Title: March 14, 2012 Research/Programming Notes & Progress
Date: March 14, 2012 4:17 PM
Category: Work
Tags: python, Bodenheimer code, finding initial conditions, from home, research

March 14, 2012 4:25 PM

Location: at home
Computing context: Macho-Mac2

From last time:

☐ *Try looking at the iterations as this setup attempts to move towards convergence...*

- *It'd be great if I could write a python script that did the model #/ iteration #/ corrections #/ evolution/ dTthresh plotting stuff all from the same script...*
- Still remains to be done... Pick up with this tomorrow, maybe.

Note to self:

- ~~Upload the working python script to the ucolick server, so that you have it backed up and available to the public (and to your future self) for teaching/ use.~~
  - Done. Available now at:
  - www.ucolick.org/~ruhlen/public_html/LabNotebook/MyFirstPythonPlottingScript.py

To do today:

- **Coding housekeeping stuff:**
  - Modify the python plotting script to take a directory name as an input argument
    - And possibly also a basename for 'the type of file we should be looking to plot'
  - ~~—~~Go into thecode.f and fix whatever's causing it to print out a model at *every* iteration, instead of every Nrit iterations.
    - ~~Is the Ntest iteration thing actually not working?~~
      - ~~Running a test simulation just to check that...~~
      - ~~It's actually working fine. I'd just forgotten that I set NTES = 1 for the pmsstar03.start simulation.~~
- **Actual science stuff w/ the code:**

- Try to answer “why aren’t the under 0.5M$_{\odot}$ (w/ no fusion) simulations converging, regardless of their timestep sizes?”
    - Plot the 0.5Msun (w/ no fusion) run results
        - Compare them to the 0.3Msun (no fusion) results.
        - Try to figure out why the 0.3M$_{\odot}$ is going off the rails, but the 0.5M$_{\odot}$ isn’t.
    - Run thecode.f with 0.45M$_{\odot}$ (no fusion)
        - plot the results
        - again, try to spot why it might be going off the rails
- See if using the ‘mass chain-down’ technique with the 0.5M$_{\odot}$ (no fusion) converged model as a starting point can produce converged models for lower mass (no fusion) balls of gas.
    - Implement a mass chain-down proceedure in thecode.f
        - Add a ‘mass chaindown?’ flag to the .start file, and modify thecode.f to be able to read it in
        - If the ‘mass chaindown’ = true,
            - read in a converged model
            - evolve it forward in time by 10(?) dTthresh steps
            - then decrease the mass of the system by some factor
                - (By what factor? How much or how little can you successfully decrease the mass at any given chain-down step? Need to think about this more once I get to this point...)