Simulations of Binary Black Hole Mergers

With the Einstein Toolkit

Daniel George
Graduate student
Department of Astronomy
University of Illinois at Urbana-Champaign

Write as if in a paper for publication — "In this paper I/we use..." etc.

Methods/Tests

Overview of the code

I will be using the Einstein toolkit for simulating the merger of binary black holes in order to analyze the gravitational waves emitted during these events. A brief overview of the codes that I am using is provided below.

Cactus Framework

citation(s)

The Cactus Framework is a free and open-source, collaborative, modular environment for high performance parallel computing. It is highly portable and can run on diverse architectures from desktops and laptop computers to supercomputers such as Blue Waters. It has seen applications across a variety of fields including numerical relativity, computational fluid dynamics, and quantum gravity.

The Cactus Framework originated from code developed by Ed Seidel's group at NCSA prior to 1995 for solving problems in numerical relativity, which was later developed at AEI and LSU.

It consists of a common infrastructure, called the flesh, along with numerous modules, called thorns, which can be selectively plugged in to implement different functionality.

The Einstein Toolkit

citation(s)

The Einstein toolkit is a particular collection of over 100 thorns within the Cactus Framework that is suited for solving, analyzing and visualizing problems encountered in relativistic astrophysics. It is currently actively developed and maintained by researchers in various institutes across the world. It comes packaged with an Adaptive Mesh Refinement (AMR) driver called Carpet.

Equations to be solved

To model the merger of black holes we have to solve Einstein's field equations in vacuum space time. These are a set of 10 highly nonlinear partial differential equations. Although the original form of Einstein's field equations is independent of any coordinate system, we have to constrain ourselves to a particular set of coordinates in order to numerically solve the equations. However, not all choices of coordinate systems can be integrated stably. Here we choose a particular approach known as the BSSN (Baumgarte-Shapiro-Shibata-Nakamura) formalism to solve for the long-term evolution of multiple black hole systems. The equation used in this formalism are given below.

BSSN formulation of Einstein equations

$$
\begin{aligned}
\partial_t \tilde{\gamma}_{ij} &= -2\alpha \tilde{A}_{ij} + \beta^k \partial_k \tilde{\gamma}_{ij} + \tilde{\gamma}_{ik}\partial_j \beta^k + \tilde{\gamma}_{jk}\partial_i \beta^k - \frac{2}{3}\tilde{\gamma}_{ij}\partial_k \beta^k, \\
\partial_t \phi &= -\frac{1}{6}\alpha K + \beta^k \partial_k \phi + \frac{1}{6}\partial_k \beta^k, \\
\partial_t \tilde{A}_{ij} &= e^{-4\phi}[-D_i D_j \alpha + \alpha R_{ij}]^{TF} + \alpha(K\tilde{A}_{ij} - 2\tilde{A}_{ik}\tilde{A}^k{}_j) \\
&\quad + \beta^k \partial_k \tilde{A}_{ij} + \tilde{A}_{ik}\partial_j \beta^k + \tilde{A}_{jk}\partial_i \beta^k - \frac{2}{3}\tilde{A}_{ij}\partial_k \beta^k, \\
\partial_t K &= -D^i D_i \alpha + \alpha(\tilde{A}_{ij}\tilde{A}^{ij} + \frac{1}{3}K^2) + \beta^k \partial_k K, \\
\partial_t \tilde{\Gamma}^i &= \tilde{\gamma}^{jk}\partial_j \partial_k \beta^i + \frac{1}{3}\tilde{\gamma}^{ij}\partial_j \partial_k \beta^k + \beta^j \partial_j \tilde{\Gamma}^i - \tilde{\Gamma}^j \partial_j \beta^i + \frac{2}{3}\tilde{\Gamma}^i \partial_j \beta^j \\
&\quad - 2\tilde{A}^{ij}\partial_j \alpha + 2\alpha(\tilde{\Gamma}^i{}_{jk}\tilde{A}^{jk} + 6\tilde{A}^{ij}\partial_j \phi - \frac{2}{3}\tilde{\gamma}^{ij}\partial_j K).
\end{aligned}
$$

Here $R_{ij} = \tilde{R}_{ij} + R^{\phi}_{ij}$, where

$$
\begin{aligned}
R^{\phi}_{ij} &= -2\tilde{D}_i \tilde{D}_j \phi - 2\tilde{\gamma}_{ij}\tilde{D}^k \tilde{D}_k \phi + 4\tilde{D}_i \phi\, \tilde{D}_j \phi - 4\tilde{\gamma}_{ij}\tilde{D}^k \phi\, \tilde{D}_k \phi, \\
\tilde{R}_{ij} &= -\frac{1}{2}\tilde{\gamma}^{lm}\partial_l \partial_m \tilde{\gamma}_{ij} + \tilde{\gamma}_{k(i}\partial_{j)}\tilde{\Gamma}^k + \tilde{\Gamma}^k \tilde{\Gamma}_{(ij)k} \\
&\quad + \tilde{\gamma}^{lm}\left(2\tilde{\Gamma}^k{}_{l(i}\tilde{\Gamma}_{j)km} + \tilde{\Gamma}^k{}_{im}\tilde{\Gamma}_{klj}\right).
\end{aligned}
$$

(Image source: Slides on "Introduction to the Einstein Toolkit" by Roland Haas)

Number figures

KRANC

Since the explicit form of the BSSN equations are extremely lengthy, we use a Mathematica based application called KRANC to convert the tensorial form of the equations to highly efficient and portable C/Fortran code that can be plugged into the Cactus code as a module.

Did you need to do this yourself? Or was there already a BSSN thorn?

Results

The program was successfully executed once with the initial conditions specified in the parameter file that I uploaded previously. However, the output files were purged from my scratch directory over spring break. On attempting to re-run the program, the job that I submitted has been waiting in the NCSA primary queue on the campus cluster for over 49 hours. At that point I simply gave up and uploaded this.

References

[1] Introduction to the Einstein Toolkit - Roland Haas (slides)

[2] The Einstein Toolkit: A Community Computational Infrastructure for Relativistic Astrophysics

[3] An Introduction to the Einstein Toolkit

[4] Kranc: a Mathematica application to generate numerical codes for tensorial evolution equations

[5] http://einsteintoolkit.org/

[6] http://cactuscode.org/

Journal article references?

You should explain the test problem(s) you do to make sure your setup works. You also need to explain how initial conditions are constructed. The code you submitted is not formally part of the paper.