

# Plasma Simulation Study with Lorentz Force

## 1 Abstract

**Abstract-** Many high performance plasma simulations use Maxwell's equations to solve for the electric and magnetic fields. Such methods are very useful when the fields generated by the plasma are much stronger than or compete with those generated by the apparatus. However, for simpler studies where the fields generated by the apparatus are much stronger than those generated by the plasma, simple methods could be used. Simulation based only on the Lorentz force could still be powerful. Such a simulation; written in Python in Jupyter notebooks, is used in this work and 3 studies are presented. The trajectories of the particles are discussed, with the aim of understanding how a plasma could be controlled with apparatus that generate electric and magnetic fields. A magnetic field setup with variable strength; perhaps due to current in a coil, and orientation is seen by itself, powerful in controlling plasma in a chamber. Such studies could help better operate small plasma devices and understand plasma processes in surface engineering techniques.

**Key words:** Lorentz force, simulation

**DOI:** if there will be

## 2 Introduction

### 2.1 Plasma Models

Plasma is primarily studied with three approaches[1].

#### 1. Single particle description

The single particle model is used to describe the motion of a charged particle under the influence of electric and magnetic fields. This is the simplest model to work with. The particle's motion is described by the Lorentz force

$$\frac{d\mathbf{v}}{dt} = \frac{q}{m} (\mathbf{E} + \mathbf{v} \times \mathbf{B}) \quad (1)$$

To describe a plasma with the single particle model, each particle can be evolved (solved for) using the Lorentz force. Under suitable approximations; for example, if the mean free path of the particles is much larger than the length scale of the problem; which maybe something like the dimension of a plasma chamber, it may be assumed that the particles do not interact with each other while each evolves under the influence of the Lorentz force. However, interaction terms could be added to the description.

#### 2. Kinetic theory

The kinetic theory describes the plasma as collection of particles described by a random variable with a density function  $f(x, y, z, v_x, v_y, v_z, t)$  which describes the number of particles at position  $(x, y, z)$  at time  $t$  with velocity components between  $v_x$  and  $v_x + dv_x$ ,

$v_y$  and  $v_y + dv_y$ ,  $v_z$  and  $v_z + dv_z$  in directions  $x$ ,  $y$  and  $z$  respectively. For a simpler notation  $f(x, y, z, v_x, v_y, v_z, t)$  is denoted as  $f(\mathbf{x}, \mathbf{v}, t)$ . Integrating over the velocities, the expression

$$\int_{all \ v_x} dv_x \int_{all \ v_y} dv_y \int_{all \ v_z} dv_z f(\mathbf{x}, \mathbf{v}, t)$$

gives the number of particles at position  $\mathbf{x}$ , at time  $t$ . The evolution of this system is described as the changing of the density function. Boltzmann equation is often used to describe this dynamics:

$$\frac{\partial f}{\partial t} + \mathbf{v} \cdot \nabla f + \frac{\mathbf{F}}{m} \cdot \partial_{\mathbf{v}} f = \left( \frac{\partial f}{\partial t} \right)_c$$

where  $\partial_{\mathbf{v}}$  means  $\mathbf{i} \frac{\partial}{\partial v_x} + \mathbf{j} \frac{\partial}{\partial v_y} + \mathbf{k} \frac{\partial}{\partial v_z}$ , the gradient in the velocity space, and the term on the right hand side represents a collision or interaction term for the particles.

### 3. Fluid model and Magnetohydrodynamics (MHD)

The fluid model and the MHD approach, describe plasma by coupling Maxwell's equations to the fluid equations. This can also be obtained by taking moments of the Boltzmann equation. Charged particles in motion generate currents and therefore themselves create electric and magnetic fields which have an effect on other charged particles. In the fluid model, this is described as the fluid moving under the influence of the electric and magnetic fields that the fluid itself generates. In situations where the fields created by the plasma is much more stronger than or competes with externally applied fields; like in astrophysical plasma or nuclear fusion, the MHD approach is very important to study.

## 2.2 Lorentz equation and Boris Algorithm

The Lorentz equation(1) is discretized as

$$\frac{\mathbf{v}_{k+1} - \mathbf{v}_k}{\Delta t} = \frac{q}{m} \left[ \mathbf{E}_k + \frac{(\mathbf{v}_{k+1} + \mathbf{v}_k)}{2} \times \mathbf{B}_k \right]$$

The discretized Lorentz equation may be solved using the Boris Algorithm[2].The Boris Algorithm splits the discretized Lorentz equation into 3 pieces:

$$\frac{\mathbf{v}^- - \mathbf{v}_k}{(\Delta t/2)} = \frac{q}{m} \mathbf{E}_k \quad \text{or} \quad \frac{\mathbf{v}^- - \mathbf{v}_k}{\Delta t} = \frac{1}{2} \frac{q}{m} \mathbf{E}_k \quad (2)$$

which is often called the first half of the electric pulse.

$$\frac{\mathbf{v}^+ - \mathbf{v}^-}{\Delta t} = \frac{q}{m} \left( \frac{\mathbf{v}^+ + \mathbf{v}^-}{2} \right) \mathbf{B}_k \quad (3)$$

which is often called rotation by the magnetic field.

$$\frac{\mathbf{v}_{k+1} - \mathbf{v}^+}{(\Delta t/2)} = \frac{q}{m} \mathbf{E}_k \quad \text{or} \quad \frac{\mathbf{v}_{k+1} - \mathbf{v}^+}{\Delta t} = \frac{1}{2} \frac{q}{m} \mathbf{E}_k \quad (4)$$

which is often called the second half of the electric pulse.

