New pseudospectral code for the construction of initial data

Alireza Rashti, Francesco Maria Fabbri, Bernd Brügmann, Swami Vivekanandji Chaurasia, Tim Dietrich, Maximiliano Ujevic, and Wolfgang Tichy

Elliptica: a new pseudo-spectral code for the construction of initial data

Alireza Rashti[1], Francesco Maria Fabbri[2], Bernd Brügmann[2], Swami
Vivekanandji Chaurasia[3], Tim Dietrich[4,5], Maximiliano Ujevic[6], Wolfgang Tichy[1]

[1]*Department of Physics, Florida Atlantic University, Boca Raton, FL 33431, USA*
[2]*Theoretical Physics Institute, University of Jena, 07743 Jena, Germany*
[3]*The Oskar Klein Centre, Department of Astronomy,*
Stockholm University, AlbaNova, SE-10691 Stockholm, Sweden
[4]*Institut für Physik und Astronomie, Universität Potsdam,*
Haus 28, Karl-Liebknecht-Strasse 24/25, 14476, Potsdam, Germany
[5]*Max Planck Institute for Gravitational Physics (Albert Einstein Institute), Am Mühlenberg 1, Potsdam 14476, Germany*
[6]*Centro de Ciências Naturais e Humanas, Universidade Federal do ABC, 09210-170, Santo André, São Paulo, Brazil*

(Dated: May 2, 2022)

Numerical studies of the dynamics of gravitational systems, e.g., black hole-neutron star systems, require physical and constraint-satisfying initial data. In this article, we present the newly developed pseudo-spectral code `Elliptica`, an infrastructure for construction of initial data for various binary and single gravitational systems of all kinds. The elliptic equations under consideration are solved on a single spatial hypersurface of the spacetime manifold. Using coordinate maps, the hypersurface is covered by patches whose boundaries can adapt to the surface of the compact objects. To solve elliptic equations with arbitrary boundary condition, `Elliptica` deploys a Schur complement domain decomposition method with a direct solver. In this version, we use cubed sphere coordinate maps and the fields are expanded using Chebyshev polynomials of the first kind. Here, we explain the building blocks of `Elliptica` and the initial data construction algorithm for a black hole-neutron star binary system. We perform convergence tests and evolve the data to validate our results. Within our framework, the neutron star can reach spin values close to breakup with arbitrary direction, while the black hole can have arbitrary spin with dimensionless spin magnitude ~ 0.8.

Keywords: black hole-neutron star binary, initial data

I. INTRODUCTION

Observations of gravitational waves [1–5] are treasure troves of information about a broad spectrum of large scale physics, such as the nature of gravity [6–11], and small scale physics, such as properties of the equation of state of supranuclear-dense matter [2, 4, 12–14]. New gravitational wave observing runs of the LIGO-Virgo Collaboration and improvements in detectors' sensitivity [15] increase the demand for a better understanding of the complex physics present around the moment of merger of the compact objects. This, on the other hand, requires high accuracy numerical-relativity simulations.

Any kind of numerical-relativity simulation requires as a starting point constraint-satisfying and self-consistent initial data (ID). In this sense, the accuracy and reliability of the simulations depend on the accuracy of the ID. Because of this, significant efforts have been put by the entire numerical-relativity community into developing ID solvers. Among these solvers are: the `COCAL` code [16, 17] for constructing quasi-equilibrium spinning binary black hole (BBH) and binary neutron star (BNS) initial data on circular, but not eccentricity-reduced orbits; the publicly available `FUKA` code [18] for computing eccentricity-reduced BBH, BNS, and black hole-neutron star (BHNS) binaries with aligned or anti-aligned spins; the public code `LORENE` [19–23] for quasi equilibrium or corotational BBH, BHNS, and BNS, where only black holes can have spins that are aligned or anti-aligned, and of low magnitude; a private version of `LORENE` [24–26]

for the construction of quasi-equilibrium, eccentricity reduced BBH, BNS, and BHNS binaries (but with limited magnitude and direction of spin); the `NRPyElliptic` code [27] for BBHs; the `SGRID` [28–30] code, capable of producing BNS configurations with arbitrary spin orientation, eccentricity, and mass ratio; `SpECTRE`'s elliptic solver [31, 32] for computing BBHs ID; the private code `Spells` [33–36] for the construction of eccentricity-reduced BBH, BNS, and BHNS binaries with arbitrary spin and asymmetric masses; and `TwoPunctures` [37–39] for BBHs and non-spinning BHNSs. As the current public ID solvers are limited to aligned or anti-aligned spin directions for BHNS systems, developing an infrastructure for the construction of ID with highly spinning black holes or neutron stars with arbitrary spin directions is important.

We have developed a new pseudo-spectral code, `Elliptica`, as an infrastructure for the construction of ID of various astrophysical compact objects. `Elliptica`'s framework is such that each compact object, like a neutron star (NS) or black hole (BH), is implemented in a separate module. Hence, in principle, one can combine different modules to create ID for BBH, BNS, and BHNS systems. In this work, we show its usage for the computation of BHNS ID. `Elliptica`'s general benefits over other available codes are its suitability for efficient eccentricity reduction, cf. [26], and its ability to compute systems in which the NS and BH spins can point in arbitrary directions, and its ability to compute highly spinning BHs and NSs.

The organization of the paper is as follows: in section II we present the foundation, such as the coordinate setup and the method used for the solution of elliptic equations; in section III we cover the formalism employed to derive the Einstein's constraint equations and Euler's equations for BHNS systems; in section IV we explain the algorithms applied to construct physical and constraint-satisfying ID for BHNSs; in section V we validate the code by performing convergence tests of BHNS ID, by comparing with analytical approximations, and by performing dynamical evolution of the data; in section VI we present our conclusions. We use geometric units with $G = c = M_\odot = 1$ in this paper. Summation over repeated indices is implied unless otherwise mentioned.

II. ELLIPTICA'S FOUNDATION

A. Overview

Elliptica is designed to construct ID for equilibrium and quasi-equilibrium astrophysical systems composed of single or binary compact objects which can be BHs or NSs. To construct the ID, one has to solve the constraint equations together with Euler's equations. The constraint equations are derived from Einstein's equations, and Euler's equations are derived from the conservation of the stress-energy tensor and the continuity equation; cf. III. These equations can be cast into the form of hyperbolic-parabolic or hyperbolic-algebraic system such as [40–43]. They can also be put into hyperbolic form and evolved forward in an unphysical time to find a steady state solution that satisfies the equations [27, 44]. However, in this work we express these constraint equations in the form of coupled nonlinear elliptic partial differential equations (PDE)s. Additionally, the NS surface location is not known a priori. This means these coupled elliptic PDEs need to be augmented by an algebraic equation for the NS surface. Therefore, in this work, making ID is tantamount to a procedure to find the solution of these equations (elliptic plus algebraic equations). Furthermore, since the ID are sought for a specific physical system with specific properties, the solution must be guided towards these physical parameters throughout this procedure; cf. IV.

Elliptica has been written completely in the C programming language, but an extensive use of structures has allowed for the incorporation of some object-oriented design principles. It currently supports only shared-memory multiprocessing. To generate numerical-relativity equations in C, the open source code Cpi has been used [45] (but other means can be used too).

As an infrastructure for construction of ID, Elliptica requires two main components, an elliptic solver and a computational grid. In this section we present these ingredients by explaining how an elliptic equation is set up and solved using a pseudo-spectral method [46, 47] together with a Schur complement domain decomposi-

tion (SCDD) method [48]. Moreover, we derive analytical expressions for a fast computation of the Jacobian of an elliptic equation. Finally, we illustrate our computational grid.

B. Elliptic Solver

A key ingredient of many ID codes is a routine that solves elliptic equations with given boundary conditions. However, this undertaking is often computationally expensive. To reduce the overhead, we take advantage of spectral techniques to efficiently compute the Jacobian needed when linearizing elliptic equations.

1. Jacobian Matrix

To explain the idea, let us consider solving the 1-dimensional Poisson equation

$$\nabla^2 u = \frac{\partial^2 u}{\partial x^2} = S, \ u|_{\partial\Omega} = 0. \tag{1}$$

for the field $u = u(x)$ with source $S = S(x)$ on a computational grid $\Omega = \{x : x \in [a, b]\}$. Here a and b are real numbers and $\partial\Omega$ denotes the boundary. First, we discretize the problem by introducing N grid points. Now, instead of one PDE we thus obtain one algebraic equation per grid point:

$$F_i(\vec{u}) := \{\nabla^2 u - S\}|_i = 0, \tag{2}$$

here, i refers to the index of a grid point, $\vec{u} = (u_0, u_1, ..., u_{N-1})^T$ is u at each grid point, and $\nabla^2 u|_i$ is the discretization of the derivative $\nabla^2 u$ at the grid point i. The Newton-Raphson method also requires the linearization of this equation. This involves the computation of the Jacobian matrix

$$J_{ij} := \frac{\delta F_i(\vec{u})}{\delta u_j}, \tag{3}$$

where indices i and j refer to the indices of the grid points after the discretization. A Newton-Raphson step then consists of $u \to u + \tilde{u}$, where $\tilde{u}$ is the solution of the linear equation $J\tilde{u} = -F$. The method starts from an initial guess for u and iterates until a given stopping criterion is met, e.g., until a desired tolerance for the L^2 norm $||\vec{F}||$ is reached or if the number of iterations exceeds some limit. The main steps of our Newton-Raphson procedure [49, 50] is summarized in Algorithm 1.

In many cases constructing the matrix J_{ij} (defined in Eq. (3)) from an analytical expression is not practical. Instead one often uses a finite difference approximation:

$$J_{ij} = \frac{\delta F_i(\vec{u})}{\delta u_j} \approx \frac{F_i(\vec{u} + h\vec{e}_j) - F_i(\vec{u})}{h}, \tag{4}$$

where h is a small value, on the order of the grid spacing, and $\vec{e}_j$ is the vector whose only nonzero component equals

Algorithm 1 Newton-Raphson algorithm: the number of iterations N_{iter} and the desired tolerance *tol* are given.

1: set $k = 0$
2: **while** $(k \leq N_{iter}$ or $||\vec{F}(\vec{u})|| \geq tol)$ **do**
3:     compute $F_i(\vec{u}) = \{\nabla^2 u - S\}|_i$ at each grid point i;
4:     compute Jacobian matrix $J_{ij} = \frac{\delta F_i(\vec{u})}{\delta u_j}$;
5:     solve matrix equation $J\tilde{u} = -F$ for $\tilde{u}$;
6:     set $u_i \rightarrow u_i + \tilde{u}_i$;
7:     set $k \rightarrow k + 1$;
8: **end while**

to 1 in its j-th entry. For a 1-dimensional problem, the time complexity of this method is about $O(N \times N \ln N)$ in which N is the number of grid points and $N \ln N$ comes from a fast Fourier transformation needed to compute the derivatives for each change $\vec{u} \rightarrow \vec{u} + h\vec{e}_j$. However, one can calculate this expression not only exactly in a closed form but also faster with time complexity of order $O(N^2)$ by using a spectral expansion. In the following we explain this more efficient method for calculating J_{ij}.

