PHYSICAL REVIEW D **85,** 064014 (2012)

New code for equilibriums and quasiequilibrium initial data of compact objects

Kōji Uryū[1] and Antonios Tsokaros[2]

[1]*Department of Physics, University of the Ryukyus, Senbaru, Nishihara, Okinawa 903-0213, Japan*
[2]*Department of I.C.S.E., University of Aegean, Karlovassi 83200, Samos, Greece*

(Received 15 August 2011; published 9 March 2012)

We present a new code, named COCAL—Compact Object CALculator, for the computation of equilibriums and quasiequilibrium initial data sets of single or binary compact objects of all kinds. In the COCAL code, those solutions are calculated on one or multiple spherical coordinate patches covering the initial hypersurface up to the asymptotic region. The numerical method used to solve field equations written in elliptic form is an adaptation of self-consistent field iterations in which Green's integral formula is computed using multipole expansions and standard finite difference schemes. We extended the method so that it can be used on a computational domain with excised regions for a black hole and a binary companion. Green's functions are constructed for various types of boundary conditions imposed at the surface of the excised regions for black holes. The numerical methods used in COCAL are chosen to make the code simpler than any other recent initial data codes, accepting the second order accuracy for the finite difference schemes. We perform convergence tests for time symmetric single black hole data on a single coordinate patch, and binary black hole data on multiple patches. Then, we apply the code to obtain spatially conformally flat binary black hole initial data using boundary conditions, including the one based on the existence of equilibrium apparent horizons.

DOI: 10.1103/PhysRevD.85.064014 PACS numbers: 04.25.D−, 04.30.Db, 04.40.Dg, 97.60.−s

I. INTRODUCTION

In the last decades, simulation codes for compact objects have been successfully developed in the field of numerical relativity, and various dynamical simulations have been performed. Those simulations include inspirals to mergers of binary neutron stars and black holes [1], a massive core collapse to a neutron star (NS) or black hole (BH) [2], and BH dynamics in higher dimensions [3]. Recent efforts are moving towards more realistic situations such as incorporating the microphysics of nuclear matter [4], performing binary black hole (BBH) merger simulations in a wider range of parameter space (mass ratio and spins [5]), or performing BBH mergers in an ambient disk [6]. Accordingly, more realistic and accurate constructions of initial data for such compact objects are required.

Several researchers have developed methods for computing various types of initial data sets for those simulations [7–14] and equilibriums of rotating compact objects (see, e.g., [15]). Many of such initial data codes are specialized to a certain problem such as a single stationary and axisymmetric neutron star or BBH data on a conformally flat initial hypersurface. An exception is LORENE [16], which is one of most used codes for computing initial data for the merger simulations of binary neutron stars and black holes. The LORENE code was originally developed for computing rapidly rotating neutron stars, but has been extended to be capable of computing various kinds of equilibriums and quasiequilibrium initial data sets.

In this paper, we introduce our project for developing new codes for computing initial data of astrophysical compact objects, a single object as well as binary compact objects of all kinds, and present several tests for the new codes. Our aim is to develop a set of codes for computing, on an initial hypersurface, a single neutron star (or a compact star such as a quark star), binary neutron stars and black holes, a central neutron star or black hole surrounded by a toroidal disk, and all these systems with magnetic fields. We call our new codes "COCAL," which is the abbreviation for Compact Object CALculator.[1] A noteworthy idea of the COCAL project is to develop a code using less technical numerical methods than the recent initial data solvers with spectral methods [8–10,16]. Also, the modules and subroutines of the FORTRAN 90 code are structured simply so that the code may be accessible by those who mastered introductory courses for programing. Such features will help future developments to incorporate more complex physics in the code, such as radiation, neutrino radiation transfers, or realistic equations of state for the high density nuclear matter.

The numerical method used in COCAL is based on the Komatsu-Eriguchi-Hachisu (KEH) method for computing the equilibrium of a rotating neutron star [17]. In our previous works [11,12], we have extended the KEH method for computing initial data for binary compact objects in quasiequilibriums. In Ref. [13], we have introduced multiple spherical coordinate patches for computing binary compact objects. We improve the idea of the multiple patches in all aspects in the new COCAL code. In Ref. [14], we have presented convergence tests and solution sequences for rotating neutron star initial data, which were calculated by the first version of COCAL. In this paper, to introduce the COCAL code, we focus on the basic setup of the multiple spherical coordinate patches and the coordinate grids, the method of

[1]"Cocàl" means "seagull" in the Trieste dialect of Italian.

the elliptic equation solver on the multiple patches, and convergence tests for binary black hole initial data. The paper is organized as follows: in Sec. II we introduce an overview of the COCAL project, then coordinate setups, the elliptic solver, and other materials on numerical computing. In Sec. III the results of convergence tests are presented. In Sec. IV solutions of BBH data on a conformally flat initial hypersurface are presented. We use geometric units with $G = c = 1$ throughout the paper.

II. COCAL CODE

A. Overview

In the COCAL project, we aim to develop numerical codes for computing a single compact object as well as binary compact objects in (quasi)equilibrium using a common numerical method as much as possible. A plan for such codes also depends on how to formulate the problem to solve such compact objects. Usually, a system of equations to describe equilibriums of compact objects involves a set of elliptic equations for the gravitational fields, and relativistic hydrodynamical equations including the Euler equations and the rest mass conservation equation, to which a stationary condition, either a time or a helical symmetry [18], is imposed. When the magnetic field is present, elliptic equations for the electromagnetic fields are added, and the equations for the fluid are replaced by magnetohydrodynamical (MHD) Euler equations. Because the stationary Euler, or MHD-Euler, equations are difficult to integrate numerically, a set of first integrals in the form of algebraic equations, a sufficient condition for the stationary (MHD) Euler equations, is derived and solved simultaneously with the field equations (see, e.g., [19,20]).

A choice for the numerical method is therefore made according to what kind of solver is used for solving the system of elliptic equations. The numerical method used in COCAL is based on the KEH method for computing equilibriums of rotating neutron stars [17]. In this method, the elliptic equations are solved on spherical coordinates using Green's formula iteratively. This is done by separating the flat Laplacian or Helmholtz operator on the variable to be solved for, then moving the remaining (possibly nonlinear) terms to the source, and rewriting it in the integral form using Green's formula. Expanding Green's function using spherical harmonics, the formula is integrated on the spherical coordinate grids numerically (see, e.g., [11,12,21]). The method is extended for computations of binary compact objects as discussed in this section.

We choose simple finite difference formulas which are mostly second order accurate, and in some cases we choose third or fourth order formulas only if they are necessary (see Sec. III B 1). No symmetry, such as an equatorial plane symmetry, is assumed *a priori* on the 3D spherical computational domain. The COCAL code is written in FORTRAN 90 language, and runs with a few GB of memory for a model with a moderate resolution.

We have developed basic subroutines for the COCAL code, including the coordinate grid setups for single and multiple spherical coordinates, as well as the elliptic solvers for a single or binary compact object, which we discuss in detail below. Mainly, two types of initial value formulations for Einstein's equation have been coded so far; one assumes spatial conformal flatness (Isenberg-Wilson-Mathews formulation [22,23]), and the other nonconformal flatness (waveless formulation [12,24]).

Also, the quadrupole formula, to compute the gravitational wave amplitude and luminosity, and a Helmholtz solver have been developed. We are in the phase to test all basic subroutines by computing simple test problems as well as known problems such as BBH initial data or rotating neutron star solutions. In the next step, we will combine these developments and start computing new equilibriums and initial data sets such as helically symmetric binary compact objects or magnetized compact objects.

B. Coordinate patches for binary systems

We assume that the spacetime $\mathcal{M}$ is foliated by a family of spacelike hypersurfaces $(\Sigma_t)_{t\in\mathbb{R}}$, $\mathcal{M} = \mathbb{R}\times\Sigma$ parametrized by $t \in \mathbb{R}$. In the COCAL code, we solve fields on an initial hypersurface Σ_t which may be stationary (in equilibrium) or quasistationary (in quasiequilibrium). The initial hypersurface Σ_t is covered by overlapping multiple spherical coordinate patches whose coordinates are denoted by (r, θ, ϕ). Angular coordinates cover all directions $(\theta, \phi) \in [0, \pi]\times[0, 2\pi]$ without any symmetry imposed. We also introduce Cartesian coordinates as a convenient reference frame in a standard manner, that is, to have the positive side of the x axis coincide with a $(\theta, \phi) = (\pi/2, 0)$ line, that of the y axis with a $(\theta, \phi) = (\pi/2, \pi/2)$ line, and that of the z axis with a $\theta = 0$ line.

In Fig. 1 a schematic diagram of three spherical coordinate patches whose coordinates are discretized in grid points is shown for the case of computing BH-NS binary systems by COCAL. We show the 2D section of the 3D hypersurface that may agree with the equatorial or meridional plane of the compact objects. Even though this may be the most complex setup for coordinate grids in COCAL, it is not technical at all compared to those of existing codes in which adaptive coordinates are used.

For the computation of binary systems, two compact objects are placed at the centers of the two patches. We call these two patches the compact object coordinate patch (COCP) and the third patch the asymptotic region coordinate patch (ARCP). A domain of the COCP is defined between two concentric spheres S_a and S_b from which an interior of an another sphere S_e is excised. Writing radii of S_a, S_b, and S_e as r_a, r_b, and r_e, respectively, we define spherical coordinates of the COCP as $(r, \theta, \phi) \in [r_a, r_b]\times[0, \pi]\times[0, 2\pi]$ and locate the center of the excised sphere at $(r, \theta, \phi) = (d_s, \pi/2, 0)$, which is on the positive side of

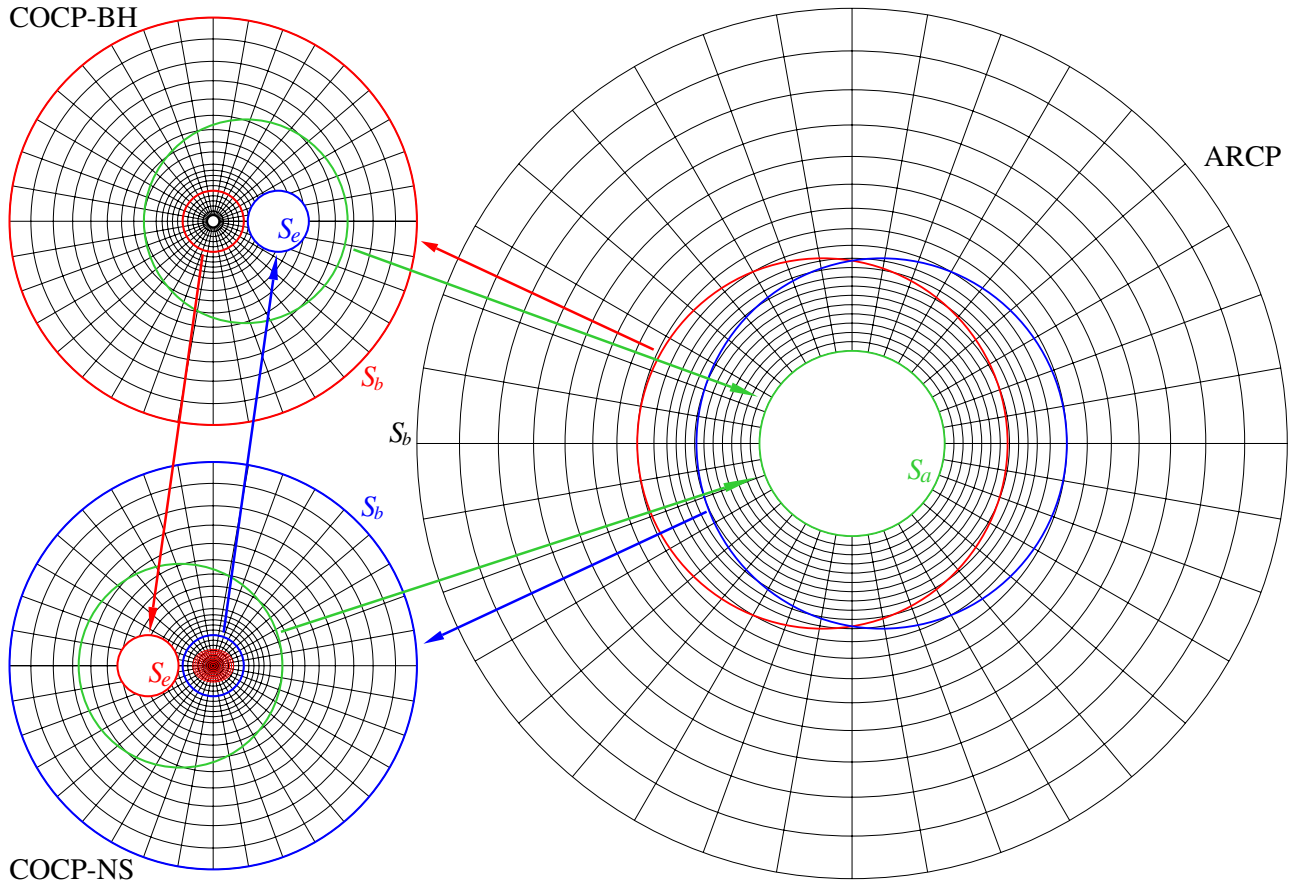

FIG. 1 (color online). A typical setup for multiple coordinate grid patches in the COCAL code for a BH-NS system. Left top and bottom patches are those for the COCP centered at each compact object (BH for the top, and NS for the bottom). The smallest circle with the thick curve in the COCP-BH is the sphere S_a, where the interior region is excised and certain BH boundary conditions are imposed. The ovals drawn in the COCP-NS denote NS. The right patch is that for the ARCP, centered at the mass center of the system. The arrows represent maps of potentials between the multiple patches. Alternatively, the radius of the COCP may be extended to the asymptotic region, instead of using the ARCP. Note that the spheres S_a, S_b, and S_e of these coordinate patches are distinct ones on a spacelike hypersurface Σ_t. The radius of each coordinate patch does not reflect the size used in actual computations.

the x axis.[2] We introduce the excision of a domain interior of the sphere S_e for computing binary systems and elucidate its role in the following section. When a single and/or axisymmetric object is computed, the excision inside S_e is not used. When BH is computed on the COCP-BH with certain BH boundary conditions on $S_a (r_a > 0)$, the region inside S_a is excised. When a NS or a puncture BH is calculated, the sphere S_a of the COCP is removed by setting $r_a = 0$, so the radial coordinate covers up to $r = 0$. A domain of the ARCP is defined between two concentric spheres S_a and S_b, and its spherical coordinates are defined as $(r, \theta, \phi) \in [r_a, r_b] \times [0, \pi] \times [0, 2\pi]$. When values of the field potentials or other variables are communicated from one patch to the other, those values on a certain sphere are mapped to a corresponding boundary sphere as indicated by arrows in Fig. 1.

Values of radii r_a, r_b, and r_e for each of the coordinate patches used in actual computations will be summarized in Sec. III. Typically, they are set as follows. For the case of using three patches as in Fig. 1, the radius r_a of the inner boundary S_a of the ARCP is taken large enough to be placed outside the excised spheres S_e for compact objects on the COCP, but small compared to the size of the domain r_b of the COCP. The outer boundary of the ARCP, the radius r_b of the sphere S_b, is extended to the asymptotic region when a field falls off like a Coulomb field, while it is truncated at the near zone when a radiation field is calculated. The center of the ARCP is located at the center of mass of binary compact objects. Therefore, for a compact object of mass M, the orders of radii become $r_a = O(M)$ (0 for NS), $r_b = O(100M)$, $r_e = O(M) - O(10M)$ for the COCP, and $r_a = O(10M)$, $r_b = O(10^6 M)$ or larger for the ARCP.

As another option for the choice of coordinate patch, the outer radius of each COCP r_b may be extended to asymptotics, say, $r_b = O(10^6 M)$, and the ARCP is removed. This option simplifies the code, but it cannot be used when a radiation field is computed by solving the Helmholtz equation. We present tests for the Helmholtz solver in a separate paper.

[2]The positive side of the x axis of the COCP-NS in Fig. 1 is pointing from the center of the coordinate grids to the left.

C. Elliptic equation solver

As mentioned earlier, the formulation for computing (quasi)equilibrium configurations of compact objects results in a coupled system of elliptic equations, either Poisson or Helmholtz equations with nonlinear source terms, coupled with algebraic equations. The numerical method used in COCAL to solve such a system of equations is an extension of the KEH method which is an application of a self-consistent field method for computing equilibriums of self-gravitating fluids to general relativistic stars [17]. A distinctive feature of these methods is the use of Green's formula for an elliptic equation solver. We have introduced in previous papers our implementation of the KEH method to compute binary neutron stars and black holes [11–14].

In the COCAL code, we have made a major change in the choice of coordinate patch, and accordingly in the elliptic equation solver. Our new implementation is better in all aspects for computing binary compact objects than our previous ones. We will come back to this point after we introduce the elliptic equation solver in COCAL.

In solving each field equation, we separate out a flat Laplacian or Helmholtz operator $\mathcal{L}$, and write it with a nonlinear source S,

$$\mathcal{L}\,\Phi = S, \tag{1}$$

on an initial slice Σ_t, where Φ represents metric potentials. For the case of Laplacian $\mathcal{L} = \Delta$, using Green's function without the boundary $G(x, x') = 1/|x - x'|$ that satisfies

$$\Delta G(x, x') = -4\pi\delta(x - x'), \tag{2}$$

Green's identity is obtained by

$$\Phi(x) = -\frac{1}{4\pi}\int_V G(x, x')S(x')d^3x' + \frac{1}{4\pi}\int_{\partial V}[G(x, x')\nabla'^a\Phi(x') - \Phi(x')\nabla'^a G(x, x')]dS'_a, \tag{3}$$

where V is the domain of integration, x, $x' \in V \subseteq \Sigma_0$, and ∂V is its boundary. For the case of the BH-NS system shown in Fig. 1, the boundary of the COCP-BH becomes $\partial V = S_a \cup S_b \cup S_e$, that of the COCP-NS $\partial V = S_b \cup S_e$, and that of the ARCP $\partial V = S_a \cup S_b$. For the evaluation of the integrals in Eq. (3), a multipole expansion of $G(x, x')$ in the associated Legendre functions on the spherical coordinate is used,

$$G(x, x') = \frac{1}{|x - x'|} = \sum_{\ell=0}^{\infty} g_\ell(r, r')\sum_{m=0}^{\ell}\epsilon_m\frac{(\ell - m)!}{(\ell + m)!} \times P_\ell^m(\cos\theta)P_\ell^m(\cos\theta')\cos m(\varphi - \varphi'), \tag{4}$$

where the radial Green's function $g_\ell(r, r')$ is defined by

$$g_\ell(r, r') = \frac{r_<^\ell}{r_>^{\ell+1}}, \tag{5}$$

with $r_> := \sup\{r, r'\}$, $r_< := \inf\{r, r'\}$, and the coefficients ϵ_m are equal to $\epsilon_0 = 1$, and $\epsilon_m = 2$ for $m \geq 1$.

Equation (3) is an integral identity but is not a solution of Eq. (1) in the sense that Φ and its derivative $n^a\nabla_a\Phi$ cannot be freely specified simultaneously. Equation (3) can be used to compute a potential over V, only if correct values of Φ and $n^a\nabla_a\Phi$ are known at the boundary ∂V. Here, n^a is an outward normal to ∂V. Therefore, as it is commonly found in standard textbooks for electromagnetism [25], a homogeneous function $F(x, x')$ for the Laplacian is added to evaluate Green's function that satisfies the boundary condition at ∂V. For example, a Green's function $G(x, x') + F(x, x') = 0$ at ∂V is used to impose the Dirichlet boundary condition. In our previous paper [13], we have developed an elliptic equation solver on multiple coordinate patches that uses such Green's functions, and we solve Eq. (3) by iteration.

