Magnetic field line Diffusion in Plasma Reconnection

Project Report

Devananth V

EP20BTECH11004

Engineering Physics,
Department of Physics

भारतीय प्रौद्योगिकी संस्थान हैदराबाद
Indian Institute of Technology Hyderabad

Supervisor: Kirit Kumar Makwana

April 2023

Contents

Abstract

This is the report on the project ”Magnetic field line diffusion in plasma reconnection”, conducted from August 2022 to April 2023 under the guidance of Assistant Professor Kirit Kumar Makwana.

Magnetic field lines are continuously created and destroyed inside a turbulent plasma due to magnetic reconnection. In such a turbulent system, the particles are chaotic and diffuse. We try to find what happens to magnetic field lines in such a plasma system. If they do diffuse, we also try to find the nature of this diffusion.

We use Python code to plot magnetic field lines from simulated magnetic field vector data. We then observe the field lines by plotting them. We also look at field lines passing through a local initial point and how they disperse with distance from the initial point.

Introduction

1.1 Magnetic Field Lines in Plasma

Plasma is one of the four states of matter made up of ions. Magnetohydrodynamics(MHD) treats the plasma as a single fluid governed by a combination of Maxwell's equations and the Navier–Stokes equations. Plasma turbulence produces a tangled magnetic field on its own. If the magnetic field contains much energy, an instability might generate plasma flow, causing the field to reorganize. This phenomenon is called Magnetic Reconnection. In dynamic plasmas, the energy conversion between the magnetic field and the plasma flow happens constantly. The plasma's magnetic structures strongly influence the conversion's amount and direction. In resistive MHD, turbulence has also shown to accelerate reconnection.

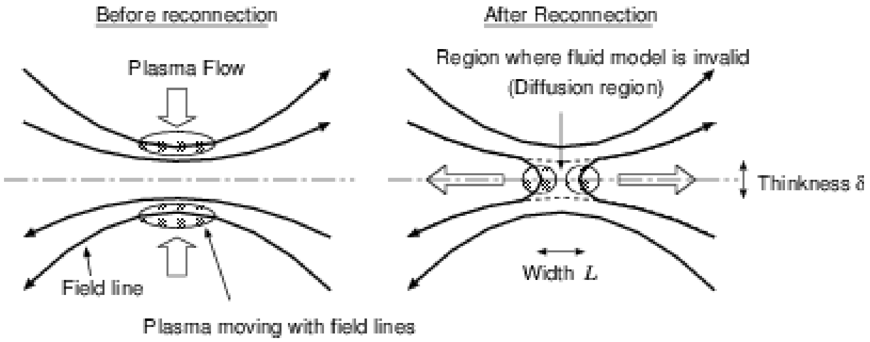


Figure 1.1: Magnetic Reconnection in Plasma

In the diffusion region where the magnetic field is inhomogeneous, the sporadic energy released through magnetic reconnection make field lines chaotic.

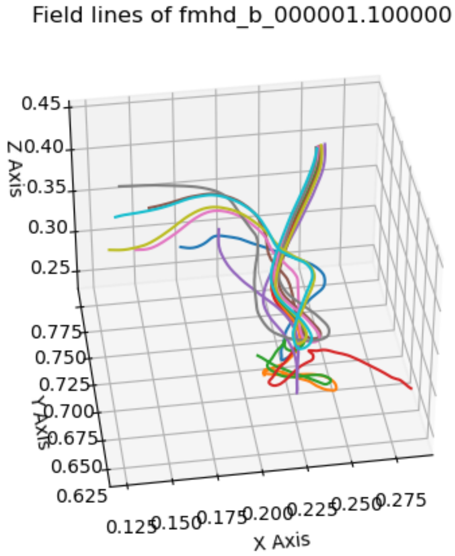


Figure 1.2: Chaotic Field Lines

The Field lines are seen to diffuse

1.2 Diffusion and Types

Diffusion is the net movement of anything generally from a region of higher concentration to a region of lower concentration. Diffusion is driven by a gradient in Gibbs free energy or chemical potential. A distinguishing feature of diffusion is that it depends on particle random walk, and that it results in mixing or mass transport without requiring the directed bulk motion.

