OPEN ACCESS
IOP Publishing
Classical and Quantum Gravity
Class. Quantum Grav. **42** (2025) 055008 (20pp)
https://doi.org/10.1088/1361-6382/ad9796

Revisiting the apparent horizon finding problem with multigrid methods

Hon-Ka Hui* **and Lap-Ming Lin**

Department of Physics, The Chinese University of Hong Kong, Hong Kong Special Administrative Region of China, People's Republic of China

E-mail: hkhui@link.cuhk.edu.hk and lmlin@cuhk.edu.hk

Received 22 April 2024; revised 1 November 2024
Accepted for publication 26 November 2024
Published 11 February 2025

Abstract

Apparent horizon plays an important role in numerical relativity as it provides a tool to characterize the existence and properties of black holes on three-dimensional spatial slices in 3+1 numerical spacetimes. Apparent horizon finders based on different techniques have been developed. In this paper, we revisit the apparent horizon finding problem in numerical relativity using multigrid-based algorithms. We formulate the nonlinear elliptic apparent horizon equation as a linear Poisson-type equation with a nonlinear source, and solve it using a multigrid algorithm with Gauss–Seidel line relaxation. A fourth order compact finite difference scheme in spherical coordinates is derived and employed to reduce the complexity of the line relaxation operator to a tri-diagonal matrix inversion. The multigrid-based apparent horizon finder developed in this work is capable of locating apparent horizons in generic spatial hypersurfaces without any symmetries. The finder is tested with both analytic data, such as Brill–Lindquist multiple black hole data, and numerical data, including off-centered Kerr–Schild data and dynamical inspiraling binary black hole data. The obtained results are compared with those generated by the current fastest finder AHFinderDirect (Thornburg 2003 *Class. Quantum Grav.* **21** 743), which is the default finder in the open source code Einstein Toolkit. Our finder performs comparatively in terms of accuracy, and starts to outperform AHFinderDirect at high angular resolutions ($\sim 1^\circ$) in terms of speed. Our finder is also more flexible to initial guess, as opposed to the

[*] Author to whom any correspondence should be addressed.

Newton's method used in AHFinderDirect. This suggests that the multigrid algorithm provides an alternative option for studying apparent horizons, especially when high resolutions are needed.

Keywords: apparent horizon, multigrid, compact finite differences, Poisson equations

1. Introduction

Apparent horizon has been an important concept for the numerical studies of spacetime that involve black holes. It serves as a real-time indicator of the existence of a black hole in numerical simulations, as its presence implies the presence of a surrounding event horizon [25]. The determination of apparent horizon during numerical simulations is made possible because it can be defined locally in time. This is in contrast to the teleological nature of event horizon, whose determination requires the full knowledge of the future evolution of a spacetime. Knowing the presence of an apparent horizon during simulation is important not only because it allows the estimation of the properties of the black hole, such as its mass and spin, but it also marks the region within which singularity may appear at a later time [32], such region sometimes is necessary to be avoided through excision to maintain stability of the simulation [40] (also see, e.g. [1, 5] for using singularity-avoiding gauge instead of excision).

The dynamical evolution of the apparent horizon is also of interest for understanding black holes and gravitational wave physics. The notion of dynamical horizons (e.g. [3, 38], also see [4] for a review), which is loosely speaking a world tube formed by the foliation of the apparent horizon at different time slices, can be used instead of event horizons to better study the interaction and merging of black holes [34]. Recent studies have also suggested correlation between the dynamics of (common) apparent horizon in black hole binaries with the gravitational waves emitted [22, 35], which could hint a possible imprint of event horizon properties on the signal.

Numerous efforts have been dedicated to developing an efficient apparent horizon finder, as discussed in a comprehensive review [43]. Currently, the fastest algorithm for locating the apparent horizon is based on Newton's method ([37, 42]), along with some variants of the matrix inversion scheme for the associated Jacobi equation. In general, such a method suffers from the poor scaling of the computational time complexity and the reliance on a good initial guess. On the other hand, while multigrid methods in general exhibit much better scaling, they have not been applied to the apparent horizon searching problem to our best knowledge. Multigrid methods, owing to their efficiency in solving elliptic partial differential equations (PDEs), have been applied to different problems in numerical relativity, including the initial data construction of binary stellar objects (see, e.g. [8, 12, 19, 21, 31]) and more recently to a relativistic hydrodynamic code under the conformally flat assumption [16]. It should also be pointed out that a review article introducing the multigrid method for numerical relativists was already written about 40 years ago [17]. With the expectation that multigrid methods can solve elliptic equations efficiently, we aim to implement an apparent horizon finder in a multigrid approach and examine whether this could be a better algorithm in terms of speed and robustness, especially for cases when one needs to resolve the apparent horizon with high enough resolution to increase the precision of the inferred properties of the black hole in simulations.

This paper is organized as follows. In section 2, we first describe the equation to solve for the apparent horizon, and present the implementation of the multigrid algorithm in our apparent horizon finder. Section 3 includes the test results of our finder in terms of accuracy and efficiency in several benchmark data. Finally, we summarize our work and give future prospects in section 4.

2. Method and implementation

Locating the apparent horizon requires solving a non-linear elliptic PDE—the expansion equation. In this section, we first introduce the variant ansatz of the expansion equation we employed. We then give a brief overview of the multigrid methods in general and the specific scheme we used in this work.

2.1. Apparent horizon and the expansion equation

A black hole region is characterized by the event horizon, which is defined as the boundary of the causal past of the future null infinity [25]. The event horizon is therefore a global property of the underlying 4-dimensional spacetime, and its determination in a numerical simulation can only be done approximately by integrating null geodesics (e.g. [26]) or null surfaces (e.g. [2]) backwards in time once the spacetime has essentially settled down to a final stationary state (see [43] for the review of these algorithms). However, if one would like to trace the motion of a black hole during a simulation, it is important to have a local (in time) characterization of the black hole. This can be achieved by the concept of apparent horizon.

Intuitively, a black hole is a region where light rays are 'trapped' and cannot escape to infinity. This idea can be formalized by introducing the notion of a trapped region in a 3-dimensional spatial slice embedded in the 4-dimensional spacetime, defined as the union of all trapped surfaces[1]. The outer boundary of the trapped region is defined to be the apparent horizon. However, this mathematical definition of the apparent horizon is not convenient for practical use in numerical simulations. In the following, we shall provide a slightly different definition of the apparent horizon that is used in numerical relativity, using the expansion function Θ. We also describe how Θ depends on the spacetime geometric variables, and the ansatz we adopted to solve for an apparent horizon.

2.1.1. Notations and definitions. Let n^a denote the timelike future-pointing unit normal vector to a spacelike hypersurface Σ_t, and s^a denote the spacelike outward-pointing unit normal vector to an arbitrary 2-dimensional smooth closed surface $\mathcal{S}$ embedded in Σ_t. The spatial 3-metric γ_{ab} induced on Σ_t and the 2-metric m_{ab} induced on $\mathcal{S}$ are given by

$$\gamma_{ab} = g_{ab} + n_a n_b, \tag{1}$$

$$m_{ab} = \gamma_{ab} - s_a s_b = g_{ab} + n_a n_b - s_a s_b, \tag{2}$$

where g_{ab} is the spacetime metric. By denoting the unit vector along the future-pointing outgoing null geodesics by $k^a = (s^a + n^a)/\sqrt{2}$, the expansion function can be expressed as (see, e.g. [9]),

$$\Theta \equiv m^{ab}\nabla_a k_b = \left(D_i s^i + K_{ij} s^i s^j - K\right)/\sqrt{2}, \tag{3}$$

[1] A trapped surface is a closed spacelike 2-surface embedded in the 3-dimensional spatial slice with the property that the expansion function Θ of both future-pointing outgoing and ingoing null geodesics is negative everywhere on the surface.

where ∇_a and D_a are the covariant derivatives associated with the metric tensors g_{ab} and γ_{ab}, respectively; K_{ij} is the extrinsic curvature, and K denotes its trace (i.e. $K \equiv \gamma^{ij} K_{ij}$). $\mathcal{S}$ is called a marginally outer trapped surface (MOTS) if $\Theta = 0$ everywhere on $\mathcal{S}$. There could also be multiple MOTSs or even nested MOTSs in Σ_t. The apparent horizon is defined to be the outermost of such surfaces.

By assuming that the apparent horizon is topologically equivalent to a 2-sphere and is a star-shaped surface around an interior local coordinate origin[2], we use standard spherical coordinates (r, θ, ϕ) to parameterize it. The apparent horizon surface is represented by a horizon function $h(\theta, \phi)$ which measures the radial coordinate distance of the surface from the local origin. The aforementioned outward-pointing normal s^i can be constructed using the gradient of a level-set function $F = r - h(\theta, \phi)$, namely

$$s_i = \lambda m_i \equiv \lambda \partial_i F = \lambda \left(1, -\partial_\theta h, -\partial_\phi h\right), \tag{4}$$

where λ is the normalization factor such that $s_i s^i = 1$. Using such parameterization, (3) can be cast into an elliptic PDE in terms of h.