2. Spectral Jacobian

`Elliptica` currently uses Chebyshev polynomials of the first kind $T_i(X) = \cos(i \arccos(X))$ as the basis of the spectral expansion [46, 47], where $X = \frac{2x-(a+b)}{a-b}$ and $i = 0, 1, \ldots, N-1$. As the collocation points, it uses the extrema of the Chebyshev polynomial $T_{N-1}(X)$ which are $X_i = \cos\left(\frac{i\pi}{N-1}\right)$ for each i. As discussed in subsection II D, a further coordinate transformation can be used to map the x coordinates to other coordinates that are better adapted to the domain shapes we intend to use. Here, for the sake of simplicity, we ignore this transformation but the generalization is straightforward. The field u as a function of X can then be approximated as

$$u(X) = \sum_{n=0}^{N-1} \eta_n c_n T_n(X), \quad (5)$$

and consequently, the value $u_i = u(X_i)$ at each grid point i is given by

$$u_i = \sum_{n=0}^{N-1} \eta_n c_n T_n(X_i), \quad (6)$$

where,

$$\eta_n = \begin{cases} 1, & \text{if } n = 0 \text{ or } n = N-1 \\ 2, & \text{otherwise} \end{cases},$$

$$c_n = \frac{1}{2(N-1)} \sum_{k=0}^{N-1} \eta_k u_k T_k(X_n). \quad (7)$$

Moreover, the second order derivative of $u(x)$ with respect to x and the variation of c_n with respect to $u_j = u(x(j))$, which are needed to calculate J_{ij}, read

$$\frac{d^2 u_i}{dx^2} = \left(\frac{dX}{dx}\right)^2 \sum_{n=0}^{N-1} \eta_n c_n \frac{d^2}{dX^2} T_n(X)|_{X=X_i}, \quad (8)$$

$$\frac{\delta c_n}{\delta u_j} = \frac{1}{2(N-1)} \sum_{k=0}^{N-1} \eta_k \delta_{jk} T_k(X_n)$$

$$= \frac{\eta_j T_j(X_n)}{2(N-1)}. \quad (9)$$

Consequently, J_{ij} is:

$$J_{ij} = \frac{\delta F_i(\vec{u})}{\delta u_j}$$

$$= \frac{\delta}{\delta u_j} \frac{d^2 u_i}{dx^2} + \overbrace{\frac{\delta}{\delta u_j} S(x)}^{=0}$$

$$= \frac{\eta_j}{2(N-1)} \left(\frac{dX}{dx}\right)^2 \times$$

$$\left\{ \sum_{n=0}^{N-1} \eta_n T_j(X_n) \frac{d^2}{dX^2} T_n(X)|_{X=X_i} \right\}, \quad (10)$$

where in the third line we have used Eq. (9). Note that the sum in Eq. (10) can be written as

$$\sum_{n=0}^{N-1} \eta_n T_j(X_n) \frac{d^2}{dX^2} T_n(X)|_{X=X_i} =$$

$$\frac{\partial^2}{\partial X^2} \{ 2 \sum_{n=0}^{N-1} T_n(X_j) T_n(X)$$

$$- T_j(X_0) T_0(X) - T_j(X_{N-1}) T_{N-1}(X) \}|_{X=X_i} \quad (11)$$

where we have used the definition of η_n in Eq. (7) and the relation $T_n(X_i) = T_i(X_n)$ which holds since

$$T_i(X_n) = \cos(i \arccos(X_n)) = \cos\left(\pi \frac{i\,n}{N-1}\right). \quad (12)$$

Additionally, we have changed the notation in Eq. (11) to emphasize that the derivative only acts on $T_n(X)$ which is then evaluated at $X = X_i$ as in Eq. (10). To further simplify the summation in Eq. (11) we use

$$2 \sum_{n=0}^{N-1} T_n(X_j) T_n(X_i) = 2 \sum_{n=0}^{N-1} \cos(n\theta_j) \cos(n\theta_i) =$$

$$\sum_{n=0}^{N-1} \cos(n(\theta_i + \theta_j)) + \sum_{n=0}^{N-1} \cos(n(\theta_i - \theta_j)). \quad (13)$$

Here we have defined $\theta = \arccos(X)$ so that $\theta_i = \frac{\pi i}{N-1}$ and $\theta_j = \frac{\pi j}{N-1}$. Using the following identity [51]:

$$\sum_{n=0}^{N} \cos(n\theta) = \frac{1}{2} + \frac{\sin((N+\frac{1}{2})\theta)}{2\sin(\frac{\theta}{2})}, \quad (14)$$

we write J_{ij} in a closed form (note $X_i = \cos(\theta_i)$):

$$J_{ij} = \frac{\eta_j}{2(N-1)} \left(\frac{dX}{dx}\right)^2 \times$$

$$\left\{ \frac{\partial^2}{\partial X_i^2} \left(\frac{\sin((N-\frac{1}{2})(\theta_i+\theta_j))}{2\sin(\frac{\theta_i+\theta_j}{2})} \right) + \right.$$

$$\frac{\partial^2}{\partial X_i^2} \left(\frac{\sin((N-\frac{1}{2})(\theta_i-\theta_j))}{2\sin(\frac{\theta_i-\theta_j}{2})} \right) -$$

$$\left. (-1)^j \frac{d^2}{dX^2} T_{N-1}(X)|_{X=X_i} \right\}. \qquad (15)$$

Note that the derivatives $\frac{\partial}{\partial X_i}$ only act on θ_i.

Some remarks are in order. First, the generalization to higher dimensions is straightforward, and analogous to a Chebyshev expansion in higher dimensions. Second, we note that there is no singularity in Eq. (11). Hence, for cases such as $\theta_i + \theta_j = 0$ or $\theta_i - \theta_j = 0$ when Eq. (15) becomes singular, we use Eq. (13) to compute J_{ij} (i.e., we do not use the identity in Eq. (14)). Third, during the construction of ID, resolution is only gradually increased, so that many iterations of the elliptic solver per resolution are required. Since Eq. (15) only depends on the number of grid points, the piece of J_{ij} coming from derivative operators, here ∇^2, remains unchanged at each resolution. Thus, J_{ij} is calculated only once and used without any changes in an iterative scheme. Furthermore, we note that the functional derivatives do not act on the Jacobian of a coordinate transformation. Thus, if there are more (and possibly nonlinear) coordinate transformations then the computation of J_{ij} involves similar steps except that some extra terms and coefficients (coming from the coordinate transformation) need to be included. Lastly, let us illustrate how the Jacobian of a more complicated equation is calculated. For instance, we assume the Jacobian of the following equation is needed:

$$f_2(u) \frac{d^2 u(x)}{dx^2} + f_1(u) \frac{du(x)}{dx} + f_0(u) = S(x), \qquad (16)$$

where $f_0(u)$, $f_1(u)$, and $f_2(u)$ are (possibly non-linear) functions of u. Hence, similar to Eq. (2)

$$F_i(\vec{u}) = \{f_2(u) \frac{d^2 u(x)}{dx^2} + f_1(u) \frac{du(x)}{dx} + f_0(u) - S(x)\}|_i = 0. \qquad (17)$$

Consequently, the Jacobian reads (no implied summation on the repeated indices)

$$J_{ij} = f_2(u_i) \frac{\delta}{\delta u_j} \frac{d^2 u_i}{dx^2} + \delta_{ij} \frac{df_2(u_i)}{du} \frac{d^2 u_i}{dx^2} +$$

$$f_1(u_i) \frac{\delta}{\delta u_j} \frac{du_i}{dx} + \delta_{ij} \frac{df_1(u_i)}{du} \frac{du_i}{dx} + \delta_{ij} \frac{df_0(u_i)}{du}, \qquad (18)$$

in which, δ_{ij} is the Kronecker delta and $\frac{\delta}{\delta u_j} \frac{d^2 u_i}{dx^2}$ term is calculated by Eq. (15). The terms $\delta_{ij} \frac{df_0(u_i)}{du}$, $\delta_{ij} \frac{df_1(u_i)}{du}$, and $\delta_{ij} \frac{df_2(u_i)}{du}$ are the analytic functional derivative of

$f_0(u_i)$, $f_1(u_i)$, and $f_2(u_i)$ respectively. To compute $\frac{\delta}{\delta u_j} \frac{du_i}{dx}$ we follow the same steps as involved in the calculation of $\frac{\delta}{\delta u_j} \frac{d^2 u_i}{dx^2}$ but instead of second order derivatives with respect to x we have first order derivatives.

In conclusion, we have presented a fast and analytic method to compute the Jacobian of an elliptic PDE using a spectral method. While this spectral Jacobian method is relatively straightforward, we are not aware of any prior publication about it. In the next subsection we discuss how the system $J\tilde{u} = -F$ is solved.

C. Matrix Solver

Having found $F_i(\vec{u})$ and J_{ij}, the Newton-Raphson algorithm 1 requires us to solve the matrix equation $J\tilde{u} = -F$. The size of the matrix J depends on the resolution, which is chosen to fit the problem under study and the coordinate patches being used. For a real production run, the size could be as high as $10^5 \times 10^5$. Direct solvers [52] are generally inefficient for matrix equations with such large dimensions, thus iterative solvers [48] with proper preconditioners are commonly used in these cases. However, direct solvers tend to be more robust and predictable and do not require preconditioners as opposed to iterative solvers. As a result, direct solvers are preferred when feasible. A possible strategy is to divide this big system of equations into mutually exclusive and collectively exhaustive subsystems with smaller dimensions, then instead of solving the whole system at once, one can separately solve these small subsystems which would enable the usage of direct solvers. To reach this goal, `Elliptica` employs the SCDD method, explained in [48], to efficiently solve $J\tilde{u} = -F$ by a direct solver.

1. Domain Decomposition Method

In general, domain decomposition methods, and in particular the SCDD method, use a divide-and-conquer principle to reduce the dimension of a matrix equation. Thus, one can use direct solvers in a parallel fashion to solve the whole matrix equation at once, which otherwise would have been infeasible due to the very large dimension. In this section we demonstrate the gist of the SCDD method used in `Elliptica`.

Often, a given manifold, here the computational grid, cannot be covered by a single patch. A known example is a 2-sphere and its pole singularities [53]. Moreover, for a spectral method, one desires to separate matter and vacuum regions into different patches to avoid Gibbs phenomena [46]. For instance, the NS and the outside of the NS should be covered by different patches. Furthermore, it is generally required to use different resolutions for different parts of the grid or to compactify the outer-boundary of the computational grid to possibly cover spatial infinity. Therefore, different patches
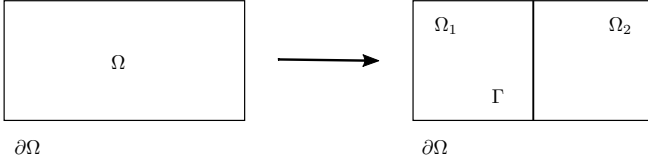
FIG. 1: An example of a 2-dimensional grid Ω that is covered by two subdomains Ω_1 and Ω_2 with a common interface Γ. If we solve an elliptic equation on the two subdomains, boundary conditions on $\partial\Omega$ and Γ are required in order to guarantee a unique solution. The boundary conditions in Eqs. 19 and 20 at Γ couple the elliptic equations in the subdomains Ω_1 and Ω_2 together. Yet, once the values of the fields are known on Γ, the elliptic equations in the two subdomains can be decoupled and solved separately.

with different properties are needed and it is natural to cover a grid with multiple patches. We exploit this property by using the SCDD method to solve equations on each of these patches separately.

To demonstrate the idea of SCDD, consider solving a 2-dimensional elliptic equation on the grid Ω, shown in Fig. 1, with a some boundary condition on $\partial\Omega$. The elliptic equation has a unique solution if appropriate boundary conditions (BCs) are imposed. Thus, if we attempt to solve this equation separately on each subdomain the system would be under-determined because the BCs on Γ are not known yet. Therefore we impose the following BCs to close the system [30]:

$$\vec{n} \cdot \vec{\nabla} u_{(\Omega_1)}|_\Gamma - \vec{n} \cdot \vec{\nabla} u_{(\Omega_2)}|_\Gamma = 0, \qquad (19)$$

$$u_{(\Omega_1)}|_\Gamma - u_{(\Omega_2)}|_\Gamma = 0, \qquad (20)$$

where $\vec{n}$ is the normal vector to the common interface Γ, and $u_{(\Omega_1)}|_\Gamma$ and $u_{(\Omega_2)}|_\Gamma$ denote the solution from domain Ω_1 and Ω_2 evaluated on the interface Γ. Eqs. (19) and (20) create a coupling between the interior of each subdomain and the interface and vice versa. Hence, if the solution was known on the interface Γ, the problem would be reduced to solve two uncoupled elliptic equations in each subdomain. Therefore, to decouple the system it is natural to find the solution on the interface Γ first. This decoupling is the main idea of the SCDD method.

