Geodynamics tutorial 1: Convection modeling with citcom

Peter van Keken, Louise Kellogg, Bruce Buffett, Michael Manga

CIDER Summer School 2008

Introduction

This tutorial will allow you to investigate some aspects of mantle convection modeling using the finite element code Citcom. We will first explore the installation and basic use of the code and associated applications before investigating the relationships between the Nusselt and Rayleigh numbers for isoviscous and (for the brave-hearted) temperature-dependent viscosity.

We recommend that you work together in teams of two, preferably while making sure there is at least one person who is somewhat comfortable with unix.

Preliminaries

Citcom
Citcom is the name associated with a number of related finite element codes that found their origin at Caltech in the mid-90s. Louis Moresi is the principle author of the version we will be using. You are welcome to keep a version of this code but -as Louis requests in the source code- please be respectful of the time that went into creating this code. The code solves the incompressible Boussinesq equations for mantle convection

$$\nabla \cdot \mathbf{u} = 0$$

$$\nabla P = \nabla \cdot \sigma - RaT\hat{\mathbf{z}}$$

$$\frac{\partial T}{\partial t} + \mathbf{u} \cdot \nabla T = \nabla^2 T + Q$$

where Q is a non-dimensional number representing internal heating.

Download and installation
The code can be downloaded from `www.geo.lsa.umich.edu/~keken/CIDER.tgz`. This file is a compressed tar file that on a Mac (or linux PC) should be automatically decompressed and the files should be installed in the Desktop section of your home directory.

To run Citcom and analyze the models you'll need to use the Unix shell. Find the shell (pictured by a window with a cursor in it) on your toolbar or in Finder->Applications. Start a shell by clicking on it. Window will appear with a prompt (probably of the form:

```
kitp134:~ yournamehere$
```

which is the command line interface. We'll shorten this full prompt to "$" in the descriptions below. You can list files with 'ls', change directory with 'cd Desktop'. Locate the CIDER directory and change into it. You should have a list of directories (such as Citcom.build.file, Perl_scripts, CIDER06). Change directories to CIDER06/examples. Here you should see a few example directories, including 00_test, a doc directory (which contains this document) and a number of executables (citcom, extract_Nu, extract_vrms). The Macs at KITP should be able to run the citcom executable as is. Check this by changing the director to 00_test and running

```
$ ../citcom  input-test
```

If you get messages like "2: divergence" you should be in business. This program should terminate in less than a minute with a message about a ppm file being written and some statements about cpu timing.

If there are error messages you may need to recompile citcom. To recompile:

```
$cd ˜/Desktop/CIDER/Shell_scripts
$build-citcom
$cp cII* CIDER06/citcom
```

and rerun the code in the test directory. If this fails ask Bruce or Louise for help.

Basic model setup

In all cases we will assume a 2D Cartesian geometry. The initial model has aspect ratio 1 and a fluid of uniform viscosity. The box is heated from below (fixed temperature 1) and cooled from above (fixed temperature 0). The side boundaries have reflecting (symmetry) boundary conditions. The boundaries of the bottom are stress-free (i.e., not rigid). The equations are solved by a multigrid method, which requires the discretization into a fairly regular grid. In these examples we'll limit ourselves to a uniform discretization, with the initial model at 33x33 nodal points.

The initial condition for temperature is a conductive solution with a small (10%) harmonic perturbation that will cause to initiate convection when the Rayleigh number is large enough. You can also specify the name of a file that contains the initial condition (which we will use to restart convection runs).

In the first few examples we will try to get a steady-state, which in this code is obtained by letting time integration continue long enough. This works fine at moderate Ra, but becomes increasingly harder when the convection 'wants' to be time-dependent.

First full convection run

Find the example directory 0_iso_1d4. Check out the input file 'input-2d_Ra1e4'. You can do this with any editor through MacOS, or with vi or emacs from the shell.

The top 10 lines set some variables which are defined as:

`rayleigh` Thermal Rayleigh number
`maxstep` Maximum number of time steps
`finetunedt` The multiplication factor for the self-selecting timestep.
`datafile` Common root of all output filenames
`storage_spacing` Store output files every ...
`write_spacing` Store output files every ...
`TDEPV` Indication whether temperature dependent viscosity is used
`rheology_T_type` Choice of viscosity function
`viscT1, viscN0` Parameters in viscosity function

`dimenx` Non-dimensional grid size of the mesh in x-direction
`levels` Number of multigrid levels.
`perturbk` Wave number of the initial perturbation
`previous_temperature_file` Name of file with initial condition.
The last variable is commented out (with a hash mark at the start of the line. Remove this hash mark when you want to do a restart from an existing file. The rest of the file is better left unchanged....

