Magnetodisc Modelling and Particle Tracing
Laboratory Exercises

N. Achilleos (Physics and Astronomy, UCL);
P. Guio (Physics and Astronomy, UCL);
MDISC service implementation by P. Guio (Physics and Astronomy, UCL)

Introduction

This 'laboratory exercise' is designed to be performed on a computer running the 'Octave' software package. Students are asked to follow the instructions below, which provide information about downloading both model outputs and source code. Once these files are obtained, various components of the source code can be executed, using Octave, in order to visualise the contents of the model output files.

Downloading Model Output Files

The 'model information' that you will explore in this exercise can be found in files which you can download in .zip format from the URL `http://www.ucl.ac.uk/∼ucapnac/MDISC/MDISC.zip`. These are files which have been modified for use with Octave and this exercise. Download and, if necessary, 'unzip' the archive file (MDISC.zip).

(For future work, if you are using Matlab rather than Octave, you can download files from the VESPA web interface (further instructions for this route are given in the Appendix)).

Model Visualisation Exercises

At this point, you should have a collection of files on your computer, including two .mat files which represent magnetodisc models for the Jovian magnetosphere in a compressed and expanded state - these files are named 'jup_mdisc_kh3e7_rmp60fix.mat' (60 $\mathrm{R_J}$ subsolar magnetopause), and 'jup_mdisc_kh3e7_rmp90fix.mat' (90 $\mathrm{R_J}$ subsolar magnetopause). You should also have several other .mat files (containing Saturn models), and Matlab / Octave code modules called 'plot_mdisc.m', 'plot_mdisc_2d.m' and 'modelcut.m'.

Start Octave on your computer and go to your working directory which contains the files.

Follow the various model visualisation exercises below, and keep a notebook or log of your results and answers to the questions. This could take the form of a Word or Latex document with appropriate Figures and illustrations. You can 'polish' the document further after the laboratory session. An example logbook will be posted on the web in due course, with which you can compare yours.

2D and 1D Visualisation

The model output files contain calculation results related to the magnetic field and plasma parameters of a rotating, axially symmetric magnetosphere. This is obviously a simplification of the real systems, but nevertheless it is a useful approximation which very often shows good agreement with the observations.

In this exercise, you will start by using Octave to create two-dimensional maps of the Jovian model magnetic field and plasma pressure.

Run the code module ‘plot_mdisc’ from the Octave command window. The script should create three figure windows, which you should examine and comment on as follows:

- Figures 1 and 2: These figures shows the cold plasma pressure distribution in a single meridian plane. Logarithm of pressure is on a colour scale and the magnetic field lines are shown superposed (in magenta colour). If your Octave window has ‘zoom’ buttons, you can use them to look in more detail at the plot. Which model (expanded or compressed) shows the thicker plasma sheet? Can you explain why, in terms of what was described in the lectures? The white contour shows a surface of constant plasma beta ($\beta = 0.2$). Can you explain why it is so much more ‘flared’ in the expanded magnetodisc?
- Figure 3: This figure represents a linear ‘cut’ through the expanded model. The title of the plot indicates the start and end points of the ‘cut’ (given by coordinates ρ and z, respectively cylindrical radial distance and altitude above / below the equator. Two plots are shown - the first plot shows the relative field components (ratio of radial and vertical field component to total field strength). For this example ‘cut’, would you describe the field as dominantly radial or vertical? Why?

  The second plot in Figure 3 shows various contributions to pressure along the ‘cut’ - magnetic pressure, cold and hot plasma pressure, and total pressure (dashed line). These three sources of pressure vary with distance - for each of the three, identify whether it is ever the maximum pressure value and, if so, over what interval of distance it is the maximum.

1D Model ‘Cuts’ and 2D Meridian Plots

For the remainder of the laboratory session, you are encouraged to learn more about the magnetodisc structure by making your own ‘model cut’ plots analogous to Figure 3. To do this, you need to use the provided function ‘modelcut’. To use it for a model of your choice, type the following in the Octave command window:

`load {filename}`

`modelcut(`ρ_1`, `ρ_2`, `z_1`, `z_2`, MD)`

where `{filename}` is the name of the file containing the model you are interested in, and the ρ and z are numbers corresponding to the coordinates of the start and end points of your linear ‘cut’ through the model domain.

Start by analysing the two Jovian models. Try radial model cuts along the equator (z = 0), and vertical cuts across the equator (ρ = constant) at various radial distances. Compare what you see in the compressed and expanded systems. Try also cuts parallel to the equator at various values of z. It is also useful to compare your cuts with the full 2D representation of the model itself (‘magnetic meridian’). You can make 2D plots of the field lines, pressure and current distribution for any model file by using the following commands:

`load {filename}`

`plot_mdisc_2d(`ρ_1`, `ρ_2`, `z_1`, `z_2`, MD)` where the `load` command creates the data structure `MD`, which is then passed as an argument to the plotting function `plot_mdisc_2d`.

As time allows, repeat these exercises using the Saturn magnetodisc models - these have ‘sat’ in the file name. Their file name also indicates the value of plasma index K_h and the magnetopause radius r_{mp}. Again, compare compressed and expanded systems, and systems with different plasma content. Try and also compare the results for the different planets. Tabulate your results in a meaningful way for your notebook.

Particle tracing in planetary magnetic fields

This laboratory exercise is using the magnetodisc models you have downloaded for the Model visualisation Exercise and extend the use of these files to calculate trajectories of trapped particles and estimate both bounce and drift periods as well latitude of mirror point.

