OPENGR

An Open Framework for Relativistic Simulations

Jon Allen

Center for Relativity, University of Texas at Austin

OpenGR is an open framework, developed here at the **University of Texas at Austin**, for relativistic simulations. OpenGR utilizes the SAMRAI (Structured Adaptive Mesh Refinement Application Infrastructure) libraries developed at Lawrence Livermore National Labratory to manage our MPI (Massively Parallel Instruction set) calls, non-linear solver, and ODE solver.

We are currently finishing stability testing of our feature set for the first release of OpenGR and the output looks promising. The current features which are not outlined to the right and proposed future development are highlighted below.

Current Features

Scalability The OpenGR framework has been tested on up to 256 processors, and seems to be scaling well.

Flexability We have implemented, and are currently testing, several differing formulations of Einstein's equations, as well as constrained evolution of both ADM and NOR formulations. OpenGR provides flexability in implementing evolution equations, setting boundary data, visualization, and choosing initial data.

Data Management OpenGR and supporting analysis tools provide restart capabilities, a well documented HDF5 interface provided by SAMRAI, and batch output analysis through the VisIt visualization application.

Future Improvements

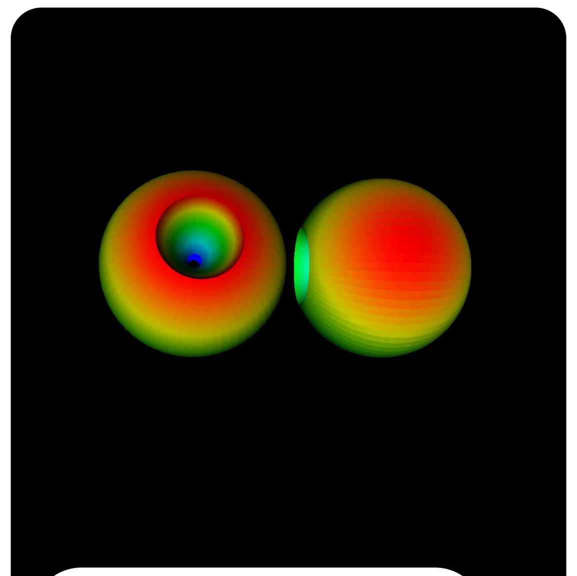
AMR Once we are evolving matter fields, adaptive mesh refinement (AMR) will provide increased stability. This is a planned improvement.

Development API To improve the reusability of our own code, and to encourage others to use our framework as a basis for their simulations, a defined application programming interface (API) is a must. This will ease the burden on developers and allow them to focus more on the physics.

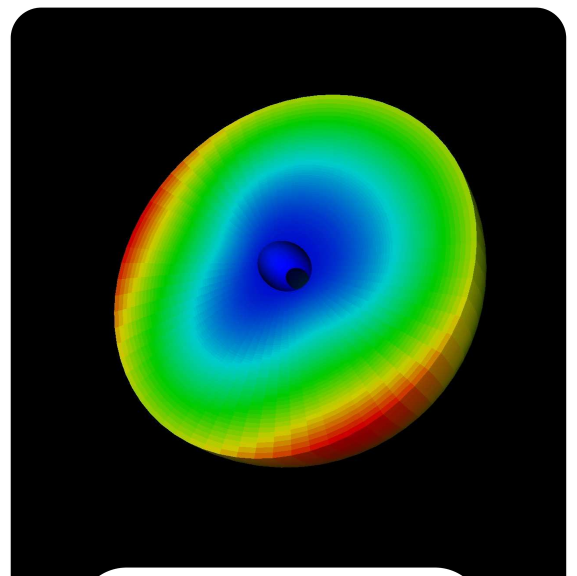
An Introduction to OpenGR

OpenGR provides the capability to define multiple coordinate patches which are interpolated at every timestep. These patches provide more reasonable coordinates in sensitive areas to improve numerical stability. The spheroidal graphic shows the two spheroidal coordinate patches which overlap to surround each of the black hole singularities. A slice of the initial Kerr-Schild g_{zz} metric term for a boosted black hole, $v = 0.7c$, is pictured second from left. To allow the use of spherical boundary conditions, we implemented an outer spherical patch, pictured second from right. A sample implementation of all coordinate patches in unison can be found at far right.