2.1.2. The ansatz adopted.

Following [28], we separate out a linear elliptic operator from the non-linear expansion equation $\Theta = 0$, such that it is in a suitable form to be solved by standard relaxation method. We first consider

$$\Delta_{\theta\phi} h - 2h = \Delta_{\theta\phi} h - 2h + \rho\Theta, \tag{5}$$

where $\rho = \rho(h, \theta, \phi)$ is a scalar function to be determined and $\Delta_{\theta\phi}$ is the flat-space Laplacian on a 2-sphere which is given by

$$\Delta_{\theta\phi} h = \partial_\theta^2 h + \cot\theta\, \partial_\theta h + \frac{1}{\sin^2\theta} \partial_\phi^2 h. \tag{6}$$

We briefly discuss how the Laplacian shows up in the expansion equation and how the scalar function ρ should be chosen.

A 3-metric tensor $\gamma_{ij}(r, \theta, \phi)$ on a spacelike hypersurface Σ_t can be expressed by its conformally related metric $\bar{\gamma}_{ij}$ and be further decomposed into the flat metric η_{ij} along with a tensor field h_{ij}, i.e.

$$\gamma^{ij} = \psi^{-4} \bar{\gamma}^{ij} = \psi^{-4} \left(\eta^{ij} + h^{ij}\right), \tag{7}$$

where the conformal factor ψ is given by

$$\psi = \left(\frac{\det \gamma_{ij}}{\det \eta_{ij}}\right)^{1/12}. \tag{8}$$

By introducing the covariant derivative $\widetilde{D}_i$ associated with the flat metric tensor η_{ij} and the corresponding connection coefficients $\widetilde{\Gamma}^k_{ij}$, the divergence term in the expansion function (3) is

$$\begin{aligned} D_i s^i &= m^{ij} D_i \left(\lambda m_j\right) = \lambda m^{ij} D_i m_j \qquad (9) \\ &= \lambda \psi^{-4} \eta^{ij} \widetilde{D}_i m_j + \lambda \left(\psi^{-4} h^{ij} - s^i s^j\right) \partial_i m_j - \lambda \left(\gamma^{ij} - s^i s^j\right) \Gamma^k_{ij} m_k \\ &\quad + \lambda \psi^{-4} \eta^{ij} \widetilde{\Gamma}^k_{ij} m_k, \qquad (10) \end{aligned}$$

[2] We refer the reader to [33] for the removal of the star-shaped assumption in the axisymmetric case.

where Γ^k_{ij} are the connection coefficients associated with the 3-metric γ_{ij}. One can see that the Laplacian, i.e. the linear elliptic part we are looking for, appears in the first term. In spherical coordinates, it reads

$$\eta^{ij}\widetilde{D}_i m_j = -\frac{1}{h^2}\left(\Delta_{\theta\phi}h - 2h\right). \tag{11}$$

The expansion function now becomes

$$\sqrt{2}\Theta = D_i s^i - \left(\gamma^{ij} - s^i s^j\right) K_{ij} \tag{12}$$

$$= -\frac{\lambda\psi^{-4}}{h^2}\left(\Delta_{\theta\phi}h - 2h\right) + \lambda\left(\psi^{-4}h^{ij} - s^i s^j\right)\partial_i m_j$$
$$- \left(\gamma^{ij} - s^i s^j\right)\left(\lambda\Gamma^k_{ij}m_k + K_{ij}\right) + \lambda\psi^{-4}\eta^{ij}\widetilde{\Gamma}^k_{ij}m_k. \tag{13}$$

Comparing with (5), the combination $\Delta_{\theta\phi}h - 2h$ on the right hand side cancels out if we choose $\rho = \sqrt{2}h^2\psi^4/\lambda$, yielding a PDE in suitable form to be solved by relaxation schemes. We make an additional remark that from (10), the linear elliptic operator in the expansion equation is indeed

$$h^2\eta^{ij}\left(\widetilde{D}_i m_j + \widetilde{\Gamma}^k_{ij}m_k\right) = -\left(\partial^2_\theta h + \frac{1}{\sin^2\theta}\partial^2_\phi h\right), \tag{14}$$

but it turns out that such form is not suitable for relaxation methods to work properly.

Finally, the ansatz we adopt in this work in explicit form is given by

$$\Delta_{\theta\phi}h - (2-\eta)h = S\left(h, \partial_i h, \partial^2 h; \gamma_{ij}, \partial_k\gamma_{ij}, K_{ij}; \eta\right), \tag{15}$$

with the source term S being

$$S = \Delta_{\theta\phi}h - (2-\eta)h + \frac{h^2\psi^4}{\lambda}\left(\sqrt{2}\Theta\right) \tag{16}$$

$$= h^2\psi^4\left[\left(\psi^{-4}h^{ij} - s^i s^j\right)\partial_i m_j - \left(\gamma^{ij} - s^i s^j\right)\left(\Gamma^k_{ij}m_k + \frac{K_{ij}}{\lambda}\right)\right]$$
$$+ h^2\eta^{ij}\widetilde{\Gamma}^k_{ij}m_k + \eta h, \tag{17}$$

where the extra parameter η can speed up convergence in the relaxation method if appropriately chosen [39]. Our experience with the benchmark tests (see section 3) suggests that the optimal value of η can only be found empirically, but in general it should take a value between 0 and 1.

2.2. Implementation of multigrid apparent horizon finder

2.2.1. Multigrid method. Iterative schemes have been a standard tool for solving general elliptic PDEs numerically. For iterative schemes that use a finite difference discretization setup, there is sometimes a trade-off between choosing a grid of higher resolution for more accurate results and a grid of relatively lower resolution for faster convergence. The persistent low-frequency components in the numerical error is one of the factors that reduce the convergent rate, but they can be eliminated rather efficiently at lower resolutions.

A multigrid scheme combines multiple levels of grid with different resolutions to tackle low-frequency components in the error in the coarser grids while retaining good accuracy of the solution in the finest grid. The major components of a multigrid scheme are as follows. *Smoothers* (e.g. standard relaxations) are applied on each but the coarsest level to eliminate error modes at different spectral range, and a *solver* is employed at the coarsest level at which an exact/approximate solution can be found more efficiently. With multiple levels of grid, the inter-grid data transfer operators—*restrictions* and *prolongations*—are there to bridge successive grid levels by transferring the current solutions and/or error corrections between them. Finally, a *cycling scheme* is chosen to specify the exact scheduling, when to jump between different levels of grid, of a multigrid solver. We refer the reader to [13, 24, 44] for detailed discussion on the multigrid methods.

2.2.2. *Linear multigrid algorithm.*

Owing to the fact that the principle Laplacian in (15) is linear, we choose to use a linear multigrid algorithm, which will be briefly outlined in the following.

Suppose we solve a linear elliptic equation $\mathcal{L}(u) = f$ on a uniform grid of spacing k, and we write

$$\mathcal{L}_k(u_k) = f_k, \tag{18}$$

where $\mathcal{L}$ is the elliptic operator, f is the source term and u is the exact solution. Let $\widetilde{u}_k$ denote the intermediate approximate solution, and $e_k = u_k - \widetilde{u}_k$ denote the corresponding error. Since $\mathcal{L}$ is linear, (18) becomes

$$\mathcal{L}_k(e_k) = f_k - \mathcal{L}_k(\widetilde{u}_k) \equiv r_k, \tag{19}$$

where r_k is called the *residual*. This residual equation is easier to solve on a coarser grid of spacing $2k$ if we consider an approximation version of it, i.e.

$$\mathcal{L}_{2k}(e_{2k}) = r_{2k}, \tag{20}$$

where the coarser grid residual r_{2k} is found by restricting the finer grid residual r_k using the restriction operator $\mathcal{R}$, i.e. $r_{2k} = \mathcal{R}(r_k)$. The solution to (20) can be thought as the correction to the approximate solution $\widetilde{u}_k$ we have for the original problem (18). Now denote the approximate solution to (20) by $\widetilde{e}_{2k}$, we can interpolate it to the finer grid by the prolongation operator $\mathcal{P}$ such that the approximate solution $\widetilde{u}_k$ is updated by

$$\widetilde{u}_k^{\text{new}} = \widetilde{u}_k + \mathcal{P}(\widetilde{e}_{2k}). \tag{21}$$

The above procedure is an illustration on a 2-grid structure and can be easily generalized for an n-grid solver. In particular, coarser grids can be recursively constructed when solving for the correction in (20).

