

The Gerris FAQ

Stéphane Popinet
popinet@users.sf.net

April 3, 2006

Contents

1	General questions	3
1.0.1	What does “Gerris” mean?	3
1.0.2	How is “Gerris” pronounced?	3
1.0.3	Where are the printable versions of the docs?	3
1.0.4	What grid generator is Gerris using?	3
1.0.5	Can Gerris handle unstructured tetrahedral meshes?	3
1.0.6	Are there any plans to extend Gerris to RANS (Reynolds-Averaged Navier-Stokes)?	3
1.0.7	What boundary conditions are in the code now and what BCs are planned for the near future?	4
1.0.8	How is Gerris parallelised? Does it use MPI or a shared memory technique?	4
1.0.9	Does adaptive refinement work in parallel?	4
1.0.10	Does the parallel version of the code do load-balancing?	4
1.0.11	When will parallel load-balancing be available?	4
1.0.12	Can Gerris handle moving/deforming solid geometries?	4
1.0.13	Can Gerris also be used for compressible fluids?	4
1.0.14	Can Gerris solve the shallow-water (Saint-Venant) equations?	4
1.0.15	How can I assist you in your development effort?	5
2	Installation and coding	5
2.0.1	How do I install the parallel version of Gerris?	5
2.0.2	Does the Gerris configure script support a LAM MPI implementation?	5
2.0.3	My crappy MPI installation prevents Gerris from compiling, how do I turn MPI support off?	5
2.0.4	Is there a Windows version of Gerris?	5
2.0.5	Are there any plans to release a more documented version of the code?	5
2.0.6	Are there any plans to release a C++/Java/Object Oriented implementation of Gerris?	6
3	Physics and dimensioning	6
3.0.1	Where are variables like viscosity, density etc...defined?	6
3.0.2	How come Gerris generates a Von Karman vortex street for an inviscid flow around a half-cylinder? I would expect the inviscid flow to remain irrotational.	6
3.0.3	How would I create a 5×5 box?	6
3.0.4	How would I modify the file you sent me (tangaroa.gfs) for a ship that is 150 meters long and exposed to a cross-flow wind velocity of 50 meters/sec?	6
3.0.5	How would I redimensionalise U,V,W and P?	6

3.0.6	It looks like <code>t</code> and <code>dt</code> output by <code>GfsOutputTime</code> are also scaled? How would I scale <code>t</code> and <code>dt</code> to time in seconds?	7
3.0.7	How do I scale Vorticity?	7
3.0.8	The code provides support for the variable density incompressible Euler equations. Does that mean you can input the density of the fluid density (air, water, etc...)?	7
3.0.9	Although the initialised problem is symmetric, the solution becomes asymmetric as time passes, why?	7
3.0.10	How do I deal with negative values of the pressure?	8
4	Representation of solid boundaries	8
4.0.1	How do I import my geometry into Gerris?	8
4.0.2	What CAD package can I use to export STL/GTS files?	8
4.0.3	Do I need to tessellate (increase the number of triangles of) my surface before importing it into GTS?	8
4.0.4	Which part of the parameter file tells Gerris where the half-cylinder is placed? How do I alter it?	8
4.0.5	Are there any tools for converting format-X (not STL) files (generated via a CAD system) to a GTS-format file?	9
4.0.6	Gerris seem to allow only one solid body, is this correct?	9
4.0.7	How do I orient my solid surfaces properly?	9
4.0.8	It looks like all STL files need to be turned “inside out”. I don’t understand why, but <code>transform -i</code> fixed the problem.	10
4.0.9	Can solids intersect?	10
4.0.10	We have a problem inserting some GTS files generated from STL files and even inserting the standard GTS files found on the GTS samples ¹ site?	10
5	Post-processing and Visualisation	10
5.0.1	Is there a way to view the results and solid(s) at the same time (with the solid in the correct location? (Geomview question)	10
5.0.2	Is there any way to output <code>U</code> , <code>V</code> , <code>W</code> , <code>P</code> , etc... at point (<code>X</code> , <code>Y</code> , <code>Z</code>) in the flow field?	10
5.0.3	Where is the description of the format of the data section of saved simulation files?	10
5.0.4	How do I compute/display the vorticity field with OpenDX?	11
5.0.5	The <code>GfsOutputSimulation</code> and <code>GfsOutputLocation</code> files only place up to eight decimals. Is there any way to increase the number of decimal places?	11
5.0.6	I would like a time-averaged velocity profile, Would I have to specify a number of monitoring points at different heights, or is there a method to time average over a line through the solution domain?	11
5.0.7	Using <code>animate</code> , the sequence of images generated by <code>OutputPPM</code> looks weird, what’s happening?	11
5.0.8	Why create a new visualisation tool like <code>GfsView</code> ? Can’t you use existing tools like <code>Mayavi/VTK</code> , <code>OpenDX</code> etc...?	12
6	Running Gerris	12
6.0.1	Are the files <code>vorticity.gfs</code> and <code>half-cylinder.gfs</code> included in the Gerris distribution?	12
6.0.2	Your flow analysis for the RV Tangaroa is just the type of problem I would like to be able to solve quickly, etc...How long did it take to setup and run this problem? Could you send me a copy of the input file?	12
6.0.3	How do you modify the mesh to give greater detail in a specific area of flow?	12
6.0.4	Is there any way to initialise the grid so that a fine grid is generated around the surface of the solid and a coarser grid is generated in the flow field (resulting in significantly fewer cells being generated)?	13

