

Numerical relativity using a generalized harmonic decomposition

Frans Pretorius

Theoretical Astrophysics, California Institute of Technology, Pasadena, CA 91125, USA

Received 30 July 2004, in final form 6 December 2004

Published 3 January 2005

Online at stacks.iop.org/CQG/22/425

Abstract

A new numerical scheme to solve the Einstein field equations based upon the generalized harmonic decomposition of the Ricci tensor is introduced. The source functions driving the wave equations that define generalized harmonic coordinates are treated as independent functions, and encode the coordinate freedom of solutions. Techniques are discussed to impose particular gauge conditions through a specification of the source functions. A 3D, free evolution, finite difference code implementing this system of equations with a scalar field matter source is described. The second-order-in-space-and-time partial differential equations are discretized directly without the use of first-order auxiliary terms, limiting the number of independent functions to 15—ten metric quantities, four source functions and the scalar field. This also limits the number of constraint equations, which can only be enforced to within truncation error in a numerical free evolution, to four. The coordinate system is compactified to spatial infinity in order to impose physically motivated, constraint-preserving outer boundary conditions. A variant of the cartoon method for efficiently simulating axisymmetric spacetimes with a Cartesian code is described that does not use interpolation, and is easier to incorporate into existing adaptive mesh refinement packages. Preliminary test simulations of vacuum black-hole evolution and black-hole formation via scalar field collapse are described, suggesting that this method may be useful for studying many spacetimes of interest.

PACS numbers: 04.25.Dm, 04.40.–b, 04.70.Bw

1. Introduction

One of the primary goals of numerical relativity today is to solve for astrophysical spacetimes that are expected to be strong sources of gravitational wave emission in the frequency bands relevant to current and planned gravitational wave detectors. Expected sources include the

inspiral and merger of compact objects, supernovae, pulsars and the big bang. An important tool for extracting physics from detector signals is the technique of matched filtering, which requires an accurate waveform of a model of the expected source. For binary black-hole mergers (in particular) it is thought that numerical relativity is the only method that will be able to provide such waveforms close to and during the plunge phase of the merger. Despite significant progress made over the past decade, a full solution to this problem still eludes researchers. One reason for the difficulty is the complexity of the field equations. This translates into significant computer resources being needed to solve the equations, which limits the turn-around time for testing new ideas. However, perhaps the largest obstacle so far has been finding a formalism to write the field equations in that is amenable to long-term, stable numerical evolution. Some of the promising techniques used today include symmetric hyperbolic formalisms [1, 2], the BSSN formalism (sometimes referred to as the NOK formalism) [3–6] and characteristic evolution (for black-hole/neutron-star systems) [7]. Several groups are also beginning to examine the possibility of constrained evolution for the 3D binary black-hole problem [8–10], and other promising directions make use of tetrad formulations of the field equations [12–14], and solution of the conformal field equations [15–18].

A method of writing the field equations that has proven very useful in analytic studies is arrived at by imposing the *harmonic coordinate condition*, where the four spacetime coordinates x^μ are chosen to individually satisfy wave equations: $\square x^\mu = 0$. The Einstein equations, when written with this condition imposed, take on a mathematically appealing form where the principal part of each partial differential equation satisfied by a metric component $g_{\alpha\beta}$ becomes the scalar wave operator $\square g_{\alpha\beta}$. This allowed for (among other things) the first existence and uniqueness proof of solutions to the field equations [19]. In numerical relativity, a solution scheme based directly upon this formulation of the field equations has recently been suggested by Garfinkle [20] (see also related work by Szilagyi and Winicour [21], and the so-called Z4 system [22], which seems to be quite similar to generalized harmonic evolution in many respects). Garfinkle considered a generalization of the harmonic coordinate condition of the form $\square x^\mu = H_\mu$, where H_μ are now arbitrary source functions, and found that the technique was successful in simulations of the approach to the singularity in certain cosmological spacetimes.

One purpose of this paper is to begin to investigate the use of the generalized harmonic decomposition in asymptotically flat spacetimes. The formalism is described in section 2. If this method is to be useful for a large class of spacetimes, one issue that needs to be addressed is how to choose gauge conditions via specification of the source functions H_μ ; this topic is discussed in section 3. A second goal of this paper is to investigate direct discretization of the second-order-in-space-and-time partial differential equations¹ (in other words, the system is *not* converted into a system of first-order equations before discretization). One reason for doing so is to have a free evolution scheme where the *only* constraints amongst the variables are the four constraint equations imposed by the Einstein equations (see also [24, 21]). The hope then is that even if this system suffers from ‘constraint violating modes’², it may be easier to analyse and cure them using (for instance) ideas suggested in research of symmetric hyperbolic

¹ Recent analytic investigations by Calabrese [23] have suggested that such a scheme may suffer from high-frequency instabilities in situations where the coefficients in front of mixed time–space derivatives are greater than the local characteristic speed. We have not yet noticed such an instability, probably because of the numerical dissipation we use, which was one of the suggested cures for the problem in [23].

² By constraint violating mode we mean a solution to the *continuum evolution* equations that is not a solution of the full Einstein equations, and furthermore exhibits exponential growth from initial data with arbitrarily small deviations from putative initial data that does satisfy the constraints.

versions of the field equations [25–29]³. The numerical code is described in section 4, along with related topics such as apparent horizon finding, excision, boundary conditions, initial conditions and the current scalar field matter source. Also described in section 4 is a variant of the *cartoon* method [32] to efficiently simulate axisymmetric spacetimes with a Cartesian code. The advantages of the method presented here are that no interpolation is used, and the axisymmetric simulation is performed on a two-dimensional slice of the Cartesian grid. In section 5 test simulations of black-hole evolution and gravitational collapse are shown, suggesting that this solution method holds promise for simulating asymptotically flat spacetimes. Concluding remarks are given in section 6, in particular, a discussion of some of the work that still needs to be done before the code could provide new physical results in situations of interest.

2. The Einstein field equations in the generalized harmonic decomposition

Consider the Einstein field equations in the form

$$R_{\alpha\beta} = 4\pi(2T_{\alpha\beta} - g_{\alpha\beta}T), \quad (1)$$

where $R_{\alpha\beta}$ is the Ricci tensor, $g_{\alpha\beta}$ is the metric tensor, $T_{\alpha\beta}$ is the stress–energy tensor with trace T , and units have been chosen so that Newton’s constant G and the speed of light c are equal to 1. The Ricci tensor is defined in terms of the Christoffel symbols $\Gamma_{\alpha\beta}^{\gamma}$ by

$$R_{\alpha\beta} = \Gamma_{\alpha\beta,\delta}^{\delta} - \Gamma_{\delta\beta,\alpha}^{\delta} + \Gamma_{\alpha\beta}^{\epsilon}\Gamma_{\epsilon\delta}^{\delta} - \Gamma_{\delta\beta}^{\epsilon}\Gamma_{\epsilon\alpha}^{\delta} \quad (2)$$

where $\Gamma_{\alpha\beta}^{\gamma}$ is

$$\Gamma_{\alpha\beta}^{\gamma} = \frac{1}{2}g^{\gamma\epsilon}[g_{\alpha\epsilon,\beta} + g_{\beta\epsilon,\alpha} - g_{\alpha\beta,\epsilon}]. \quad (3)$$

The notation $f_{,\alpha}$ and $\partial_{\alpha}f$ is used interchangeably to denote ordinary differentiation of some quantity f with respect to the coordinate x^{α} .

Introduce a set of four *source functions* H^{μ} via

$$H^{\mu} \equiv \square x^{\mu} \quad (4)$$

$$= \frac{1}{\sqrt{-g}}\partial_{\alpha}(\sqrt{-g}g^{\alpha\beta}x_{,\beta}^{\mu}) \quad (5)$$

$$= \frac{1}{\sqrt{-g}}\partial_{\alpha}(\sqrt{-g}g^{\alpha\mu}), \quad (6)$$

or, equivalently, defining $H_{\mu} = g_{\mu\nu}H^{\nu}$, we have

$$H_{\mu} = (\ln \sqrt{-g})_{,\mu} - g^{\alpha\nu}g_{\nu\mu,\alpha}. \quad (7)$$

The symmetrized gradient of H_{μ} is thus

$$H_{(\mu,\nu)} = (\ln \sqrt{-g})_{,\mu\nu} - g^{\alpha\beta}{}_{(\nu}g_{\mu)\beta,\alpha} - g^{\alpha\beta}g_{\beta(\mu,\nu)\alpha}. \quad (8)$$

The generalized harmonic decomposition involves replacing particular combinations of first and second derivatives of the metric in the Ricci tensor (2) by the equivalent quantities in (7), (8), and then *promoting the source functions H_{μ} to the status of independent quantities*. Specifically, one can rewrite the field equations (1) as

$$g^{\delta\gamma}g_{\alpha\beta,\gamma\delta} + g^{\gamma\delta}{}_{,\beta}g_{\alpha\delta,\gamma} + g^{\gamma\delta}{}_{,\alpha}g_{\beta\delta,\gamma} + 2H_{(\alpha,\beta)} - 2H_{\delta}\Gamma_{\alpha\beta}^{\delta} + 2\Gamma_{\delta\beta}^{\gamma}\Gamma_{\gamma\alpha}^{\delta} = -8\pi(2T_{\alpha\beta} - g_{\alpha\beta}T). \quad (9)$$

³ For it appears that it may *not* be possible to construct a constrained–transport-type numerical evolution scheme that satisfies all of the Einstein equations to machine precision [30, 31].

As H_μ are now four independent functions, one needs to provide four additional, independent differential equations to solve for them, which we write schematically as

$$\mathcal{L}_\mu H_\mu = 0 \quad (\text{no summation}). \quad (10)$$

\mathcal{L}_μ is a differential operator that in general can depend upon the spacetime coordinates, the metric and its derivatives, and the source functions and their derivatives. Note however that the principal part of (9) is now the simple wave operator $g^{\delta\gamma}\partial_\gamma\partial_\delta$ acting upon each metric component $g_{\alpha\beta}$; this subsystem of equations is manifestly hyperbolic given certain reasonable conditions on the metric⁴ and as long as the coupling between (9), (10) and any matter evolution equations that may be needed do not affect the characteristic structure of (9). We will not discuss the well-posedness of this system of equations here, though this is certainly a topic worth pursuing.

The Einstein field equations are thus equivalent to the system of equations (9) and (10), *provided* that the harmonic ‘constraints’ (7) are satisfied for all time $t \equiv x^0$. The claim then is that, at the analytical level, if (9) is used to evolve $g_{\alpha\beta}$, and (10) is used to evolve H_μ , then (7) will be satisfied for all time provided that initial conditions are specified so that (7) and (8) are satisfied then. For the special case where H^μ are given as *a priori* functions of the coordinates x^μ , the preceding statement has been proven before [33] (the case $H^\mu = 0$ was first shown in [19]). The idea behind the proof is as follows. Define the harmonic constraint function C^μ as

$$C^\mu \equiv H^\mu - \square x^\mu. \quad (11)$$

For any solution to the Einstein equations (1), C^μ is identically zero. Using the contracted Bianchi identity and conservation of stress energy, one can show that C^μ satisfies the following homogeneous wave equation:

$$\square C^\mu = -R^\mu{}_\nu C^\nu. \quad (12)$$

Therefore, given any $g_{\mu\nu}$ that satisfies (9) for all time together with some H^μ that satisfies both $C^\mu = 0$ and $\partial_t C^\mu = 0$ at $t = 0$, (12) guarantees that $g_{\mu\nu}$ will also solve the Einstein equations (1) for all time. We cannot prove such a result for a general evolution system where H^μ is specified via some arbitrary set of differential equations. Rather, we will take the more pragmatic approach in the numerical code of demonstrating convergence to a solution of the Einstein equations for any particular evolution system we use. In fact, such a convergence test is the *only* measure of the validity of the numerical solution, regardless of any analytic properties of the underlying continuum problem.

