Physics 305 – Computational Physics, Fall 2020
Term Project
Full project submission Due Date: Tuesday December 15, 5pm
Presentation Phase: November 30 - December 11

The program in your term project can be either submitted as a python program or ipython notebook, where the latter is preferred. The program, an explanation of what the program does, along with answers to **all** questions asked should be uploaded to d2l.

You are expected to write a term paper (in word or Latex) on your project that discusses the problem you are trying to solve, the basic equations that govern the problem, includes plots that show the solutions, and describes the solution and the numerical method involved. In addition, you must demonstrate that your solution is correct by showing that the code converges at the expected order. If your code does not converge at the expected order you should try to identify potential reasons for why this is the case. You are expected to work on your term project by yourself..

Your term project will receive full credit **only** if: (a) the program runs successfully without errors using python 3, (b) the programs have explanatory comments and variable names that identify with the problem equations you are trying to solve, (c) give the correct output, and (d) demonstrate the validity of the solution through convergence plots. No credit will be given to late term projects.

The term paper is as important as the code (50% of the term project credit will go to the code and the other 50% to the paper). Answers to the questions and analysis requested below should be elaborated in the report. Plots should be clearly labeled and be properly described in the report, and not just shown. You will need to explain what each and every plot demonstrates. A polished paper written in word or LaTex (preferred, e.g. please try overleaf) is expected to get full credit.

Note: Before you present results from numerical integrations that answer the questions in the project, it is critical to *first* perform the convergence tests for one case, and to estimate errors. This will tell you how small a step size is necessary for accurate solutions. Only after errors are estimated, does it make sense to run your code for producing results that help you learn more about the system you study.

I. CHARGED PARTICLE MOTION IN A MAGNETIC MAGNETIC DIPOLE

You will be studying magnetic trapping of charged particles, i.e., motion of charged particles confined in a magnetic dipole similar to that of the Earth. The equations of motion (EOM) for charged particles (neglecting radiation) are simple (adopting Gaussian units)

$$m\frac{d^2\vec{r}}{dt^2} = \frac{1}{c}q\vec{v}\times\vec{B} \tag{1}$$

which in 3D can be written as

$$\begin{aligned} m\frac{d^2x}{dt^2} &= \frac{1}{c}q(v^yB^z - v^zB^y) \\ m\frac{d^2y}{dt^2} &= \frac{1}{c}q(v^zB^x - v^xB^z) \\ m\frac{d^2z}{dt^2} &= \frac{1}{c}q(v^xB^y - v^yB^x) \end{aligned} \tag{2}$$

To solve this systen of ODEs you need to supply initial conditions. Initial conditions are necessary for the initial position x, y, z, and the initial velocity v^x, v^y, v^z. The position will be located close to one of the poles of the magnetic dipole. You will be varying the initial velocity to test what particles are trapped and which escape (i.e., are said to be in the loss cone).

The basic reason for trapping particles has to do with the existence of adiabatic invariants and energy conservation. In particular the particle magnetic moment is (approximately) conserved

$$\mu = \frac{mv_\perp^2}{2B}. \tag{3}$$

where $v_\perp$ is the component of the velocity perpendicular to the magnetic field. Thus, if the particle enters an area where B increases, then for μ to remain a constant $v_\perp$ must increase. But, energy conservation implies

$$E_{\rm tot} = \frac{1}{2}mv_{||}^2 + \frac{1}{2}mv_\perp^2. \tag{4}$$

Thus, if $v_\perp$ increases $v_{||}$ must decrease to conserve energy. In fact, it turns out that $v_{||}$ can also change sign. For this to happen we need special conditions. Let the magnetic mirror ratio r be defined as

$$r = \frac{B_{\rm max}}{B_{\rm min}} \tag{5}$$

Particles whose pitch angle (the angle between the velocity and the magnetic field) is larger than a critical value are trapped. In particular, trapped particles have velocities

$$\frac{v_\perp}{v} > \frac{1}{\sqrt{r}} \tag{6}$$

In this project you will be testing all of this theory. But, first you need to construct a magnetic dipole. The simplest possible magnetic dipole can be constructed by the magnetic field from a current loop. The magnetic field of one current loop (with symmetry axis the z-axis) can be approximated by the following equations