¹<http://gts.sourceforge.net/samples.html>

6.0.5	Is it possible to turn off adaptive meshing in defined areas?	13
6.0.6	Is there a way to control the maximum size of a simulation?	13
6.0.7	My simulation file looks fine but does not work, why?	13
6.0.8	How do I make all the boundaries of a 2D box no-slip?	13
6.0.9	How do I make boundary conditions time-dependent?	13
6.0.10	How do I run Gerris in parallel?	14

1 General questions

1.0.1 What does “Gerris” mean?

Gerris is the Latin (and French) name of the water strider (or water boatman), an aquatic insect which uses surface tension to “walk” on the surface of the water. Have a look at the logo on the front page of the Gerris web site² for a graphical description.

1.0.2 How is “Gerris” pronounced?

With a soft ‘g’ like ‘genetics’ or ‘general’.

1.0.3 Where are the printable versions of the docs?

The tutorial, FAQ and examples pages all have printable PDF (tutorial.pdf³, faq.pdf⁴, examples.pdf⁵) and Postscript (tutorial.ps.gz⁶, faq.ps.gz⁷, examples.ps.gz⁸) versions, the reference manual only exists as an HTML document⁹.

1.0.4 What grid generator is Gerris using?

Gerris uses an “embedded boundary” technique. Grid generation reduces to the computation of the “shape” (surface and volume fractions) of Cartesian (cubic) cells cut by the solid boundaries. These “boolean operations” between solids are performed automatically using GTS (the GNU Triangulated Surface Library). The cells cut by the boundaries can then be refined automatically using the quad/octree structure of the discretisation. Mesh generation is entirely automatic and works for any input geometry (provided it is topologically consistent, i.e. an orientable non-self-intersecting manifold).

1.0.5 Can Gerris handle unstructured tetrahedral meshes?

Gerris uses a quadtree (octree in 3D) finite volume discretisation. It cannot handle unstructured meshes.

1.0.6 Are there any plans to extend Gerris to RANS (Reynolds-Averaged Navier-Stokes)?

The focus is on time-dependent Navier-Stokes, so RANS is not really in my mind but nothing is in the way if this is what you need.

Large Eddy Simulation (LES) models is what I am thinking about as far as turbulence modelling is concerned.

²<http://gfs.sf.net>

³<http://gfs.sf.net/tutorial/tutorial.pdf>

⁴<http://gfs.sf.net/faq/faq.pdf>

⁵<http://gfs.sf.net/examples/examples.pdf>

⁶<http://gfs.sf.net/tutorial/tutorial.ps.gz>

⁷<http://gfs.sf.net/faq/faq.ps.gz>

⁸<http://gfs.sf.net/examples/examples.ps.gz>

⁹<http://gfs.sf.net/reference/book1.html>

1.0.7 What boundary conditions are in the code now and what BCs are planned for the near future?

Slip, no-slip solid boundaries, inflow, outflow, periodic... but everything is there to implement your own boundary conditions as Gerris can not supply all the boundary conditions users could think of.

1.0.8 How is Gerris parallelised? Does it use MPI or a shared memory technique?

Gerris uses a domain decomposition approach using MPI for synchronisation at domain boundaries. For the moment it does not do dynamic balancing of domain sizes which limits its applicability to statically refined problems.