Equivalent to enforcing (8) at $t = 0$ is to make sure that the usual constraint equations, namely

$${}^{(3)}R + K^2 - K_{ab}K^{ab} = 16\pi\rho, \quad (13)$$

$$K_a{}^b{}_{|b} - K_{|a} = 8\pi J_a \quad (14)$$

are satisfied then, which from a practical standpoint is easier to solve than (8) using existing, well-established techniques [2, 42]. In the above, K_{ab} is the extrinsic curvature tensor of the $t = \text{const}$ hypersurface with the induced metric h_{ab} , K is the trace of K_{ab} , ${}^{(3)}R$ is the Ricci scalar of h_{ab} , $|$ denotes the covariant derivative operator compatible with h_{ab} , and ρ and J_a are the projected matter energy and momentum densities respectively. Note that we use notation where Greek indices denote four-dimensional quantities and run from 0 to 3, and Latin indices denote three-dimensional spatial quantities and run from 1 to 3.

⁴ For example one would need a single coordinate to be timelike and the rest to be spacelike throughout the integration volume.

At this stage the system (9), (10) is completely general in that we have not yet specified any time slicing or spatial coordinates. Choosing a gauge amounts to specifying a set of source functions through (10), and thus the source functions play a role analogous to the lapse function and shift vector in the traditional ADM decomposition. One disadvantage of the harmonic decomposition is that (to my knowledge) there is no simple geometric description of the relationship between H_μ and the resulting spacetime coordinates. In some cases one can appeal to the ADM lapse and shift view of coordinate freedom to motivate a particular choice of H_μ . We will discuss these and several other classes of gauge conditions that may be useful for numerical evolution in the following section.

3. Specifying a gauge

Within the generalized harmonic decomposition one can think of the source functions H_μ as representing the four coordinate degrees of freedom available in general relativity. There are many conceivable ways of choosing H_μ ; in this section we will give a few suggestions, several of which are used in the evolutions presented in section 5. However, the discussion here is rather heuristic in that we do not consider how any of these gauge choices may affect the character of the coupled Einstein-gauge evolution system. Note that gauge source functions were discussed by Friedrich [43] in some detail, though not specifically within the context of supplying additional evolution equations for them.

The simplest gauge choice in this formalism is to set the source functions equal to some arbitrary functions of the spacetime coordinates:

$$H_\mu = f_\mu(x^\alpha). \quad (15)$$

The case $f_\mu = 0$ is standard harmonic coordinates. The next condition we consider is a coordinate system that evolves towards harmonic coordinates:

$$\partial_t H_\mu = -\kappa_\mu(t) H_\mu \quad (\text{no summation}), \quad (16)$$

where κ_μ are a set of four arbitrary though positive functions of time, which if non-zero will cause H_μ to evolve to zero.

A useful method to derive coordinate conditions for the harmonic decomposition is to appeal to the manner in which the coordinate system is specified in the ADM decomposition. This makes available a tremendous amount of research that has gone into gauge related issues for the ADM-based evolution [2]. In the ADM formalism the metric element is written as

$$ds^2 = -\alpha^2 + h_{ij}(dx^i + \beta^i dt)(dx^j + \beta^j dt), \quad (17)$$

where the *lapse function* α measures the rate of change of proper time with respect to coordinate time t of hypersurface normal observers, h_{ij} is the intrinsic metric of $t = \text{const}$ slices, and the *shift vector* β^j describes how the spatial coordinates change for normal observers from one time slice to the next. The normal component and spatial projection of the source function H_μ are⁵

$$\begin{aligned} H \cdot n &\equiv H_\mu n^\mu \\ &= -K - \partial_\nu (\ln \alpha) n^\nu \end{aligned} \quad (18)$$

$$\begin{aligned} \perp H^i &\equiv H_\mu h^{\mu i} \\ &= -\bar{\Gamma}_{jk}^i h^{jk} + \partial_j (\ln \alpha) h^{ij} + \frac{1}{\alpha} \partial_\nu \beta^i n^\nu, \end{aligned} \quad (19)$$

⁵ Note however that H_μ does not transform as a one-form under coordinate transformations, and hence the projections are not covariant objects.

where $\bar{\Gamma}_{jk}^i$ is the connection associated with h_{ij} , and n^ν is the hypersurface normal vector given by

$$n_\nu = -\alpha \partial_\nu t. \quad (20)$$

Note that in (18) and (19) the time derivative of α only appears in $H \cdot n$, while the time derivative of β^i only appears in the corresponding component of $\perp H^i$. In other words, the choice of the normal component $H \cdot n$ in an evolution directly affects the rate of change of α with respect to time, and therefore $H \cdot n$ controls the time-slicing of the spacetime; similarly, $\perp H^i$ controls the manner in which the spatial coordinates evolve with time (another way of stating this is that (18), (19) are generalizations of the hyperbolic equations governing the lapse and shift within harmonic coordinates [44]). One way in which an ADM style gauge condition can be used within the harmonic decomposition is to substitute the corresponding choices of α and β^i into (18), (19), and use the result as the source functions for the harmonic evolution. The simplest class of gauge conditions that can be implemented in this fashion are the so-called ‘driver’ conditions [45–49], where one directly specifies the time derivatives of α and β^i to achieve, for example, approximate maximal slicing and minimal distortion gauges respectively.

The manner in which the ADM driver conditions are implemented suggests a similar way in which such gauge conditions can be used in a harmonic evolution: instead of substituting in the forms for α and β^i in (18), (19) to try to satisfy the conditions exactly, choose source functions to *drive* the gauge towards the desired one. To see how this can be done, first rewrite (18), (19) as evolution equations for the lapse and shift:

$$\partial_t \alpha = -\alpha^2 H \cdot n + \dots \quad (21)$$

$$\partial_t \beta^i = \alpha^2 \perp H^i + \dots, \quad (22)$$

where the ellipses denote the rest of the terms that do not contain $\partial_t \alpha$, $\partial_t \beta^i$ or H^μ . Now suppose at some instant of time the *desired* value of the lapse and shift are calculated (by whatever means) to be α_0 and β_0^i respectively. Then from (21), (22) *one possible* set of choices for the source functions that will cause the lapse and shift to evolve towards the desired values are

$$H \cdot n = \kappa_n(t) \frac{\alpha - \alpha_0}{\alpha^2} \quad (23)$$

$$\perp H^i = -\kappa_i(t) \frac{\beta^i - \beta_0^i}{\alpha^2} \quad (\text{no summation}), \quad (24)$$

where κ_n and κ_i are positive functions of time that can be used to control the rate of evolution.

In many circumstances it may make more sense to implement the above style driver conditions as evolution equations, rather than algebraic conditions. This could be, for example, if the initial conditions for H_μ are not compatible with the desired gauge choice, and so implementing (23), (24) will result in discontinuous source functions at the initial time. A couple of alternative possibilities include

$$\frac{\partial H \cdot n}{\partial t} = \kappa_n(t) \frac{\alpha - \alpha_0}{\alpha^2} \quad (25)$$

and

$$\square(H \cdot n) = -\kappa_n(t) \frac{\alpha - \alpha_0}{\alpha^2} + \xi_n(t) (H \cdot n)_{,\mu} n^\mu, \quad (26)$$

with similar expressions for the spatial parts of H_μ . $\xi_n(t)$ is a positive function that can be used to add a dissipative term to (26). One advantage of using a wave operator (26) to evolve

the source functions is then the principal parts of all equations in the system (9), (10) have the same characteristic structure (as long as α_0 and β_0^i depend at most on first derivatives of the fundamental variables). This may be important to establish well-posedness of the coupled system of equations [50].

It is beyond the scope of this paper to investigate how well any of these suggested gauge conditions performs in situations of interest; however in section 5 we will show some preliminary results indicating that these ideas can be implemented in a stable fashion.

4. Numerical code

In this section we describe a 3D numerical code based upon the generalized harmonic decomposition. This code has several features of note.

- *Direct discretization of (9), (10).* In other words, we do not convert the system of equations into the first-order form—the only variables used are the ten unique metric components $g_{\alpha\beta}$, four source functions H_μ and matter variables.
- *Spatially compactified Cartesian coordinates.* The spatial coordinates are compactified to include i^0 , to simplify the imposition of physically realistic boundary conditions for asymptotically flat spacetimes.
- *Black-hole excision.* Black-hole excision is used to evolve spacetimes containing black holes, whereby portions of the computational domain inside apparent horizons are ‘excised’ to remove the singularities.
- *Built within a parallel adaptive mesh refinement (AMR) framework.* The code utilizes a new set of parallel AMR libraries which we will describe elsewhere, though the Berger and Olinger style AMR algorithm used is very similar to the one presented in [51, 52].
- *Efficient simulation of axisymmetric spacetimes using a variant of the Cartoon method [32].* The algorithm presented here does not require interpolation, and only utilizes a single two-dimensional slice of the Cartesian grid, simplifying incorporation into existing AMR packages.

In the remainder of this section we will describe certain aspects of the code in more detail.

4.1. Discretization scheme

Here onwards we will use the coordinate names $t \equiv x^0$ and $(x, y, z) \equiv (x^1, x^2, x^3)$. Also, as discussed in the next section, we use a compactified coordinate system in the code. This necessitates the use of regularized variables for some of the metric and source function components; however to keep the discussion in this section simpler we ignore that aspect of the code here. In appendix B we present a stability analysis of this discretization method applied to a one-dimensional wave equation in flat space. The purpose of the analysis is to give a simple, concrete example of the numerical method, and to show that there are no fundamental instabilities in it. Of course, this cannot prove that the full, nonlinear problem in compactified coordinates will be stable—doing so is beyond the scope of this paper.

We use second-order accurate finite difference techniques to discretized (9), (10) and the scalar field evolution equation (45) presented in section 4.4. This is a set of 15 equations for 15 unknown functions—the ten non-trivial metric components $g_{\alpha\beta}$, the four source functions H_μ and the scalar field Φ . In the discretized version of (9) all Christoffel symbols, contravariant metric elements and their gradients are replaced with the appropriate sum of covariant metric elements and their gradients. As (9) and (45) are second-order partial differential equations in time, second-order accurate discretization requires a three time-level scheme

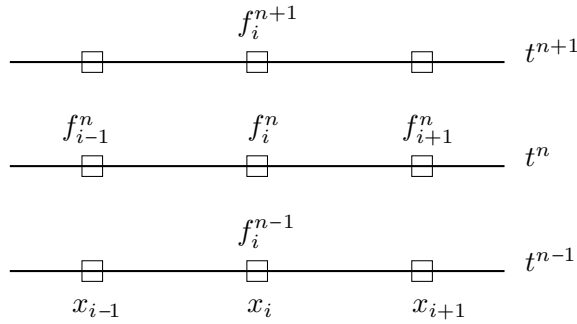


Figure 1. The discretization of a variable $f(t, x, y, z)$ in the $x-t$ plane.