Run the citcom code with this input file

```
$ ../citcom input-2d_Ra1e4
```

If you don't like the output rolling on the screen or would like to check things in this shell while the code runs you can redirect the output and run it in the background with

```
$ nohup ../citcom input-2d_Ra1e4 &
```

Use 'jobs' to check whether it is still running, 'fg' to bring it to the foreground, 'ctrl-C' to kill it.

The code will produce files ending in .horiz_ave, .node_data, and .ppm. The first contains, well, horizontal averages; the second contains the temperature in the nodes, and last one is a graphics file that can be converted to gif by a program called GraphicConvector. Some information about the run is printed in the logs ending in .log (which contains basic info about the convergence of the multi grid cycle, computational time used) and .timelogs (which prints model time, Nu, Vrms etc). We will extract time series Nu(t) and Vrms(t) from the latter one using the utilities ../extract_Nu and ../extract_Vrms; the time series can be plotted with xmgrace.

Graphics output: temperature
Citcom produces plots of the temperature field and surface velocity in PPM format. Open GraphicConverter to view these files (Finder->Applications->GraphicConverter). If all goes well you can navigate to the run directory and find images of the ppms. Click on the thumbnails for a larger view. The interface is not very 'smart' (i.e., you'll need to move out of and back into a directory to refresh the file list) but it works. In the images for the first run you should see a spin-up with diffusion slowly becoming more important in the middle of the box.

Graphics output: time series
If you have't done so already, run

```
$ ../extract_Nu ; ../extract_Vrms
```

which generates the files Nu.dat and Vrms.dat. You can inspect the final values by

```
$ tail Nu.dat Vrms.dat
```

the last values should be around 4.8 and 42. Check with Louise or Bruce if this isn't the case (at all). To plot these files you can import them in your favorite graph generator (e.g., matlab, gmt, excel, grace, gnuplot). A simple way to plot the files is with gnuplot (available as ˜vonkeken/gnuplot/bin/gnuplot) You'll need to run gnuplot in a X-windows terminal. Find the X-windows terminal in the toolbar or Finder->Applications->Utilities->X11. Run from this window

```
$ ˜vonkeken/gnuplot/bin/gnuplot
```

and type

```
plot "Nu.dat"
```

at the prompt (the quotation marks are essential).
You should see the spin-up and stabilization of the convection reflected in the time-series. Inspect the last 20 or so entries in Nu.dat; you may notice that the model hasn't gotten to steady-state yet. Fix this.

Some suggestions for exploring mantle convection.

Check out the input file in 1_iso_1d4_aspect2 to see how you set up the isoviscous problem in a box of aspect ratio 2. How do Nu and Vrms change? What is a good estimate for the effective aspect ratio of convection in the Earth?

Explore the variation of Nu and Vrms with Ra for a reasonable range (around 10^4). Can you estimate the critical Rayleigh number? Verify the accuracy of your results with the 'best' answers in the thermal benchmark (Blankenbach et al., GJI, 1989):
Ra= 10^4; Nu=4.8844, Vrms=42.865
Ra= 10^5; Nu=10.5412 , Vrms=193.215
Ra=10^6; Nu=21.972 , Vrms=833.990
What happens at the higher Rayleigh numbers in terms of accuracy of the solution? Are there other diagnostics you could use to see there's numerical trouble?

It may be quite hard to get steady state results. Pick Louise's brain how to improve on that.

Refine the grids for those cases that you don't feel comfortable with by increasing the input parameter levels (note: increase by 1 causes doubling of number of grid points in each direction and an increase in computational time of at least 4 times).

Explore temperature-dependence of viscosity as provided in 2_etaT_1d4. Note that there are two input files: one to set up an initial condition (input-iso_initial), and one to use the final result as input for the temperature-dependent case. To run this case:

```
$ ../citcom  input-iso_initial
$ cp ra1e4-2d.00500.node_data init.node_data
$ ../citcom  input-etaT_Ra1e4
```

The viscosity in this case is prescribed by

$$\eta(T) = \exp(-bT)$$

with the parameter b represented by viscT1 in the input file. The input file in this directory is set up for the Blankenbach benchmark 2a (Ra=10^4, $b = ln1000$, which has 'best' values of Nu=10.066 and Vrms=480.43). Why are Nu and Vrms in this case higher than that for the isoviscous case with teh same Ra? Can you get accurate results at 33x33?