Adding equations (2), (3) and (4) gives

$$\frac{\mathbf{v}_{k+1} - \mathbf{v}_k}{\Delta t} = \frac{q}{m} \left[ \mathbf{E}_k + \frac{(\mathbf{v}^+ + \mathbf{v}^-)}{2} \times \mathbf{B}_k \right]$$

which is almost the discretized Lorentz equation except that  $(\mathbf{v}^+ + \mathbf{v}^-)$  is substituted for  $(\mathbf{v}_{k+1} + \mathbf{v}_k)$ . However, subtracting equation (2) from equation (4) gives  $(\mathbf{v}^+ + \mathbf{v}^-) = (\mathbf{v}_{k+1} + \mathbf{v}_k)$ , giving the discretized Lorentz equation. This means that the Boris algorithm is equivalent to the discretized Lorentz equation. The equations (2), (3) and (4) can be written slightly different as

$$\begin{aligned} \mathbf{v}^- &= \mathbf{v}_k + q' \mathbf{E}_k \\ \mathbf{v}^+ &= \mathbf{v}^- + 2q' (\mathbf{v}^- \times \mathbf{B}_k) \\ \mathbf{v}_{k+1} &= \mathbf{v}^+ + q' \mathbf{E}_k \\ \mathbf{x}_{k+1} &= \mathbf{x}_k + \Delta t \mathbf{v}_{k+1} \end{aligned} \tag{5}$$

where  $q' = \frac{q}{m} \frac{\Delta t}{2}$ , alongwith the equation discretizing the velocity in terms of the position. These equations are used to update the velocities and positions of the particles in the plasma under the influence of the Lorentz force.

### 3 Setting up the Simulation

In the context of plasma processes like Magnetron Sputtering where particles have a considerably large mean free path compared to the chamber dimensions, it can be convenient not to use the fluid model and the MHD descriptions of a plasma. In such processes, it is almost always the case that the strength of the electric and magnetic fields applied by the apparatus is far more stronger than the electric and magnetic fields generated by charged particles in a plasma. This approximation allows one to avoid solving Maxwell's equations and use only the Lorentz force. So in this work the single particle model is used for plasma evolution, where each particle is updated under the influence of the Lorentz force using the Boris Algorithm discussed earlier. It is assumed that the particles do not interact with each other. However, one interesting decision comes up. In a simulation; where one studies the positions and velocities of the particles as they change during the evolution of the system, assigning the initial positions and velocities of the particles can be interesting in itself. Initializing all the positions and velocities to zero doesn't capture the similarity of the system to an actual device. So in this work, in an attempt to use a bit of kinetic theory, the initial speeds of the particles are sampled from probability distributions. However, the evolution of particles does not use any ideas from kinetic theory.

The simulation is written in Jupyter notebooks as Python code. The program is accessible in a GitHub repository[3]. The three basic components required are:

1. Particle
2. Fields
3. Initialization

1) A particle (described by an instance object of a class using the Object oriented programming paradigm) has a velocity and position which are updated during the simulation, and charge, mass and a name; for convenience, which do not change during the simulation. A particle can use the associated Boris update function which updates the velocity and position of the particle according to the Boris Algorithm.

2) A field object can use associated functions to return the values of electric and magnetic fields. A function returning uniform electric field; same as the input argument, and a function returning uniform magnetic field are used. A function returning the magnetic field due to a pair of Helmholtz coils according to the expression  $B = \left(\frac{4}{5}\right)^{3/2} \frac{\mu_0 n I}{R}$  where the input

current  $I$  is varied, and the number of turns in the coil  $n$  and the radius of the coil  $R$  is fixed when defining the function. The field is assumed to be uniform along a direction given as an input argument, so the field isn't exactly like one created by a Helmholtz coil but rather uniform and equal in magnitude as the field midway between a pair of Helmholtz coils. However, varying the current is more intuitive and similar to how one might use an actual Helmholtz coil setup to control a real plasma, than varying the value of the magnetic field directly. A function was defined to mimic the electric field created by an electrode connected to a certain voltage of electric supply, however, the effect of such an electric field was not clear in the plots and animations as that of the magnetic field defined to mimic the one created by a Helmholtz coil and so the electric field was set to zero in two of the simulation studies instead. It might be interesting to define other field configurations either to mimic instruments in the lab, or with analytic expressions.

3) A sampler object can sample arrays of positions and velocities using a few distributions. Positions can be sampled such that all particles start in the same position, intended to mimic the injection of particles into the chamber through a port. Positions can also be sampled such that the particles are all a certain distance from a point, intended to mimic particles being sprayed from the walls of the chamber, or a flux of particle being reflected off the walls of the chamber or being directed by a magnetic field. Speeds can be sampled using the Maxwellian distribution as defined in the SciPy library (`scipy.stats.maxwell[4]`). The Maxwellian distribution is a standard distribution used in plasma physics and study of gaseous systems. Speeds can also be sampled using the Symmetric Beta distribution as defined in the SciPy library (`scipy.stats.rdist[5]`) with parameter  $c = 4$  which gives a half parabolic distribution. A parabolic distribution is used because the shape of a parabolic distribution could be used as an approximate for a Maxwellian distribution. A full parabolic (two halves) distribution would have been better in comparing against a Maxwellian distribution. The sampled positions and velocities are saved in csv files and read when the simulation is initialized so that the same data can be used, as it is convenient if the simulation gives the same results when run later. Saving the positions and velocities at some point during a simulation and using that for initializing another batch of particles could also be an interesting thing to do; but hasn't been done in this work.

Some auxiliary components of the program include:

1. Batch
2. Run

### 3. Constants

- 1) A batch of particles initializes and updates a number of particles at the same time, which is convenient to work with a large number of particles, even 100.
- 2) A run instance describes the running of a plasma simulation, where different batches can be initialized and updated. This could have been achieved with batch and multiple batches have not been used in this work, but it's convenient to use sometimes.
- 3) All the constants used in the study are recorded together to be accessed when needed to avoid having to input them every time which may lead to slightly different values and precision being used.

Two checks were performed on the simulation. The first check was done to verify the update of the positions and velocities, comparing the output of the program against calculations done by hand. It was seen that the algorithm has no errors in the updates, up to the precision of computation. The second check was done to verify the sampling from the Maxwellian distribution, comparing the average speed of the sampled particles against the average speed of particles in a Maxwellian distribution calculated using analytic expression. For a 100 particles, the error was small enough to be attributed to statistical fluctuations involved in statistical sampling; using a large number of particles is expected to reduce the error even more.