The construction of such a Green's function that satisfies a boundary condition is, however, possible only when a certain specific geometry of the domain of computation is adapted to the coordinate systems. In the present case for the COCP of the COCAL code in Fig. 1, a Green's function that satisfies boundary conditions at S_a and S_b may not be derived in a practical form of the equation, because we excised the region inside the sphere S_e. To impose boundary conditions at S_a and S_b, we introduce a homogeneous solution $\chi(x)$, and write a formal solution as

$$\Phi(x) = \chi(x) + \Phi_{\rm INT}(x), \tag{6}$$

where $\Phi_{\rm INT}$ is equal to the right-hand side of Eq. (3),

$$\Phi_{\rm INT}(x) = -\frac{1}{4\pi}\int_V G(x, x')S(x')d^3x' + \frac{1}{4\pi}\int_{\partial V}[G(x, x')\nabla'^a\Phi(x') - \Phi(x')\nabla'^a G(x, x')]dS'_a. \tag{7}$$

The homogeneous solution is computed so that the potential Φ satisfies the boundary conditions, which are either Dirichlet, Neumann, or Robin boundary conditions at the boundary spheres S_a or S_b:

$$\text{Dirichlet:}\quad \Phi_{\rm BC} = f_{\rm D}, \tag{8}$$

$$\text{Neumann:}\quad n^a\nabla_a\Phi_{\rm BC} = f_{\rm N}, \tag{9}$$

$$\text{Robin:}\quad n^a\nabla_a\Phi_{\rm BC} + \tfrac{1}{4}\nabla_a n^a\Phi_{\rm BC} = f_{\rm R}, \tag{10}$$

where $f_{\rm D}$, $f_{\rm N}$, and $f_{\rm R}$ are given functions on the spheres S_a or S_b. Formulas for $\chi(x)$ are derived by using a Legendre expansion in the usual manner, as shown in Appendixes A and B. Noticing $\chi(x) = \Phi(x) - \Phi_{\rm INT}(x)$ the formulas for $\chi(x)$ can be written analogously to the surface integral terms of Green's formula, but with a different kernel function $G^{\rm BC}$,

$$\chi(x) = \frac{1}{4\pi}\int_{S_a\cup S_b}[G^{\rm BC}(x, x')\nabla'^a(\Phi_{\rm BC} - \Phi_{\rm INT})(x') - (\Phi_{\rm BC} - \Phi_{\rm INT})(x')\nabla'^a G^{\rm BC}(x, x')]dS'_a. \tag{11}$$

TABLE I. List of Green's functions available in the COCAL code. The second and third columns correspond to the types of boundary conditions imposed on the boundary spheres S_a and S_b, respectively. The case with no boundary condition is denoted by "None."

$G^{\rm BC}(x, x')$	Boundary S_a	Boundary S_b
$G^{\rm NB}(x, x')$	None	None
$G^{\rm DD}(x, x')$	Dirichlet	Dirichlet
$G^{\rm ND}(x, x')$	Neumann	Dirichlet
$G^{\rm DN}(x, x')$	Dirichlet	Neumann
$G^{\rm NN}(x, x')$	Neumann	Neumann
$G^{\rm RD}(x, x')$	Robin	Dirichlet
$G^{\rm DR}(x, x')$	Dirichlet	Robin

The function $G^{\rm BC}(x, x')$ is expanded in terms of the associated Legendre functions,

$$G^{\rm BC}(x,x') = \sum_{\ell=0}^{\infty} g_\ell^{\rm BC}(r,r') \sum_{m=0}^{\ell} \epsilon_m \frac{(\ell-m)!}{(\ell+m)!} \times P_\ell^m(\cos\theta) P_\ell^m(\cos\theta') \cos m(\varphi-\varphi'), \quad (12)$$

where the radial function $g_\ell^{\rm BC}(r, r')$ is chosen according to the type of boundary conditions used. We derive such radial functions used in the corresponding surface integrals for various cases of boundary conditions as listed in Table I. Concrete forms of these functions are presented in Appendix B.

D. Iteration procedure

The final solution will be obtained from the iteration of Eq. (6), with Eqs. (7) and (11), where the explicit form of Eq. (11) for $\chi(x)$ depends on the boundary condition, for example, Eq. (A9) or (A12).

We summarize the nth step of the Poisson solver in the COCAL code as follows (here, intermediate variables during an iteration step are denoted with a hat as $\hat{\Phi}_{\rm INT}$, $\hat{\chi}$, and $\hat{\Phi}$):

(1) Compute the volume source term $S(\Phi^{(n-1)})$ as well as the surface source terms on all possible surfaces S_a, S_b, S_e.
(2) Compute the volume integral and the surface integral at S_e for obtaining $\hat{\Phi}_{\rm INT}(x)$ from Eq. (7).
(3) Compute the effective source for the integral on S_a and S_b. For the Dirichlet boundary condition it will be $\Phi_{\rm BC} - \hat{\Phi}_{\rm INT}$, while for Neumann it will be $\frac{\partial \Phi_{\rm BC}}{\partial r} - \frac{\partial \hat{\Phi}_{\rm INT}}{\partial r}$.
(4) Compute the surface integrals at S_a and S_b for obtaining $\hat{\chi}(x)$ according to Eq. (11) using the appropriate function $G^{\rm BC}$ for the boundary conditions of the problem.
(5) Add the results from steps (2) and (4) to obtain $\hat{\Phi}(x)$ from Eq. (6).
(6) Update $\Phi^{(n)}$ according to

$$\Phi^{(n)}(x) := c\hat{\Phi}(x) + (1-c)\Phi^{(n-1)}(x),$$

where $0.1 \le c \le 0.4$.
(7) Check if

$$2\frac{|\Phi^{(n)} - \Phi^{(n-1)}|}{|\Phi^{(n)}| + |\Phi^{(n-1)}|} < \epsilon_c$$

for all points of the grids, where $\epsilon_c = 10^{-6} - 10^{-8}$ is taken in typical computations, and $\epsilon_c = 10^{-7}$ in this paper. If yes, exit. If no, go back to step (1).

The above iteration procedure is applied to each coordinate patch one after the other. In step (1) the sources of the surface terms are computed either from boundary conditions to be imposed on the surface, or from data of the corresponding surface on the other patch (see Fig. 1 for how the potentials are transferred from one boundary surface to the other). Several different iteration schemes are possible for solving a set of elliptic equations for more than one variable. In all our experiments, the convergence of the iteration does not depend on the order of computing those variables at each iteration step.

We will see this elliptic equation solver produce accurate solutions for test problems of binary black hole data. Two comments on the elliptic equation solver are in order. Although $\Phi_{\rm INT}$ in Eq. (7) involves surface integrals on all S_a, S_b, and S_e, those on S_a and S_b are not included in $\Phi_{\rm INT}$ in an actual computation. Those computations are redundant because the homogeneous solution $\chi(x)$ is determined again from the surface integrals on S_a and S_b as in Eq. (11). So far, we do not plan to develop elliptic solvers for vector (tensor) fields in which Green's functions are expanded in vector (tensor) spherical harmonics. Instead, we write the Cartesian components of vector or tensor equations and solve each component as a scalar equation on spherical grids for simplicity. We will see an example in Sec. IV (see also [12]).

E. Grid spacing

We apply a finite difference scheme to solve the system of equations for compact objects on the spherical domain introduced in Sec. II B (see Fig. 1). Spherical coordinates (r, θ, ϕ) for the COCP and the ARCP are bounded by two concentric spheres S_a and S_b of radius r_a and r_b, respectively, with the possible excision of a sphere S_e of radius r_e inside the COCP. The origin of the radial coordinate r is at the common center of S_a and S_b, and a compact object is placed as its center agrees with the origin of the COCP. The excised sphere for a binary companion S_e is always positioned at a positive value on the x axis at a distance d_s from the origin. Clearly, $r_a < d_s - r_e$. For neutron star calculations the sphere S_a is absent and the coordinate system extends from $r = 0$ to r_b.

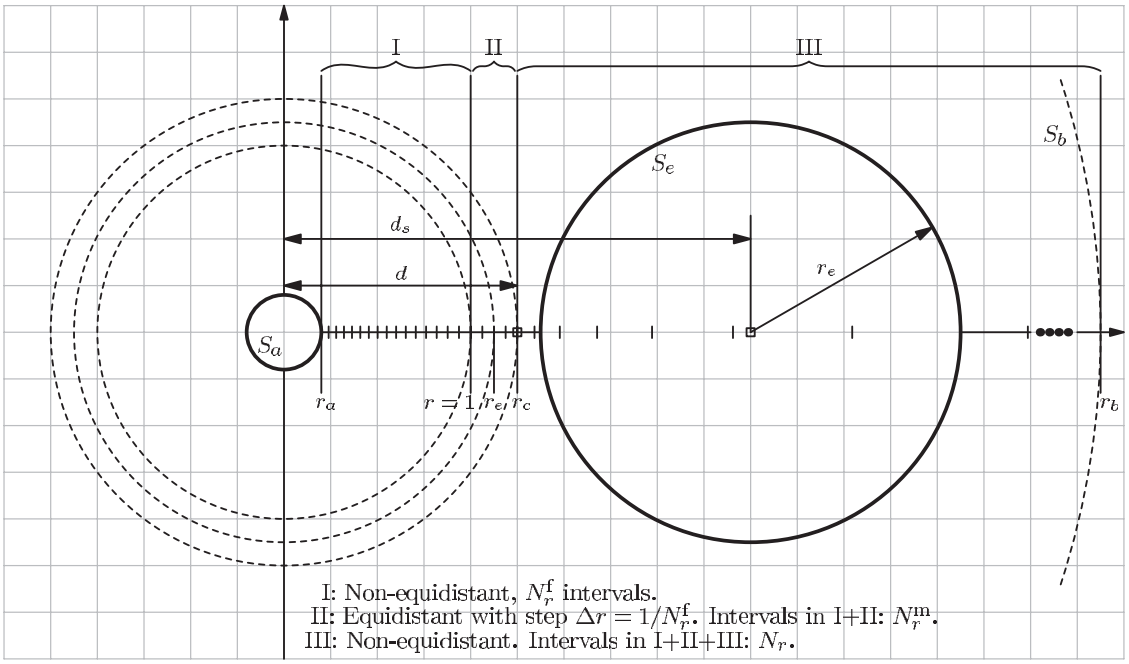

FIG. 2. Radial coordinate grids for the COCP for the case of regions for BHs and binary companions being excised. The radial coordinate grids correspond to those of lowest resolutions, A1, B1, D1, and F1 in Tables III and IV. The position of the center of mass d may vary depending on the mass ratio of the binary system.

In the COCAL code, the spacing of all coordinate grid points (r_i, θ_j, ϕ_k), with $i = 0, \ldots, N_r$, $j = 0, \ldots, N_\theta$, and $k = 0, \ldots, N_\phi$, are freely specifiable. However, in (θ, ϕ) directions, uniform grids are recommended to resolve evenly the trigonometric and associated Legendre functions used in the elliptic equation solver, as well as the structure of compact objects. That is, we set the grid interval in these directions as

$$\Delta\theta_j = \theta_j - \theta_{j-1} = \Delta\theta = \frac{\pi}{N_\theta}, \tag{13}$$

$$\Delta\phi_k = \phi_k - \phi_{k-1} = \Delta\phi = \frac{2\pi}{N_\phi}. \tag{14}$$

The grid spacing in the radial direction r is usually constructed, on one hand, to resolve the vicinity of the compact object with finer grid spacings and, on the other hand, to extend to asymptotics using increasingly sparse spacings.[3] The setup for radial grids of the COCP in the present computation is illustrated in Fig. 2. The grid is composed of regions I, II, and III. For the case with $r_a < 1$, region I is set by $r \in [r_a, 1]$, region II by $r \in [1, r_c]$, and region III by $r \in [r_c, r_b]$. We introduce grid numbers $N_r^{\rm f}$, $N_r^{\rm m}$, which correspond to the numbers of intervals in regions I and I + II, respectively. We introduce a standard grid spacing Δr as $\Delta r = 1/N_r^{\rm f}$. For the case with $r_a < 1$, the grid intervals, $\Delta r_i := r_i - r_{i-1}$, are defined by

$$\Delta r_{i+1} = h\Delta r_i, \quad \text{for } i = 1, \cdots, N_r^{\rm f} - 1, \tag{15}$$

$$\Delta r_i = \Delta r, \quad \text{for } i = N_r^{\rm f}, \cdots, N_r^{\rm m}, \tag{16}$$

$$\Delta r_{i+1} = k\Delta r_i, \quad \text{for } i = N_r^{\rm m}, \cdots, N_r - 1, \tag{17}$$

which correspond to regions I, II, and III in Fig. 2, respectively, where the ratios $h(\geq 1)$ and $k(>1)$ are, respectively, determined from the relations

$$1 - r_a = \Delta r \frac{1 - h^{N_r^{\rm f}}}{1 - h}, \tag{18}$$

$$r_b - r_c = \Delta r \frac{k(k^{N_r - N_r^{\rm m}} - 1)}{k - 1}. \tag{19}$$

For the case with $r_a > 1$, which is mostly for the ARCP, the grid intervals Δr_i are defined by

$$\Delta r_i = \Delta r \quad \text{for } i = 1, \cdots, N_r^{\rm m}, \tag{20}$$

$$\Delta r_{i+1} = k\Delta r_i \quad \text{for } i = N_r^{\rm m}, \cdots, N_r - 1, \tag{21}$$

where the ratio k is determined from Eq. (19). Parameters for the grid setup are listed in Table II.

F. Finite differences and multipole expansion

Approximations made in our numerical method are a truncation of the series of Legendre expansions at a finite order of the multipole, and an evaluation of a solution on discretized grids—the finite differencing. The accuracy of the code is, therefore, determined from finite difference formulas to be used, the number of grid points and their spacings, and the number of multipoles being included.

[3]For the case of solving the Helmholtz equation, the region extends to a near zone (the size of several wavelengths of a dominant mode).

TABLE II. Summary of grid parameters. The radius r_e is defined only for the COCP.

r_a	: Radial coordinate where the radial grids start.
r_b	: Radial coordinate where the radial grids end.
r_c	: Radial coordinate between r_a and r_b where the radial grid spacing changes from equidistant to nonequidistant.
r_e	: Radius of the excised sphere.
N_r	: Number of intervals Δr_i in $r \in [r_a, r_b]$.
$N_r^{\rm f}$	: Number of intervals Δr_i in $r \in [r_a, 1]$ for $r_a < 1$ or $r \in [r_a, r_a + 1]$ for $r_a \geq 1$.
$N_r^{\rm m}$	: Number of intervals Δr_i in $r \in [r_a, r_c]$.
N_θ	: Number of intervals $\Delta\theta_j$ in $\theta \in [0, \pi]$.
N_ϕ	: Number of intervals $\Delta\phi_k$ in $\phi \in [0, 2\pi]$.
d	: Coordinate distance between the center of $S_a(r = 0)$ and the center of mass.
d_s	: Coordinate distance between the center of $S_a(r = 0)$ and the center of S_e.
L	: Order of included multipoles.

In the COCAL code we use a second order midpoint rule for numerical integrations and differentiations, along with the second order linear interpolation rule. In the elliptic equation solver (6), the source terms are evaluated at the midpoints,

$$(r_{i-1/2}, \theta_{j-1/2}, \phi_{k-1/2}) = \left(\frac{r_i + r_{i-1}}{2}, \frac{\theta_j + \theta_{j-1}}{2}, \frac{\phi_k + \phi_{k-1}}{2}\right),$$

and integrated with the weights $\Delta r_i \Delta\theta_j \Delta\phi_k$ (other than a Jacobian). The midpoint rule has a few advantages. The second order accuracy of the midpoint rule for a quadrature formula is maintained even with a discontinuity of the derivative of Green's function for a volume integral at $r = r'$. It may be possible to derive a higher order quadrature formula for numerically integrating such functions, but for instance, a Simpson rule does not guarantee fourth order accuracy at grid points r_i, with i being odd integers. Also, an excision of a region inside a sphere S_e for a binary companion on the COCP complicates a derivation of a higher order quadrature formula which maintains the degree of precision near the sphere. Because of the simplicity of the midpoint rule, it is not difficult to modify the weights for an integration to maintain the accuracy. Another advantage of the midpoint rule is that it avoids the coordinate singularities of the spherical coordinates.

In some cases, we also use a third or higher order finite difference formula for the numerical differentiations. In particular, it is found that it is necessary to use the third order finite difference formula for the radial derivatives to maintain *second order* convergence of the field near the BH (see Sec. III B 1). Interpolations of scalar functions from the grid points to the midpoints are done using a second order linear interpolation formula. We often need to interpolate a function from one coordinate patch to the other, such as to compute the source term at S_e of the COCP. In such a case, the functions are interpolated using the fourth order Lagrange formula. For example, when the surface integral at the excised sphere is computed, we need the potential and its derivative at point x' on S_e as seen from the center of S_e. These values are taken by interpolating the nearby $4^3 = 64$ points of the other coordinate system. Some examples of finite difference formulas frequently used in the COCAL code are summarized in Appendix C.

As discussed in [13], the excised region S_e is introduced to improve the resolution in angular directions and, accordingly, to reduce the number of multipoles to resolve a companion object. Without this excised region, the size of the companion object itself has to be resolved by angular grids, while in our setup, it is enough to resolve the size of the excised region, which is usually taken as large as half of the separation $\sim d_s/2$. Then the angle to be resolved can always be about $\sim 2\arcsin(1/2) = \pi/3$. Note that although it is, in principle, possible to excise S_e with a different radius from each COCP, and it is allowed in COCAL, it is more practical to have the same size of the excised region for the same reason as above. To summarize, the angular resolution of a COCP is determined from the degree of accuracy needed to resolve the deformation of the compact objects centered at the patch, and to resolve the size of the excised sphere, $\sim\pi/3$. The angular resolution of the ARCP depends on just how many multipoles one wishes to keep in the near zone to asymptotics. For both the COCP and the ARCP, the number of Legendre expansions is in the range $\ell \sim 10$–16 for computing binary systems.

The Legendre polynomial P_ℓ^m may have ℓ zero crossings in $\theta \in [0, \pi)$, and $\sin m\phi$ or $\cos m\phi$ have $2m$ zeros in $\phi \in [0, 2\pi)$. The number of grid points along the angular coordinates has to be large enough to resolve these multipoles with maximum ℓ or m, say, 4 times more than the number of zeros.

