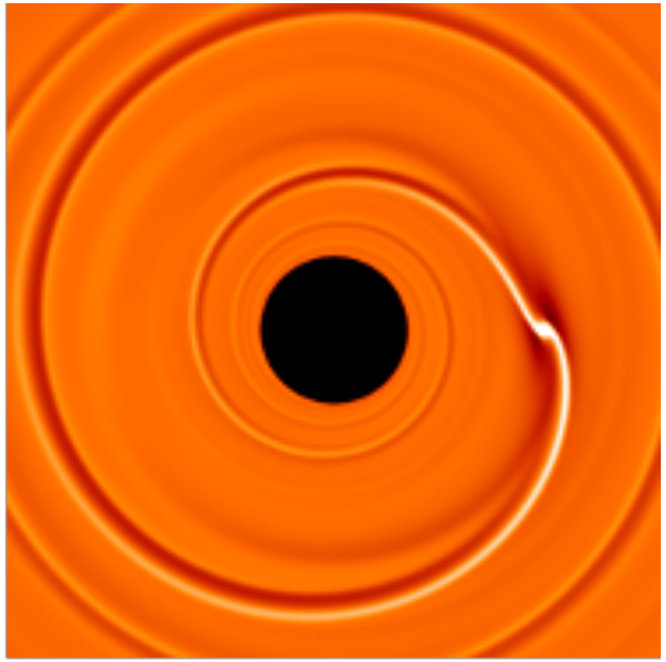
FARGO3D

Hydrodynamics and magnetohydrodynamics parallel code

David Velasco-Romero

Introduction



FARGO3D is a hydrodynamics and magnetohydrodynamics parallel code.

- Eulerian mesh code.
- Multidimensional (1D, 2D & 3D).
- Several geometries (Cartesian, cylindrical and spherical).
- Non-inertial reference frames (including shearing box for Cartesian setups).
- Adiabatic or Isothermal Equation of State (EOS). It is easy to implement another EOS.
- Designed mainly for disks, but works well for general problems.
- Solves the equations of hydrodynamics (continuity, Navier-Stokes, and energy) and magnetohydrodynamics (MHD).
- Includes ideal MHD (Method Of Characteristics & Constrained Transport).
- Includes non-ideal MHD terms (Ohmic and Ambipolar Diffusion and Hall effect).
- Includes the full viscous stress tensor in the three geometries.
- Simple N-body integrator, for embedded planets.
- FARGO algorithm implemented in Cartesian, cylindrical and spherical coordinates.
- The FARGO or “orbital advection” scheme is also implemented for MHD.
- Possible run-time visualization.
- Multi-platform:
  - Sequential Mode, one process on a CPU.
  - Parallel Mode, for clusters of CPU (distributed memory, with MPI).
  - One GPU (CUDA without MPI).
  - Parallel GPU Mode, for clusters of GPUs (mixed MPI-CUDA version).

Download

https://bitbucket.org/fargo3d/public.git

HTTPS:

```
git clone https://bitbucket.org/fargo3d/public.git
```

SSH:

```
git clone git@bitbucket.org:fargo3d/public.git
```

Compilation

Go to fargo3d repository:

```
$: cd USER_PATH/fargo3d
```

You should find the following:

```
:USER_PATH/fargo3d/$ ls
bin  doc  Makefile  outputs  planets README  scripts
setups  src  std  test_suite  utils
```

The code is build here (not in src/):

```
:USER_PATH/fargo3d/$ make
          FARGO3D SUMMARY:
          ===============

This built is SEQUENTIAL. Use "make para" to change that


SETUP:      'fargo'
(Use "make SETUP=[valid_setup_string]" to change set up)
(Use "make list" to see the list of setups implemented)
(Use "make info" to see the current sticky build options)
```

First run

If you have a look at the content of the main directory, you will see that after the compilation a new file has been created, called ‘fargo3d’. This file is the binary file. We can now perform the first run:

