List of test cases to experiment with

Let think about what form AMRVAC solves the equation in: <http://amrvac.org/md_doc_equations.html>

Why is solved in this form?

For full information overload of everything that's in the parfiles consult the following (I will refer to this link as IOL):

<http://amrvac.org/md_doc_par.html#par_filelist>

I recommend that you don't read it only consult it when you want to see your options for certain parameter already existing in the par file for the time being.

*Top Tip: Use* ***ctrl+r*** *to reverse search term, e.g. try* ***ctrl+r*** *then type* ***set****.*

Feel free to run any of the test cases that catch your eye, but below are a list of test cases that definitely run.

**amrvac/tests/nonlinear/burgers2d**

While this runs, let’s think about the form the equation is written in (see line 41 in **modusr.t)**

**amrvac/tests/hd/Riemann\_1D**

[**http://wonka.physics.ncsu.edu/pub/VH-1/bproblems.php**](http://wonka.physics.ncsu.edu/pub/VH-1/bproblems.php)

[**https://www3.nd.edu/~dbalsara/Numerical-PDE-Course/ch5/Chp5\_RP\_Euler.pdf**](https://www3.nd.edu/~dbalsara/Numerical-PDE-Course/ch5/Chp5_RP_Euler.pdf)

This (hopefully) is a good problem to get a feel for the effect of different numerical schemes. Lets experiment with the parfile (e.g. **amrvac.par**) and don’t look in the **modusr.t** yet!. Find **iprob** in the parfile, change it **1**, **2** or **3** and let us also change the **base\_filename** so we don’t overwrite our simulation each time. Run the simulations for **iprobs** 1 to 3. View your simulation and try explain your results? Peek at the main body of code (**mod\_usr.t**) to see if you were right.

After playing with the iprobs, choose the one you deem most exciting and mess around with the numerical schemes. Look at the **methodlist** in both the parfile and IOL. Try different **flux\_scheme** or different **limiter** (try ‘minmod’ and ‘superbee’ to see both extremes). How does this affect your results?

**amrvac/tests/hd/blast\_wave\_Cartesian\_2D**

Now we know how to change resolution lets makes our simulations more interesting. Change the physical size of computational domain to 2 by 3 and the level one grid to 32 by 48. Increase the levels of AMR to 4.

Next we are going to submit this job to the queue in SHARC and take full advantage of its computational prowess. Edit the [template submit file](https://drive.google.com/open?id=14aGUeDnXy9JtSpChfVe9lLj6JuWUvlNC) I have provided, see <https://www.sheffield.ac.uk/cics/research/hpc/sharc/batch> for further information on what the flags mean. Once this is done simply type:

**qsub sub\_BW\_HD**

If you have fed the SHARC correctly then when you type **qstat** you will see you have a 2 jobID’s and one of them will be waiting in the queue (look under the state column).

**amrvac/tests/mhd/Riemann\_1.75D**

What does it mean by 1.75D? How does it compare to other Riemann simulations with no magnetic field? Can you explain what we see?

**amrvac/tests/hd/Riemann\_2D**

Experiment with the parfile (e.g. amrvac.par). How do you increase levels of AMR? How do you change the duration the simulation? What happens if I use ‘minmod’ as flux limiter as suppose to ‘superbee’?

**amrvac/tests/mhd/blast\_wave\_Cartesian\_2D**

Do the same as above. Once both simulations are run, compare them to each other. Can you explain what you see in both simulations?

Let’s start making changes to the code. Go back to the test folder and create a new directory (e.g. **mkdir my\_test**). Go to your new directory and copy the MHD version of the blast wave (**scp -r ../mhd/blast\_wave\_Cartesian\_2D .**). If we look at the **modusr.t** we will see that **B** has the same magnitude in x and y. Lets change this so we have either magnetic in x or y. The magnetic field strength is set in the **get\_B** subroutine. Then run your code and you should get something similar to this:

**amrvac/tests/mhd/lfff\_extrapolation\_3D**

We will use this test to see how visualise magnetic field lines in paraview.

**/amrvac/tests/mhd/potential\_field\_source\_surface\_3D**

There are errors with this code let's fix them!

**/amrvac/tests/mhd/solar\_flare\_reconnection\_2.5D**

See <https://arxiv.org/pdf/1710.06140.pdf> section 3.4 for more information on numerical setup.

**amrvac/tests/hd/Rayleigh\_Taylor\_2D**

This is a good test to see how gravity is implemented.

**Note: next session focus on paraview, maybe if we have time do stretch grids**

More info:

Conservative form: <https://physics.stackexchange.com/questions/70496/conservation-vs-non-conservation-forms-of-conservation-equations>

MHD blast waves: <https://www.astro.princeton.edu/~jstone/Athena/tests/blast/blast.html>

Sec 6.5 <https://www.sciencedirect.com/science/article/pii/S1384107608000754>

Burgers equations and Riemann:

Sec 3.4 [Numerical solution to Burgers equations](https://www.uni-muenster.de/imperia/md/content/physik_tp/lectures/ws2016-2017/num_methods_i/burgers.pdf)

[More info on Riemann problem](https://www.uv.es/astrorela/simulacionnumerica/node35.html)

<https://perswww.kuleuven.be/~u0016541/Talks/nonlinCM.pdf>

<https://services.math.duke.edu/education/joma/sarra/sarra3.html#rare>