III. CODE TESTS

A. A toy problem for black holes

Convergence tests for time symmetric BH and BBH data are performed to check the numerical method of COCAL presented in Sec. II. We assume the spacetime $\mathcal{M}$ is foliated by a family of spacelike hypersurfaces $(\Sigma_t)_{t\in\mathbb{R}}$, $\mathcal{M} = \mathbb{R} \times \Sigma$ parametrized by $t \in \mathbb{R}$. To obtain simple black hole solutions on Σ_t, we assume time symmetric initial data and that the extrinsic curvature K_{ab} on Σ_t vanishes, or in other words, we assume the line element in the neighborhood of Σ_t,

$$ds^2 = -\alpha^2 dt^2 + \psi^4 f_{ij} dx^i dx^j, \tag{22}$$

where f_{ij} is the flat spatial metric. Decomposing Einstein's equation $G_{\alpha\beta} = 0$ with respect to the foliation using hypersurface normal n^α to Σ_t, and the projection tensor $\gamma^{ab} = g^{\alpha\beta} + n^\alpha n^\beta$ to it, we write the Hamiltonian constraint $G_{\alpha\beta} n^\alpha n^\beta = 0$, and a combination of the

spatial trace of Einstein's equation and the constraint $G_{\alpha\beta}(\gamma^{\alpha\beta} + \frac{1}{2}n^\alpha n^\beta) = 0$, as

$$\nabla^2\psi = 0 \quad \text{and} \quad \nabla^2(\alpha\psi) = 0. \tag{23}$$

These equations have solutions which correspond to the Schwarzschild metric in isotropic coordinates for a single BH. For a two-BH case, a BBH solution is given by Brill and Lindquist [26],

$$\psi = 1 + \frac{M_1}{2r_1} + \frac{M_2}{2r_2} \quad \text{and} \quad \alpha\psi = 1 - \frac{M_1}{2r_1} - \frac{M_2}{2r_2}, \tag{24}$$

where subscripts 1 and 2 correspond to those of the first and second BH; r_1 and r_2 are distances from the first and second BH, respectively; and M_1 and M_2 are mass parameters. The coordinates r_1 and r_2 are written in terms of each other; for example, in the first coordinate system of the COCP, the radial coordinates are

$$r_1 = r \quad \text{and} \quad r_2 = \sqrt{r^2 + d_s^2 - 2rd_s\sin\theta\cos\phi},$$

where θ, ϕ are the angular spherical coordinates of the first coordinate system, and $1 \leftrightarrow 2$ are for the second COCP. On the third coordinate system of the ARCP,

$$r_1 = \sqrt{r^2 + d_1^2 - 2rd_1\sin\theta\cos\phi},$$
$$r_2 = \sqrt{r^2 + d_2^2 - 2rd_2\sin\theta\cos\phi},$$

where d_1 and d_2 are the distance between the center of the ARCP and that of the COCP, and hence $d_s = d_1 + d_2$.

Instead of solving two Laplace equations, Eq. (23), we write an equation for α with a source on the whole domain of Σ_t,

$$\nabla^2\psi = 0 \quad \text{and} \quad \nabla^2\alpha = -\frac{2}{\psi}f^{ij}\partial_i\psi\partial_j\alpha. \tag{25}$$

In an actual computation, the BH centered at the COCP is excised at the radii r_a of S_a, and the binary companion is excised at the radii r_e of S_e which is centered at $x = d_s$. Boundary conditions for these elliptic equations at S_a are taken from analytic solutions (24) when Dirichlet boundary conditions are imposed. Neumann boundary conditions can be imposed with the use of

$$\frac{\partial\psi}{\partial r_1} = -\frac{M_1}{2r_1^2} - \frac{M_2}{2r_2^2}\frac{\partial r_2}{\partial r_1} \quad \text{at } r_1 = r_a, \tag{26}$$

$$\frac{\partial\alpha}{\partial r_1} = \frac{1}{\psi}\left(\frac{M_1}{2r_1^2} + \frac{M_2}{2r_2^2}\frac{\partial r_2}{\partial r_1} - \alpha\frac{\partial\psi}{\partial r_1}\right) \quad \text{at } r_1 = r_a. \tag{27}$$

For the outer boundary conditions, we choose Dirichlet boundary conditions whose data are taken from the analytic solution Eq. (24) in all tests in this section. When the third patch, the ARCP, is not used, Dirichlet data are imposed on $S_b(r = r_b)$ of the COCP, while when the ARCP is used, as in Fig. 1, Dirichlet data are imposed only at S_b of the ARCP.

B. Convergence tests

Convergence tests are performed to examine that the code produces solutions with an expected order of finite difference errors, and to find experimentally an (almost) optimally balanced set of resolutions for each coordinate grid (r_i, θ_j, ϕ_k) which is not over-resolved in one coordinate direction so as not to waste the computational resources. We find, from convergence tests for a single BH solution, it is necessary to use a third order finite difference formula for a radial derivative in the volume source terms in Eq. (7). We also find an optimally balanced resolution between the r_i and θ_j grids. From convergence tests for BBH data, we find appropriate resolutions for the ϕ_i direction and a number of multipoles. Results for the convergence tests are discussed in this section.

1. Single BH

For the first test, we compute a single BH solution with mass parameter $M_1 = 2r_a = 0.4$ (and $M_2 = 0$). Equations (25) are solved on a single patch with a single excision region inside S_a for the BH. In Fig. 3 a fractional error of the lapse α,

$$\left|\frac{\delta\alpha}{\alpha}\right| := \left|\frac{\alpha - \alpha_{\text{exact}}}{\alpha_{\text{exact}}}\right|, \tag{28}$$

is plotted along the x axis for different resolutions in the radial coordinate grids r_i (top and bottom left panels), in the zenith angle grids θ_j (top right panel), and in all grids (bottom right panel). These resolutions are tabulated in Table III and are indicated by A1–A4, B1–B4, and D1–D4, respectively. In the set A1–A4, the radial resolution Δr is doubled, in B1–B4, the zenith angle resolution $\Delta\theta$ is doubled, and in D1–D4, the resolutions in all directions are doubled at each level. Another difference in these results is the order of the finite difference formula used to compute a radial derivative in the volume source term in Eq. (7), where the second order (midpoint) formula is used in the top left and right panels, and the third order (Lagrange) formula is used in the bottom left and right panels. We notice from the top left panel in Fig. 3 that the error does not decrease as $O(\Delta r^2)$ when the number of radial grid points is increased as the parameter sets A1–A4 in Table III even for such a spherically symmetric solution.

It appears that there are two reasons for that. In the top right panel, a convergence test is performed, changing the number of grid points in zenith angle θ_j as the parameter sets B1-B4. This test shows an improvement of the accuracy in $O(\Delta\theta^2)$ in the larger radius $r \gtrsim 100$; that is, the error in this region is dominated by the finite differencing in the θ direction to resolve the integration of the Legendre polynomial accurately. However, the accuracy near the BH is not improved in either test of A1-A4 and B1-B4. In the bottom left panel of Fig. 3, the same convergence test as in the top left panel is performed, but the finite difference formula for the radial derivatives is replaced by that of the

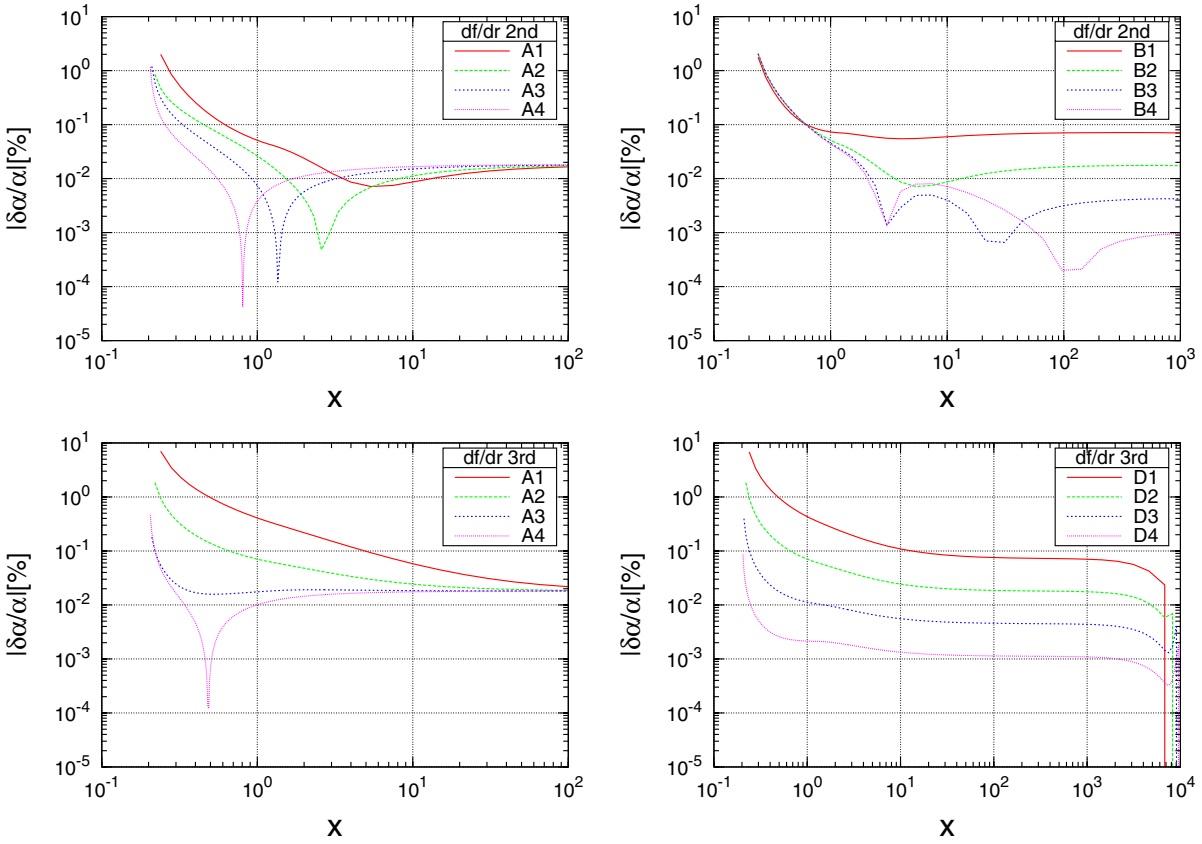

FIG. 3 (color online). Plots of fractional errors in the lapse $\delta\alpha/\alpha$ along the positive x axis. Plots show the errors by changing the number of radial grid points r_i as A1–A4 (top left panel), zenith grid points θ_i as B1–B4 (top right panel), radial grid points r_i as A1–A4 (bottom left panel), and all grid points (r_i, θ_j, ϕ_k) as D1–D4 (bottom right panel). In each panel, solid (red), long dashed (green), dashed (blue), and dotted (magenta) lines are in order from lowest to highest resolutions. In the top panels, a second order finite difference formula is used for calculating radial derivatives, while the third order formula is used in the bottom panels.

third order Lagrange formula $O(\Delta r^3)$. This shows that it is necessary to set the order of the finite difference formula for the radial derivative as $O(\Delta r^3)$ to see $O(\Delta r^2)$ accuracy near the BH. This $O(\Delta r^2)$ error must be due to the mid-point rule used in the numerical integration. However, the error in the larger radius does not decrease in the outer region with the radius $r \gtrsim 100$. Finally, as shown in the bottom right panel of Fig. 3, the error decreases in the second order for the set D1-D4, in which the third order finite difference formula is used for the radial derivatives. Convergence tests are done for increasing grid points in ϕ directions ϕ_k and also for changing the number of Legendre expansions as $L = 4 - 10$, but they do not change the results for such spherically symmetric BH tests.

TABLE III. Grid parameters used in convergence tests for a single BH and equal mass BBH data solved on a single coordinate patch, the COCP.

Type	r_a	r_b	r_c	$N_r^{\rm f}$	$N_r^{\rm m}$	N_r	N_θ	N_ϕ	L
A1	0.2	10^4	1.25	16	20	48	48	96	12
A2	0.2	10^4	1.25	32	40	96	48	96	12
A3	0.2	10^4	1.25	64	80	192	48	96	12
A4	0.2	10^4	1.25	128	160	384	48	96	12
B1	0.2	10^4	1.25	16	20	48	24	96	12
B2	0.2	10^4	1.25	16	20	48	48	96	12
B3	0.2	10^4	1.25	16	20	48	96	96	12
B4	0.2	10^4	1.25	16	20	48	192	96	12
D1	0.2	10^4	1.25	16	20	48	24	24	12
D2	0.2	10^4	1.25	32	40	96	48	48	12
D3	0.2	10^4	1.25	64	80	192	96	96	12
D4	0.2	10^4	1.25	128	160	384	192	192	12

2. Equal mass BBH computed with a single patch

In Fig. 4 results of convergence tests for equal mass BBH data are plotted. In this test we used only a single patch, shown in Fig. 5, where the potential at the radius $r = r_e$ rotated by π in the ϕ coordinate is mapped to the excision sphere S_e on the same patch to compute the equal mass data when the elliptic equations are solved. This amounts to imposing the π-rotation symmetry about the center of mass which is located at $(r, \theta, \phi) = (d_s/2, \pi/2, 0)$. In this test the number of grid points is chosen as D1–D4 in Table III, the separation between the coordinate centers of two BHs (a distance between the

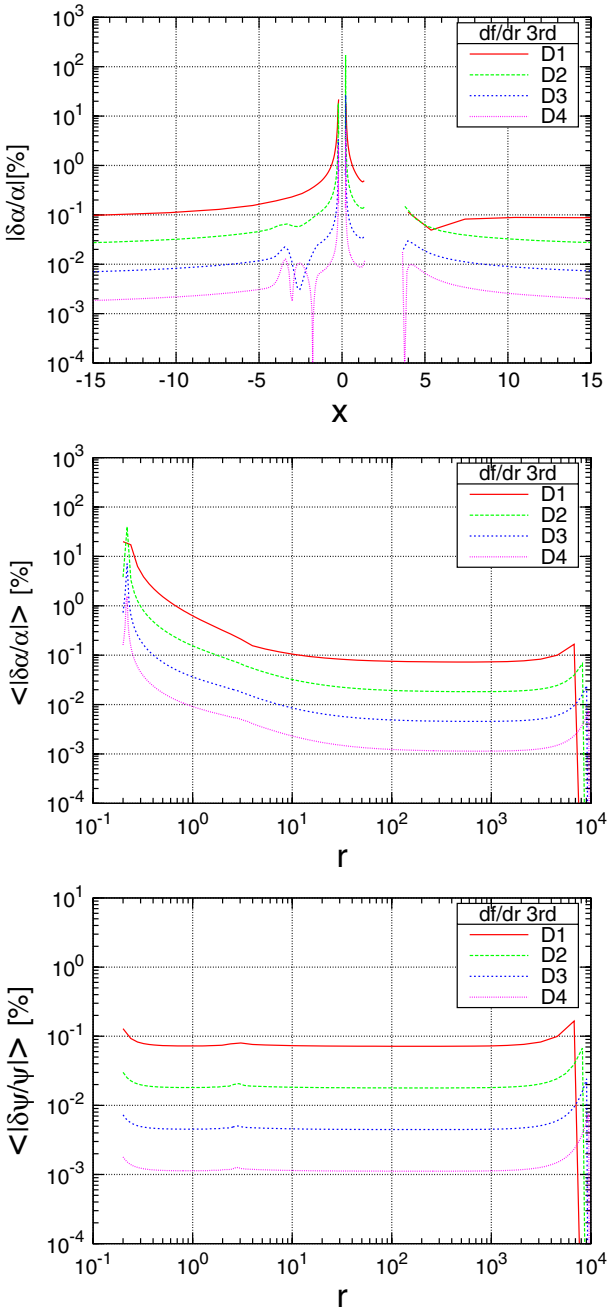

FIG. 4 (color online). Same as Fig. 3 but for the equal mass BBH data calculated on a single patch, Fig. 5. Top panel: Plots for the fractional errors in the lapse $\delta\alpha/\alpha$ for the equal mass BBH data along the positive x axis. Middle panel: Averaged fractional errors in the lapse $\langle|\delta\alpha/\alpha|\rangle$. Bottom panel: Averaged fractional errors in the conformal factor $\langle|\delta\psi/\psi|\rangle$. Plots show the errors by changing the number of all grid points as D1–D4 in Table III.

centers of S_a and S_e) is set as $d_s = 2.5$, the excision radius of the BH $r_a = 0.2$, the excision radius of the binary companion $r_e = 1.125$, and the mass parameters $M_1 = M_2 = 2r_a$. Hereafter, the third order finite difference formula is always used for computing the radial derivatives, as discussed in Sec. III B 1.

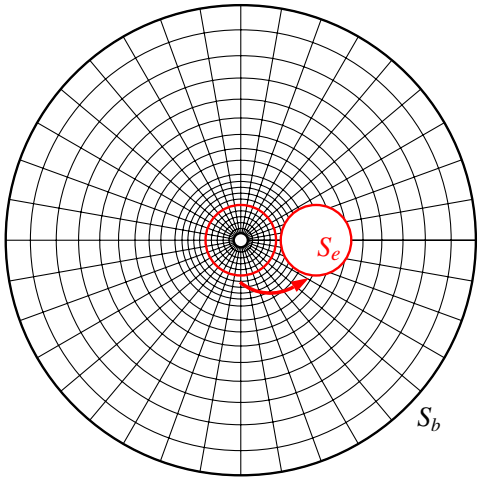

FIG. 5 (color online). The setup for a single coordinate grid patch for the calculation of equal mass BBHs. The radius of the coordinate patch does not reflect the actual size.

In the top panel of Fig. 4, fractional errors of the lapse $|\delta\alpha/\alpha|$ defined in Eq. (28) are plotted along the x axis near the BH. Because of the excision of the interior of the sphere S_e and of the use of Legendre expansion in the elliptic solver, a certain modulation is seen in the errors. Therefore, hereafter we show fractional errors averaged over the number of (θ_j, ϕ_k) grids points at a radius $r = r_i$ defined by

$$\left\langle \left| \frac{\delta\alpha}{\alpha} \right| \right\rangle := \frac{1}{\#(\mathcal{G}_i)} \sum_{\theta_j, \phi_k \in \mathcal{G}_i} \left| \frac{\alpha - \alpha_{\rm exact}}{\alpha_{\rm exact}} \right|, \qquad (29)$$

TABLE IV. Grid parameters used in convergence tests for nonequal mass BBH data solved on multiple coordinate patches. The separation of two BHs is set as $d_s = 2.5$.