2.2.3. *Grid discretization.*

We discretize the apparent horizon surface $h(\theta, \phi)$ using a *vertex-centered grid*. At the finest grid level, there are N_θ points along the polar direction and N_ϕ points along the azimuthal direction. The grid points sit at

$$\theta_i = i\Delta\theta = i\left(\frac{\pi}{N_\theta - 1}\right), \qquad i = 0, \ldots, N_\theta - 1; \tag{22}$$

$$\phi_j = j\Delta\phi = j\left(\frac{2\pi}{N_\phi - 1}\right), \qquad j = 0, \ldots, N_\phi - 1. \tag{23}$$

Note that we require the numbers of grid points in the finest grid level, i.e. N_θ and N_ϕ, to be odd number, but this condition can be relaxed for the subsequent coarser grid levels. We further discretize the apparent horizon equation (15) using a 4th order central finite difference representation for the Laplacian such that it is in a suitable form for conventional relaxation methods.

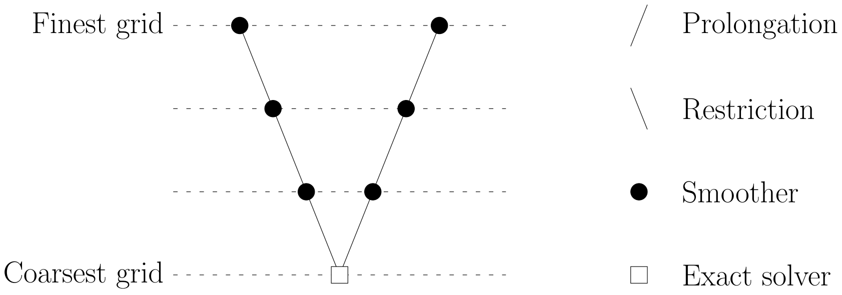

Figure 1. The structure of one V-cycle with 4 grid levels. The operators are applied in chronological order from left to right. In our code, we use the smoother instead of an exact solver at the coarsest grid as well.

2.2.4. Boundary condition. Special treatments are carried out to the boundary points of this vertex-centered spherical grid. In the azimuthal direction, periodic boundary conditions are applied. The points at $\phi_j = 2\pi$ are ghost points and their values are set by $h(\theta_i, 2\pi) = h(\theta_i, 0)$. The boundary conditions in the polar direction are a bit more tricky because both the Laplacian $\Delta_{\theta\phi}$ and the expansion function Θ exhibit singularity at the poles. The polar singularities are avoided by *not* updating the horizon function h with (15) there. Instead, we first interpolate one value of h at the (north) pole, say at $(\theta = \theta_0 = 0, \phi = \phi_k)$ for some k, through a cubic polynomial constructed using the points $h(\theta_{-2}, \phi_k)$, $h(\theta_{-1}, \phi_k)$, $h(\theta_1, \phi_k)$ and $h(\theta_2, \phi_k)$, where the negative indices of θ represent points on the opposite side across the (north) pole. This value of $h(0, \phi_k)$ is then copied to all other points at $\theta = 0$, that is, all $h(0, \phi_j)$ are forced to be the same for each j. In practice, the value of k does not matter much and we have set $k = 0$, i.e. $\phi_k = 0$, for convenience. The same procedure is done at the south pole to ensure the smoothness of the surface across both poles.

2.2.5. Smoothers and inter-grid transfer operators. In the multigrid algorithm, we need to specify a way to obtain approximate solutions to (18) and (20). We use the Gauss–Seidel line relaxation method [36] in the ϕ-direction as a smoother (see section 2.3 for more details) at all grid levels including the coarsest level. The smoothing operator applied before restrictions and after prolongations consists of 1 and 2 sweeps of relaxation, respectively. At the coarsest level, the smoothing operator consists of 100 sweeps of relaxation.

For the inter-grid transfer operators, we choose the bi-linear prolongation operator $\mathcal{P}$ and the full-weight restriction operator $\mathcal{R}$. These operators can be represented in the stencil notations (see, e.g. [45]) by

$$\mathcal{P} = \frac{1}{4}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}, \qquad \mathcal{R} = \frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}. \tag{24}$$

2.2.6. Cycling algorithm and the solving procedure. Among different multigrid cycling algorithms, we choose to use the V-cycle (see figure 1) owing to its simplicity. Successive V-cycles are carried out until some prescribed tolerance is satisfied to locate the apparent horizon.

Assuming that the geometric objects γ_{ij} and K_{ij} are given on a hypersurface, an initial trial surface $h^{(0)}$ is specified for the calculation of the source term S according to (17). Note that we do not need the value of S at the poles since these points are excluded from the relaxation domain. One V-cycle is then performed to obtain a new guess surface $h^{(1)}$ while keeping the source term fixed. The new surface $h^{(1)}$ is used to update the source term S before entering the next V-cycle. This process is repeated until the maximum change in h between successive trial surfaces, denoted by $\|\delta h\|_\infty$, and/or the maximum value of the expansion on the final guess surface, denoted by $\|\Theta\|_\infty$, are less than some tolerance ϵ_h and ϵ_Θ, respectively, within the relaxation domain (i.e. excluding the poles).

2.2.7. Handling of numerical spacetimes. The ability to process numerical spacetime is important for an apparent horizon finder to be applicable in numerical simulations, where the geometric objects γ_{ij} and K_{ij} are given only at discrete locations. To update the source term (17) between each V-cycle with numerical spacetime data, we need the values (in spherical basis) of γ_{ij}, K_{ij} and also $\partial_k\gamma_{ij}$ on the current trial surface $h(\theta,\phi)$. The current implementation of the interpolator in our finder takes input of γ_{ij} and K_{ij} on a uniform Cartesian grid, interpolates γ_{ij}, $\partial_k\gamma_{ij}$ and K_{ij} in Cartesian basis onto the trial surface using tricubic Hermite spline, and finally transforms the components from Cartesian basis to spherical polar basis. In approximating the first to third order derivatives of γ_{ij} and K_{ij} in Cartesian basis which are required for tricubic Hermite interpolation, a standard 4th order finite differencing scheme is used to maintain an overall third order accuracy in the interpolator.

2.3. Line relaxation smoothing operator

As mentioned in the previous section, we have chosen the Gauss–Seidel line relaxation in the azimuthal direction as the smoothing operator in our multigrid code. Due to the inherent asymmetry between the polar and azimuthal directions of the spherical Laplacian $\Delta_{\theta\phi}$, the line relaxation has better convergence, especially near the poles, than a simple point-wise relaxation with or without red-black ordering updates [6]. Because of the periodic boundary condition in the azimuthal direction, one step of the ϕ-line relaxation requires solving a cyclic banded diagonal system [36]. In particular, a cyclic penta-diagonal system is needed to be solved for each relaxation step if we use a standard 4th order 5-point finite difference formula. This obviously requires much more work than solving a tri-diagonal system, which is the case with a 2nd order 3-point finite difference scheme. In order to take advantage of the efficiency in solving tri-diagonal system while achieving 4th order accuracy, we aim for a compact finite difference scheme for (15) (see e.g. [46] and [15] respectively for such treatment in Cartesian Poisson equation and polar Helmholtz equation).

The compact finite difference scheme, or the Mehrstellenverfahren discretization (e.g. [44]) is not a new concept and can be traced back to as early as the 1960s [18], which was then referred to as the Hermitian method. It aims at increasing the accuracy of a finite difference scheme without involving more grid points that are *too* far away from the current pivoting point. As an example, for the Poisson equation $(\partial_x^2+\partial_y^2)u=f$ on a square Cartesian grid of step size k, the traditional 4th order (non-compact) finite difference scheme involves the second next neighboring grids, as represented by the stencil notation

$$\frac{1}{12k^2}\begin{bmatrix} & & -1 & & \\ & & 16 & & \\ -1 & 16 & -60 & 16 & -1 \\ & & 16 & & \\ & & -1 & & \end{bmatrix} u_{ij} = f_{ij},$$

while its 4th order compact finite difference representation is [18]

$$\frac{1}{6k^2}\begin{bmatrix} 1 & 4 & 1 \\ 4 & -20 & 4 \\ 1 & 4 & 1 \end{bmatrix} u_{ij} = \frac{1}{12}\begin{bmatrix} 0 & 1 & 0 \\ 1 & 8 & 1 \\ 0 & 1 & 0 \end{bmatrix} f_{ij}.$$

Notice the smaller stencil size (width) in the compact finite difference scheme. This compact representation can be viewed as the incorporation of a correction to the 2nd order finite difference formula to eliminate the leading error in its Taylor's expansion (see, e.g. [7, 15, 20, 23, 27, 46] for a variety of PDEs that such method has been applied to). The compact form for the spherical Poisson equation, to which the apparent horizon equation (15) belongs, can be derived in a similar manner. We refer the reader to [20] for a summary of compact scheme for Poisson equation on Cartesian grids.

