

## Solving the Einstein constraints in periodic spaces with a multigrid approach

This content has been downloaded from IOPscience. Please scroll down to see the full text.

2014 Class. Quantum Grav. 31 035004

(<http://iopscience.iop.org/0264-9381/31/3/035004>)

View [the table of contents for this issue](#), or go to the [journal homepage](#) for more

Download details:

IP Address: 194.94.224.254

This content was downloaded on 12/02/2014 at 11:09

Please note that [terms and conditions apply](#).

# Solving the Einstein constraints in periodic spaces with a multigrid approach

**Eloisa Bentivegna**

Max-Planck-Institut für Gravitationsphysik, Albert-Einstein-Institut,  
Am Mühlenberg 1, D-14476 Golm, Germany

E-mail: [eloisa.bentivegna@aei.mpg.de](mailto:eloisa.bentivegna@aei.mpg.de)

Received 28 August 2013, revised 21 November 2013

Accepted for publication 25 November 2013

Published 23 December 2013

## Abstract

Novel applications of Numerical Relativity demand more flexible algorithms and tools. In this paper, I develop and test a multigrid solver, based on the infrastructure provided by the Einstein Toolkit, for elliptic partial differential equations on spaces with periodic boundary conditions (PBCs). This type of boundary often characterizes the numerical representation of cosmological models, where space is assumed to be made up of identical copies of a single fiducial domain, so that only a finite volume (with PBCs at its edges) needs to be simulated. After a few tests and comparisons with existing codes, I use the solver to generate initial data for an infinite, periodic, cubic black-hole lattice.

Keywords: numerical relativity, initial data

PACS numbers: 02.60.Lj, 04.20.Ex, 04.25.dg, 98.80.Jk

(Some figures may appear in colour only in the online journal)

## 1. Introduction

In recent years, the techniques of Numerical Relativity have more and more frequently been used to explore domains outside of the usual compact-object applications (for a review, see e.g. the collection introduced in [1]). Numerical studies in classical, late-time cosmology, examples of which can be found throughout the history of Numerical Relativity [2–6], have also seen a surge both in demand [7–13] and in results [14, 15].

This increase in popularity motivates the need for new, more general tools for solving Einstein's equation in three (or more) dimensions. In particular, components such as the solvers of elliptic partial differential equations (PDEs) are typically fine-tuned to the geometry and symmetries of the physical system that one desires to study. The current freely available tools [16–18], designed to tackle asymptotically-flat spaces containing only a few compact

sources, are therefore unsuitable to solve the elliptic PDEs that constitute the constraint part of the Einstein system in a generic slice of a cosmological spacetime. The difference between compact-object and cosmological spaces is particularly acute under one aspect: on asymptotically-flat spaces, elliptic PDEs are usually solved with Dirichlet boundary conditions, whilst on cosmological spaces, one typically models a fiducial cell with periodic boundary conditions (PBCs). It will be shown below that this aspect has repercussions on the formulation of the problem and, consequently, on the numerical algorithms necessary to solve it.

Notice that the use of PBCs to describe cosmological spacetimes is by no means a definitive choice: studies on Newtonian simulations have shown how this type of boundary conditions can introduce artificial effects [19] and suppress the backreaction of inhomogeneities [20]. However, the few viable alternatives come with comparable drawbacks [21]; when one is interested in the simulation of infinite volumes with no macroscopic inhomogeneity, PBCs remain the most natural and robust option. Finding out whether similar considerations extend to General Relativity is by itself a compelling motivation to develop elliptic solvers that work with PBCs. Notice that, for general-relativistic evolutions, one also has the option to neglect boundary conditions altogether by using null boundaries and the zooming technique illustrated in [22]. However, this results in a hard limit on the time extension of the simulated domain; furthermore, this method does not apply to the initial-data generation, which still requires boundary conditions of some sort.

In this paper, I develop and test a solver based on the multigrid paradigm. The solver is built as a component of the open-source Einstein Toolkit [23], and is released as free software<sup>1</sup>. Due to the power and flexibility of this paradigm, which does not require special tuning to the individual properties of different problems, examples of multigrid solvers appear regularly in the Numerical-Relativity literature [24–26]. None of these works, however, mentions PBCs; an explicit treatment of this aspect constitutes the focus of this paper.

In the following, I will briefly discuss the properties of elliptic PDEs in conjunction with PBCs in section 2. I will then present an outline of multigrid techniques in section 3. In section 4, I will illustrate these concepts and test the implementation with a number of examples, including Poisson’s equation, Helmholtz’s equation and the system of constraints of General Relativity, in cases where a solution is known exactly or can be obtained via an independent code. In section 5, I will finally use the solver to generate a solution of the Einstein constraints representing an infinite, cubic black-hole lattice, which can be compared qualitatively to the results of [14]. The usual  $G = c = 1$  convention will be used throughout.

## 2. Elliptic problems on spaces without boundaries

Elliptic problems with PBCs can be underdetermined with respect to their Dirichlet-boundaries counterparts, as the former sort of boundary specification is typically weaker than the latter. From this standpoint, these problems are close to elliptic equations with Neumann boundary conditions. In this section, I will identify the degrees of freedom that are left undetermined by PBCs in different cases, and illustrate some ways in which they can be fixed. Whilst, in principle, any condition on the unconstrained modes is sufficient to close the system and lead to a unique solution, the details of how and when these conditions are applied in the solution algorithm are rather important from the practical standpoint, as the extra information must come at the right time so as not to disturb the convergence process.

<sup>1</sup> A copy can be obtained from the git repository at [git@bitbucket.org:eloisa/cosmology.git](https://git@bitbucket.org:eloisa/cosmology.git)

### 2.1. Well-posedness without Dirichlet boundaries

Let us examine the problem of well-posedness (thereby referring to the formulation of a problem which has one, and exactly one solution) in a few concrete cases. All the problems below will be solved on cubic domains with boundary conditions on the six faces.

Let us begin with the linear case. A simple example is Poisson's equation, which reads

$$\Delta f + d = 0. \quad (1)$$

First off, let us recall that, similarly to the Neumann BCs case [27, 28], PBCs add an integrability condition to this equation: since the integral of the laplacian operator of any sufficiently smooth function will vanish on a periodic domain, then the integral of  $d$  must also vanish over the same region, lest the equation be insoluble. Now, let us assume we have a solution  $\bar{f}$  of (1); obviously any  $f = \bar{f} + J$  is also a solution of the same equation, if  $J$  is a constant. The elliptic equation does not constrain this zero-frequency mode, nor do the PBCs. Somewhere in the course of the solution process, this information must be supplied by hand, for instance by rescaling  $f$  over the whole domain by prescribing a desired value  $f_{\mathcal{O}}$  at a particular point  $\mathcal{O}$ :

$$f \rightarrow f - f(\mathcal{O}) + f_{\mathcal{O}}. \quad (2)$$

This is equivalent to the choice of  $J = f_{\mathcal{O}} - \bar{f}(\mathcal{O})$ .

Another example of linear elliptic PDE is Helmholtz's equation, which reads

$$\Delta f + cf = 0 \quad (3)$$

where  $c$  is a constant. Here, it is the integral of  $f$  that needs to be zero in order for it to be a viable solution. Again, we notice that, once we find a solution  $\bar{f}$ , we can generate a family of other solutions by simply taking  $f = k\bar{f}$ , where  $k$  is a constant. In fact, given the periodicity, the phase transformation  $f(x) = \bar{f}(x + \delta)$  also generates a family of solutions. We can resolve these ambiguities by, for instance, imposing a definite parity for the solution (which fixes the phase  $\delta$ ) along with the value of  $f$  at a point, and rescaling the solution, effectively fixing  $k = f_{\mathcal{O}}/\bar{f}(\mathcal{O})$ .

Things begin to be non-trivial when both linear terms are included, in the inhomogeneous Helmholtz's equation

$$\Delta f + cf + d = 0. \quad (4)$$

Here, the integral condition reads

$$\int_{\mathcal{V}} (cf + d) dV = 0 \quad (5)$$

where  $\mathcal{V}$  denotes the periodic cell. Now, let us assume that  $\bar{f}$  is a solution of (4), and explore the conditions under which  $f = k\bar{f} + J$  is a solution too. In principle, there are two

$$\Delta f + cf + d = 0 \quad (6)$$

$$\int_{\mathcal{V}} (cf + d) dV = 0 \quad (7)$$

which could be used to fix both  $k$  and  $J$ . However, using the fact that  $\bar{f}$  is a solution, it is easy to show that (6) implies (7). We thus cannot use these two conditions independently, and we again need to supplement the system with extra information, perhaps through the usual condition that  $f$  assume a specific value at a certain point, i.e. a constraint on the zero-frequency mode. Notice that substituting the definition of  $f$  into equation (6), one obtains

$$k(J) = \frac{cJ}{d} + 1. \quad (8)$$

If  $c$ ,  $k$ , and  $J$  are spatial constants, this implies, in particular, that  $f = k\bar{f} + J$  is a solution if and only if  $d$  is a constant as well. If that is the case, it is also straightforward to prove that (i) any solution is a superposition of Fourier modes of wave number zero or  $\sqrt{c}$ , (ii) the amplitude of the zero-frequency mode is equal to  $-d/c$  and (iii) the amplitude of the mode of frequency  $\sqrt{c}$  is arbitrary, i.e. extra information has to be supplied to render the problem well-posed. If, on the other hand,  $d$  is not a constant,  $f = k\bar{f} + J$  is not a solution (unless, of course,  $J = 0$ ). These three statements are proven in the [appendix](#).