## 4 Short Studies

3 studies were done. Study 1 was about varying the electric field while keeping the magnetic field constant; where the value of the field strength was directly specified as an input. Study 2 was about varying the magnetic field while the electric field was set to 0 throughout. Initially the electric field configuration intended to mimic that created by an electrode connected to a supply voltage was used; which was later removed because it did not seem to make the results (plots and animations) clear. Study 3 was about using different initializations than those in study 2, while the field configurations were the same.

Plots have been generated and animations for those plots have been generated as well. Animations for plots of positions of 10 particles are very helpful in understanding the simulations, although they have not been presented here. Due to a bug in the program, unlike in study 1 and study 2, the positions and velocities for 10 particles could not be plotted in the same plot and hence an alternative approach to plot them in different subplots in the same figure was used. Plotting them in the same plot would have been more helpful in understanding the simulation, especially the animation and plot of the positions of 10 particles. Nonetheless, plotting them in different subplots is also quite helpful.

### 4.1 Study 1- Varying Electric field, Constant Magnetic field

The following data was used for the study:

- 100 particles of Hydrogen (relative atomic mass:  $1.008 \text{ g mol}^{-1}$ )
- duration of 1 step of update: 0.1 ms

#### Sampling of the Initial Positions and Velocities of the particles

In this study the following sampling (initialization) data was used:

- Speeds are sampled from a Maxwellian distribution with plasma temperature 10000 K.
- Velocity directions are sampled from uniform distribution.
- Positions are sampled such that all particles start at  $[-0.5, 0, 0]$  (A box  $1\text{m} \times 1\text{m} \times 1\text{m}$  from  $[-0.5, -0.5, -0.5]$  to  $[0.5, 0.5, 0.5]$  may be considered for reference)

The standard Maxwellian distribution was used for the initial speeds of the particles and all particles were initialized at the same position.

### Configurations of the Electric and Magnetic Fields

The following field configurations were used in the study:

- Magnetic field: 10 mT along the  $y$ -axis  $[0, 1, 0]$
- Electric field  $1000 \text{ Vm}^{-1}$  scaled by  $[0, 0, 1, -1, 2, -2, 3, -3, 4, -4, 5, -5]$  for 100 steps each.

The positions of 1 and 10 particles in the simulation are plotted respectively in figures (1) and (2).

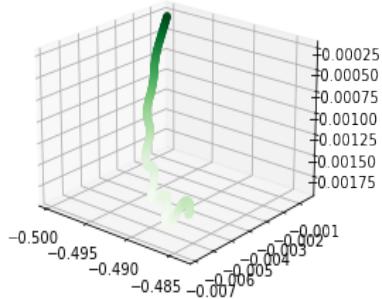


Figure 1: position of a particle

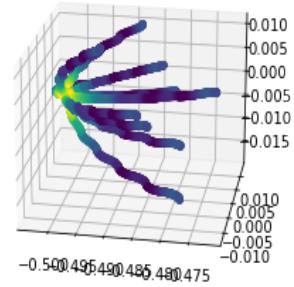


Figure 2: positions of 10 particles

The components of positions and velocities for 10 particles are also plotted.