$$\begin{aligned}
B^x(a,I_0,x,y,z) &= \frac{3\pi a^2 I_0 xz\left(43a^4 + a^2\left(51\left(x^2+y^2\right)+16z^2\right) + 8\left(x^2+y^2+z^2\right)^2\right)}{8\left(a^2+x^2+y^2+z^2\right)^{9/2}}\\
B^y(a,I_0,x,y,z) &= \frac{3\pi a^2 I_0 yz\left(43a^4 + a^2\left(51\left(x^2+y^2\right)+16z^2\right) + 8\left(x^2+y^2+z^2\right)^2\right)}{8\left(a^2+x^2+y^2+z^2\right)^{9/2}}\\
B^z(a,I_0,x,y,z) &= \\
&\frac{\pi a^2 I_0\left(46a^6 + a^4\left(9\left(x^2+y^2\right)+78z^2\right) + 3a^2\left(36z^2\left(x^2+y^2\right) - 15\left(x^2+y^2\right)^2 + 16z^4\right) - 8\left(x^2+y^2-2z^2\right)\left(x^2+y^2+z^2\right)^2\right)}{8\left(a^2+x^2+y^2+z^2\right)^{9/2}},
\end{aligned} \tag{7}$$

where a is the loop radius, and I_0 the current. Setting $a = 1$, $I_0 = 1$ results in an approximate magnetic dipole, whose field-lines of which are shown in Fig. 1.

1. You will use RK4 to integrate numerically the system of ODEs (2) from $\tilde{t} = 0$ and for sufficiently long times. You will need initial conditions to integrate the equations, so first let us discuss these. Initialize particle on the $x - z$ plane near the z-axis, i.e., at $t = 0$, set $x = 5.0$, $y = 0$, $z = 10$.

   - Make a 2D density plot of the magnitude of the magnetic field for $z = 0$ for $x \in [-15, 15]$, $z \in [-15, 15]$. Determine the the magnetic field amplitude near the initial location of the particle, and a characteristic "minimum" value of the field. This will enable you to find what the approximate mirror ratio is, and hence the approximate value of the critical pitch angle.
   - Find the magnetic field components at the initial location of the particle, this determines the local magnetic field vector $\vec{B} = (B^x, B^y, B^z)$ and its magnitude B. Consider the initial velocity $\vec{v} = (v_x, v_y, v_z)$ at the initial location of the particle. The components of the velocity parallel to the B field are given by $v_{||}^i = \left(\vec{v}\cdot\vec{B}\right)\frac{B^i}{B^2}$, i.e.,

$$\begin{aligned}
v_{||}^x &= \left(\vec{v}\cdot\vec{B}\right)\frac{B^x}{B^2}\\
v_{||}^y &= \left(\vec{v}\cdot\vec{B}\right)\frac{B^y}{B^2}\\
v_{||}^z &= \left(\vec{v}\cdot\vec{B}\right)\frac{B^z}{B^2}
\end{aligned} \tag{8}$$

   The components of the velocity perpendicular to the magnetic field are $v_\perp^i = \sum_{j=1}^3(\delta^{ij} - \frac{B^iB^j}{B^2})v^j, i = 1, 2, 3,$ where δ^{ij} is the Kronecker delta, $v^1 = v^x, v^2 = v^y, v^3 = v^z$ and similarly for the magnetic field. In other words