Table 1. A sample of the finite-difference stencils used to convert the differential equations into difference equations. The column on the right shows the second-order accurate representation (with y and z indices suppressed for clarity) of the corresponding derivative operator to the left, evaluated at the point (t^n, x_i, y_j, z_k) . Similar stencils are used for terms containing y and z derivatives.

| | |
|-----------|--|
| $f_{,x}$ | $(f_{i+1}^n - f_{i-1}^n) / (2\Delta x)$ |
| $f_{,t}$ | $(f_i^{n+1} - f_i^{n-1}) / (2\Delta t)$ |
| $f_{,xx}$ | $(f_{i+1}^n - 2f_i^n + f_{i-1}^n) / (\Delta x)^2$ |
| $f_{,tt}$ | $(f_i^{n+1} - 2f_i^n + f_i^{n-1}) / (\Delta t)^2$ |
| $f_{,tx}$ | $(f_{i+1}^{n+1} - f_{i-1}^{n+1} - f_{i+1}^{n-1} + f_{i-1}^{n-1}) / (4\Delta x \Delta t)$ |

(at a minimum). Figure 1 shows a schematic representation (with two spatial dimensions suppressed) of the discretization of a variable $f(t, x, y, z)$. $f(t, x, y, z)$ evaluated at a grid location $(t^n, x_i, y_j, z_k) = (n\Delta t, i\Delta x, j\Delta y, k\Delta z)$ is denoted by f_{ijk}^n , where n, i, j and k are integers, and $\Delta t, \Delta x, \Delta y$ and Δz are the temporal and spatial discretization scales respectively. Table 1 contains a representative sample of the finite-difference operators used to evaluate derivatives on the mesh. Replacing the continuum variables with discrete variables, and the derivative operators with difference operators will result in a difference equation

$$\mathcal{L}_f |_{ijk}^n = 0 \tag{27}$$

for each variable f at each grid point $(t, x, y, z) = (t^n, x_i, y_j, z_k)$ in the computational domain.

We solve the system of equations (27) using a Newton–Gauss–Seidel relaxation scheme, as follows. Initial data for a single time step at $t = t^n$ consist of all the variables at time levels t^n and t^{n-1} . The unknowns are the variables at time level t^{n+1} . Denote an approximate value of the unknown f_{ijk}^{n+1} by \hat{f}_{ijk}^{n+1} . The iteration is set-up using function values at time level n as an initial guess to the solution at time level $n + 1$. One step of the iteration then proceeds by updating each unknown, in turn, via

$$\hat{f}_{ijk}^{n+1} \rightarrow \hat{f}_{ijk}^{n+1} - \frac{\mathcal{R}_f |_{ijk}^n}{\mathcal{J}_f |_{ijk}^n}, \tag{28}$$

where $\mathcal{R}_f |_{ijk}^n$ is the residual of the difference equation (the left-hand side of (27) evaluated using the approximate solution) and $\mathcal{J}_f |_{ijk}^n$ is the diagonal element of its Jacobian

$$\mathcal{J}_f |_{ijk}^n = \frac{\partial \mathcal{L}_f |_{ijk}^n}{\partial f_{ijk}^{n+1}}, \tag{29}$$

again evaluated with the approximate solution. In other words, (28) is simply solving a linearized version of (27) for f_{ijk}^{n+1} assuming all other unknowns are fixed. The iteration is repeated until the residual for all variables is below some specified tolerance.

4.1.1. Numerical dissipation. Some form of numerical dissipation is necessary to stably evolve certain spacetimes, in particular those that contain black holes. We use Kreiss–Oliger-style dissipation [53]; however, rather than modify the discrete evolution equations as is typically done (and note also that [53] considered first order in time systems), we apply the dissipation as a *filter* to the discrete variables, at *both* past time levels t^n and t^{n-1} , prior to updating t^{n+1} . Specifically, at a given time level we define the high-frequency component η_{ijk}^x of grid function f_{ijk} , in the x direction, as

$$\begin{aligned} \eta_{ijk}^x &= \frac{1}{16}(f_{i-2jk} - 4f_{i-1jk} + 6f_{ijk} - 4f_{i+1jk} + f_{i+2jk}), & 2 < i < N_x - 2 \\ &= 0 & \text{elsewhere,} \end{aligned} \quad (30)$$

where the local size of the mesh is N_x points in the x direction. After η_{ijk}^x has been calculated over the entire local grid, it is subtracted from f as follows:

$$f_{ijk} = f_{ijk} - \epsilon \eta_{ijk}^x, \quad (31)$$

where ϵ is a constant, required to be in the range $0 \dots 1$ for stability. In practice we use values of ϵ in the range 0.2 to 0.5. Once the high-frequency components in the x direction have been subtracted, the procedure is repeated for the high-frequency components η_{ijk}^y and η_{ijk}^z of f_{ijk} in the y and z directions respectively, which are given by expressions similar to (30).

We did experiment with extending the dissipation filter to the grid boundaries as outlined in [54]; however this did not seem to have a significant effect on the solution in most circumstances, and seemed to produce more error (as measured by residuals of the field equations) next to excision boundaries without offering improved stability. However, the excision method proposed in [54] was for cubical excision boundaries, and for schemes satisfying summation by parts, so it is questionable how appropriate it is to apply that method here. Also note that applying the above filters to *both* past time levels at each evolution step is essential for long-term stability. We do not know why this is so important; naively one would think that only applying the filter to time level t^n would be sufficient, as the update step does not alter the variables at t^n , and since t^n is copied to t^{n-1} after each update step, both t^n and t^{n-1} are effectively smoothed. Also, a simple extension of the analysis in appendix B to account for different amounts of dissipation applied to each of the past time levels shows that the one-dimensional, flat space wave equation remains stable; hence the need to dissipate both time levels is particular to black-hole spacetimes as far as we can tell.

4.2. Coordinate system and boundary conditions

To simplify the imposition of asymptotically flat boundary conditions we use the following spatially compactified coordinate system. First, consider an uncompactified Cartesian coordinate system of the form

$$ds^2 = \bar{g}_{tt} dt^2 + 2\bar{g}_{ti} d\bar{x}^i dt + \bar{g}_{ij} d\bar{x}^i d\bar{x}^j. \quad (32)$$

Here $(\bar{x}^1, \bar{x}^2, \bar{x}^3) \equiv (\bar{x}, \bar{y}, \bar{z})$ runs from $-\infty$ to $+\infty$, and in the limit where $\bar{x}^i \rightarrow \pm\infty$ the metric becomes the Minkowski metric

$$ds^2 = -dt^2 + d\bar{x}^2 + d\bar{y}^2 + d\bar{z}^2. \quad (33)$$

The following coordinate transformation

$$\bar{x}^i = \tan(\pi x^i / 2) \quad (34)$$

(with $\bar{t} = t$) will bring (32) into the form

$$ds^2 = g_{tt} dt^2 + 2g_{ti} dx^i dt + g_{ij} dx^i dx^j, \quad (35)$$

where

$$\begin{aligned} g_{ti} &= \frac{\pi}{2} \sec^2(\pi x^i/2) \bar{g}_{ti}, \\ g_{ij} &= \frac{\pi^2}{4} \sec^2(\pi x^i/2) \sec^2(\pi x^j/2) \bar{g}_{ij}. \end{aligned} \quad (36)$$

Now x^i runs from -1 to 1 , and spacelike infinity i^0 corresponds to the surfaces $x^i = \pm 1$. Note that in this limit the compactified (unbarred) metric elements are singular; however the uncompactified parts are still well behaved and asymptote to their Minkowski values. In the code we thus evolve the uncompactified components g_{tt} , \bar{g}_{ti} and \bar{g}_{ij} , analytically substituting the values (36) into (9) prior to discretization. Furthermore, in the compactified coordinate system (35) we *define* the spatial source functions H_i to take the form

$$H_i = \bar{H}_i - \pi \tan(\pi x^i/2) \quad (37)$$

and evolve only the regularized components \bar{H}_i (for note that in compactified Minkowski coordinates $H_i = \pi \tan(\pi x^i/2)$ from (7)). Therefore, the outer boundary conditions we impose on the regularized metric and source functions are

$$\begin{aligned} \bar{g}_{tt}(t, i^0) &= -1, & \bar{g}_{ti}(t, i^0) &= 0, \\ \bar{g}_{ii}(t, i^0) &= 1, & \bar{g}_{ij}(t, i^0) &= 0, & i \neq j \\ H_i(t, i^0) &= 0, & \bar{H}_i(t, i^0) &= 0, \end{aligned} \quad (38)$$

where the notation (t, i^0) refers to any one of the six boundaries $x = \pm 1$, $y = \pm 1$ and $z = \pm 1$.

We conclude this section by discussing several concerns about evolving the field equations in a coordinate system compactified to spatial infinity. It is beyond the scope of this paper to analyse these issues in more detail; however, we are currently investigating them. However, note that a similar compactification scheme was used to model black strings in five dimensions [38], with no notable adverse effects.

First, the metric, hence equations, are formally singular at i^0 . The singular behaviour is dealt with using regularized variables and enforcing Minkowski space boundary conditions at i^0 , as described above. Nevertheless, there are terms in the equations that grow as $1/h^4$ at grid locations near the outer boundary, where h is the mesh spacing there. Therefore, for the equations to remain regular near i^0 during evolution requires that the leading order behaviour of the metric and scalar field variables always approach asymptotic values sufficiently fast to cancel this divergent behaviour (this is essentially the same problem one must deal with in an axisymmetric code near the axis singularity). For the simple test results presented in section 5 the evolution near i^0 is well behaved; however, we cannot guarantee that this will be the case for all classes of asymptotically flat initial data.

A second issue is the propagation of outgoing waves towards i^0 . The compactification causes the wavelengths and speeds to decrease. Thus, for any fixed resolution near i^0 , such waves will eventually be poorly resolved on the grid⁶. This could lead to a couple of undesirable effects. First, numerical dissipation will significantly decrease the amplitude of the waves, making waveform extraction in the outer regions of the domain impractical. Second, some portion of the wave will get ‘reflected’ back to the interior of the domain, which is not physical and may adversely affect the accuracy of the interior solution.

⁶ Keeping the waves well resolved with AMR is not a practical solution in general, as the outgoing wavetrain one expects from a binary inspiral, for example, is volume filling.

4.3. Apparent horizon finder and excision

We use the following *flow method* to search for single, simply-connected apparent horizons in the spacetime (this is the same algorithm used in [38]; see [39] for a review of most current methods, and [40, 41] for some recent work on fast, elliptic-solver-based apparent horizon finders). Consider the level set function $F(r, \theta, \phi)$ defined by

$$F(r, \theta, \phi) = r - R(\theta, \phi), \quad (39)$$

where the spherical polar coordinates (r, θ, ϕ) are defined in terms of *uncompactified* coordinates, relative to some centre $(\bar{x}_0, \bar{y}_0, \bar{z}_0)$, via

$$\bar{x} = \bar{x}_0 + r \cos \phi \sin \theta \quad \bar{y} = \bar{y}_0 + r \sin \phi \sin \theta \quad \bar{z} = \bar{z}_0 + r \cos \theta. \quad (40)$$

We want to find the function $R(\theta, \phi)$ such that the hypersurface $F = 0$ has zero outward null expansion Θ

$$\Theta = \ell_{\alpha;\beta} h^{\alpha\beta}, \quad (41)$$

where $h^{\alpha\beta}$ is the spatial metric (17) and ℓ^α is the outward pointing null vector normal to $F = \text{const}$ surfaces:

$$\ell_\alpha = n_\alpha + \frac{h^\beta{}_\alpha \partial_\beta F}{\sqrt{h^{\delta\gamma} \partial_\delta F \partial_\gamma F}}. \quad (42)$$

The flow method involves specifying some initial guess for R , then evolving the following equation until the magnitude of the norm of Θ evaluated along $F = 0$ is as close to zero as desired:

$$\frac{dR(\theta, \phi)}{d\lambda} = -\Theta(r, \theta, \phi)|_{r=R}, \quad (43)$$

where Θ is evaluated along $F = 0$. This equation is parabolic in ‘time’ λ , hence $d\lambda$ must be of order $(\Delta x^i)^2$ for stability.

During a typical evolution where a black hole forms via scalar-field collapse we initialize $R(\theta, \phi) = r_0$, where r_0 is a constant close to though outside⁷ of where we expect the apparent horizon (AH) to first form, and periodically (every tens to hundreds of time steps) search for an AH using this initial guess until one is found. For subsequent AH searches we use the previously found surface as an initial guess for $R(\theta, \phi)$. If multiple black holes form we search for each AH independently.