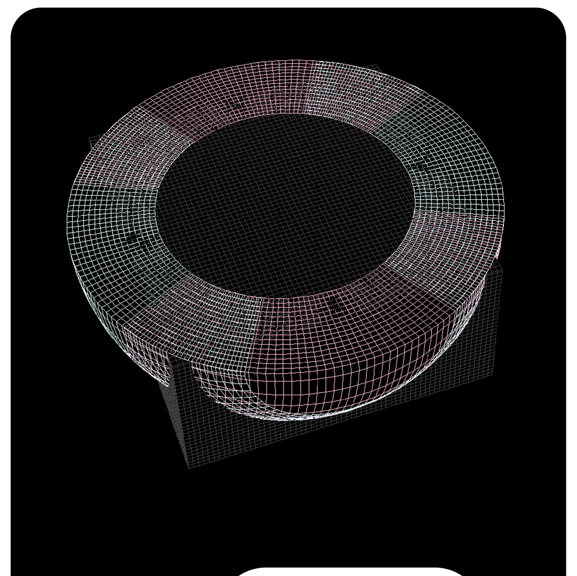
Overlapping Grids



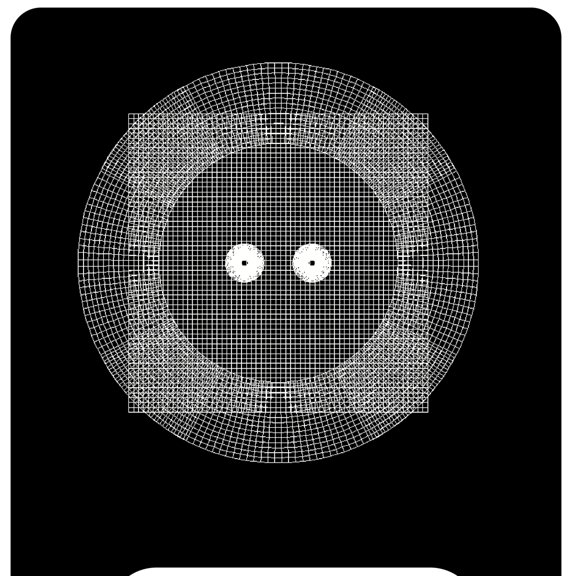
Spheroidal



Gzz Slice



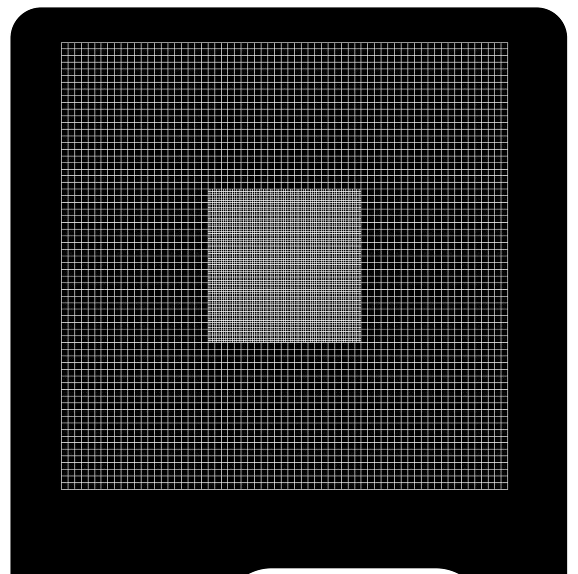
Outer



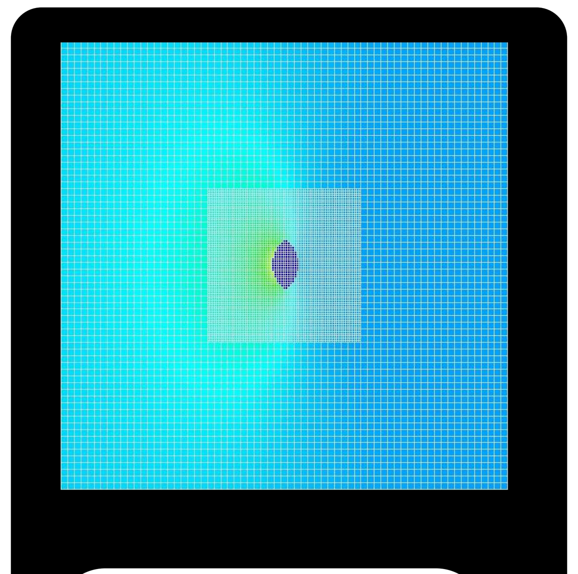
All Grids

Mesh refinement is used to efficiently implement higher resolution in areas of interest. The mesh layout of a simple fixed cartesian refinement surrounding the hole is pictured to the right. This refinement and boosted Kerr-Schild g_{xx} term are pictured second from left. The boosted g_{xx} term without the mesh is pictured second from right; note the differing granularity at the refinement boundary. The mask variable identifies the proper action for a computational zone. The orange areas are assigned boundary values or interpolated, the red area is supplied values interpolated from the spheroidal grids, the dark blue area is the excised region, and the light blue area are the evolved points.