Two features of this test must be highlighted: first, the integral equation does not need to be explicitly enforced as it is not independent from the corresponding PDE, and it will be progressively better and better satisfied as the algorithm converges to a solution; and second, the need to supply additional information to equation (6) in order to single out a solution depends on the nature of the source term  $d$ : if  $d$  is a constant, there is a rescaling freedom for the non-zero frequency components of  $f$ ; if it is not, such freedom does not exist.

Carrying out a similar analysis for a nonlinear equation is hardly, if at all, viable. I thus employ an empirical approach: in section 4.2, I will try and solve equation (6), with constant  $d$ , without imposing any additional constraint, and observe the solver's behavior. Not surprisingly, the symptom of an unconstrained zero-frequency mode is a runaway behavior of  $f$ , with an exponentially increasing error norm. Further down, in the tests involving the Einstein constraints with PBCs, I will again start out relaxing the equations with no extra conditions: if the error norm converges to below a tolerance set by the magnitude of the truncation error (see below for details), I will assume that the system has a single solution  $\bar{f}$ , or at least that any other solution is sufficiently far away from  $\bar{f}$  that it does not disturb the algorithm's convergence to  $\bar{f}$  via relaxation.

A final observation is that, whilst supplying extra information on, say, the expected zero-frequency mode is not necessary, it may be helpful to factor it into the initial guess for  $f$ , so as to speed up the convergence to a solution.

### 3. Implementation of a multigrid solver for spaces without boundaries

#### 3.1. Elements of multigrid theory

Elliptic PDEs can be solved with a variety of strategies, that vary in robustness, convergence speed and degree of generality. A fairly universal approach is that embodied by *relaxation methods*, whose basic idea is to transform an elliptic equation like

$$Lf = 0 \tag{9}$$

into a parabolic equation like

$$\dot{f} = Lf \tag{10}$$

the dot denoting a derivative with respect to time. Clearly, if the latter equation admits stationary solutions, those will also be solutions of the original problem. One can then start with an initial guess for  $f$  and solve the initial value problem, hoping to end up in a stationary state after some amount of time. Whilst this technique is rather dependable in most circumstances, the time required to drive  $\dot{f}$  to zero within the required tolerance becomes quickly prohibitive as the dimension of the numerical grid used to discretize the problem grows.

At this point, it is useful to introduce a few definitions. The *residual* of equation (9) is defined as

$$r := -Lf. \tag{11}$$

If  $\tilde{f}$  is a solution of (9), then its residual will be exactly zero, and  $\dot{f} = 0$ . Otherwise, the relaxation process will act on  $f$  according to

$$\dot{f} = -r \quad (12)$$

which explains why the convergence slows down as one gets closer to the solution, as  $r$  (and thus the time derivative of  $f$ ) decreases. Another useful quantity is the *solution error*, defined as the difference between a solution  $\tilde{f}$  of (9) and a given iterate  $f$ :

$$e := \tilde{f} - f. \quad (13)$$

*Multigrid* solvers are based on relaxation methods, and are designed to increase the convergence speed. They do so via two improvements.

- Since the higher-frequency modes of  $f$  relax faster than the low-frequency ones, one can speed up the algorithm's convergence by discretizing the equation on a hierarchy of grids, each covering the same physical domain with a different number of points. Each grid will be responsible to relax a particular section of  $f$ 's spectrum: finer grids will smooth the high-frequency portion of  $f$  while coarser ones will take care of the long-range modes.
- Since the relaxation speed is high when the residual is high, but decreases as one approaches the solution, one could conceive stopping the process as soon as  $e$  reaches a plateau, and start solving for  $e$  directly. The equation satisfied by  $e$  is straightforward to derive starting from the definitions, and reads

$$L(f + e) - L(f) = r. \quad (14)$$

One can then relax this equation until the convergence speed drops, then move to relax the equation for the error of  $e$ , and keep repeating until necessary.

These two improvements are then combined together: in the so-called V-cycle, for instance, one solves (9) on a grid, moves  $\psi$  and  $e$  to a coarser grid and solves (14), then updates  $\psi$  on the original grid and relaxes (9) again. More sophisticated schemes exist that improve on this basic strategy and generalize it to deeper grid hierarchies [28].

A variant of the multigrid approach, sometimes referred to as *multilevel* [28], involves a hierarchy of grids in which a subset of levels (typically the coarser ones) cover the entire domain, the rest being local fine grids in the regions where more resolution is needed, so as to extend the multigrid paradigm to adaptive-mesh-refinement (AMR) grids. For a discussion of this method in the context black-hole initial data, see [25]. In the case of interest here, the notable feature of multilevel schemes is that the global grids will have PBCs, while the local one will have Dirichlet boundary conditions (typically deriving from the prolongation of boundary points from the next coarser grid).

The detailed recipe used to solve an equation with a multigrid (or multilevel) algorithm can vary based on the equation's properties (e.g., linearity); an overview of this topic can be found in [28]. In the following subsection, I will describe the actual recipe adopted in this work.

### 3.2. Implementation details

In order to solve generic elliptic equations on periodic spaces, I have implemented a solver based on the multigrid recipe described in the previous subsection. The solver also has the capability to treat AMR grid hierarchies via a basic multilevel implementation, described below, which will only be used for the test in section 4.3.3.

I have implemented the solver as a component for the ET infrastructure, coupled to the Carpet AMR package [29]. The implementation is rather straightforward. I use a nonlinear

Gauss–Seidel relaxation algorithm to smooth out the solution on each level, and transfer information between grids using Carpet’s prolongation and restriction operators. I use fourth-order differencing to discretize the equation; prolongation is obtained by means of fifth-order interpolation and restriction is a point-wise copy. Given the nonlinear character of the equation, I adopt the full approximation scheme to solve the error equation. Different grids are accessed following the Full MultiGrid (FMG) prescription. When local AMR grids are present,  $f$  is relaxed there with simple Gauss–Seidel iterations and Dirichlet boundary conditions interpolated from the coarser grids; the error equation is not used on these levels.

When integrals are necessary, I calculate them using the midpoint rule. The corresponding information is then injected after each relaxation step, after boundary conditions are applied.

#### 4. Code tests

I discuss three test cases in order both to show the effectiveness of the solver in different scenarios and illustrate some of the subtleties involved in the solution of periodic problems with a multigrid scheme. The first two examples are Poisson’s and Helmholtz’s equations with a source term, i.e. the first and third case discussed in section 2.1. I then tackle the Einstein constraint system directly, first using a smooth solution corresponding to a spherically-symmetric stress–energy source, and then reproducing the initial data for a pair of equal-mass, non-spinning black holes in quasicircular orbit. For each problem (with the exception of the binary black-hole case), I first discuss the corresponding Dirichlet version, to verify the solver’s behavior on familiar grounds, and then solve the PBCs counterpart, highlighting the differences between the two cases.