1.2.1 Normal Diffusion

In a typical diffusion process, like Brownian motion, described by Einstein and Smoluchowski, the Mean Square Distance(MSD) is linear in time. It is shown by the blue line in the figure below. $[r^2 \propto t]$

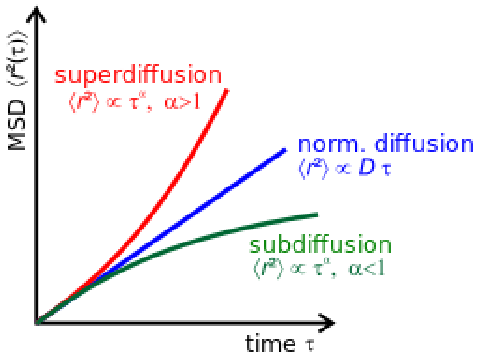


Figure 1.3: Types of Diffusion

1.2.2 Anomalous diffusion

Anomalous diffusion is a diffusion process with a non-linear relationship between the mean squared displacement (MSD) and time. Consider $r^2 \propto t^\alpha$ where $\alpha ! = 1$.

Subdiffusion: Subdiffusion is shown by the green curve, where we can see that $r^2 \propto t^\alpha$ where $\alpha < 1$

Superdiffusion: Superdiffusion is shown by the red curve, where we can see that $r^2 \propto t^\alpha$ where $\alpha > 1$

Richardson Diffusion: The Richardson diffusion describes the explosive growth of the separation of particles in a turbulent medium. It is inferred from previous fluids experiments. It is a type of superdiffusion with $\alpha \sim 3$.

1.2.3 Changes Made

In this project, we are taking ”l”, where ”l” is the distance traveled from the initial point, as analogous to ”t” given above.

Also, in this report, we are taking diffusion in terms of ”r” and not ”r^2”, by which α becomes $\frac{\alpha}{2}$.

Methodology - Python Code

2.1 Magnetic Field Lines

2.1.1 Field Line Equation

We know that the field line is tangential to the field vector. Here, $\vec{B}$ is the field vector, $\vec{dr}$ is the differential element for the position vector, and $\vec{dl}$ is the field line differential.

$$\hat{b} = \frac{\vec{B}}{|B|} \tag{2.1}$$

$$\vec{dr} = \hat{b}dl \tag{2.2}$$

$$dx\hat{i} + dy\hat{j} + dz\hat{k} = b_x dl\hat{i} + b_y\hat{j} + b_z dl\hat{k} \tag{2.3}$$

Equating the coefficients, we get 3 Ordinary Differential Equation(ODE):

$$\frac{dx}{dl} = b_x = \frac{B_x(x, y, z)}{|B|} \tag{2.4}$$

$$\frac{dy}{dl} = b_y = \frac{B_y(x, y, z)}{|B|} \tag{2.5}$$

$$\frac{dz}{dl} = b_z = \frac{B_z(x, y, z)}{|B|} \tag{2.6}$$

By solving the given ODEs we get x=x(l), y=y(l), and z=z(l), using which we can trace field lines for different values of 'l'.

2.1.2 Runge-Kutta Method

We use the 4th Order Multivariable Adaptive Step-size Runge-Kutta Method, to solve the ODEs. In fourth order Runge-Kutta, we use the equation given below to integrate the ODEs.

$$k_1 = hf(x_n, y_n), \tag{2.7}$$

$$k_2 = hf(x_n + \frac{h}{2}, y_n + \frac{k_1}{2}), \tag{2.8}$$

$$k_3 = hf(x_n + \frac{h}{2}, y_n + \frac{k_2}{2}), \tag{2.9}$$

$$k_4 = hf(x_n + h, y_n + k_3), \tag{2.10}$$

$$y_{n+1} = y_n + \frac{k_1}{6} + \frac{k_2}{3} + \frac{k_3}{3} + \frac{k_4}{6} + O(h^5). \tag{2.11}$$

In the said method the step-size is varied on the situation to decrease computation time and increase accuracy. We decrease the step-size when the error is above desired error. Otherwise we increase the step-size. It is made multivariable by considering the variables as vectors.