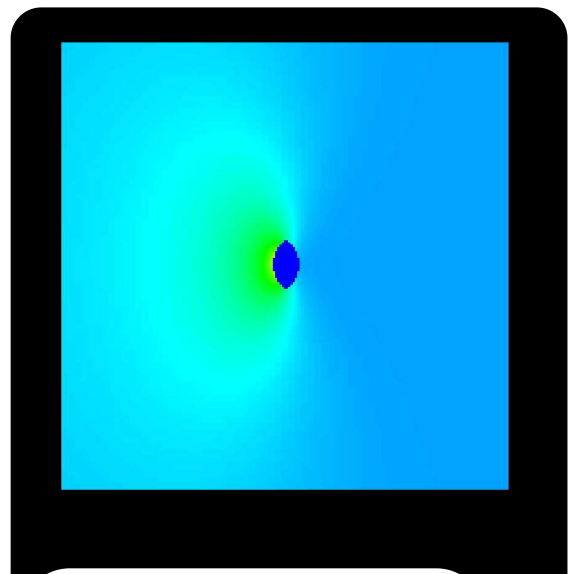
Mesh Refinement



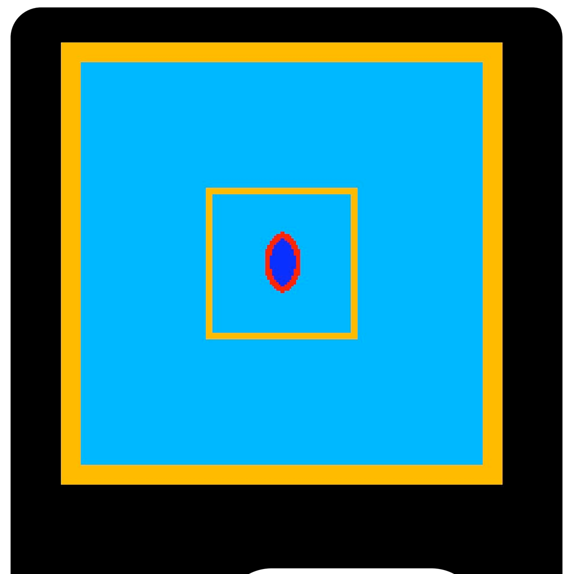
Mesh



Gxx Mesh



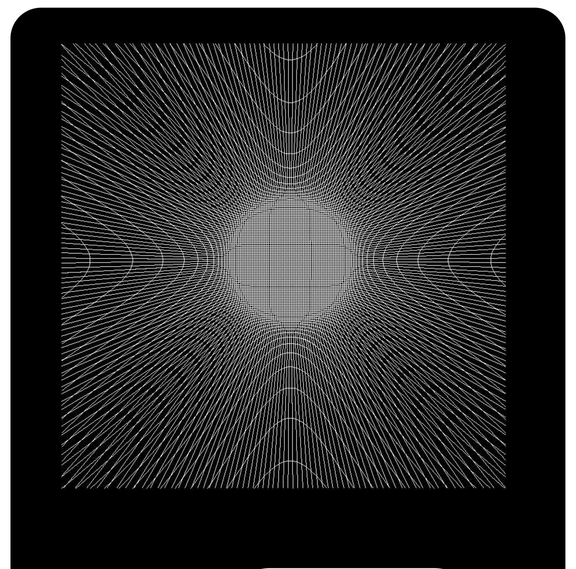
Gxx Refine



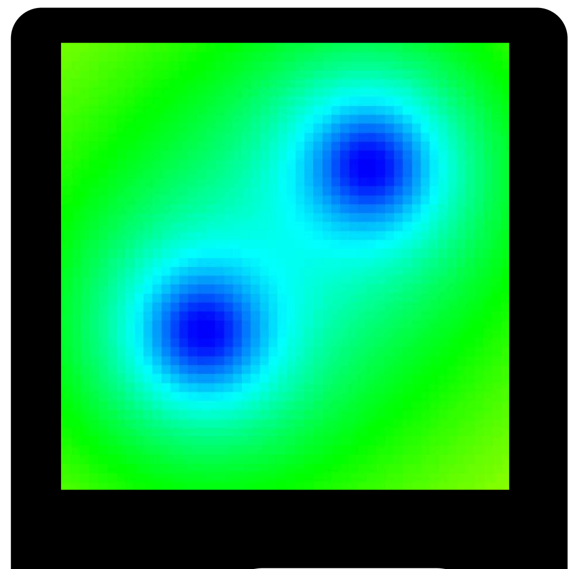
Mask

The fisheye coordinate system is a form of static mesh refinement which changes the cartesian coordinates, moving points outside a fixed radius much further away while keeping a plateau of constant volume zones inside. A fisheye transformed mesh can be seen at right. We use fisheye currently to evolve