2. Schur Domain Decomposition Method

We consider a grid Ω with outer-boundary $\partial\Omega$ which is covered by subdomains (patches) Ω_1, Ω_2, $\ldots$, and Ω_s, where s is the number of subdomains. Moreover, any two subdomains might have one or more common interfaces which results in a coupling of the two subdomains. As we mentioned earlier the goal is to find the solution on the common interfaces first (to decouple them) and then solve the elliptic equation for each patch independently. Therefore, following the algorithm 1 after setting up the matrix equation $J\tilde{u} = -F$, we reorder this system of equations, such that it has the following general

structure:

$$
\begin{pmatrix}
B_1 & & & & E_1 & & & \\
& B_2 & & & & E_2 & & \\
& & \ddots & & & & \ddots & \\
& & & B_s & & & & E_s \\
F_{11} & F_{12} & \ldots & F_{1s} & C_{11} & C_{12} & \ldots & C_{1s} \\
F_{21} & F_{22} & \ldots & F_{2s} & C_{21} & C_{22} & \ldots & C_{2s} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
F_{s1} & F_{s2} & \ldots & F_{ss} & C_{s1} & C_{s2} & \ldots & C_{ss}
\end{pmatrix}
\begin{pmatrix}
\tilde{v}_1 \\
\tilde{v}_2 \\
\vdots \\
\tilde{v}_s \\
\tilde{w}_1 \\
\tilde{w}_2 \\
\vdots \\
\tilde{w}_s
\end{pmatrix}
=
\begin{pmatrix}
f_1 \\
f_2 \\
\vdots \\
f_s \\
g_1 \\
g_2 \\
\vdots \\
g_s
\end{pmatrix},
$$
(21)

where empty entries are zero. Let us define $p, q = 1, \ldots, s$. Then, the B_p are the sub-matrices of the matrix J that are interior to the p-th subdomain and include outer boundary conditions (if any exist for this subdomain). The E_p represent the sub-matrices of the matrix J due to the coupling of interior and interface of the p-th subdomain. The F_{pq} describe the coupling between the p-th interface and the q-th interior. Finally, C_{pq} represent all of the couplings between the p-th interface to the q-th interface. Furthermore, each $\tilde{v}_p$ is the subvector of unknowns that is interior to the p-th subdomain and each $\tilde{w}_p$ represents the subvector of unknowns for all interfaces. Each f_p shows the source term portion in the p-th subdomain and each g_p is the left hand side of Eq. (19) or Eq. (20) depending which one has been imposed for the corresponding subdomain.

In this new arrangement of the matrix J, all of the coupled and unknown values are encapsulated in the subvector $\tilde{w} = (\tilde{w}_1, \tilde{w}_2, \ldots, \tilde{w}_s)^T$. Thus, finding these values results in decoupling of the system. To better explain the idea, it is convenient to write Eq. (21) like a system of equations with two unknowns as follows:

$$\begin{pmatrix} B & E \\ F & C \end{pmatrix} \begin{pmatrix} \tilde{v} \\ \tilde{w} \end{pmatrix} = \begin{pmatrix} f \\ g \end{pmatrix}. \qquad (22)$$

In order to find $\tilde{w}$ we write the first row in Eq. (22) as

$$B\tilde{v} + E\tilde{w} = f \Rightarrow \tilde{v} = B^{-1}(f - E\tilde{w}). \qquad (23)$$

Then by substituting $\tilde{v}$ in the second row of Eq. (22), we find the reduced system

$$F\tilde{v} + C\tilde{w} = g \Rightarrow (C - FB^{-1}E)\tilde{w} = g - FB^{-1}f. \quad (24)$$

The solution $\tilde{w}$ of Eq. (24) can now be found independently of the values of $\tilde{v}$ and then $\tilde{w}$ can be back substituted into Eq. (23) to find $\tilde{v}$.

In practice, we implement the following equations for each subdomain p (no summation is implied on the repeated indices) [48]:

$$B_p \tilde{v}_p + E_p \tilde{w}_p = f_p, \qquad (25)$$

$$\sum_{q=1}^{s} (C_{pq} - F_{pq} E'_q) \tilde{w}_q = g_p - \sum_{q=1}^{s} (F_{pq} f'_q), \qquad (26)$$

Algorithm 2 Schur complement domain decomposition method.

1: Solve $BE' = E$ for E';
2: Solve $Bf' = f$ for f';
3: Compute $g' = g - Ff'$;
4: Compute $S = (C - FE')$;
5: Solve $S\tilde{w} = g'$ for $\tilde{w}$;
6: Compute $\tilde{v} = f' - E'\tilde{w}$;

where

$$E'_p = B_p^{-1} E_p,$$
$$f'_p = B_p^{-1} f_p. \tag{27}$$

The system of equations (26) only involves $\tilde{w}$, i.e., $\tilde{u}$ at the interface points. It can be summarized as:

$$S\tilde{w} = g', \tag{28}$$

where, S is called Schur complement matrix. After solving Eq. (28) for $\tilde{w}_p$, $\tilde{v}_p$ is found using Eq. (25) which can be written as $\tilde{v}_p = f'_p - E'_p\tilde{w}_p$. A summary of SCDD method to solve elliptic equations is shown in algorithm 2.

Regarding the implementation of SCDD in `Elliptica`, a few comments are in order. First, the system of equations is set up in the same order as Eq. (21), thus, no overhead for reordering of the system is incurred. Second, in Eq. (27) instead of inverting the matrix B_p directly, equations $B_p E'_p = E_p$ and $B_p f' = f_p$ are solved for E'_p and f', respectively. Also, we note that these calculations can be performed in parallel since B_p's describe the uncoupled blocks in the J matrix. Third, many columns in the F_{pq} and C_{pq} matrices are equal to zero because each subdomain p only has interfaces where it touches other subdomains, which only happens for neighboring subdomains. Since all the matrices are stored in compressed column storage [52], there is no computational cost for setting zero entries of F_{pq} and C_{pq}. Fourth, the Schur complement matrix consists of the coupling information coming from the interfaces, thus its dimension is as big as the total number of points on all interfaces. If we use the cubed spherical grid of Fig. 3 with 20 points in each direction in each patch, the matrix S is approximately 56000-dimensional. Nevertheless it is quite sparse and hence, the use of an efficient sparse solver is feasible to solve this matrix equation. Lastly, to invert B_p matrices and solve Eq. (28) the open source unsymmetric multifrontal direct solver `UMFPACK` [54] is used.

As an example, suppose we wish to solve the elliptic equation $\nabla^2 u(x,y) = S(x,y)$ together with BC Eqs. (19) and (20) on the 2-dimensional domain Ω that is divided into two subdomains Ω_1 and Ω_2 as shown on the right of Fig. 1. Assume further that we cover each subdomains by a grid of four by four points. Then, the subdomains share four points along the common interface Γ where
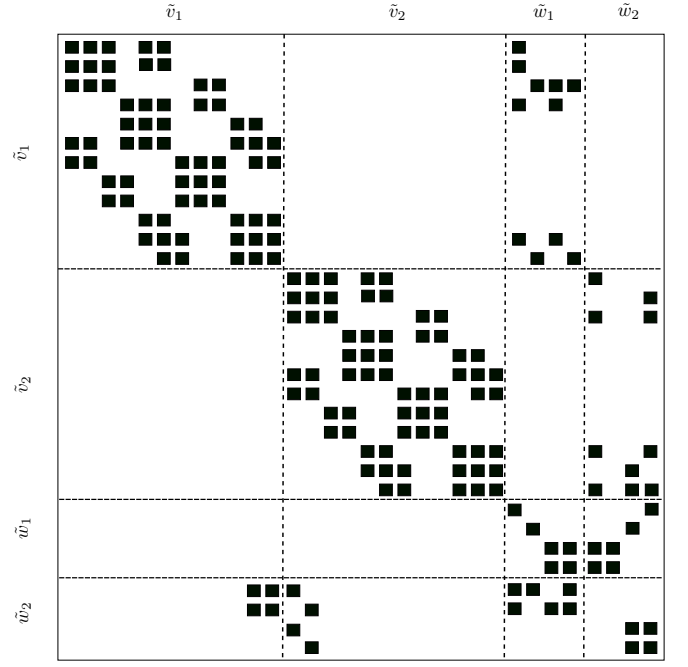

FIG. 2: Schematic representation of the Jacobian matrix J for the PDE $\nabla^2 u(x,y) = S(x,y)$ and BC Eqs. (19) and (20) when solved on the two subdomains of Fig. 1, with four by four points in each subdomain. Non-zero entries in J are represented by small filled squares. The dashed lines delineate sub-matrices, analogous to the sub-matrices in Eq. (21). Here F_{11} and F_{12} are empty since Eq. (20) is imposed from the side of subdomains Ω_1. On the other hand, F_{21} and F_{22} are non-empty as Eq. (19) is imposed from the side of subdomains Ω_2.

the BCs are imposed. These BCs couple the equations on both subdomains. The equations are then reordered following the general structure given in Eq. 21. After this reordering, the Jacobian coming from the equations has the schematic form depicted in Fig. 2.

D. Coordinate Patch

Although equations in general relativity do not depend on coordinate patches on the spacetime manifold, explicit coordinate patches are needed when it comes to numerical-relativity calculations. `Elliptica` uses, but is not limited to, Cartesian and cubed spherical [55] coordinate systems. In this section, we illustrate how the computational grid of a BHNS system is covered.

The essence of the transformation between Cartesian coordinates $x^i = (x, y, z)$ and cubed spherical coordi-

nates $X^i = (X, Y, Z)$ is as follows [30]:

$$X = \frac{x^I}{x^K},$$

$$Y = \frac{x^J}{x^K},$$

$$Z = \frac{x^K - r_{in}}{r_{out} - r_{in}}, \tag{29}$$

where, the indices $I, J, K \in \{1, 2, 3\}$ are all distinct, and $X, Y \in [-1, 1], Z \in [0, 1]$. Here

$$r_{\text{in}} = \frac{\sigma_{in}(X, Y)}{\sqrt{1 + X^2 + Y^2}},$$

$$r_{\text{out}} = \frac{\sigma_{out}(X, Y)}{\sqrt{1 + X^2 + Y^2}}, \tag{30}$$

in which, the inner boundary of the patch is determined by $\sigma_{\text{in}}(X, Y)$ and the outer boundary by $\sigma_{\text{out}}(X, Y)$. Moreover, each $\sigma(X, Y)$ is related to Cartesian coordinates by the equation $\sigma(X, Y) = \sqrt{x^2 + y^2 + z^2}$. For instance, if a perfect 2-sphere is required as the boundary of a patch then $\sigma(X, Y) = C$, similarly for a planar boundary $\sigma(X, Y) = C\sqrt{1 + X^2 + Y^2}$, where C is a constant.

To better capture the field falloff (expected to occur as powers of r^{-1} at large radii), we replace Z by,

$$\widetilde{Z} = \frac{\sigma_{\text{out}}}{\sigma_{\text{out}} - \sigma_{\text{in}}} \left(1 - \frac{\sigma_{\text{in}}}{r}\right), \tag{31}$$

in the outermost patches [30]. Here $r = \sqrt{x^2 + y^2 + z^2}$ and still $\widetilde{Z} \in [0, 1]$. Lastly, in order to avoid coordinate singularities at the center of spheres, each spheroidal object contains a Cartesian coordinate patch around its center.