1.0.9 Does adaptive refinement work in parallel?

Yes but there may be load-balancing issues. I don't use the MPI version for the moment. For the type of studies I am interested in, it is usually much easier to do "direct parallelism" i.e. run several simulations with different parameters "in parallel".

1.0.10 Does the parallel version of the code do load-balancing?

No, the code does not have parallel load-balancing capabilities at the moment. It can do static load-balancing i.e. "optimal" partitioning of the domain so that an initial mesh is divided in roughly equal-sized subdomains while minimising the size of the communication boundaries.

I don't personally use the parallel capability of the code very much. What I usually need is several different sequential computations with a different set of parameters. This is of course the ideal case for "parallelism". I don't usually require to run "one shot" very large parallel computations.

1.0.11 When will parallel load-balancing be available?

Load-balancing is not very high on my list of priorities at the moment. I don't see fundamental obstacles to dynamic load-balancing. The main limitation of the current code which would be hard to do away with is the fact that only "coarse grain" parallelism is possible (i.e. domains can be partitioned only at the GfsBox level not at an individual cell level). This is a limitation only when the ratio of total number of cells to total number of CPUs becomes small however.

What's ultimately needed to implement "full" load-balancing is a way in the code to transfer entire GfsBoxes from one processor to the other (ensuring the correct restructuring of associated boundary conditions). This is a technical problem but which could be solved relatively easily by someone with a good understanding of the code structure at the GfsBox/GfsBoundary level.

An intermediate possibly easier (but not as clean) solution can be to save the simulation and stop the code when load-balancing becomes too bad, then do a static load-balancing step and restart a "new" simulation with this as initial state. I would probably first experiment with this approach to get used to the problems involved first and then move to the full internally-coded solution.

1.0.12 Can Gerris handle moving/deforming solid geometries?

Not yet but this is planned for a next phase.

1.0.13 Can Gerris also be used for compressible fluids?

No, but this would be possible. The existing shallow-water solver in particular can be seen as one form of compressible flow solver.

1.0.14 Can Gerris solve the shallow-water (Saint-Venant) equations?

Yes, starting with version 0.6.0, although I would not consider it ready for "general consumption" right now. For simple examples on how this works have a look in the source file in `test/ocean`.

1.0.15 How can I assist you in your development effort?

Thanks for asking. The easiest way you can help me is first by using the code. Setting up your own test cases etc... And reporting problems, either in term of usability, unexpected results etc...

By doing that you will certainly help me ensure that the code is as robust as possible and you will soon find areas which need improvement and which you might like to work on (preferably after consultation with me so that we can coordinate our efforts).

An important point is also to remember to try to send your questions/comments to one of the two Gerris mailing lists (gfs-users or gfs-devel) so that other people can benefit from the exchange (I will also more readily reply to a message on the mailing list than to one addressed directly to me).

2 Installation and coding

2.0.1 How do I install the parallel version of Gerris?

If when running `./configure` you got lines looking like

```
checking for mpicc... yes
```

then you don't have anything else to do. Otherwise, you need to make sure that you have MPI installed and that the `mpicc` command is in your `PATH`.

For running the code in parallel, you will have to wait until I have written the next chapter of the tutorial...

2.0.2 Does the Gerris configure script support a LAM MPI implementation?

Yes, more generally it should support any MPI implementation which defines a working `mpicc` command accessible in the `PATH`.

2.0.3 My crappy MPI installation prevents Gerris from compiling, how do I turn MPI support off?

```
% ./configure --disable-mpi
```

2.0.4 Is there a Windows version of Gerris?

Certainly not a native Windows version (Gerris relies on features found in professional operating systems) but one should be able to compile and install it on Windows using cygwin¹⁰.

Gerris also runs on MacOSX.

My personal advice would be "why use Windows?"

2.0.5 Are there any plans to release a more documented version of the code?

I have chosen the "classical" point of view that, if the general description of the arguments and of what the function does (given just before the body of (almost) all the exported functions) together with the code in the function itself does not clearly describe what the function is doing, then this is a problem with the code itself not with the documentation. A counter-example of that would be a very long monolithic code described by comments every few lines.

Also, from my personal experience, working with a number of research and commercial codes, I would consider Gerris to be fairly well documented. The code is also quite modular, so you shouldn't (hopefully) need to go through all the 16000 lines of code...

Of course, I would be glad to address any detailed problem you may have (unclear documentation etc...)