2.1.3 Trilinear Interpolation

To use the Runge-Kutta method we require continuous values, however, our data is discrete. To solve this, we use Trilinear Interpolation(3D-Interpolation) to get the required points. Trilinear Interpolation is the process of linearly interpolating points within a box (3D) with values given at the box's vertices. Trilinear interpolation is implemented by taking linear interpolation of two bilinear interpolations.

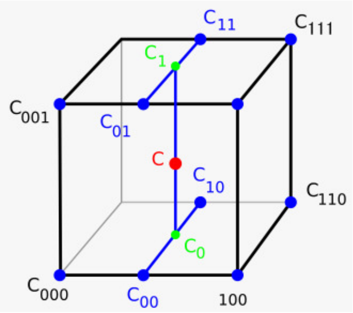
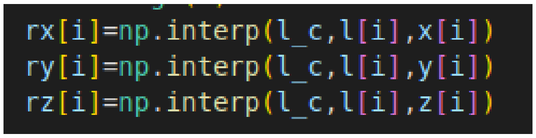


Figure 2.1: c_{000}, c_{100}... etc.,are the known values, which is used to form the corner points of the 3D box.

2.2 Diffusion of Magnetic Field Lines

We use the magnetic field line data from above to find the diffusion of magnetic field lines. For this we consider magnetic field lines originating from the same locality and measure the mean distance between the field lines for each value of ”l”. Here, we run into a problem. In our data, the values of ”l” may be different for each field line. This is because we are using variable step-size in the Runge-Kutta method. To overcome this, we use the **interp()** function available in the NumPy library.

The interp(x,xp,yp) function returns the corresponding values of y(within yp) for the given x(within xp).

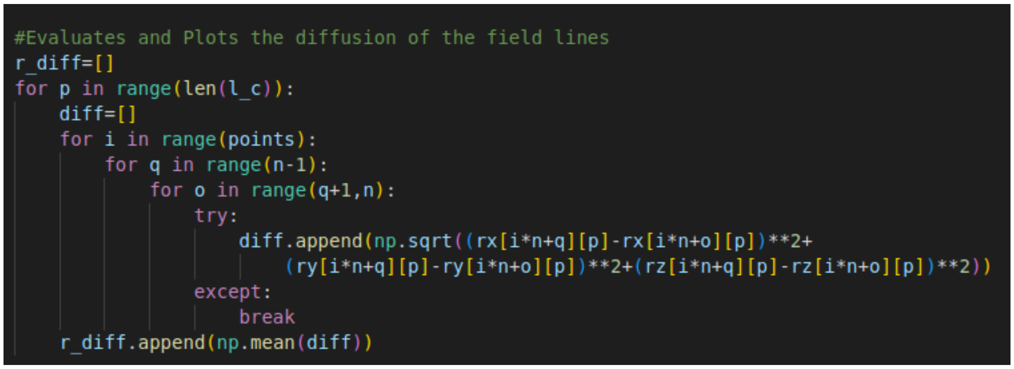


```
rx[i]=np.interp(l_c,l[i],x[i])
ry[i]=np.interp(l_c,l[i],y[i])
rz[i]=np.interp(l_c,l[i],z[i])
```

Figure 2.2: 1-D Interpolation Function

To find the diffusion of magnetic field lines in the plasma, we take ’n’ different localities with ’m’ magnetic field lines each. For each value of ”l”, we find the mean distance between the ’m’ field lines for the ’n’ points and take their average. We do this to get an accurate representation of the diffusion of the plasma.