Some form of excision is necessary for long-term evolution of spacetimes containing black holes. Excision means that one places interior boundaries inside all black holes such that all physical singularities are removed from the computational domain. This assumes that cosmic censorship holds, which further implies that a black hole’s event horizon will be *outside* any apparent horizon, and hence one can use the apparent horizon as a guide to where to excise. For each black hole, we excise along an ellipsoid in *compactified* coordinate space, where the shape of the ellipsoid is chosen to match that of the apparent horizon as closely as possible along the ellipsoid’s principal axis (which currently are required to lie along the coordinate axis). The size of the ellipsoid is typically a bit smaller than that of the AH, to give some buffer zone between the excision surface and the AH. Any point on the grid *inside* the ellipsoid is defined to be excised, hence the excised region will necessarily be a grid-based approximation to the smooth ellipsoidal shape (this is often referred to as ‘lego excision’ in the literature).

⁷ The underlying assumption in (43) is that $\Theta > 0$ implies that the surface is outside the apparent horizon, which is *not* true everywhere at early times during a gravitational collapse simulation.

Table 2. A sample of the finite difference stencils used to convert the differential equations into difference equations adjacent to an excision surface. The column on the right shows the second order accurate representation (with y and z indices suppressed for clarity) of the corresponding derivative operator to the left, evaluated at the point (t^n, x_i, y_j, z_k) . The operators shown above are used when the point x_{i-1} is inside the excision surface and the points x_i, x_{i+1}, \dots are outside of it.

$$\begin{array}{l} \underline{f_{,x}} \quad (-3f_i^n + 4f_{i+1}^n - f_{i+2}^n) / (2\Delta x) \\ \underline{f_{,xx}} \quad (2f_i^n - 5f_{i+1}^n + 4f_{i+2}^n - f_{i+3}^n) / (\Delta x)^2 \end{array}$$

In general, boundary conditions need to be applied along the excision surface; however, in a free evolution (such as described here) where all the characteristics on the excision surface are directed inward, no boundary conditions should be placed on the field variables. In the current version of the code we *assume* that this is true, though we do not explicitly compute any of the characteristics. For a finite-difference scheme, such a ‘no boundary’ boundary condition means that the evolution equations are applied at the excision surface, with centred difference operators replaced, as appropriate, by forward or backward difference operators so as not to reference grid values inside the excised region. See table 2 for samples of the particular stencils we use. Note that we define the excision surface to be constant in time, and hence only spatial difference stencils need to be modified. During evolution, if the excision surface moves such that previously excised points (interior points) become ‘unexcised’, we initialize them via fourth-order extrapolation from adjacent exterior points at *all* time levels in the grid hierarchy. We cannot *a priori* prove that this excision method is stable, rather, as discussed in section 2, we will require convergence to a self-consistent solution of the field equations as a proof-by-example that the code is stable and correct.

4.4. Matter source

The present matter source modelled in the code is a massless scalar field Φ . The corresponding stress-energy tensor $T_{\mu\nu}$ is given by

$$T_{\mu\nu} = 2\Phi_{,\mu}\Phi_{,\nu} - g_{\mu\nu}\Phi_{,\gamma}\Phi^{,\gamma}, \quad (44)$$

and the evolution of Φ is governed by the wave equation

$$\square\Phi \equiv \Phi_{;\mu}{}^{\mu} = 0. \quad (45)$$

Note that (44) differs by a factor of 2 from the convention of Hawking and Ellis [55], which amounts to rescaling Φ by a factor of $\sqrt{2}$.

4.5. Scalar field initial data

At this stage, for scalar field gravitational collapse, we only consider time-symmetric initial data with a conformally flat spatial metric. Specifically, at $t = 0$, the metric and its first time derivatives take the following form:

$$\begin{aligned} \bar{g}_{tt}(t = 0, x, y, z) &= -1 \\ \bar{g}_{ti}(t = 0, x, y, z) &= 0 \\ \bar{g}_{ij}(t = 0, x, y, z) &= 0, \quad i \neq j \\ \bar{g}_{ij}(t = 0, x, y, z) &= \Psi^4(x, y, z), \quad i = j \\ \partial_t \bar{g}_{\alpha\beta}(t = 0, x, y, z) &= 0. \end{aligned} \quad (46)$$

The scalar field is thus the source of all non-trivial geometry at $t = 0$, and we initialize it as a sum of Gaussian-like functions of the following form:

$$\Phi(t = 0, x, y, z) = \sum_i f^i(x, y, z), \quad \partial_t \Phi(t = 0, x, y, z) = 0, \quad (47)$$

with

$$\begin{aligned} f^i(x, y, z) &= A^i \exp(-[\rho^i(x, y, z)/\Delta^i]^2), \\ \rho^i(x, y, z) &= \left\{ [1 - \epsilon_x^{i2}] [\bar{x}(x) - \bar{x}_0^i]^2 + [1 - \epsilon_y^{i2}] [\bar{y}(y) - \bar{y}_0^i]^2 + [1 - \epsilon_z^{i2}] [\bar{z}(z) - \bar{z}_0^i]^2 \right\}^{1/2}, \end{aligned} \quad (48)$$

where A_i , Δ^i , ϵ_x^i , ϵ_y^i , ϵ_z^i , \bar{x}_0^i , \bar{y}_0^i and \bar{z}_0^i are constants, and $\bar{x}(x)$, $\bar{y}(y)$ and $\bar{z}(z)$ are given by (34).

In (46), $\Psi(x, y, z)$ is solved for using the Hamiltonian constraint (13) equation, using an adaptive multigrid routine as discussed in [51, 52]. Note however that some complications do arise when attempting to solve an elliptic equation in compactified coordinates using multigrid; we will briefly discuss these issues in section 4.5.1. The momentum constraints (14) are trivially satisfied with the above initial conditions. Once the constraints have been solved, we initialize the source functions $\bar{H}_\alpha(t = 0, x, y, z)$ using (37) and (7).

With a three time level evolution scheme, the past time level at $t = -\Delta t$ needs to be initialized as well. To obtain second-order accurate convergence of the solution at late times, the past time level needs to be consistent with the initial data to within Δt^2 . In the code we have implemented a couple of methods to achieve this; the first is to use a Taylor expansion along with the equations of motion, the second is to evolve backward in time to $t = -\Delta t$ with a smaller time step. The first method works as follows [56]. For any one of the evolved grid functions f_{ijk}^n the past time level $n = -1$ is initialized to second-order accuracy using a Taylor expansion about $t = 0$

$$f_{ijk}^{-1} = f_{ijk}^0 - f_{ijk}'^0 \Delta t + f_{ijk}''^0 \frac{\Delta t^2}{2}, \quad (49)$$

where $f_{ijk}'^0$ is the first time derivative of $f(t, x, y, z)$ at $t = 0$ (from the initial data), and $f_{ijk}''^0$ is the second time derivative of $f(t, x, y, z)$ at $t = 0$, evaluated by substituting the initial data into the relevant equations of motion (9, 45) and solving for $\partial_t \partial_t f$. For the second method, the past time level is only initialized to first order using f_{ijk}^0 and $f_{ijk}'^0$, however with a smaller time step $\Delta t_s \approx \Delta t^2$. These initial data are then evolved backward in time until $t = -\Delta t$, and the solution obtained there is used to initialize the past time level for the actual evolution.

4.5.1. Multigrid in a compactified coordinate system. Standard geometric multigrid (MG) methods that use pointwise relaxation as a smoother (as we do) are only efficient when the size of the coefficient functions multiplying each of the principal parts of the elliptic operator are of comparable size [57]. This is not the case near i^0 in our compactified coordinate system. To illustrate, consider the form of the spatial Laplacian ∇^2 using the coordinates of section 4.2 in flatspace

$$\nabla^2 = \frac{4}{\pi^2} \left(\cos^4(\pi x/2) \frac{\partial^2}{\partial x^2} + \cos^4(\pi y/2) \frac{\partial^2}{\partial y^2} + \cos^4(\pi z/2) \frac{\partial^2}{\partial z^2} \right) + \dots, \quad (50)$$

where the \dots denote lower-order terms. Note that near any one of the outer boundaries $x^i = \pm 1$ the corresponding coefficient of the second derivative term goes to zero. We have not solved the issue of multigrid inefficiency in this part of the domain; however, if we use scalar field initial data of compact support in a sufficiently small region about $(x, y, z) = (0, 0, 0)$, then we find that the fine-grid relaxation performed within MG is adequate in obtaining a

solution of sufficient accuracy near i^0 (for Ψ will then go as $1 + O(1/r)$, and the initial guess of $\Psi = 1$ is close enough to the solution that relatively few relaxation sweeps are needed).

A more serious problem is that relaxation using standard centred difference approximations for first derivatives is unstable near i^0 . A partial solution to this problem is to use the following 4-way corner-averaged difference operator at the grid point (i, j, k) (shown here for the x derivative; the other first difference operators are similarly modified)

$$\begin{aligned} f_{,x} = & (f_{i+1,j+1,k+1} - f_{i-1,j+1,k+1})/(8\Delta x) + (f_{i+1,j+1,k-1} - f_{i-1,j+1,k-1})/(8\Delta x) \\ & + (f_{i+1,j-1,k+1} - f_{i-1,j-1,k+1})/(8\Delta x) \\ & + (f_{i+1,j-1,k-1} - f_{i-1,j-1,k-1})/(8\Delta x) + O(\Delta x^2). \end{aligned} \quad (51)$$

In the limit where the mesh spacing goes to zero in the vicinity of $x^i = \pm 1$, even this modification exhibits relaxation instabilities for initial data that are not sufficiently compact about $r = 0$. However, this is not a problem for the kinds of physical system we plan to use the code for, as all the interesting dynamics will be confined to a small region about $r = 0$, and this will be part of the hierarchy with high resolution.

4.6. Exact Schwarzschild black-hole initial data

For some of the tests described here we use analytic initial data from a Schwarzschild black-hole solution in Painlevé–Gullstrand-like (PG) coordinates. The non-compactified components of the metric are

$$ds^2 = - \left(1 - \frac{2M}{\bar{r}} \right) dt^2 + d\bar{x}^2 + d\bar{y}^2 + d\bar{z}^2 + \frac{2\sqrt{2M}}{\bar{r}^{3/2}} [\bar{x} d\bar{x} + \bar{y} d\bar{y} + \bar{z} d\bar{z}] dt, \quad (52)$$

where $\bar{r} \equiv \sqrt{\bar{x}^2 + \bar{y}^2 + \bar{z}^2}$ and M is the mass of the black hole. This (with appropriate spatial compactification) gives initial data for the metric at $t = 0$, and is also used to evaluate (7) for the initial values of the source functions.

4.7. Efficient simulation of axisymmetric spacetimes

In this section a variant of the ‘symmetry without symmetry’, or *cartoon* method [32] for efficient evolution of an axisymmetric spacetime with a Cartesian-based three-dimensional code is described. The advantages to the approach presented here are that *no* interpolation is ever performed, the axisymmetric grid structure is a two-dimensional slice of the Cartesian grid, rather than a thin three-dimensional slab, and the method is not specific to finite difference based codes, so can readily be applied to a spectral code, for instance. Having the grid structure be two-dimensional is helpful in that it allows easy integration of the code with standard adaptive mesh refinement (AMR) packages. The reason is as follows. In the original cartoon algorithm, the third, thin dimension is one finite-difference stencil-width thick. However, most AMR algorithms can only refine a given *volume* of a grid, which would increase the width of the slab-dimension on finer levels, and thereby reduce the efficiency of the cartoon method. Of course the AMR algorithm could be modified to deal with such a situation; however by using a two-dimensional grid structure one avoids this problem altogether. Note also that the purpose of the algorithm presented here is merely to provide an efficient way to simulate axisymmetric spacetimes with a Cartesian code, and *not* to address any issues of axis stability in axisymmetric codes, which was one of the original motivations behind the cartoon. Some recent work [34] has suggested that the standard cartoon algorithm may not be stable. Here we deal with the axis by applying appropriate regularity conditions and numerical dissipation, which has proven to be an effective method for dealing with the axial singularity

in axisymmetric codes [35, 36] (note also that in some cases stability in axisymmetric codes can be obtained by constructing schemes with a conserved discrete energy, using operators that satisfy summation by parts—see for example [37]).