##### 4.1. Poisson’s equation

Poisson’s equation simply reads

$$\Delta f + d = 0 \quad (15)$$

where  $d$  has no functional dependence on  $f$  but is an otherwise arbitrary function of the spatial coordinates. We can, for instance, choose it so that a certain function  $\bar{f}$  is a solution

$$d := -\Delta \bar{f}. \quad (16)$$

Let us set, for example

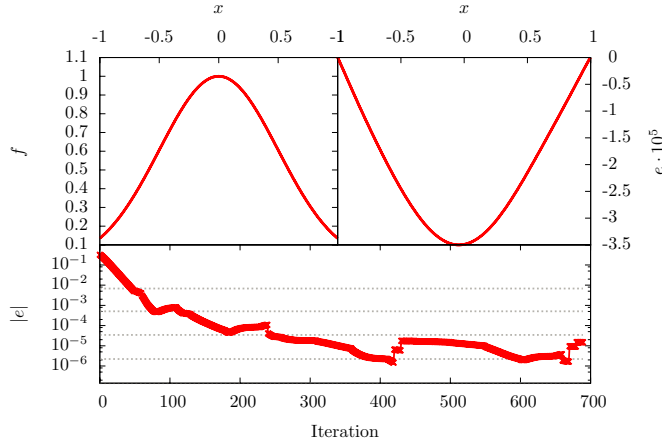
$$\bar{f} = e^{-r^2/2\sigma^2} \quad (17)$$

$$d = -\Delta \bar{f} = \left( \frac{3}{\sigma^2} - \frac{r^2}{\sigma^4} \right) e^{-r^2/2\sigma^2}. \quad (18)$$

The corresponding error equation reads

$$\Delta e - r = 0. \quad (19)$$

First, let us solve this problem with Dirichlet boundary conditions (derived from the exact solution) on a cubic domain with outer boundaries at  $x = \pm 1$ ,  $y = \pm 1$ , and  $z = \pm 1$  (we will use this domain for all of the examples below). To initialize the scheme, I set  $f = 0$  and  $e = 0$ , and then perform a FMG cycle on a hierarchy of five refinement levels, all covering the entire domain, with grid spacings  $\Delta = 0.2, 0.1, 0.05, 0.025, 0.0125$ . The Gauss–Seidel step is applied 50 times on the coarsest level, 10 on the finest level, 50 times on the other levels on the downward parts of the cycle, and 10 times otherwise. The error after each step is illustrated in figure 1. Notice that, as the discretized equation represents the continuum equation only up to



**Figure 1.** Top: final solution and associated error for Poisson's equation with Dirichlet boundary conditions. Bottom:  $L_1$  norm of the solution error. The dashed grey lines denote the norm of the truncation error on the five grids (the topmost corresponding to the coarsest grid).

the truncation error, this quantity constitutes an estimate of the lowest solution error attainable with a given grid configuration. In the following, for all the test cases for which a solution is known, I simply define the truncation error as the difference between the continuum laplacian operator applied to the exact solution and its finite-difference version. I plot the truncation error on the five grids along with the solution error for each test, to provide a reference for when the convergence of the solver is expected to stop. As is evident from the plots below, this limit is always achieved.

Let us turn to the PBCs case. Our considerations in the previous section imply that, in order for this equation to admit a solution,  $d$  must integrate to zero over the periodic cell. Since  $d$  is a given, fixed function, it suffices to guarantee that this condition is satisfied at the beginning.

As an exact solution to the problem, for instance, one could take

$$\bar{f} = \cos \pi x \cos \pi y \cos \pi z \quad (20)$$

provided that  $d = \Delta \bar{f} = -3\pi^2 \bar{f}$ , which does indeed have a zero integral over the periodic cell.

Let us attempt to obtain a solution with the multigrid solver. I again initialize both  $f$  and  $e$  to zero and begin a FMG cycle. After each relaxation sweep, I perform a rescaling so that

$$f \rightarrow f - f(\mathcal{O}) + 1 \quad (21)$$

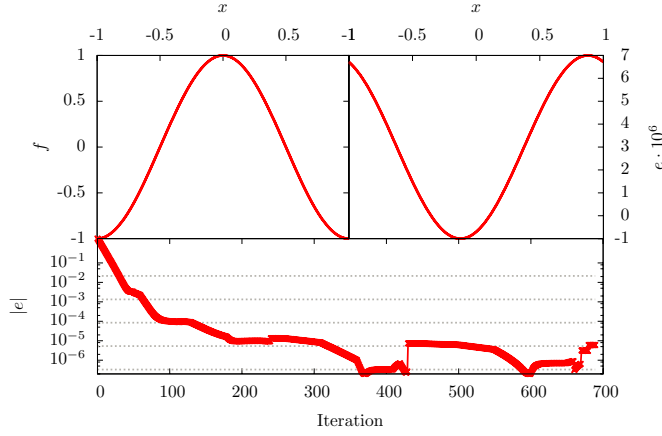
$$e \rightarrow e - e(\mathcal{O}) \quad (22)$$

where  $\mathcal{O}$  in this case denotes the origin. As discussed in the previous section, this rescaling is necessary since the linear equation in combination with PBCs does not constrain zero-frequency modes in either  $f$  or  $e$ , so it is necessary to set these constants by hand.

In this case, there is in fact an additional reason for the rescaling: any constant mode that creeps into either  $f$  or  $e$  due to numerical error will grow linearly in absolute value, since both the equation for  $f$  and that for the error take the form

$$\dot{f} = C \quad (23)$$





**Figure 2.** Top: final solution and associated error for Poisson's equation with PBCs. Bottom:  $L_1$  norm of the solution error. The dashed grey lines denote the norm of the truncation error on the five grids (the topmost corresponding to the coarsest grid).

for spatially-constant  $f$ . The scaling is thus necessary not only to make the problem well-posed, but also to counteract the danger of runaway behavior.

Inside the FMG cycle, I again relax 50 times in the downward portions and 10 times on the upward ones, with 50 iterations being executed on the coarsest level and 10 on the finest. In figure 2, I show the norm of the true error  $e$  as a function of the iteration, along with a plot of the final profiles for  $f$  and  $e$ .

#### 4.2. Inhomogeneous Helmholtz's equation

The inhomogeneous Helmholtz's equation is another linear elliptic PDE:

$$\Delta f + cf + d = 0. \quad (24)$$

In the Dirichlet BCs case, we can proceed as in the Poisson's equation's case and set

$$\bar{f} = e^{-r^2/2\sigma^2} \quad (25)$$

$$d = -\Delta \bar{f} - c\bar{f} = \left( \frac{3}{\sigma^2} - \frac{r^2}{\sigma^4} - c \right) e^{-r^2/2\sigma^2}. \quad (26)$$

Let us choose, for instance,  $c = -1$  (notice that a positive  $c$  would be problematic during the Gauss–Seidel relaxation as the equation would have exponentially growing modes). The value of  $c$  (implying that of  $d$ ) and the boundary conditions derived from the exact solution are sufficient to determine the solution. As illustrated in figure 3, the solver finds no problem converging to this solution.

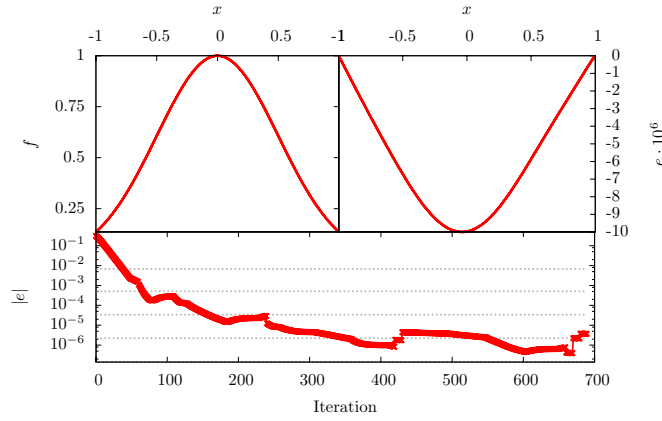
Let us turn to the periodic case: we can again find a solution to this problem by choosing, for instance

$$\bar{f} = 5 + \cos \pi x \cos \pi y \cos \pi z \quad (27)$$

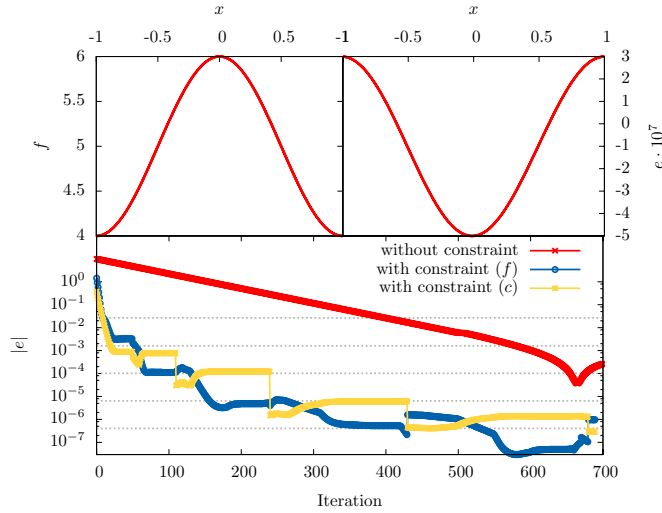
again with  $c = -1$  and

$$d = -\Delta \bar{f} + \bar{f} = (1 - 3\pi^2) \cos \pi x \cos \pi y \cos \pi z + 5. \quad (28)$$

Since  $d$  is not constant, according to the discussion in section 2.1 we do not need to worry about specifying a closure relation. Figure 4 shows that the solver can slowly converge to



**Figure 3.** Top: final solution and associated error for Helmholtz's equation with Dirichlet boundary conditions. Bottom:  $L_1$  norm of the solution error. The dashed grey lines denote the norm of the truncation error on the five grids (the topmost corresponding to the coarsest grid).