¹⁰<http://www.cygwin.com>

2.0.6 Are there any plans to release a C++/Java/Object Oriented implementation of Gerris?

No. Gerris is already object-oriented (with class inheritance etc...), see the tutorial for an example of how this works.

3 Physics and dimensioning

3.0.1 Where are variables like viscosity, density etc... defined?

By default, the density is unity and the molecular viscosity is zero (i.e. there is no explicit viscous term in the momentum equation). In practice, it does not mean that there is no viscosity at all however, because any discretisation scheme always has some numerical viscosity. Of course, the lower the numerical viscosity, the better. Gerris has quite good properties in this respect.

3.0.2 How come Gerris generates a Von Karman vortex street for an inviscid flow around a half-cylinder? I would expect the inviscid flow to remain irrotational.

This is perfectly right in the case of flow around smooth solid boundaries. If there is a sharp corner (as for the half-cylinder), the potential flow solution is singular in the sense that the velocity tends to infinity as one gets closer to the corner. In practice (finite difference numerical solution) and in reality, the local numerical (or real) viscosity near the corner, smears out the singularity, which results in the creation of a (point) source of vorticity which is then carried away by the mean flow (as you can see on the half-cylinder example).

Even in the case of a smooth geometry, numerical inaccuracies in the boundary conditions on the solid surface can lead to the generation of a small amount of vorticity (much smaller than what is generated at a discontinuity though).

3.0.3 How would I create a 5×5 box?

It is possible to change the size of the unit `GfsBox`, however, I would encourage you to think in “relative units” rather than “absolute units”. When studying fluid mechanics (and other physical) problems it is almost always a good idea to use non-dimensional units. This makes relevant independent parameters (such as the Reynolds number for example) immediately apparent. When using Gerris I would recommend scaling all your physical input parameters by a reference length (the physical length of the `GfsBox`). This also eliminates the need for changing the length of the `GfsBox`.

3.0.4 How would I modify the file you sent me (`tangaroa.gfs`) for a ship that is 150 meters long and exposed to a cross-flow wind velocity of 50 meters/sec?

You would have to non-dimensionalise both the model ship geometry and wind speed.

The reference length of the `GfsBox` would be 3×150 meters, so you would scale the model geometry by a factor of $1/(\text{reference length})$ or $1/450$.

You might want to use the `transform` program to do that, something like this:

```
% transform --scale 2.22222e-3 < model.gts > model_scaled.gts
```

You also have to keep in mind that the bottom boundary of a 3D box is at $z = -0.5$. You want to have that coinciding with the sea level (i.e. translate your model vertically by the correct amount).

3.0.5 How would I redimensionalise U,V,W and P?

$$U \text{ meters/sec} = U * U_{ref} = U * 50 \text{ meters/sec}$$

$$V \text{ meters/sec} = V * U_{ref} = V * 50 \text{ meters/sec}$$

$$W \text{ meters/sec} = W * U_{ref} = W * 50 \text{ meters/sec}$$

$$P \text{ Pascals} = P * DENSITY * U_{ref}^2$$

However, keep in mind that the only relevant parameter for the (constant density) Navier-Stokes equations is the Reynolds number. If you do not include any explicit viscous term the (theoretical) Reynolds number is always infinite. In practice this means that the inflow velocity has only a uniform scaling influence on the final solution. For example

simulation 1: inflow velocity set to 1.0

simulation 2: inflow velocity set to 2.0

then, the velocity field of simulation 1 at time t is exactly equal (to machine precision) to the velocity field for simulation 2 at time $t/2.0$, divided by 2.0.

3.0.6 It looks like `t` and `dt` output by `GfsOutputTime` are also scaled? How would I scale `t` and `dt` to time in seconds?

Let's say your reference scale is $L = 450 \text{ meters}$, your reference speed $U = 50 \text{ meters/sec}$, your reference time is then $T = L/U = 9 \text{ sec}$

You thus need to multiply both `t` and `dt` by $T = 9 \text{ sec}$.

3.0.7 How do I scale Vorticity?

The units of vorticity are

$$LT^{-1}/L \rightarrow T^{-1}$$

and

$$T' = T * Lref/Vref$$

therefore

$$VORTICITY' = VORTICITY * Vref/Lref$$

3.0.8 The code provides support for the variable density incompressible Euler equations. Does that mean you can input the density of the fluid density (air, water, etc...)?

