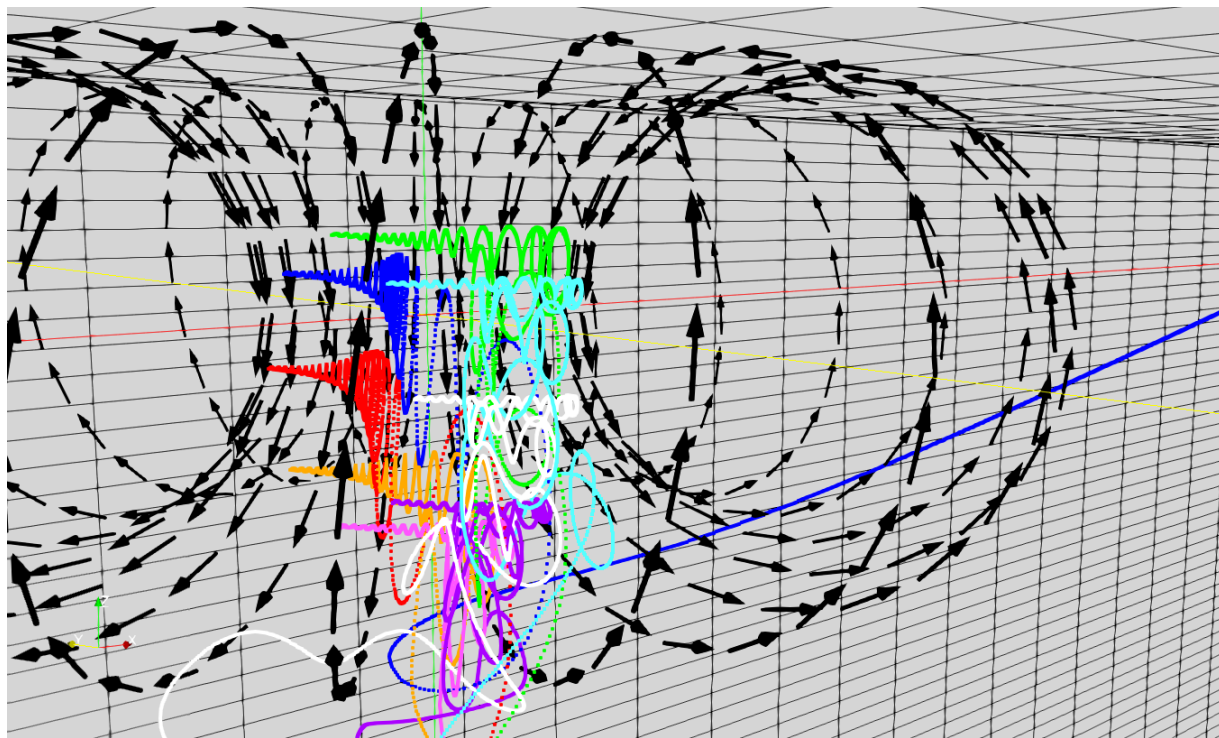


図4. 6. 1 トロイダルコイル内の荷電粒子の軌跡(1)

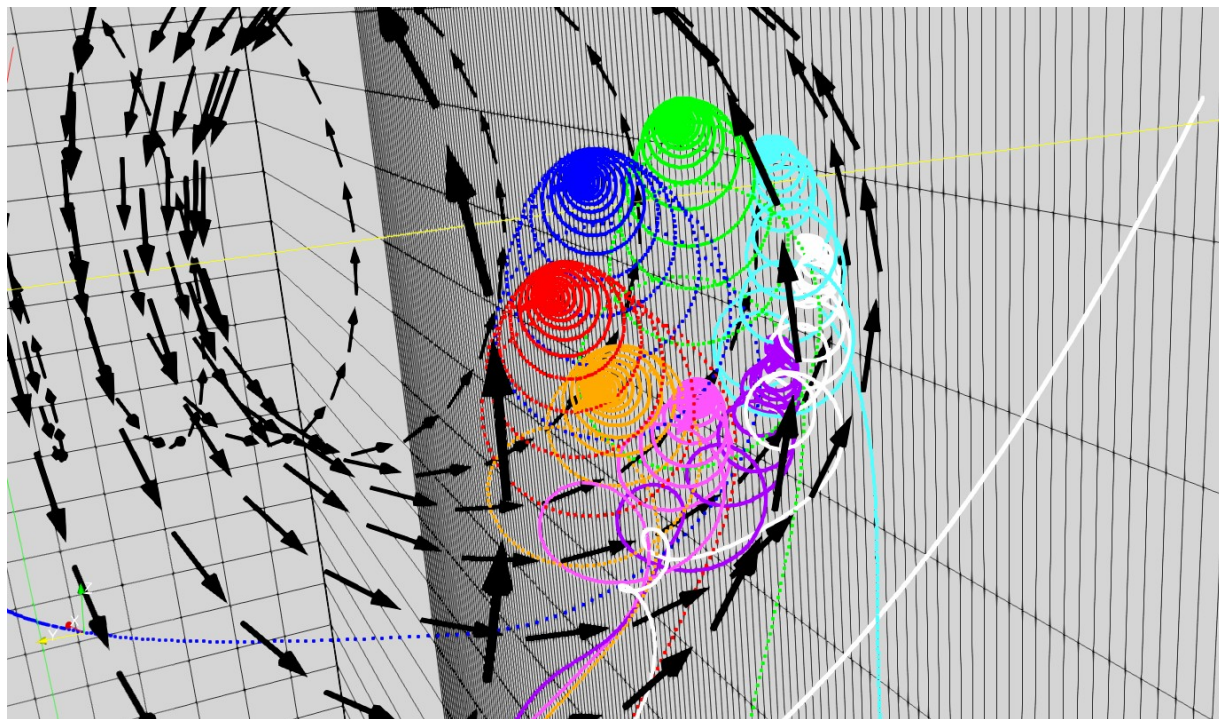


図4. 6. 2 トロイダルコイル内の荷電粒子の軌跡(2)