**Figure 4.** Top: final solution and associated error for Helmholtz's equation with  $c = -1$  and periodic boundary conditions. Bottom:  $L_1$  norm of the solution error, in a run that does not enforce the integral constraint (red), and in two runs that do (blue via  $f$  and yellow via  $c$ ). The dashed grey lines denote the norm of the truncation error on the five grids (the topmost corresponding to the coarsest grid).

the exact solution without aid. However, as discussed above, enforcing the integral constraint preconditions the iterate in a way that speeds up convergence tremendously, as shown in figure 4. Therefore, here and below, I will always apply this step even when not strictly necessary from the well-posedness standpoint. Notice also that the integral condition, which in this case reads

$$\int (cf + d) dV = 0 \quad (29)$$

can be enforced in at least two ways: tuning  $c$ , or setting the scale of  $f$ . In figure 4, I show the result for both methods, showing that rescaling  $f$  leads to a slightly superior performance. This observation also holds true qualitatively for the nonlinear examples below. I conjecture that this is because, if we set the value of  $c$  after each relaxation step, we are concurrently relaxing the solution *and* the equation, as  $c$  keeps changing too. This makes it harder for the algorithm to settle to a solution, since the equation itself oscillates between an iteration and the next. Furthermore, it is arguably more natural to specify the equation's coefficients (including  $c$ ) and leave the algorithm to find the appropriate scale for the solution, than specifying the scale *a priori* and find out at the end which equation we were trying to solve.

#### 4.3. Einstein constraints

In this section, we focus on the solution of the system of the four Einstein constraints, in the Lichnerowicz–York form:

$$\tilde{\Delta}\psi - \frac{\tilde{R}}{8}\psi - \frac{K^2}{12}\psi^5 + \frac{1}{8}\tilde{A}_{ij}\tilde{A}^{ij}\psi^{-7} = -2\pi\rho\psi^5 \quad (30)$$

$$\tilde{D}_i\tilde{A}^{ij} - \frac{2}{3}\psi^6\tilde{\gamma}^{ij}\tilde{D}_iK = 8\pi j^j\psi^{10} \quad (31)$$

where the tilde denotes conformal quantities, the conformal metric being related to the physical one by  $\tilde{\gamma}_{ij} = \psi^{-4}\gamma_{ij}$ . If one assumes  $\tilde{\gamma}_{ij}$  to be flat, and further decomposes  $\tilde{A}_{ij} = \tilde{D}_iX_j + \tilde{D}_jX_i + 2/3\tilde{\gamma}_{ij}\tilde{D}_kX^k$ , the following system is obtained (dropping the tilde from now on):

$$\Delta\psi - \frac{K^2}{12}\psi^5 + \frac{1}{8}A_{ij}A^{ij}\psi^{-7} = -2\pi\rho\psi^5 \quad (32)$$

$$\Delta X_i + D_iD_jX^j - \frac{2}{3}\psi^6\delta^{ij}D_iK = 8\pi j^i\psi^{10}. \quad (33)$$

In the general case, this is a coupled system of one nonlinear and three linear elliptic PDEs. As in the previous sections, one can trade some of the free data (say,  $\rho$  and  $j^i$ ) in exchange for the possibility to work with an exact solution; this is accomplished by choosing a form  $\bar{\psi}$  and  $\bar{X}^i$  for the conformal factor and the vector  $X^i$ , applying the right-hand sides of (32) to these functions, and setting the left-hand sides of the same system to the result:

$$\Delta\bar{\psi} - \frac{K^2}{12}\bar{\psi}^5 + \frac{1}{8}A_{ij}A^{ij}\bar{\psi}^{-7} = s \quad (34)$$

$$\Delta\bar{X}_i + D_iD_j\bar{X}^j - \frac{2}{3}\bar{\psi}^6\delta^{ij}D_iK = s^i \quad (35)$$

where

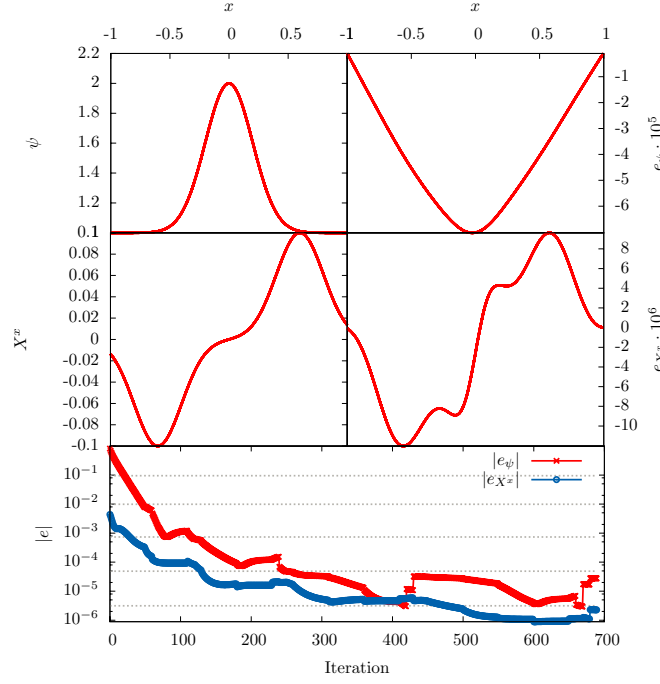
$$s = \Delta\bar{\psi} - \frac{K^2}{12}\bar{\psi}^5 + \frac{1}{8}A_{ij}A^{ij}\bar{\psi}^{-7} \quad (36)$$

$$s^i = \Delta\bar{X}_i + D_iD_j\bar{X}^j - \frac{2}{3}\bar{\psi}^6\delta^{ij}D_iK. \quad (37)$$

Since, naturally, there is no physical control or insight into this solution, it will be only used as a tool for code testing, in sections 4.3.1 and 4.3.2. I will then use this form for the Hamiltonian constraint, along with an analytical prescription for  $A_{ij}$ , to solve for the initial data of two non-spinning puncture in quasicircular orbit in section 4.3.3.

**4.3.1. Generic spherical source.** A simple form for  $\bar{\psi}$  and  $\bar{X}^i$  can be

$$\bar{\psi} = e^{-r^2/2\sigma^2} \quad (38)$$



**Figure 5.** Top: final solution and associated error for the Hamiltonian constraint equation with Dirichlet boundary conditions and a spherical source. Middle: final solution and associated error for the  $x$ -component of the momentum constraint equation. Bottom:  $L_1$  norm of the solution error for both quantities. The dashed grey lines denote the norm of the truncation error on the five grids (the topmost corresponding to the coarsest grid).

$$\bar{X}^i = e^{-r_{(i)}^2/2\sigma^2} \quad (39)$$

where  $r_{(i)} = |x^i - 0.5|$ . The idea is to represent the gravitational field of a smooth, isolated mass distribution, the nature of which plays no role in this test. I also choose

$$K = e^{-r^2/2\sigma^2} \quad (40)$$

and set the sources according to (36). Solving this equation on the same grid setup presented before, one obtains the results illustrated in figure 5.

**4.3.2. Sinusoidal source.** It is again instructive to study the same system in conjunction with PBCs. I now assume a solution of the form

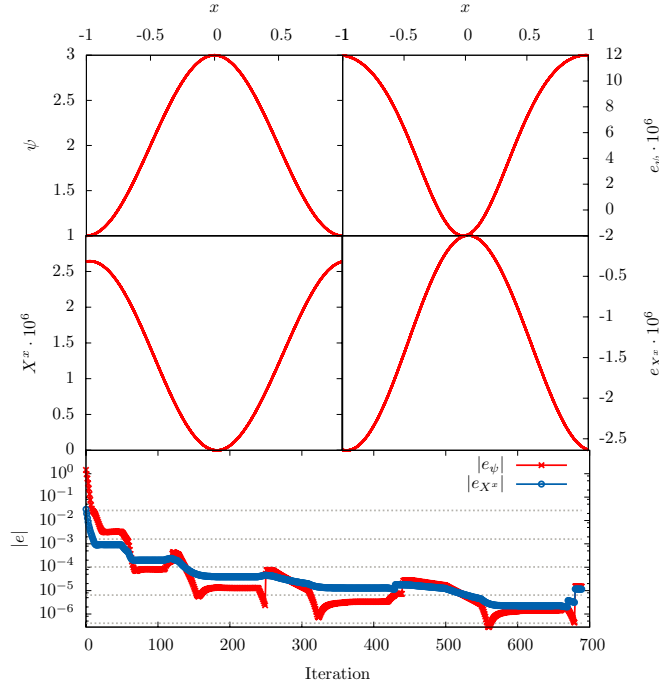
$$\bar{\psi} = 2 + \cos \pi x \cos \pi y \cos \pi z \quad (41)$$

$$\bar{X}^i = \sin \pi x \sin \pi y \sin \pi z \quad i = 1, 2, 3 \quad (42)$$

with a constant  $K = -0.1$ . Here, the integrability condition due to the periodic boundaries reads

$$\int \left( -\frac{K^2}{12} \psi^5 + \frac{1}{8} A_{ij} A^{ij} \psi^{-7} - s \right) dV = 0. \quad (43)$$