The code snippet is given below.



```
#Evaluates and Plots the diffusion of the field lines
r_diff=[]
for p in range(len(l_c)):
    diff=[]
    for i in range(points):
        for q in range(n-1):
            for o in range(q+1,n):
                try:
                    diff.append(np.sqrt((rx[i*n+q][p]-rx[i*n+o][p])**2+
                        (ry[i*n+q][p]-ry[i*n+o][p])**2+(rz[i*n+q][p]-rz[i*n+o][p])**2))
                except:
                    break
    r_diff.append(np.mean(diff))
```

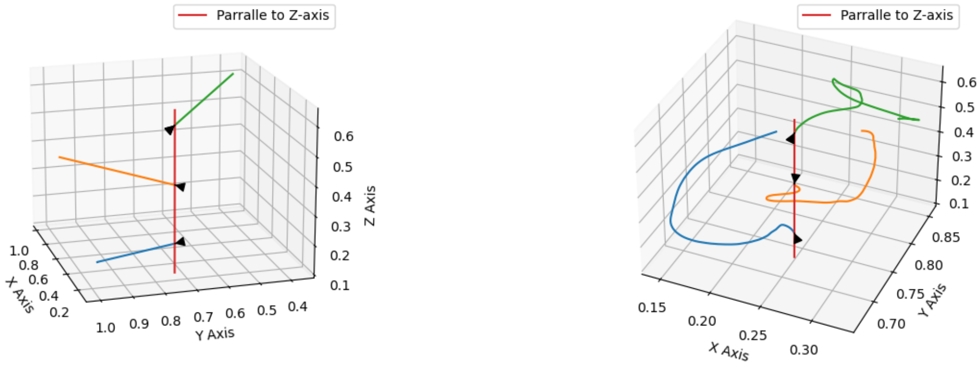
Figure 2.3: Evaluating Diffusion

Observations

3.1 Magnetic Field lines

3.1.1 Field lines at a Seperation

The images given below show the form of the field lines at 't=0' and 't=100000' with initial points in a straight line parrallel to z-axis(x=0.2 and y= 0.7) at a seperation.



(a) Field lines at T=0

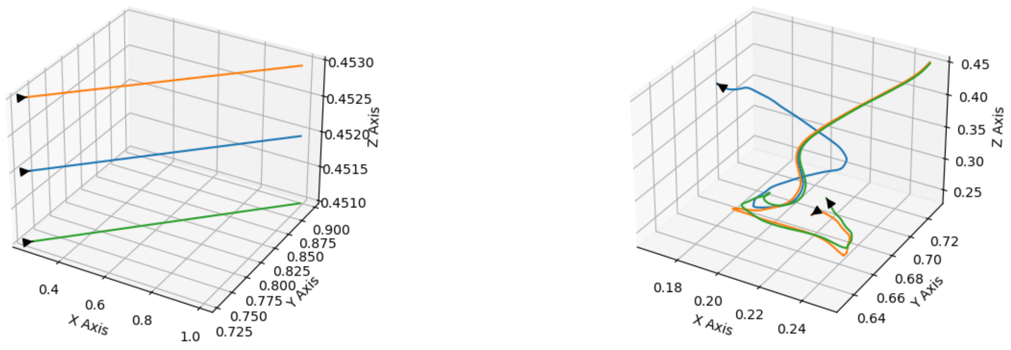
(b) Field Lines at T=100000

Figure 3.1: Field lines around a straight line parrallel to z-axis

We observe parrallel field lines at 't=0' which becomes chaotic at 't=100000', due to plasma turbulance. The field lines at 't=0' are straight lines rotated spherically about the line parrallel to the z-axis as the magnetic field is of a current sheet. However, at 't=100000', it loses its shape due to turbulence.

3.1.2 Adjoining Field Lines

The images given below show the form of the field lines at 't=0' and 't=100000' with adjoined initial points.