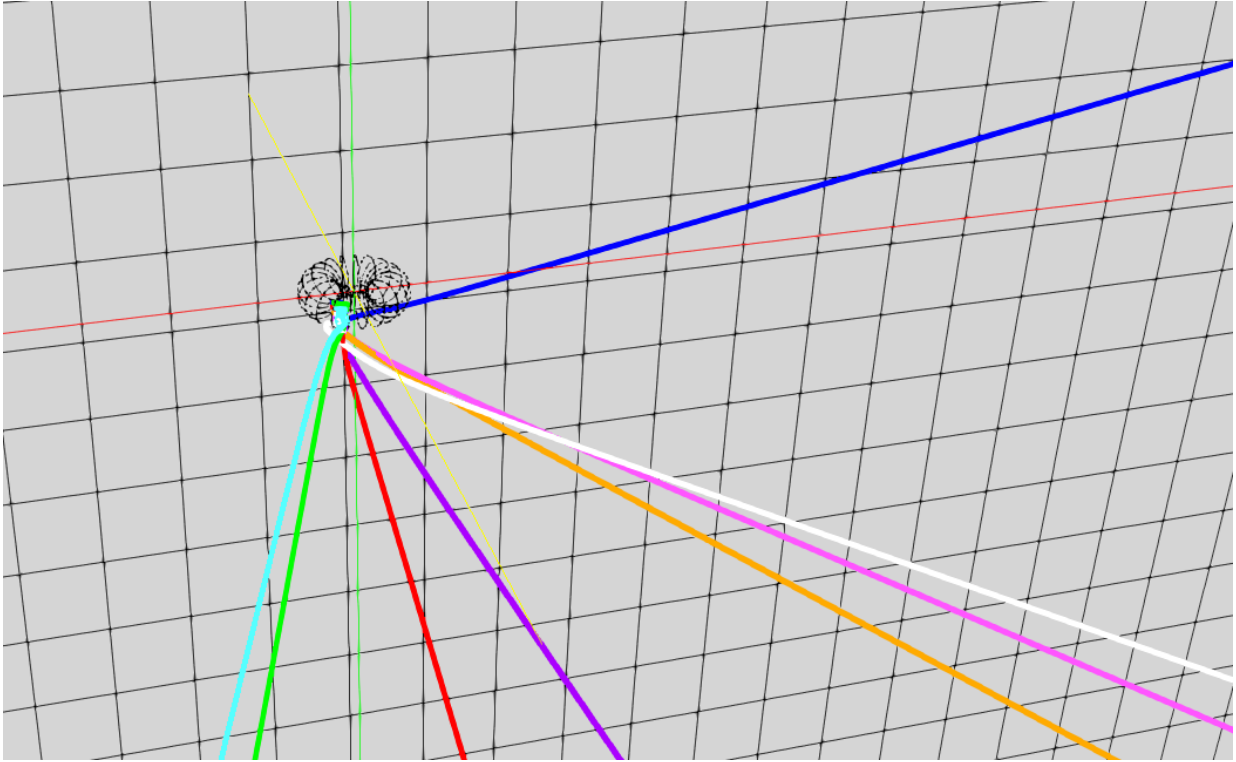


図4. 6. 3 トロイダルコイル内の荷電粒子の軌跡 (全景)

ログファイルの一部 (トロイダルコイルモデル)

```

=== calculation model is No.5 (main) ===
=== variable time step OFF : 2.0000e-009 (s) (main) ===

=== initialization of particles start (particle_init) ===
= tube type test particle =
= particle_num      :          40 (-) =
= radius partition number :          5 (-) =
= angle partition number :           8 (-) =
= particle mass     : 1.6726e-027 (kg) =
= particle charge   : 1.6022e-019 (C) =
= start position (rx) : -8.0000e-002 (m) =
= start position (ry) : -1.0000e-001 (m) =
= start position (rz) : 0.0000e+000 (m) =
= start velocity (vx) : 1.0000e+004 (m/s) =
= tube radius        : 4.0000e-002 (m) =
=== initialization of particles completion (particle_init) ===

=== toroidal coil (toroidal_coil_SM) ===
= coil current : 2.0000e+002 (A) =
=== toroidal coil (toroidal_coil_SM) ===

=== toroidal coil (toroidal_coil) ===
= angle partition number :          20 (-) =
= winding number         :          20 (-) =
= circle axis radius     : 1.0000e-001 (m) =
= circle thickness radius : 6.0000e-002 (m) =
=== toroidal coil (toroidal_coil) ===

```

4. 7. 模擬地磁気(case6)

模擬地磁気モデルのシミュレーション結果とログファイルの一部を下記に示す。なお、図中の各球の半径は次の通りで、青色の線は陽電子、赤色の線は陽子の軌跡である。本シミュレーションにおける陽電子と陽子の古典的運動エネルギーは等しく、陽電子の初速は太陽風の平均速度 450(km/s)である。（※本シミュレーションでは陽電子を用いているが、実際は電子が飛来してくる。）