As for Helmholtz's equation, the solver does not need this step in order to converge to a solution, but enforcing this condition improves the convergence rate. This can be done by finding a root of



**Figure 6.** Top: final solution and associated error for the Einstein constraints with periodic boundary conditions and a spherical source. Middle: final solution and associated error for the  $x$ -component of the momentum constraint equation. Bottom:  $L_1$  norm of the solution error for both quantities. The dashed grey lines denote the truncation error on the five grids (the topmost corresponding to the coarsest grid).

$$f(A) = K^2 \int (\psi + A)^5 dV - \frac{1}{8} \int A_{ij} A^{ij} (\psi + A)^{-7} dV - \int s dV \quad (44)$$

which can easily be solved via the Newton–Raphson method after each Gauss–Seidel iteration. A test performed on the configurations described in this section shows that this improvement leads to a reduction of the final residual of more than order of magnitude with respect to the case where  $K$  is updated. It is worth noting that, in either case (updating  $K$  or updating  $\psi$ ), this step produces an additional numerical error that will only scale with the second power of the grid spacing, since it involves an integral calculated with the midpoint rule. The results for this test are illustrated in figure 6.

Notice that there is no integrability condition associated to the momentum constraint since the first two terms are divergences and the third has a zero integral over the cell due to the system’s symmetry under parity transformations  $x \rightarrow -x$ ,  $y \rightarrow -y$  and  $z \rightarrow -z$ .

**4.3.3. Two punctures.** As a last test, I will illustrate the behavior of the solver in a familiar case: that of two non-spinning black holes, in the puncture representation and in a quasicircular orbit around each other. Whilst an exact solution to this problem does not exist, I will compare the result with an existing component of the Einstein Toolkit, the TwoPunctures initial-data generator [17].

In this case, the metric is assumed to be conformally flat:

$$\gamma_{ij} = \psi^4 \delta_{ij} \quad (45)$$

and the extrinsic curvature is purely traceless ( $K = 0$ ) and given by

$$A_{ij} = \sum_{N=1}^2 \frac{3}{2r_N^2} (P_i^N n_j^N + P_j^N n_i^N + (\delta_{ij} - n_i^N n_j^N) n_N^k P_k^N) \quad (46)$$

where  $r_N$  is the distance to puncture  $N$ ,  $n_N^i$  the outgoing unit radial vector from this point, and  $P_N^i$  is its linear momentum. The only equation to solve is the Hamiltonian constraint for  $\psi$ :

$$\Delta\psi + \frac{1}{8} A_{ij} A^{ij} \psi^{-7} = 0. \quad (47)$$

The conformal factor  $\psi$  is also typically decomposed in a regular part  $u$  and a singular,  $1/r$  term according to

$$\psi = u + \sum_N \frac{m_N}{r_N} \quad (48)$$

where  $m_N$  represent the *bare masses* of the black holes. The equation to solve becomes

$$\Delta u + \frac{1}{8} A_{ij} A^{ij} \psi^{-7} = 0. \quad (49)$$

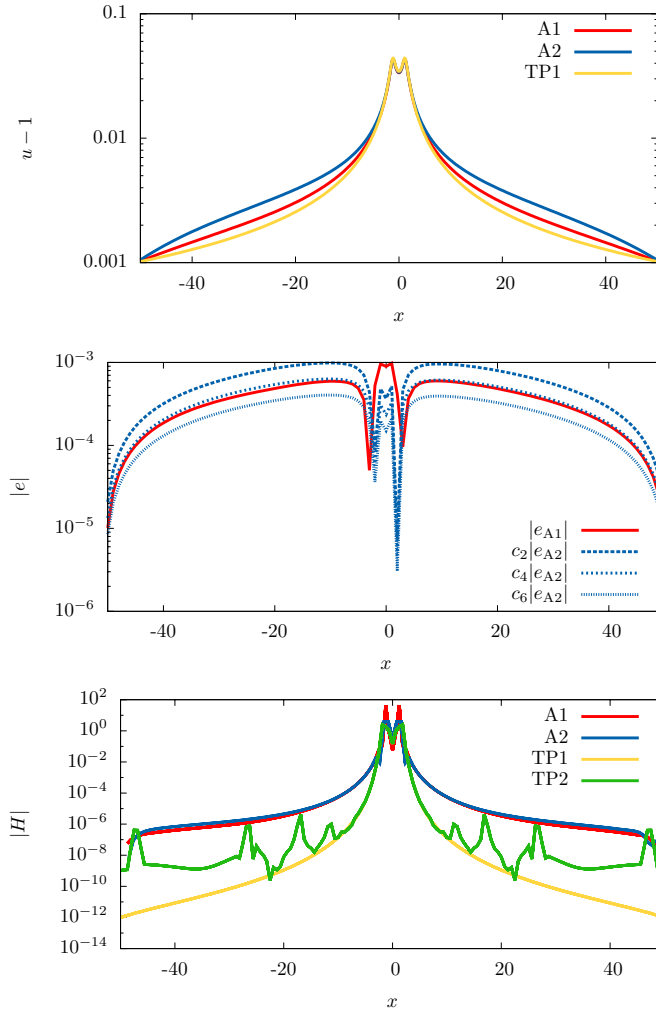
Since the spatial domain has a finite extent, I use Dirichlet boundary conditions obtained by setting  $u$  at the boundaries equal to the solution of `TwoPunctures`.

For this test, I have chosen to solve for the QC0 configuration [30], in which the black holes have bare masses equal to 0.453, are located on the  $x$ -axis at coordinates  $x = \pm 1.168\,642\,873$ , and have linear momenta parallel to the  $y$ -axis and equal to  $\pm 0.333\,191\,7498$ . In test A1, the grid structure contains a coarse grid C4 extending out to  $x = y = z = \pm 50$  with spacing 0.625, and three finer refinement levels C5, C6 and C7 of radii 25, 12 and 6 respectively, each with spacing equal to half that of the previous level. In addition, extra coarse grids C3 to C0, covering the whole domain with spacings equal to 1.25, 2.5, 5 and 10 respectively, are added in order to speed up the convergence according to the multigrid prescription. In this case, a canonical FMG cycle is performed on the domain-covering levels, whilst the local refinement levels are relaxed one by one, each providing the initial data and the boundary conditions for the next. In each V-cycle, the bottom level is relaxed 50 times, the top one 10, and the others 50 times in the downward part and 10 in the upward. Test A2 has the same grid structure as A1, but with all the grid spacings multiplied by 1.25.

The results for this test are illustrated in figure 7. Here, the final conformal factor is shown for A1 and A2. These are also compared to the solution from `TwoPunctures`, obtained using the ‘Evaluation’ option (TP1) on the same grid structure of A1; TP1 will be considered the ‘exact’ solution to test the convergence of the multigrid solver. The results agree to within a few percent; the difference can be accounted for by observing that the magnitude of the truncation error on C4 can be as high as  $10^{-2}$  (to obtain this estimate, I have calculated the numerical laplacian of the ‘exact’ solution from `TwoPunctures` on C3 and C4, and evaluated the difference, shown in figure 8). Figure 7 supports this claim by showing that the errors  $e_{A*} \equiv u_{TP1} - u_{A*}$  scale quartically with the grid spacing, as would be expected of the truncation error in a fourth-order finite-differencing scheme.

For reference, in figure 7, I also show the Hamiltonian-constraint violation obtained by tests A1 and A2, by the reference run TP1 and by a run TP2 in which `TwoPunctures` is used in the ‘Taylor expansion’ mode. I include this run for fairness of comparison: A1 and A2, run on 48 and 24 processing cores respectively, take about 1 min to complete, whilst TP1 requires about 50 times as long on 48 cores<sup>2</sup>. TP2, using a faster interpolation method on the same number of cores, requires about half a minute.

<sup>2</sup> In [31], the performance of `TwoPunctures` in the ‘Evaluation’ mode has recently been improved by roughly an order of magnitude. This would bring the walltime requirement from fifty times that of A1 down to about five times.