Since our ultimate goal is to create a stencil that is suitable for reducing the ϕ-line relaxation to solving a tri-diagonal system, we only need to 'squeeze' the usual 5-point-wide stencil in the ϕ direction, which we refer to as *semi-compact*. We derive such *semi-compact* stencil for a general 2-dimensional spherical Poisson equation in the following[3]. Suppose we solve

$$\partial_\theta^2 u + \cot\theta\,\partial_\theta u + \frac{1}{\sin^2\theta}\partial_\phi^2 u = f(\theta,\phi)\,. \tag{25}$$

First observe that using a 2nd order finite difference method for the third term, we have

$$\frac{1}{\sin^2\theta}\frac{\partial^2 u}{\partial\phi^2} = \left(\frac{1}{\sin^2\theta}\frac{\partial^2 u}{\partial\phi^2}\right)_{\mathrm{FD2}} - \frac{k^2}{12\sin^2\theta}\frac{\partial^4 u}{\partial\phi^4} + \mathcal{O}\left(k^4\right), \tag{26}$$

where the subscript FDn indicates that the corresponding term is approximated by an nth order finite difference scheme, and k is the step size in the ϕ direction. By differentiating (25) twice w.r.t. ϕ and substituting the result to (26), we obtain

$$\frac{1}{\sin^2\theta}\frac{\partial^2 u}{\partial\phi^2} = \left(\frac{1}{\sin^2\theta}\frac{\partial^2 u}{\partial\phi^2}\right)_{\mathrm{FD2}} - \frac{k^2}{12}\left[\frac{\partial^2 f}{\partial\phi^2} - \frac{\partial^4 u}{\partial\theta^2\partial\phi^2} - \cot\theta\frac{\partial^3 u}{\partial\theta\partial\phi^2}\right] + \mathcal{O}\left(k^4\right). \tag{27}$$

With a factor of k^2 in front, the terms inside the square bracket only need to be evaluated at 2nd order accuracy to maintain an overall 4th order accuracy. Finally, assuming that the step size in θ direction is also k for simplicity, the *semi-compact* 4th order finite difference representation of (25) can be given by

[3] For completeness, we also derive the *fully*-compact representation in appendix for a general 2-dimensional Poisson equation on the sphere. We do not use the *fully*-compact form because it requires the value of the source term f at the poles, but the apparent horizon equation (15) contains the expansion function Θ which is undefined at the poles in spherical coordinates (see (13)).

Table 1. Comparison of the efficiency between using point-wise relaxation and line-relaxation as the smoothing operator to locate the apparent horizon for the analytic Kerr–Schild spacetime of a black hole with mass $M = 1$ and spin parameter $a = 0.8$. The columns represent the resolution, number of grid levels n, and the number of V-cycle and user CPU time (in seconds) required to reach the solution for both relaxation methods. The tolerance parameters are set to $\epsilon_h = \epsilon_\Theta = 10^{-8}$. The results in the rows marked with (without) an asterisk are obtained using $\eta = 0.8$ (0). For the pointwise relaxation, the number of relaxation sweeps before restrictions and after prolongation is changed to 10 and 20, and that at the coarsest level to 200. Both the results of pointwise relaxation using the non-compact (NC) and the semi-compact (SC) finite difference scheme are shown for comparison.

		Point relax. (NC)		Point relax. (SC)		Line relaxation	
$N_\theta \times N_\phi$	n-grid	V-cycle	Time(s)	V-cycle	Time(s)	V-cycle	Time(s)
65×65	3	36	0.251	35	0.330	34	0.204
	3*	22	0.161	20	0.194	19	0.122
129×129	3	80	2.170	66	2.671	45	1.035
	4	64	1.652	53	1.794	34	0.735
	4*	67	1.743	54	1.822	20	0.446
257×257	3	318	33.502	254	37.065	128	11.537
	4	272	27.828	217	28.664	46	3.903
	4*	278	28.332	219	29.177	35	3.009
	5	260	25.884	205	26.693	34	2.857
	5*	257	26.624	204	26.529	21	1.814

$$\begin{aligned}
&\left(\frac{\partial^2 u}{\partial\theta^2} + \cot\theta \frac{\partial u}{\partial\theta}\right)_{\mathrm{FD4}} + \left(\frac{1}{\sin^2\theta}\frac{\partial^2 u}{\partial\phi^2}\right)_{\mathrm{FD2}} \\
&\quad - \frac{k^2}{12}\left[\frac{\partial^2 f}{\partial\phi^2} - \frac{\partial^4 u}{\partial\theta^2\partial\phi^2} - \cot\theta\frac{\partial^3 u}{\partial\theta\partial\phi^2}\right]_{\mathrm{FD2}} + \mathcal{O}\left(k^4\right) = f.
\end{aligned} \tag{28}$$

The adaption of this form to our apparent horizon equation (15) is straight-forward, despite the fact that the source term now also depends on the solution. Notice that the stencil is only 3-point-wide in the ϕ direction such that a ϕ-line relaxation can be done by a more efficient tri-diagonal algorithm.

We make some final remarks to conclude this section. First, a simple point-wise Gauss–Seidel relaxation (using either 5-point-wide non-compact or semi-compact finite difference scheme) can certainly serve as the smoothing operator, but the efficiency of the solver will be much degraded (see table 1 and the discussion in section 3.1). It is worth noting that although using semi-compact scheme gives a slightly better convergence (in terms of the number of V-cycles) than using the usual non-compact scheme, the computational costs of the two are roughly the same due to the more arithmetic operations needed in the semi-compact scheme. Second, the asymmetry in the spherical Laplacian may change direction from the pole to the equator depending on the grid sizes in the θ- and ϕ-directions [6]. In the case when the number of points in the θ-direction is much larger than that in the ϕ-direction, the ϕ-line relaxation may not provide good convergence, especially near the equator, or it may even fail. A remedy is to use a combined relaxation [6], sometimes referred to as the segment relaxation scheme, which applies the θ- and ϕ-line relaxations to different domains according to the direction of asymmetry in the Laplacian. Although this could be relevant for further increasing the efficiency of our finder when applied to spacetimes with axisymmetry, we will continue to use the ϕ-line relaxation here.

3. Results

We have tested our multigrid apparent horizon finder with several benchmark spacetimes and the results are reported in this section. All tests are run on a 3.6 GHz processor and the timings are reported in user CPU time.

The geometric objects γ_{ij} and K_{ij} are given analytically in spherical coordinates for the calculation of the source term S (17), except for numerical data where they are interpolated onto the surface. Unless otherwise specified, the initial guess surface is a sphere of coordinate radius $1.5M$, where M is the mass of the black hole. The tolerance to stop the iteration process is set to $\epsilon_h = \epsilon_\Theta = 10^{-8}$, and the convergence parameter is set to $\eta = 0.8$. After locating the apparent horizon, its proper area is calculated numerically by [28]

$$A = \int_0^{2\pi} \mathrm{d}\phi \int_0^{\pi} \mathrm{d}\theta \left[\left(\gamma_{rr} h_{,\theta}^2 + 2\gamma_{r\theta} h_{,\theta} + \gamma_{\theta\theta}\right) \left(\gamma_{rr} h_{,\phi}^2 + 2\gamma_{r\phi} h_{,\phi} + \gamma_{\phi\phi}\right) \right.$$
$$\left. - \left(\gamma_{rr} h_{,\theta} h_{,\phi} + \gamma_{r\phi} h_{,\theta} + \gamma_{r\theta} h_{,\phi} + \gamma_{\theta\phi}\right)^2 \right]^{1/2}. \tag{29}$$