Using these different maps, we can setup different patches with different surfaces for various needs, e.g. to cover the NS or the space between the compact objects. These patches touch each other and never overlap. They tile the entire computational domain, as depicted in Fig. 3. We have also decided to increase the number of outermost patches in `Elliptica` with respect to `SGRID` [30]. This leads to a more symmetric grid, higher angular resolution, and no need for interpolation within interfaces when setting up interface BCs, i.e., Eq. (19) and Eq. (20).

III. FORMALISM

In this section, we present the formalism applied to construct ID for quasi-equilibrium BHNSs. The ID of these systems are obtained by solving Einstein's equations and Euler's equations. In order to make these equations amenable to the numerical methods used here, they are cast into elliptic type PDEs. Specifically, we use the extended conformal thin sandwich method
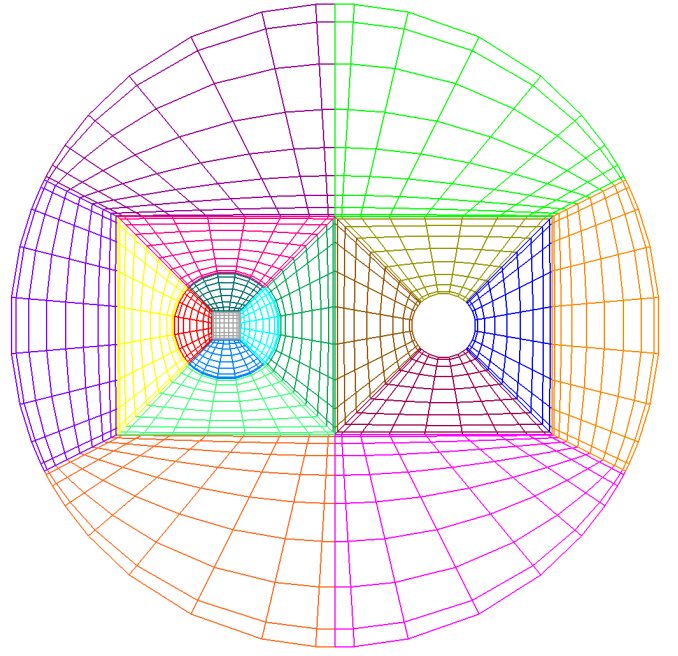

FIG. 3: An example of a computational grid that is using cubed spherical coordinates. Shown is the intersection of the domains with the xy-plane. The domains on the left cover a NS, the ones on the right cover the region around an excised BH.

(XCTS) [56, 57] for the Einstein's equations. For the fluid equations we use the method described in [29]. Furthermore, there are two common approaches to treat BH singularities in this context, the excision approach [58] and the puncture approach [59]. `Elliptica` uses the excision approach: the BH singularity is excised from the computation domain and then boundary conditions are imposed on the excised surface. Below, we present the formulas and conventions used in `Elliptica`.

Using the $3 + 1$ formalism (see e.g. [60]), we write the line element of a spacetime manifold as

$$ds^2 = g_{\mu\nu}dx^\mu dx^\nu = -\alpha^2 dt^2 + \gamma_{ij}(dx^i + \beta^i dt)(dx^j + \beta^j dt), \tag{32}$$

where $g_{\mu\nu}$ is the pseudo-Riemannian metric of the spacetime manifold, α is the lapse function, β^i is the shift vector and γ_{ij} is the induced metric of a spatial hypersurface Σ_t. Using the normal vector n^μ orthogonal to Σ_t, which can be interpreted as the 4-velocity of an Eulerian observer, we have $\gamma_{\mu\nu} = g_{\mu\nu} + n_\mu n_\nu$. The extrinsic curvature on Σ_t is defined by $K_{\mu\nu} = -\frac{1}{2}\pounds_n\gamma_{\mu\nu}$, in which $\pounds_n$ is the Lie derivative along the normal vector.

The NS matter is described by a perfect fluid, therefore, the stress-energy tensor is

$$T_{\mu\nu} = (\rho_0 + \rho_0\epsilon + P)u_\mu u_\nu + Pg_{\mu\nu}, \tag{33}$$

$$= \rho_0 h u_\mu u_\nu + Pg_{\mu\nu}, \tag{34}$$

where ρ_0, ϵ, P, h, and u^μ are, respectively, the rest mass density, specific internal energy, pressure, specific enthalpy and the 4-velocity of the fluid.

In order to employ the XCTS formalism, it is necessary to rescale the 3-metric γ_{ij} and decompose the extrinsic curvature K^{ij} as follows:

$$\gamma_{ij} = \psi^4 \bar{\gamma}_{ij}, \tag{35}$$

$$K^{ij} = A^{ij} + \frac{1}{3} K \gamma^{ij}. \tag{36}$$

Here ψ is the conformal factor, $\bar{\gamma}_{ij}$ the conformal 3-metric, A^{ij} the traceless part of K^{ij} and $K = \gamma_{ij} K^{ij}$. Furthermore, it is convenient to project and then rescale the matter quantities as:

$$E = n_\mu n_\nu T^{\mu\nu} \qquad = \psi^{-6} \bar{E}, \tag{37}$$

$$S = \gamma^{ij} \gamma_{i\mu} \gamma_{j\nu} T^{\mu\nu} \qquad = \psi^{-6} \bar{S}, \tag{38}$$

$$j^i = -\gamma^i_\mu n_\nu T^{\mu\nu} \qquad = \psi^{-6} \bar{j}^i. \tag{39}$$

Moreover, in the XCTS formalism A^{ij} is related to the shift and the time derivative of conformal metric as follows:

$$A^{ij} = \psi^{-10} \bar{A}^{ij}, \tag{40}$$

$$\bar{A}^{ij} = \frac{1}{2\bar{\alpha}} \left((\bar{L}\beta)^{ij} - \bar{\gamma}^{ik} \bar{\gamma}^{jl} \bar{u}_{kl} \right), \tag{41}$$

where

$$\bar{u}_{ij} = \frac{\partial \bar{\gamma}_{ij}}{\partial t}, \tag{42}$$

$$(\bar{L}\beta)^{ij} = \bar{D}^i \beta^j + \bar{D}^j \beta^i - \frac{2}{3} \bar{\gamma}^{ij} \bar{D}_k \beta^k, \tag{43}$$

$$\alpha = \psi^6 \bar{\alpha}, \tag{44}$$

and $\bar{D}$ is the covariant derivative compatible with $\bar{\gamma}_{ij}$. Finally, the Einstein equations result in five coupled non-linear elliptic PDEs [56, 57]:

$$\bar{D}^2 \psi - \frac{1}{8} \psi \bar{R} - \frac{1}{12} \psi^5 K^2 + \frac{1}{8} \psi^{-7} \bar{A}_{ij} \bar{A}^{ij}$$
$$+ 2\pi \psi^{-1} \bar{E} = 0, \tag{45}$$

$$2\bar{\alpha} \left[\bar{D}_j (\frac{1}{2\bar{\alpha}} (\bar{L}\beta)^{ij}) - \bar{D}_j (\frac{1}{2\bar{\alpha}} \bar{u}^{ij}) - \frac{2}{3} \psi^6 \bar{D}^i K \right]$$
$$- 16\pi \bar{\alpha} \psi^4 \bar{j}^i = 0, \tag{46}$$

$$\bar{D}^2 (\bar{\alpha} \psi^7) - (\bar{\alpha} \psi^7) \left[\frac{1}{8} \bar{R} + \frac{5}{12} \psi^4 K^2 + \frac{7}{8} \psi^{-8} \bar{A}_{ij} \bar{A}^{ij} \right]$$
$$+ \psi^5 (\partial_t K - \beta^k \partial_k K) - 2\pi \bar{\alpha} \psi^5 (\bar{E} + 2\bar{S}) = 0, \tag{47}$$

where $\bar{R}$ is the Ricci scalar coming from $\bar{\gamma}_{ij}$. These equation must be solved with appropriate boundary conditions at spatial infinity and on the surface of the excised region H in order to have a unique physical solution. Before mentioning these boundary conditions, it is useful to decompose β^i such that in an inertial frame it has clearly identifiable rotational and inspiral pieces at spatial infinity:

$$\vec{\beta} = \vec{\beta}_0 + \vec{\Omega}_{\text{BHNS}} \times (\vec{r} - \vec{r}_{\text{CM}}) + \frac{v_r}{r_{\text{BHNS}}} (\vec{r} - \vec{r}_{\text{CM}}). \tag{48}$$

Here, $\vec{\Omega}_{\text{BHNS}}$ denotes the angular velocity of the BHNS system, $\vec{r}_{\text{CM}}$ is the center of mass of the system, v_r is the radial velocity, r_{BHNS} is the coordinate distance between the NS and BH's centers, and $\vec{\beta}_0$ is determined by the elliptic Eq. (46). Finally, the following boundary conditions are imposed at spatial infinity:

$$\lim_{r\to\infty} \psi = 1, \quad \lim_{r\to\infty} \beta_0^i = 0, \quad \lim_{r\to\infty} \alpha\psi = 1. \tag{49}$$

The boundary conditions in Eq. (49) come from assuming asymptotic flatness at spatial infinity, and furthermore going to a frame that is rotating with $\vec{\Omega}_{\text{BHNS}}$ with respect to the asymptotically flat inertial frame. In order to have the excised surface H corresponding to an apparent horizon (zero expansion for outgoing null rays) in a state of equilibrium, the following boundary conditions are imposed on H [58]:

$$\left\{ \bar{s}^i \bar{D}_i \ln \psi + \frac{1}{4} \bar{h}^{ij} \bar{D}_i \bar{s}_j - \frac{K}{6} \psi^2 + \frac{\psi^{-4}}{4} \bar{A}^{ij} \bar{s}_i \bar{s}_j \right\}_H = 0, \tag{50}$$

$$\left\{ \beta^i - \alpha s^i - \epsilon_{ijk} \Omega_{\text{BH}}^j (x^k - \bar{x}_{\text{BH}}^k) \right\}_H = 0, \tag{51}$$

$$\left\{ \bar{s}^i \bar{D}_i (\alpha\psi) \right\}_H = 0, \tag{52}$$

where, s^i is the outward pointing unit normal on H, $\bar{s}^i = \psi^2 s^i$, the induced metric on H is $h_{ij} = \gamma_{ij} - s_i s_j$ and $\bar{h}^{ij} = \psi^4 h^{ij}$. Moreover, in Eq. (51), ϵ_{ijk} is the totally anti-symmetric symbol and summation over repeated indices is implied, Ω_{BH}^j is a free vector to adjust the BH spin and $\bar{x}_{\text{BH}}^k$ is the coordinate of the BH's center.

