Gauge Choice in Numerical Evolution of the Brill Data

A. Khirnov and T. Ledvinka

Charles University in Prague, Faculty of Mathematics and Physics, Prague, Czech Republic.

Abstract. While numerical relativity has largely resolved the most important problem of a binary black hole collision and merger with the help of the moving puncture gauge, there are still interesting regimes that elude successful numeric treatment. One such is the evolution of axisymmetric gravitational waves, followed for sufficiently strong field by their collapse into a black hole. A specific family of initial data, the Brill waves, has proven difficult to evolve with common hyperbolic gauge conditions. We present a comparison of the effects of different gauge choice on the numerical evolution of this data.

Introduction

The evolution of pure gravitational waves has been the object of some attention in numerical relativity. One motivation for this is that it allows us to study gravitational collapse and black hole formation using just a vacuum code, without having to deal with matter dynamics.

Pioneering simulations of this kind [*Abrahams and Evans* 1993, 1994] used a family of axisymmetric initial data, known as the Teukolsky waves [*Teukolsky* 1982], with one parameter a controlling the strength of the waves. The results showed that there exists a critical amplitude a^* such that for $a < a^*$ the waves disperse to infinity and leave flat space behind, while for $a > a^*$ the waves collapse and form a black hole. For values of a close to a^*, critical behaviour could be observed with some properties markedly similar to those reported by [*Choptuik* 1993] for the spherically symmetrical collapse of a scalar field.

Several other published attempts used a different family of axisymmetric initial data, the Brill waves [*Brill* 1959], again with a single amplitude parameter. The Brill data is defined by the following spatial line element in cylindrical coordinates $\{\rho, z, \varphi\}$

$$\mathrm{d}l^2 = \psi^4 \left[e^{2q}\left(\mathrm{d}\rho^2 + \mathrm{d}z^2\right) + \rho^2 \mathrm{d}\varphi^2\right] \tag{1}$$

with a chosen "seed function" $q(\rho, z)$ and the assumption of time symmetry of the initial slice, which implies vanishing extrinsic curvature $K_{ij} = 0$. The momentum constraints are then satisfied trivially and the Hamiltonian constraint reduces to

$$\Delta\psi + \frac{1}{4}\left(\partial_\rho q + \partial_z q\right)\psi = 0 \tag{2}$$

with Δ the flat space Laplacian. A common choice of the seed function — e.g. [*Alcubierre et al.* 2000] and others — and the one we also use in this paper, has the form

$$q = A\rho^2 e^{-\rho^2 - z^2} \tag{3}$$

where A is the amplitude parameter.

Several published results about Brill wave simulations used maximal slicing and were able to observe the formation of an apparent horizon for sufficiently strong initial data [*Alcubierre et al.* 2000, *Garfinkle et al.* 2001, *Rinne* 2008]. Maximal slicing, however, is not very practical for 3D simulations so there has been interest in hyperbolic gauge conditions for evolving this data. Of note is therefore the result from [*Hilditch et al.* 2013] showing that the moving puncture gauge, commonly used for 3D black hole simulations, fails for the Brill data close to the critical amplitude.

In our ongoing work, we are trying to advance further in this direction. We have written a pseudospectral elliptic solver to construct the Brill data in the Einstein Toolkit framework. Building on it, we have also implemented the maximal slicing condition. In this paper, we present a comparison of the Brill wave evolution with the 1+log and maximal slicings.