puncture data, which does not have an excised region, but foliates space-time in such a way so that on each slice, the area surrounding the hole is not evolved. This "many fingered time" parameter, α, is pictured (enlarged) second from left. The shift vector for this slicing with spin, $a = \pm 0.8$, is pictured (enlarged) second from right. The determinant of the metric is pictured at far right.

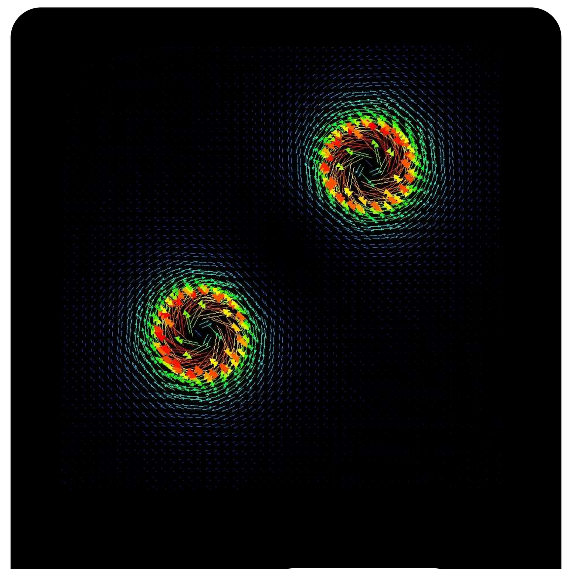
Fisheye & Puncture



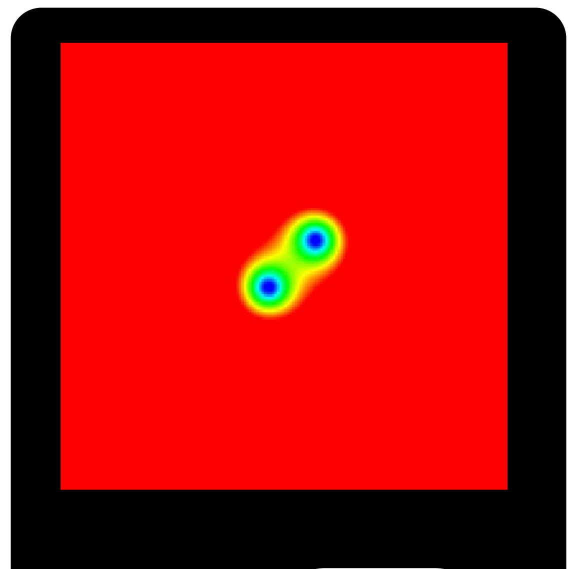
Mesh



Alpha



Beta



Chi