Not really if what you mean is a constant density throughout the domain. In the case of the incompressible constant-density Navier-Stokes equations, the density is irrelevant. It is only a scaling factor for the pressure.

What this really means is that Gerris can deal with flows where the density varies across the domain (e.g. a mixture of two miscible fluids, or density variations due to salinity variations in the sea for example).

3.0.9 Although the initialised problem is symmetric, the solution becomes asymmetric as time passes, why?

The code is indeed not perfectly numerically symmetrical. This is due mainly to the tolerance in the solution for the pressure equation, if you decrease the tolerance you should see smaller asymmetries. You can do this using

```
ApproxProjectionParams {
  tolerance = 1e-6
}
ProjectionParams {
  tolerance = 1e-6
}
```

3.0.10 How do I deal with negative values of the pressure?

Your question is interesting, it comes down to the meaning of “pressure” for incompressible flows.

For *compressible* flows “pressure” has a thermodynamic definition and is directly linked to other physical quantities through an equation of state. It is defined on an absolute scale.

For *incompressible* flows “pressure” does not have a thermodynamic definition (there is no equation of state linking it to other physical quantities), rather it comes about as the stress field necessary to enforce the incompressibility condition. In this context, only its gradients are relevant, not its absolute value i.e. one can add any constant to the pressure field without changing the solution.

Conclusion: If you don’t like negative pressures just add any constant necessary to make them positive.

4 Representation of solid boundaries

4.0.1 How do I import my geometry into Gerris?

You need to convert your geometry into a set of triangulated surfaces and be able to export it in the GTS format (very simple, described here¹¹) or alternatively in the STL format (which can be converted to GTS using the `stl2gts` program).

The tricky bit is that the surfaces you export must represent proper solid objects i.e. they must be orientable, closed, manifold and non self-intersecting surfaces.

4.0.2 What CAD package can I use to export STL/GTS files?

Blender can do that and is open-source, also have a look at ac3d, k3d and Pro/Engineer, Rhino. There are plenty of others.

4.0.3 Do I need to tessellate (increase the number of triangles of) my surface before importing it into GTS?

If your solid boundary is exactly defined using a few triangles, there is no need to use more. In short, the mesh size generated by Gerris is completely independent from the “triangle size” of the input surface (in contrast to what happens in “classical” unstructured mesh solvers).

If for example, you want to resolve the boundary layers around your solid, you could tell Gerris to use a “fine enough” mesh like this:

```
GfsRefineSolid 10
```

which tells Gerris to use 10 levels of refinement near the solid surface. “Fine enough” is going to depend on the details of the physics (most importantly Reynolds number) and on the constraints in term of computational time, memory size etc. . .

4.0.4 Which part of the parameter file tells Gerris where the half-cylinder is placed? How do I alter it?

The position of the solid object is defined (obviously) through the coordinates of its vertices. If you created it using a CAD or similar program, you can translate, rotate etc. . . the object using this same program.

Alternatively, you can use the `transform` program which comes with GTS.

```
% transform -h
```

will give you a summary of the transformations you can make, currently

¹¹<http://gts.sourceforge.net/reference/gts-surfaces.html#GTS-SURFACE-WRITE>

Usage: transform [OPTION] < file.gts
Apply geometric transformations to the input.

-r	ANGLE	--rx=ANGLE	rotate around x-axis
-m	ANGLE	--ry=ANGLE	rotate around y-axis
-n	ANGLE	--rz=ANGLE	rotate around z-axis
-s	FACTOR	--scale=FACTOR	scale by FACTOR
-R	FACTOR	--sx=FACTOR	scale x-axis by FACTOR
-M	FACTOR	--sy=FACTOR	scale y-axis by FACTOR
-N	FACTOR	--sz=FACTOR	scale z-axis by FACTOR
-t	V	--tx=V	translate of V along x-axis
-u	V	--ty=V	translate of V along y-axis
-w	V	--tz=V	translate of V along z-axis
-i		--revert	turn surface inside out
-o		--normalize	fit the resulting surface in a cube of size 1 centered at the origin
-v		--verbose	print statistics about the surface
-h		--help	display this help and exit

Reports bugs to popinet@users.sourceforge.net

The resulting (transformed) object is written on the standard output.

For example, if you want the half-cylinder in the second cell do:

```
% transform -t 1 < half-cylinder.gts > half-cylinder1.gts
```

and use `half-cylinder1.gts` in the parameter file.