The evolution system

We follow the standard 3+1 splitting procedure [*Misner et al.* 2008; see also *Baumgarte et al.* 2010, *Alcubierre* 2008] for the introduction in the context of numerical relativity) to decompose the

4-dimensional metric into the spatial part γ_{ij}, lapse α and shift β^i

$$\mathrm{d}s^2 = (-\alpha^2 + \beta_i\beta^i)\mathrm{d}t^2 + 2\beta_i \mathrm{d}t\mathrm{d}x^i + \gamma_{ij}\mathrm{d}x^i\mathrm{d}x^j. \tag{4}$$

The specific form of the evolution equations we use is BSSN [*Shibata et al.* 1995, *Baumgarte et al.* 1998], where the spatial metric γ_{ij} and the extrinsic curvature K_{ij} are replaced with new variables

$$\begin{aligned} \varphi &= \det(\gamma_{ij})^{-\frac{1}{6}} & \bar{A}_{ij} &= \varphi^2\left(K_{ij} - \frac{1}{3}K\right) \\ K &= \gamma_{ij}K^{ij} & \bar{\Gamma}^i &= \bar{\gamma}^{jk}\bar{\Gamma}^i_{jk} \\ \bar{\gamma}_{ij} &= \varphi^2\gamma_{ij} \end{aligned} \tag{5}$$

where $\bar{\Gamma}^i_{jk}$ are the Christoffel symbols associated with $\bar{\gamma}_{ij}$. The evolution equations for those quantities then are

$$\begin{aligned} (\partial_t - \mathcal{L}_\beta)\,\varphi &= \frac{1}{3}\varphi\alpha K, \\ (\partial_t - \mathcal{L}_\beta)\,\bar{\gamma}_{ij} &= -2\alpha\bar{A}_{ij}, \\ (\partial_t - \mathcal{L}_\beta)\,K &= -D^2\alpha + \alpha\left(A_{ij}A^{ij} + \frac{1}{3}K^2\right), \\ (\partial_t - \mathcal{L}_\beta)\,\bar{A}_{ij} &= \varphi^2\left[-D_iD_j\alpha + \alpha R_{ij}\right]^{\mathrm{TF}} + \alpha\left(K\bar{A}_{ij} - 2\bar{A}_{ik}\bar{A}^k_j\right), \\ (\partial_t - \mathcal{L}_\beta)\,\bar{\Gamma}^i &= \bar{\gamma}^{jk}\partial_j\partial_k\beta^i + \frac{1}{3}\bar{\gamma}^{ij}\partial_j\partial_k\beta^k - 2\bar{A}^{ij}\partial_j\alpha \\ &\quad + 2\alpha\left(\bar{\Gamma}^i_{jk}\bar{A}^{jk} + 6\bar{A}^{ij}\partial_j\varphi - \frac{2}{3}\bar{\gamma}^{ij}\partial_j K\right), \end{aligned} \tag{6}$$

where $\mathcal{L}$ is the Lie derivative, D_i and R_{ij} are respectively the covariant derivative and the Ricci tensor associated with γ_{ij} and $[\dots]^{\mathrm{TF}}$ designates the trace-free part of the expression in brackets.

The shift is always set to zero in the simulations described here, while the lapse is one of:

$$(\partial_t - \mathcal{L}_\beta)\,\alpha = -2\alpha K \qquad \text{“1+log” slicing} \tag{7}$$

$$D^2\alpha - K_{ij}K^{ij}\alpha = 0 \qquad \text{maximal slicing} \tag{8}$$

Numerical method

Initial data

As the problem is axisymmetric, we solve the Brill wave equation (2) on any $\varphi = \text{const.}$ half-plane with a pseudospectral method, writing the conformal factor as

$$\psi = 1 + \sum_{i=0}^{N_x-1}\sum_{j=0}^{N_z-1} c_{ij}\mathrm{SB}_{2i}(\rho)\mathrm{SB}_{2j}(z) \tag{9}$$

where $N_{\{i,j\}}$ is the number of basis functions chosen in each direction and $\mathrm{SB}_i(x)$ are the basis functions complementary to the “rational Chebyshev functions” [*Boyd* 1987]:

$$\mathrm{SB}_i(x) = \sin\left((n+1)\operatorname{arccot}\left(\frac{x}{L}\right)\right). \tag{10}$$

Here L is the scale parameter of the mapping, which needs to be tuned for the problem being solved; we have determined empirically that a value of 3 works well. The even basis functions SB_{2i} are symmetric with respect to the origin and decay to zero as x^{-1} when $x \to \infty$. Those are exactly the boundary conditions imposed on ψ, so we need not enforce them explicitly.