Type	Patch	r_a	r_b	r_c	r_e	$N_r^{\rm f}$	$N_r^{\rm m}$	N_r	N_θ	N_ϕ	L
E1	COCP-1	0.2	10^4	1.25	1.125	16	20	64	24	24	12
	COCP-2	0.4	10^4	1.25	1.125	16	20	64	24	24	12
E2	COCP-1	0.2	10^4	1.25	1.125	32	40	128	48	48	12
	COCP-2	0.4	10^4	1.25	1.125	32	40	128	48	48	12
E3	COCP-1	0.2	10^4	1.25	1.125	64	80	256	96	96	12
	COCP-2	0.4	10^4	1.25	1.125	64	80	256	96	96	12
E4	COCP-1	0.2	10^4	1.25	1.125	128	160	512	192	192	12
	COCP-2	0.4	10^4	1.25	1.125	128	160	512	192	192	12
F1	COCP-1	0.2	10^2	1.25	1.125	16	20	48	24	24	12
	COCP-2	0.4	10^2	1.25	1.125	16	20	48	24	24	12
	ARCP	5.0	10^6	6.25	⋯	4	5	48	24	24	12
F2	COCP-1	0.2	10^2	1.25	1.125	32	40	96	48	48	12
	COCP-2	0.4	10^2	1.25	1.125	32	40	96	48	48	12
	ARCP	5.0	10^6	6.25	⋯	8	10	96	48	48	12
F3	COCP-1	0.2	10^2	1.25	1.125	64	80	192	96	96	12
	COCP-2	0.4	10^2	1.25	1.125	64	80	192	96	96	12
	ARCP	5.0	10^6	6.25	⋯	16	20	192	96	96	12
F4	COCP-1	0.2	10^2	1.25	1.125	128	160	384	192	192	12
	COCP-2	0.4	10^2	1.25	1.125	128	160	384	192	192	12
	ARCP	5.0	10^6	6.25	⋯	32	40	384	192	192	12

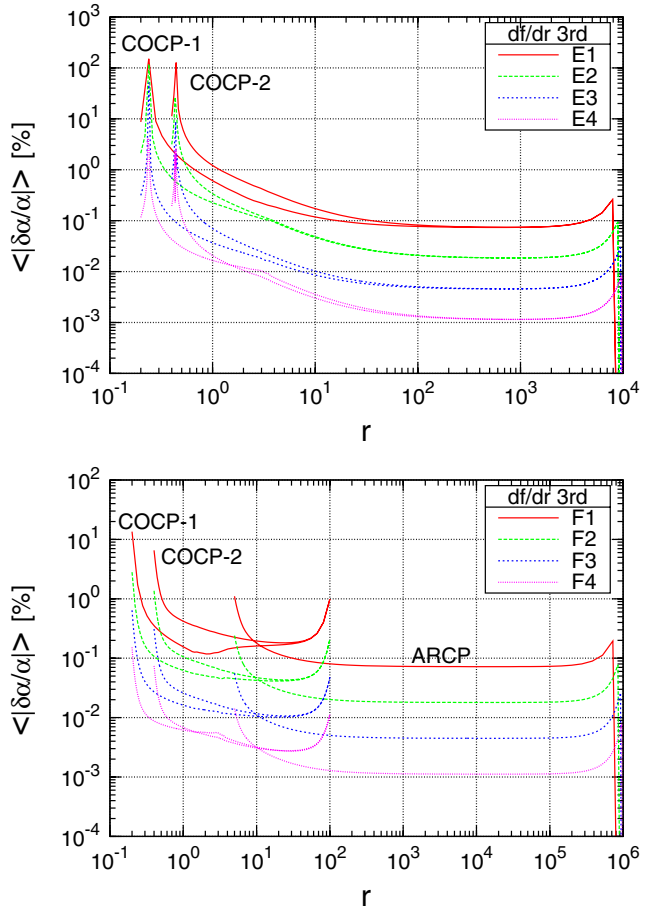

FIG. 6 (color online). Same as Fig. 3, but averaged fractional errors in the lapse $\langle|\delta\alpha/\alpha|\rangle$ are plotted along the radial coordinate r for the nonequal mass BBH data calculated on multiple patches, Fig. 1. Top panel: Data computed on two multiple patches by changing the number of grid points as E1–E4 in Table III. Bottom panel: Data computed on three multiple patches by changing the number of all grid points as F1–F4 in Table III.

where, writing a grid point (r_i, θ_j, ϕ_k) by p, we define a set $\mathcal{G}_i$ by $\mathcal{G}_i := \{(r_i, \theta_j, \phi_k) | p \in V \setminus S_e^{\rm in}$ and $r_i = {\rm const}\}$, where $S_e^{\rm in}$ is an interior domain of S_e. Then, $\#(\mathcal{G}_i)$ is the number of points included in $\mathcal{G}_i$.

The averaged fractional errors for the lapse are plotted along the radial coordinate r in the middle panel of Fig. 3. As expected, second order convergence is observed when the grid points are increased as D1-D4. In the figure, it is seen that a couple of grid points in the vicinity of the BH boundary S_a have (averaged) errors as large as $\sim 1\%$, even for the highest resolution D4. This is due to our choice of the boundary $r_a = M/2$, which is the same as the single BH test in the previous section. With this choice, the value of α becomes negative at the BH excision radius $r = r_a$. Hence, the fractional error diverges at radii r (depending on θ and ϕ), where α crosses zero, even though the grid points are slightly off from the zeros. In this way, the worst possible error in computation for the metric potentials of

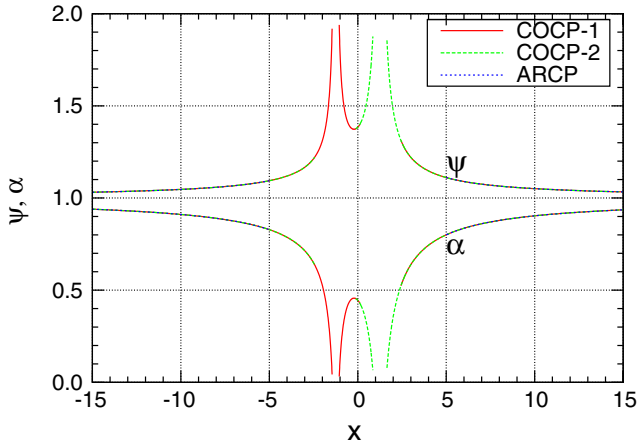

FIG. 7 (color online). Plots for the conformal factor ψ and the lapse α, computed on the three multi-patches. The model is the same as that in the bottom panel of Fig. 6.

BHs is estimated. Even near the radius for $\alpha = 0$, the second order convergence is maintained, as seen in the figure. We also show, in the bottom panel of Fig. 3, averaged fractional errors for the conformal factor ψ. The value of ψ is about 2 near $r = r_a$, and for such a potential the convergence of the solution is almost uniform in all radii, as observed.

3. Nonequal mass BBH computed with multiple patches

A convergence test for nonequal mass BBH data using multi-coordinate patches discussed in Sec. II is performed with grid parameters presented in Table IV. In the first example, BBH data are computed on two COCPs whose boundary radius $r = r_b$ is taken to be large enough to reach the asymptotic region $r_b = 10^4$. In this computation, the number of grid points is chosen as E1–E4 in Table IV, in which the resolution in radial grids r_i in the region $r > r_c$ is increased by $44/24$ times the corresponding level of resolution for the equal mass BBH case, D1–D4. Separation between the coordinate centers of two BHs is set as $d_s = 2.5$, the excision radius and mass parameter of the first BH are $r_a = 0.2$ with $M_1 = 2r_a = 0.4$, and those of the second BH are $r_a = 0.4$ with $M_2 = 2r_a = 0.8$. The results of the averaged fractional error in the top panel of Fig. 6 are similar to those of the equal mass BBH case in Fig. 4, which proves our multiple patch methods work accurately as expected.

Finally, the BBH data are computed using three multiple patches, as shown in Fig. 1. The number of grid points is chosen as F1–F4 in Table IV, and the separation is set as $d_s = 2.5$. In the bottom panel of Fig. 6, the results for the averaged fractional errors $\langle|\delta\alpha/\alpha|\rangle$ are shown. In this computation we decreased the values of the mass parameter to $M_1 = 0.8 \times 2r_a = 0.32$ with $r_a = 0.2$ for the first BH and $M_2 = 0.8 \times 2r_a = 0.64$ with $r_a = 0.4$ for the second BH, so that the lapse α is always positive even near the BH excision boundary at $r = r_a$. As seen in the

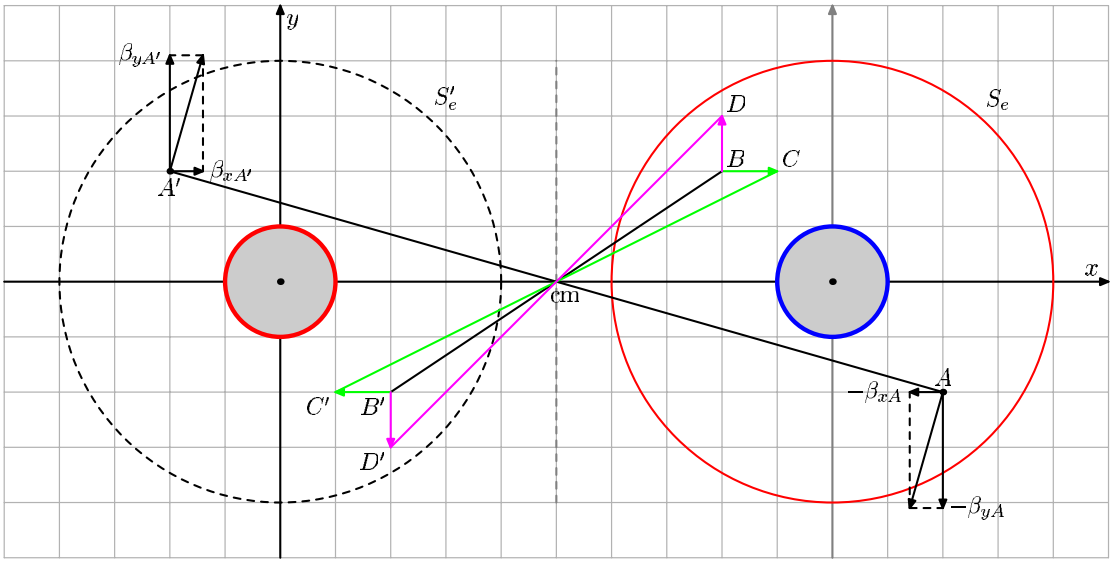

FIG. 8 (color online). Schematic figure for the π rotation symmetry of x and y components of the shift from a region inside S'_e to the excised region inside the sphere S_e.

figure, the error near the BH boundary is decreased to about $1/10$ of the previous case, although the resolutions near the BH are the same. In Fig. 7 we present the plots of the potentials α and ψ for the same model to show a smooth transition of potentials from one patch to the other.

IV. INITIAL DATA FOR BINARY BLACK HOLES

Finally, we present examples for initial data sets for equal mass BBHs, which have been widely used for BBH merger simulations in the literature. Among several formulations for computing such data sets (see, e.g., [1,7] and references therein), we adopt the Isenberg-Wilson-Mathews (IWM) formulation. In this section we show the solutions computed from two different types of boundary conditions. The first are simple boundary conditions used in our previous paper [13]. The second are the apparent horizon boundary conditions, which have been used to compute quasicircular initial data for BBHs (see, e.g., [7,8,27,28]).

A. IWM formulation and boundary conditions

IWM formulation has been widely used for constructing quasiequilibrium initial data for binary compact objects. We summarize the basic equations below (for more details see, e.g., [7,22,23]). The spacetime metric on Σ_t is written in $3+1$ form as

$$ds^2 = g_{\mu\nu}dx^\mu dx^\nu = -\alpha^2 dt^2 + \gamma_{ij}(dx^i + \beta^i dt)(dx^j + \beta^j dt), \tag{30}$$

where the spatial three-metric γ_{ij} on the slice Σ_t is assumed to be conformally flat, $\gamma_{ij} = \psi^4 f_{ij}$. Here, field variables ψ, α, and β^i are the conformal factor, lapse, and shift vector, respectively. We also assume maximal slicing to Σ_t, so that the trace of the extrinsic curvature $K_{ij} := -\frac{1}{2\alpha}(\mathcal{L}_t\gamma_{ij} - \mathcal{L}_\beta\gamma_{ij})$ vanishes. Then, writing its trace-free part A_{ij}, the conformally rescaled quantity $\tilde{A}_{ij}$ becomes

$$\tilde{A}_{ij} = \frac{1}{2\alpha}\left(\partial_i\tilde{\beta}_j + \partial_j\tilde{\beta}_i - \frac{2}{3}f_{ij}\partial_k\tilde{\beta}^k\right), \tag{31}$$

where the derivative ∂_i is associated with the flat metric f_{ij}, and conformally rescaled quantities with tildes are defined by $\tilde{A}_i{}^j = A_i{}^j$ and $\tilde{\beta}^i = \beta^i$, whose indices are lowered (raised) by f_{ij} (f^{ij}). The system to be solved, which includes Hamiltonian and momentum constraints and the spatial trace of Einstein's equations, becomes

$$\Delta\psi = -\frac{\psi^5}{8}\tilde{A}_{ij}\tilde{A}^{ij}, \tag{32}$$

$$\Delta\tilde{\beta}_i = -2\alpha\tilde{A}_i{}^j\partial_j\ln\frac{\psi^6}{\alpha} - \frac{1}{3}\partial_i\partial_j\tilde{\beta}^j, \tag{33}$$

$$\Delta(\alpha\psi) = \tfrac{7}{8}\alpha\psi^5\tilde{A}_{ij}\tilde{A}^{ij}, \tag{34}$$

where $\Delta := \partial_i\partial^i$ is a flat Laplacian. It is noted that, for the shift equation (33), the Cartesian components are solved on the spherical coordinates.

As a first set of boundary conditions at the BH excision boundary S_a and at the boundary of computational domain S_b for the above system, Eqs. (32)–(34), we choose the following, for simplicity,

TABLE V. Boundary conditions and their parameters used in the computations for BBH initial data. The first column, Type, denotes the types of boundary conditions used, TU corresponds to Eq. (35), and AH corresponds to apparent horizon boundary conditions, Eqs. (36)–(38). The model of Fig. 9 is computed with a binary separation $d_s = 2.8$, a radius of BH excision $r_a = 0.1$, and a radius of binary excision $r_e = 1.3$. All the other models are computed with the parameter set D3 in Table III.

Type	n_1	n_0	Ω	Ω_s	Spin axis	Figures
TU	3.0	1.0	0.3	0.0	...	Figure 9
TU	2.2	0.005	0.08	0.0	...	Figure 10 and 11
AH	...	0.1	0.08	0.0	...	Figures 12–18
AH	...	0.005	0.08	0.0	...	Figures 12–16
AH	...	0.1	0.08	0.1	z axis	Figure 17
AH	...	0.1	0.08	0.1	y axis	Figure 18

$$\psi|_{r=r_a} = n_1, \quad \beta^i|_{r=r_a} = -\Omega\phi^i_{\rm cm}, \quad \alpha|_{r=r_a} = n_0 \ \text{and}$$
$$\psi|_{r=r_b} = 1.0, \quad \beta^i|_{r=r_b} = 0.0, \quad \alpha|_{r=r_b} = 1.0, \tag{35}$$

where n_1 and n_0 are arbitrary positive constants taken as $n_1 \gtrsim 2$ and $n_0 \lesssim 1$, $\phi^i_{\rm cm} = (-y_{\rm cm}, x_{\rm cm}, 0)$ is the rotational vector with respect to the center of mass of the binary associated with coordinates $(x_{\rm cm}, y_{\rm cm}, z_{\rm cm}) := (x - d, y, z)$, and Ω corresponds to the orbital angular velocity. The radius r_b is taken to be large enough, as in the test problems in Sec. III. Despite the fact that the boundary conditions above may be of the simplest type for the IWM formulation deduced for acquiring BBH data with nonzero orbital angular momentum in an asymptotically flat system, they capture the qualitative functional behavior of the unknown fields $\{\psi, \alpha, \beta^i\}$, as more realistic boundary conditions mentioned below. The solutions calculated from these boundary conditions (35) are compared with our previous code [13] that uses a different structure for the multiple spherical coordinate patches, as well as the solutions of different boundary conditions.

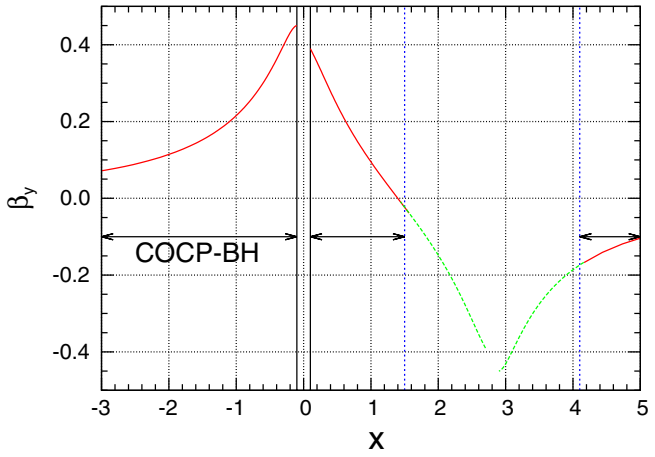

FIG. 9 (color online). Plots for the y component of shift vector along the x axis of BBH initial data computed from the boundary conditions (35). Parameters in the boundary conditions are the same as the solution presented in [13] as $n_1 = 3$, $n_0 = 0.1$, and $\Omega = 0.3$. Two BHs are located at $x = 0$ and $x = 2.8$. The region inside the excised sphere S_e is interpolated using π-rotation symmetry. Thin solid (black) vertical lines are the boundaries at S_a with the radius $r_a = 0.1$, and dotted (blue) vertical lines are the boundaries at S_e with the radius $r_e = 1.3$.

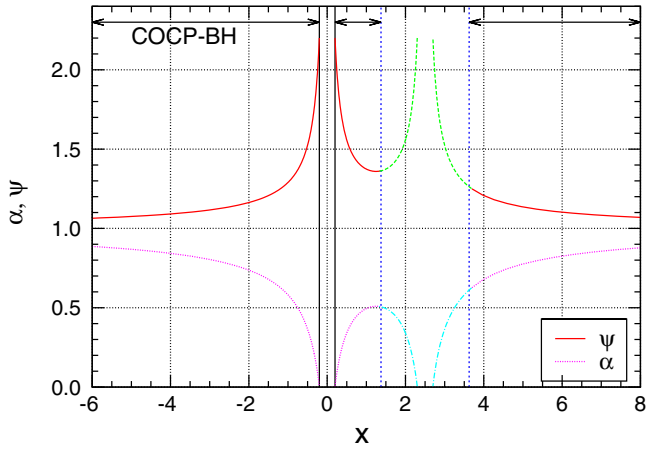

FIG. 10 (color online). Plots for the conformal factor ψ and lapse function α along the x axis of BBH initial data computed for the boundary condition (35). Parameters in the boundary conditions are chosen as $n_1 = 0.1$, $n_0 = 0.005$, and $\Omega = 0.08$. The two BHs are located at $x = 0$ and $x = 2.5$. The region inside the excised sphere S_e is interpolated using π-rotation symmetry. Thin solid (black) vertical lines are the boundaries at S_a with the radius $r_a = 0.2$, and dotted (blue) vertical lines are the boundaries at S_e with the radius $r_e = 1.125$.

For the second set, we impose more realistic boundary conditions at the BH boundary S_a, in particular, those that represent apparent horizons in equilibrium [7,8,27–29],

$$\left.\frac{\partial\psi}{\partial r} + \frac{\psi}{2r}\right|_{r=r_a} = -\frac{\psi^3}{4}K_{ij}s^i s^j, \tag{36}$$

$$\beta^i|_{r=r_a} = \frac{n_0}{\psi^2}s^i - \Omega\phi^i_{\rm cm} - \Omega_{\rm s}\phi^i_{\rm s}, \tag{37}$$

$$\alpha|_{r=r_a} = n_0, \tag{38}$$

where n_0 is an arbitrary positive constant for which we choose $n_0 \lesssim 0.1$, s^i is the unit normal to the sphere S_a, and $\Omega_{\rm s}$ represents the spin of each black hole. The vector $\phi^i_{\rm s}$ is the rotational vector with respect to the coordinate center of the BH that generates the BH spin. The spin axis is not necessarily parallel to the z axis. Demanding the sphere S_a to be an apparent horizon (AH) results in Eq. (36), while demanding the horizon to be in equilibrium results in Eq. (37).