- 緑色球（地球）：半径 6,000(km)
- 紫色球：半径 10,000(km)
- 橙色球：半径 20,000(km)

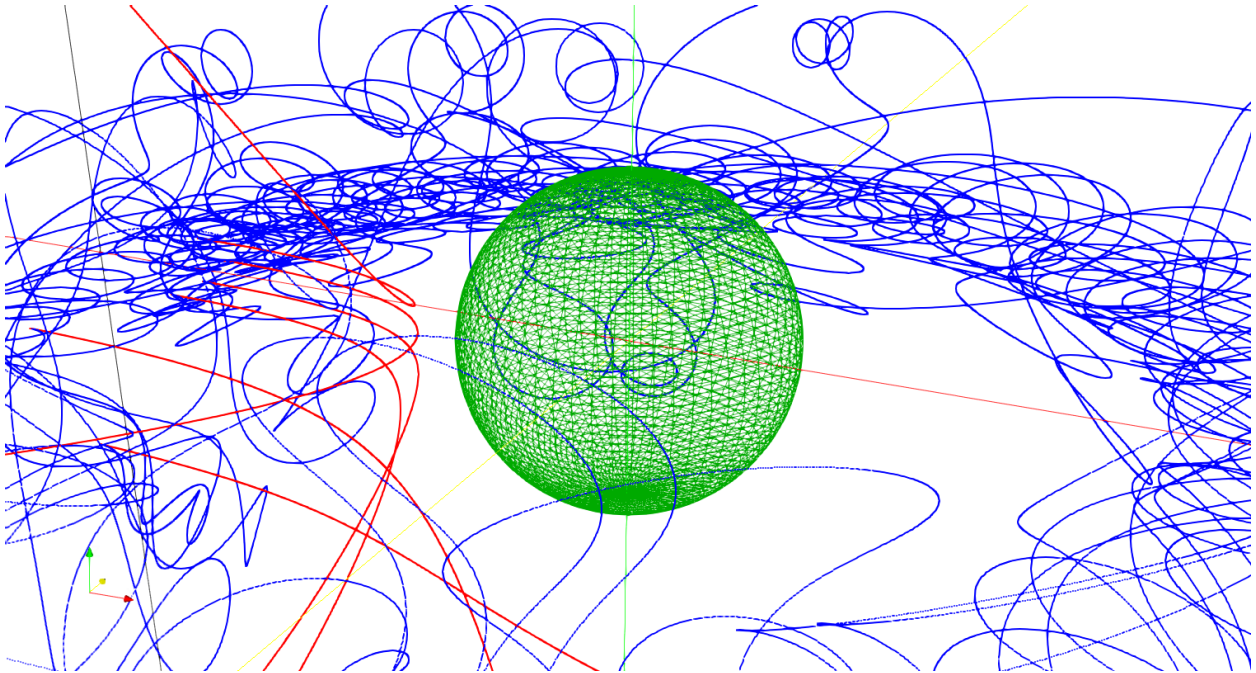


図 4. 7. 1 ヴァン・アレン帯の形成(1)

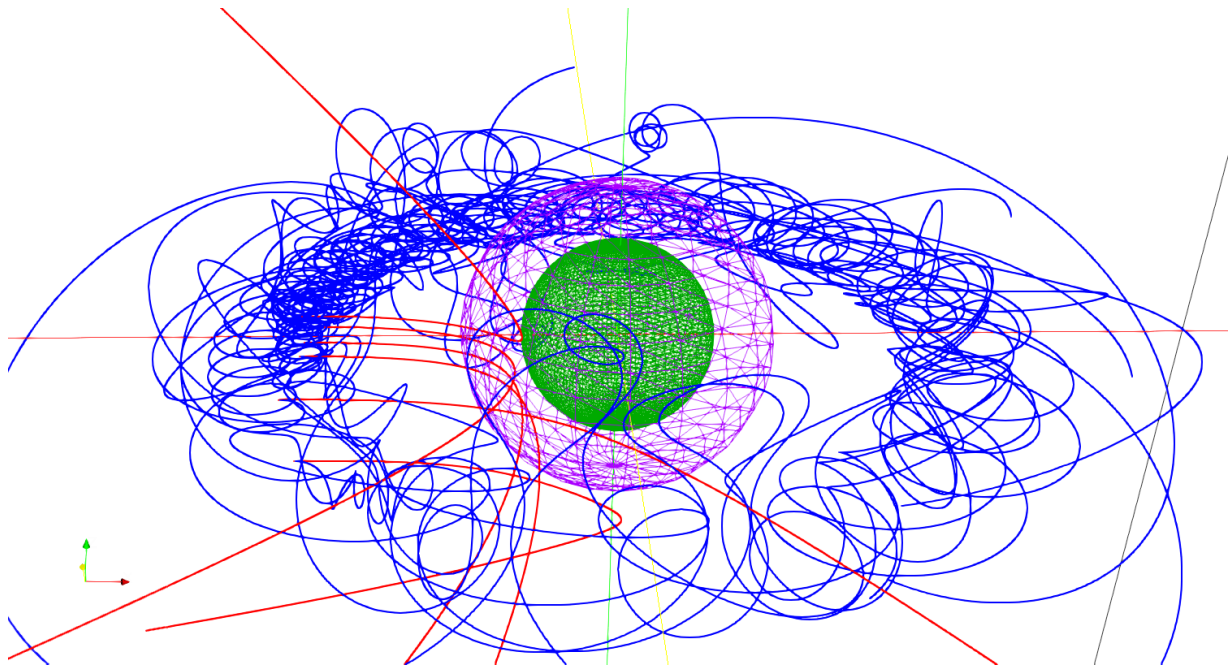


図 4. 7. 2 ヴァン・アレン帯の形成(2)