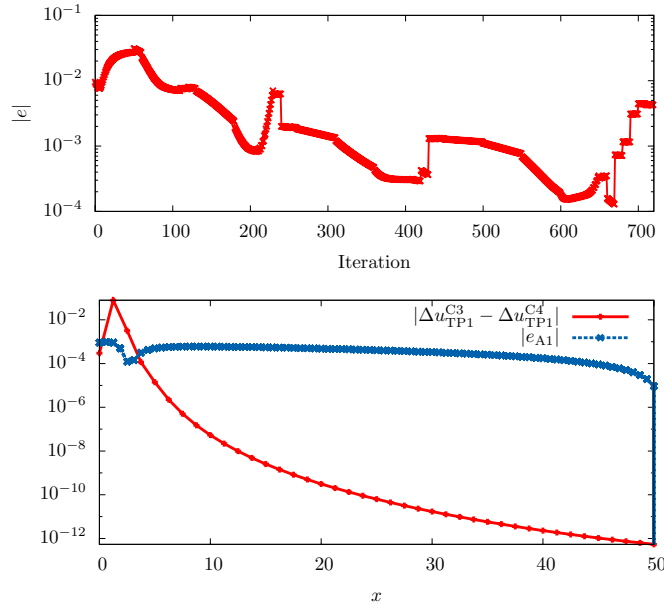


**Figure 7.** Conformal factor for the QC0 configuration. Top:  $u - 1$  for two different resolutions, from runs A1 and A1, plotted against the TwoPuncture solution TP1. Middle: error  $e_{A*} \equiv u_{TP1} - u_{A*}$ . The error of the coarse resolution A2 has been scaled according to the second-, fourth- and sixth-order expectation. The data shows fourth-order scaling, consistently with the spatial differencing scheme used by the solver. Bottom: Hamiltonian-constraint violation for runs A1, A2, TP1 and TP2.

## 5. Puncture lattice

In this section, I will use the solver described and tested above to solve the constraint system for a space filled with a cubic black-hole lattice. The solution will in practice be carried out on a single cell of this lattice, with PBCs at the cell faces. Each cell will contain a single black hole at its center.

As discussed in [15], if one is interested in conformally flat initial data, the extrinsic curvature cannot be zero. A possible construction for a lattice of non-spinning, non-maximally-sliced black holes has been proposed in [14]: according to this recipe, one starts with the following *ansatz* for the conformal factor



**Figure 8.** Top: solution error for run A1. Notice that it is not possible to estimate the truncation error here as in the previous examples, due to the lack of an analytical exact solution. One can, however, take the absolute value of the difference in the numerical laplacian  $\Delta u_{\text{TP1}}$  computed on level C3 and C4 and compare it with the solution error  $e_{\text{A1}} = u_{\text{TP1}} - u_{\text{A1}}$  on C4 (bottom): this shows that the large truncation error due to finite-differencing around the puncture sets a limit for the convergence of  $u_{\text{A1}}$  to  $u_{\text{TP1}}$  over the whole domain.

$$\psi = u + \frac{m}{2r}(1 - W(r)) \quad (50)$$

and for the trace of the extrinsic curvature:

$$K = K_c W(r) \quad (51)$$

where  $K_c$  is a constant and  $W(r)$  is a function which is equal to zero inside a sphere around  $r = 0$ , equal to one at large distances from  $r = 0$ , and has a transition region in between, e.g.:

$$W(r) = \begin{cases} 0 & \text{for } 0 \leq r \leq \ell \\ ((r - \ell - \sigma)^6 \sigma^{-6} - 1)^6 & \text{for } \ell \leq r \leq \ell + \sigma \\ 1 & \text{for } \ell + \sigma \leq r. \end{cases} \quad (52)$$

This has the advantage of starting out close to familiar solutions (the Schwarzschild solution for  $r < \ell$  and a Friedmann–Lemaître–Robertson–Walker model for  $r > \ell + \sigma$ ) on most of the domain.

The system to solve is then

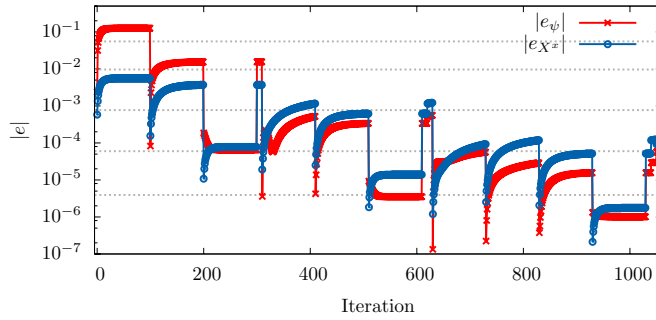
$$\Delta u - \Delta \left( \frac{m}{2r} W(r) \right) - \frac{K^2}{12} \psi^5 + \frac{1}{8} A_{ij} A^{ij} \psi^{-7} = 0 \quad (53)$$

$$\Delta X^i + \partial^i \partial_j X^j - \frac{2}{3} \psi^6 \partial^i K = 0 \quad (54)$$

with

$$A_{ij} = \partial_i X_j + \partial_j X_i - \frac{2}{3} \delta_{ij} \partial_k X^k \quad (55)$$





**Figure 9.** Norm of the solution errors corresponding to  $u$  (red) and  $X^x$  (blue) for run L1. The dashed grey lines denote the truncation error on the five grids (the topmost corresponding to the coarsest grid).

for  $u$  and  $X^i$ . As in section 4.3.2, an integral condition, which now reads

$$K_c^2 \int W^2 \psi^5 - 2\pi \left( m + \frac{1}{8} \int A_{ij} A^{ij} \psi^{-7} \right) = 0 \quad (56)$$

has to be respected by all the solutions. Again, I find that enforcing this condition explicitly is not necessary, but doing so greatly speeds up the convergence. In [14], the condition is imposed by solving for  $K_c$ , and updating this parameter in the Hamiltonian constraint equation at the next iteration. However, resetting the scale of  $\psi$  as in section 4.3.2 leads to an even faster convergence. Here, we need to find a root of

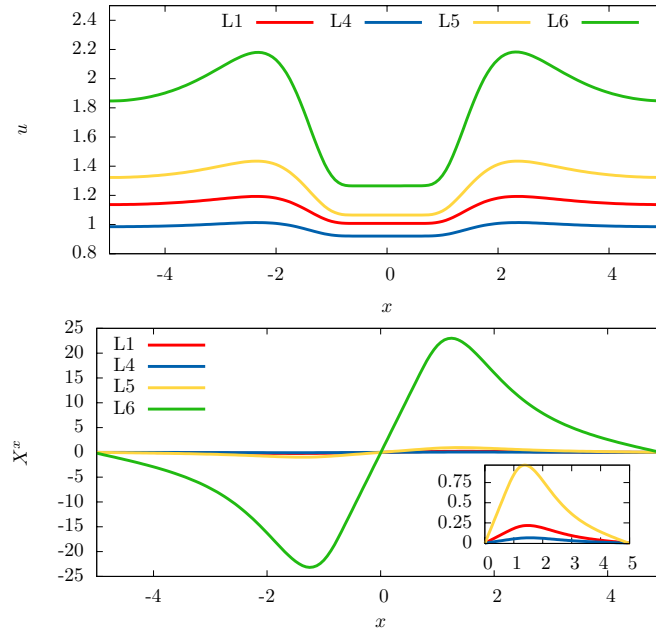
$$f(A) = K_c^2 \int W^2 (\psi + A)^5 - 2\pi \left( m + \frac{1}{8} \int A_{ij} A^{ij} (\psi + A)^{-7} \right) \quad (57)$$

which again is solved with the Newton–Raphson method. For the same reasons described in 4.3.2, there is no condition associated to the momentum constraint.

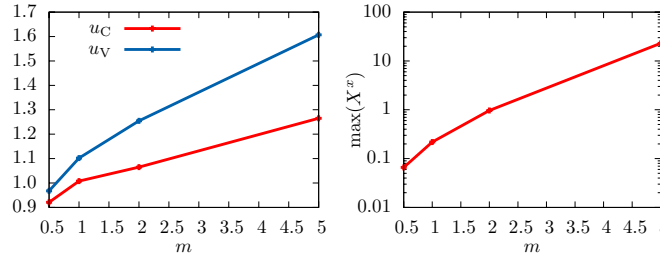
I first solve for a configuration L1 where the outer boundaries of the periodic cell are located at  $x = \pm 5$ ,  $y = \pm 5$ , and  $z = \pm 5$ , the black hole at the origin has a mass  $m = 1$ , the trace of the extrinsic curvature  $K_c = -0.21$  initially, and the parameters defining the transition region are  $\ell = 0.5$  and  $\sigma = 4$ . The cell is covered by five refinement levels (all extending out to the outer boundaries), with coarse spacing  $\Delta_0 = 1$ . This time, the Gauss–Seidel solver operates 100 times on the coarsest level, 10 on the finest, and 100 and 10 times on the intermediate ones on the downward and upward portion of each V-cycle, respectively.