For the present calculations for BBH initial data, we also assume π-rotation symmetry of the system around the

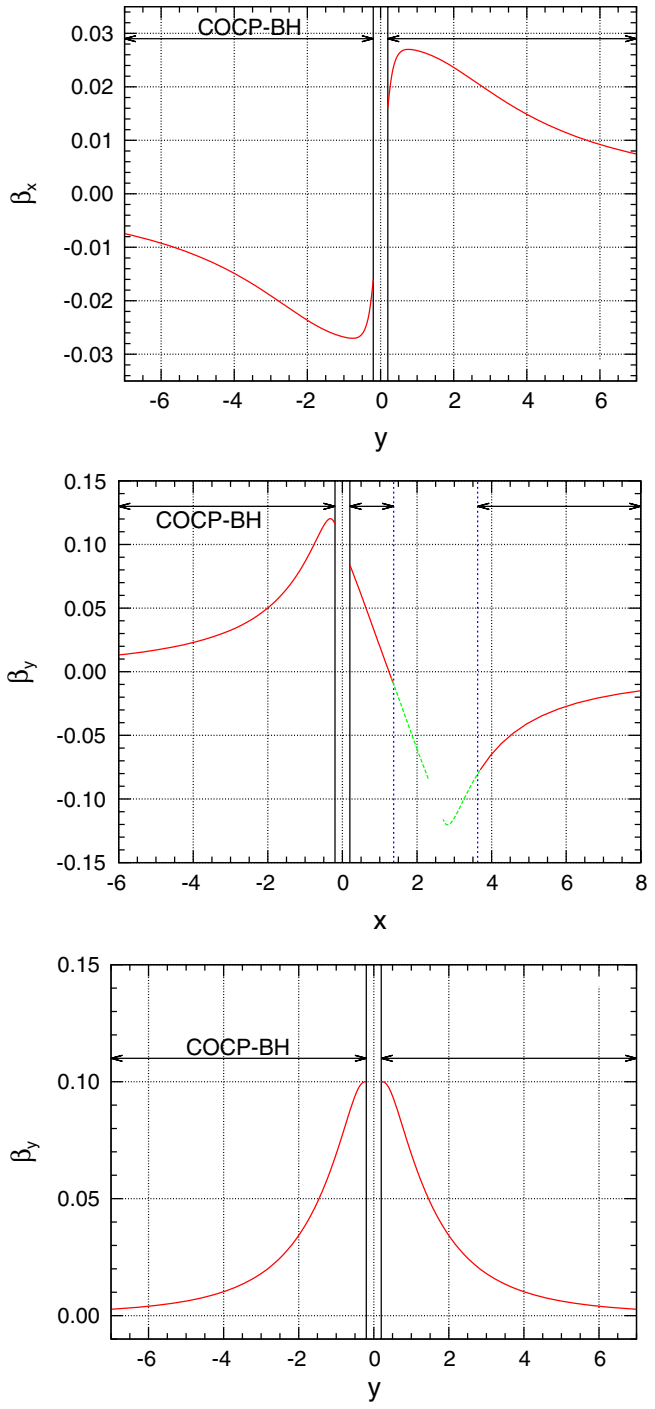

FIG. 11 (color online). Plots for the same model as Fig. 10 but for the components of the shift β_i. Top panel: β_x along the y axis. Middle panel: β_y along the x axis. Bottom panel: β_y along the y axis.

center of mass. That is, two BHs have equal masses and π-rotation symmetric spins, if any. In other words, the same boundary conditions are imposed on both BHs. In those cases, the single patch method discussed in Sec. III B 1 can be used for simplicity. As shown in Fig. 8, the metric potentials are mapped to the excised

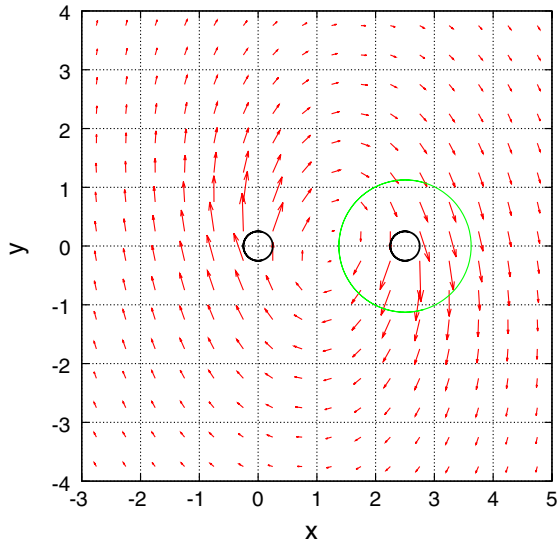

FIG. 12 (color online). Shift vector on the xy plane of the BBH initial data for the case with AH boundary conditions (36)–(38). Parameters in the boundary conditions are chosen as $n_0 = 0.1$, $\Omega = 0.08$, and $\Omega_s = 0$. The center of mass is located at $x = 1.25$ on the x axis. In the computation, the region inside the thick circle on the left centered at the origin with a radius $r_a = 0.2$, and the region inside the thin circle (green) with a radius $r_e = 1.125$ centered at $x = 2.5$ on the x axis are excised. The data inside the thin circle are interpolated by a symmetry. Note that the center of mass does not coincide with the origin of the x axis.

sphere S_e from the corresponding sphere S'_e, taking into account the parity of the variables with respect to the π rotation. As an example, the π-rotation symmetries of the shift components on the xy plane are shown schematically (the shift at A' is mapped to point A), together with the corresponding rules for the derivatives of a function along the x axis inside the excised sphere ($B'C'$ mapped to BC) and along the y axis ($B'D'$ mapped to BD). In terms of the center of mass coordinates, the π-rotation symmetries of the components of the shift vector become

$$\beta_x(-x_{\rm cm}, -y_{\rm cm}, z_{\rm cm}) = -\beta_x(x_{\rm cm}, y_{\rm cm}, z_{\rm cm}),$$
$$\beta_y(-x_{\rm cm}, -y_{\rm cm}, z_{\rm cm}) = -\beta_y(x_{\rm cm}, y_{\rm cm}, z_{\rm cm}).$$

The z component β_z is mapped as a scalar quantity.

Mapped quantities are used in the elliptic equation solver when the sources of surface integrals on the excised sphere S_e, Eq. (7), are evaluated. Also, at the end of each iteration step, obtained potentials $\{\psi, \alpha, \beta_i\}$ between the spheres S_a and S'_e are interpolated inside the sphere S_e following the same rules for the parity of each variable. At the end we have the solution at every point inside S_b and outside of the two black holes of radius r_a positioned at the origin and at $x = d_s$ on the x axis.

In the case of no spins $\Omega_s = 0$, or spins that are parallel to one of the coordinate axes, additional symmetries with respect to the $xy|_{\rm cm}$ plane occur. In our previous codes [11–13], a part or all of these symmetries were

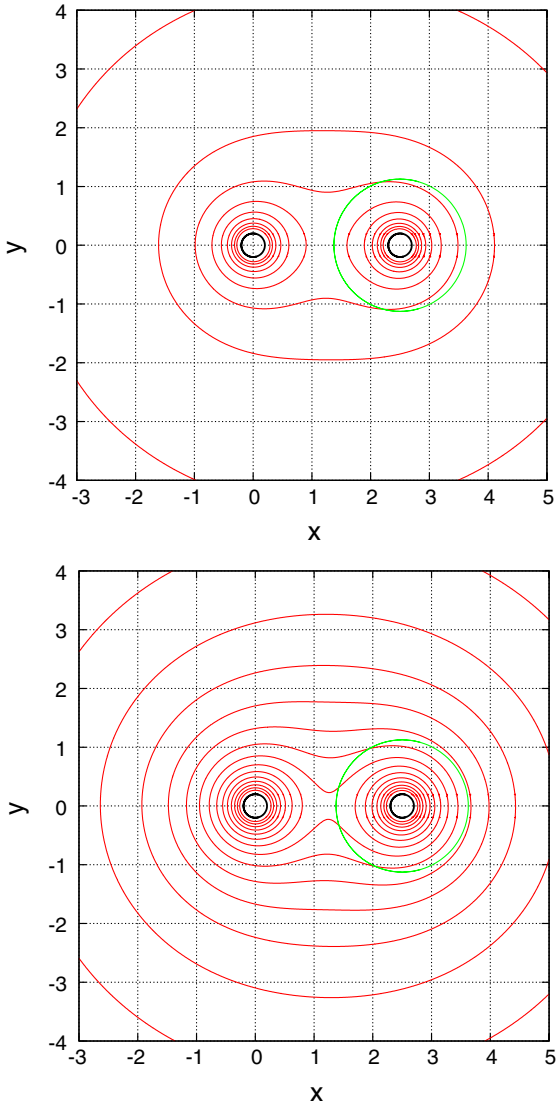

FIG. 13 (color online). Contour plots on the xy plane for the conformal factor ψ (top panel) and the lapse function α (bottom panel) for the same model as Fig. 12. For ψ we draw isolines from $\psi = 1.1$ to $\psi = 2.0$ with steps of 0.1. For α we draw isolines from $\alpha = 0.2$ to $\alpha = 0.9$ with steps of 0.05.

encoded in the elliptic solver. As a result, a computational domain was reduced by assuming the symmetries, and hence only a part of the whole hypersurface Σ_t was solved. In COCAL we are solving the whole Σ_t, and therefore such symmetries are satisfied in the solution within negligible numerical errors.

B. Solutions for BBH initial data

Finally, we present the BBH initial data sets computed from the above formulation with several parameter sets for the boundary conditions listed in Table V. As we concentrate on testing the COCAL code, we do not discuss the physical contents of the initial data sets much, but display the plots of the fields to check their behaviors.

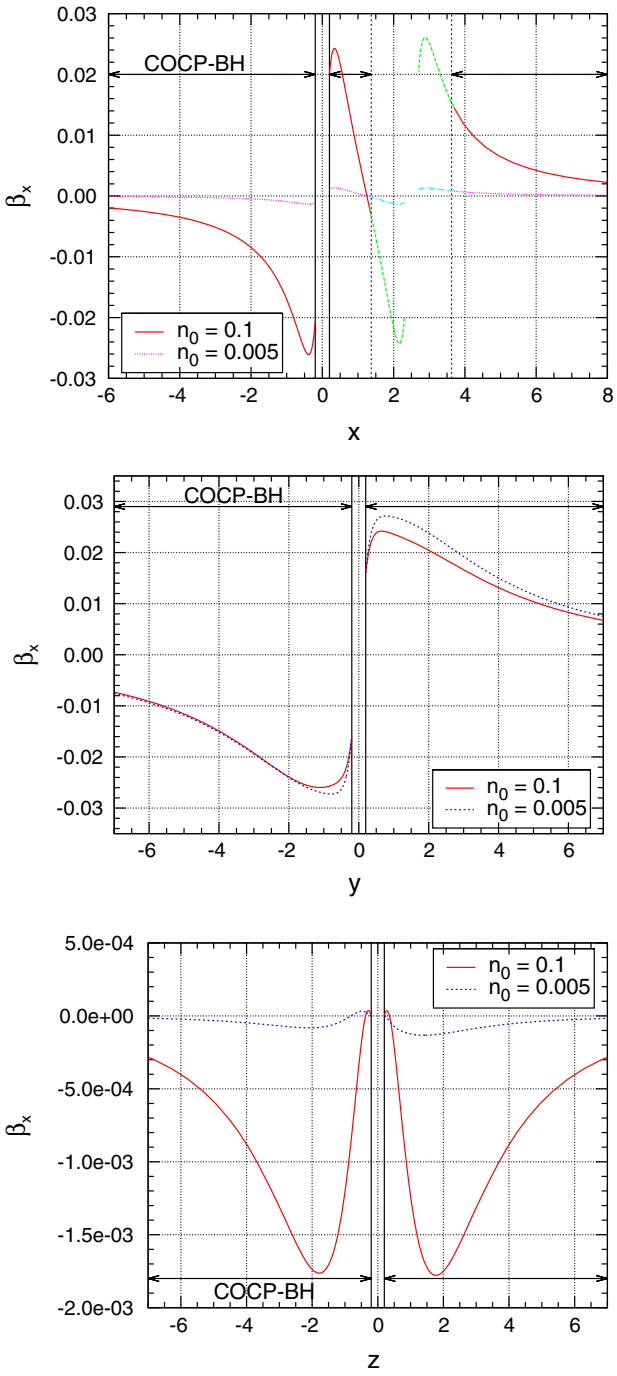

FIG. 14 (color online). Plots for the x component of the shift β_x of BBH initial data for the case with AH boundary conditions (36)–(38). Parameters in the boundary conditions are chosen as $n_0 = 0.1$ (solid red lines) and $n_0 = 0.005$ (dashed green lines), with $\Omega = 0.08$ and $\Omega_s = 0$. Top panel: Along the x axis. Middle panel: Along the y axis. Bottom panel: Along the z axis.

In Fig. 9 we calculated BBH data for the same model as shown in our previous paper [13] for a comparison. Parameters in the boundary condition (35) are chosen as $n_1 = 3.0$, $n_0 = 1.0$, and $\Omega = 0.3$. Only in this model, we choose the binary separation as $d_s = 2.8$ and the BH excision radius $r_a = 0.1$. We find the solution agrees well

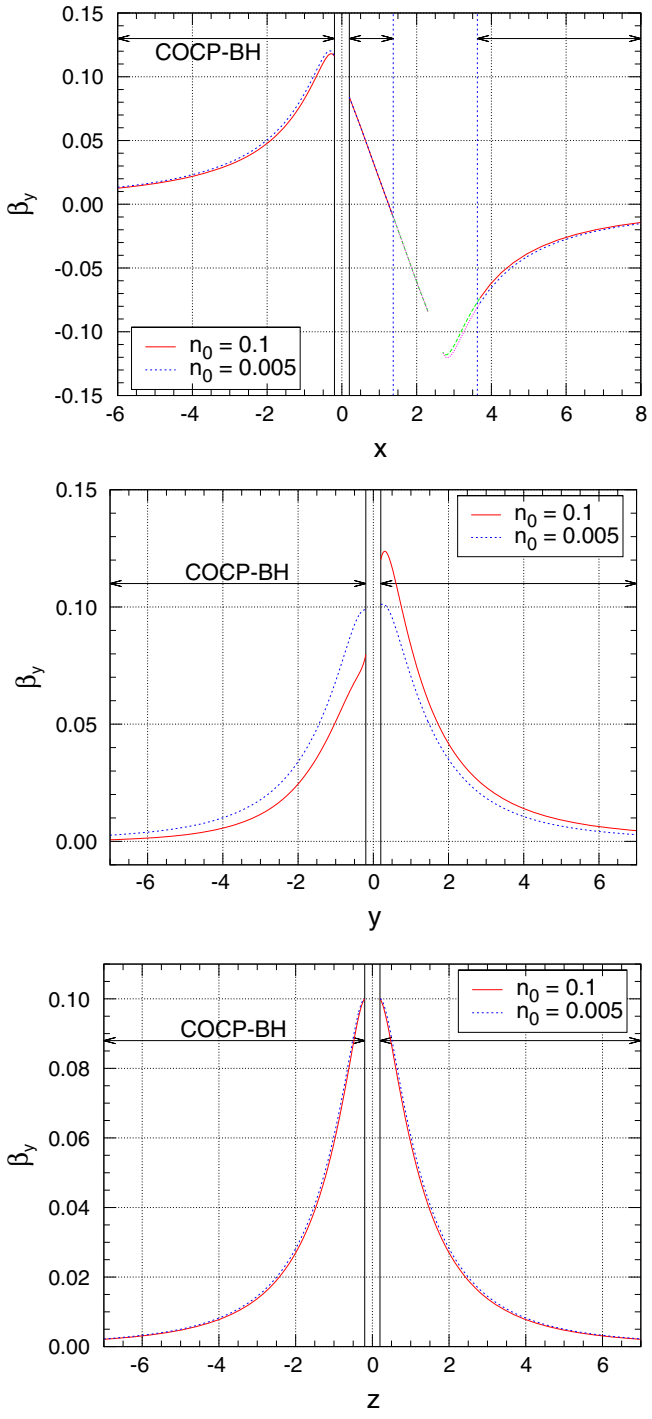

FIG. 15 (color online). Plots for the same model as Fig. 14 but for the y component of the shift β_y.

with Fig. 11 of [13] as expected, although the structures of coordinate patches are different in each code. In Figs. 10 and 11 we use the same boundary conditions (35) but with different parameters which may be more common values for the BBH data. In the computation the grid parameters used are D3 in Table III. When the value of Ω is increased, the magnitude of the lapse and the

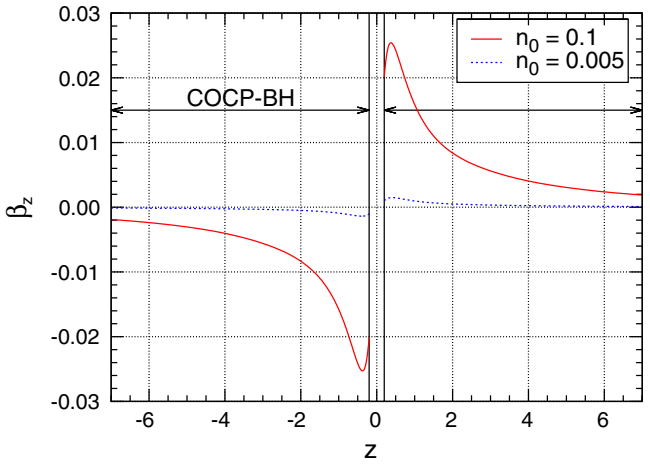

FIG. 16 (color online). Plots for the same model as Fig. 14 but for the z component of the shift β_z along the z axis.

conformal factor remain almost the same, while the magnitude of the components of the shift increase, with the functional behavior staying the same. For example, when $\Omega = 0.1$, β_x on the y axis varies between ± 0.05, while β_y on the x axis varies between ± 0.15. For these boundary conditions the code blows up when Ω is approximately greater than 0.2.[4]

Solutions for the BBH initial data with the AH boundary conditions (36)–(38) are shown in Figs. 12–16. In this calculation the resolution is D3 in Table III with $\Omega = 0.08$ and $\Omega_s = 0$. The shift vector plots and the contour plots of the conformal factor ψ and the lapse α in the xy plane are for the model with $n_0 = 0.1$. The behavior of the field ψ and α are analogous to those from the simple boundary condition (35) shown in Fig. 10. Because of the choice $\Omega_s = 0$, the solution satisfies xy-plane symmetry.

From a comparison between the results of parameters $n_0 = 0.1$ and 0.005 shown in Figs. 14–16, it is found that the results with $n_0 = 0.005$ become more similar to the results of the first boundary conditions (35) shown in Fig. 11. For example, it is most evident in the plot for β_y along the y axis, Fig. 15 (middle panel) and Fig. 11 (bottom panel). This seems to be the correct behavior because the first term of Eq. (37) contributes less as the value of alpha (parameter n_0) becomes smaller.[5] However, we do not

[4]The convergence of the iteration is not improved by changing the convergence parameter c for this model. For the same model, the value of Ω for the circular solution becomes around 0.1. Note also that the iteration converges for $\Omega = 0.3$ for the first example of BBH data with the TU boundary condition, to which the larger value of the lapse $\alpha = 1$ is given at the BH boundary. Also, we can calculate solutions with AH boundary conditions for Ω values greater than 0.2, again by changing the value of the lapse function at the horizon. These suggest that the initial data with a higher velocity may be calculated in a different slicing using the COCAL code.

[5]For the first boundary conditions (35), the solution of the shift does not depend much on n_0 in the region $n_0 \lesssim 0.1$.