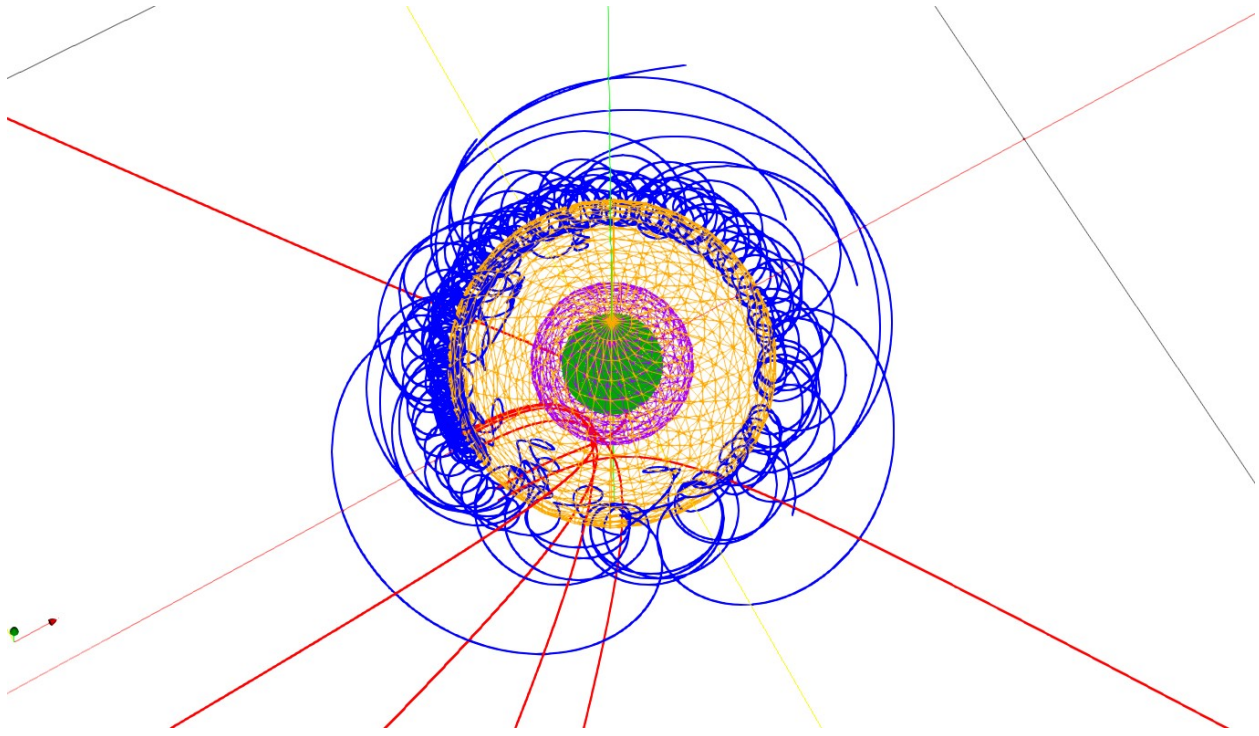


図4. 7. 3 ヴァン・アレン帯の形成(3)

ログファイルの一部 (陽子の軌道計算・模擬地磁気モデル)

```

=== calculation model is No.6 (main) ===
=== variable time step OFF : 2.0000e+000 (s) (main) ===

=== initialization of particles start (particle_init) ===
= tube type test particle =
= particle_num          :          48 (-) =
= radius partition number :          6 (-) =
= angle partition number :          8 (-) =
= particle mass         : 1.6726e-027 (kg) =
= particle charge      : 1.6022e-019 (C) =
= start position (rx)  : -2.0000e+007 (m) =
= start position (ry)  : 0.0000e+000 (m) =
= start position (rz)  : 0.0000e+000 (m) =
= start velocity (vx)  : 1.0502e+004 (m/s) =
= tube radius          : 8.0000e+006 (m) =
=== initialization of particles completion (particle_init) ===

=== annular current 01 (annular_current_01_SM) ===
= layer partition number      :          10 (-) =
= angle partition number      :          20 (-) =
= circumferential partition number :          30 (-) =
= single current              : -1.5000e+007 (A) =
= circle axis radius          : 3.0000e+006 (m) =
= circle thickness radius     : 3.0000e+006 (m) =
=== annular current 01 (annular_current_01_SM) ===

= case 6 point charge (EB_field) : -4.6300e-004 (C) =

```

ログファイルの一部 (陽電子の軌道計算・模擬地磁気モデル)

```

=== calculation model is No.6 (main) ===
=== variable time step OFF : 4.0000e-002 (s) (main) ===

=== initialization of particles start (particle_init) ===
= tube type test particle =
= particle_num      :      48 (-) =
= radius partition number :      6 (-) =
= angle partition number :      8 (-) =
= particle mass     : 9.1094e-031 (kg) =
= particle charge   : 1.6022e-019 (C) =
= start position (rx) : -2.0000e+007 (m) =
= start position (ry) : 0.0000e+000 (m) =
= start position (rz) : 0.0000e+000 (m) =
= start velocity (vx) : 4.5000e+005 (m/s) =
= tube radius       : 8.0000e+006 (m) =
=== initialization of particles completion (particle_init) ===

=== annular current 01 (annular_current_01_SM) ===
= layer partition number      :      10 (-) =
= angle partition number     :      20 (-) =
= circumferential partition number :      30 (-) =
= single current             : -1.5000e+007 (A) =
= circle axis radius         : 3.0000e+006 (m) =
= circle thickness radius    : 3.0000e+006 (m) =
=== annular current 01 (annular_current_01_SM) ===

= case 6 point charge (EB_field) : -2.5216e-007 (C) =
    
```