4.0.5 Are there any tools for converting format-X (not STL) files (generated via a CAD system) to a GTS-format file?

The GTS file format is described here¹². It is very simple. You should be able to write your own filter using your favourite scripting language. You might want to have a look at the `cleanup` utility which comes with GTS (in the `examples/` directory) It will allow you to link unlinked faces, remove duplicate vertices etc. . .

4.0.6 Gerris seem to allow only one solid body, is this correct?

No, there is no limitation on the number and/or complexity of solid bodies (as long as they are properly oriented, manifold geometrical surfaces). Multiple bodies are possible, either as a single GTS file containing multiple separate bodies or as multiple calls to `GtsSurface` in the parameter file with several non-intersecting GTS surfaces.

```
GtsSurfaceFile box_1.gts
```

```
GtsSurfaceFile box_2.gts
```

Note however that the solids cannot intersect.

4.0.7 How do I orient my solid surfaces properly?

The orientation of the faces of your solid defines where the fluid side is (by convention the counter-clockwise (CCW) normal direction to a face points toward the solid side). If your solid is not oriented properly you can use the `--revert` or `-i` option of `transform` to turn it “inside out”.

¹²<http://gts.sourceforge.net/reference/gts-surfaces.html#GTS-SURFACE-WRITE>

4.0.8 It looks like all STL files need to be turned “inside out”. I don’t understand why, but transform -i fixed the problem.

It is just a matter of different conventions. The program you use has chosen to orient the CCW face normals toward the “outside” of the solid object.

4.0.9 Can solids intersect?

No, you first need to use the “boolean operations” or “constructive solid geometry” operations of your solid modeller to generate the union of your solids.

This may change in the future.

4.0.10 We have a problem inserting some GTS files generated from STL files and even inserting the standard GTS files found on the GTS samples¹³ site?

The samples files on the GTS site are not necessarily describing consistent geometric surfaces (i.e. they can be open, non-manifold etc...)

5 Post-processing and Visualisation

5.0.1 Is there a way to view the results and solid(s) at the same time (with the solid in the correct location)? (Geomview question)

Yes, you need to select the Inspect→Appearance→Normalise→None option in the Geomview menu. The default is to “normalise” (i.e. rescale) each object individually (Normalise→Individual option) so that it fits at the centre of the viewed area.

5.0.2 Is there any way to output U, V, W, P, etc... at point (X, Y, Z) in the flow field?

Yes, use

```
GfsOutputLocation { step = 1 } data -0.209371 -0.0166124 -0.449834
```

where the last three numbers are the x, y, z coordinates of the location where you want the values of the variables or

```
GfsOutputLocation { step = 1 } data positions
```

where `positions` is a file containing newline-separated coordinates.

The file `data` will contain the data as described in the first line of the file, a “comment” line starting with `#`.

5.0.3 Where is the description of the format of the data section of saved simulation files?

If you intend to read the simulation files (to convert them or do other operations/calculations etc...) I would highly recommend that you do so through the functions provided by the Gerris library (`gfs_simulation_read()` and so on). This way you will not reinvent the wheel and you will be able to use all the functionalities provided by the library (traversal of the octree structure, computation of gradients, interpolations etc...). This would also ensure that your code is independent of the format changes in the simulation file.

Just to give you an example on how this can bite you:

the `GfsOutputSimulation` object can be used like this (in simulation files)

```
GfsOutputSimulation { step = 0.1 } sim-%3.1f { variables = P,C }
```

¹³<http://gts.sourceforge.net/samples.html>

in this case, the simulations files (`sim-0.1`, `sim-0.2` etc...) will only contain the P and C variables.

Your code which reads the simulation files would need to know about this. The `gfs_simulation_read()` function deals with that for you and other functions give you easy access to this kind of information (what variables were contained in the simulation file etc...)

5.0.4 How do I compute/display the vorticity field with OpenDX?

The `DivCurl` tool in OpenDX works for general grids. You can use it to compute the vorticity (vector) from the U field returned by `GfsImport`, however this will work only for 3D fields.

5.0.5 The GfsOutputSimulation and GfsOutputLocation files only place up to eight decimals. Is there any way to increase the number of decimal places?

Good question. No there isn't, short of editing and recompiling the source code. That would be a nice option to have in the simulation file.