3.1. Kerr–Schild spacetime

For describing a Kerr black hole of mass $M = 1$ and spin parameter a ($0 \leqslant a < 1$), one can use the Kerr–Schild coordinates [30], of which the spatial metric γ_{ij} and the extrinsic curvature K_{ij} are given in Cartesian coordinates by

$$\gamma_{ij} = \eta_{ij} + 2H\ell_i \ell_j; \tag{30}$$

$$K_{ij} = 2\alpha H \ell^k \partial_k \left(H\ell_i \ell_j\right) + \alpha \left[\partial_i \left(H\ell_j\right) + \partial_j \left(H\ell_i\right)\right], \tag{31}$$

where $\eta_{ab} = \mathrm{diag}(-1,1,1,1)$ is the flat metric and $\alpha \equiv \sqrt{1+2H}$ is the lapse function. The function H and the auxiliary components ℓ^i are defined by

$$H \equiv \frac{MR^3}{R^4 + a^2 z^2}; \qquad \ell_i = \ell^i \equiv \left(\frac{Rx + ay}{R^2 + a^2}, \frac{Ry - ax}{R^2 + a^2}, \frac{z}{R}\right), \tag{32}$$

with the parameter R being the Boyer–Lindquist radial coordinate which satisfies

$$\frac{x^2 + y^2}{R^2 + a^2} + \frac{z^2}{R^2} = 1. \tag{33}$$

The analytic forms of γ_{ij}, $\partial_k \gamma_{ij}$ and K_{ij}, after a coordinate transformation, serve as the input for the calculation of the source term S in (15). There are two horizons at $R = r_\pm \equiv M(1 \pm \sqrt{1 - a^2})$ and the apparent horizon is given by the outer one, which is, in spherical coordinates (r, θ, ϕ),

$$r^2 = \frac{r_+^2 \left(r_+^2 + a^2\right)}{r_+^2 + a^2 \cos^2 \theta}, \tag{34}$$

with the corresponding proper area given by

$$A_{\mathrm{Kerr}} = 4\pi \left(r_+^2 + a^2\right). \tag{35}$$

Table 2. Finding the apparent horizon in analytic Kerr–Schild spacetime of spin parameter $a = 0.6$ and 0.9. The columns represent the resolution $N_\theta \times N_\phi$ on the finest grid level, the number of grid levels n; the relative error in the proper area $\delta A/A_{\mathrm{Kerr}}$; the number of V-cycle and the user CPU time (in seconds) required to locate the apparent horizon. In the first column, the angle (e.g. 5°) inside the parenthesis in each row is approximately the angular separation between two neighboring grid points in the θ- or ϕ-direction.

		$a = 0.6$			$a = 0.9$		
$N_\theta \times N_\phi$	n-grid	$\lvert\delta A/A_{\mathrm{Kerr}}\rvert$	$\#V$	Time (s)	$\lvert\delta A/A_{\mathrm{Kerr}}\rvert$	$\#V$	Time (s)
$37 \times 73\,(5^\circ)$	4	2.8×10^{-6}	16	0.063	1.2×10^{-6}	25	0.096
$61 \times 121\,(3^\circ)$	4	3.6×10^{-7}	16	0.162	1.7×10^{-7}	26	0.249
$61 \times 121\,(3^\circ)$	5	3.6×10^{-7}	16	0.160	1.7×10^{-7}	25	0.243
$91 \times 181\,(2^\circ)$	4	7.2×10^{-8}	17	0.377	3.9×10^{-8}	26	0.562
$91 \times 181\,(2^\circ)$	5	7.2×10^{-8}	16	0.350	4.2×10^{-8}	25	0.525
$91 \times 181\,(2^\circ)$	6	7.2×10^{-8}	16	0.349	4.0×10^{-8}	24	0.510

Before considering the performance of our apparent horizon finder, we first use the analytic Kerr–Schild spacetime as a test case to show the advantage of using a line relaxation smoother over a point-wise one in a multigrid code as noted in section 2.3. As shown in table 1, the number of V-cycles required to reach the solution is less when a line relaxation is used. The difference becomes more significant with higher resolution and more grid levels. Combined with the enhancement given by tuning the convergence parameter η, the use of line relaxation over point relaxation leads to an order-of-magnitude overall boost in terms of speed in certain test cases.

We use several resolutions to locate the apparent horizon in Kerr–Schild spacetime with spin parameter $a = 0.6$ and 0.9. The results are reported in table 2. With the same number of grid levels, increasing the resolution slightly increases the number of iteration needed to reach convergence. This, however, can be compensated by introducing more grid levels. Overall, we find that the number of iterations (V-cycles) to reach convergence is more or less the same for a fixed a, independent of the resolutions. The run time reported here is only determined by the searching algorithm, excluding the interpolation routines required in numerical spacetimes (see section 3.3). We find that the run time is proportional to the total number of grid points on the finest grid level.

3.2. Brill–Lindquist spacetime

The second test for our finder is to locate the apparent horizon in Brill–Lindquist multiple black hole data [14]. This is a classic analytic solution for the constraint equations in 3+1 numerical relativity that represents a multiple black hole spacetime. The black holes are assumed to be momentarily at rest. The derivation of this solution can be found in standard numerical relativity textbooks (e.g. [9]). Under time symmetry and the assumption that the spatial slice is conformally flat, the 3-metric and the extrinsic curvature for N-black-hole spacetime in isotropic coordinates $(\bar{r}, \theta, \phi)$ are, respectively,

$$\gamma_{ij} = \left(1 + \sum_{\alpha=1}^{N} \frac{M_\alpha}{2|\bar{\boldsymbol{r}} - \bar{\boldsymbol{r}}_\alpha|}\right)^4 \mathrm{diag}\left(1, \bar{r}^2, \bar{r}^2 \sin^2\theta\right), \tag{36}$$

$$K_{ij} = 0, \tag{37}$$

Table 3. Finding the apparent horizon for analytic Brill–Lindquist equal-mass 2-black-hole spacetime at $d = 1.532 \approx d_{\text{crit}}$. While the spacetime is axisymmetric, we do not impose it in the tests.

$N_\theta \times N_\phi$	A	$\lVert\Theta\rVert_\infty$	Time (s)
$41 \times 73\,(5^\circ)$	196.4138	1.57×10^{-8}	1.22
$61 \times 121\,(3^\circ)$	196.4158	1.50×10^{-8}	1.93
$91 \times 181\,(2^\circ)$	196.4162	1.52×10^{-8}	4.16
$181 \times 361\,(1^\circ)$	196.4163	1.57×10^{-8}	16.20

where M_α and $\bar{\boldsymbol{r}}_\alpha$ are the mass at infinite separation and the coordinate position of the αth black hole, respectively. The isotropic coordinates here are the analog of those in the isotropic Schwarzschild metric, which can be obtained by substituting the radial coordinate in the Schwarzschild metric by $r = \bar{r}[1 + M/(2\bar{r})]^2$. In the following, we consider the case of $N = 2$ and $N = 3$. We use very high resolution of $N_\theta \times N_\phi = 385 \times 769$ (angular spacing $<0.5^\circ$) with 6 levels of grid in these tests without assuming any symmetry of the apparent horizon. We relax the convergence criteria in this section to $\epsilon_h = 10^{-8}$ and $\epsilon_\Theta = 10^{-6}$.

3.2.1. Brill–Lindquist 2-black-hole spacetime. Let us first consider the Brill–Lindquist data for two black holes. When the two black holes are far away from each other, each black hole possesses its own apparent horizon. With decreasing separation distance d, there exists a critical separation $d = d_{\text{crit}}$ at which the two black holes start to have a common apparent horizon. One of the benchmark tests is to determine the value of d_{crit}. The center of the two black holes are on the z-axis at $(0, 0, \pm d/2)$ in the following.

We first consider two equal-mass black holes with $M_1 = M_2 = 1$. Our finder reports the critical separation to be $d_{\text{crit}} = 1.532\,3948$, which agrees well with the generally accepted values in the literature (e.g. 1.532 3949 [42], 1.532 [28]). The critical area is found to be $A_{\text{crit}} = 196.407\,95$ (agrees with [42] to 5 decimal places). At the critical separation we found, the maximum expansion on the numerically-computed apparent horizon is $\lVert\Theta\rVert_\infty = 2.3 \times 10^{-8}$. We also report the results for finding the apparent horizon near the critical separation at $d = 1.532 \approx d_{\text{crit}}$ for lower resolutions in table 3.

We then consider unequal-mass black holes with $M_1 = 0.2$ and $M_2 = 0.8$. The initial guess surface for this case is changed to a sphere of coordinate radius $\bar{r} = 1.5$. Our finder reports the critical separation to be $d_{\text{crit}} = 0.698\,7161$ with the proper area being $A = 49.688\,602$. The maximum expansion on the numerically-computed apparent horizon is $\lVert\Theta\rVert_\infty = 2.1 \times 10^{-7}$. The value of the critical separation agrees with that reported recently in [33] by using an axisymmetric apparent horizon finder.

3.2.2. Brill–Lindquist 3-black-hole spacetime. We now consider a system of three black holes which are placed to form an equilateral triangle on the $\bar{x}$–$\bar{z}$ plane, each at a coordinate distance R from the origin, and one of them sits on the $+\bar{z}$-axis. Each black hole has unity mass. We find that the critical parameter for the three black holes to have a common apparent horizon is $R_{\text{crit}} = 1.195\,4995$, which agrees with the result in [42] to seven decimal places. The critical area and the maximum expansion on the numerically-computed apparent horizon are $A = 444.756\,23$ and $\lVert\Theta\rVert_\infty = 2.3 \times 10^{-8}$, respectively. The left panel of figure 2 shows the cross section (red dashed line) of the common horizon on the $\bar{x}$–$\bar{z}$ plane for the case $R = 1.195 \approx R_{\text{crit}}$. The right panel shows the side view of the full surface of the common horizon.