The main functions for tracing are `dipbtracer` for the dipole Boris algorithm particle tracer, and `mdbtracer` for the magnetodisc Boris algorithm particle tracer. You can get information on how to run these commands from within Octave or Matlab by running the following commands

```
help dipbtracer
```

with following output

```
function dipbtracer(planet,partype,Ep,Ri,ai,timespec,savefile,pauseOn)

Dipole Boris particle tracer

planet    : planet’s name (’earth’, ’jupiter’, ’saturn’).
partype   : particle type ’p’ -> proton, ’e’ -> electron.
Ep        : particle energy in MeV.
Ri        : initial particle equatorial position (in planet’s radius Rp).
ai        : initial particle pitch angle (0..180 degrees).
timespec  : defined as tmax = sum(timespec.*[1,tc,tb,td])
            where 1 is in units of seconds, tc in units of gyroperiod at Ri,
            tb in units of bounce period and td in units of drift period.
savefile  : filename to save the simulation data,
            if not given, do not save simulation data.
pauseOn   : flag to pause on/ pause off.
            if not given, pause on.
npertc    : optional number of Boris iterations per gyroperiod (default is 5)

For instance

  planet = ’earth’;                 % Earth
  partype = ’p’;                    % proton
  Ep = 10;                          % energy 10 MeV
  Ri = 4;                           % initial equatorial distance 4*Rp
  ai = 30;                          % initial pitch angle 30 degrees
  timespec = [0,0,2,0];             % run for 2 dipole bounce periods
```

```
   savefile = 'my_earth_mdisc_sim';  % name for result mat-file

   dipbtracer(planet,partype,Ep,Ri,ai,timespec,savefile);

 runs the Boris algorithm in Earth's dipole for a proton of 10 MeV at
 an initial position 4 Rp and pitch angle 30 deg, for 2 tb, i.e.
 two bounce periods, and save the simulation data in file
 'my_earth_dip_sim.mat'.

 Similarly for Jupiter and Saturn dipole approximation and for a
 100 MeV proton at initial position 4 Rp and pitch angle 30 deg

   partype = 'p';                    % proton
   Ep = 100;                         % energy 100 MeV
   Ri = 4;                           % initial equatorial distance 4*Rp
   ai = 30;                          % initial pitch angle 30 degrees
   timespec = [0,0,2,0];             % run for 1 dipole bounce period

   diptracer('jupiter',partype,Ep,Ri,ai,timespec);
   diptracer('saturn',partype,Ep,Ri,ai,timespec);

 but don't save the simulation data.
```

and a similar help is provided for `mdbtracer` with

```
help mdbtracer
```

One example to use each command is provided (`dipoleExample` for `dipbtracer`, and `mdiscExample` for `mdbtracer`.

Try and run both examples for the dipole and magnetodisc configurations, to get familiar with the diagnostic and output of each run. Then make a copy of each example and start to change the parameters.

You could start to investigate the time specification of the run time (for instance experiment with the dipole for Earth with different number of bounce periods, then set the number of bounce periods to zero and try to set different values (0.1, 0.3 and 0.5) to the drift period parameter in the `timespec`.

You should repeat this exercise using the Jupiter magnetodisc (or any other magnetodisc file you might want to try).

Then try to change the energy `Ep` of the proton, decreasing the energy of the proton first, then increase it. Make notes of your observations and the various bounce and drift periods and check whether the guiding centre approximations described in the class are in good agreement with the particle tracing. You could also investigate the effect of increasing the particle energy on the gyro radius by looking at the latitude plots. You might note that when the energy is increased, the trajectory does not look as regular or even trapped. You could discuss the reason for this (think about the approximation of moving on a field line, is this fulfilled for very energetic particles?)

You can in a same way further investigate the effects of changing the parameters for the initial position `Ri` of the particle on tha e equator and its initial pitch angle `ai`.

If you have time you could also investigate the accuracy of the Boris algorithm by changing the default parameter value `npertc`.

Useful Octave Commands

It is easy to clutter your computer with many Octave figures and variables when doing this kind of exercise. A useful command to know is 'close all' which closes all figures in the Octave session. 'clear all' clears all the variables from the workspace. If you wish to make illustration files for your notebook from the Octave plots, the 'print' command is useful. Type 'help print' in Octave for more information.

Appendix

For future work with Matlab instead of Octave, you can download .mat files containing magnetodisc model outputs from the VESPA web interface at `http://vespa.obspm.fr/planetary/data/epn/query/all/`. This is a model library which we have developed as part of the 'Europlanet' project.

1. Use a web browser to go to the above URL. Enter 'Jupiter' in the field labelled 'Target Name'. Then click on the 'Submit' button at the bottom of the web page.
2. You should then see a 'results' page that lists all the VESPA online resources related to Jupiter. It is worth your while when you have more time to browse through some of these databases. For this exercise, however, you need to find the entry labelled 'MDISC - UCL Magnetodisc Model...'. It is about the seventh entry from the top of the list.
3. Click on the 'Display Results' icon in the 'MDISC' entry. It is the first symbol that looks like a page with an itemised list.
4. You should now see a 'Results in Service MDISC' page with a list of files. Download the files with the suffix '.mat' in their names by using the column 'Access URL'. Take also a moment to hover your cursor over the access URL, and you should see a colour thumbnail image representing the magnetic field strength contained in the magnetodisc model (the meaning of this thumbnail should become clearer as you proceed through the rest of this exercise).