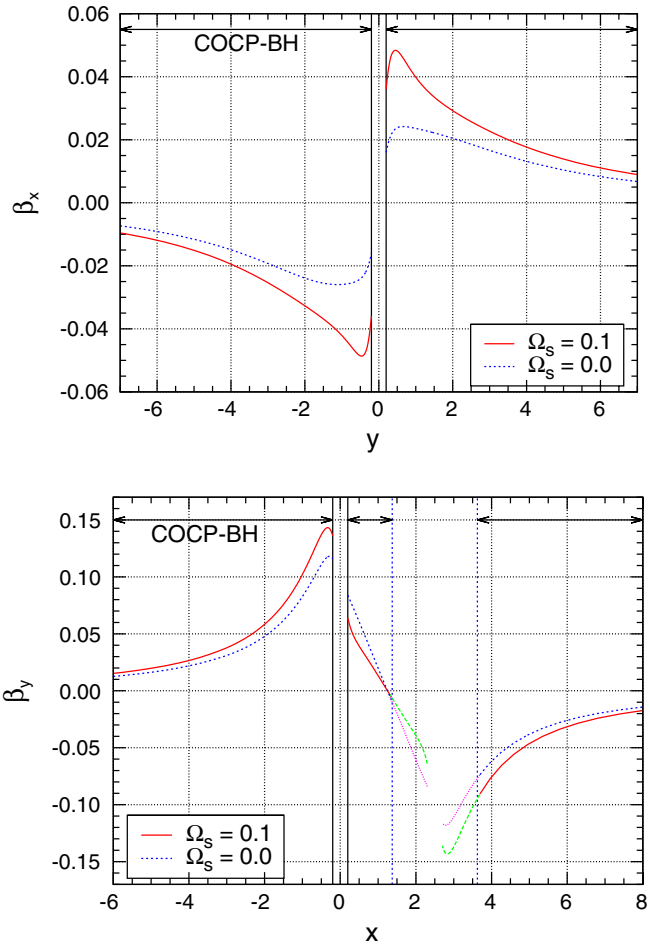

FIG. 17 (color online). Plots for the components of the shift β_i of BBH initial data for the case with AH boundary conditions (36)–(38). Parameters in the boundary conditions are chosen as $n_0 = 0.1$ with spin parameters $\Omega_s = 0$ (solid red lines) and $\Omega_s = 0.1$ (dashed green lines). The spins are aligned to the orbital angular momentum (i.e. parallel to the z axis). Top panel: β_x component along the y axis. Bottom panel: β_y component along the x axis. Solid red curves correspond to those in Figs. 14–16.

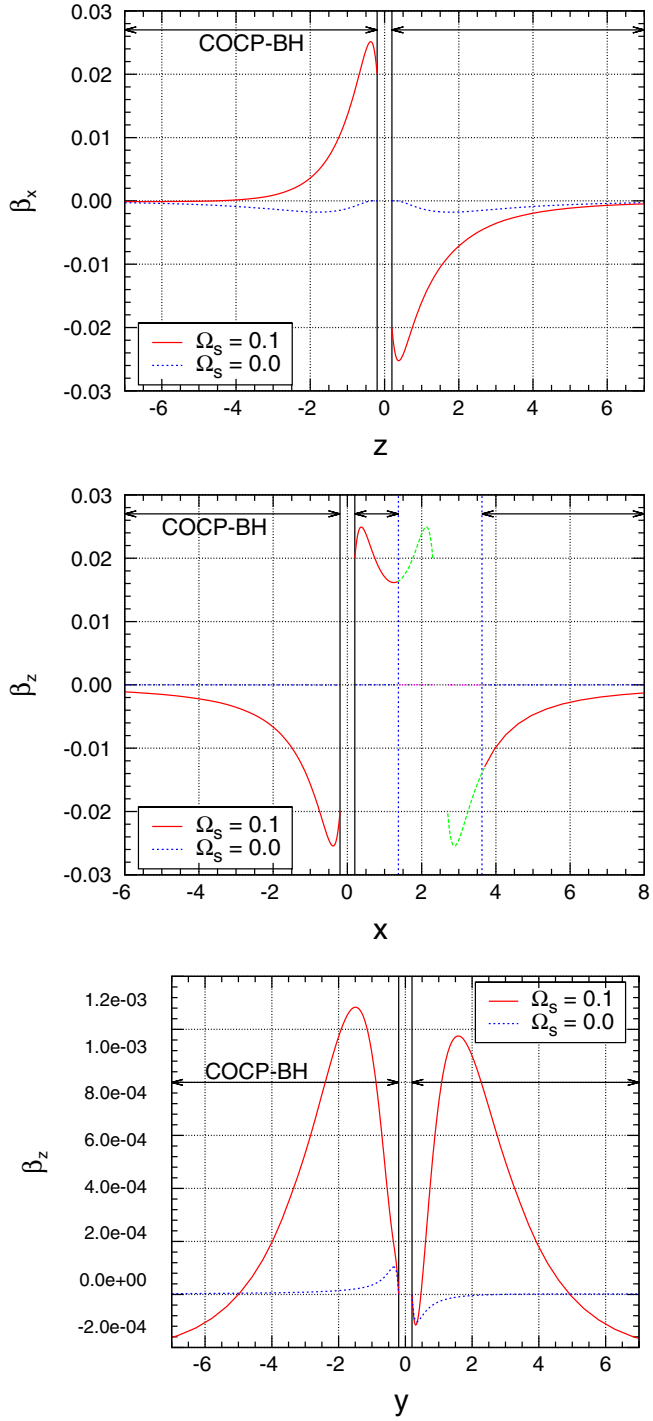

FIG. 18 (color online). Same as Fig. 17, but the direction of the spin is aligned parallel to the y axis. Top panel: β_x component along the z axis. Middle panel: β_z component along the x axis. Bottom panel: β_z component along the y axis.

expect a solution in the limit $n_0 \to 0$; in fact, for both types of boundary conditions iterations diverge when $n_0 \lesssim 0.004$ in the COCAL code.

All of the solutions above have zero black hole spins. Setting spins $\Omega_s = 0.1$ in the same direction as the orbital motion (rotation on the xy plane) we get a solution similar to Figs. 14–16, except for β_x along the y axis and β_y along the x axis. The results with and without spins are compared in Fig. 17 for the case with $n_0 = 0.1$. For more general black hole spins we obtain Fig. 18, where we have taken a spin $\Omega_s = 0.1$ along the y axis. In these plots for the solutions with the spins, we confirm that all components of the shift vectors behave correctly along each coordinate axis.

V. DISCUSSION

Although the numerical method presented in this paper may seem similar to the one presented in our previous paper [13], the robustness of the convergence and the control of the numerical errors are greatly improved. In some cases, the previous method failed to compute a continuous solution at the interface between multiple patches during the iterations, and therefore a convergence

to a solution could not be achieved. A major reason for this failure turned out to be a lack of enough overlap region between coordinate patches. The numerical errors of the field variables are relatively larger near the boundary of the computational domain as seen, for example, in Fig. 6. Hence, if the overlap region is small, those potentials with larger numerical errors overlap, which seems to cause the nonconvergence of the iteration. In the COCAL code the overlap region is almost as large as the whole domain for the two coordinate patch configuration, and is large enough even for the three coordinate patch configuration. So far we have not observed a discontinuous behavior of a field in the solutions of COCAL.

The COCAL code currently runs only on a serial processor, which is sufficient to maintain the accuracy presented in Sec. III. In the computation for BBH initial data shown in Sec. IV, the size of the main memory and CPU times per 1 iteration cycle used by COCAL are about 800 MB 50 sec for D3 grid and 6 GB 8 min for D4 grid, and around 50–150 iterations are needed for a convergence, where the iterations start from an initial guess $\psi = \alpha = 1$ and $\beta_i = 0$. Because we use second order accurate formulas in COCAL, we can decrease the numerical error by 2 orders of magnitude with 10 times as many grid points in each direction, that is, 10^3 times as many grid points in total. Considering specs of common parallel computer systems it seems to be feasible to achieve this accuracy by parallelizing COCAL. We have started to develop a prototype of such parallelized COCAL code, whose results would be presented elsewhere.

The most advantageous feature of COCAL is its simplicity in coding. This helps the users to introduce more complex physics on top of the current code. For example, we have developed subroutines for solving spatially conformally flat data (IWM formulation) first, and later added subroutines for solving nonconformally flat data (waveless formulation) on top. In the same way, it will be straightforward to incorporate subroutines to solve electromagnetic fields, which enables us to investigate, for example, magnetar models. We are developing codes for computing various kinds of astrophysically realistic equilibriums and quasiequilibrium data and to provide the results to applications including initial data for numerical relativity simulations. We plan to make the COCAL code and computed initial data sets available for public use in the near future.

ACKNOWLEDGMENTS

This work was supported by JSPS Grant-in-Aid for Scientific Research(C) No. 23540314 and No. 22540287, and MEXT Grant-in-Aid for Scientific Research on Innovative Area No. 20105004. K. U. thanks Charalampos Markakis, Noriyuki Sugiyama, and members of LUTH at the Paris Observatory for discussions.

APPENDIX A: COMPUTATIONS FOR HOMOGENEOUS SOLUTIONS

In this appendix we show a concrete derivation for the homogeneous solution $\chi(x)$ used in the elliptic solver (6) for two cases, one with the Neumann boundary condition at the inner boundary sphere S_a and the Dirichlet boundary condition at the outer boundary sphere S_b, and the other with Dirichlet conditions at both S_a and S_b. We summarize the other cases in Appendix B.

As explained in Sec. II C, the solution of $\mathcal{L}\Phi = S$ is written $\Phi(x) = \chi(x) + \Phi_{\rm INT}(x)$ to impose certain boundary conditions at the two spheres S_a and S_b. In order to do so, the homogeneous solution χ of the Laplacian is split into two functions $\chi_a(x)$ and $\chi_b(x)$ as $\chi(x) = \chi_a(x) + \chi_b(x)$. Both are solutions of Laplace equations, one for the exterior of the sphere S_a and the other for the interior of the sphere S_b (see Fig. 19). Since r^ℓ, $r^{-\ell-1}$ are the solutions of the radial part of the Laplacian, the contribution χ_a is taken to be a series of $r^{-\ell-1}$, while the contribution χ_b is taken to be a series of r^ℓ. Therefore, we write

$$\chi_a(x) = \frac{1}{4\pi}\sum_{\ell=0}^{\infty}\sum_{m=0}^{\ell}\epsilon_m\frac{(\ell-m)!}{(\ell+m)!}P_\ell^m(\cos\theta)r^{-\ell-1} \times[A_{\ell m}\cos(m\phi) + B_{\ell m}\sin(m\phi)], \tag{A1}$$

$$\chi_b(x) = \frac{1}{4\pi}\sum_{\ell=0}^{\infty}\sum_{m=0}^{\ell}\epsilon_m\frac{(\ell-m)!}{(\ell+m)!}P_\ell^m(\cos\theta)r^{\ell} \times[C_{\ell m}\cos(m\phi) + D_{\ell m}\sin(m\phi)], \tag{A2}$$

where $A_{\ell m}$, $B_{\ell m}$, $C_{\ell m}$, and $D_{\ell m}$ are constants. One common choice is

$$\left.\frac{\partial\Phi}{\partial r}\right|_{r=r_a} = \left.\frac{\partial\Phi_{\rm BC}}{\partial r}\right|_{r=r_a}, \tag{A3}$$

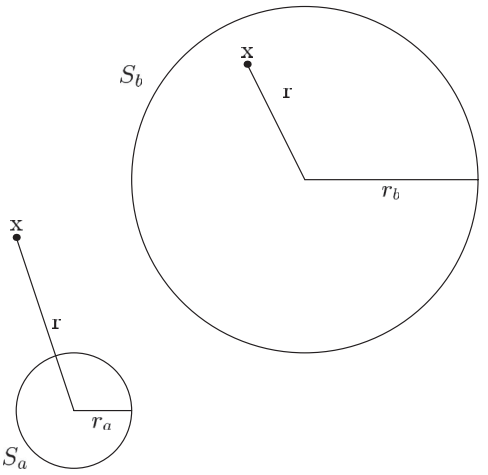

FIG. 19. Surface integral for the exterior and the interior of a sphere.

$$\Phi|_{r=r_b} = \Phi_{\rm BC}|_{r=r_b}, \tag{A4}$$

where $\frac{\partial\Phi_{\rm BC}}{\partial r}|_{r=r_a}$ and $\Phi_{\rm BC}|_{r=r_b}$ are known functions defined at boundaries S_a and S_b, respectively; we consider the case with the Neumann boundary condition at the inner surface S_a and the Dirichlet boundary condition at the outer one S_b.

From boundary condition (A3), with the use of Eqs. (A1), (A2), (6), and (7), and the orthogonality relations

$$\int_0^\pi P_\ell^m(\cos\theta)P_{\ell'}^m(\cos\theta)\sin\theta d\theta = \frac{2}{2\ell+1}\frac{(\ell+m)!}{(\ell-m)!}\delta_{\ell\ell'},\quad \int_0^{2\pi}\sin(m\phi)\cos(m'\phi)d\phi = 0,\quad \int_0^{2\pi}\cos(m\phi)\cos(m'\phi)d\phi = \frac{2\pi}{\epsilon_m}\delta_{mm'},$$

we get

$$-\frac{\ell+1}{r_a^{\ell+2}}A_{\ell m} + \ell r_a^{\ell-1}C_{\ell m} = (2\ell+1)\int_0^\pi\int_0^{2\pi}\left(\frac{\partial\Phi_{\rm BC}}{\partial r} - \frac{\partial\Phi_{\rm INT}}{\partial r}\right)_{r=r_a} P_\ell^m(\cos\theta)\cos(m\phi)d\Omega, \tag{A5}$$

$$-\frac{\ell+1}{r_a^{\ell+2}}B_{\ell m} + \ell r_a^{\ell-1}D_{\ell m} = \frac{2(2\ell+1)}{\epsilon_m}\int_0^\pi\int_0^{2\pi}\left(\frac{\partial\Phi_{\rm BC}}{\partial r} - \frac{\partial\Phi_{\rm INT}}{\partial r}\right)_{r=r_a} P_\ell^m(\cos\theta)\sin(m\phi)d\Omega. \tag{A6}$$

Using now boundary condition (A4) and again the same equations, we get

$$r_b^{-\ell-1}A_{\ell m} + r_b^\ell C_{\ell m} = (2\ell+1)\int_0^\pi\int_0^{2\pi}(\Phi_{\rm BC} - \Phi_{\rm INT})_{r=r_b}P_\ell^m(\cos\theta)\cos(m\phi)d\Omega, \tag{A7}$$

$$r_b^{-\ell-1}B_{\ell m} + r_b^\ell D_{\ell m} = \frac{2(2\ell+1)}{\epsilon_m}\int_0^\pi\int_0^{2\pi}(\Phi_{\rm BC} - \Phi_{\rm INT})_{r=r_b}P_\ell^m(\cos\theta)\sin(m\phi)d\Omega. \tag{A8}$$

The system of four equations with respect to $A_{\ell m}$, $B_{\ell m}$, $C_{\ell m}$, $D_{\ell m}$ can be solved to yield

$$\begin{aligned}
A_{\ell m}\left[1+\frac{\ell}{\ell+1}\left(\frac{r_a}{r_b}\right)^{2\ell+1}\right] &= \frac{\ell(2\ell+1)}{\ell+1}r_a^{\ell+1}\left(\frac{r_a}{r_b}\right)^\ell\int_{S_b}(\Phi_{\rm BC}-\Phi_{\rm INT})P_\ell^m(\cos\theta)\cos(m\phi)d\Omega\\
&\quad -\frac{2\ell+1}{\ell+1}r_a^{\ell+2}\int_{S_a}\left(\frac{\partial\Phi_{\rm BC}}{\partial r}-\frac{\partial\Phi_{\rm INT}}{\partial r}\right)P_\ell^m(\cos\theta)\cos(m\phi)d\Omega,\\
C_{\ell m}\left[1+\frac{\ell}{\ell+1}\left(\frac{r_a}{r_b}\right)^{2\ell+1}\right] &= (2\ell+1)b^{-\ell}\int_{S_b}(\Phi_{\rm BC}-\Phi_{\rm INT})P_\ell^m(\cos\theta)\cos(m\phi)d\Omega\\
&\quad +\frac{2\ell+1}{\ell+1}\frac{r_a^2}{r_b^{\ell+1}}\left(\frac{r_a}{r_b}\right)^\ell\int_{S_a}\left(\frac{\partial\Phi_{\rm BC}}{\partial r}-\frac{\partial\Phi_{\rm INT}}{\partial r}\right)P_\ell^m(\cos\theta)\cos(m\phi)d\Omega,\\
B_{\ell m}\left[1+\frac{\ell}{\ell+1}\left(\frac{r_a}{r_b}\right)^{2\ell+1}\right] &= \frac{2\ell(2\ell+1)}{(\ell+1)\epsilon_m}r_a^{\ell+1}\left(\frac{r_a}{r_b}\right)^\ell\int_{S_b}(\Phi_{\rm BC}-\Phi_{\rm INT})P_\ell^m(\cos\theta)\sin(m\phi)d\Omega\\
&\quad -\frac{2(2\ell+1)}{(\ell+1)\epsilon_m}r_a^{\ell+2}\int_{S_a}\left(\frac{\partial\Phi_{\rm BC}}{\partial r}-\frac{\partial\Phi_{\rm INT}}{\partial r}\right)P_\ell^m(\cos\theta)\sin(m\phi)d\Omega,\\
D_{\ell m}\left[1+\frac{\ell}{\ell+1}\left(\frac{r_a}{r_b}\right)^{2\ell+1}\right] &= \frac{2(2\ell+1)}{\epsilon_m}r_b^{-\ell}\int_{S_b}(\Phi_{\rm BC}-\Phi_{\rm INT})P_\ell^m(\cos\theta)\sin(m\phi)d\Omega\\
&\quad +\frac{2(2\ell+1)}{(\ell+1)\epsilon_m}\frac{r_a^2}{r_b^{\ell+1}}\left(\frac{r_a}{r_b}\right)^\ell\int_{S_a}\left(\frac{\partial\Phi_{\rm BC}}{\partial r}-\frac{\partial\Phi_{\rm INT}}{\partial r}\right)P_\ell^m(\cos\theta)\sin(m\phi)d\Omega.
\end{aligned}$$

Substituting the above in Eqs. (A1) and (A2), we have

$$\begin{aligned}
\chi_a(x)+\chi_b(x) = \frac{1}{4\pi}\sum_{\ell=0}^\infty\sum_{m=0}^\ell\epsilon_m\frac{(\ell-m)!}{(\ell+m)!}P_\ell^m(\cos\theta)\Bigg\{&(2\ell+1)\left(\frac{r_a}{r_b}\right)^\ell\frac{(\frac{r}{r_a})^\ell+\frac{\ell}{\ell+1}(\frac{r_a}{r})^{\ell+1}}{1+\frac{\ell}{\ell+1}(\frac{r_a}{r_b})^{2\ell+1}}\\
&\times\int_{S_b}[\Phi_{\rm BC}-\Phi_{\rm INT}]P_\ell^m(\cos\theta')\cos[m(\phi-\phi')]d\Omega' + \frac{-2\ell-1}{\ell+1}\frac{r_a^{\ell+2}}{r_b^{\ell+1}}\frac{(\frac{r_b}{r})^{\ell+1}-(\frac{r}{r_b})^\ell}{1+\frac{\ell}{\ell+1}(\frac{r_a}{r_b})^{2\ell+1}}\\
&\times\int_{S_a}\left(\frac{\partial\Phi_{\rm BC}}{\partial r}-\frac{\partial\Phi_{\rm INT}}{\partial r}\right)P_\ell^m(\cos\theta')\cos[m(\phi-\phi')]d\Omega'\Bigg\}.
\end{aligned} \tag{A9}$$

Similarly, when we have Dirichlet boundary conditions on both the inner and the outer spheres S_a and S_b,

$$\Phi|_{r=r_a} = \Phi_{\rm BC}|_{r=r_a}, \tag{A10}$$

$$\Phi|_{r=r_b} = \Phi_{\rm BC}|_{r=r_b}, \tag{A11}$$

the contribution to the potential will be

$$\begin{aligned}\chi_a(x) + \chi_b(x) = \frac{1}{4\pi}\sum_{\ell=0}^{\infty}\sum_{m=0}^{\ell}\epsilon_m\frac{(\ell-m)!}{(\ell+m)!}P_\ell^m(\cos\theta)\Bigg\{&(2\ell+1)\left(\frac{r_a}{r_b}\right)^\ell\frac{(\frac{r}{r_a})^\ell - (\frac{r_a}{r})^{\ell+1}}{1-(\frac{r_a}{r_b})^{2\ell+1}}\\ &\times\int_{S_b}[\Phi_{\rm BC}-\Phi_{\rm INT}]P_\ell^m(\cos\theta')\cos[m(\phi-\phi')]d\Omega' + (2\ell+1)\left(\frac{r_a}{r_b}\right)^{\ell+1}\frac{(\frac{r_b}{r})^{\ell+1} - (\frac{r}{r_b})^\ell}{1-(\frac{r_a}{r_b})^{2\ell+1}}\\ &\times\int_{S_a}[\Phi_{\rm BC}-\Phi_{\rm INT}]P_\ell^m(\cos\theta')\cos[m(\phi-\phi')]d\Omega'\Bigg\}.\end{aligned} \tag{A12}$$

The final solution will be obtained from the iteration of

$$\Phi(x) = \chi(x) + \Phi_{\rm INT}(x) = \chi_a(x) + \chi_b(x) + \Phi_{\rm INT}(x),$$

where $\chi_a + \chi_b$ are taken from Eq. (A9) or (A12) depending on the boundary condition. For the other boundary value problem, $\chi(x) = \chi_a(x) + \chi_b(x)$ is modified accordingly as shown in the next appendix.