By demanding that the equation (2) be satisfied exactly at the N_xN_z collocation points, we obtain a set of N_xN_z linear equations for the coefficients c_{ij}, which we solve by LU decomposition with `LAPACK`. Through the equation (9) we then evaluate ψ everywhere on the numerical grid.

Evolution

We use the Einstein Toolkit [*Löffler et al.* 2012] as the basis for our simulations. The numerical grids with a fixed mesh refinement scheme are provided by the Carpet software [*Schnetter et al.* 2004]. The implementation of the evolution equations (6)–(7) comes from the McLachlan project.

The evolved metric quantities (except for the lapse when maximal slicing is used) are discretized on equidistant numerical grids. The grids form a nested hierarchy, with each finer one using a twice smaller distance between the grid points. The grids are all centered around the coordinate origin.

We use 8th order centered finite differences to approximate spatial derivatives. The method of lines is then used to integrate the evolution equations in time, with the standard RK4 method as the time integrator. Kreiss–Oliger dissipation is applied to damp high-frequency noise.

Since the data is symmetric with respect to the plane $z = 0$, we apply reflection boundary conditions there and evolve only the $z \geq 0$ region. To exploit the axial symmetry, we use the Cartoon [*Alcubierre et al.* 2001] method. That is — in the y direction we only keep a thin slice around $y = 0$, just enough for the finite difference operators at $y = 0$. Those extra points are repopulated at each time step by interpolating the $y = 0$ values. The interpolation operator used is fourth order accurate. On the $x = 0$ plane we also apply the reflection symmetry condition. On the outer boundaries, we apply the outgoing wave boundary conditions to all the evolved variables (see part 5.4.2 in [*Löffler et al.* 2012] for details).

Maximal slicing

In the runs using maximal slicing, we enforce the constraint $K = 0$ at all times. Motivated mainly by reducing code development effort, we use the same pseudospectral method as for the initial data (reusing some of that code) to solve the equation (8) for the lapse. The main difference from the initial data is that here we have to interpolate the data from the Carpet grids onto the pseudospectral grid. Because of the mesh refinement being used, this interpolation is both spatial and temporal. For this to work, the pseudospectral grid needs to be contained within the coarsest Carpet one, which can be achieved by suitably picking the scaling parameter L.

For performance reasons, we now use a combination of the LU decomposition and the BiCGSTAB iterative method for solving the linear system. From the LU decomposition of the pseudospectral matrix, we calculate its inverse and use it as the preconditioner for subsequent iterative solutions. As the initial guess we simply take the last obtained solution. Once the convergence rate of the iteration drops sufficiently, we do a LU decomposition again.

In practice it turns out that the method used just as described above produces a large amount of high-frequency noise, presumably due to aliasing instabilities. The problem can be mitigated by using a filter that damps the highest coefficients. We use the same form of the filter as in [*Szilágyi et al.* 2009], that is

$$\mathcal{F}(c_{ij}) = e^{-\alpha\left(\frac{i}{N_x-1}\right)^p} e^{-\alpha\left(\frac{j}{N_z-1}\right)^p} c_{ij} \tag{11}$$

with the same value of $\alpha = 36$, but a much smaller value of p seems to be required in our simulations. We assume this is caused by the specifics of our case, but more investigation is needed. We use $p = 5$ in the runs described here.

Results

We run two sets of simulations, using maximal and 1+log slicing respectively. All the simulations use five levels of mesh refinement, with the coarsest grid covering $\{x, z\} = \{0 - 64, 0 - 64\}$. The highest-resolution runs, from which most of the plots are drawn, use 256 grid points in the x and z directions at each level. For testing convergence, we also run simulations with 192 and 128 grid points. For maximal slicing, we typically use 60 basis functions in each direction, with 40 and 80 for convergence testing.