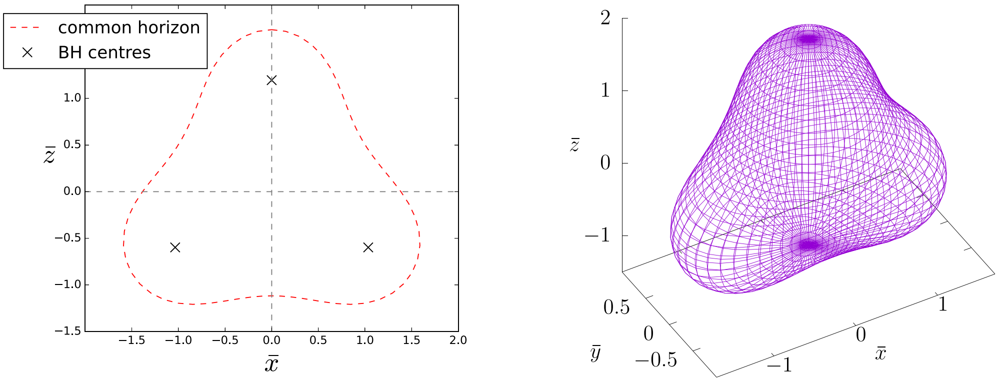

Figure 2. The apparent horizon for Brill–Lindquist equal-mass 3-blackhole spacetime at $R = 1.195 \approx R_{\text{crit}}$. The left panel shows the cross-section (red dashed line) of the common horizon on the $\bar{x}$–$\bar{z}$ plane; the right panel is the side view of the full surface of the common horizon.

3.3. *Numerical spacetimes*

We prepare numerical spacetimes to further test our finder using the open source code EINSTEIN TOOLKIT [29]. The simulation variables γ_{ij} and K_{ij} are post-processed to produce data on a uniform rectangular Cartesian grid before importing to our finder. In the case of dynamical binary black hole simulation where adaptive mesh refinement is turned on, we use the python package KUIBIT [10] for the post-processing. We also activate the inherent apparent horizon finder in the code, AHFINDERDIRECT [41, 42], for comparison.

3.3.1. Numerical Kerr–Schild spacetime. We first test our finder with an off-centered Kerr–Schild data. The numerical spacetime data spans the Cartesian grid space of $x, y, z \in [-2.5, 2.5]$ with a grid spacing of $\Delta x = \Delta y = \Delta z = 0.05$. We use a sphere of coordinate radius 1.5 centered at the coordinate origin as the initial guess surface. We use the ILUCG matrix routine in all cases for AHFINDERDIRECT.

We consider a black hole of unity mass and spin a located at $(0.2, 0.2, 0.2)$ in Cartesian coordinates. We tabulate the results for $a = 0.6$ and $a = 0.8$ in table 4. The time to locate the solution surface mainly depends on two factors—the overall convergence of the algorithm and the computational cost within each iteration. The convergence order of the algorithm is important in the sense that as many as 30 interpolations for the geometric variables are required at every grid point at each (outer) iteration. Since we adopt a pure multigrid algorithm, which has a slower convergence than the quadratic convergence in Newton's method, one can see that our finder needs more iterations (number of V-cycles) for convergence, so more time is spent on the interpolation routines when compared to AHFINDERDIRECT. Notice that the number of iterations is insensitive to the resolution used for fixed a. Had we used the pointwise Gauss–Seidel relaxation as the smoothing operator, the number of iterations would grow linearly with the number of grid points, and the speed of the code would become very slow due to the large amount of interpolations required.

At high resolutions, however, we find that our finder generally locates the apparent horizon faster than AHFINDERDIRECT. This is somewhat expected because, at a rough estimation,

Table 4. Finding the apparent horizon in numerical off-centered Kerr–Schild data with $a = 0.6$ (top) and $a = 0.8$ (bottom). The first column represents the resolution to describe the horizon surface. The following four columns represent the number of grid levels n, the relative error in the proper area $\delta A/A_{\rm Kerr}$, the number of V-cycle and the time (in seconds) required to locate the apparent horizon using our code. The last two columns represent the number of Newton steps and time (in seconds) required for AHFinderDirect. See text for the numerical setup of the spacetime.

$a = 0.6$	MGAHF				AHFinderDirect	
Resolution	n-grid	$\lvert\delta A/A_{\rm Kerr}\rvert$	#V	Time (s)	#Newton	Time (s)
$37 \times 73\,(5^\circ)$	4	2×10^{-6}	18	2.635	8	1.604
$61 \times 121\,(3^\circ)$	5	2×10^{-7}	17	6.294	8	4.180
$91 \times 181\,(2^\circ)$	6	1×10^{-8}	17	12.387	8	11.393
$181 \times 361\,(1^\circ)$	7	1×10^{-8}	18	33.802	8	88.632
$a = 0.8$	MGAHF				AHFinderDirect	
Resolution	n-grid	$\lvert\delta A/A_{\rm Kerr}\rvert$	#V	Time (s)	#Newton	Time (s)
$37 \times 73\,(5^\circ)$	4	2×10^{-6}	23	3.251	8	1.628
$61 \times 121\,(3^\circ)$	5	3×10^{-7}	22	7.864	8	4.170
$91 \times 181\,(2^\circ)$	6	4×10^{-8}	21	14.461	8	11.009
$181 \times 361\,(1^\circ)$	7	1×10^{-7}	21	37.178	8	87.173

the operation cost of performing the matrix LU decomposition of the Jacobian equation in AHFinderDirect is $\mathcal{O}(N^3)$ but that of one multigrid cycle is $\mathcal{O}(N)$, where N is the total number of grid points used to describe the horizon surface and $N{=}N_\theta \times N_\phi$ in our case. The total search time is a competition between the two factors mentioned above. The computational cost of each iteration becomes more important with increasing resolutions. Although the overall convergence rate of our multigrid algorithm is not as fast as Newton's method, there is a critical resolution at which our multigrid algorithm starts to work faster[4]. In particular, our finder can be more than two times faster when the angular separation between two neighboring grids is about 1° as shown in table 4. We have performed the same test for additional values of the spin parameter a at 1° resolution, and the runtimes are plotted in figure 3. Although the runtime of our finder grows with a, it takes lesser time to converge in all cases. Recall that the performance of our finder has extra dependence on the convergence parameter η. At the extreme case of $a = 0.99$, the optimal value of η shifts to around 1.45, and the runtime is reduced by half comparing to the case $\eta = 0.8$.

We further test on the case where the spin parameter is $a = 0.99$, but with the black hole relocated to $(0.2, 0.2, 0)$. The initial guesses are spherical surfaces centered at the coordinate origin as before, with radius r_0 ranging from 1.5 to 1.6 with increments of 0.01. We found that AHFinderDirect is more volatile to the initial guess surface, and is able to find the apparent horizon only at $r_0 = 1.54, 1.57, 1.58$ and 1.6 (it fails to converge otherwise). On the other hand, our finder is able to find the correct solution for all these cases.

[4] There is a variant multigrid algorithm, namely the Newton-multigrid method [13], that deals with nonlinear equations directly. This method uses an outer Newton step as in AHFinderDirect, but solves the inner Jacobi equation with linear multigrid methods instead. We expect that this should (theoretically) outperform both the algorithms used in AHFinderDirect and our work. We have not used the Newton-multigrid here not only because it requires the manipulation of the Jacobian matrix, but also that it is not directly compatible with the compact finite difference scheme.

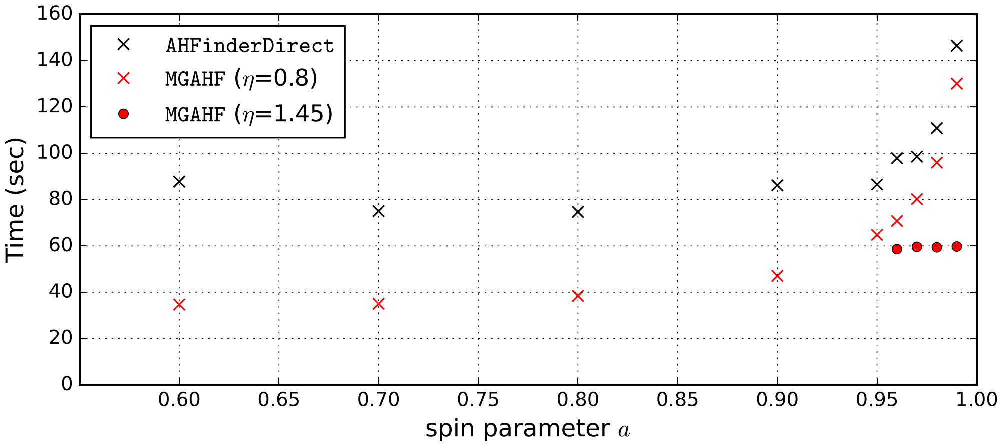

Figure 3. Comparison between our finder and AHFinderDirect on the runtime to locate the apparent horizon in numerical off-centered Kerr–Schild data with different values of spin parameter a. A spherical surface of radius 1.6 centered at the coordinate origin is used as the initial guess. The angular resolution for the runs is 1°. The rightmost 4 sets of data points correspond to $a = 0.96, 0.97, 0.98, 0.99$.