(a) Field lines at T=0

(b) Field Lines at T=100000

Figure 3.2: Field lines with initial points close to each other

We observe parallel field lines at 't=0', which becomes chaotic at 't=100000'. In Figure(a), we observe that field lines close to each other form straight parallel lines, when there is no turbulence(t=0). However, at 't=100000', we observe that the field lines diffuse chaotically due to turbulence. We can observe that the field lines travels without dispersing much initially but then disperses rapidly.

3.2 Diffusion of Field Lines

To find the mean diffusion of magnetic field line in the plasma we take 3 diffrent localities with 10 magnetic field lines each. The below given image shows the plot of ”log(Diffusion)” vs ”log(l)”.

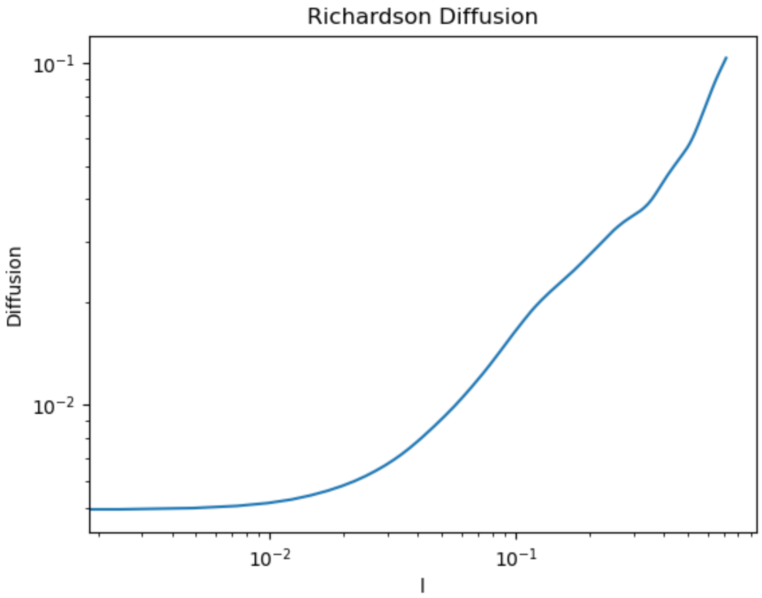


Figure 3.3: Plot of diffusion vs ”l”

The graph shows that diffusion initially does not change much with ’l’, however, after some distance the plot becomes almost a straight line with a positive slope α.

We can try to find the slope α of this line.

The straight line is of the form $log(r) = \alpha log(l) + c$, where α is the slope and c is the y-intercept. We consider two end points (x_1, y_1) and (x_2, y_2).

$$log(y_1) = \alpha log(x_1) + c \tag{3.1}$$

$$log(y_2) = \alpha log(x_2) + c \tag{3.2}$$

Taking Eqn (3.2) - Eqn (3.1) we get:

$$log(\frac{y_2}{y_1}) = \alpha log(\frac{x_2}{x_1}) \tag{3.3}$$

$$\alpha = \frac{log(\frac{y_2}{y_1})}{log(\frac{x_2}{x_1})} \tag{3.4}$$

We take (x_1=0.0494441, y_1=0.00927756) and (x_2=0.200011, y_2=0.351961) from the plot to calculate α.

Calculating we get α= **0.954066858**

Since we get $\alpha > \frac{1}{2}$, it means that the field lines show **Superdiffusion**.