The idea behind our modified cartoon algorithm is as follows. In a d -dimensional axisymmetric spacetime we have an azimuthal Killing vector ξ^μ , hence the metric $g_{\mu\nu}$ and the scalar-field matter source Φ satisfy

$$\mathcal{L}_\xi g_{\mu\nu} = 0, \quad \mathcal{L}_\xi \Phi = 0. \quad (53)$$

What these equations imply is that all non-trivial structure of the metric and scalar field are encoded within a $(d - 1)$ -dimensional sub-manifold \mathcal{S} of the spacetime, as long as ξ^μ is nowhere tangent to \mathcal{S} . Therefore, one only needs to solve the field equations on \mathcal{S} , and (53) can be used to extend the solution throughout the spacetime. In a numerical evolution, it makes most sense to have \mathcal{S} coincide with a constant coordinate hypersurface, which we set to $\bar{z} = 0$ for concreteness. We then choose coordinates such that ξ^μ has the explicit form (in uncompactified coordinates),

$$\xi^\mu = \bar{y} \left(\frac{\partial}{\partial \bar{z}} \right)^\mu - \bar{z} \left(\frac{\partial}{\partial \bar{y}} \right)^\mu, \quad (54)$$

which implies that ξ^μ is orthogonal to $\bar{z} = 0$, and the axis of symmetry runs along the \bar{x} direction and is centred at $\bar{y} = 0$.⁸ To solve the field equations on $\bar{z} = 0$ requires first and second derivatives of metric variables both within the hypersurface $\bar{z} = 0$, and orthogonal to it in the \bar{z} direction. To calculate \bar{z} derivatives the original cartoon method effectively extends the solution using (53) to a sufficient number of grid points above and below $\bar{z} = 0$, so that the usual finite-difference stencils can be used to calculate \bar{z} derivatives. The approach taken here is to substitute the explicit form of the Killing vector (54) into the definition (53), and use the resulting expression to evaluate the \bar{z} gradients directly. In other words, the same numerical method is used to solve equations as outlined in section 4.1; however instead of calculating \bar{z} derivatives using finite-difference approximation, the \bar{z} derivatives are replaced with appropriate combinations of \bar{x} and \bar{y} gradients. In appendix A we list the results of this calculation for all relevant gradients of the metric in compactified coordinates; here we illustrate the technique for the simpler case of the scalar field Φ .

Evaluating (53) for Φ using (54), we obtain

$$\frac{\partial \Phi}{\partial \bar{z}} = \frac{\bar{z}}{\bar{y}} \frac{\partial \Phi}{\partial \bar{y}}. \quad (55)$$

Taking the \bar{z} derivative of this equation, and replacing any \bar{z} gradients of Φ appearing on the right-hand side with (55), gives

$$\frac{\partial^2 \Phi}{\partial \bar{z}^2} = \frac{1}{\bar{y}} \frac{\partial \Phi}{\partial \bar{y}} + \frac{\bar{z}^2}{\bar{y}^2} \left(-\frac{1}{\bar{y}} \frac{\partial \Phi}{\partial \bar{y}} + \frac{\partial^2 \Phi}{\partial \bar{y}^2} \right). \quad (56)$$

Evaluating these equations at $\bar{z} = 0$ gives

$$\left. \frac{\partial \Phi}{\partial \bar{z}} \right|_{\bar{z}=0} = 0, \quad \left. \frac{\partial^2 \Phi}{\partial \bar{z}^2} \right|_{\bar{z}=0} = \frac{1}{\bar{y}} \frac{\partial \Phi}{\partial \bar{y}}. \quad (57)$$

All other mixed second derivatives of Φ involving \bar{z} are zero.

One thing to note from equation (57) is that the axis $\bar{y} = 0$ is singular. Therefore a regularity condition needs to be applied there, which can easily be seen from (57) to be $\partial \Phi / \partial \bar{y} = 0$ at $\bar{y} = 0$.

⁸ In other words, (54) is merely the Cartesian form of $(\partial/\partial\phi)^\mu$, where ϕ is a standard azimuthal coordinate with the axis of symmetry coincident with the \bar{x} axis.

5. Preliminary results

In this section we present results from several test simulations demonstrating certain aspects of the code. Significantly more work needs to be done before the code may be able to produce new physical results; however the current simulations suggest that the generalized harmonic decomposition could be a viable alternative to the ADM decomposition for many problems of interest.

In section 5.1 we show a convergence test of scalar-field evolution in 3D, section 5.2 evolves a Schwarzschild black hole in Painlevé–Gullstrand coordinates, and section 5.3 demonstrates gravitational collapse of scalar-field initial data to a Schwarzschild black hole.

5.1. Convergence test

For a 3D convergence test we used the following initial conditions for Φ (47), which describes three prolate spheroids slightly offset from one another so that there is no spatial symmetry in the problem:

$$\begin{aligned}
 A^1 &= 0.034, & A^2 &= 0.033, & A^3 &= 0.033 \\
 \Delta^1 &= 0.1, & \Delta^2 &= 0.1, & \Delta^3 &= 0.1 \\
 (\bar{x}_0^1, \bar{y}_0^1, \bar{z}_0^1) &= (0.025, 0, 0), \\
 (\bar{x}_0^2, \bar{y}_0^2, \bar{z}_0^2) &= (0, -0.025, -0.025), \\
 (\bar{x}_0^3, \bar{y}_0^3, \bar{z}_0^3) &= (-0.025, 0.025, 0.025) \\
 \epsilon_x^1 &= 0.1, & \epsilon_y^2 &= 0.1, & \epsilon_z^3 &= 0.1
 \end{aligned} \tag{58}$$

and all other initial data parameters for Φ are zero. With these parameters the ADM mass of the spacetime is roughly 0.005, so the initial distribution of energy is concentrated in a radius about ten times larger than its effective Schwarzschild radius. For the H_i coordinate condition we used a slightly modified version of (26), where we eliminated the coupling (through the normal n^μ) to H_i and added an arbitrary power n of α in the denominator:

$$\square H_i = -\kappa_i(t) \frac{\alpha - \alpha_0}{\alpha^n} + \xi_i(t) H_{i,\mu} n^\mu, \tag{59}$$

where $\kappa_i(t) = \kappa_0 q(t)$, $\xi_i(t) = \xi_0 q(t)$ and $q(t)$ is given by

$$\begin{aligned}
 q(t) &= \left(\frac{t}{t_1}\right)^3 \left[6 \left(\frac{t}{t_1}\right)^2 - 15 \frac{t}{t_1} + 10 \right], & 0 \leq t \leq t_1 \\
 &= 0, & \text{elsewhere.}
 \end{aligned} \tag{60}$$

$q(t)$ provides a smooth (twice differentiable) transition from 0 at $t = 0$ to 1 at $t = t_1$, and makes the evolution of the source functions consistent with the choice of time-symmetric initial data. We evolve \bar{H}_i to zero using a version of (16) with H_i replaced by \bar{H}_i , and $\kappa_i(t) = \kappa_0 q(t)$. For this particular simulation we had $\kappa_0 = 50$, $\xi_0 = 10$, $n = 3$, $t_1 = 1/10$ and $\alpha_0 = 1$.

A convergence test involves running a given simulation at several different resolutions, and comparing the results to ensure that the solution of the finite-difference equations is converging to a solution of the partial differential equations. We ran three simulations of differing resolution. The coarsest resolution run had a base grid size of 17^3 , and we specified a value for the maximum desired truncation error so that up to five additional levels of 2 : 1 refinement were used, giving an *effective* finest resolution of 513^3 —see figure 2 for a depiction of the mesh structure at two times during the simulation. We used a Courant factor of 0.25

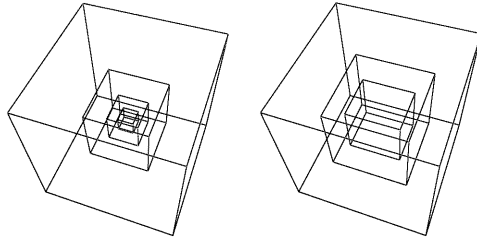


Figure 2. A depiction of the adaptive mesh structure for the convergence test simulation described in section 5.1. The image to the left corresponds to the mesh structure at $t = 0$, while that to the right at $t = 0.5$. The largest box in each figure, whose faces are at i^0 , actually represents two levels of (2:1) refinement. The increase in the size of the finer levels and loss of the finest level of refinement by $t = 0.5$ is due to the outward propagation of the initial distributions of energy.

at each level in the hierarchy (i.e. $\Delta t = 0.25\Delta x^i$). For the medium and finest resolution simulations we used the same grid hierarchy produced by the coarsest resolution simulation (which was produced using standard truncation error estimate methods), though doubled and quadrupled the resolution of all the grids respectively, keeping the same Courant factor. To keep the computational cost of the highest resolution run manageable, we only ran the simulation until $t = 0.5$; however this corresponds to roughly five light-crossing times of the central region of the grid where the scalar field is concentrated, and so a reasonable amount of dynamics does occur. Also, this run time is sufficiently long that possible adverse effects from the AMR algorithm, such as from regridding or high-frequency ‘noise’ from parent–child refinement boundaries, can be captured by the convergence test.

Label some grid function f from the finest resolution simulation f_h , from the medium one f_{2h} and from the coarsest one f_{4h} . Then the convergence factor Q_f we calculate is

$$Q_f = \frac{1}{\ln 2} (\ln \|f_{4h} - f_{2h}\| - \ln \|f_{2h} - f_h\|), \quad (61)$$

where before the subtraction we interpolate the grid functions to a common uniform grid, and then compute the ℓ_2 norm of the differences. For an n th order accurate scheme one would expect Q_f to approach a value of n in the limit as the mesh spacing goes to zero. See figure 3 for the convergence factors from the above simulations for several representative functions. The plot shows that we do see convergence close to second order. At early times, the convergence factor is slightly worse than second order; we surmise that the reason for this is a small amount of unphysical, high-frequency solution components (‘noise’) present at parent–child mesh refinement boundaries at the initial time. This noise seems to come from the multigrid algorithm we use to solve for the initial data, where linear interpolation is used to prolong from the coarse to fine meshes. Linear interpolation introduces high-frequency components in the fine-grid solution, which is smoothed by relaxation; however relaxation is only applied at interior points. Presumably some form of explicit dissipation at parent–child boundaries, or higher-order interpolation could cure this problem, though we find that the dissipation we use during the subsequent evolution is also quite effective at reducing the magnitude of this noise. At late times, several grid functions seem to show anomalously high convergence factors. This seems to be due to the fact that the simulations (in particular the coarsest resolution one) are not yet that close to the convergent regime. To test this would require a higher-resolution simulation, which would be impractical because of our computer resource

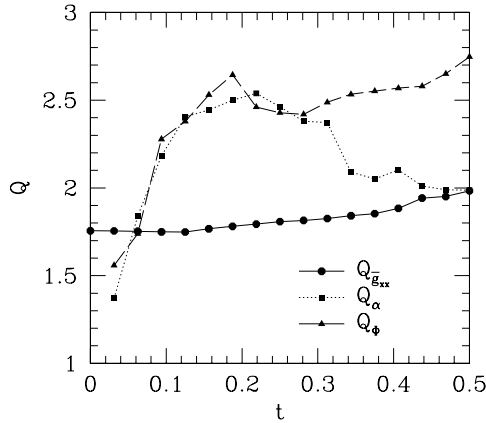


Figure 3. Convergence factors (61) for representative grid functions from the simulation described in section 5.1. The points denote the times when Q was calculated, and correspond to times when the entire grid hierarchy was in sync. Note that we only show $Q_{\bar{g}_{xx}}$ at $t = 0$, as all the other functions are exactly known then, and hence Q is ill defined. This plot shows that the solution is close to second-order convergent, with some caveats discussed in the text.

limitations⁹. However, by looking at an independent residual of the Einstein equations, as described next, we can already see the *trend* towards second-order convergence using only three simulations.