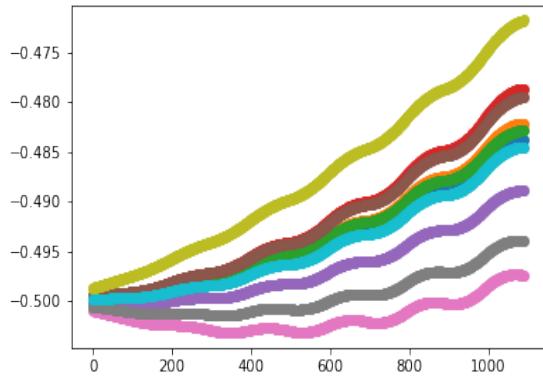


Figure 3:  $x$ -component of positions

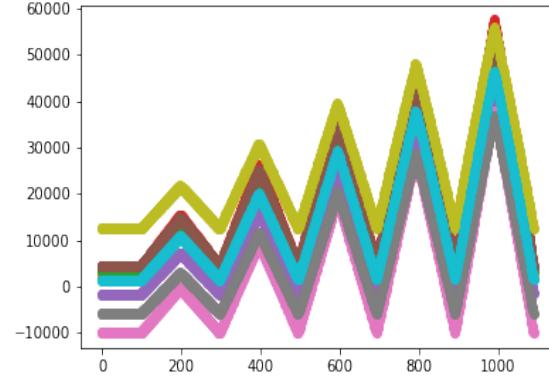


Figure 4:  $x$ -component of velocities

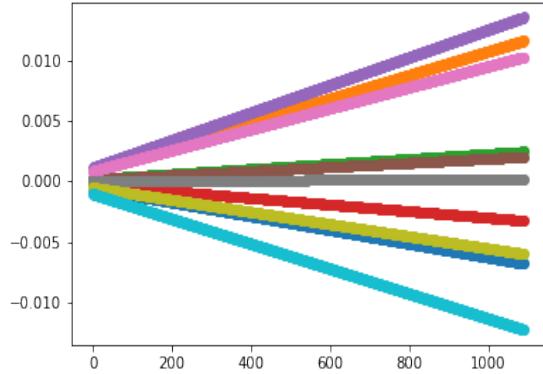


Figure 5:  $y$ -component of positions

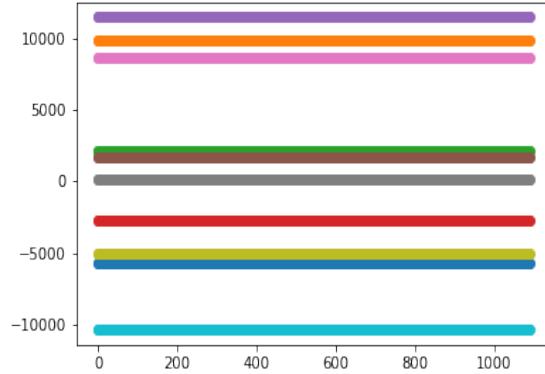


Figure 6:  $y$ -component of velocities

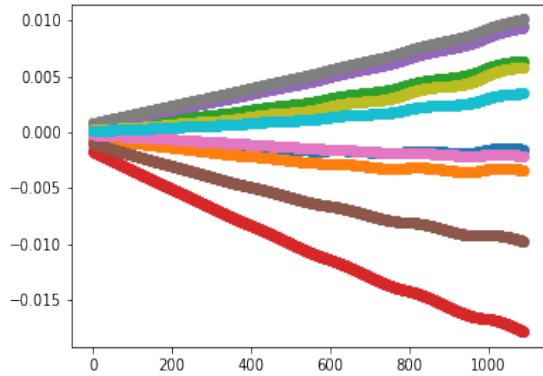


Figure 7:  $z$ -component of positions

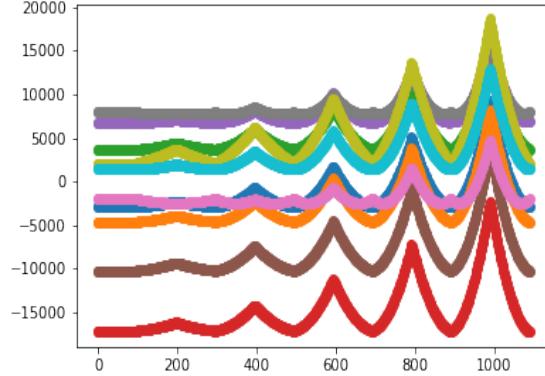


Figure 8:  $z$ -component of velocities

Figures (3), (5) and (7) show that the particles all start at  $[-0.5, 0, 0]$  as intended and spread out based on their initial velocities and velocities after subsequent updates. As the electric field is along the  $x$ -axis, the magnetic field is along the  $y$ -axis, and according to the Lorentz force equation(1) the acceleration of a particle,  $\mathbf{a} \propto \mathbf{E} + (\mathbf{v} \times \mathbf{B})$ , there is non-zero acceleration along the  $x$ -axis, zero acceleration along the  $y$ -axis and non-zero acceleration along  $\mathbf{v} \times \mathbf{B}$  which is dependent on the velocity of a particle at a given instance. In figure (4), the increase and decrease in the  $x$ -component of velocities are marked by sharp changes after every 100 steps when the sign of the electric field was reversed; the effect of both  $\mathbf{E}$  and  $\mathbf{v} \times \mathbf{B}$  terms is observed in the acceleration. The velocities also increase to a greater value later in the simulation as the electric field is scaled by increasing values. In figure (6), it is observed that the  $y$ -component of velocities stay the same as expected. In figure (8), the effect of  $\mathbf{v} \times \mathbf{B}$  term on the  $z$ -component of acceleration can be observed. The events of reversals of the electric field can hence also be observed.

## 4.2 Study 2- Maxwellian distributed speeds and Helmholtz coils

In this study the same sampling data as in study 1 was used.

### 4.2.1 Reversed Magnetic field

For the first part of this study, the following field configurations were used:

#### Configurations of the Electric and Magnetic Fields

- Magnetic field due to a Helmholtz coil (number of turns: 1000 in each coil, radius: 0.1 m): for 100 steps each
  - first 20A current, orientation along the  $z$ -axis [0,0,1]
  - second -20A current, orientation along the  $z$ -axis [0,0,1]
  - third 20A current, orientation along the  $z$ -axis [0,0,1].
- Electric field constantly set to 0.

The orientation of the coil was fixed in the  $z$ -direction and the direction of the current was changed while keeping the amplitude at a constant value of 20 A. The positions of 1 and 10 particles in the simulation are plotted respectively in figures (9) and (10).