When an NS is present we also need matter equations. In order to solve the fluid equations for an NS with an arbitrary spin, a purely spatial vector spin (to encapsulate the rotational part of the fluid) is introduced in [29] as follows:

$$w^i = \epsilon_{ijk} \Omega_{\text{NS}}^j (x^k - \bar{x}_{\text{NS}}^k), \tag{53}$$

in which, $\bar{x}_{\text{NS}}^k$ denotes the coordinate of NS's center and Ω_{NS}^j is a vector related to the spinning motion of the NS. Furthermore, assuming the BHNS is in a quasi-equilibrium state, we introduce an approximate helical time-like Killing vector ξ^μ [29]. Consequently, the fluid 4-velocity can be projected along ξ^μ and a pure spatial vector V^μ [29, 61] as:

$$u^\mu = u^0 (\xi^\mu + V^\mu) = \frac{g^{\mu\nu} \frac{\partial\phi}{\partial x^\nu} + w^\mu}{h}, \tag{54}$$

where scalar ϕ encompasses the irrotational part of the fluid. Using the approximations described in [29, 30, 61] the Euler equation becomes an elliptic PDE for ϕ

$$\frac{c(\rho_0)\,\alpha}{h}\psi^{-4}\bar{\gamma}^{ij}\partial_i\partial_j\phi - \frac{\rho_0\alpha}{h}\psi^{-4}\bar{\gamma}^{ij}\bar{\Gamma}^k_{ij}\partial_k\phi$$
$$+ 2\frac{\rho_0\alpha}{h}\psi^{-5}\bar{\gamma}^{ij}(\partial_i\psi)(\partial_j\phi) + \left(D_i\frac{\rho_0\alpha}{h}\right)\left(D^i\phi\right)$$
$$D_i\left[\frac{\rho_0\alpha}{h}w^i - \rho_0\alpha u^0(\beta^i + \xi^i)\right] = 0. \tag{55}$$

Here, following [30]

$$c(\rho_0) = \rho_0 + \epsilon\rho_{0c}\left(\frac{\rho_{0c} - \rho_0}{\rho_{0c}}\right)^4, \tag{56}$$

ρ_{0c} is rest mass density at the NS's center, and ϵ is a small number (generally 0.1). On the NS's surface we have $\rho_0 = 0$, hence, the following boundary condition is imposed on the surface [77]

$$D^i\phi D_i\rho_0 + w^i D_i\rho_0 - hu^0(\beta^i + \xi^i)D_i\rho_0 = 0. \tag{57}$$

One more equation is needed to close the system of Einstein-Euler equations. This equation is the equation of state (EoS) for NS's matter. In this work, we use a piecewise polytropic EoS. The pieces valid between the densities $\rho_1, \rho_2, \ldots, \rho_n$ [62] are written as

$$P = \begin{cases} K_0\rho_0^{\Gamma_0} & \rho_0 \leq \rho_1 \\ K_1\rho_0^{\Gamma_1} & \rho_1 \leq \rho_0 \leq \rho_2 \\ \vdots & \\ K_{n-1}\rho_0^{\Gamma_{n-1}} & \rho_{n-1} \leq \rho_0 \leq \rho_n, \end{cases} \tag{58}$$

in which, Γ_i denotes the polytropic exponent and K_i the polytropic constant. Moreover, one can write the rest mass density ρ_0, pressure P and specific internal energy ϵ in terms of specific enthalpy h:

$$\rho_0(h) = K_i^{-n_i}\left(\frac{h - 1 - a_i}{n_i + 1}\right)^{n_i},$$
$$P(h) = K_i^{-n_i}\left(\frac{h - 1 - a_i}{n_i + 1}\right)^{n_i+1},$$
$$\epsilon(h) = \frac{a_i + n_i(h - 1)}{n_i + 1}, \tag{59}$$

where $n_i = \frac{1}{\Gamma_i - 1}$ is the polytropic index and a_i's are constants which ensure the continuity of EoS:

$$a_0 = 0,$$
$$a_i = a_{i-1} + \frac{K_{i-1}}{\Gamma_{i-1} - 1}\rho_i^{\Gamma_{i-1}-1} - \frac{K_i}{\Gamma_i - 1}\rho_i^{\Gamma_i - 1}. \tag{60}$$

Lastly, the specific enthalpy h is determined by an algebraic equation in terms of metric variables and fluid velocities [29]:

$$h = \sqrt{L^2 - (D_i\phi + w_i)(D^i\phi + w^i)},$$
$$L^2 = \frac{b + \sqrt{b^2 - 4\alpha^4[(D_i\phi + w_i)w^i]^2}}{2\alpha^2},$$
$$b = [(\xi^i + \beta^i)D_i\phi - C]^2 + 2\alpha^2(D_i\phi + w_i)w^i, \tag{61}$$

in which C denotes a constant of integration that determines the baryonic mass of the NS. We observe that the specific enthalpy connects metric variables and matter variables, in the other words, macro-physics and micro-physics. In the next section, we explain how the solution of these elliptic equations and free parameters (such as the C above) are found in order to construct proper ID.

IV. NUMERICAL METHOD

In this section, we demonstrate the main iteration algorithm for the construction of ID for a BHNS system. Input parameters for `Elliptica` are the baryonic mass M_B, the EoS, and the angular velocity Ω^j_{NS} in Eq. (53) for the spin vector of the NS. Moreover, we can specify the irreducible mass M_{irr} and dimensionless spin χ^j of the BH and also the coordinate distance between the centers of the NS and BH, as well as the orbital angular velocity and the radial velocity of the system. For the free data, we currently use:

$$\bar{\gamma}_{ij} = \delta_{ij}, \tag{62}$$
$$K = 0, \tag{63}$$
$$\bar{u}_{ij} = 0. \tag{64}$$

To account for quasi-equilibrium we also set $\partial_t K = 0$. Since our numerical method is iterative, we need an initial guess for the fields ψ, β^i, α, ϕ, and h on the computational grid. We use a superposition of well known analytic solutions for single objects for this guess. For the star we use a Tolman-Oppenheimer-Volkoff (TOV) solution and for the BH a Schwarzschild solution in isotropic coordinates. The initial value of the enthalpy h then is computed from the TOV star and the irrotational velocity potential is set to $\phi = -\Omega_{BHNS}(y_{NS} - y_{CM})x$, where y_{NS} is the y-coordinate of NS's center and y_{CM} is the y-coordinate of the system's center of mass (note the objects are centered on the y-axis).

Having determined the initial guess, we can now use the Newton-Raphson algorithm (1) to solve the pertinent coupled elliptic PDEs (45), (46), (47) and (55). However, due to the presence of the matter (NS) and the use of the XCTS formalism, several obstacles need to be overcome before we can find a solution with the correct properties, e.g., see [36]. First, the surface of the NS star is not known in advance and is changing at each iteration. As a result, one should find this surface and then adjust the coordinate patches such that the surface is a patch boundary. Otherwise spectral convergence cannot be achieved since the matter fields are not smooth across the star surface. Second, the NS's center and mass start drifting from the desired values which usually causes instabilities. Third, BH's mass and spin deviate from the target values which results in a solution with the wrong physical properties. Finally, the ADM momentum of the system grows which gives rise to instabilities in our iterative procedure, and will also impart a kick on the system's

center of mass visible during subsequent evolution. Such a drift can also cause undesirable coordinate effects when extrapolating gravitational waves to infinity.

Therefore, it is crucial to monitor, adjust and control various parameters and values at each step of the Newton-Raphson algorithm (1) in order to construct ID with the correct properties.

A. Diagnostics

As mentioned in the previous subsection, while we are solving the coupled elliptic Eqs. (45), (46), (47) and (55), there is no guarantee that the mass and spin of BH or the mass of NS reach the target values specified in the parameter file. Moreover, the ADM momentum of the system generally does not vanish. In this section, we show the formulas used to calculate spins, masses, ADM momenta and angular momenta for a BHNS system. We also explain how we attain the desired values for these quantities.

Starting with the NS, the baryonic mass density 4-current is defined [60]:

$$J_B^\mu = \rho_0 u^\mu, \tag{65}$$

thus, the baryonic mass density as measured by an Eulerian observer is $-J_B^\mu n_\mu$ and the baryonic mass is:

$$M_B = \int_{\mathrm{NS}} -J_B^\mu n_\mu dV. \tag{66}$$

In `Elliptica` we write Eq. (66) in terms of the $3+1$ decomposition formalism [60] to compute the baryonic mass:

$$M_B = \int_{\mathrm{NS}} \rho_0 \alpha \psi^6 \sqrt{\bar\gamma} d^3 x, \tag{67}$$

where $\bar\gamma$ is the determinant of $\bar\gamma_{ij}$ and the integration is taken over the volume of the NS. We note that $\rho_0 = \rho_0(h)$ and as shown in Eq. (61) the enthalpy depends on a constant C, i.e., $h = h(C)$, which implies $\rho_0 = \rho_0(C)$. So by adjusting C we can keep the baryonic mass constant at each step of the iteration.

For the BH, we calculate two masses, the irreducible mass M_{irr} and the Christodoulou mass M_{Chr} which are defined as follows respectively:

$$M_{\mathrm{irr}} = \sqrt{\frac{\oint_{\mathrm{AH}} dA}{16\pi}}, \tag{68}$$

$$M_{\mathrm{Chr}} = \sqrt{M_{\mathrm{irr}}^2 + \frac{S_{\mathrm{BH}}^2}{4M_{\mathrm{irr}}^2}}, \tag{69}$$

in which dA is the proper surface element of the apparent horizon defined in `Elliptica` as

$$dA = \sqrt{\gamma_{ij} \frac{\partial x^i}{\partial y^a} \frac{\partial x^j}{\partial y^b}} dy^a dy^b, \tag{70}$$

where y^a are coordinates on the apparent horizon and S_{BH} is the spin of the BH which will be defined shortly. The apparent horizon is a perfect 2-sphere in the coordinates used, owing to the boundary conditions on the BH, i.e., Eqs. (50, 51, and 52). Thus, by adjusting the radius of the apparent horizon, we can drive the irreducible mass to the value prescribed in the parameter file.

For the BH's spin, we use the flat space coordinate rotational Killing vector following [63] on the apparent horizon:

$$\vec\phi_x = -(z - z_c)\vec\partial_y + (y - y_c)\vec\partial_z,$$
$$\vec\phi_y = +(z - z_c)\vec\partial_x + (x - x_c)\vec\partial_z,$$
$$\vec\phi_z = -(y - y_c)\vec\partial_x + (x - x_c)\vec\partial_y, \tag{71}$$

in which (x_c, y_c, z_c) is the coordinate center of the BH and $(\vec\partial_i)_{i\in\{1,2,3\}}$ are the basis vectors associated with the coordinates used. The following integral over the apparent horizon (AH) yields the spin of the BH:

$$S_i = \frac{1}{8\pi} \oint_{\mathrm{AH}} (\vec\phi_i)^j s^k K_{jk} dA. \tag{72}$$

The dimensionless spin is defined by

$$\chi_i := \frac{S_i}{M_{\mathrm{Chr}}^2}. \tag{73}$$

To adjust the value of the spin to the target value we note that $K^{ij} = K^{ij}(\vec\beta)$ on the apparent horizon and $\beta^i = \beta^i(\vec\Omega_{\mathrm{BH}})$ by Eq. (51); thus by adjusting Ω_{BH}^i the spin is controlled.

For the NS spin, we have two options. First we can use the method described in [30],

$$S_i = J_i - \epsilon_{ijk} R_c^j P_k, \tag{74}$$

where J_i, R_c^i, and P_i are, respectively, the angular momentum, center and momentum of the NS defined in [30]. The second option is to evaluate Eq. (72) on the NS's surface. Note that from Ω_{NS}^j in Eq. (53) we cannot directly infer the NS spin, but the values of Ω_{NS}^j for various spins can be found in [18, 30, 36].

Since the chosen free data $\bar\gamma_{ij}$ in Eq. (62) satisfies the quasi-isotropic gauge condition [60] and since K in Eq. (62) meets the asymptotic maximal gauge condition [60], the ADM linear momenta and angular momenta of the system can be defined [60] as follows:

$$P_i^\infty = \frac{1}{8\pi} \lim_{S_t \to \infty} \oint_{S_t} (K_{jk} - K\gamma_{jk})(\vec\partial_i)^j s^k dA, \tag{75}$$

$$J_i^\infty = \frac{1}{8\pi} \lim_{S_t \to \infty} \oint_{S_t} (K_{jk} - K\gamma_{jk})(\vec\phi_i)^j s^k dA, \tag{76}$$

and

$$\vec\phi_x = -(z - z_{\mathrm{CM}})\vec\partial_y + (y - y_{\mathrm{CM}})\vec\partial_z,$$
$$\vec\phi_y = +(z - z_{\mathrm{CM}})\vec\partial_x + (x - x_{\mathrm{CM}})\vec\partial_z,$$
$$\vec\phi_z = -(y - y_{\mathrm{CM}})\vec\partial_x + (x - x_{\mathrm{CM}})\vec\partial_y. \tag{77}$$

We note that K_{ij} is a function of $\vec{\beta}$, while $\vec{\beta}$ itself is a function of $\vec{r}_{CM}$ through Eq. (48). Therefore, by adjusting the freely specifiable parameter $\vec{r}_{CM}$, P_i^∞ can be driven to zero during the solve.