In figure 9, it is shown how the errors  $u$  and  $X^x$  converge up to the discretization error, in a manner similar to what was presented in section 4 above. Notice, however, that the definitions of solution error and truncation error adopted here are rather different, given to the lack of an exact solution to take as reference. Therefore,  $e$  in figure 9 is not the difference between the numerical solution and the (unknown) exact solution, but the estimate of this error obtained in the multigrid scheme, i.e. by relaxing equation (14). Likewise, the truncation error is obtained by subtracting the analytic derivative of  $W$  in (52) from the numerical one; this is reasonable given that the largest contribution to the finite-differencing error is likely to come from computing derivatives in the transition region. However, it has to be kept in mind that this is not the truncation error associated to any of the solution functions.

Figure 10 shows the final solution for  $u$  and  $X^x$  (the other two components of  $X^i$  can be obtained by remembering the symmetry of this system under permutations of the three Cartesian directions). On the same plot, three other variants L4, L5 and L6, obtained with the same parameters but changing  $m$  to 0.5, 2 and 5 respectively, are displayed. Qualitatively,



**Figure 10.** Regular part of the conformal factor  $u$  (top) and  $x$ -component of the vector  $X^i$  for a black-hole lattice with  $m = 1$  at the origin, cell faces at  $x = \pm 5$ ,  $y = \pm 5$ , and  $z = \pm 5$ , and parameters  $\ell = 0.5$  and  $\sigma = 4$ .

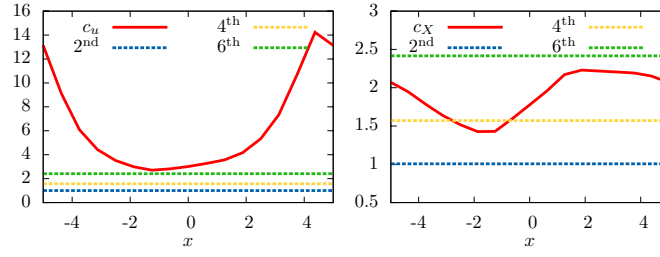


**Figure 11.** Left: values of the regular part of the conformal factor  $u$  at the center of the cell and at a vertex, as a function of  $m$ . Right: maximum of  $X^x$  on the  $x$ -axis as a function of  $m$ .

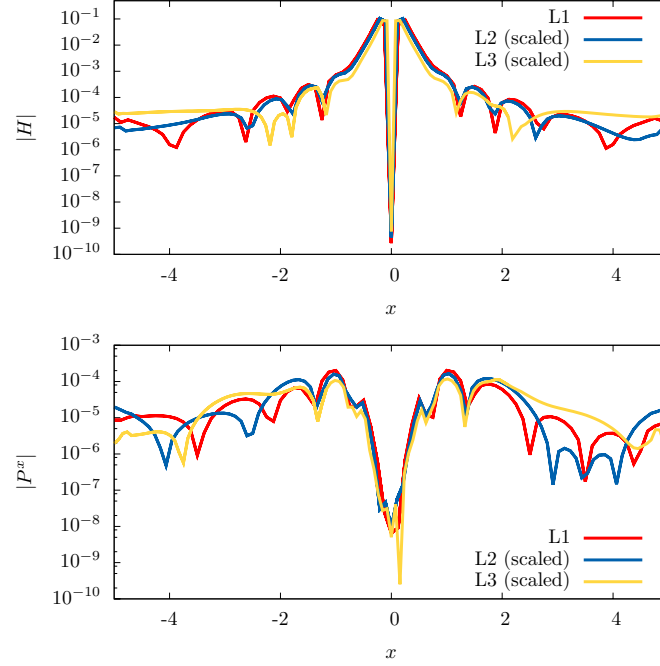
runs with different  $m$  appear similar, with a change in scale of both  $u$  and  $X^x$  as the major difference. This is quantified in figure 11, where the value of  $u$  at the center of the cell and at a vertex, and the maximum of  $X^x$  on the  $x$ -axis are shown as a function of  $m$ . Since  $u$  represents the nonlinear interaction between the black holes in the lattice, this result reflects the intuitive expectation that larger black-hole masses should lead to stronger nonlinear effects. Due to the coupling term in the momentum constraint, this property is inherited by the traceless part of the extrinsic curvature.

Given the lack of an exact solution, I next perform a three-point resolution study with runs L1, L2 (with  $\Delta_0 = 0.8\bar{3}$ ) and L3 (with  $\Delta_0 = 0.625$ ). The results are illustrated in figure 12, where the ratio of the difference between solutions at different resolutions

$$c_u = \frac{|u_{L1} - u_{L2}|}{|u_{L2} - u_{L3}|} \quad (58)$$



**Figure 12.** Convergence factors  $c_u$  and  $c_X$  along the  $x$ -axis.

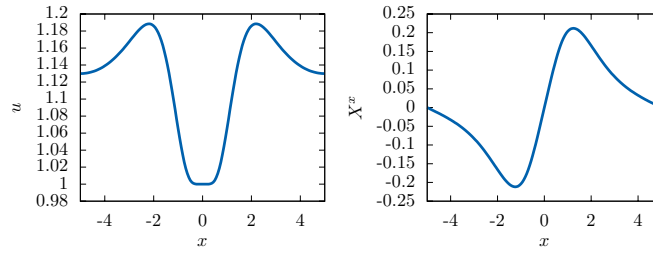


**Figure 13.** Hamiltonian (top) and momentum (bottom) constraint violations for runs L1, L2 and L3. Runs L2 and L3 have been rescaled by the factor expected for fourth-order convergence.

$$c_X = \frac{|X_{L1}^x - X_{L2}^x|}{|X_{L2}^x - X_{L3}^x|} \quad (59)$$

is plotted against the expected value for a second-, fourth- and sixth-order numerical error. Whilst the values of  $c_u$  and  $c_X$  support a scaling in this range, the analysis is clearly much less straightforward than in the asymptotically-flat example of figure 7, especially with respect to the convergence of  $u$  near the boundaries, where the difference between solutions at different resolutions seems to decrease much more rapidly than elsewhere. Coupling the relaxation process with the update of  $\psi$  (a step which entails the calculation of the two integrals in (57), with the associated error) could be responsible for this effect.

In figure 13, the value of the Hamiltonian constraint and of the  $x$ -component of the momentum constraint are plotted for L1, L2 and L3, with the two finest resolutions rescaled for fourth-order convergence. Here, again, one observes that the results are acceptable in



**Figure 14.** Configuration L7. Left: regular part of the conformal factor. Right:  $x$ -component of  $X^t$ . This is to be compared to figure 4 and 6 of [14].

**Table 1.** Summary of the lattice-puncture simulations discussed in section 5. The missing horizon mass for run L4 is due to the fact that it is not possible to locate such a small horizon at this resolution.

Run	$m$	$\ell$	$\sigma$	$m_{\text{hor}}$	$L_p$	$\Delta_0$	$ H $	Reset
L1	1	0.5	4	1.007 962	12.2607	1	$1.29 \times 10^{-4}$	$\psi$
L2	1	0.5	4	1.008 002	12.2607	$0.8\bar{3}$	$7.03 \times 10^{-5}$	$\psi$
L3	1	0.5	4	1.008 005	12.2604	0.625	$2.72 \times 10^{-5}$	$\psi$
L4	0.5	0.5	4	—	9.413 88	1	$1.61 \times 10^{-3}$	$\psi$
L5	2	0.5	4	2.130 278	15.9943	1	$2.36 \times 10^{-5}$	$\psi$
L6	5	0.5	4	6.332 967	26.9543	1	$8.99 \times 10^{-4}$	$\psi$
L7	1	0.1	4.8	1.003 605	12.1985	1	$1.30 \times 10^{-4}$	$K$

the central region, where finite-differencing dominates the error, but degrade toward the boundaries, where the finite-differencing error decreases below the error introduced by the volume integrals.

Finally, I also search for the black-hole apparent horizon and calculate the corresponding mass in each case. All these results are summarized in table 1.

Notice that L7, the last run in table 1, has the same physical parameters of L1, but a slightly larger transition region, and is obtained by updating  $K_c$  through the integral condition (and resetting  $\psi$  to 1). The results are illustrated in figure 14, and provide a comparison with an independent code, as this run is identical to the second configuration presented in [14] (see figures 4 and 6 there, but notice the typo immediately after equation (38), where  $\ell$  is attributed the value of  $\sigma$ , and viceversa). In particular, L7 correspond to the  $L = 10R_g$  run in [14] for  $m = 1$  (so that  $R_g = 0.5$ ); using this value of  $R_g$ , the  $y = 0$  sections of the third panel in figures 4 and 6 are indistinguishable from the two panels in figure 14. This constitutes another independent check of the solver on a problem that does not have an analytical solution.