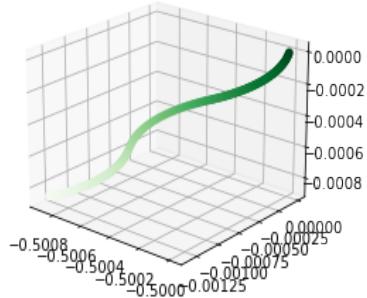


Figure 9: position of a particle

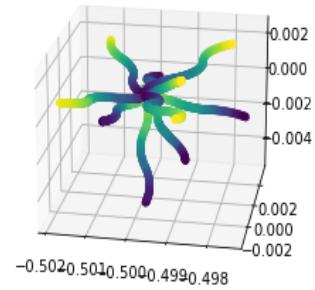


Figure 10: positions of 10 particles

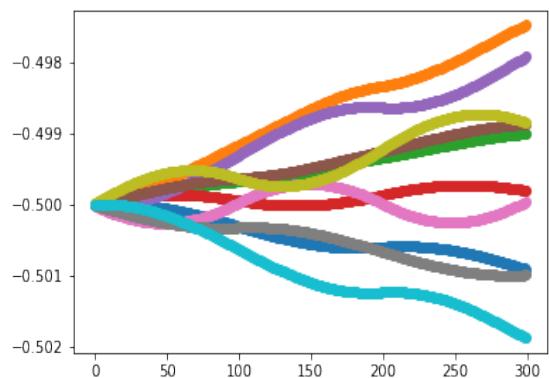


Figure 11:  $x$ -component of positions

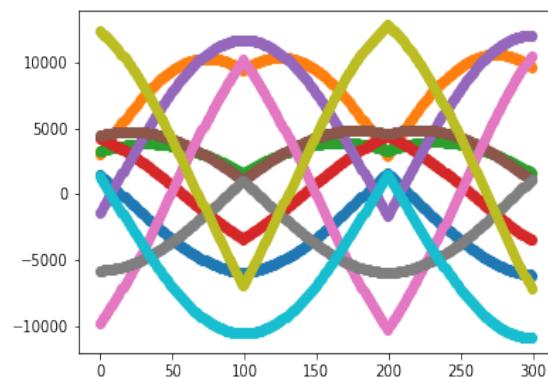


Figure 12:  $x$ -component of velocities

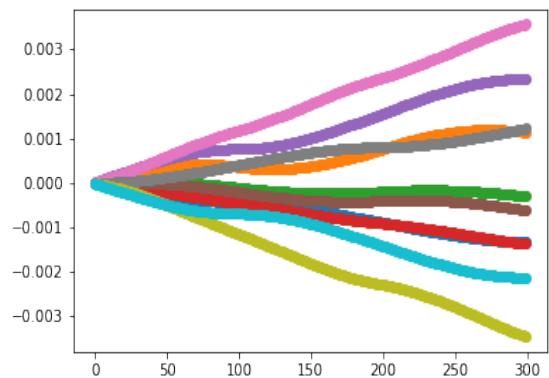


Figure 13:  $y$ -component of positions

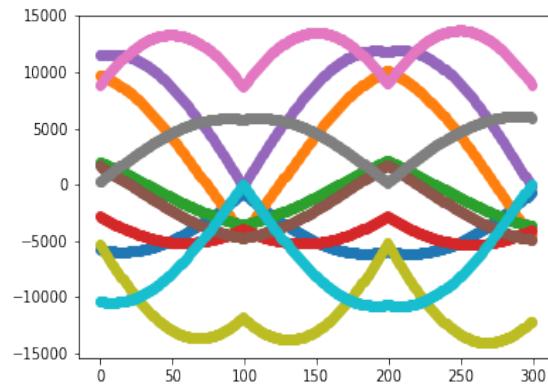


Figure 14:  $y$ -component of velocities

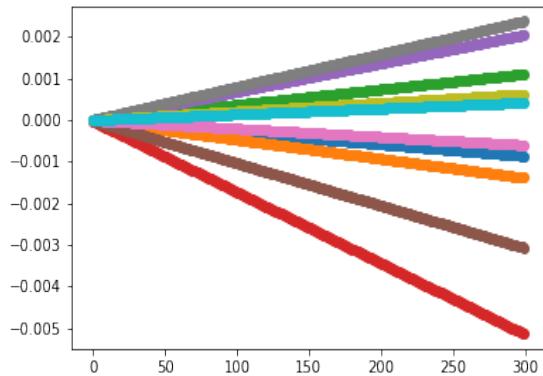


Figure 15:  $z$ -component of positions

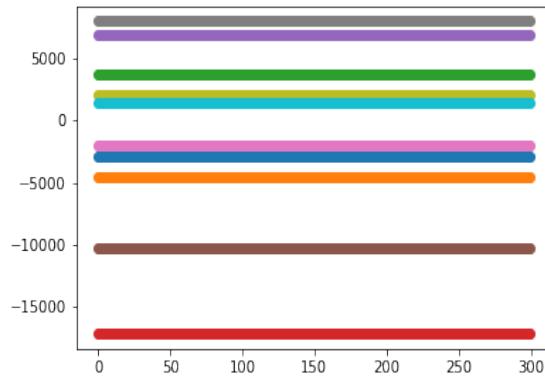


Figure 16:  $z$ -component of velocities

Figures (11), (13) and (15) show that the particles all start at  $[-0.5, 0, 0]$  as intended and spread out based on their initial velocities and velocities after subsequent updates; like in study 1 as the same sampling procedure for positions was used. In figure (16), the  $z$ -component of velocities stay the same as the magnetic field is along the  $z$ -axis (while the electric field is set to 0) and hence the acceleration along the  $z$ -axis is zero. Figures (12) and (14) show three epochs of velocity changes (which can in such cases be shown to be simple harmonic motion as done in section 2.2.1 of chapter 2 in [1]), with 2 sharp changes at the 100<sup>th</sup> and the 200<sup>th</sup> steps when the direction of the magnetic field was reversed. Three epochs of curves can also be observed in figures (11), (13), (15) and (9), and interesting pattern is observed in figure (10) because of this behavior.

#### 4.2.2 Changing Magnetic field strength and direction

##### Configurations of the Electric and Magnetic Fields

For the second part of the study, a slightly different magnetic field configuration was used. The axis of the coil and the current were both varied.

- Magnetic field due to a Helmholtz coil (number of turns: 1000 in each coil, radius: 0.1 m): for 100 steps each
  - first 100A current, orientation along the  $z$ -axis  $[0,0,1]$
  - second -20A current, orientation along the  $x$ -axis  $[1,0,0]$
  - third 50A current, orientation along the  $y$ -axis  $[0,1,0]$ .
- Electric field constantly set to 0.

The positions of 1 and 10 particles in the simulation are plotted respectively in figures (17) and (18).