APPENDIX B: GREEN'S FUNCTIONS AND SURFACE INTEGRALS

In this appendix we present the explicit forms of the kernel functions denoted by $G^{\rm BC}(x, x')$ in Eq. (12), which appear in the surface integrals of the homogeneous solution $\chi(x)$ (11). Various types of boundary conditions are imposed on a spherical domain bounded by two concentric spheres S_a and S_b, and the corresponding kernel functions available in COCAL are tabulated in Table I. In [13] we used Green's functions $G^{\rm NB}(x, x')$, $G^{\rm DD}(x, x')$, and $G^{\rm ND}(x, x')$. Here we also construct $G^{\rm DN}(x, x')$, $G^{\rm NN}(x, x')$, $G^{\rm RD}(x, x')$.

The surface integral

$$\begin{aligned}\chi(x) = \frac{1}{4\pi}\int_{S_a\cup S_b}[&G^{\rm BC}(x,x')\nabla'^\alpha\hat\Phi(x')\\ &-\hat\Phi(x')\nabla'^\alpha G^{\rm BC}(x,x')]dS'_\alpha,\end{aligned} \tag{B1}$$

where $\hat\Phi(x') := \Phi^{\rm BC}(x') - \Phi_{\rm INT}(x')$, is written below for both the exterior and the interior problem. Noticing that dS'_α is pointing outward, S_a and S_b are the concentric spheres, and $\nabla'^\alpha f(x')dS'_\alpha = \partial_{r'}fr'^2d\Omega'$, we have

$$\begin{aligned}\chi_a = \frac{1}{4\pi}\int_{S_a}\sum_{\ell=0}^{\infty}&[-g_\ell^{\rm BC}(r,r')\partial_{r'}\hat\Phi + \partial_{r'}g_\ell^{\rm BC}(r,r')\hat\Phi]_{r'=r_a}\\ &\times\sum_{m=0}^{\ell}\epsilon_m\frac{(\ell-m)!}{(\ell+m)!}P_\ell^m(\cos\theta)P_\ell^m(\cos\theta')\\ &\times\cos[m(\phi-\phi')]r_a^2d\Omega'\end{aligned} \tag{B2}$$

and

$$\begin{aligned}\chi_b = \frac{1}{4\pi}\int_{S_b}\sum_{\ell=0}^{\infty}&[g_\ell^{\rm BC}(r,r')\partial_{r'}\hat\Phi - \partial_{r'}g_\ell^{\rm BC}(r,r')\hat\Phi]_{r'=r_b}\\ &\times\sum_{m=0}^{\ell}\epsilon_m\frac{(\ell-m)!}{(\ell+m)!}P_\ell^m(\cos\theta)P_\ell^m(\cos\theta')\\ &\times\cos[m(\phi-\phi')]r_b^2d\Omega'.\end{aligned} \tag{B3}$$

Note that these χ_a and χ_b in this section are defined differently from those in the previous section, Eqs. (A1) and (A2). As shown in Fig. 19, we denote by r_a the radius of the sphere S_a for the exterior problem and by χ_a the corresponding integral, and by r_b the radius of the sphere S_b for the interior problem and χ_b the corresponding integral.

1. Kernel function $G^{\rm NB}(x, x')$

The radial part for the kernel function without the boundary, $G^{\rm NB}(x, x')$, is

$$g_\ell^{\rm NB}(r,r') = \frac{r_<^\ell}{r_>^{\ell+1}}, \tag{B4}$$

where $r_> := \max\{r, r'\}$ and $r_< := \min\{r, r'\}$. The radial part of the kernel function in the surface integral on S_a (B2) becomes

$$g_\ell^{\rm NB}(r,r_a) = \frac{r_a^\ell}{r^{\ell+1}}, \qquad \partial_{r'}g_\ell^{\rm NB}(r,r_a) = \ell\frac{r_a^{\ell-1}}{r^{\ell+1}}, \tag{B5}$$

while the surface integral on S_b is

$$g_\ell^{\rm NB}(r,r_b) = \frac{r^\ell}{r_b^{\ell+1}}, \quad \partial_{r'}g_\ell^{\rm NB}(r,r_a) = -(\ell+1)\frac{r^\ell}{r_b^{\ell+2}}. \tag{B6}$$

2. Kernel function $G^{\rm DD}(x, x')$

When Dirichlet boundary conditions are imposed on both S_a and S_b, the kernel function $G^{\rm DD}(x, x')$ satisfies

$$G^{\rm DD}(x,x')|_{S_a} = 0, \qquad G^{\rm DD}(x,x')|_{S_b} = 0.$$

These conditions lead to the vanishing of the radial part $g_\ell^{\rm DD}(r, r')$ on the two spheres S_a and S_b,

$$g_\ell^{\rm DD}(r, r_a) = g_\ell^{\rm DD}(r, r_b) = 0,$$

which results in the following formula for $g_l^{\rm DD}(r, r')$:

$$g_\ell^{\rm DD}(r, r') = \left[1 - \left(\frac{r_a}{r_b}\right)^{2\ell+1}\right]^{-1} \frac{r_a^\ell}{r_b^{\ell+1}}\left[\left(\frac{r_<}{r_a}\right)^\ell - \left(\frac{r_a}{r_<}\right)^{\ell+1}\right] \times \left[\left(\frac{r_b}{r_>}\right)^{\ell+1} - \left(\frac{r_>}{r_b}\right)^\ell\right]. \tag{B7}$$

The radial kernel function at S_a becomes

$$g_\ell^{\rm DD}(r, r_a) = 0, \quad \partial_{r'} g_\ell^{\rm DD}(r, r_a) = (2\ell+1)\frac{r_a^{\ell-1}}{r_b^{\ell+1}} \frac{(\frac{r_b}{r})^{\ell+1} - (\frac{r}{r_b})^\ell}{1 - (\frac{r_a}{r_b})^{2\ell+1}}, \tag{B8}$$

and at S_b,

$$\begin{aligned} g_\ell^{\rm DD}(r, r_b) &= 0, \\ \partial_{r'} g_\ell^{\rm DD}(r, r_b) &= -(2\ell+1)\frac{r_a^\ell}{r_b^{\ell+2}} \frac{(\frac{r}{r_a})^\ell - (\frac{r_a}{r})^{\ell+1}}{1 - (\frac{r_a}{r_b})^{2\ell+1}}. \end{aligned} \tag{B9}$$

A special case is when the surface S_b is absent in the limit $r_b \to \infty$,

$$\partial_{r'} g_\ell^{\rm DD}(r, r_a) = (2\ell+1)\frac{r_a^{\ell-1}}{r^{\ell+1}}, \tag{B10}$$

for the surface integral at S_a, or when the surface S_a is absent in the limit $r_a \to 0$,

$$\partial_{r'} g_\ell^{\rm DD}(r, r_b) = -(2\ell+1)\frac{r^\ell}{r_b^{\ell+2}}, \tag{B11}$$

for the surface integral at S_b. The latter will be used for computing neutron stars.

3. Kernel function $G^{\rm ND}(x, x')$

When Neumann and Dirichlet boundary conditions are imposed on S_a and S_b, respectively, the kernel function $G^{\rm ND}(x, x')$ satisfies

$$\partial_{r'} G^{\rm ND}(x, x')|_{S_a} = 0, \qquad G^{\rm ND}(x, x')|_{S_b} = 0,$$

or in terms of $g_\ell^{\rm ND}(r, r')$,

$$\partial_{r'} g_\ell^{\rm ND}(r, r_a) = g_\ell^{\rm ND}(r, r_b) = 0.$$

Then the radial part of $G^{\rm ND}(x, x')$ is

$$g_\ell^{\rm ND}(r, r') = \left[1 + \frac{\ell}{\ell+1}\left(\frac{r_a}{r_b}\right)^{2\ell+1}\right]^{-1} \frac{r_a^\ell}{r_b^{\ell+1}}\left[\left(\frac{r_<}{r_a}\right)^\ell + \frac{\ell}{\ell+1}\left(\frac{r_a}{r_<}\right)^{\ell+1}\right]\left[\left(\frac{r_b}{r_>}\right)^{\ell+1} - \left(\frac{r_>}{r_b}\right)^\ell\right]. \tag{B12}$$

The radial kernel function at S_a becomes

$$g_\ell^{\rm ND}(r, r_a) = \frac{2\ell+1}{\ell+1}\frac{r_a^\ell}{r_b^{\ell+1}} \frac{(\frac{r_b}{r})^{\ell+1} - (\frac{r}{r_b})^\ell}{1 + \frac{\ell}{\ell+1}(\frac{r_a}{r_b})^{2\ell+1}}, \quad \partial_{r'} g_\ell^{\rm ND}(r, r_a) = 0 \tag{B13}$$

and at S_b,

$$\begin{aligned} g_\ell^{\rm ND}(r, r_b) &= 0 \\ \partial_{r'} g_\ell^{\rm ND}(r, r_b) &= -(2\ell+1)\frac{r_a^\ell}{r_b^{\ell+2}} \frac{(\frac{r}{r_a})^\ell + \frac{\ell}{\ell+1}(\frac{r_a}{r})^{\ell+1}}{1 + \frac{\ell}{\ell+1}(\frac{r_a}{r_b})^{2\ell+1}}. \end{aligned} \tag{B14}$$

A special case is when the surface S_b is absent in the limit $r_b \to \infty$,

$$g_\ell^{\rm ND}(r, r_a) = \frac{2\ell+1}{\ell+1}\frac{r_a^\ell}{r^{\ell+1}}, \tag{B15}$$

for the surface integral at S_a.

4. Kernel function $G^{\rm DN}(x, x')$

When Dirichlet and Neumann boundary conditions are imposed on S_a and S_b, respectively, the kernel function $G^{\rm DN}(x, x')$ satisfies

$$G^{\rm DN}(x, x')|_{S_a} = 0, \qquad \partial_{r'} G^{\rm DN}(x, x')|_{S_b} = 0,$$

or in terms of $g_\ell^{\rm DN}(r, r')$,

$$g_\ell^{\rm DN}(r, r_a) = \partial_{r'} g_\ell^{\rm DN}(r, r_b) = 0.$$

Then the radial part of $G^{\rm DN}(x, x')$ is

$$g_\ell^{\rm DN}(r, r') = \left[\frac{\ell}{\ell+1} + \left(\frac{r_a}{r_b}\right)^{2\ell+1}\right]^{-1} \frac{r_a^\ell}{r_b^{\ell+1}}\left[\left(\frac{r_<}{r_a}\right)^\ell - \left(\frac{r_a}{r_<}\right)^{\ell+1}\right] \times \left[\left(\frac{r_>}{r_b}\right)^\ell + \frac{\ell}{\ell+1}\left(\frac{r_b}{r_>}\right)^{\ell+1}\right]. \tag{B16}$$

The radial kernel function at S_a becomes

$$\begin{aligned} g_\ell^{\rm DN}(r, r_a) &= 0, \\ \partial_{r'} g_\ell^{\rm DN}(r, r_a) &= (2\ell+1)\frac{r_a^{\ell-1}}{r_b^{\ell+1}} \frac{(\frac{r}{r_b})^\ell + \frac{\ell}{\ell+1}(\frac{r_b}{r})^{\ell+1}}{\frac{\ell}{\ell+1} + (\frac{r_a}{r_b})^{2\ell+1}}, \end{aligned} \tag{B17}$$

and at S_b,

$$g_\ell^{\rm DN}(r, r_b) = \frac{2\ell+1}{\ell+1}\frac{r_a^\ell}{r_b^{\ell+1}} \frac{(\frac{r}{r_a})^\ell - (\frac{r_a}{r})^{\ell+1}}{\frac{\ell}{\ell+1} + (\frac{r_a}{r_b})^{2\ell+1}}, \quad \partial_{r'} g_\ell^{\rm DN}(r, r_b) = 0. \tag{B18}$$

5. Kernel function $G^{\rm RD}(x, x')$

When Robin and Dirichlet boundary conditions are imposed on S_a and S_b, respectively, the kernel function $G^{\rm RD}(x, x')$ satisfies

$$\left[\frac{\partial G^{\rm RD}}{\partial r}+\frac{G^{\rm RD}}{2r}\right]_{S_a}=0, \qquad G^{\rm RD}(x,x')|_{S_b}=0,$$

or in terms of $g^{\rm RD}_\ell(r,r')$,

$$\left[\frac{\partial g^{\rm RD}_\ell}{\partial r}+\frac{g^{\rm RD}_\ell}{2r}\right]_{r=r_a}=0, \qquad g^{\rm RD}_\ell(r,r_b)=0.$$

Then radial part of $G^{\rm RD}(x,x')$ is

$$g^{\rm RD}_\ell(r,r')=\left[1+\left(\frac{r_a}{r_b}\right)^{2\ell+1}\right]^{-1}\frac{r_a^\ell}{r_b^{\ell+1}}\left[\left(\frac{r_<}{r_a}\right)^\ell+\left(\frac{r_a}{r_<}\right)^{\ell+1}\right]\times\left[\left(\frac{r_b}{r_>}\right)^{\ell+1}-\left(\frac{r_>}{r_b}\right)^\ell\right]. \tag{B19}$$

For the surface integral at S_a, it is more convenient to rewrite Eq. (B2),

$$\chi_a=\frac{1}{4\pi}\int_{S_a}\sum_{\ell=0}^{\infty}\Bigg[-g^{\rm RD}_\ell(r,r')\left(\partial_{r'}\hat\Phi+\frac{\hat\Phi}{2r'}\right)+\left(\partial_{r'}g^{\rm RD}_\ell(r,r')+\frac{g^{\rm RD}_\ell(r,r')}{2r'}\right)\hat\Phi\Bigg]_{r'=r_a}\times\sum_{m=0}^{\ell}\epsilon_m\frac{(\ell-m)!}{(\ell+m)!}P^m_\ell(\cos\theta)P^m_\ell(\cos\theta')\times\cos[m(\phi-\phi')]r_a^2d\Omega', \tag{B20}$$

while for χ_b, Eq. (B3) is used. Here, the radial kernel function at S_a becomes

$$g^{\rm RD}_\ell(r,r_a)=2\frac{r_a^\ell}{r_b^{\ell+1}}\frac{(\frac{r_b}{r})^{\ell+1}-(\frac{r}{r_b})^\ell}{1+(\frac{r_a}{r_b})^{2\ell+1}}\partial_{r'}g^{\rm RD}_\ell(r,r_a)+\frac{g^{\rm RD}_\ell(r,r_a)}{2r_a}=0. \tag{B21}$$

The radial kernel function at S_b in Eq. (B3) becomes

$$g^{\rm RD}_\ell(r,r_b)=0, \quad \partial_{r'}g^{\rm RD}_\ell(r,r_b)=-(2\ell+1)\frac{r_a^\ell}{r_b^{\ell+2}}\frac{(\frac{r}{r_a})^\ell+(\frac{r_a}{r})^{\ell+1}}{1+(\frac{r_a}{r_b})^{2\ell+1}}. \tag{B22}$$

A special case is when the surface S_b is absent in the limit $r_b\to\infty$. In that case, we use in Eq. (B20), for $\ell=1,2,\ldots,$

$$g^{\rm RD}_\ell(r,r_a)=2\frac{r_a^\ell}{r^{\ell+1}}. \tag{B23}$$

6. Kernel function $G^{\rm DR}(x,x')$

When Dirichlet and Robin boundary conditions are imposed on S_a and S_b, respectively, the kernel function $G^{\rm DR}(x,x')$ satisfies

$$G^{\rm DR}(x,x')|_{S_a}=0, \qquad \left[\frac{\partial G^{\rm DR}}{\partial r}+\frac{G^{\rm DR}}{2r}\right]_{S_b}=0,$$

or in terms of $g^{\rm DR}_\ell(r,r')$,

$$g^{\rm DR}_\ell(r,r_a)=0, \qquad \left[\frac{\partial g^{\rm DR}_\ell}{\partial r}+\frac{g^{\rm DR}_\ell}{2r}\right]_{r=r_b}=0.$$

Then the radial part of $G^{\rm DR}(x,x')$ is

$$g^{\rm DR}_\ell(r,r')=\left[1+\left(\frac{r_a}{r_b}\right)^{2\ell+1}\right]^{-1}\frac{r_a^\ell}{r_b^{\ell+1}}\left[\left(\frac{r_<}{r_a}\right)^\ell-\left(\frac{r_a}{r_<}\right)^{\ell+1}\right]\times\left[\left(\frac{r_b}{r_>}\right)^{\ell+1}+\left(\frac{r_>}{r_b}\right)^\ell\right]. \tag{B24}$$

For the surface integral at S_b, it is more convenient to rewrite Eq. (B3),

$$\chi_b=\frac{1}{4\pi}\int_{S_b}\sum_{\ell=0}^{\infty}\Bigg[g^{\rm DR}_\ell(r,r')\left(\partial_{r'}\hat\Phi+\frac{\hat\Phi}{2r'}\right)-\left(\partial_{r'}g^{\rm DR}_\ell(r,r')+\frac{g^{\rm DR}_\ell(r,r')}{2r'}\right)\hat\Phi\Bigg]_{r'=r_b}\times\sum_{m=0}^{\ell}\epsilon_m\frac{(\ell-m)!}{(\ell+m)!}P^m_\ell(\cos\theta)P^m_\ell(\cos\theta')\times\cos[m(\phi-\phi')]r_b^2d\Omega', \tag{B25}$$

while for χ_a, Eq. (B2) is used. Here, the radial kernel function at S_a becomes

$$g^{\rm DR}_\ell(r,r_a)=0, \quad \partial_{r'}g^{\rm DR}_\ell(r,r_a)=(2\ell+1)\frac{r_a^{\ell-1}}{r_b^{\ell+1}}\frac{(\frac{r}{r_b})^\ell+(\frac{r_b}{r})^{\ell+1}}{1+(\frac{r_a}{r_b})^{2\ell+1}}. \tag{B26}$$

The radial kernel function at S_b in Eq. (B3) becomes

$$g^{\rm DR}_\ell(r,r_b)=2\frac{r_a^\ell}{r_b^{\ell+1}}\frac{(\frac{r}{r_a})^\ell-(\frac{r_a}{r})^{\ell+1}}{1+(\frac{r_a}{r_b})^{2\ell+1}}, \qquad \partial_{r'}g^{\rm DR}_\ell(r,r_b)+\frac{g^{\rm DR}_\ell(r,r_b)}{2r_b}=0. \tag{B27}$$

A special case is when the surface S_a is absent in the limit $r_a\to 0$. In that case, we use in Eq. (B27), for $\ell=1,2,\ldots,$

$$g^{\rm DR}_\ell(r,r_b)=2\frac{r^\ell}{r_b^{\ell+1}}. \tag{B28}$$

Such kernel functions for imposing Robin boundary conditions at the outer surface S_b may improve the accuracy of the solution, especially near the boundary [30].

