Lab Notebook

Date & Time: Jan. 14, 2013
Location: Campus
Computing context: MachoMac

Continuing from last time:

Compared the following values b/w Peter and Helena for the dTime > 0 case, to see what's causing them to differ b/w the codes.

Lots of debugging followed, which won't be fully recorded here. But, the gist of it is shown in Figure 1.

Figure 1:

A comparison of Peter and Helena runs for dTime=1e8 seconds, and with the following Helena values set to their Peter counterparts: G2J, G4J, row4 of D and E matrices.

These substitutions mimic the effect of adding the full dconv_nab calculations w/in Helena. Right now, Helena calculates nabla simply as max(radnab,adnab), but Peter's code does something a bit more sophisticated.

I've figured out that the calculations involving temperature (so, G4J, and the 4th row of the CDE matrices) depend slightly on radius in Peter's code.
That's because **there's a slight radius dependence in his dcond_nab calculations, which my code's simpler gradient calculations neglect.**

Because adding the dconv_nab calculations into my code looks like it will be a little tricky, and full of opportunities to make stupid careless errors, I first wanted to check that it was really causing the differences I've been observing b/w the codes' dX profiles.

I'm not sure why the G2J values differ b/w the codes, or why that difference matters. For comparison, Figure 2 shows what happens when I do everything exactly as in Figure 1, EXCEPT for setting Helena's G2J values equal to Peter's.

The most noticeable difference is how the dL profiles behave. In Figure 1, they don't agree, but at least their signs are the same. In Figure 2, that's no longer true. So, **figuring out this G2J business isn't optional**.

Figure 2:

Exactly as Figure 1, only without setting Helena's G2J values equal to their Peter counterparts.

Before jumping into adding the dconv_nab calculations to Helena, I'd like to track down what's going on w/ the G2J calculations, and why they differ b/w the two codes. Because those differences end up having significant effects.

Lab Notebook

Played around with the terms in the G2J equation in python for a while, trying to see how/where they differ b/w the codes. **It looks like the dM, rho, etc. values vary SLIGHTLY b/w the codes, which may have something to do w/ the accuracy of the calculations, or of the variable read-in, in one of the codes.**

Start here next time, continuing with this step.

Today's Work:

A note on purchase-order stuff:

While looking into what external hard drive I should get, I started to wonder how cross-platform backups would work. Also, if there's an earthquake and everything in the office gets destroyed, then having the external hard drive would be pointless.

On the other hand, backing everything up online could take a really long time.

While googling around about all these questions, I came across a service named BackBlaze. It backs up the contents of your computer (including, if you wish/have one, an external hard drive). It's ~$5/month, though you get discounts if you buy a year's worth of service all at once. You can either download your backup files from their servers, or have them mail you a physical drive w/ your files on it in the event of a hardware failure. Extra charges apply if they mail you a physical item.

They have a free trial version available, so I've downloaded and installed it, just to try it out.

And now, for the G2J debugging.

1. Have both Peter and Helena print out the radius/dM/rho values that they're using for each G2J calculation.
2. Parse results into text files.
3. Plot and compare the results in python.

Turns out that all the G2J quantities are exactly the same b/w the two codes, EXCEPT for the densities. (See Figure 3).