In Fig. 1 we show the results of the convergence tests. For 1+log slicing, we plot the convergence factor, defined for two simulations a and b as $\log\left(||H_a||_\infty / ||H_b||_\infty\right) / \log(k)$, where k is the ratio of the grid resolutions. The solid line is for the 128/192-point runs, while the dashed one is for 192/256 points. Both lines seem to be very roughly consistent with fourth order convergence, which is the order of the Cartoon interpolation. The convergence factor also falls off rapidly shortly before the crash at $t \approx 5.5$.

With maximal slicing we have a mix of pseudospectral and finite difference algorithms on different grids, so we cannot as easily define an order of convergence. For a pure spectral code, with all the functions analytic, one hopes for exponential convergence with the number of basis functions. That, however, is clearly not the case here. Therefore, on the left subplot we simply show the infinity norm of the Hamiltonian constraint for simulations with 128 grid points and 40, 60 and 80 basis functions in

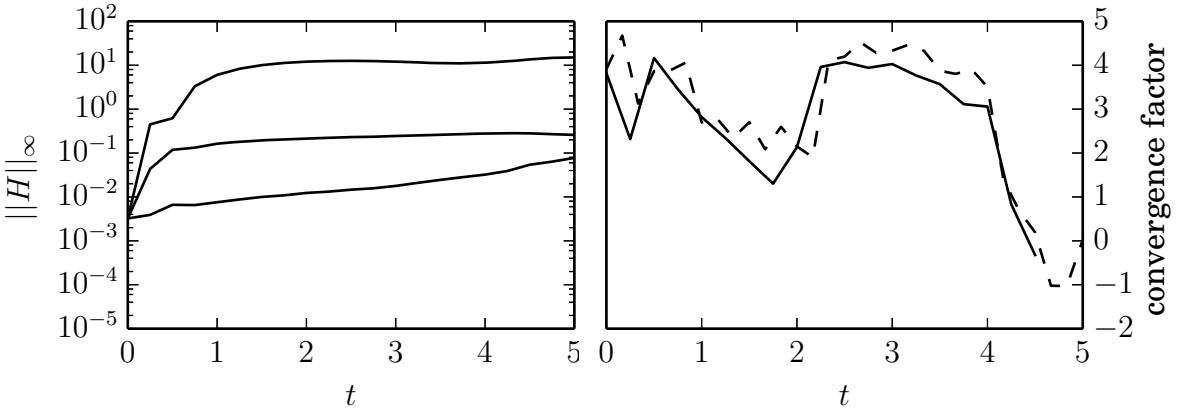
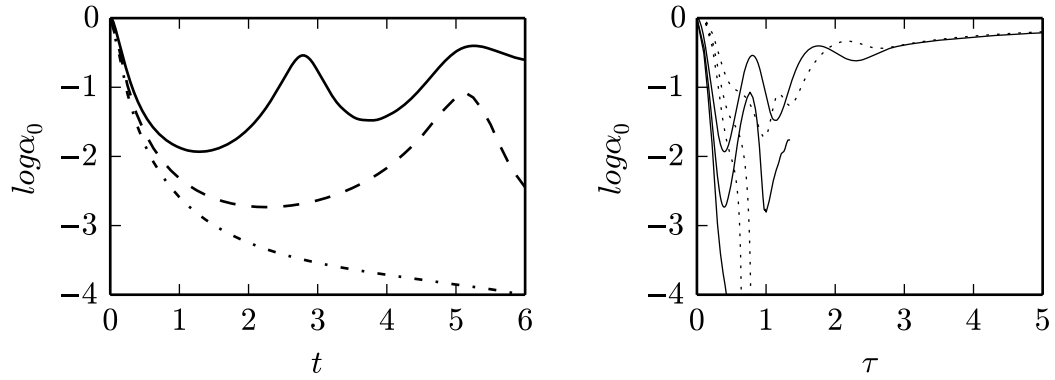