5.0.6 I would like a time-averaged velocity profile, Would I have to specify a number of monitoring points at different heights, or is there a method to time average over a line through the solution domain?

There is no method to do line averaging at the moment, however there is a method which averages (or more exactly stores the sum) of a given variable in time over the whole domain. You can do it like that:

```
EventSum { start = 1 istep = 1 } U SUx
EventSum { start = 1 istep = 1 } V SUy
EventSum { start = 1 istep = 1 } W SUz
EventSum { start = 1 istep = 1 } U*U SU2x
EventSum { start = 1 istep = 1 } V*V SU2y
EventSum { start = 1 istep = 1 } W*W SU2z
OutputSimulation { start = end } simulation-sum {
    variables = SUx,SUy,SUz,SU2x,SU2y,SU2z
}
```

which would add U to SUx at every timestep (`istep = 1`) starting from time 1 (`start = 1`) etc... and `U*U` to `SU2x` at every timestep etc... The resulting sums are then written at the end of the simulation in the file `simulation-sum`. This file can then be post-processed (using `gfs2oogl` for example) to obtain averages, standard deviations etc... (along any curves you want of course).

5.0.7 Using animate, the sequence of images generated by OutputPPM looks weird, what's happening?

This is probably an artefact of the way the `animate` command displays a series of PPM images. What happens is that `OutputPPM` generates PPM images which are just big enough to contain all the data in your simulation e.g. if you use 7 levels of refinement and one box, `OutputPPM` will generate images with 128×128 pixels. If you use an adaptive resolution with a maximum level of 6, the size of the resulting image generated by `OutputPPM` can be anything in 1×1 , 2×2 , 4×4 , 8×8 , 16×16 , 32×32 , 64×64 depending on the maximum number of levels necessary to verify your adaptation criterion. As a result, `animate` can see a series of PPM images with a variable size, if you look carefully you will see that the weird patterns you see are smaller-size images of your simulation, displayed in the top-left corner of the initial image. What `animate` should really do is blank out the previous larger image before displaying the smaller image, to make the difference in size clear.

The solution is simple, you can set the size of the images generated by `OutputPPM` using:

```
OutputPPM { step = 0.05 } tracer.ppm { v = T maxlevel = 6 }
```

which will result in PPM images of size 64×64 , independently of the maximum level of refinement in the simulation.

5.0.8 Why create a new visualisation tool like GfsView? Can't you use existing tools like Mayavi/VTK, OpenDX etc...?

Most visualisation packages assume that the data is defined on either structured Cartesian meshes (this includes curvilinear coordinates) or fully unstructured meshes (tetrahedra etc...).

The octrees used by Gerris need first to be converted into unstructured tetrahedra and then imported into OpenDX etc... This is quite slow and memory-hungry and loses most of the advantages of the octree: in particular the multilevel representation of the solution is very useful from a visualisation point of view.

I am not aware of any good visualisation tool which understands octrees. It would be a good idea to post messages on OpenDX, Mayavi, VTK etc... mailing lists asking about support for octrees. I did that and got little feed back, but more messages would show the developers of these projects that there is a desire for such a feature.

GfsView makes the most of the octree structure to accelerate visualisation, computation of iso-surfaces etc...

6 Running Gerris

6.0.1 Are the files vorticity.gfs and half-cylinder.gfs included in the Gerris distribution?

No, you will need to type them following the tutorial...

6.0.2 Your flow analysis for the RV Tangaroa is just the type of problem I would like to be able to solve quickly, etc... How long did it take to setup and run this problem? Could you send me a copy of the input file?

The longest was to get a "proper" CAD model of the vessel. We had it made by the ship designers but it was full of topological inconsistencies (folds, degenerate faces etc...). It was a real pain to fix it. Once you have a proper orientable, manifold solid there is nothing more to do really.

Here is a parameter file (replace `tangaroa.gts` with your model)

```
1 0 GfsSimulation GfsBox GfsGEdge {} {
  Time { end = 3 }
  GtsSurfaceFile tangaroa.gts
  Refine 5
  RefineSolid 9
  Init {} { U = 1. }
  AdaptVorticity { istep = 1 } { maxlevel = 8 cmax = 1e-2 }
  OutputSolidStats {} stdout
  OutputTime { istep = 1 } stdout
  OutputBalance { istep = 1 } stdout
  OutputProjectionStats { istep = 1 } stdout
  OutputSimulation { start = 0.1 step = 0.1 } simulation-%03.1f {
variables = U,V,W,P
  }
  OutputTiming { start = end } stdout
}
GfsBox { right = BoundaryOutflow left = BoundaryInflowConstant 1. }
```

It took about 12 hours (and 100 MB RAM) on a single CPU 350 MHz Pentium (knowing that the maximum length of the model I use is about a third of the domain size).