Table 5. Comparison between our finder and AHFinderDirect on the proper areas of the individual horizons and the common apparent horizon found in merging binary black hole spacetime. The relative difference is reported in the last column.

	MGAHF	AHFinderDirect	$\lvert\delta A/A\rvert$
Horizon of m_1	19.0239	19.0097	7×10^{-4}
Horizon of m_2	9.4808	9.4671	1×10^{-3}
Common apparent horizon	39.8666	39.8691	6×10^{-5}

3.3.2. Dynamical binary black hole spacetime. We simulate the merger of an unequal-mass black hole binary using Einstein Toolkit [29]. The initial mass ratio of the two non-spinning black holes is set to be $q \simeq 0.7$. We use standard evolution schemes and gauge conditions, with multiple layers of mesh refinements to perform the simulation. The finest grids of grid spacing $\Delta x = \Delta y = \Delta z = 0.031\,25$ cover the two black holes at all times. We export the numerical data to a uniform grid of the finest grid spacing by Kuibit [10] to search for the horizons using our finder. For the two individual horizons, the local coordinate origin is chosen by the position of singularity provided by the PunctureTracker thorn in the simulation code, while that of the common apparent horizon is simply set to be the origin of the simulation grid.

We report the results of horizon finding at the simulation time when the common apparent horizon first appears in table 5 and the corresponding snapshot of the three horizon surfaces in figure 4. The horizon surfaces are modeled by 5° angular resolution ($N_\theta \times N_\phi = 37 \times 73$). In figure 4, the common apparent horizon is represented by the purple surface, inside of which are the two individual horizons represented by the green and blue surfaces. In terms of the proper areas, our finder agrees with the default AHFinderDirect in the simulation code to within 0.2%. The difference is the biggest for the horizon of the smaller black hole, as the spherical grids become overly stacked at that radius.

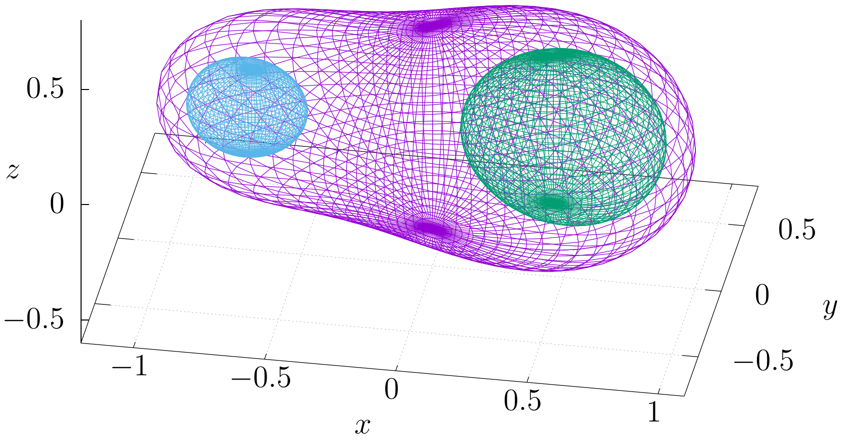

Figure 4. The common apparent horizon and the individual horizons found by our finder in a dynamical spacetime of merging unequal-mass black hole binary at a time when the common apparent horizon first appears.

4. Conclusions

We have presented a new apparent horizon searching algorithm using a multigrid method in this work. We have evaluated the performance of our finder on analytic spacetimes, including the Brill–Lindquist data with two or three black holes. We have recovered the results of the critical parameter for a common apparent horizon to appear in these spacetimes. We have also tested our finder on numerical spacetimes, namely the off-centered Kerr–Schild data and the inspiraling binary black hole spacetime. We have demonstrated that, in terms of speed, the multigrid searching algorithm begins to outperform the currently fastest algorithm (Thornburg's AHFinderDirect [41, 42]) at high resolutions. We have also shown that our finder is capable of capturing the first appearance of the common apparent horizon in merging black hole binary simulation.

From a numerical perspective of viewing our finder as a general multigrid solver for Poisson equations on the 2-sphere, there are some notable aspects. We have verified that the line relaxation scheme is preferable (in terms of convergence) to a pointwise relaxation scheme, even when dealing with a solution-dependent nonlinear source term. We have derived the (semi-)compact 4th order finite difference scheme for solving the spherical Poisson equation. The derivation follows a similar procedure used for Cartesian grids (see [20] and references therein). Our study demonstrates the applicability of this scheme in reducing the complexity of a line relaxation technique on spherical coordinates, from a penta-diagonal matrix inversion to a more efficient tri-diagonal matrix inversion. We believe the formula derived should be directly applicable to other 2-dimensional Poisson equations on the sphere.

We make some final remarks on how our multigrid algorithm can be further improved. First, the segment relaxation scheme (see section 2.3) could be used to improve the convergence. We only use ϕ-line relaxation as the smoother in our multigrid solver, but the line relaxation direction is preferred to be aligned with the anisotropy direction in the differential operator as it could give better convergence [6]. Second, using a variant multigrid algorithm might also improve the convergence. We solve the non-linear apparent horizon equation by turning it into a linear elliptic equation with a non-linear source term. In this way, the nonlinearity of the principle equation is avoided. However, by not addressing it directly, the rate of convergence could be compromised. The common multigrid methods to tackle nonlinearity directly are the full approximation scheme [11] or Newton-multigrid [13] algorithm. We have not adopted these methods in the current work since they in principle bring up technical implementation issues in the case of the apparent horizon equation. Exploring these alternative multigrid schemes could be one future investigation direction for increasing the efficiency of our multigrid code.

Data availability statement

No new data were created or analyzed in this study.

Acknowledgments

We thank Kenneth Chen and Yun-Kau Lau for useful discussions. L M L is also grateful for the hospitality of the Morningside Center of Mathematics, Chinese Academy of Sciences, where the idea of this project was initiated. This work is partially supported by a grant from the Research Grants Council of the Hong Kong SAR, China [Project No. CUHK 14304322]

Appendix. *Fully*-compact 4th order finite difference scheme for spherical Poisson equations

Suppose we solve a general Poisson equation in spherical coordinates,

$$\left(\frac{\partial^2}{\partial\theta^2}+\cot\theta\frac{\partial}{\partial\theta}+\frac{1}{\sin^2\theta}\frac{\partial^2}{\partial\theta^2}\right)u\left(\theta,\phi\right)=f(\theta,\phi)\,, \tag{A.1}$$

on a spherical grid of grid size[5] $h=\Delta\theta$ and $k=\Delta\phi$. Using a 2nd order finite difference approximation, the first two terms on the left side are

$$\frac{\partial^2 u}{\partial\theta^2}+\cot\theta\frac{\partial u}{\partial\theta}=\left(\frac{\partial^2 u}{\partial\theta^2}+\cot\theta\frac{\partial u}{\partial\theta}\right)_{\mathrm{FD2}}-\frac{h^2}{12}\left(\frac{\partial^4 u}{\partial\theta^4}+2\cot\theta\frac{\partial^3 u}{\partial\theta^3}\right)+\mathcal{O}\left(h^4\right). \tag{A.2}$$

We then seek an expression to replace the third and fourth order derivatives in the second parenthesis. Differentiating (A.1) w.r.t. θ gives

$$\frac{\partial^3 u}{\partial\theta^3}+\cot\theta\frac{\partial^2 u}{\partial\theta^2}-\frac{1}{\sin^2\theta}\frac{\partial u}{\partial\theta}-\frac{2\cos\theta}{\sin^3\theta}\frac{\partial^2 u}{\partial\phi^2}+\frac{1}{\sin^2\theta}\frac{\partial^3 u}{\partial\theta\partial\phi^2}=\frac{\partial f}{\partial\theta}. \tag{A.3}$$

The second derivative reads

$$\begin{aligned}&\frac{\partial^4 u}{\partial\theta^4}+\cot\theta\frac{\partial^3 u}{\partial\theta^3}-\frac{2}{\sin^2\theta}\frac{\partial^2 u}{\partial\theta^2}+\frac{2\cos\theta}{\sin^3\theta}\frac{\partial u}{\partial\theta}\\&\quad+\frac{6-4\sin^2\theta}{\sin^4\theta}\frac{\partial^2 u}{\partial\phi^2}-\frac{4\cos\theta}{\sin^3\theta}\frac{\partial^3 u}{\partial\theta\partial\phi^2}+\frac{1}{\sin^2\theta}\frac{\partial^4 u}{\partial\theta^2\partial\phi^2}=\frac{\partial^2 f}{\partial\theta^2}.\end{aligned} \tag{A.4}$$

[5] We follow the standard notation to use h for the grid spacing, which should not be confused with the horizon function $h(\theta,\phi)$ used elsewhere in the paper.