Lastly, to calculate the total ADM mass of the system, we use the following [60]:

$$
M_{ADM} = M_H +
$$
$$
\int_{\Sigma_t} \left[\psi^5 E + \frac{1}{16\pi} \left(\bar{A}_{ij} \bar{A}^{ij} \Psi^{-7} - \bar{R}\psi - \frac{2}{3} K^2 \psi^5 \right) \right] \sqrt{\bar{\gamma}} d^3 x.
\tag{78}
$$

Here, Σ_t is the spatial hypersurface where the ID are constructed and $\bar{\gamma}$ is the determinant of $\bar{\gamma}_{ij}$. M_H is defined as

$$
M_H := -\frac{1}{2\pi} \oint_H \bar{s}^i \bar{D}_i \psi \sqrt{\bar{h}} \, d^2 y,
\tag{79}
$$

where $\bar{h} := \det(\bar{h}_{ij})$ and $\bar{h}_{ij}$ is the induced metric by $\bar{\gamma}_{ij}$ on the excised surface H.

In summary, we use the free parameters of the system, for instance, C and $\vec{r}_{CM}$ among others, to obtain the requested physical properties of the system. In the next subsection, we show how to adjust these parameters (in a slow and smooth way) to reach a stable solution.

B. Iteration Algorithm

Having set the initial fields and free data, the next step is to refine the answers. Loosely speaking, the overall procedure is to start at low resolution and to keep solving the PDEs and adjusting the parameters until the error is below a desired tolerance; then, increase the resolution as often as needed and solve again (still adjusting the parameters). The detailed explanation of this iteration scheme to find a stable, unique and physical solution of Einstein-Euler equations is as follows:

1. Solve each elliptic Eqs. (45), (46), (47) and (55) one after another using the Newton-Raphson method. For each equation, only one step is taken in the method [78]

2. Update the field values $\Psi = \{\phi, \psi, \alpha\psi, B^i\}$ using the relaxation scheme $\Psi = \lambda\Psi_{new} + (1-\lambda)\Psi_{old}$, where Ψ_{new} is the solution just found by the Newton-Raphson method and Ψ_{old} is its previous value (usually $\lambda = 0.2$ [79]).

3. Adjust Ω_{BH}^i to reach the target value χ_*^i for the dimensionless BH's spin using

$$
\Omega_{BH,new}^i = \Omega_{BH,old}^i + \lambda\Delta\chi^i\Omega_{BHNS},
\tag{80}
$$

where $\Delta\chi^i = \chi_*^i - \chi^i$ and χ^i is the current value of dimensionless spin, and λ is usually set to 0.3.

4. Adjust the excision radius of the BH to reach the target value for the irreducible mass. The new radius is

$$
r_{new} = r_{old}\left(1 + \lambda\frac{\Delta M}{M_*}\right).
\tag{81}
$$

Here $\Delta M = M_* - M$, M_* is the target value, M is the current value of the irreducible mass of the BH, and λ is generally set to 0.3.

5. Find the constant C in Eq. (67) to achieve the prescribed value for baryonic mass of the NS.

6. Adjust $\vec{r}_{CM}$ to drive the linear ADM momentum of the BHNS system to zero. The linear momentum in the z-direction is very small ($\frac{|P_z^\infty|}{M_{ADM}} < 10^{-9}$) therefore we only need to adjust x_{CM} and y_{CM} as follows:

$$
x_{CM,new} = x_{CM,old} + \lambda\frac{P_y^\infty}{\Omega_{BHNS}M_{ADM}},
$$
$$
y_{CM,new} = y_{CM,old} - \lambda\frac{P_x^\infty}{\Omega_{BHNS}M_{ADM}},
\tag{82}
$$

in which, M_{ADM} is the total ADM mass of the system; λ generally is set to 0.2.

7. Update the enthalpy in the patches that cover the NS interior using Eq. (61) with the relaxation method $h = \lambda h_{new} + (1-\lambda)h_{old}$; here, λ is usually set to 0.1.

8. If we want to also determine Ω_{BHNS} (and not just use a given value for it), we use the force balance Eq. (83). Specifically, we find Ω_{BHNS} such that the following holds at the NS's center [61]:

$$
\partial_i \ln[\alpha^2 - (\beta^i + \xi^i + \frac{w^i}{hu^0})(\beta_i + \xi_i + \frac{w_i}{hu^0})] = -2\partial_i \ln\Gamma,
$$
$$
\Gamma = \frac{\alpha u^0[1 - (\beta^i + \xi^i + \frac{w^i}{hu^0})\frac{D_i\phi}{\alpha^2 hu^0} - \frac{w_i w^i}{(\alpha^2 hu^0)^2}]}{\sqrt{1 - (\beta^i + \xi^i + \frac{w^i}{hu^0})(\beta_i + \xi_i + \frac{w_i}{hu^0})\frac{1}{\alpha^2}}}.
\tag{83}
$$

Here $\partial_i = \frac{\partial}{\partial x^i}$ and Γ is kept fixed. We use this along the line connecting the centers of the two objects.

9. Extrapolate matter fields ϕ, w^i, and h outside the NS. This serves two purposes. First, a smoothly extrapolated h helps with step 11, where we use a root finder to update the star surface location. Second, to interpolate the matter fields to the new grid (of step 12) we need the values of ϕ and w^i, even outside the star, if the NS surface expands in step 11. To extrapolate w^i outside, we apply Eq. (53). For the fields ϕ and h, at each collocation point on the NS's surface with coordinate radius r_0, we extrapolate them using:

$$
f(r) = \left(a + \frac{b}{r}\right)\exp\left(-c_0\frac{r}{r_0}\right),
\tag{84}
$$

where the coefficients a and b are found by demanding C^1 continuity across the surface, c_0 is generally set to 0.01, and r is the coordinate distance from the star's center. Eq. 84 damps the matter fields exponentially so that the NS's surface cannot expand too much in one iteration. We have found that this particular, continuous extrapolation function works best within the iterative solve.

10. To avoid the drifting of the NS's center located at $\vec{r}_0$ we shift the NS matter (given by h) to keep the star's center fixed. Using a Taylor expansion (for small shifts) we find that h needs to be updated as follows:

$$
h_{new}(\vec{r}) = h_{old}(\vec{r}) - (\vec{r} - \vec{r}_0) \cdot \vec{\nabla}h_{old}(\vec{r}).
\tag{85}
$$

11. Find the new surface of the NS. Using a root finder we find r such that

$$h(r) - 1 = 0, \qquad (86)$$

where r is the distance from the NS's center. This yields the new location of the NS's surface in spherical coordinates (r, θ, ϕ) and thereby determines the pertinent $\sigma(X, Y)$ introduced in Sec. II D.

12. If the resolution is changed or the surface of NS or BH has been changed, create a new grid and set the fields values using a spectral interpolation from the previous grid. At this step a new grid with the new surface fitting patches for the NS and BH surface are created and the values of the fields are interpolated from the previous grid.

13. Exit, if all the iterations at all requested resolutions have reached our error criterion, otherwise proceed to step 1. The criterion we use to exit is when Hamiltonian and momenta constraints, Eqs. (87) and (88), are no longer decreasing (because they have reached the truncation error for the resolution).

To give some context, the number of outermost iterations for this BHNS system is generally about 250 at the lowest resolution (generally 12 points at each direction in each patch). It decreases when increasing the resolution and ends up at about 50 at the highest resolution (generally 20 points at each direction per patch). The maximum resolution itself is determined by the maximum constraint violations we are willing to tolerate. Using shared-memory multiprocessing, the ID computation for this configuration generally takes ~ 104 hours of actual wall clock time on a single Intel Xeon node with 20 cores of FAU's KOKO cluster. A summary of this iterative scheme is shown in algorithm (3).

V. RESULTS

In this section, we present several validations of our implementation and provide first proof-of-principle dynamical simulations. In sec V A we test the spectral convergence of the code for a non-spinning BHNS system (named SXS1) and for a system (named S1S2), where both BH and NS have high spins in arbitrary directions. In sec V B we construct different sets of ID to confirm the validity of the data when comparing to analytical approximations. We further push the NS spin close to breakup value. Finally in sec V C we evolve the ID (summarized in Tab. I) using the BAM code [64–68], to check that the binaries behave as expected. In all configurations, the EoS is polytropic with $K = 92.12$ and $\Gamma = 2$ unless otherwise is mentioned.

A. Spectral Convergence Test

The first test is the verification of exponential convergence expected for a spectral code. We measure the vio-

---

Algorithm 3 main iteration scheme to find the ID for a BHNS system

1: **for** each resolution **do**
2:     **while** constraints have not plateaued yet **do**
3:         Solve the elliptic Eqs. (45), (46), (47), and (55) one after another;
4:         Update the fields being solved for using the relaxation scheme $\Psi = \lambda \cdot \Psi_{\text{new}} + (1 - \lambda) \cdot \Psi_{\text{old}}$;
5:         Adjust the BH parameter Ω_{BH}^i to satisfy the target dimensionless spin;
6:         Adjust the BH radius to satisfy the target irreducible mass;
7:         Find Euler's equation constant C to fix the NS baryonic mass;
8:         Adjust the system's center of mass $\vec{r}_{\text{CM}}$ to drive the ADM linear momenta to zero;
9:         Update the enthalpy using Eq. (61) with the relaxation method $h = \lambda h_{\text{new}} + (1 - \lambda) h_{\text{old}}$.
10:         If desired, use the force balance Eq. (83) to adjust Ω_{BHNS}.
11:         Extrapolate matter fields ϕ, w^i and h to the outside of the NS.
12:         Shift the matter to keep the NS's center fixed.
13:         Find the new location of NS surface;
14:         If needed, create a new grid and interpolate the fields values from the previous grid.
15:     **end while**
16: **end for**

TABLE I: The ID that are evolved via the BAM code in this work. We have listed the irreducible mass M_{irr}, baryonic mass M_{B}, the mass ratio of the binary q, the initial coordinate distance s, the total dimensionless spin of each object χ_{BH}, χ_{NS}, and the EoS. Here K92 denotes a polytropic EoS with $K = 92.12$ and $\Gamma = 2$. The S1S2 ID has $\vec{\chi}_{BH} = (-0.46, -0.46, -0.46)$ and $\vec{\chi}_{NS} = (0, 0.32, 0.32)$.