4. 8. 球面配置点電荷群(case7)

球面配置点電荷群モデルのシミュレーション結果とログファイルの一部を下記に示す。

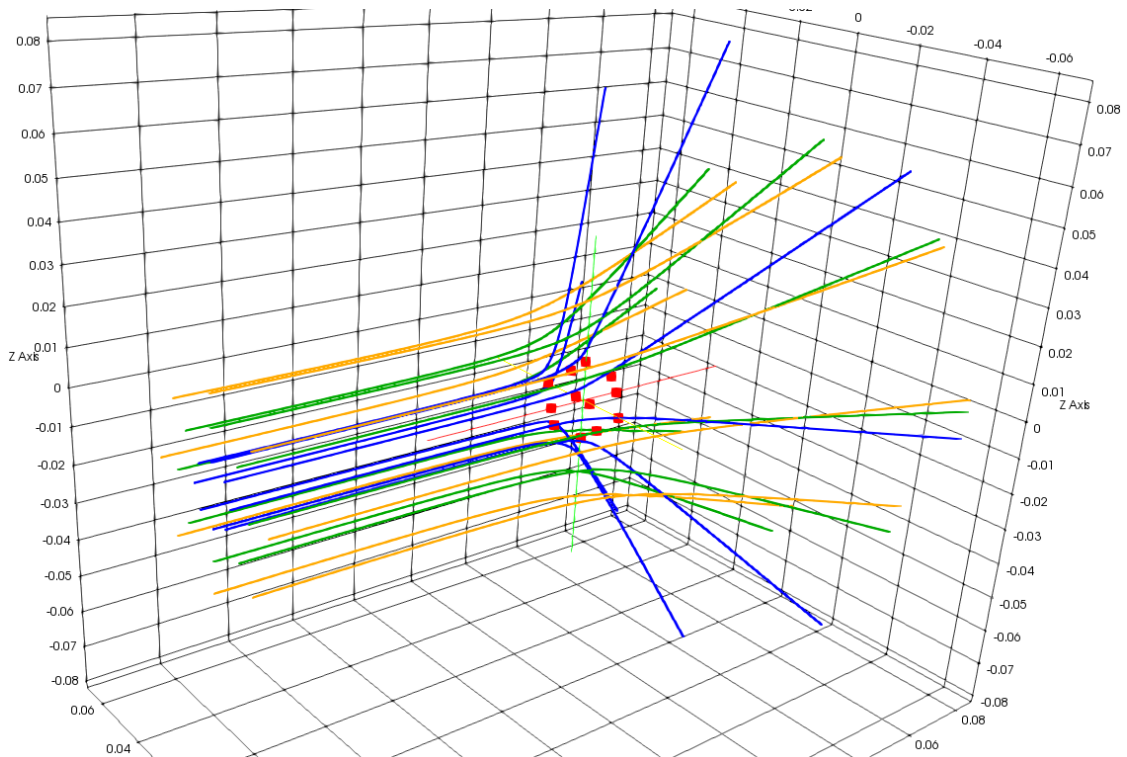


図 4. 8. 1 球面配置点電荷群近傍の荷電粒子の軌跡

ログファイルの一部（球面配置点電荷群モデル）

```
=== calculation model is No.7 (main) ===
=== variable time step OFF : 1.0000e-008 (s) (main) ===

=== initialization of particles start (particle_init) ===
= tube type test particle =
= particle_num      :          32 (-) =
= radius partition number :          4 (-) =
= angle partition number :          8 (-) =
= particle mass     : 9.1094e-031 (kg) =
= particle charge   : 1.6022e-019 (C) =
= start position (rx) : -1.0000e-001 (m) =
= start position (ry) : 0.0000e+000 (m) =
= start position (rz) : 0.0000e+000 (m) =
= start velocity (vx) : 1.0000e+004 (m/s) =
= tube radius       : 3.0000e-002 (m) =
=== initialization of particles completion (particle_init) ===

=== regular icosahedron (regular_icosahedron_SM) ===
= particle number      :          12 (-) =
= particle charge     : 2.0000e-017 (C) =
= radius of the outer sphere : 1.0000e-002 (m) =
= area length        : 1.0515e-002 (m) =
=== regular icosahedron (regular_icosahedron_SM) ===
```

4. 9. 可変タイムステップ機能の有効性

本プログラムで作成した機能である可変タイムステップ機能の有効性について比較を行った。本節では定量的な比較は行わず、画像のみを示す。

ログファイルの一部（可変タイムステップの有効性検証・トロイダルコイルモデル）

```
=== calculation model is No.5 (main) ===
=== variable time step ON : 1.0000e-004 (m) (main) ===

=== initialization of particles start (particle_init) ===
= test particle =
= particle_num :          1 (-) =
= pop[0][0]   : 1.6726e-027 (kg) =
= pop[0][1]   : 1.6022e-019 (C) =
= rx[0][0]    : 0.0000e+000 (m) =
= ry[0][0]    : 1.0000e-001 (m) =
= rz[0][0]    : 0.0000e+000 (m) =
= vx[0][0]    : -1.0000e+004 (m/s) =
= vy[0][0]    : 0.0000e+000 (m/s) =
= vz[0][0]    : 0.0000e+000 (m/s) =
=== initialization of particles completion (particle_init) ===

=== toroidal coil (toroidal_coil_SM) ===
= coil current : 2.0000e+002 (A) =
=== toroidal coil (toroidal_coil_SM) ===

=== toroidal coil (toroidal_coil) ===
= angle partition number :          20 (-) =
= winding number        :          20 (-) =
= circle axis radius    : 1.0000e-001 (m) =
= circle thickness radius : 6.0000e-002 (m) =
=== toroidal coil (toroidal_coil) ===
```