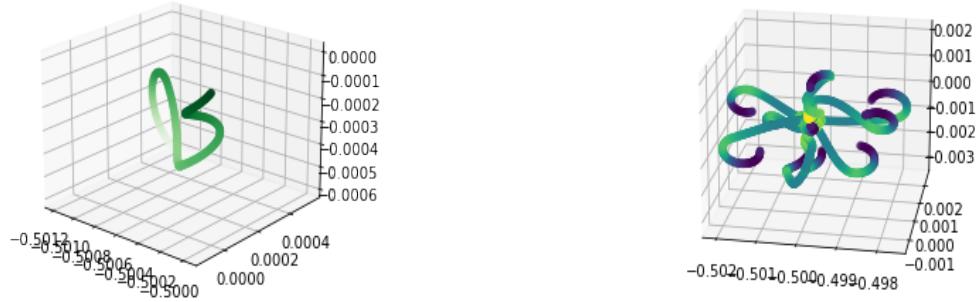


Figure 17: position of a particle

Figure 18: positions of 10 particles

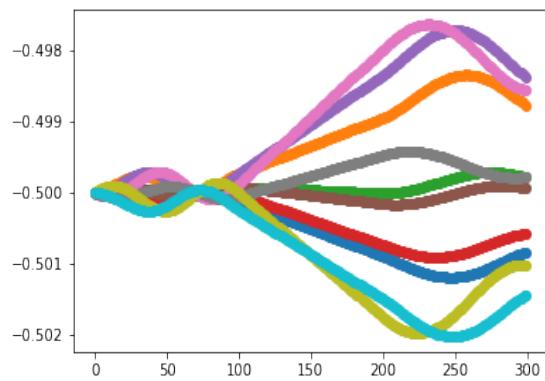


Figure 19:  $x$ -component of positions

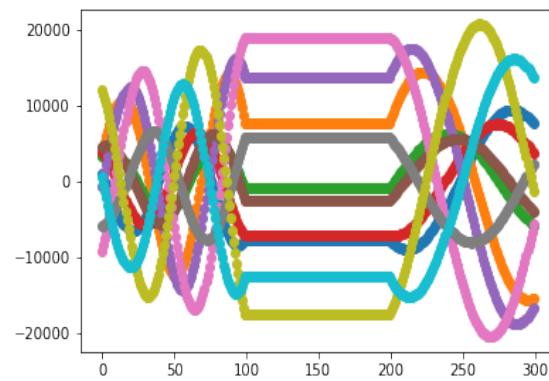


Figure 20:  $x$ -component of velocities

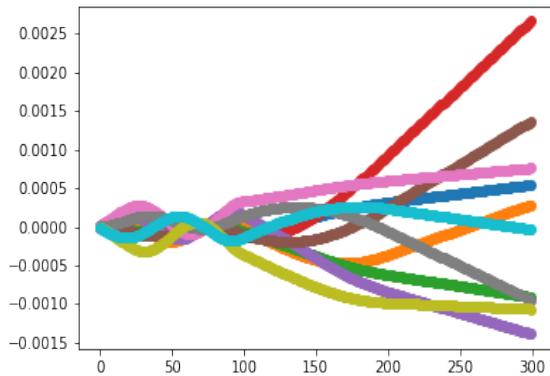


Figure 21:  $y$ -component of positions

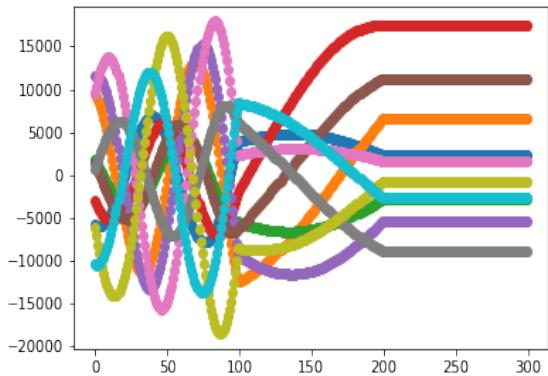


Figure 22:  $y$ -component of velocities

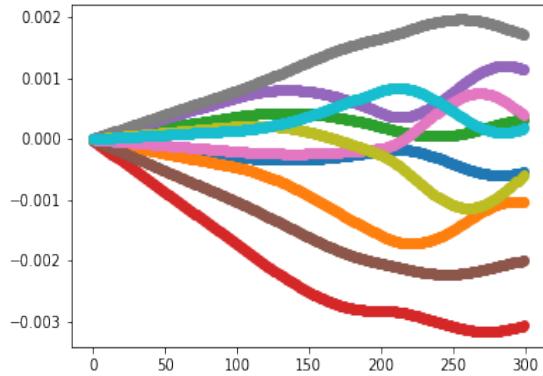


Figure 23:  $z$ -component of positions

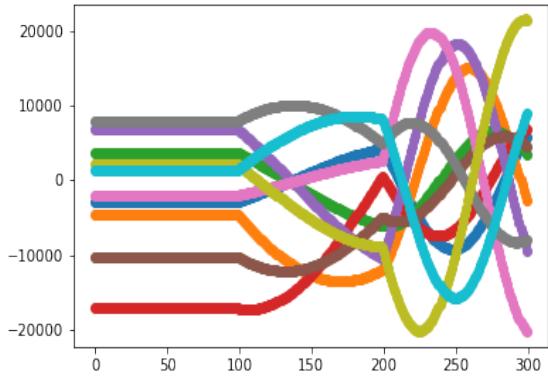


Figure 24:  $z$ -component of velocities

Like in the first part of the study figures (19), (21) and (23) also show that all the particles start at  $[-0.5, 0, 0]$ . Figures (20), (22) and (24) show 1 epoch of constant velocities when the magnetic field is along the respective axes (and hence the acceleration is zero along the respective axes) and 2 epochs of curved velocities of harmonic motion; as discussed in the first part of the study, when the magnetic field is along an axis perpendicular to the respective axes. 2 epoch of curves and 1 epoch of straight lines can also be seen in figures (19), (21) and (23). In figure (17), one can almost see that the three sections of the curve are in mutually perpendicular planes while one of the loops is much clearer. This behavior creates some interesting pattern in figure (18).

### 4.3 Study 3- Parabolic distributed speeds and Helmholtz coils

#### Sampling of the Initial Positions and Velocities of the particles

The following sampling data was used in the study:

- Speeds are sampled from a parabolic distribution with an equivalent temperature of 10000 K for a Maxwellian distribution.
- Velocity directions are sampled from uniform distribution.
- Positions are sampled such that all particles start at 0.5 m distance from the center [0, 0, 0] (Particles injected or reflected from the walls of the chamber may be considered for reference).
- Positions are based on uniform distribution sampling of the position vector.