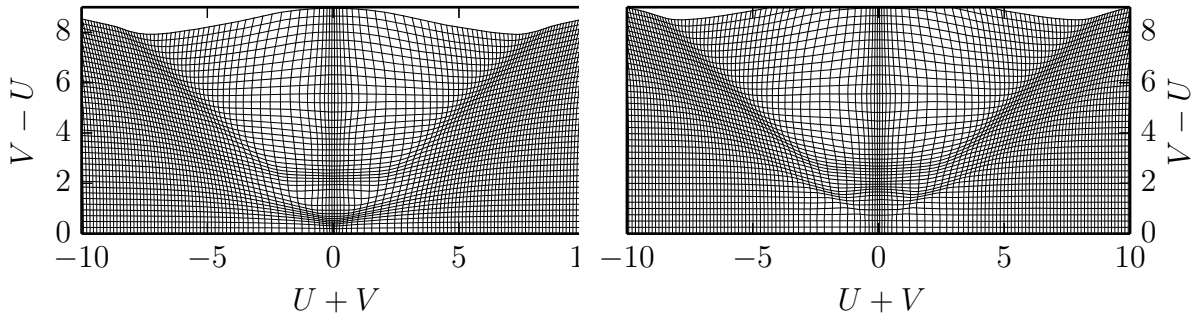
Figure 1. Convergence of the Hamiltonian constraint with time, $A = 5$. Right — 1+log, plotted is the convergence factor, 128/192 and 192/256 points; left — maximal, plotted is the norm of the constraint, since the convergence factor cannot be easily defined, 40/60/80 basis functions. See main text for details.



(a) As a function of the coordinate time t, maximal slicing. Cf. figure 2 in [*Garfinkle et al.* 2001].

(b) As a function of the proper time τ at the origin, maximal slicing (solid) and 1+log slicing (dashed).

Figure 2. Evolution of the value of $\log(\alpha)$ at origin for amplitudes $A = \{4, 5, 6\}$ (top to bottom).



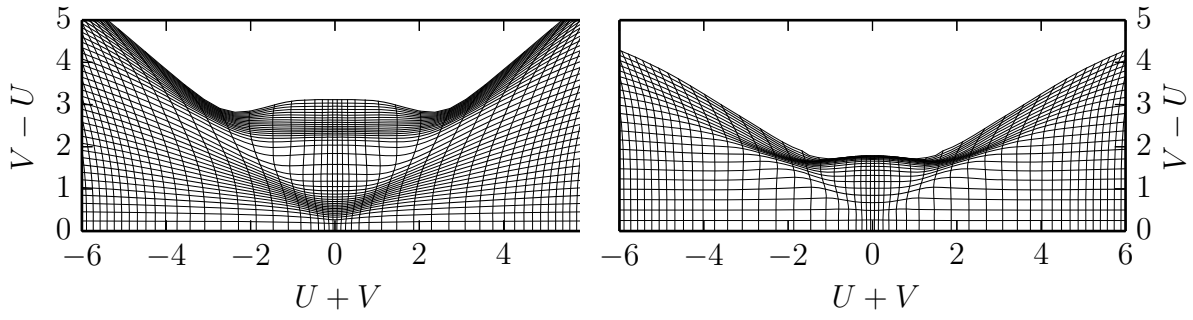
(a) Maximal slicing

(b) 1+log slicing

Figure 3. Grid coordinates (x, t) as functions of the null coordinates (U, V), amplitude $A = 4$.

each direction (top to bottom). Clearly, the pseudospectral code is now the main source of error, which converges away as more basis functions are added. We assume this is primarily because of very strong filtering applied to the coefficients, so the filter process will require more attention.