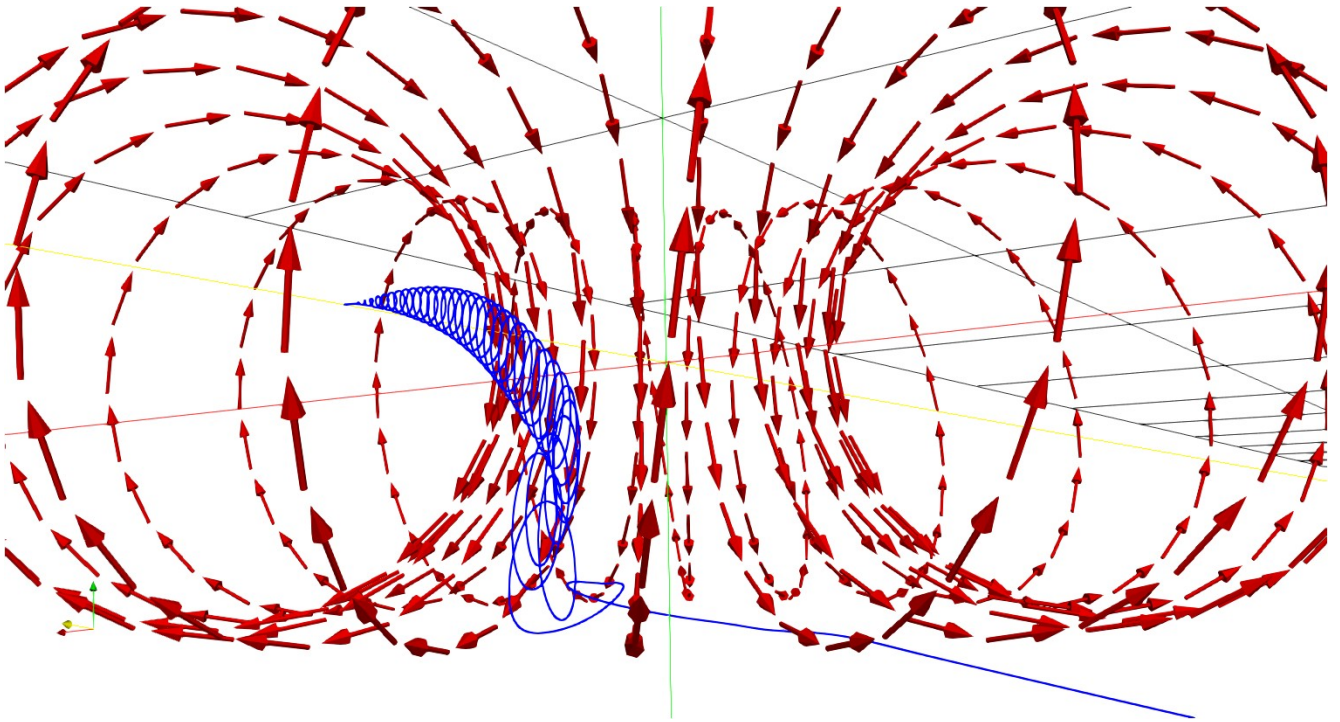


図4. 9. 1 可変タイムステップ機能 : ON(1.0×10^{-4} m/step)