The mean speed of a Maxwellian distribution is given by

$$\langle v \rangle = \sqrt{\frac{8RT}{\pi M}} = \sqrt{\frac{8 \cdot 8.31446261815324 \text{ J K}^{-1} \text{ mol}^{-1} \cdot 10000 \text{ K}}{\pi \cdot 1.008 \times 10^{-3} \text{ g mol}^{-1}}} = 14492.952993825971 \text{ m s}^{-1}$$

and the root mean square speed is given by  $v_{rms} = \sqrt{\frac{3RT}{M}}$ . Hence the variance,  $\sigma^2 = \langle v^2 \rangle - \langle v \rangle^2$  giving the standard deviation

$$\sigma = \sqrt{\frac{RT}{M}} \sqrt{3 - \frac{8}{\pi^2}} = \sqrt{\frac{8.31446261815324 \text{ J K}^{-1} \text{ mol}^{-1} \cdot 10000 \text{ K}}{1.008 \times 10^{-3} \text{ g mol}^{-1}}} \cdot \sqrt{3 - \frac{8}{\pi^2}} = 13438.549997326772 \text{ m s}^{-1}$$

The speeds for the parabolic distribution were sampled using the Symmetric Beta distribution as defined in the SciPy library (`scipy.stats.rdist[5]`) with parameter  $c = 4$  which gives a half parabolic distribution. Input parameters `loc = 14492.952993825971`, `scale = 30049.5113130523` were used, which produces samples from the distribution with mean  $14492.952993825971 \text{ m s}^{-1}$  and standard deviation  $13438.549997326772 \text{ m s}^{-1}$ . The value for the scale was obtained by iterating until the value close to the required was produced until the maximum precision used by the library.

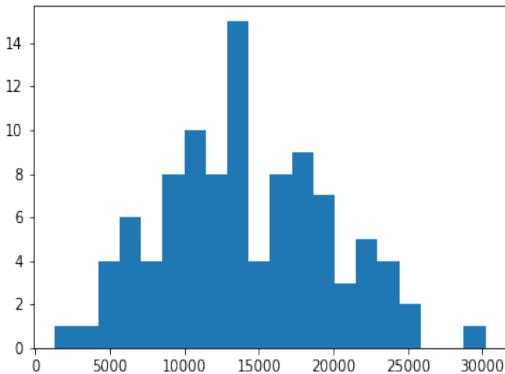


Figure 25: Maxwellian speeds of study 2

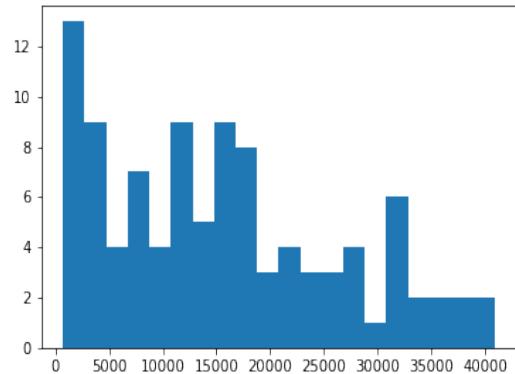


Figure 26: Parabolic speeds of study 3

On the left in figure (25) the histogram of Maxwellian sampled speeds from study 2 are plotted. On the right in figure (26) the histogram of speeds sampled from a Parabolic

distribution are plotted. Both distributions have the same mean and standard deviation. One can observe that a parabolic distribution approximates half of the Maxwellian distribution, which is why it could be interesting. It would also be interesting to have a double parabolic distribution that would approximate the entire Maxwellian distribution.

#### 4.3.1 Reversed Magnetic field

For the first part of this study, the same field configurations as in the first part of study 2 were used. As the fields used in the first part of this study are identical to the fields used in the first part of study 2, similar evolution behavior is expected (and observed) and hence the components of positions and velocities are not plotted and discussed; although they have been generated.

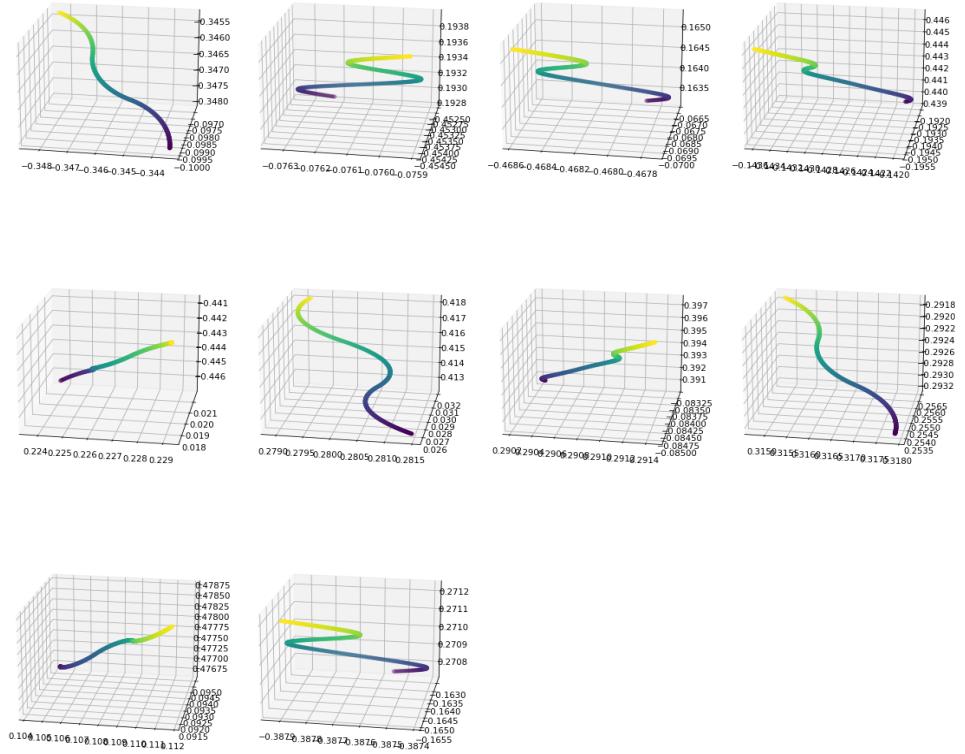


Figure 27: positions of 10 particles

Plotting the positions of the particles in figure (27), shows three epochs of curves for each particle as in the first part of study 2. Although, particles now all don't start in the same position as a different sampling procedure was used for the initialization of the positions.

#### 4.3.2 Changing Magnetic field strength and direction

For the second part of the study, the same field configurations as in the second part of study 2 were used. As the fields used in the second part of this study are identical to the fields used in the second part of study 2, similar evolution behavior is expected (and observed) and hence the components of positions and velocities are not plotted and discussed; although they have been generated.