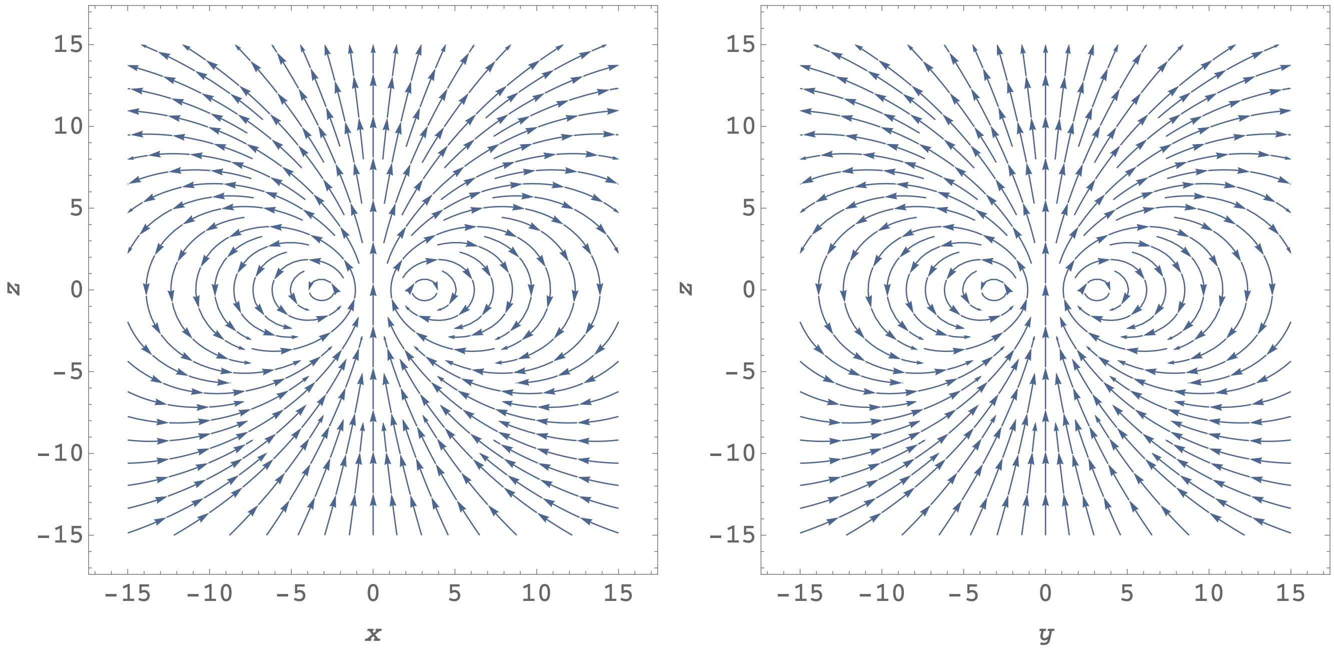

FIG. 1. Magnetic dipole from a current loop magnetic field defined in Eq. (7). The left panel shows that magnetic field lines on the $x-y$ plane, and the right panel on the $y-z$ plane.

$$
\begin{aligned}
v_\perp^x &= \sum_{j=1}^{3}(\delta^{xj} - \frac{B^x B^j}{B^2})v^j = (1 - \frac{B^x B^x}{B^2})v^x - \frac{B^x B^y}{B^2}v^y - \frac{B^x B^z}{B^2}v^z \\
v_\perp^y &= \sum_{j=1}^{3}(\delta^{yj} - \frac{B^y B^j}{B^2})v^j = -\frac{B^x B^y}{B^2}v^x + (1 - \frac{B^y B^y}{B^2})v^y - \frac{B^y B^z}{B^2}v^z \\
v_\perp^z &= \sum_{j=1}^{3}(\delta^{zj} - \frac{B^z B^j}{B^2})v^j = -\frac{B^x B^z}{B^2}v^x - \frac{B^y B^z}{B^2}v^y + (1 - \frac{B^z B^z}{B^2})v^z
\end{aligned}
\tag{9}
$$

Given that setting the components of the particle velocity normal and parallel to the magnetic field requires two equations, you can set one of the 3-components of the particle velocity to 0 initially. As we want the particle to move in the z-direction to traverse the dipole, we can set $v^y = 0$ at $t = 0$. We can also fix v^z and vary v^x to control the ratio $\frac{v_\perp}{v}$. Therefore, the particle velocity components parallel to the B field become

$$
\begin{aligned}
v_{||}^x &= (v^x B^x + v^z B^z)\,\frac{B^x}{B^2} \\
v_{||}^y &= (v^x B^x + v^z B^z)\,\frac{B^y}{B^2} \\
v_{||}^z &= (v^x B^x + v^z B^z)\,\frac{B^z}{B^2}
\end{aligned}
\tag{10}
$$

and the perpendicular component become

$$
\begin{aligned}
v_\perp^x &= (1 - \frac{B^x B^x}{B^2})v^x - \frac{B^x B^z}{B^2}v^z \\
v_\perp^y &= -\frac{B^x B^y}{B^2}v^x - \frac{B^y B^z}{B^2}v^z \\
v_\perp^z &= -\frac{B^x B^z}{B^2}v^x + (1 - \frac{B^z B^z}{B^2})v^z
\end{aligned}
\tag{11}
$$