Name	M_{irr}	M_{B}	q	s	χ_{BH}	χ_{NS}	EoS
SXS1	8.4	1.4	6.5	82	0.0	0.0	$K92$
S1S2	5.2	1.4	4	56	0.8	0.45	$K92$

---

lation of Hamiltonian and momentum constraints [60, 61] using:

$$H := R - K_{ij}K^{ij} + K^2 - 16\pi E = 0, \qquad (87)$$
$$M^i := D_j(K^{ij} - \gamma^{ij}K) - 8\pi j^i = 0. \qquad (88)$$

We construct two sets of ID. One is SXS1 of Tab. I where both BH and NS have zero spin. The other system, S1S2, has $\vec{\chi}_{BH} = (-0.46, -0.46, -0.46)$ and $\vec{\chi}_{NS} = (0, 0.32, 0.32)$. Figs. 4 and 5, show the L^2 norm of H and M^i of Eqs. (87) and (88) after the final iteration at each resolution. Evidently, the constraint violations decay exponentially for both ID sets, i.e., we find the expected spectral convergence.
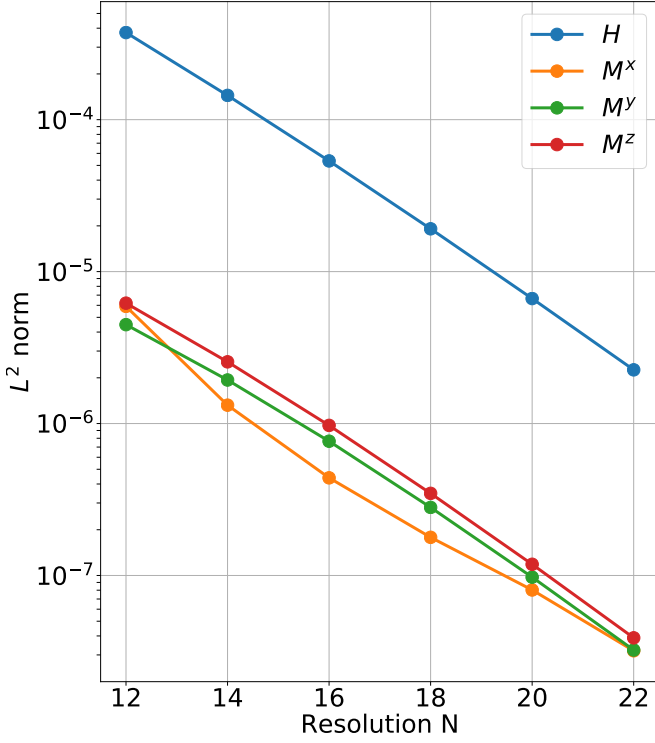
FIG. 4: Shown is the spectral convergence of the Hamiltonian and Momentum constraints for the `SXS1` ID with respect to resolution. The vertical axis shows the L^2 norm of each constraint. The horizontal axis shows the number of points used in each direction inside each cubed sphere patch.
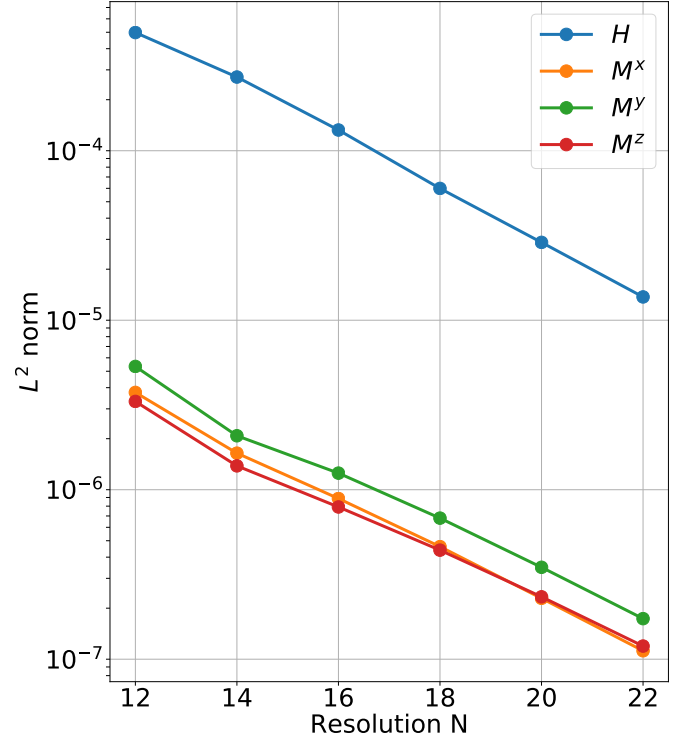

FIG. 5: Shown is the spectral convergence of the Hamiltonian and Momentum constraints for the `S1S2` ID with respect to resolution. The vertical axis shows the L^2 norm of each constraint. The horizontal axis shows the number of points in each direction in a cubed spherical patch.

B. Quasi-equilibrium Configurations

We next compare our numerical ID with known analytic approximations. As such, we compute the binding energy E_b [80] of a sequence in which $M_{\mathrm{NS}} = 1.31$, $M_{\mathrm{Chr}} = 5.81$, and $\vec{\chi}_{BH} = \pm 0.1\hat{z}$ are kept constant but the orbital angular velocity (Ω_{BHNS}) is varied. We use the 3.5 post-Newtonian (PN) approximation from [69] plus the next-to-next-to-leading order correction of spin-orbit (SO) interaction from [69] and compare it against our numerical results. As shown in Fig. 6 we find good agreement between E_b computed by PN+SO and `Elliptica`.

In order to test the NS spin limits in `Elliptica`, we construct a sequence of BHNS ID where $\chi_{BH} = 0$, $M_{\mathrm{irr}} = 5.2$ and $M_{\mathrm{B}} = 1.4$ are kept constant but Ω_{NS}^z is increased until the maximum achievable value (0.0229) for this EoS is reached. This maximum Ω_{NS}^z is given by the value from which onwards `Elliptica`'s NS surface finder fails to converge (presumably because Ω_{NS}^z then is too close to the breakup spin of the NS). We use Eq. (74) to calculate the NS spin χ_{NS} for each Ω_{NS}^z. As shown in Fig. 7, for low values of Ω_{NS}^z the spin χ_{NS} is a linear function of Ω_{NS}^z, but for higher Ω_{NS}^z the spin rises more and more quickly.

C. Evolution Tests

To further test the IDs, we evolve them using the `BAM` code [68]. Since for the construction of the ID we use an excision method, but want to use the moving puncture method for the evolution, we have to fill the inside of the BH with smooth data [71–74]. Here, smooth means that the fields have to be at least C^2 across the apparent horizon. In order to have low eccentricity in the SXS1 (see Tab. I) inspiral, we perform three eccentricity-reduction steps as in [30] to obtain the target eccentricity $\lesssim 10^{-3}$. Fig. 8 shows the results of this eccentricity reduction algorithm for each step.

In Fig. 9, we plot the trajectories of the BH and NS from the eccentricity reduced simulation of `SXS1`. As expected, they spiral in and merge on low eccentricity orbits, without any visible drift of the center of mass.

The trajectories show an initially straight motion for both objects, which is related to adjustments of the gauges. To keep the plot clean, we only show the trajectories of the BH and NS's centers, and not the extent of the two objects. We track the location of the NS by the minimum in the lapse. The small wiggles and/or jumps around merger time are mostly due to the minimum of the lapse no longer being an appropriate indicator for the center of the NS.
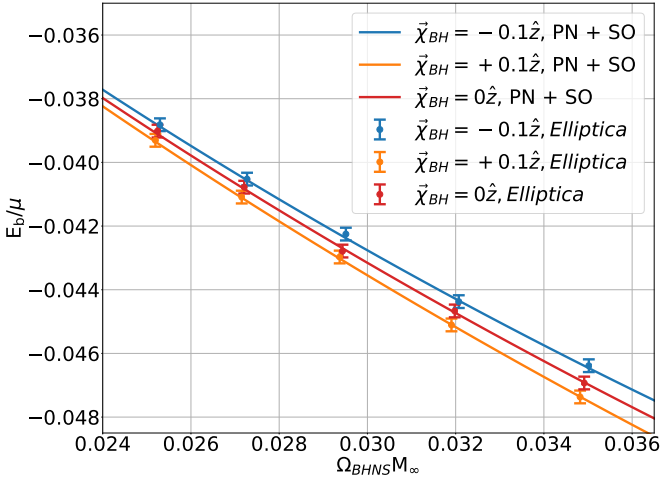
FIG. 6: The binding energy (E_b) of a BHNS system versus orbital angular velocity (Ω_{BHNS}), where the NS spin is zero and $\vec{\chi}_{BH} = \pm 0.1\hat{z}$. Here, $M_\infty = M_{\mathrm{Chr}} + M_{\mathrm{NS}}$ and $\mu = \frac{M_{\mathrm{Chr}} M_{\mathrm{NS}}}{M_\infty}$. The small vertical bar is the error in the numerically computed Eq. (78). It is estimated by setting analytical ID for a Schwarzschild BH and then using Eq. (78) to numerically find M_{ADM}. The difference between this numerical value and the known ADM mass yields the error.

SXS1 has the same physical parameters as the first run in the SXS collaboration catalog [75] (SXS:BHNS:0001). In Fig. 10, we depict the extracted gravitational wave signal emitted by this system, in the form of the dominant mode ($l = m = 2$). The waveforms are plotted against the retarded time u calculated as

$$u = t - r_* = t - r_{extr} - 2M \ln(r_{extr}/2M - 1), \quad (89)$$

where r_{extr} is the extraction radius, set to $\sim 1500\ M_\odot$, and M is the sum of the isolated BH and NS's masses. As is visible in the bottom panel, our result agrees with the configuration SXS:BHNS:0001 from the SXS collaboration catalog [75]. Note that both systems share the same physical properties, i.e., masses, spins, and EoS, but start from different initial separations.

As a second example, we evolve the S1S2 BHNS system (already described in Sec. V A) that has $\vec{\chi}_{BH} = (-0.46, -0.46, -0.46)$ and $\vec{\chi}_{NS} = (0, 0.32, 0.32)$. This system presents a setup with misaligned spins (with respect to the initial orbital angular momentum), which leads to precession. Fig. 11 depicts the 3-dimensional trajectories of the objects. The initial coordinate distance between BH and NS is $s = 56$. This leads to 3 orbits before the merger. Further analysis and gravitational waves of such systems, with longer inspiral, are left for future work.

VI. SUMMARY

The construction of ID for self-gravitating astrophysical systems is an indispensable task for an accurate dy-
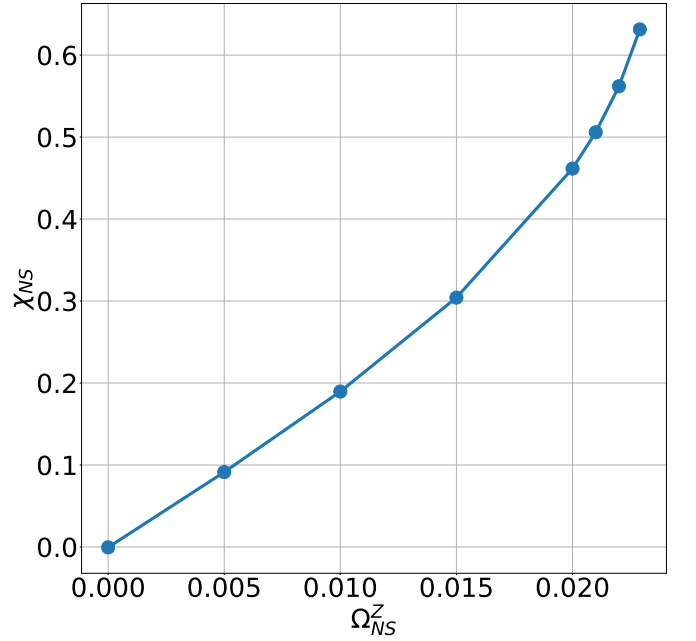

FIG. 7: Dimensionless neutron star spin χ_{NS} as a function of spin-angular velocity parameter Ω_{NS}^z. Here, $\chi_{BH} = 0$, $M_{\mathrm{irr}} = 5.2$, $M_{\mathrm{B}} = 1.4$, and the EoS is polytropic with $K = 92.12$ and $\Gamma = 2$. The maximum χ_{NS} we can achieve is 0.63. The breakup spin for an NS is approximately 0.7 [70].

namical evolution, and consequently for an understanding of compact binary coalescences. For this purpose, we have developed a new code, Elliptica, which provides infrastructure to solve the constraint equations and produce ID of such astrophysical systems.