Combining the last two equations, we have

$$\begin{aligned}\frac{\partial^4 u}{\partial\theta^4}+2\cot\theta\frac{\partial^3 u}{\partial\theta^3} &= \left(\cot\theta\frac{\partial f}{\partial\theta}+\frac{\partial^2 f}{\partial\theta^2}\right)+\left(1+\frac{1}{\sin^2\theta}\right)\frac{\partial^2 u}{\partial\theta^2}-\frac{\cos\theta}{\sin^3\theta}\frac{\partial u}{\partial\theta}\\ &\quad+\frac{2\sin^2\theta-4}{\sin^4\theta}\frac{\partial^2 u}{\partial\phi^2}+\frac{3\cos\theta}{\sin^3\theta}\frac{\partial^3 u}{\partial\theta\partial\phi^2}-\frac{1}{\sin^2\theta}\frac{\partial^4 u}{\partial\theta^2\partial\phi^2}.\end{aligned}\tag{A.5}$$

The right hand side of this equation can be evaluated by 2nd order finite difference approximation and be substituted back to (A.2) while the overall error is maintained at $\mathcal{O}(h^4)$. We can then combine the result with (27) to obtain the *fully*-compact 4th order finite difference approximation for the spherical Poisson equation (A.1). The explicit expression is

$$\begin{aligned}&\left(\frac{\partial^2 u}{\partial\theta^2}+\cot\theta\frac{\partial u}{\partial\theta}\right)_{\mathrm{FD2}}+\left(\frac{1}{\sin^2\theta}\frac{\partial^2 u}{\partial\phi^2}\right)_{\mathrm{FD2}}-\frac{k^2}{12}\left[\frac{\partial^2 f}{\partial\phi^2}-\frac{\partial^4 u}{\partial\theta^2\partial\phi^2}-\cot\theta\frac{\partial^3 u}{\partial\theta\partial\phi^2}\right]_{\mathrm{FD2}}\\ &-\frac{h^2}{12}\left[\cot\theta\frac{\partial f}{\partial\theta}+\frac{\partial^2 f}{\partial\theta^2}+\left(1+\frac{1}{\sin^2\theta}\right)\frac{\partial^2 u}{\partial\theta^2}-\frac{\cos\theta}{\sin^3\theta}\frac{\partial u}{\partial\theta}+\frac{2\sin^2\theta-4}{\sin^4\theta}\frac{\partial^2 u}{\partial\phi^2}\right.\\ &\left.+\frac{3\cos\theta}{\sin^3\theta}\frac{\partial^3 u}{\partial\theta\partial\phi^2}-\frac{1}{\sin^2\theta}\frac{\partial^4 u}{\partial\theta^2\partial\phi^2}\right]_{\mathrm{FD2}}+\mathcal{O}\left(h^4,h^2k^2,k^4\right)\quad=f.\end{aligned}\tag{A.6}$$

ORCID iDs

Hon-Ka Hui https://orcid.org/0000-0002-2815-527X
Lap-Ming Lin https://orcid.org/0000-0002-4638-5044

References

[1] Alcubierre M, Brügmann B, Diener P, Koppitz M, Pollney D, Seidel E and Takahashi R 2003 *Phys. Rev.* D **67** 084023

[2] Anninos P, Bernstein D, Brandt S, Libson J, Massó J, Seidel E, Smarr L, Suen W M and Walker P 1995 *Phys. Rev. Lett.* **74** 630–3

[3] Ashtekar A and Krishnan B 2003 *Phys. Rev.* D **68** 104030

[4] Ashtekar A and Krishnan B 2004 *Living Rev. Relativ.* **7** 1–91

[5] Baiotti L and Rezzolla L 2006 *Phys. Rev. Lett.* **97** 141101

[6] Barros S R 1991 *J. Comput. Phys.* **92** 313–48

[7] Baruch G, Fibich G and Tsynkov S 2009 *J. Comput. Phys.* **228** 3789–815

[8] Baumgarte T W, Cook G B, Scheel M A, Shapiro S L and Teukolsky S A 1998 *Phys. Rev.* D **57** 7299–311

[9] Baumgarte T W and Shapiro S L 2010 *Numerical Relativity: Solving Einstein's Equations on the Computer* (Cambridge University Press)

[10] Bozzola G 2021 *J. Open Source Softw.* **6** 3099

[11] Brandt A 1977 *Math. Comput.* **31** 333–90

[12] Brandt S and Brügmann B 1997 *Phys. Rev. Lett.* **78** 3606–9

[13] Briggs W L, Henson V E and McCormick S F 2000 *A Multigrid Tutorial* (SIAM)

[14] Brill D R and Lindquist R W 1963 *Phys. Rev.* **131** 471–6

[15] Britt S, Tsynkov S and Turkel E 2010 *J. Sci. Comput.* **45** 26–47

[16] Cheong P C K, Lin L M and Li T G F 2020 *Class. Quantum Grav.* **37** 145015

[17] Choptuik M and Unruh W G 1986 *Gen. Relativ. Gravit.* **18** 813–43

[18] Collatz L 1960 *The Numerical Treatment of Differential Equations* (Springer)

[19] Cook G B, Choptuik M W, Dubal M R, Klasky S, Matzner R A and Oliveira S R 1993 *Phys. Rev.* D **47** 1471–90

[20] Deriaz E 2020 *BIT Numer. Math.* **60** 199–233

[21] East W E, Ramazanoğlu F M and Pretorius F 2012 *Phys. Rev.* D **86** 104053

[22] Evans C, Ferguson D, Khamesra B, Laguna P and Shoemaker D 2020 *Class. Quantum Grav.* **37** 15LT02

[23] Ge Y 2010 *J. Comput. Phys.* **229** 6381–91

[24] Hackbusch W 1985 *Multi-Grid Methods and Applications* (*Springer Series in Computational Mathematics* vol 4) (Springer)

[25] Hawking S W and Ellis G F R 1973 *The Large Scale Structure of Space-Time* (*Cambridge Monographs on Mathematical Physics*) (Cambridge University Press)

[26] Hughes S A, Keeton C R, Walker P, Walsh K T, Shapiro S L and Teukolsky S A 1994 *Phys. Rev.* D **49** 4004–15

[27] Lai M C and Tseng J M 2007 *J. Comput. Appl. Math.* **201** 175–81

[28] Lin L M and Novak J 2007 *Class. Quantum Grav.* **24** 2665

[29] Löffler F *et al* 2012 *Class. Quantum Grav.* **29** 115001

[30] Misner C W, Thorne K S and Wheeler J A 1973 *Gravitation* (W. H. Freeman)

[31] Moldenhauer N, Markakis C M, Johnson-McDaniel N K, Tichy W and Brügmann B 2014 *Phys. Rev.* D **90** 084043

[32] Penrose R 1965 *Phys. Rev. Lett.* **14** 57–59

[33] Pook-Kolb D, Birnholtz O, Krishnan B and Schnetter E 2019 *Phys. Rev.* D **99** 064005

[34] Pook-Kolb D, Hennigar R A and Booth I 2021 *Phys. Rev. Lett.* **127** 181101

[35] Prasad V, Gupta A, Bose S, Krishnan B and Schnetter E 2020 *Phys. Rev. Lett.* **125** 121101

[36] Press W H, Teukolsky S A, Vetterling W T and Flannery B P 2007 *Numerical Recipes 3rd Edition: The Art of Scientific Computing* (Cambridge University Press)

[37] Schnetter E 2003 *Class. Quantum Grav.* **20** 4719

[38] Schnetter E, Krishnan B and Beyer F 2006 *Phys. Rev.* D **74** 024028

[39] Shibata M 1997 *Phys. Rev.* D **55** 2002–13

[40] Sykes B, Mueller B, Cordero-Carrión I, Cerdá-Durán P and Novak J 2023 *Phys. Rev.* D **107** 103010

[41] Thornburg J 1996 *Phys. Rev.* D **54** 4899–918

[42] Thornburg J 2003 *Class. Quantum Grav.* **21** 743

[43] Thornburg J 2007 *Living Rev. Relativ.* **10** 1–68

[44] Trottenberg U and Oosterlee C W and 2000 *Multigrid A* Schuller (Academic)

[45] Wesseling P 1992 *An Introduction to Multigrid Methods* (Wiley)

[46] Zhang J 2002 *J. Comput. Phys.* **179** 170–9