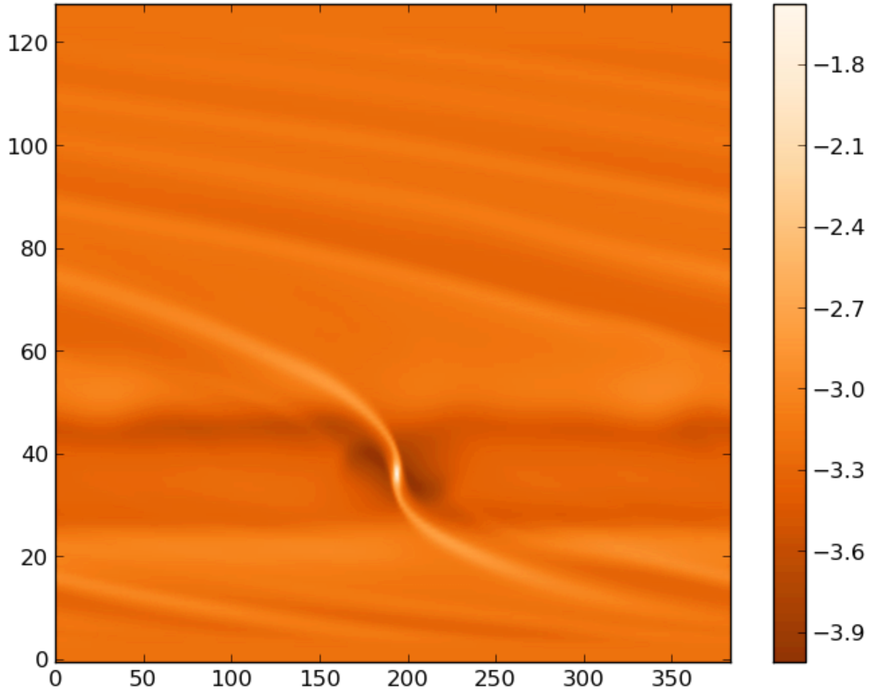
```
$: ./fargo3d setups/fargo/fargo.par
x = 1.0000000000     y = 0.0000000000         z = 0.0000000000
vx = -0.0000000000   vy = 1.0004998751        vz = 0.0000000000
Non-accreting.
Doesn't feel the disk potential
Doesn't feel the other planets potential

Found 0 communicators
OUTPUTS 0 at date t = 0.000000 OK
TotalMass = 0.0121800000
............
..............
..............
................
................
..................
................
................
```

Visualization

Python and matplotlib

Here is an example of visualizing the 10th output for the density:

```
%pylab inline
rho = fromfile(“gasdens10.dat").reshape(128,384)
imshow(log10(rho),origin='lower',cmap=cm.Oranges_r,aspect='auto')
colorbar()
```



First parallel run

In order to compile the code for parallel execution with MPI you can either do:

```
$: make PARALLEL=1
```

or use the shortcut:

```
$: make para
```

The code can then be executed with multiple processors.

```
$: mpirun -np 4 ./fargo3d setups/fargo/fargo.par
```

First GPU run

In order to compile the code for GPU runs, one can either issue:

```
$: make PARALLEL=0 GPU=1
```

or use the shortcut:

```
$: make PARALLEL=0 gpu
```

Another option is:

```
$: make mrproper
```

Which resets all sticky built options to their default values, then one can issue:

```
$: make gpu
```

The code can then be executed making use of a GPU.

First GPU run

You will see at the end of the building process the message:

```
FARGO3D SUMMARY:
          ===============

This built is SEQUENTIAL. Use "make para" to change that

This built can be launched on
a CPU with a GPU card (1 GPU only).


SETUP:       'fargo'
(Use "make SETUP=[valid_setup_string]" to change set up)
(Use "make list" to see the list of setups implemented)
(Use "make info" to see the current sticky build options)
```

Documentation

https://fargo3d.bitbucket.io/index.html