3.3 Results and Discussion

Slope/Power α= **0.954066858**

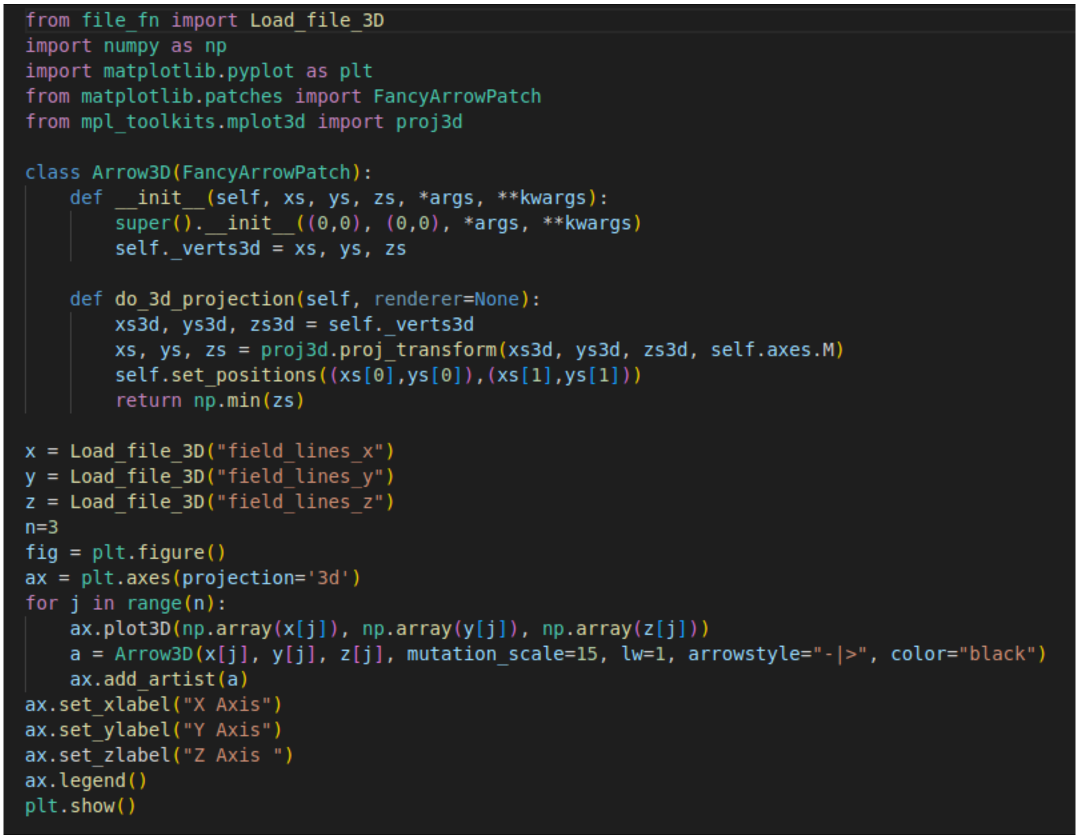
Diffusion $r \propto l^{0.954066858}$

The Magnetic Field lines show **Superdiffusion**.

The required alpha value of 1.5 for diffusion was not achieved due to the dataset's restriction to only the earlier stages of the phenomenon under study. To ensure accurate analyses, it is necessary to apply the code to a dataset that includes later stages of the phenomenon.

Appendix

4.1 Code to Plot 3D Field lines with Directional Arrows



```python
from file_fn import Load_file_3D
import numpy as np
import matplotlib.pyplot as plt
from matplotlib.patches import FancyArrowPatch
from mpl_toolkits.mplot3d import proj3d

class Arrow3D(FancyArrowPatch):
    def __init__(self, xs, ys, zs, *args, **kwargs):
        super().__init__((0,0), (0,0), *args, **kwargs)
        self._verts3d = xs, ys, zs

    def do_3d_projection(self, renderer=None):
        xs3d, ys3d, zs3d = self._verts3d
        xs, ys, zs = proj3d.proj_transform(xs3d, ys3d, zs3d, self.axes.M)
        self.set_positions((xs[0],ys[0]),(xs[1],ys[1]))
        return np.min(zs)

x = Load_file_3D("field_lines_x")
y = Load_file_3D("field_lines_y")
z = Load_file_3D("field_lines_z")
n=3
fig = plt.figure()
ax = plt.axes(projection='3d')
for j in range(n):
    ax.plot3D(np.array(x[j]), np.array(y[j]), np.array(z[j]))
    a = Arrow3D(x[j], y[j], z[j], mutation_scale=15, lw=1, arrowstyle="-|>", color="black")
    ax.add_artist(a)
ax.set_xlabel("X Axis")
ax.set_ylabel("Y Axis")
ax.set_zlabel("Z Axis ")
ax.legend()
plt.show()
```