Our maximal slicing code supports parallelism through shared-memory multithreading, provided by OpenMP and an optimized BLAS implementation [*OpenBLAS*]. However, it does not yet support MPI,



(a) Maximal slicing

(b) 1+log slicing

Figure 4. Grid coordinates (x, t) as functions of the null coordinates (U, V), amplitude $A = 5$. Note that the 1+log simulation runs only until $t \approx 5.5$, where it crashes. The maximal run finishes at $t = 10$.

so our simulations are essentially limited by the amount of working memory available to one CPU. The machine we use has an Intel i7 processor with 6 physical (12 logical) cores and 64GB of RAM. Because of those performance constraints, we run each simulation until coordinate time $t = 10$ or, in the case of the supercritical 1+log runs, until it crashes.

Using the Γ-driver shift [*Alcubierre et al.* 2003] together with maximal slicing has proved to be unstable. For the time being, we work around this by simply setting the shift to zero in all the simulations (both maximal and 1+log ones). However, it is clear that a non-zero shift vector is necessary for successful long-term runs containing black holes, so this issue will need to be investigated in detail.

We mainly concentrate on the simulations with amplitudes $A = \{4, 5, 6\}$. The literature indicates that $A = 4$ is subcritical and the other two are supercritical, with $A = 5$ being quite close to critical. In Fig. 2 we plot the behaviour of the lapse at coordinate origin. On the left we show the dependence on the coordinate time for maximally sliced runs only — the results are consistent with [*Alcubierre et al.* 2000, *Garfinkle et al.* 2001]. On the right we use the proper time τ, which allows us to compare different slicings.

One other way to compare differently sliced runs is used in Fig. 3 and 4. We restrict ourselves to the x-t plane and introduce double null coordinates (U, V), which are constant along null geodesics emitted in the positive and negative x directions respectively. For ease of presentation we use linear combinations $U + V$ and $V - U$ in the plots. The specific values are defined by requiring that $U + V$ is equal to the proper distance from the origin at $t = 0$, so $U(t = 0, x) = V(t = 0, x) = \frac{1}{2}\int_0^x \sqrt{\gamma_{xx}(t = 0, x)}\mathrm{d}x$. We then integrate the null geodesics forward in time and from the resulting data we obtain the functions $t_{x=x_i}(U + V, V - U)$, $x_{t=t_j}(U + V, V - U)$ by interpolation.

While for $A = 4$, both slicings are usable, for $A = 5$, [*Hilditch et al.* 2013] reports that the 1+log-sliced simulation appears to develop a coordinate singularity and subsequently crashes at coordinate time $t \approx 5.5$. We are able to reproduce the same result with the Einstein Toolkit (the original article used the BAM code). In Fig. 4 we see how the lapse collapses and coordinate time stops advancing long before it would be really necessary. In contrast, the maximally sliced run has a much smoother behaviour and is able to continue beyond that point.

Conclusion

We have implemented the Brill wave initial data solver and maximal slicing within the Einstein Toolkit framework and verified that they work correctly. Using that code, we have run 1+log- and maximally-sliced evolutions of the Brill data with results matching existing literature. We have visualised the comparisons of the two slicings using coordinate-independent quantities to better understand the pathologies in the supercritical 1+log-sliced simulations.

In the continuation of this work, we plan to:

- adapt the maximal slicing code, so that it can be used efficiently on clusters with a large number of CPUs,
- investigate the shift conditions,
- use the insight gained from comparing the maximal and 1+log slicings to attempt to construct a better behaved hyperbolic slicing.

Acknowledgments. This work is supported by the Charles University in Prague, project GA UK No 2000314 and SVV-260211. T.L. would like to thank the project GACR 14-37086G. We would also like to thank the developers of the Einstein Toolkit for making open-source numerical relativity possible.