To check that we are solving the Einstein equations we compute an independent residual $\mathcal{R}_{\alpha\beta}$ of (1):

$$\mathcal{R}_{\alpha\beta} = R_{\alpha\beta} - 4\pi(2\pi T_{\alpha\beta} - g_{\alpha\beta}T). \quad (62)$$

After discretizing the ten residuals $\mathcal{R}_{\alpha\beta}$ using the finite-difference stencils described in the preceding section, we compute the residual grid function \mathcal{R} at each grid point as the infinity norm over the ten residuals. Note that we compute (62) *without* reference to the source functions, using only the *compactified* metric elements and scalar field. Since we know that \mathcal{R} should converge to zero in the limit, it is sufficient to compute its convergence factor using two resolutions, for example

$$Q_{\mathcal{R}} = \frac{1}{\ln 2}(\ln\|\mathcal{R}_{2h}\| - \ln\|\mathcal{R}_h\|). \quad (63)$$

Figure 4 shows $Q_{\mathcal{R}}$ computed using both $[\mathcal{R}_{2h}, \mathcal{R}_h]$ and $[\mathcal{R}_{4h}, \mathcal{R}_{2h}]$. This plot shows that we are tending towards second-order convergence as the resolution is increased.

5.2. Schwarzschild black-hole evolution

In this section we briefly show how well the current code can evolve a Schwarzschild black hole in Painlevé–Gullstrand coordinates. The analytic solution is used for initial conditions as described in section 4.6, with $M = 0.05$, and using (15) to keep the source functions frozen in during evolution. A (‘lego’) spherical excision region of radius $1.2M$ was used. We ran three axisymmetric simulations, each with identical grid hierarchy, though successively higher resolution as described in the previous section. The lowest-resolution simulation had a base

⁹ Alternatively, we can choose initial data that are better resolved on the coarsest grid; however, the kind of resolution we have here is more representative of the resolution we will be able to achieve in the near future with the computer power we have access to, and so we think this is a fair test of the code.

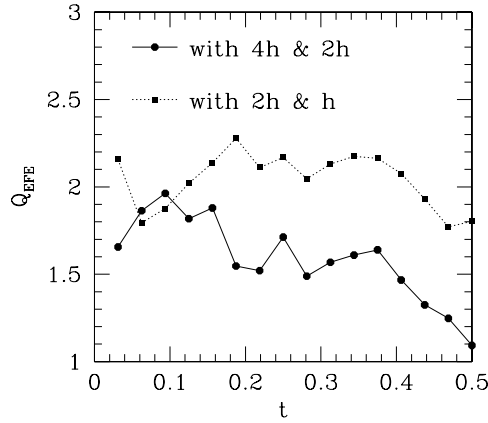


Figure 4. Convergence factor of the independent residual (62, 63) of the Einstein equations from the simulation described in section 5.1. The points denote the times when $Q_{\mathcal{R}}$ was calculated, and correspond to times when the entire grid hierarchy is in sync *after* an evolution time step (hence there are no points at $t = 0$). This plot shows we are tending towards a solution that is second-order convergent.

grid of 33×17 (spanning $-1 \dots 1$ in x and $0 \dots 1$ in y), using six additional levels of 2:1 refinement, and a Courant factor of 0.125 (the mesh structure is very similar to that depicted in figure 2, however in this example the refinement is constant in time). To compare, we also ran the two lowest-resolution simulations of the equivalent problem in full 3D; lack of computational resources prevented us from running the highest-resolution simulation in 3D and, for the same reason, we were not able to run the medium resolution simulation as long as the axisymmetric one¹⁰.

As a measure of the accuracy of the simulation, we calculate the mass M of the black hole from the area A of the apparent horizon:

$$M = \sqrt{\frac{A}{16\pi}}. \quad (64)$$

The mass for the five simulations is shown in figure 5. Note that we have not calculated any errors associated with the numerical integration of the apparent horizon area; we used the same resolution sphere (33 points in θ , ranging from 0 to π) in all cases, hence the error in the area calculation will be roughly the same for each run. There are a couple of significant things to note from this figure. First, even though we were not able to fully compare the axisymmetric results with a 3D code, what the partial comparison *suggests* is that explicitly enforcing axisymmetry in this case does *not* have a significant effect on the accuracy or runtime of the simulation. Second, even though the length of time that we can simulate a black hole to within a given accuracy with this code is not too long compared to the state of the art these days, the trend in increased runtime with resolution is promising. In particular, there is not much evidence of exponential growth of error early on (though once the error has grown to a certain magnitude, the code crashes quickly); rather, these plots *suggest* that the leading-order

¹⁰ Specifically, the medium resolution ($2h$) simulation in 3D took 160 h of runtime on 128 nodes of the Westgrid Xeon cluster to reach $t = 55M$, using about 120 MB of memory on each node. By comparison, the highest-resolution (h) axisymmetric simulation took approximately 240 h on 24 nodes of UBC's vn4 Xeon cluster to reach $t = 220M$. In 3D (2D), doubling the resolution typically requires 16 (8) times the runtime, and 8 (4) times the memory to evolve to a given physical time in the simulation.

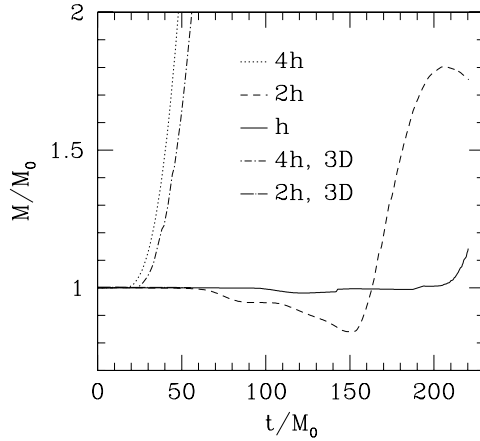


Figure 5. Normalized mass (64) for the evolution of a vacuum $M_0 = 0.05$ Schwarzschild black hole in Painlevé–Gullstrand coordinates. The curve labelled $4h$ corresponds to the lowest-resolution axisymmetric simulation, while the $2h$ (h) curves are from axisymmetric simulations with twice (four times) the resolution. The curves $4h, 3D$ and $2h, 3D$ are from runs with identical resolution to the $4h$ and $2h$ axisymmetric simulations respectively, though the simulations were in full 3D. Note that the $2h, 3D$ simulation curve only extends till roughly $t/M_0 = 55$ (as we ran out of computer time then), and is effectively hidden behind the other curves as $M/M_0 \approx 1$ up till then.

truncation error term has somewhere between linear and quadratic dependence on time. To see this, let us compare the putative runtime of a simulation where the leading-order error grows exponentially with time, versus polynomial growth of the error. For the exponential case, assume the norm of the error $E(t)$ as a function of time takes the following form:

$$E(t) = Ch^2 e^{\lambda t}, \quad (65)$$

where C is some constant, λ is the continuum growth factor and h is the mesh spacing. In other words, this situation describes an exponential ‘constraint violating mode’ driven by truncation-error terms. Let us solve for the evolution time t_h , to reach a specified error $E(t) = E_0$ with mesh spacing h :

$$t_h = \frac{\ln E_0 - \ln C - 2 \ln h}{\lambda}. \quad (66)$$

Now consider the following quantity ζ computed using three simulations with differing resolutions:

$$\zeta \equiv \frac{t_h - t_{2h}}{t_{2h} - t_{4h}}. \quad (67)$$

Evaluating ζ for the case of exponential growth using (66) gives

$$\zeta_\lambda = 1. \quad (68)$$

Repeating the calculation for the case of polynomial error growth of the form

$$E(t) = Ch^2 t^p \quad (69)$$

gives

$$\zeta_p = \frac{2^{2/p} - 1}{1 - 2^{-2/p}}. \quad (70)$$

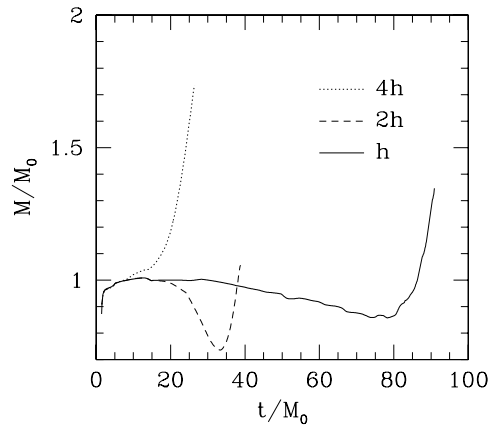


Figure 6. Normalized mass (64) from the axisymmetric evolution of a black hole formed via the gravitational collapse of a scalar field. The value of M_0 used was the largest, convergent value of M from the three simulations, which is a reasonable estimate of the final mass of the black hole. For comparison, the grid structure used for the simulations was identical to the corresponding axisymmetric simulations shown in figure 5.

For linear error growth, $\zeta_{p=1} = 4$, for quadratic growth $\zeta_{p=2} = 2$ and $\zeta_p \rightarrow 1$ in the limit as $p \rightarrow \infty$. If we evaluate ζ by defining the error to be that in M/M_0 from figure 5, using a value of 3% for E_0 we compute $\zeta \approx 2.7$, suggesting polynomial rather than exponential growth. However, this number changes as E_0 changes (for example, setting E_0 to 10% suggests faster than exponential growth), so we cannot conclusively rule out exponential growth. Regardless, from the practical point of view of using the current code to investigate black-hole physics in 3D, we need prohibitively high resolutions to get to a useful runtime range of several hundred M , so significantly more work needs to be done to improve the code for black-hole simulations.

5.3. Black-hole formation

The final test presented here is gravitational collapse of scalar-field initial data to a black hole, in axisymmetry. To compare with the vacuum black-hole simulation of the previous section, we used an identical grid structure, and chose initial data so that a black hole of roughly the same mass (0.05) forms. Specifically, we used a spherically symmetric Gaussian pulse (47) with

$$A^1 = 0.35, \quad \Delta^1 = 0.055, \quad (71)$$

with the rest of the initial data parameters for Φ set to zero. We used the same gauge conditions for H_μ as described in section 5.1 for the convergence test, except here the corresponding parameters were $\kappa_0 = 40$, $\xi_0 = 30$, $n = 5$, $t_1 = 1/80$ and $\alpha_0 = 1$. Note that the results are not very sensitive to this particular choice of gauge parameters; the rule of thumb is that κ_0 and ξ_0 of order $1/\Delta^1$, t_1 of order Δ^1 and n of order unity works reasonably well. Figure 6 shows the corresponding plot of apparent horizon mass versus time. The black hole forms after about $2M$ of evolution, after which some accretion of scalar field occurs, causing the mass to grow by a bit early on. Note also that once we detect an apparent horizon, we excise a spherical region 60% the size of the horizon, so approximately at $1.2M$, again for comparison with the previous evolution. At a given resolution this simulation (as judged by the mass estimate)

has less accuracy compared to the corresponding vacuum simulation; however the trend of increased accuracy with increased resolution is roughly the same.