For our problem we only want to set the perpendicular component at $t = 0$. As stated in Eq. (6) we want to set the ratio $v_\perp$ to v, i.e.,

$$\frac{v_\perp}{v} = \frac{\sqrt{(v_\perp^x)^2 + (v_\perp^y)^2 + (v_\perp^z)^2}}{\sqrt{(v^x)^2 + (v^z)^2}} \tag{12}$$

The z component of the velocity can be set to a reasonable value to traverse the magnetic dipole. The particle starts at height $z = 10$, and hence going back and forth will give a path length of order 20. Thus setting $v^z = 0.01$, will traverse the magnetic dipole in $\sim 20/0.01 = 2000$ time units (assuming the magnetic field was uniform).

To summarize, for the initial velocity we set $v^y = 0$, $v^z = 0.1$ and set v^x using Eqs. (11)-(12) to control the ratio $\frac{v_\perp}{v}$ such that it goes above and below the trapping pitch angle.

- Another important factor that must be considered has to do with the charge-to-mass ratio of the particle. The perpendicular component of the particle makes the particle gyrate around the guiding center of a magnetic field line. Perform a dimensional analaysis of Eq. (1) to determine what the timescale (period) of gyration is. We do not want the gyration timescale to be too small compared to the traversal timescale, because it will be impossible to numerically resolve both the gyration and the traversal/bouncing of the particles within the magnetic bottle. As the gyration timescale depends on the magnetic field (it becomes smaller for larger magnetic fields) you can use the value of the magnetic field magnitude at the initial point to set the the ratio mc/q so that the gyration timescale is at most $1/100$ of the magnetic dipole traversal timescale, otherwise solving the problem accurately could take a very long time.
- To integrate Eq. (2) you will need to cast the equations to first-order form. Implement the RK4 algorithm to solve the system of six first-order equations. You will need to be plotting the orbits x, y, z vs t to see what the particle is doing, and whether you have integrated long enough.
- As mentioned above the kinetic energy of the particle must be conserved, i.e., $E_k/m = (1/2)((v^x)^2 + (v^y)^2 + (v^z)^2)$ as a function of time must be the same as the initial value it starts out with. This is a useful diagnostic to confirm that your numerical implementation is working.
- Vary the initial v^x, and run your code to determine the critical ratio $\frac{v_\perp}{v}$ above which particles remain trapped, while below it they go into the loss cone. Make plots of the particle 3D trajectories, showing trapped orbits and orbits in the loss cone.

Use your judgement as to how small a step size you need to solve this system accurately. If you cannot figure this out from pure thought, experiment with different step sizes. Use $\delta E = |(E_k - E_k(t = 0))/E_k(t = 0)|$ to determine the accuracy. If δE is smaller than 10^{-3} for all integration times, then you have decent accuracy.

2. **Convergence:** Demonstrate that as you decrease the step size h, δE at the final time converges to 0. Make a plot to determine the order of convergence of δE, and discuss why or why not this matches your expectation.

3. **Optional:** you can use analytic solutions to the set of equations you are integrating to ensure that your code works correctly. In particular, you can place the particle in a uniform magnetic field, on the x-axis at some distance from the origin, and give the particle the precise velocity it should have to simply perform a circular orbit. Although optional, this is a good method to ensure your RK4 code works.

4. **Self-convergence**: Use a number of step sizes and make a plot to demonstrate that the code solution for x, y, and z at the final time self-converges.

5. Using the order of convergence you determined, employ Richardson extrapolation to determine an error for the solution for $x(t)$, $y(t)$, $z(t)$ at a time t of your choosing.