References

Abrahams A. M., and Evans C. R., Critical behavior and scaling in vacuum axisymmetric gravitational collapse, *Phys. Rev. Lett.*, 70, 2980–2983, 1993.

Abrahams A. M., and Evans C. R., Universality in axisymmetric vacuum collapse, *Phys. Rev. D*, 49, 3998–4003, 1994.

Alcubierre M., Introduction to 3+1 Numerical Relativity, Oxford University Press, UK, 2008.

Alcubierre M., Allen G., Brügmann B., Lanfermann G., Seidel E., Suen W., and Tobias M., Gravitational collapse of gravitational waves in 3D numerical relativity, *Phys. Rev. D*, 61, 041501, 2000.

Alcubierre M., Brandt S., Brügmann B., Holz D., Seidel E., Takahashi R., and Thornburg J., Symmetry without symmetry: Numerical simulation of axisymmetric systems using Cartesian grids., *Int. J. Mod. Phys. D*, 10, 273–289, 2001.

Alcubierre M., Brügmann B., Diener P., Koppitz M., Pollney D., Seidel E., and Takahashi R., Gauge conditions for long-term numerical black hole evolutions without excision, *Phys. Rev. D*, 67, 084023, 2003.

Arnowitt R. L., Deser S., and Misner C. W., The dynamics of general relativity, *General Relativity and Gravitation*, 1997–2027, 2008.

Baumgarte T. W., and Shapiro S. L., Numerical integration of Einstein's field equations, *Phys. Rev. D*, 59, 024007, 1998.

Baumgarte T. W., and Shapiro S. L., Numerical Relativity: Solving Einstein's Equations on the Computer, Cambridge University Press, 2010.

Boyd J. P., Spectral methods using rational basis functions on an infinite interval, *Journal of Computational Physics*, 69, 112–142, 1987.

Brill D. R., On the positive definite mass of the Bondi–Weber–Wheeler time-symmetric gravitational waves, *Annals of Physics*, 7, 466–483, 1959.

Choptuik M. W., Universality and scaling in gravitational collapse of a massless scalar field, *Phys. Rev. Lett.*, 70, 9–12, 1993.

Garfinkle D., and Duncan, G. C., Numerical evolution of Brill waves, *Phys. Rev. D*, 63, 044011, 2001.

Hilditch D., Baumgarte T. W., Weyhausen A., Dietrich T., Brügmann B., Montero P. J., and Müller E., Collapse of nonlinear gravitational waves in moving-puncture coordinates, *Phys. Rev. D*, 88, 103009, 2013.

Löffler F., Faber J., Bentivegna E., Bode T., Diener P., Haas R., Hinder I., Mundim B. C., Ott C. D., Schnetter E., Allen G., Campanelli M., and Laguna P., The Einstein Toolkit: a community computational infrastructure for relativistic astrophysics, *Classical and Quantum Gravity*, 29, 115001, 2012.

McLachlan, a Public BSSN Code, `http://www.cct.lsu.edu/ eschnett/McLachlan/`

OpenBLAS, an optimized BLAS library `http://www.openblas.net/`

Rinne O., Constrained evolution in axisymmetry and the gravitational collapse of prolate Brill waves, *Classical and Quantum Gravity*, 25, 135009, 2008.

Schnetter E., Hawley S. H., and Hawke I., Evolutions in 3-D numerical relativity using fixed mesh refinement, *Class. Quantum Grav.*, 21, 1465–1488, 2004.

Shibata M., and Nakamura T., Evolution of three-dimensional gravitational waves: Harmonic slicing case, *Phys. Rev. D*, 52, 5428–5444, 1995.

Szilágyi B., Lindblom L., and Scheel M. A., Simulations of binary black hole mergers using spectral methods, . *Phys. Rev. D*, 80, 124010, 2009.

Teukolsky S. A., Linearized quadrupole waves in general relativity and the motion of test particles, *Phys. Rev. D*, 26, 745–750, 1982.