Figure 4.1: Arrows are drawn in the 3D field line using the first and second point of the field line as reference

4.2 Change in Field Lines with 't'

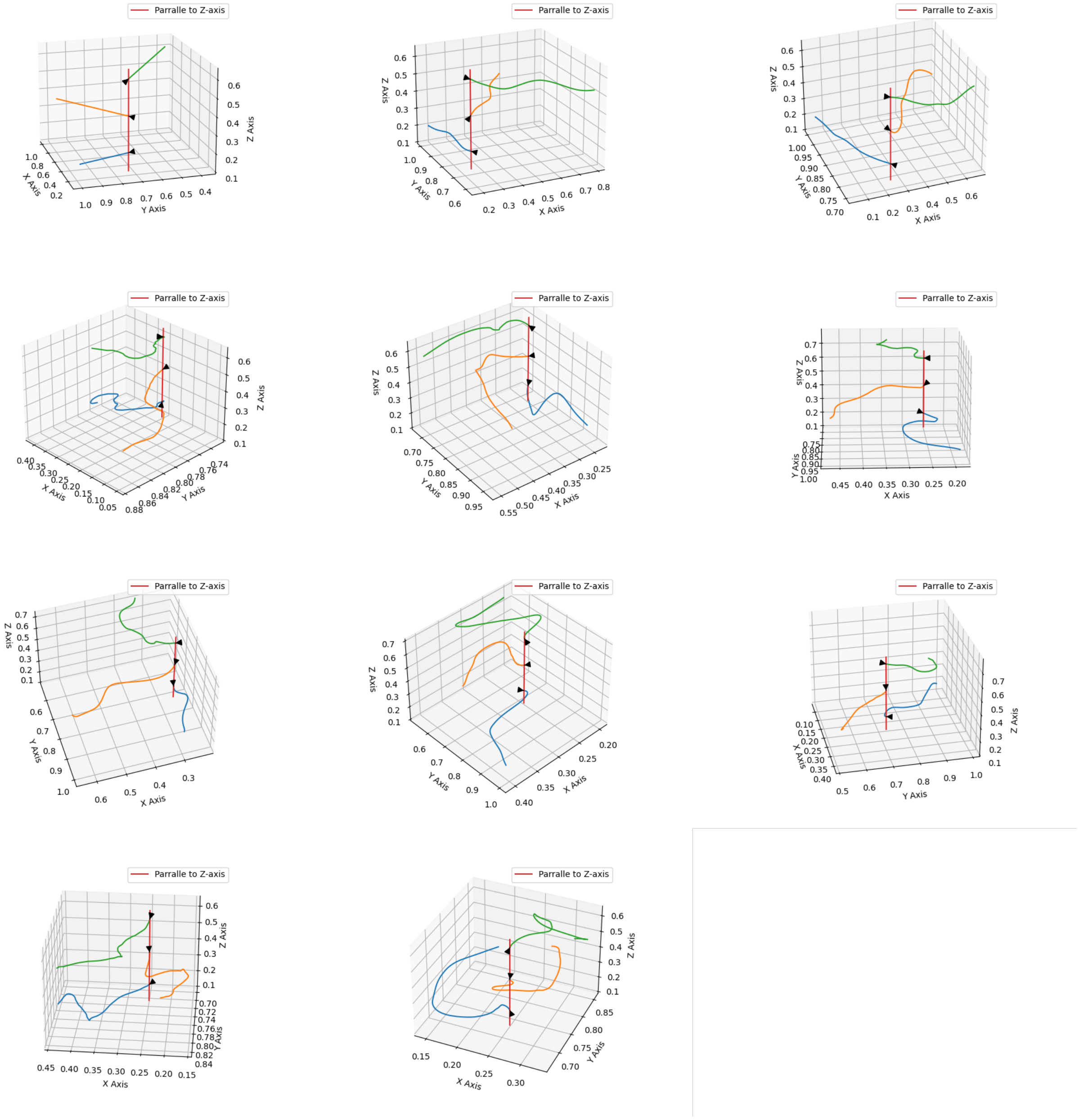


Figure 4.2: 't' increases from right to left and then down

4.3 Plot of Magnetic Field Lines Used in Finding Diffusion

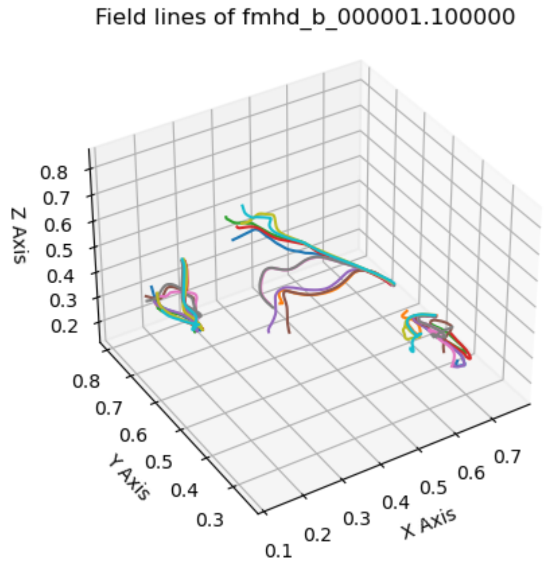


Figure 4.3: Shows the 3 initial points with 10 field lines each.

4.4 Acknowledgement

I am incredibly grateful to Asst Prof. Kirit Kumar Makwana for providing me with an opportunity of working on this project and guiding me through it. I want to thank him for his unwavering support and patience throughout the project and his willingness to answer my numerous questions and provide timely feedback. This endeavor has taught me a lot and is an invaluable asset for my future.

References

1. William H. Press, Saul A. Teukolsky, ”Adaptive Stepsize Runge-Kutta Integration”, Published by the American Institute of Physics in 1992.

2. Snehanshu Maiti, Kirit D. Makwana, Heshou Zhang, and Huirong Yan, ”Cosmic-ray Transport in Magnetohydrodynamic Turbulence”, Published in the The Astrophysical Journal by the American Astronomical Society on February 2022.

3. Peera Pongkitiwanichakul, Kirit D. Makwana, and David Ruffolo, ”Driving reconnection in sheared magnetic configurations with forced fluctuations”, Published in Physics of Plasmas on February 2018.

4. Abhinav Poddar, Engineering Physics, Department of Physics, Indian Institute of Technology Hyderabad, ”Tracing of field lines from MHD dataset”, April 2022.

5. Fig: 1.1:”https://www.quora.com/What-are-the-necessary-conditions-for-magnetic-reconnection”

6. Fig: 1.3: https://en.wikipedia.org/wiki/Anomalous_diffusion

7. Fig: 2.1: https://en.wikipedia.org/wiki/Trilinear_interpolation

7. ”https://en.wikipedia.org/wiki/Magnetic_reconnection”,”https://en.wikipedia.org/wiki/Plasma_(physics)”

8. ”https://royalsocietypublishing.org/doi/10.1098/rsta.2014.0144”

9. ”https://numpy.org/doc/stable/reference/generated/numpy.interp.html”

10.”https://www.nature.com/articles/nature10827.pdf?pdf=reference”

11.”https://link.springer.com/article/10.1007/s41116-022-00032-9”

12.”https://www.compadre.org/nexusph/course/Diffusion_and_random_walks”