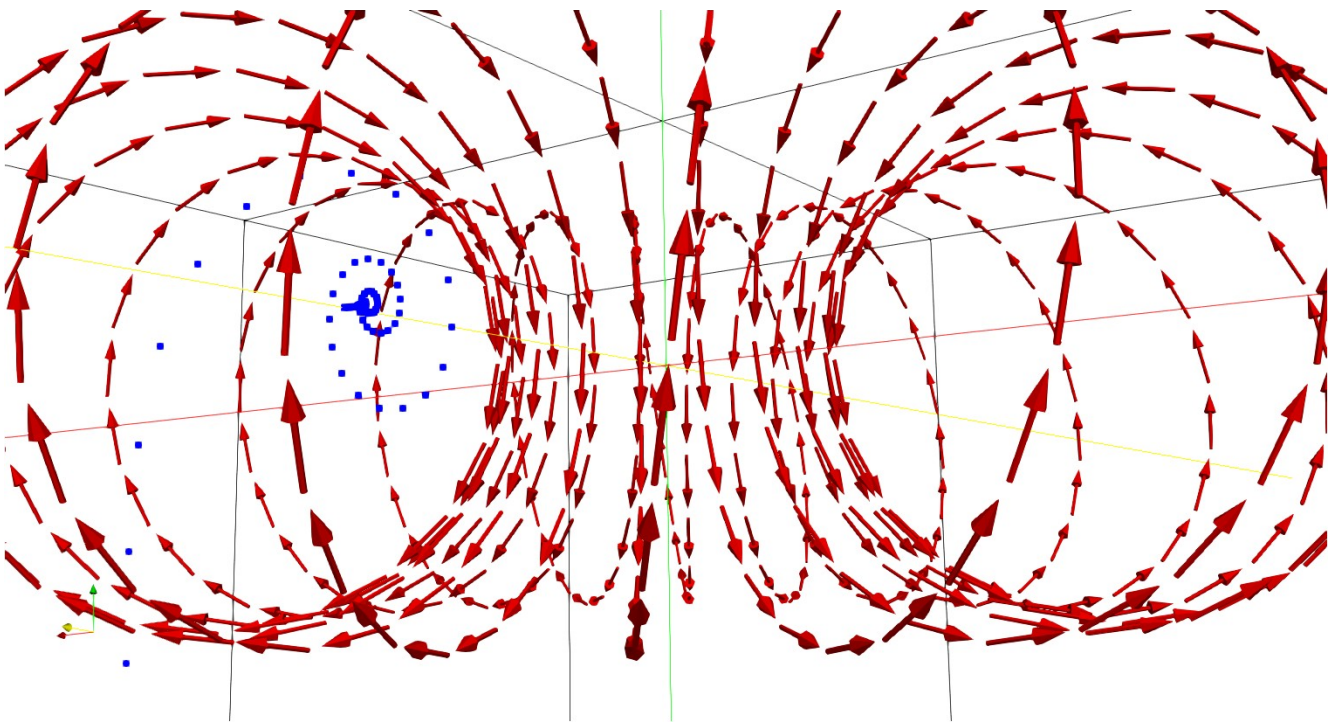


図4. 9. 2 可変タイムステップ機能 : OFF(1.0×10^{-8} s/step)

本シミュレーションにおける荷電粒子の速度は原理的に常に 1.0×10^4 (m/s) で一定であり、可変タイムステップ機能で設定した 1.0×10^{-4} (m) を進むのに要する時間は 1.0×10^{-8} (s) である。図4. 9. 1 (可変タイムステップ機能 : ON) と図4. 9. 2 (可変タイムステップ機能 : OFF) を比較すると、可変タイムステップ機能が働いている場合の方がシミュレーション結果が発散しにくいように見える。

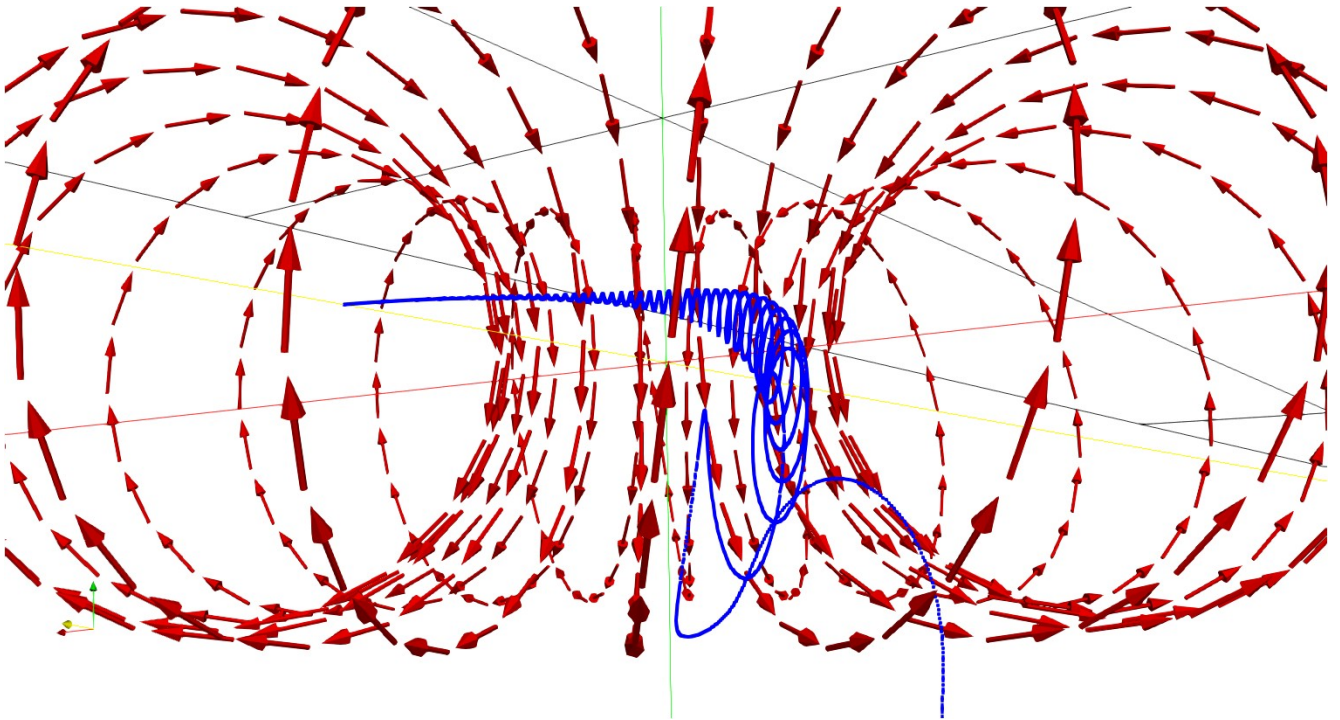


図4. 9. 3 可変タイムステップ機能 : OFF(1.0×10^{-9} s/step)