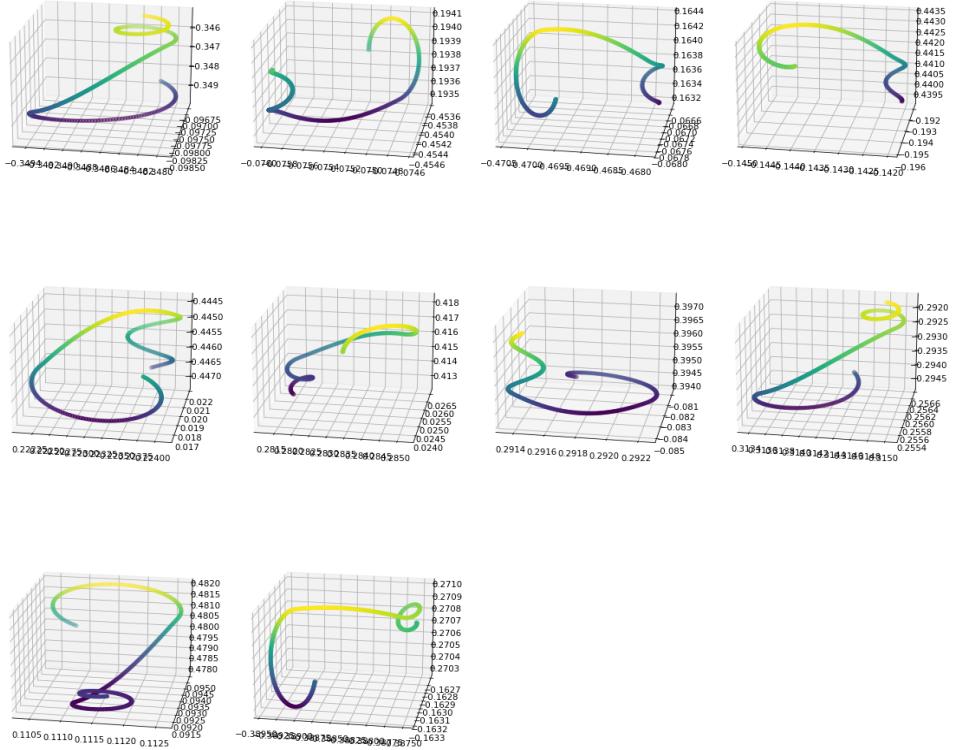


Figure 28: positions of 10 particles

Plotting the positions of the particles in figure (28), shows three circular curves of different

orientations and radius; which is more clear for some particles than others, as in the second part of study 2. However, since the initial positions and velocities of the particles were sampled differently in study 3 than in study 2, a wider variety of trajectories are observed in the second part of study 3.

## 5 Prospects of Plasma Simulation Studies

Further studies could be done with the available infrastructure or a similar one, including studies that really help one use plasma devices better during plasma processes such as those in surface engineering practices. Some interesting studies that could be done might be:

1. Define new field configurations; either analytical or based on expressions that mimic coils or instruments used to create fields.
2. Vary the fields (either slowly or abruptly) and see how the particles evolve.
3. Define new distributions to sample the initial position and velocity distributions.
4. Use different distributions in the same study to mimic particles entering the chamber through multiple routes.
5. Save the states of the particles at some point in the study and use that as initial distribution for another study.
6. Allow functionality to remove particles from the plasma to mimic absorption of particles in coating processes or ejection particles from the chamber.
7. Define interactions between the particles.
8. Create an interactive plasma simulation which keeps running on a computer and the user input can change the particles, fields and other properties of the simulation; so that it is like continuously running a plasma device in a lab.
9. Study the effect of plasma temperature on certain plasma processes.

## 6 Conclusions

Study 1 mostly served the purpose of showing that the program works well. In study 1, the magnetic field was kept constant, while electric field was varied as one might in a lab experiment by scaling by certain numbers. Study 2 was about studying how one can control particles in a plasma by varying the magnetic field. In the first part of study 2, it was observed that one can trap the particles along the magnetic field. In the second part of study 2, it was observed that by changing the direction of the magnetic field, one can trap the particles in any direction. Possibly using a coil (like a Helmholtz coil setup) whose orientation can be changed, one can create a magnetic trap for a plasma chamber. This understanding is made clear by study 2. By controlling the current in the coil and its orientation, the plasma in the chamber can be controlled very well. In study 3, a different initial distribution of the velocities and positions for particles when they enter the chamber, were used. Initial distribution of positions describes where the particles are injected into the chamber. For example, maybe they are all injected at the same position through a port or valve like in study 2, or maybe they are sprayed through the walls of the chamber or perhaps one studies

the flux of particles reflected off of the wall of the chamber; like in study 3. The velocity distribution of the particles entering the chamber also help one understand how the particles were prepared before entering the chamber; may be they were accelerated in an electric field, and also understand parameters like plasma temperature. In these studies the fields were changed in a few epochs so that these changes can be clearly noticed in the plots. Other strategies like changing the fields smoothly or slowly, would be difficult to observe well in the plots; although they would be very interesting from a computational perspective. With some extensions to a project like this, one could perform computational studies and understand optimization techniques to model and operate the behavior of a real plasma device; which would be very useful in operating real plasma devices and understanding plasma processes in surface engineering techniques.

## 7 Funding

No funds, grants, or other support was received.

## 8 Conflict of Interest

The authors declare that they have no conflicts of interest.

## 9 Open Access

## References

- [1] Chen, F. F. (1984). *Introduction to plasma physics and controlled fusion* (Vol. 1, pp. 19-51). New York: Plenum press.
- [2] Boris, J. P. (1973). Relativistic Plasma Simulation-Optimization. In *4th Conference on Numerical Simulation of Plasma* (p. 3).
- [3] 18BME2104. *MagneticMirror*. GitHub.  
<https://github.com/18BME2104/MagneticMirror>
- [4] SciPy's Maxwellian distribution.  
<https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.maxwell.html>
- [5] SciPy's Symmetric Beta distribution.  
<https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.rdist.html>