Lab Notebook

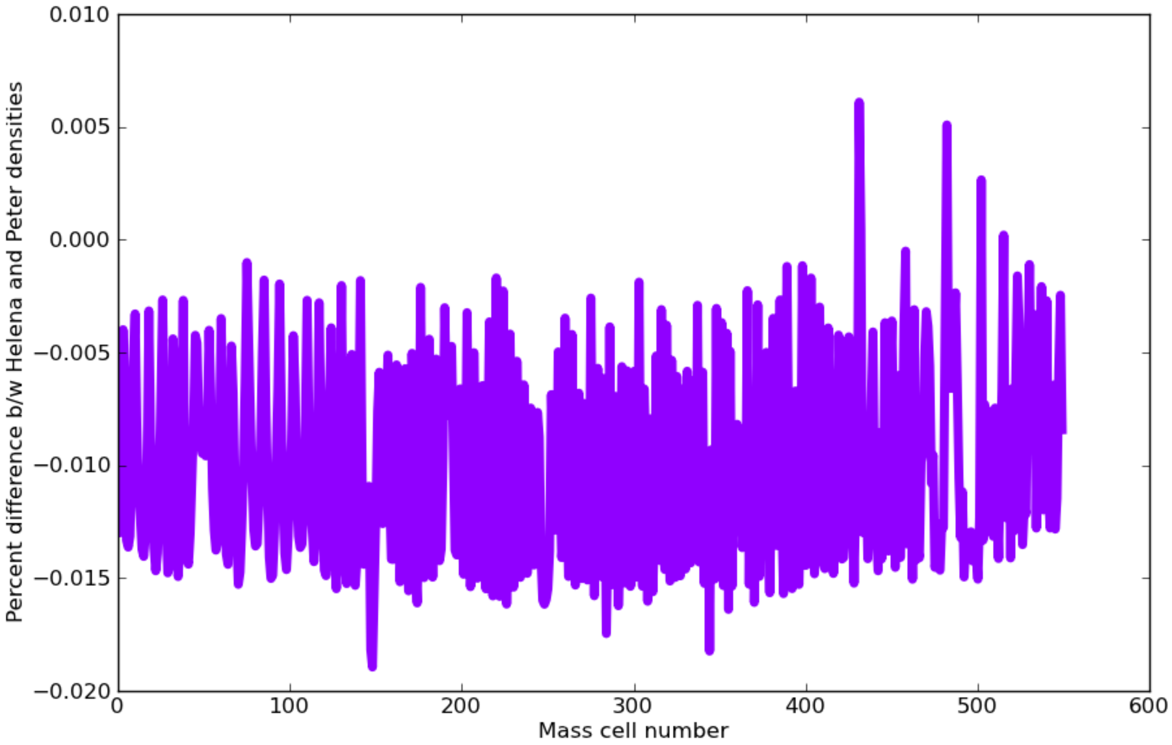

Figure 3:

Plot of % difference b/w the Peter and Helena densities for the dTime = 1e8 seconds run.

These difference don't seem like much, but they are, in fact, causing the G2J values to differ significantly b/w the two codes.

Figure 4 shows that the Helena densities are always larger than the Peter ones:

Lab Notebook

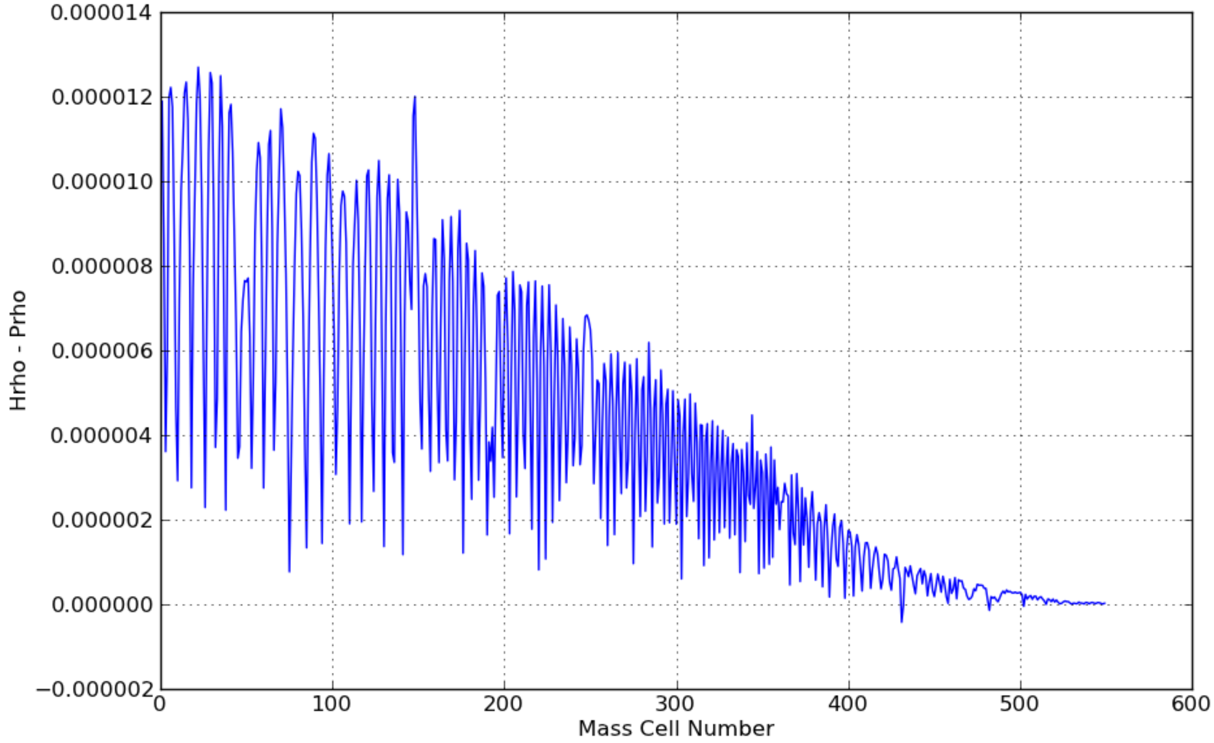

Figure 4:

Helena densities - Peter densities. Notice that the curve is almost always above zero. This means that the Helena densities are almost always slightly larger than their Peter counterparts.

A couple of questions:

1. What's causing the difference in the densities?
    1. Messed up interpolation in Helena?
    2. Maybe something to do w/ Peter's code doing that thing where it decomposes the Pressure values into Pgas and Prad? (Though I thought I'd gotten around that by setting Crad = 0 in the table-value-calculating code...?)
2. Why wasn't this a problem in the dTime = 0 case?

Start tomorrow by looking at #2, and then at #1.