6.0.3 How do you modify the mesh to give greater detail in a specific area of flow?

As you saw in the tutorial, the meshing is automatic and follows user-defined criteria. Vorticity and gradient-based criteria for example, can be used. The level of refinement used for both the

initial refinement and the adaptive refinement can both be functions of space, time, other variables etc...which give almost total flexibility. Other criteria can be added within the object-oriented framework of the code if necessary.

6.0.4 Is there any way to initialise the grid so that a fine grid is generated around the surface of the solid and a coarser grid is generated in the flow field (resulting in significantly fewer cells being generated)?

Sure. You can use something like:

```
GfsRefine 6
GfsRefineSolid 8
```

which will first create a uniform level 6 grid and then add two extra levels (up to level 8) only in the cells cut by the solid boundary.

6.0.5 Is it possible to turn off adaptive meshing in defined areas?

You could do this like that for example:

```
AdaptVorticity { istep = 1 } {
    cmax = 1e-2
    levelmin = (x > 0 ? 6 : 0)
    levelmax = (x > 0 ? 6 : 7)
}
```

which would use a constant resolution (6 levels) on the right side of the domain ($x > 0$) and adaptive resolution (up to 7 levels) on the left side.

6.0.6 Is there a way to control the maximum size of a simulation?

Use something like:

```
AdaptVorticity { maxlevel = 10 maxcells = 400000 cmax = 1e-2 }
```

the `maxcells` option tells the adaptive algorithm to use a maximum of 400,000 cells to discretise the domain. When this maximum number is reached the algorithm minimises the maximum cost of the refinement by optimally distributing the cells across the domain. You have to be aware however that this means that the accuracy of the simulation will not be constant in time.

6.0.7 My simulation file looks fine but does not work, why?

Are you sure your text editor does not include special characters in your files? (the infamous DOS line ending comes to mind).

6.0.8 How do I make all the boundaries of a 2D box no-slip?

```
GfsBox {
    top = Boundary { BcDirichlet U 0 }
    bottom = Boundary { BcDirichlet U 0 }
    left = Boundary { BcDirichlet V 0 }
    right = Boundary { BcDirichlet V 0 }
}
```

6.0.9 How do I make boundary conditions time-dependent?

What about this page¹⁴ of the tutorial? As described¹⁵ in the reference manual functions can depend on time (just use variable `t`). You should be able to use this to implement your boundary conditions.

¹⁴<http://gfs.sourceforge.net/tutorial/tutorial/node15.html>

¹⁵<http://gfs.sourceforge.net/reference/gfs-functions.html>

6.0.10 How do I run Gerris in parallel?

The principle is relatively simple. Each `GfsBox` can take a `pid` argument which defines the number of the process on which the solution for this `GfsBox` will be computed. If you take the “half cylinder” example and do something like:

```
4 3 GfsSimulation GfsBox GfsGEdge {} {
  Time { end = 10 }
  Refine 6
  GtsSurfaceFile half-cylinder.gts
  Init {} { U = 1 }
  OutputProjectionStats { step = 0.02 } stderr
  OutputSimulation { step = 1 } simulation-%d-%3.1f {
    variables = U,V,P,C
  }
  OutputTiming { start = end } stderr
}
GfsBox { pid = 0 left = BoundaryInflowConstant 1 }
GfsBox { pid = 1 }
GfsBox { pid = 2 }
GfsBox { pid = 3 right = BoundaryOutflow }
1 2 right
2 3 right
3 4 right
```

if you run this using

```
% gerris2D half-cylinder.gfs
```

it will run on one processor. If you now do

```
% mpirun -np 4 gerris2D half-cylinder.gfs
```

it will run on 4 processor with each of the `GfsBoxes` assigned to a different processor. Gerris takes care of the communications necessary at the boundaries between `GfsBoxes` on different processors.

The data will be written in different files e.g:

```
vorticity-0.ppm, vorticity-1.ppm etc\dots
```

where the number is the pid of the processor.

For a more complete description you will have to wait until I find the time to write the corresponding section in the tutorial.