Gung-Ho Dynamics:

A primal-dual finite element based method for solution of the rotating shallow water equations on the sphere.

Brief notes for users.

John Thuburn
17 December, 2012

1 Introduction

A shallow water code based on certain finite element operators on an arbitrary polygonal grid and its dual, with a semi-implicit forward in time integration scheme, has been developed for Gung-Ho. The scheme overcomes certain limitations of an earlier mimetic finite volume scheme. The code has been tested on two families of grids: hexagonal-icosahedral Voronoi grids and modified equiangular cubed sphere grids. Details of the scheme are presented in a separate report (`pdfem_corrected.pdf`, 7 Dec, 2012). These brief notes are intended to allow others to get started using the code.

2 Generating grids

Files: `gengrid_hex.f`, `dodecahedron.xref`, `gengrid_cube.f90`

The grid generation codes are used in exactly the same way as for the mimetic finite volume model. See the document `mimetic_usernotes.pdf` dated 16 March 2012.

3 Pre-processing to build finite element operators

Files: `builop_fem.f90`, `gridmap_hex_xxxxxxxxxx.dat`, `gridmap_cube_xxxxxxxxxx.dat`

Many of the finite element operators used by the scheme are precomputed after the grid generation stage before running the model itself. In contrast to the mimetic finite volume model, a single pre-processor code is used for all grid types. Currently the code is only able to provide the multigrid restriction operator for the hexagonal-icosahedral grid and the cubed sphere grid. The remaining operators can be built for arbitrary polygonal grids.

Once the desired grid has been generated, it is simple to run the pre-processor code: compile and run the fortran file `buildop_fem.f90`, and answer the questions asking which type of grid (hexagonal or cubed sphere) and how many faces. The answers to these questions determine the name of the gridmap file in which the pre-processor expects to find the grid information.

4 Running the model

Files: `femswe.f90`, `swenml.in`, `gridopermap_hex_xxxxxxxxxx.dat`, `gridopermap_cube_xxxxxxxxxx.dat`

Most model options are set via the namelist file `swenml.in`.

`ygridfile`	Name of the gridopermap file to use.
`ygridcoords`	Name of file in which to dump grid coordinates for subsequently generating reference solutions.
`degree`	Degree of polynomial fit to use for advection. Should work for 0, 1, 2, 3, or 4 but only tested for 2 and 4.
`dt`	Time step in seconds.
`niter`	Number of iterations of nonlinear solver. Only tested with 4.
`alpha_v`	
`alpha_pg`	Off-centring parameters for velocity and pressure gradient terms. Testing to date has used 0.5d0.
`nstop`	Length of integration in steps.
`noutput`	Number of steps between output dumps.

Other namelist switches are not used.

Constants such as Earth's radius, acceleration due to gravity, and Earth's rotation rate can be set by changing the values in `Module constants`.

The orography can be modified by editing `SUBROUTINE setorog`.

The initial condition can be set by editing `SUBROUTINE setini`. Several standard test case initial conditions are already coded and can be selected by setting the variable `ic`.

The model can output fields on primal grid faces (dual grid vertices) and on dual grid faces (primal grid vertices). Edit `SUBROUTINE output` to choose the desired fields. You may need to add some code to compute the desired fields from the prognostic variables `phi2` (primal cell area integral of ϕ) and `u1` (primal edge integral of normal velocity) if this is not already done.

Once all the desired options have been selected, compile and run `femswe.f90`. The code reads switches from the namelist file `swenml.in` and grid and operator information from the file defined by the switch `ygridfile`.

5 Plotting output

Files: `join.m`, `jtaxes.m`, `jtcontour.m`, `jtplotgrid.m`, `jtrotplot.m`, `dump1_yyyyyyyy.m`, `dump2_yyyyyyyy.m`, `fort.43`

Plotting of output is exactly the same as for the mimetic finite volume model. See the document `mimetic_usernotes.pdf`.

6 Generating reference solutions

The code contains a subroutine that will dump the grid cocordinates in a simple format suitable for subsequently generating reference solutions and interpolating them to the model's grid using the ENDGame shallow water code. First set the namelist switch `ygridcoords` to a suitable file name, e.g. `gridcoords_hex_xxxxxxxxxx.dat` or `gridcoords_cube_xxxxxxxxxx.dat`, where `xxxxxxxxxx` is an integer giving the number of faces of the grid. Next modify the main program routine to comment out the CALL to subroutine `integrate` and to uncomment the CALL to `dumpgrid`. Then compile and run the `femswe.f90`.

Once the grid coordinates have been dumped, follow the instructions given in the notes `EG_refsoln_doc.pdf` of (7 Dec. 2012) for generating the ENDGame reference solutions. Having generated the reference solution data, edit `femswe.f90` to uncomment the CALL to subroutine `diffref` from subroutine `integrate` and modify subroutine `diffref` as necessary in order to compute and output differences from the reference solution.