## 6. Conclusions

In the first part of this work, I have presented a new multigrid code developed as a module for the Cactus framework. The code is particularly suited to tackle elliptic PDEs in periodic spaces, as it implements the necessary infrastructure to deal with the additional conditions that may be associated with this type of problems. After a discussion of what these conditions may look like in a few simple scenarios, I presented a series of tests, with the goal of illustrating the behavior of relaxation methods in periodic spaces as well as establishing the correct functioning of this implementation.

In the second part, I applied the code to the solution of the Einstein constraints in a periodic cell containing a single black hole at its center. The solution represents a spatial slice of an infinite, cubic black-hole lattice, and can be used as initial data to study the evolution of such a system in the case of non-zero initial mean curvature. This result is of primary importance for the study of black-hole lattices as models of inhomogeneous universes, as their evolution has been obtained so far, in full General Relativity, only in the case of zero initial mean curvature [15], where initial data could be constructed analytically.

This work lays the basis for the study of expanding black-hole lattices, which is presented elsewhere [32], and, more generally, for the numerical construction of arbitrary periodic spacetimes. Three-dimensional simulations of cosmological models in full General Relativity can, in particular, extend the current analytical results [33] regarding the role of small-scale inhomogeneities in the late universe; whether periodic boundary conditions represent a reasonable assumption for these studies remains, for the time being, an open question.

Further improvements on the solver, such as the implementation of higher-order schemes or colored stencils, may also be the subject of future work.

## Acknowledgments

I acknowledge Lars Andersson, Mikołaj Korzyński, Ian Hinder, Bruno Mundim, Oliver Rinne and Erik Schnetter for many clarifications and suggestions. This work was supported by a Marie Curie International Reintegration Grant (PIRG05-GA-2009-249290); computations were carried out on the MPI-GP Damiana and Datura clusters, as well as on SuperMUC at the Leibniz-Rechenzentrum in Munich.

## Appendix. Helmholtz's equation with a constant source

In this appendix, I explain the three statements quoted in 2.1, and valid for Helmholtz's equation

$$\Delta f + cf + d = 0 \quad (\text{A.1})$$

with a constant coefficient  $c$  and a constant source term  $d$ . I will restrict myself to the one-dimensional case; the generalization is obvious.

- If  $d$  is constant, the only solutions are Fourier modes with frequency zero or  $\sqrt{c}$ . This can be proven by writing a Taylor expansion for  $f$ :

$$f = \sum_{n=0}^{\infty} a_n x^n \quad (\text{A.2})$$

and substituting it into Helmholtz's equation, obtaining

$$\sum_{n=2}^{\infty} a_n n(n-1)x^{n-2} + c \sum_{n=0}^{\infty} a_n x^n + d = 0 \quad (\text{A.3})$$

which, for constant  $d$ , is only satisfied if

$$ca_0 + d + 2a_2 = 0 \quad (\text{A.4})$$

$$ca_1 + 6a_3 = 0 \quad (\text{A.5})$$

$$ca_n + (n+1)(n+2)a_{n+2} = 0. \quad (\text{A.6})$$

One is free to choose  $a_0$  and  $a_1$ , and the other coefficients will be given by

$$a_{2p} = (-1)^p (d - ca_0) \frac{c^{p-1}}{(2p)!} \quad p \geq 1 \quad (\text{A.7})$$

$$a_{2p+1} = (-1)^p \frac{c^p a_1}{(2p+1)!} \quad p \geq 1. \quad (\text{A.8})$$

Substituting back into  $f$ , and after a few manipulations, one finally obtains

$$f(x) = -\frac{d}{c} + \left(\frac{d}{c} + a_0\right) \cos(\sqrt{c}x) + \frac{1}{\sqrt{c}} \sin(\sqrt{c}x). \quad (\text{A.9})$$

- The amplitude of the zero-frequency mode of  $f$  is fixed and equal to  $-d/c$ ; the other modes can have arbitrary amplitude. This can be shown by working out the conditions under which  $h = kf + J$  is a solution, i.e.

$$\Delta h + ch + d = 0. \quad (\text{A.10})$$

One recovers the familiar relation

$$k(J) = \frac{cJ}{d} + 1 \quad (\text{A.11})$$

so that a family of solutions is given by

$$\begin{aligned} h &= \left(\frac{cJ}{d} + 1\right) f + J \\ &= -\frac{d}{c} + \left(\frac{cJ}{d} + 1\right) \left( \left(\frac{d}{c} + a_0\right) \cos(\sqrt{c}x) + \frac{1}{\sqrt{c}} \sin(\sqrt{c}x) \right) \end{aligned} \quad (\text{A.12})$$

with arbitrary  $J$ .

## References

- [1] Garfinkle D and Lehner L 2012 *Class. Quantum Grav.* **29** 240301
- [2] Anninos P 2001 *Living Rev. Rel.* **4** 2
- [3] Laguna P, Kurki-Suonio H and Matzner R A 1991 *Phys. Rev. D* **44** 3077–86
- [4] Kurki-Suonio H, Laguna P and Matzner R A 1993 *Phys. Rev. D* **48** 3611–24 (arXiv:astro-ph/9306009)
- [5] Anninos P 1998 *Phys. Rev. D* **58** 064010
- [6] Anninos P and McKinney J 1999 *Phys. Rev. D* **60** 064011
- [7] Zhao X and Mathews G J 2011 *Phys. Rev. D* **83** 023524 (arXiv:0912.4750)
- [8] Ellis G F R 2011 *Class. Quantum Grav.* **28** 164001 (arXiv:1103.2335v1)
- [9] Buchert T 2011 *Class. Quantum Grav.* **28** 164007
- [10] Clarkson C, Ellis G, Larena J and Umeh O 2011 *Rep. Prog. Phys.* **74** 112901 (arXiv:1109.2314)
- [11] Roukema B F, Bajtlik S, Biesiada M, Szaniewska A and Jurkiewicz H 2007 *Astron. Astrophys.* **463** 861 (arXiv:astro-ph/0602159)
- [12] Roukema B F and Rozanski P T 2009 *Astron. Astrophys.* **502** 27 (arXiv:astro-ph/0902.3402)
- [13] Ostrowski J J, Roukema B F and Bulinski Z P 2012 *Class. Quantum Grav.* **29** 165006 (arXiv:astro-ph/1109.1596)
- [14] Yoo C M, Abe H, Nakao K i and Takamori Y 2012 *Phys. Rev. D* **86** 044027 (arXiv:1204.2411)
- [15] Bentivegna E and Korzynski M 2012 *Class. Quantum Grav.* **29** 165007 (arXiv:1204.3568)
- [16] LORENE 2013 <http://www.lorene.obspm.fr>
- [17] Ansorg M, Bruggmann B and Tichy W 2004 *Phys. Rev. D* **70** 064011 (arXiv:gr-qc/0404056)
- [18] Grandclement P 2010 *J. Comput. Phys.* **229** 3334–57 (arXiv:0909.1228)
- [19] Seto N 1999 *Astrophys. J.* **523** 24–31
- [20] Buchert T and Ehlers J 1997 *Astron. Astrophys.* **320** 1–7 (arXiv:astro-ph/9510056)
- [21] Hernquist L, Bouchet F R and Suto Y 1991 *Astrophys. J.* **75** 231–40
- [22] Lim W C, Andersson L, Garfinkle D and Pretorius F 2009 *Phys. Rev. D* **79** 123526 (arXiv:0904.1546)

- [23] Loffler F *et al* 2012 *Class. Quantum Grav.* **29** 115001 (arXiv:1111.3344)
- [24] Brandt S and Bruegmann B 1997 *Phys. Rev. Lett.* **78** 3606–9 (arXiv:gr-qc/9703066)
- [25] Brown J D and Lowe L L 2005 *J. Comput. Phys.* **209** 582–98 (arXiv:gr-qc/0411112)
- [26] East W E, Ramazanoglu F M and Pretorius F 2012 *Phys. Rev. D* **86** 104053 (arXiv:1208.3473)
- [27] Choquet-Bruhat Y, Isenberg J and Pollack D 2007 *Class. Quantum Grav.* **24** 809–28 (arXiv:gr-qc/0610045)
- [28] Briggs W, Henson V and McCormick S 2000 *A Multigrid Tutorial (Miscellaneous Books)* (Philadelphia, PA: SIAM) ISBN 9780898714623
- [29] Schnetter E, Hawley S H and Hawke I 2004 *Class. Quantum Grav.* **21** 1465–88
- [30] Baker J G, Campanelli M, Lousto C and Takahashi R 2002 *Phys. Rev. D* **65** 124012 (arXiv:astro-ph/0202469)
- [31] Paschalidis V, Etienne Z B, Gold R and Shapiro S L 2013 arXiv:1304.0457
- [32] Bentivegna E and Korzynski M 2013 *Class. Quantum Grav.* **30** 235008 (arXiv:1306.4055)
- [33] Andersson L and Coley A 2011 *Class. Quantum Grav.* **28** 160301