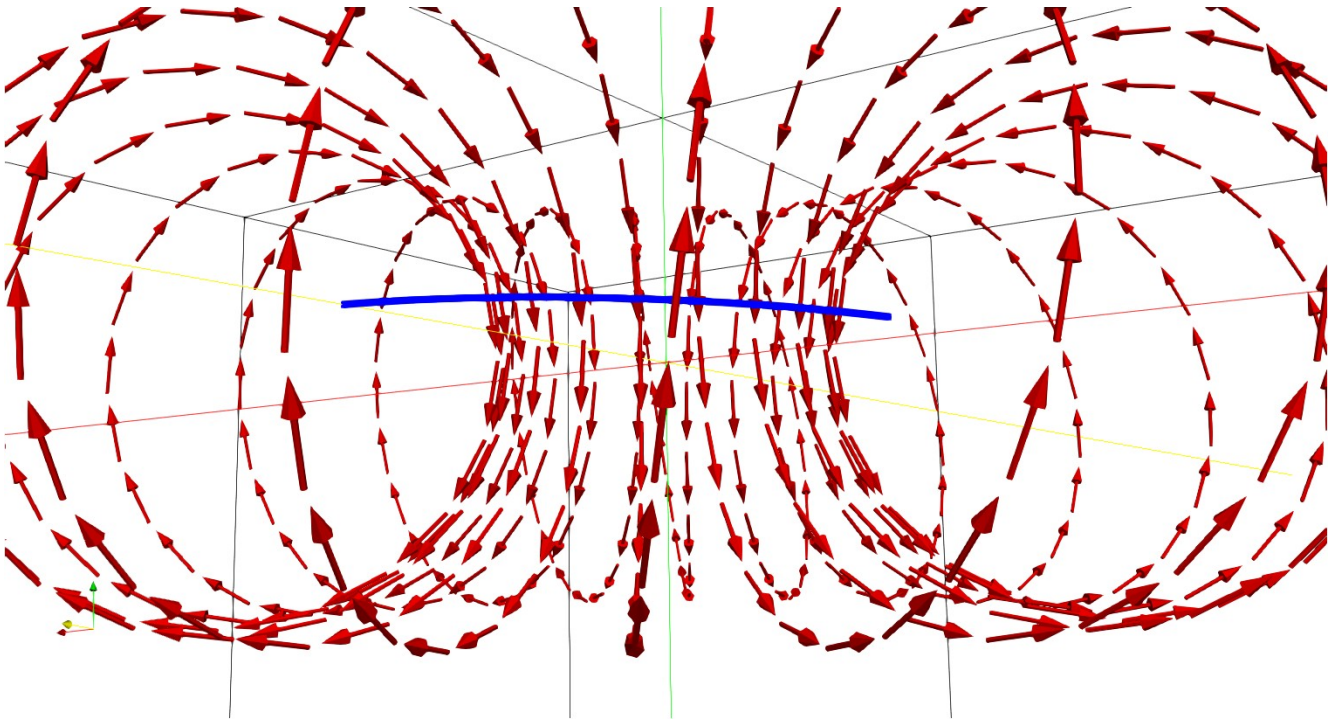


図4. 9. 4 可変タイムステップ機能 : OFF(1.0×10^{-10} s/step)

タイムステップ機能を OFF にし時間幅を細かくしてシミュレーションを行うと、安定性が増すことが分かる。以上から可変タイムステップ機能による計算精度の向上は微々たるもので、それよりも1タイムステップ毎の時間幅を細かくする事の方が容易に精度の向上につながる事が分かる。

5. 考察(Discussion)

今回作成したプログラムは定性的なシミュレーションを行う上ではある程度役立つものと思われる。しかしながら初期パラメータ（電磁場の強度、モデル形状、タイムステップ等）が適切に設定されていなければ、全く予想外の粒子軌道を描いたり、すぐに計算結果が発散したりと有効なシミュレーションを行う事ができなかった。特に点電荷・電流片を用いて電磁場を構成しているモデルについては積分計算があるため、より計算誤差が蓄積しやすく、この部分についてはある程度対策（情報落ち対策）が必要であると考えられる。

6. 結言(Summary)

本プログラムは適当に思いついた電磁場モデルにおける荷電粒子の運動を計算してみようという考えで作成したもので、特に何がしたいというものでは無かった。また、ParaView を使って色々な描画を行ってみたかった。

7. 文献(References)

- [1] : 物理入門コース 3 電磁気学 I 電場と磁場 長岡洋介著 岩波書店
- [2] : <http://www.quant-ph.cst.nihon-u.ac.jp/~takasugi/plasma/note/node10.html>
- [3] : http://physics.thick.jp/Plasma_Physics/Plasma_Physics_top.html
- [4] : <https://ja.wikipedia.org/wiki/%E3%83%B4%E3%82%A1%E3%83%B3%E3%83%BB%E3%82%A2%E3%83%AC%E3%83%B3%E5%B8%AF>
- [5] : <https://ja.wikipedia.org/wiki/%E5%A4%AA%E9%99%BD%E9%A2%A8>
- [6] : <http://wdc.kugi.kyoto-u.ac.jp/index-j.html>
- [7] : <https://ja.wikipedia.org/wiki/%E6%AD%A3%E4%BA%8C%E5%8D%81%E9%9D%A2%E4%BD%93>

8. 著者(Author)

氏名：志多 友史（工学修士）

略歴：

2011年：下位国立大学 工学部電気系学科卒業

2013年：同大学大学院 工学研究科修了

2013年：研究開発機関へ就職

興味：物理・数学・コンピュータ・電気電子工作

9. 備考(Notes)

特になし。