Project Problem 5

Garrett King and Katie Schram

April 2018

Weekly Progress Update

In this week's project work, the two main goal were to improve our calculations and to produce results with the published code, *pyro*. The shared objective for the week was to clean up the code using the finite difference methods and the Euler step. Some of the parts were rewritten to make them less messy and easier to find the errors. After rewriting a good portion of the code to make it more concise, this method still ran into the problem of jumping off to infinity whenever it was run. Therefore, we determined that, since the problem still occurred with cleaner versions of all of the steps, that the error was not in how it was written, but in the numeric of the code. Using smaller time steps did seem to prevent the jump to infinity, which corroborates the fact that it is a stiff equations with errors compounding quickly. Garrett's individual task for the week was to look in the Fourth Order Runge-Katta Method and the scipy integrator to see if they could be implemented in the code and solve the problem. After reading about what these methods do, the next part of the problem was deciding how it fits into the scheme already written up. Trying to figure out how one of these methods fit in is still something being worked on and will be included in the version *rk4mhd.py* in the Dropbox once it is complete. Katie's individual action item for the week has been looking into implementing *pyro* with the initial conditions we have been trying to simulate with our own code. The graphics from that section were produced during this step using the shear force initial conditions that come loaded with the code. As a learning code, the author intended students to write their own modules for what they want to solve with it. Therefore, it could be in the best interest of the projectto generate a module that begins with the TG initial conditions we have been trying to simulate on our own that uses the solver already developed. Another route, instead of continuing to develop *rk4mhd* could be to use a tool that is out there and interface it withour own module. We already can generate the initial vortex and B field required, so it would just be a matter of making it work with this code. In summary, we confirmed that it was the numerics of the code that have been causing the problem, as was suggested, by testing the impact of smaller step sizes. To avoid this, we are looking it the Fourth Order Runge-Katta Method; however, figuring out how it will work in this equations of MHD has been puzzling and is still being coded up. While those details are being worked on, there is a published code that can create visuals and demonstrate the end goal our code has. The ability to write out own modules for this published solver could be the fix that we need to finish our simulation in a timely fashion.

Graphics and Captions

Currently, the plan for the poster is to include the following sections: Introduction, Method(s), Results, Conclusion, and References.

In the Introduction section, where we discuss the usage of this code and what it is able to do, the following figure and caption will be inlcuded:

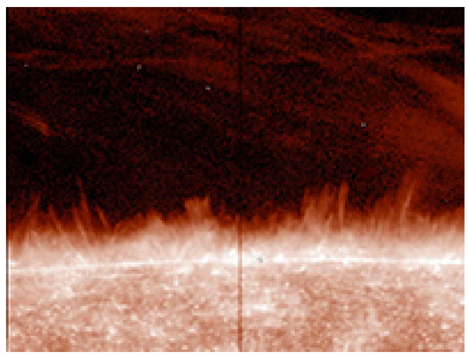

Figure 1: An image from the Sun's Corona obtained from Reference [*]. MHD simulations have many important applications to astrophysics, such as modeling jets from the Sun's Corona or modelling its magnetic field. For large scale simulations like this, parallel supercomputing is necessary to do the calculations. While more complex, it follows from the same principles that govern the magnetic fields from decaying dynamos.

[*] T. Pereira and J.M. Sykora, Uncovering the Sun's Swirling Jets, NASA@SC17, `https://www.nas.nasa.gov/SC17/demos/demo17.html`

This photo will be used to demonstrate an application of what MHD is useful for beyond just the dynamos of our simple test problem.

In the method section, the next photos will be included. These two photos demonstrate the intial conditions that we began with, which is important to the method. In that section, we can discuss how these fields fit into the requirements for the plasma and the B field given by the MHD equations.

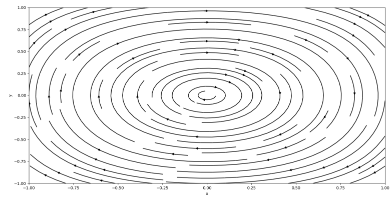

Figure 2: The Taylor-Green initial conditions for our simulation. The vector plot shows how the plasma starts out moving in a vortex. In time, the kinetic energy of the plasma will turn into magnetic energy as the vortex decays. The heat map shows the initial conditions for the magnetic field in this scenario, as well. The magnetic field will change with time and grow as the dynamo decays.