6. Conclusion

We have described a new computational scheme for numerically solving the Einstein equations based upon generalized harmonic coordinates. This extends the earlier work of Garfinkle [20], and in some respects is similar to the direction pursued by Szilagyı and Winicour [21]. Some of the topics covered included suggestions for imposing dynamical gauge conditions, a new technique of implementing the cartoon method [32] for simulating axisymmetric spacetimes with a Cartesian code, a direct discretization scheme for second-order-in-space-and-time partial differential equations, and the use of a spatially compactified coordinate system. One attractive feature of harmonic evolution is that the principal part of the Einstein equations reduce to wave equations for each metric element. This, together with the use of a second-order discretization scheme, keeps the number of variables and constraint equations to a minimum, and the hope is that this will make it easier to achieve stable evolution. The use of a spatially compactified domain allows one to impose correct asymptotic boundary conditions for the simulation, and thus we automatically have constraint preserving boundary conditions. The advantage of our cartoon method over the original is that no interpolation is needed, and the simulation is performed on a 2D slice of the spacetime, thus simplifying the process of incorporating the code into an adaptive mesh refinement framework. Furthermore, the technique is not particular to finite-difference codes, and can be used with spectral methods, for instance.

Preliminary test simulations of black-hole spacetimes suggest that this scheme holds promise for being applicable to many problems of interest, including the binary black-hole problem, black-hole–matter interactions and critical gravitational collapse. However, a lot of research still needs to be done, at both the analytical and numerical levels, before this scheme may produce new physical results. In particular, it would be useful to analyse the mathematical well-posedness of the fully discrete system, including a variety of possible gauge evolution equations. The majority of techniques for analysing hyperbolic systems require reduction to first-order form (recently similar techniques have been developed for second order in space, first order in time systems [58–60]; also, in [61] the BSSN system is analysed by converting into first-order form; however the constraints introduced by this reduction are shown to obey a closed evolution system that is independent of the other constraints, implying that the original second-order system is well-posed). At the numerical level, a broader class of initial conditions needs to be explored, such as black-hole–matter interactions and black-hole collisions. This is of course one of the primary long-term goals of the code; however early tests indicate that a significant number of adjustments and improvements (to dissipation and extrapolation operators, for example) may be needed, in addition to more sophisticated gauge conditions than discussed here, before such scenarios could be simulated with sufficient accuracy and length of time for new results to be obtained.

Acknowledgments

I would like to thank Matthew Choptuik for many stimulating discussions about the work presented here. I gratefully acknowledge research support from NSF PHY-0099568, NSF PHY-0244906 and Caltech’s Richard Chase Tolman Fund. Simulations were performed on UBC’s **vn** cluster, (supported by CFI and BCKDF), and the **Westgrid** cluster (supported by CFI, ASRI and BCKDF).

Appendix A. Evolution of axisymmetric spacetimes

As described in section 4.7, we can efficiently simulate axisymmetric spacetimes along a single $z = 0$ slice of the spacetime by replacing all z gradients in the field and matter evolution equations with appropriate x and y gradients, as dictated by (53). Here we list the equations for gradients of the *regular* components of the metric $g_{\mu\nu}$ and scalar field Φ with respect to the *compactified* coordinates (section 4.2), and give the corresponding on-axis regularity conditions.

The first z derivatives are

$$\begin{aligned}
\partial_z \bar{g}_{tt}|_{z=0} &= 0, & \partial_z \bar{g}_{t\bar{x}}|_{z=0} &= 0 \\
\partial_z \bar{g}_{t\bar{y}}|_{z=0} &= -\bar{g}_{t\bar{z}}(\pi/2\bar{y}), & \partial_z \bar{g}_{t\bar{z}}|_{z=0} &= \bar{g}_{t\bar{y}}(\pi/2\bar{y}) \\
\partial_z \bar{g}_{\bar{x}\bar{x}}|_{z=0} &= 0, & \partial_z \bar{g}_{\bar{x}\bar{y}}|_{z=0} &= -\bar{g}_{\bar{x}\bar{z}}(\pi/2\bar{y}) \\
\partial_z \bar{g}_{\bar{x}\bar{z}}|_{z=0} &= \bar{g}_{\bar{x}\bar{y}}(\pi/2\bar{y}), & \partial_z \bar{g}_{\bar{y}\bar{y}}|_{z=0} &= -\bar{g}_{\bar{y}\bar{z}}(\pi/\bar{y}) \\
\partial_z \bar{g}_{\bar{y}\bar{z}}|_{z=0} &= (\bar{g}_{\bar{y}\bar{y}} - \bar{g}_{\bar{z}\bar{z}})(\pi/2\bar{y}), & \partial_z \bar{g}_{\bar{z}\bar{z}}|_{z=0} &= \bar{g}_{\bar{y}\bar{z}}(\pi/\bar{y}) \\
\partial_z \Phi|_{z=0} &= 0.
\end{aligned} \tag{A1}$$

Mixed z - t , z - x and z - y second derivatives are calculated by taking the appropriate derivative of (A1). Second derivatives with respect to z are computed as follows:

$$\partial_z \partial_z \bar{g}_{\alpha\beta}|_{z=0} = \frac{\pi}{2} \left(\frac{\partial_y \bar{g}_{\alpha\beta}}{\bar{y}(1+\bar{y}^2)} + \frac{\pi C_{\alpha\beta}}{2\bar{y}^2} \right), \quad \partial_z \partial_z \Phi|_{z=0} = \frac{\pi \partial_y \Phi}{2\bar{y}(1+\bar{y}^2)} \tag{A2}$$

where the coefficients $C_{\alpha\beta}$ are

$$\begin{aligned}
C_{tt} &= 0, & C_{t\bar{x}} &= 0 \\
C_{t\bar{y}} &= -\bar{g}_{t\bar{y}}, & C_{t\bar{z}} &= -\bar{g}_{t\bar{z}} \\
C_{\bar{x}\bar{x}} &= 0, & C_{\bar{x}\bar{y}} &= -\bar{g}_{\bar{x}\bar{y}} \\
C_{\bar{x}\bar{z}} &= -\bar{g}_{\bar{x}\bar{z}}, & C_{\bar{y}\bar{y}} &= 2(\bar{g}_{\bar{z}\bar{z}} - \bar{g}_{\bar{y}\bar{y}}) \\
C_{\bar{y}\bar{z}} &= -4\bar{g}_{\bar{y}\bar{z}}, & C_{\bar{z}\bar{z}} &= 2(\bar{g}_{\bar{y}\bar{y}} - \bar{g}_{\bar{z}\bar{z}}).
\end{aligned} \tag{A3}$$

The on-axis regularity conditions are

$$\begin{aligned}
\partial_y \bar{g}_{tt}|_{y=0} &= 0, & \partial_y \bar{g}_{t\bar{x}}|_{y=0} &= 0 \\
\bar{g}_{t\bar{y}}|_{y=0} &= 0, & \bar{g}_{t\bar{z}}|_{y=0} &= 0 \\
\partial_y \bar{g}_{\bar{x}\bar{x}}|_{y=0} &= 0, & \bar{g}_{\bar{x}\bar{y}}|_{y=0} &= 0 \\
\bar{g}_{\bar{x}\bar{z}}|_{y=0} &= 0, & \partial_y \bar{g}_{\bar{y}\bar{y}}|_{y=0} &= 0 \\
\partial_y \bar{g}_{\bar{y}\bar{z}}|_{y=0} &= 0, & \partial_y \bar{g}_{\bar{z}\bar{z}}|_{y=0} &= 0 \\
\partial_y \Phi|_{y=0} &= 0, & \bar{g}_{\bar{y}\bar{y}}|_{y=0} &= \bar{g}_{\bar{z}\bar{z}}|_{y=0}.
\end{aligned} \tag{A4}$$

To compute the z gradients and regularity conditions for the source functions in the code, we simply substitute the results from the calculation for the metric into the definition of the source functions (7), (37).

Appendix B. Stability analysis of a second order in space and time evolution scheme

Here we give a von Neumann-like stability analysis of the one-dimensional flat space wave equation using the discretization scheme described in section 4. Second order in time schemes

for the wave equation are not very common in the literature, so the example given here is to demonstrate that the method is inherently stable, ignoring the complications of boundaries, excision, non-constant coefficients and nonlinear lower-order terms of the full problem (the analysis of which is beyond the scope of this paper). Even though dissipation is not needed in this example, we add it as applied in the code to demonstrate how it works.

The model wave equation for $\Phi(x, t)$ is

$$\Phi_{,tt} - \Phi_{,xx} = 0. \quad (\text{B1})$$

Discretization of this equation using the stencils in table 1 gives

$$\Phi_j^{n+1} - 2\Phi_j^n + \Phi_j^{n-1} - \lambda^2(\Phi_{j+1}^n - 2\Phi_j^n + \Phi_{j-1}^n) = 0, \quad (\text{B2})$$

where $\Phi_j^n \equiv \Phi(x = j\Delta x, t = n\Delta t)$ and $\lambda \equiv \Delta t/\Delta x$ is the Courant factor. This immediately gives an explicit update scheme for the unknown Φ^{n+1} given information at two past time levels, Φ^n, Φ^{n-1} :

$$\Phi_j^{n+1} = 2\Phi_j^n - \Phi_j^{n-1} + \lambda^2(\Phi_{j+1}^n - 2\Phi_j^n + \Phi_{j-1}^n) \quad (\text{B3})$$

(note that the iterative relaxation method described in section 4 gives exactly the same update scheme in this case). It is mathematically simpler to analyse this equation using an equivalent two time level scheme by introducing the variable

$$\Psi_j^n \equiv \Phi_j^{n-1}, \quad (\text{B4})$$

after which (B3) becomes

$$\Phi_j^{n+1} = 2\Phi_j^n - \Psi_j^n + \lambda^2(\Phi_{j+1}^n - 2\Phi_j^n + \Phi_{j-1}^n), \quad \Psi_j^{n+1} = \Phi_j^n. \quad (\text{B5})$$

As (B5) is linear with constant coefficients, we can completely characterize its stability properties by analysing the evolution of individual Fourier modes of the form $c(t) e^{ikx}$. To this end, let

$$\Phi(x, t) \equiv a(t) e^{ikx} \quad (\text{B6})$$

$$\Psi(x, t) \equiv b(t) e^{ikx}. \quad (\text{B7})$$

Substituting this into (B5) gives

$$a^{n+1} = 2a^n - b^n - 4\lambda^2\xi^2 a^n, \quad b^{n+1} = a^n, \quad (\text{B8})$$

where

$$\xi \equiv \sin(k\Delta x/2), \quad (\text{B9})$$

and we have used the identity $-4\sin^2(k\Delta x/2) = e^{-ik\Delta x} - 2 + e^{ik\Delta x}$. Note that the smallest wavelength that can be represented on a numerical grid is $2\Delta x$ (the Nyquist limit), which corresponds to a largest possible wave number $k = \pi/\Delta x$, hence ξ ranges from 0 to 1.