7. Kernel function $G^{\rm NN}(x,x')$

When Neumann boundary conditions are imposed on both S_a and S_b, the kernel function $G^{\rm NN}(x,x')$ satisfies

$$\partial_{r'}G^{\rm NN}(x,x')|_{S_a}=G_a, \qquad \partial_{r'}G^{\rm NN}(x,x')|_{S_b}=G_b,$$

where G_a, G_b cannot both be zero. In that case $G^{\rm NN}(x,x')$ does not exist since the $\ell=0$ mode cannot be satisfied. Therefore, the boundary conditions for the radial part will be

$$\partial_{r'} g_\ell^{\rm NN}(r, r_a) = G_a \delta_{0\ell}, \qquad \partial_{r'} g_\ell^{\rm NN}(r, r_b) = G_b \delta_{0\ell}.$$

Then for $\ell = 1, 2, \ldots$ we get

$$g_\ell^{\rm NN}(r, r') = \left[1 - \left(\frac{r_a}{r_b}\right)^{2\ell+1}\right]^{-1} \frac{r_a^\ell}{r_b^{\ell+1}} \frac{\ell+1}{\ell} \left[\left(\frac{r_<}{r_a}\right)^\ell + \frac{\ell}{\ell+1}\left(\frac{r_a}{r_<}\right)^{\ell+1}\right]\left[\left(\frac{r_>}{r_b}\right)^\ell + \frac{\ell}{\ell+1}\left(\frac{r_b}{r_>}\right)^{\ell+1}\right] \tag{B29}$$

(symmetric in r, r'), while for $\ell = 0$

$$g_0^{\rm NN}(r, r') = \frac{1}{r_>} - \frac{G_a r_a^2}{r} + h(r'),$$

where $h(r')$ is an arbitrary function. Symmetry is imposed by choosing $h(r') = -G_a r_a^2/r'$; therefore,

$$g_0^{\rm NN}(r, r') = \frac{1}{r_>} - r_a^2 G_a \left(\frac{1}{r_>} + \frac{1}{r_<}\right). \tag{B30}$$

Note also that in order to satisfy the $\ell = 0$ mode, the following condition must hold:

$$1 = G_a r_a^2 - G_b r_b^2.$$

Therefore, G_a and G_b cannot be chosen arbitrarily. The surface integral at S_a is

$$\chi_a = \frac{1}{4\pi} \sum_{\ell=0}^{\infty} \sum_{m=0}^{\ell} \epsilon_m \frac{(\ell-m)!}{(\ell+m)!} P_\ell^m(\cos\theta) \times \int_{S_a} \left(-g_\ell^{\rm NN}(r, r_a) \frac{\partial \Phi}{\partial r} + G_a \delta_{0\ell} \Phi\right) \times P_\ell^m(\cos\theta') \cos[m(\phi - \phi')] r_a^2 d\Omega', \tag{B31}$$

where for $\ell = 1, 2, \ldots$,

$$g_\ell^{\rm NN}(r, r_a) = \frac{2\ell+1}{\ell} \frac{r_a^\ell}{r_b^{\ell+1}} \frac{(\frac{r}{r_b})^\ell + \frac{\ell}{\ell+1}(\frac{r_b}{r})^{\ell+1}}{1 - (\frac{r_a}{r_b})^{2\ell+1}}, \tag{B32}$$

and for $\ell = 0$,

$$g_0^{\rm NN}(r, r_a) = \frac{1}{r} - r_a^2 G_a \left(\frac{1}{r} + \frac{1}{r_a}\right). \tag{B33}$$

The surface integral at S_b is

$$\chi_b = \frac{1}{4\pi} \sum_{\ell=0}^{\infty} \sum_{m=0}^{\ell} \epsilon_m \frac{(\ell-m)!}{(\ell+m)!} P_\ell^m(\cos\theta) \times \int_{S_b} \left(g_\ell^{\rm NN}(r, r_b) \frac{\partial \Phi}{\partial r} - G_b \delta_{0\ell} \Phi\right) \times P_\ell^m(\cos\theta') \cos[m(\phi - \phi')] r_b^2 d\Omega', \tag{B34}$$

where for $\ell = 1, 2, \ldots$,

$$g_\ell^{\rm NN}(r, r_b) = \frac{2\ell+1}{\ell} \frac{r_a^\ell}{r_b^{\ell+1}} \frac{(\frac{r}{r_a})^\ell + \frac{\ell}{\ell+1}(\frac{r_a}{r})^{\ell+1}}{1 - (\frac{r_a}{r_b})^{2\ell+1}}, \tag{B35}$$

and for $\ell = 0$,

$$g_0^{\rm NN}(r, r_b) = \frac{1}{r_b} - r_a^2 G_a \left(\frac{1}{r_b} + \frac{1}{r}\right). \tag{B36}$$

A special case is when the surface S_a is absent in the limit $r_a \to 0$. In that case, we use in Eq. (B34), for $\ell = 1, 2, \ldots$,

$$g_\ell^{\rm NN}(r, r_b) = \frac{2\ell+1}{\ell} \frac{r^\ell}{r_b^{\ell+1}}, \tag{B37}$$

and for $\ell = 0$,

$$g_0^{\rm NN}(r, r_b) = \frac{1}{r_b}. \tag{B38}$$

APPENDIX C: FINITE DIFFERENCE FORMULAS

The second order finite difference formulas used in the elliptic equation solvers of the COCAL code are summarized in this section. In evaluating the integrals of the solver, such as Eq. (7), we use the midpoint rule. Hence, we need to evaluate the source terms in the integrand at the midpoints $(r_{i-1/2}, \theta_{j-1/2}, \phi_{k-1/2})$ of the grid points that may involve values of potentials and their derivatives. Those are calculated, respectively, by

$$f(r_{i-1/2}, \theta_{j-1/2}, \phi_{k-1/2}) \simeq \frac{1}{8} \sum_{I=i-1}^{i} \sum_{J=j-1}^{j} \sum_{K=k-1}^{k} f(r_I, \theta_J, \phi_K), \tag{C1}$$

$$\frac{\partial f}{\partial r}(r_{i-1/2}, \theta_{j-1/2}, \phi_{k-1/2}) \simeq \frac{1}{4} \sum_{J=j-1}^{j} \sum_{K=k-1}^{k} \frac{f(r_i, \theta_J, \phi_K) - f(r_{i-1}, \theta_J, \phi_K)}{\Delta r_i}, \tag{C2}$$

$$\frac{\partial f}{\partial \theta}(r_{i-1/2}, \theta_{j-1/2}, \phi_{k-1/2}) \simeq \frac{1}{4} \sum_{I=i-1}^{i} \sum_{K=k-1}^{k} \frac{f(r_I, \theta_j, \phi_K) - f(r_I, \theta_{j-1}, \phi_K)}{\Delta \theta_j}, \tag{C3}$$

$$\frac{\partial f}{\partial \phi}(r_{i-1/2}, \theta_{j-1/2}, \phi_{k-1/2}) \simeq \frac{1}{4} \sum_{I=i-1}^{i} \sum_{J=j-1}^{j} \frac{f(r_I, \theta_J, \phi_k) - f(r_I, \theta_J, \phi_{k-1})}{\Delta \phi_k}. \tag{C4}$$

The quadrature formula for the second order midpoint rule at the interval $[r_{i-1}, r_i] \times [\theta_{j-1}, \theta_j] \times [\phi_{k-1}, \phi_k]$ is written

$$\int_{r_{i-1}}^{r_i} dr \int_{\theta_{j-1}}^{\theta_j} d\theta \int_{\phi_{k-1}}^{\phi_k} d\phi S(r, \theta, \phi) \simeq S(r_{i-1/2}, \theta_{j-1/2}, \phi_{k-1/2}) \Delta r_i \Delta \theta_j \Delta \phi_k. \tag{C5}$$

[1] M. Alcubierre, *Introduction to* 3 + 1 *Numerical Relativity* (Oxford University Press, New York, 2008); T. W. Baumgarte and S. L. Shapiro, *Numerical Relativity: Solving Einstein's Equations on the Computer* (Cambridge University Press, New York, 2010).

[2] C. D. Ott, A. Burrows, L. Dessart, and E. Livne, Astrophys. J. **685**, 1069 (2008); T. D. Brandt, A. Burrows, C. D. Ott, and E. Livne, Astrophys. J. **728**, 8 (2011); Y. Sekiguchi and M. Shibata, Astrophys. J. **737**, 6 (2011).

[3] M. Shibata and H. Yoshino, Phys. Rev. D **81**, 104035 (2010); H. Okawa, K. i. Nakao, and M. Shibata, Phys. Rev. D **83**, 121501 (2011); L. Lehner and F. Pretorius, arXiv:1106.5184.

[4] Y. T. Liu, S. L. Shapiro, Z. B. Etienne, and K. Taniguchi, Phys. Rev. D **78**, 024012 (2008); M. Anderson, E. W. Hirschmann, L. Lehner, S. L. Liebling, P. M. Motl, D. Neilsen, C. Palenzuela, and J. E. Tohline, Phys. Rev. Lett. **100**, 191101 (2008); B. Giacomazzo, L. Rezzolla, and L. Baiotti, Phys. Rev. D **83**, 044014 (2011); J. S. Read, C. Markakis, M. Shibata, K. Uryu, J. D. E. Creighton, and J. L. Friedman, Phys. Rev. D **79**, 124033 (2009); K. Hotokezaka, K. Kyutoku, H. Okawa, M. Shibata, and K. Kiuchi, Phys. Rev. D **83**, 124008 (2011); M. Shibata, Y. Suwa, K. Kiuchi, and K. Ioka, Astrophys. J. Lett. **734**, L36 (2011); Y. Sekiguchi, K. Kiuchi, K. Kyutoku, and M. Shibata, Phys. Rev. Lett. **107**, 051102 (2011); Y. Sekiguchi, Prog. Theor. Phys. **124**, 331 (2010); Classical Quantum Gravity **27**, 114107 (2010); K. Kiuchi, Y. Sekiguchi, M. Shibata, and K. Taniguchi, Phys. Rev. Lett. **104**, 141101 (2010); K. Kiuchi, S. Yoshida, and M. Shibata, Astron. Astrophys. **532**, A30 (2011); S. L. Liebling, L. Lehner, D. Neilsen, and C. Palenzuela, Phys. Rev. D **81**, 124023 (2010); S. Chawla, M. Anderson, M. Besselman, L. Lehner, S. L. Liebling, P. M. Motl, and D. Neilsen, Phys. Rev. Lett. **105**, 111101 (2010); K. Kyutoku, H. Okawa, M. Shibata, and K. Taniguchi, Phys. Rev. D **84**, 064018 (2011).

[5] E. Berti, V. Cardoso, J. A. Gonzalez, U. Sperhake, M. Hannam, S. Husa, and B. Bruegmann, Phys. Rev. D **76**, 064034 (2007); J. A. Gonzalez, U. Sperhake, B. Bruegmann, M. Hannam, and S. Husa, Phys. Rev. Lett. **98**, 091101 (2007); J. A. Gonzalez, M. D. Hannam, U. Sperhake, B. Bruegmann, and S. Husa, Phys. Rev. Lett. **98**, 231101 (2007); B. Bruegmann, J. A. Gonzalez, M. Hannam, S. Husa, and U. Sperhake, Phys. Rev. D **77**, 124047 (2008); U. Sperhake, V. Cardoso, C. D. Ott, E. Schnetter, and H. Witek, Phys. Rev. D **84**, 084038 (2011).

[6] P. Mosta, C. Palenzuela, L. Rezzolla, L. Lehner, S. Yoshida, and D. Pollney, Phys. Rev. D **81**, 064017 (2010); arXiv:0912.2330.C. Palenzuela, L. Lehner, and S. L. Liebling, Science **329**, 927 (2010).

[7] G. B. Cook, Living Rev. Relativity **3**, 5 (2000), http://relativity.livingreviews.org/Articles/lrr-2000-5.

[8] G. B. Cook, Phys. Rev. D **50**, 5025 (1994); S. Brandt and B. Brügmann, Phys. Rev. Lett. **78**, 3606 (1997); H. P. Pfeiffer, S. A. Teukolsky, and G. B. Cook, Phys. Rev. D **62**, 104018 (2000); P. Marronetti and R. A. Matzner, Phys. Rev. Lett. **85**, 5500 (2000); G. B. Cook, Phys. Rev. D **65**, 084003 (2002); H. P. Pfeiffer, G. B. Cook, and S. A. Teukolsky, Phys. Rev. D **66**, 024047 (2002); G. Lovelace, R. Owen, H. P. Pfeiffer, and T. Chu, Phys. Rev. D **78**, 084017 (2008); M. Caudill, G. B. Cook, J. D. Grigsby, and H. P. Pfeiffer, Phys. Rev. D **74**, 064011 (2006); M. Ansorg, Phys. Rev. D **72**, 024018 (2005); M. Ansorg, B. Bruegmann, and W. Tichy, Phys. Rev. D **70**, 064011 (2004); W. Tichy, B. Brügmann, M. Campanelli, and P. Diener, Phys. Rev. D **67**, 064008 (2003); B. J. Kelly, W. Tichy, M. Campanelli, and B. F. Whiting, Phys. Rev. D **76**, 024008 (2007); B. C. Mundim, B. J. Kelly, Y. Zlochower, H. Nakano, and M. Campanelli, Classical Quantum Gravity **28**, 134003 (2011); P. Grandclement, J. Comput. Phys. **229**, 3334 (2010).

[9] J. R. Wilson and G. J. Mathews, Phys. Rev. Lett. **75**, 4161 (1995); P. Marronetti, G. J. Mathews, and J. R. Wilson, Phys. Rev. D **60**, 087301 (1999); T. W. Baumgarte, G. B. Cook, M. A. Scheel, S. L. Shapiro, and S. A. Teukolsky, Phys. Rev. D **57**, 6181 (1998); **57**, 7299 (1998); J. A. Faber, P. Grandclement, F. A. Rasio, and K. Taniguchi, Phys. Rev. Lett. **89**, 231102 (2002); W. Tichy, Classical Quantum Gravity **26**, 175018 (2009).

[10] K. Taniguchi, T. W. Baumgarte, J. A. Faber, and S. L. Shapiro, Phys. Rev. D **72**, 044008 (2005); **74**, 041502(R) (2006); **75**, 084005 (2007); **77**, 044003 (2008); K. Kyutoku, M. Shibata, and K. Taniguchi, Phys. Rev. D **79**, 124018 (2009); F. Foucart, L. E. Kidder, H. P. Pfeiffer, and S. A. Teukolsky, Phys. Rev. D **77**, 124051 (2008).

[11] K. Uryu and Y. Eriguchi, Phys. Rev. D **61**, 124023 (2000); K. Uryu, M. Shibata, and Y. Eriguchi, Phys. Rev. D **62**, 104015 (2000).

[12] K. Uryu, F. Limousin, J. L. Friedman, E. Gourgoulhon, and M. Shibata, Phys. Rev. Lett. **97**, 171101 (2006); K. Uryu, F. Limousin, J. L. Friedman, E. Gourgoulhon, and M. Shibata, Phys. Rev. D **80**, 124004 (2009).

[13] A. Tsokaros and K. Uryū, Phys. Rev. D **75**, 044026 (2007).

[14] X. Huang, C. Markakis, N. Sugiyama, and K. Uryu, Phys. Rev. D **78**, 124023 (2008).

[15] N. Stergioulas, Living Rev. Relativity **6**, 3 (2003), http://www.livingreviews.org/lrr-2003-3; R. Meinel, M. Ansorg, A. Kleinwächter, G. Neugebauer, and D. Petroff, *Relativistic Figures of Equilibrium* (Cambridge University Press, New York, 2008).

[16] S. Bonazzola, E. Gourgoulhon, and J.-A. Marck, Phys. Rev. Lett. **82**, 892 (1999); E. Gourgoulhon, P. Grandclement, K. Taniguchi, J.-A. Marck, and S. Bonazzola, Phys. Rev. D **63**, 064029 (2001); K. Taniguchi and E. Gourgoulhon, Phys. Rev. D **66**, 104019 (2002); **68**, 124025 (2003); M. Bejger, D. Gondek-Rosinska, E. Gourgoulhon, P. Haensel, K. Taniguchi, and J. L. Zdunik, Astron. Astrophys. **431**, 297 (2005); K. Taniguchi and M. Shibata, Astrophys. J. Suppl. Ser. **188**, 187 (2010).

[17] H. Komatsu, Y. Eriguchi, and I. Hachisu, Mon. Not. R. Astron. Soc. **237**, 355 (1989).

[18] J. L. Friedman, K. Uryu, and M. Shibata, Phys. Rev. D **65**, 064035 (2002); **70**, 129904 (2004).

[19] K. Uryu, E. Gourgoulhon, and C. Markakis, Phys. Rev. D **82**, 104054 (2010).

[20] E. Gourgoulhon, C. Markakis, K. Uryu, and Y. Eriguchi, Phys. Rev. D **83**, 104007 (2011).

[21] S. Yoshida, B. C. Bromley, J. S. Read, K. Uryu, and J. L. Friedman, Classical Quantum Gravity **23**, S599 (2006).

[22] J. Isenberg, Int. J. Mod. Phys. D **17**, 265 (2008); J. Isenberg and J. Nester, in *General Relativity and Gravitation*, edited by A. Held (Plenum Press, New York, 1980), Vol. 1.

[23] J. R. Wilson and G. J. Mathews, in *Frontiers in Numerical Relativity*, edited by C. R. Evans, L. S. Finn, and D. W. Hobill (Cambridge University Press, Cambridge, England, 1989), p. 306.

[24] M. Shibata, K. Uryu, and J. L. Friedman, Phys. Rev. D **70**, 044044 (2004); **70**, 129901 (2004).

[25] J. D. Jackson, *Classical Electrodynamics* (John Wiley & Sons, New York, 1975), 2nd ed.

[26] D. R. Brill and R. W. Lindquist, Phys. Rev. **131**, 471 (1963).

[27] E. Gourgoulhon and J. L. Jaramillo, Phys. Rep. **423**, 159 (2006).

[28] I. Booth, Can. J. Phys. **83**, 1073 (2005).

[29] J. Thornburg, Classical Quantum Gravity **4**, 1119 (1987); S. Dain, Classical Quantum Gravity **21**, 555 (2004); **22**, 769(E) (2005).

[30] J. W. York and T. Piran, in *Spacetime and Geometry: The Alfred Schild Lectures*, edited by R. A. Matzner and L. C. Shepley (University of Texas Press, Austin, Texas, 1982).