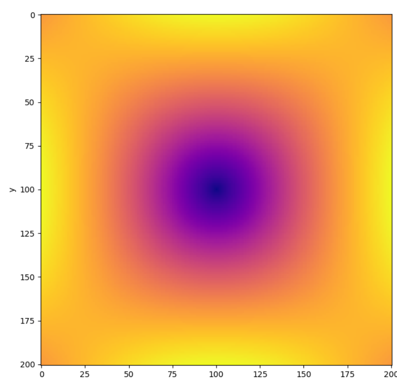

Figure 3: The heat map shows the initial conditions for the magnetic field in this scenario, as well. The magnetic field will change with time and grow as the dynamo decays.

The following image could either be included in the section for Results or Methods. It was the first result that we obtained and demonstrated where our first method fell apart. Therefore, it has elements of both.

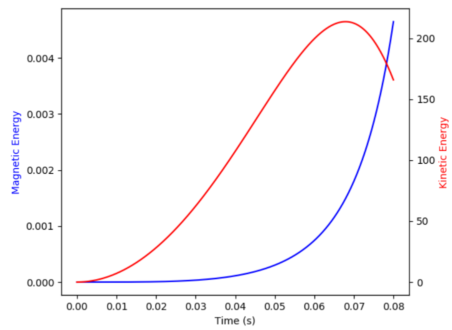

Figure 4: Results of our code when we used the finite difference method to calculate the derivatives of the velocity and the B field and the Euer step method to find how the two fields evovled in time. This plot shows how the magnetic energy grew in time as the dynamo decayed; however, the numerics of the code require very small step sizes to prevent error from compounding rapidly. For this reason, we had to start looking into other methods.

The plots obtaind using *pyro* will be featured as results, as it was not our method used to generate it.

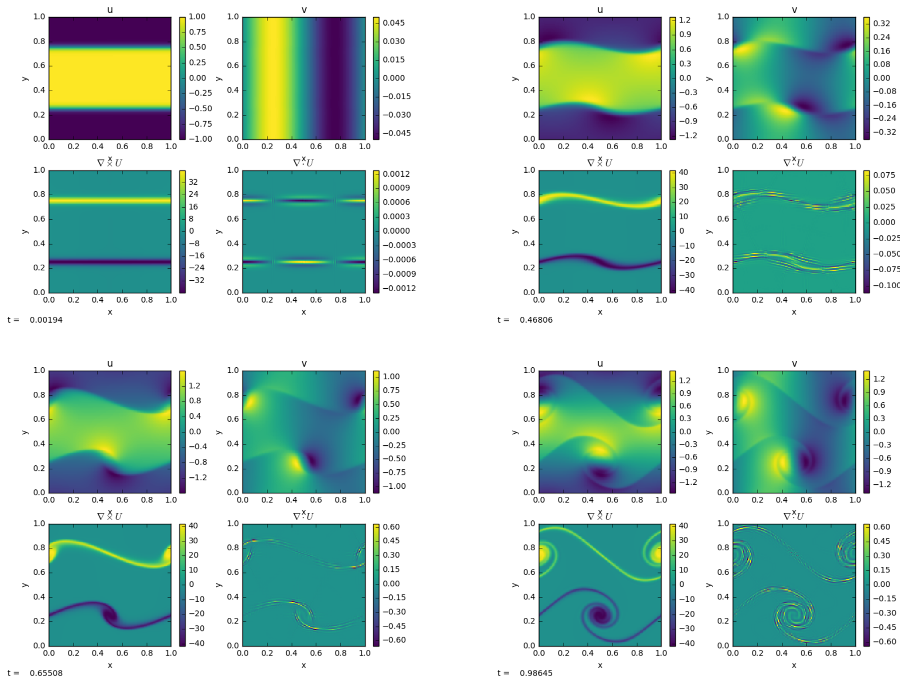

Figure 5: Time evolution of shear force initial condition on a plasma. This is one of the modules that pyro comes preloaded with and is a demonstration of the behavior of a plasma with time in an MHD calculation. U denotes the total velocity field and u and v denote the x and y components of the velocity, respectively.

The size of all of the graphics can be scaled up to match whatever size poster we are using, so we can adjust them as necessary to make the fonts readable.