As described in section 4.1.1, we apply numerical dissipation to all past time level variables, prior to the update step, by first calculating the high-frequency component of the function using (30), and then subtracting it from the function via (31). For a grid function $f_j^n = c^n e^{ikx_j}$, the high-frequency component η_j^n is defined as

$$\begin{aligned} \eta_j^n &= \frac{1}{16} (f_{j-2}^n - 4f_{j-1}^n + 6f_j^n - 4f_{j+1}^n + f_{j+2}^n) \\ &= f_j^n \xi^4, \end{aligned} \quad (\text{B10})$$

and filtering amounts to modifying f_j^n as follows:

$$\begin{aligned} f_j^n &\rightarrow f_j^n - \epsilon \eta_j^n \\ &= f_j^n (1 - \epsilon \xi^4) \\ &= \bar{\epsilon} f_j^n, \end{aligned} \quad (\text{B11})$$

where $\bar{\epsilon} \equiv 1 - \epsilon \xi^4$. As $\xi \in [0 \dots 1]$ and $\epsilon \in [0 \dots 1]$, $\bar{\epsilon} \in [0 \dots 1]$. With this form of dissipation (which is linear, and hence fits into the Fourier analysis of the evolution scheme) applied to both Φ_j^n and Ψ_j^n , (B8) becomes

$$a^{n+1} = \bar{\epsilon}[2a^n(1 - 2\lambda^2\xi^2) - b^n], \quad b^{n+1} = \bar{\epsilon}a^n. \quad (\text{B12})$$

In matrix form, the update step can be written as

$$\begin{bmatrix} a \\ b \end{bmatrix}^{n+1} = \mathbf{A} \begin{bmatrix} a \\ b \end{bmatrix}^n, \quad (\text{B13})$$

where

$$\mathbf{A} = \bar{\epsilon} \begin{bmatrix} 2(1 - 2\lambda^2\xi^2) & -1 \\ 1 & 0 \end{bmatrix}. \quad (\text{B14})$$

The numerical evolution will be stable if the eigenvalues Λ_{\pm} of \mathbf{A} all lie on or within the unit circle in the complex plain. A straightforward calculation gives

$$\Lambda_{\pm} = \bar{\epsilon}[1 - 2\xi^2\lambda^2 \pm i2\xi\lambda\sqrt{1 - \xi^2\lambda^2}]. \quad (\text{B15})$$

The expression within the square root of (B15) is strictly non-negative if we require that $\lambda \in [0 \dots 1]$. The magnitude of the eigenvalues are

$$\|\Lambda_{\pm}\| = \bar{\epsilon}. \quad (\text{B16})$$

Hence, for $\bar{\epsilon} \leq 1$ the numerical scheme is stable; in fact, without dissipation ($\bar{\epsilon} = 1$) the scheme is inherently stable and non-dissipative.

References

- [1] Reula O A 1998 Hyperbolic methods for Einstein's equations *Living Rev. Rel.* **1**:3
- [2] Lehner L 2001 Numerical relativity: a review *Class. Quantum Grav.* **18** R25
- [3] Nakamura T, Oohara K and Kojima Y 1987 *Prog. Theor. Phys. Suppl.* **90** 1
- [4] Shibata M and Nakamura T 1995 Evolution of three-dimensional gravitational waves: harmonic slicing case *Phys. Rev. D* **52** 5428
- [5] Baumgarte T W and Shapiro S L 1999 Numerical integration of Einstein's field equations *Phys. Rev. D* **59** 024007
- [6] Bruegmann B, Tichy W and Jansen N 2004 Numerical simulation of orbiting black holes *Phys. Rev. Lett.* **92** 211101
- [7] Bishop N T, Gomez R, Husa S, Lehner L and Winicour J 2003 A numerical relativistic model of a massive particle in orbit near a Schwarzschild black hole *Phys. Rev. D* **68** 084015
- [8] Anderson M and Matzner R A 2003 Extended lifetime in computational evolution of isolated black holes *Preprint gr-qc/0307055*
- [9] Bonazzola S, Gourgoulhon E, Grandclement P and Novak J 2004 A constrained scheme for Einstein equations based on Dirac gauge and spherical coordinates *Phys. Rev. D* **70** 104007
- [10] Holst M, Lindblom L, Owen R, Pfeiffer H P, Scheel M A and Kidder L E 2004 Optimal constraint projection for hyperbolic evolution systems *Phys. Rev. D* **70** 084017
- [11] Buchman L T and J M 2003 A hyperbolic tetrad formulation of the Einstein equations for numerical relativity *Phys. Rev. D* **67** 084017
- [12] Estabrook F B, Robinson R S and Wahlquist H D 1997 Hyperbolic equations for vacuum gravity using special orthonormal frames *Class. Quantum Grav.* **14** 1237
- [13] Buchman L T and Bardeen J M 2003 A hyperbolic tetrad formulation of the Einstein equations for numerical relativity *Phys. Rev. D* **67** 084017
- [14] Estabrook F B 2004 Mathematical structure of tetrad equations for numerical relativity *Preprint gr-qc/0411029*
- [15] Friedrich H 1981 On the regular and the asymptotic characteristic initial value problem for Einstein's vacuum field equations *Proc. R. Soc. A* **375** 169
- [16] Friedrich H 1981 The asymptotic characteristic initial value problem for Einstein's vacuum field equations as an initial value problem for a first-order quasilinear symmetric hyperbolic system *Proc. R. Soc. A* **378** 401

- [17] Hubner P 1999 A scheme to numerically evolve data for the conformal Einstein equation *Class. Quantum Grav.* **16** 2823
- [18] Husa S 2003 Numerical relativity with the conformal field equations *Lect. Notes Phys.* **617** 159
- [19] Bruhat Y 1962 The Cauchy problem *Gravitation: An Introduction to Current Research* ed L Witten (New York: Wiley)
- [20] Garfinkle D 2002 Harmonic coordinate method for simulating generic singularities *Phys. Rev. D* **65** 044029
- [21] Szilagyai B and Winicour J 2003 Well-posed initial-boundary evolution in general relativity *Phys. Rev. D* **68** 041501
- [22] Bona C, Ledvinka T, Palenzuela C and Zacek M 2003 General-covariant evolution formalism for numerical relativity *Phys. Rev. D* **67** 104005
- [23] Calabrese G 2004 Finite differencing second order systems describing black hole spacetimes *Preprint gr-qc/0410062*
- [24] Kreiss H and Ortiz O E 2002 Some mathematical and numerical questions connected with first and second order time dependent systems of partial differential equations *Lect. Notes Phys.* **604** 359
- [25] Kelly B, Laguna P, Lockitch K, Pullin J, Schnetter E, Shoemaker D and Tiglio M 2001 A cure for unstable numerical evolutions of single black holes: adjusting the standard ADM equations *Phys. Rev. D* **64** 084013
- [26] Shinkai H and Yoneda G 2002 Re-formulating the Einstein equations for stable numerical simulations: formulation problem in numerical relativity *Preprint gr-qc/0209111*
- [27] Tiglio M 2003 Dynamical control of the constraints growth in free evolutions of Einstein's equations *Preprint gr-qc/0304062*
- [28] Tiglio M, Lehner L and Neilsen D 2004 3D simulations of Einstein's equations: symmetric hyperbolicity, live gauges and dynamic control of the constraints *Phys. Rev. D* **70** 104018
- [29] Lindblom L, Scheel M A, Kidder L E, Pfeiffer H P, Shoemaker D and Teukolsky S A 2004 Controlling the growth of constraints in hyperbolic evolution systems *Phys. Rev. D* **69** 124025
- [30] Meier D L 2003 Constrained transport algorithms for numerical relativity. I. Development of a finite difference scheme *Astrophys. J.* **595** 980
- [31] Di Bartolo C, Gambini R and Pullin J 2004 Consistent and mimetic discretizations in general relativity *Preprint gr-qc/0404052*
- [32] Alcubierre M, Brandt S, Bruegmann B, Holz D, Seidel E, Takahashi R and Thornburg J 2001 Symmetry without symmetry: numerical simulation of axisymmetric systems using Cartesian grids *Int. J. Mod. Phys. D* **10** 273
- [33] Friedrich H 1985 On the hyperbolicity of Einstein's and other gauge field equations *Commun. Math. Phys.* **100** 525
- [34] Frauendiener J 2002 On discretizations of axisymmetric systems *Phys. Rev. D* **66** 104027
- [35] Garfinkle D and Duncan G C 2001 Numerical evolution of Brill waves *Phys. Rev. D* **63** 044011
- [36] Choptuik M W, Hirschmann E W, Liebling S L and Pretorius F 2003 An axisymmetric gravitational collapse code *Class. Quantum Grav.* **20** 1857
- [37] Calabrese G and Neilsen D 2004 Spherical excision for moving black holes and summation by parts for axisymmetric systems *Phys. Rev. D* **69** 044020
- [38] Choptuik M W, Lehner L, Olabarrieta I, Petryk R, Pretorius F and Villegas H 2003 Towards the final fate of an unstable black string *Phys. Rev. D* **68** 044001
- [39] Thornburg J 1996 Finding apparent horizons in numerical relativity *Phys. Rev. D* **54** 4899
- [40] Thornburg J 2004 A fast apparent-horizon finder for 3-dimensional Cartesian grids in numerical relativity *Class. Quantum Grav.* **21** 743
- [41] Schnetter E 2003 Finding apparent horizons and other two-surfaces of constant expansion *Class. Quantum Grav.* **20** 4719
- [42] Cook G B 2000 Initial data for numerical relativity *Living Rev. Rel.* **3** 5
- [43] Friedrich H 1996 Hyperbolic reductions for Einstein's equations *Class. Quantum Grav.* **13** 1451
- [44] York J W 1978 *Sources of Gravitational Radiation* ed L Smarr (Seattle, WA: Cambridge University Press)
- [45] Bona C, Masso J, Seidel E and Stela J 1995 New formalism for numerical relativity *Phys. Rev. Lett.* **75** 600
- [46] Balakrishna J, Daues G, Seidel E, Suen W, Tobias M and Wang E 1996 Coordinate conditions and their implementation in 3D numerical relativity *Class. Quantum Grav.* **13** L135
- [47] Alcubierre M, Bruegmann B, Pollney D, Seidel E and Takahashi R 2001 Black hole excision for dynamic black holes *Phys. Rev. D* **64** 061501
- [48] Lindblom L and Scheel M A 2003 Dynamical gauge conditions for the Einstein evolution equations *Phys. Rev. D* **67** 124005
- [49] Alcubierre M, Bruegmann B, Diener P, Koppitz M, Pollney D, Seidel E and Takahashi R 2003 Gauge conditions for long-term numerical black hole evolutions without excision *Phys. Rev. D* **67** 084023
- [50] Bona C 2004 Private communication

-
- [51] Pretorius F and Choptuik M W Adaptive mesh refinement for coupled elliptic-hyperbolic systems, in preparation
 - [52] Pretorius F 2002 Numerical simulations of gravitational collapse *PhD Thesis* University of British Columbia, Canada
 - [53] Kreiss H and Olinger J 1973 Methods for the approximate solution of time dependent problems *Global Atmospheric Research Programme (Publications Series No. 10)*
 - [54] Calabrese G, Lehner L, Reula O, Sarbach O and Tiglio M 2003 Summation by parts and dissipation for domains with excised regions *Preprint gr-qc/0308007*
 - [55] Hawking S W and Ellis G F R 1973 *The Large Scale Structure of Space-Time* (Cambridge: Cambridge University Press)
 - [56] Choptuik M W 2003 Private communication
 - [57] Brandt A 1977 Multi-level adaptive solutions to boundary-value problems *Math. Comput.* **31** 330
 - [58] Nagy G, Ortiz O E and Reula O A 2004 Strongly hyperbolic second order Einstein's evolution equations *Phys. Rev. D* **70** 044012
 - [59] Gundlach C and Martin-Garcia J M 2004 Symmetric hyperbolic form of systems of second-order evolution equations subject to constraints *Phys. Rev. D* **70** 044031
 - [60] Gundlach C and Martin-Garcia J M 2004 Symmetric hyperbolicity and consistent boundary conditions for second-order Einstein equations *Phys. Rev. D* **70** 044032
 - [61] Beyer H and Sarbach O 2004 On the well posedness of the Baumgarte–Shapiro–Shibata–Nakamura formulation of Einstein's field equations *Phys. Rev. D* **70** 104004