The current version of the code uses Chebyshev polynomials of the first kind to spectrally expand the fields over the computational grid. This grid is covered by several cubic and cubed sphere coordinate patches. To solve the constraint equations, which are coupled nonlinear elliptic PDEs, a Newton-Raphson method is used in which the linearized equations are solved by a SCDD method and the Jacobian is set analytically. Furthermore, the code supports polytropes and piecewise polytropic equations of state for the NS. The NS can have a high spin in an arbitrary direction. The BH can also have an arbitrary spin direction with a maximum dimensionless spin magnitude ~ 0.8.

For testing and proof of concept, we have constructed ID for various cases of spinning and non-spinning BH and NS for BHNS systems. We have further verified that sequences of BHNS ID with $\chi_{\mathrm{BH}} = \pm 0.1\hat{z}$ agree well with analytical PN approximations. Moreover, we have constructed ID for BHNS system SXS1 that has the same physical parameters as SXS:BHNS:0001 from the SXS catalog [75], as well as BHNS system S1S2 where both BH an NS have generic spins. For both cases we have confirmed the expected spectral convergence of their Hamiltonian and momentum constraints. We have also evolved both SXS1 and S1S2 and verified that their or-
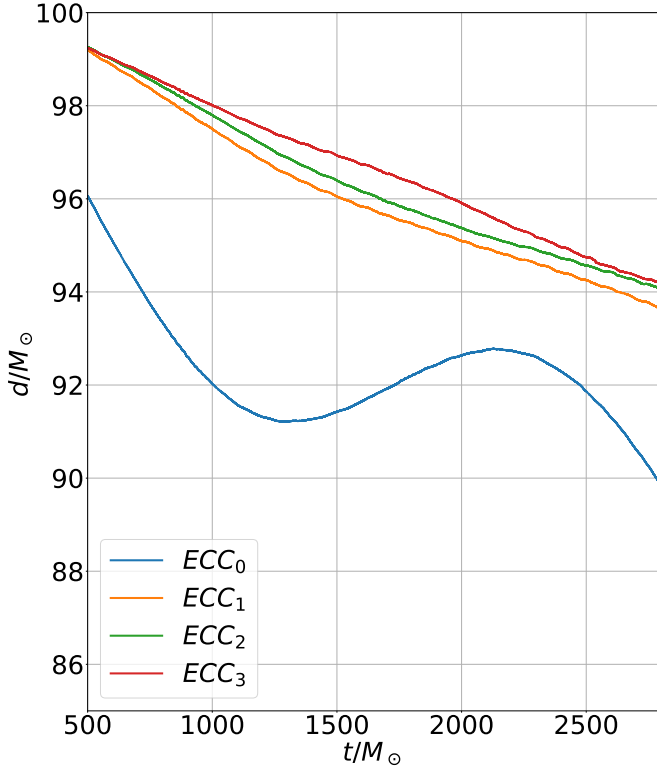
FIG. 8: The proper distance (d) between BH and NS versus time (t) of the SXS1 system for different Ω_{BHNS}. After evolving for about 3 orbits the eccentricity reduction algorithm in [30] was used to adjust Ω_{BHNS} and v_r. The eccentricity of each curve is as follows: $ECC_0 = 3.1 \times 10^{-2}$, $ECC_1 = 3.8 \times 10^{-3}$, $ECC_2 = 3.4 \times 10^{-3}$, and $ECC_3 = 9.0 \times 10^{-4}$. For the reduction we exclude the first $500 M_\odot$ so that the required fits are not affected by the initial gauge adjustments (as mentioned in V C).
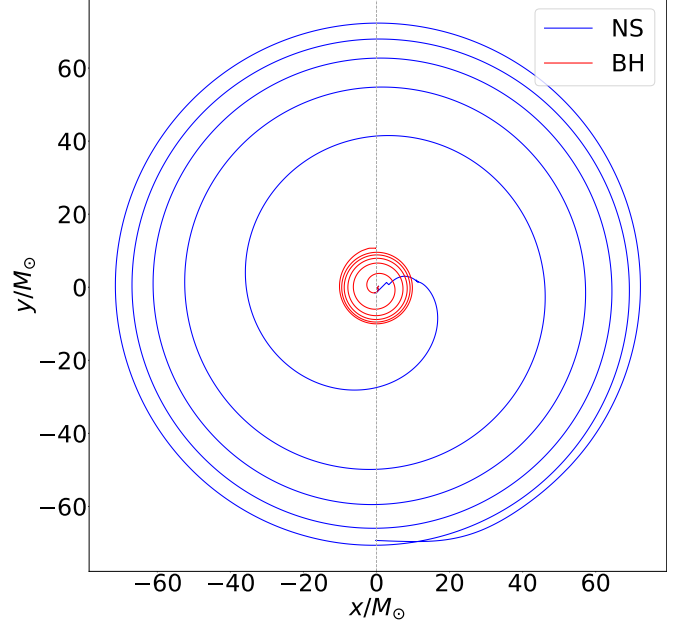
FIG. 9: Trajectories of BH (in red) and NS (in blue) up until merger as described in V C. The objects start their evolution aligned on the $x = 0$ axis (dashed line), as computed from Elliptica. The tracks follow the evolution of the minimum of the lapse relative to each object. The straight motion at the beginning of each trajectory is due to gauge adjustments.

bits show the expected inspiral behavior. Furthermore, we have confirmed that the emitted gravitational waves of the SXS1 system agree well with results from prior studies.

In the future, we plan to achieve maximal BH spin angular momentum by changing the conformal metric in the vicinity of the BH like in [36] or use a puncture method as in [76].

Acknowledgments

[1] B. P. Abbott et al. (Virgo, LIGO Scientific), Phys. Rev. Lett. **116**, 061102 (2016), 1602.03837.

[2] B. P. Abbott et al. (LIGO Scientific, Virgo), Phys. Rev. Lett. **119**, 161101 (2017), 1710.05832.

[3] R. Abbott et al. (LIGO Scientific, VIRGO, KAGRA), Astrophys. J. Lett. **915**, L5 (2021), 2106.15163.

[4] B. P. Abbott et al. (LIGO Scientific, Virgo), Phys. Rev. **X9**, 031040 (2019), 1811.12907.

[5] R. Abbott et al. (LIGO Scientific, Virgo), Phys. Rev. X **11**, 021053 (2021), 2010.14527.

[6] J. M. Ezquiaga and M. Zumalacrregui, Phys. Rev. Lett. **119**, 251304 (2017), 1710.05901.
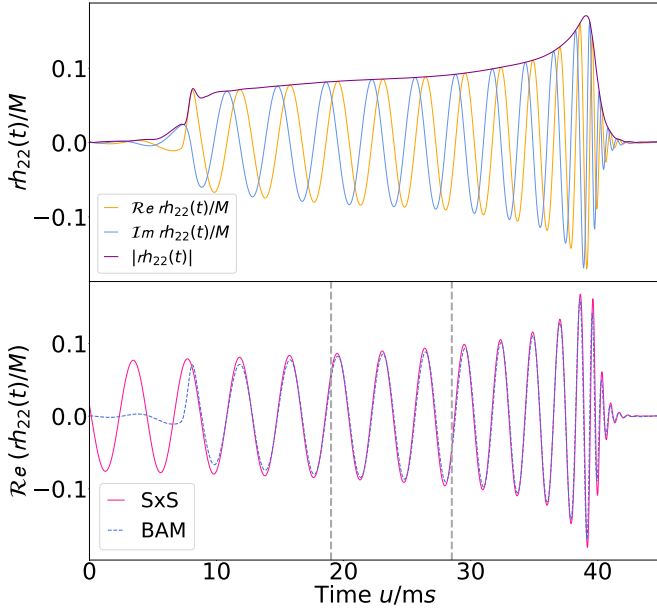
FIG. 10: Top: Extracted gravitational wave strains (rh_{22}) for the $l = m = 2$ mode of the SXS1 system from table I. Bottom: Comparison of rh_{22} between the SXS:BHNS:0001 system from [75] (solid line) and the SXS1 system (dashed line). For visual clarity, only the real part of both strains ($\mathcal{R}e(rh_{22})$) is shown in the plot. The alignment interval is marked with the vertical dashed line. Waveforms in both panels are plotted against the retarded time u defined at Eq. (89) and M is the sum of the isolated BH and NSs masses.
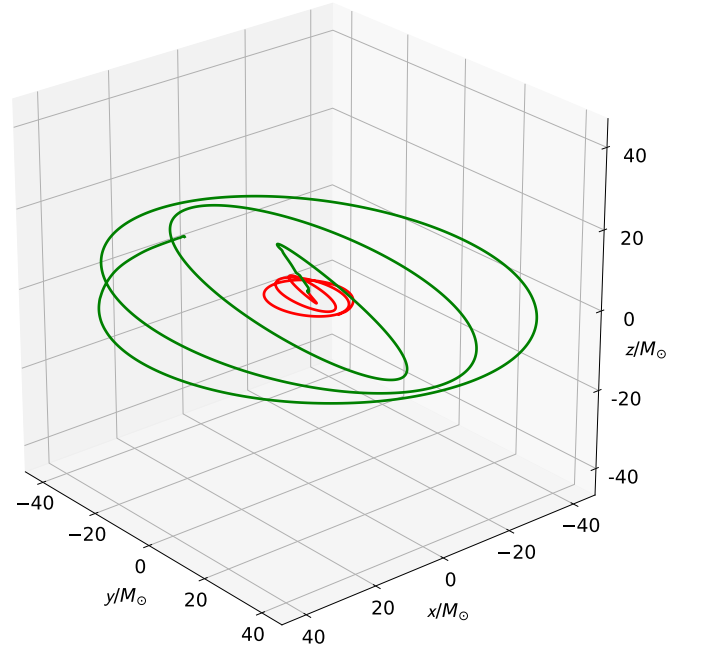

FIG. 11: 3-D inspiraling and precessing trajectories of the NS (green trajectory) and BH (red trajectory). Here, $\vec{\chi}_{BH} = (-0.46, -0.46, -0.46)$, $\vec{\chi}_{NS} = (0, 0.32, 0.32)$, and the initial separation $s = 56$.

[7] T. Baker, E. Bellini, P. G. Ferreira, M. Lagos, J. Noller, and I. Sawicki, Phys. Rev. Lett. **119**, 251301 (2017), 1710.06394.

[8] P. Creminelli and F. Vernizzi, Phys. Rev. Lett. **119**, 251302 (2017), 1710.05877.

[9] R. Abbott et al. (LIGO Scientific, Virgo), Phys. Rev. D **103**, 122002 (2021), 2010.14529.

[10] B. P. Abbott et al. (LIGO Scientific, Virgo), Phys. Rev. D **100**, 104036 (2019), 1903.04467.

[11] B. P. Abbott et al. (LIGO Scientific, Virgo), Phys. Rev. Lett. **123**, 011102 (2019), 1811.00364.

[12] B. P. Abbott et al. (LIGO Scientific, Virgo), Phys. Rev. X **9**, 011001 (2019), 1805.11579.

[13] B. P. Abbott et al. (LIGO Scientific, Virgo), Phys. Rev. Lett. **121**, 161101 (2018), 1805.11581.

[14] S. De, D. Finstad, J. M. Lattimer, D. A. Brown, E. Berger, and C. M. Biwer, Phys. Rev. Lett. **121**, 091102 (2018), 1804.08583.

[15] B. P. Abbott et al. (KAGRA, LIGO Scientific, Virgo, VIRGO), Living Rev. Rel. **21**, 3 (2018), 1304.0670.

[16] K. Uryu and A. Tsokaros, Phys. Rev. D **85**, 064014 (2012), 1108.3065.

[17] A. Tsokaros, K. Uryū, and L. Rezzolla, Phys. Rev. D **91**, 104030 (2015), 1502.05674.

[18] L. J. Papenfort, S. D. Tootle, P. Grandclément, E. R. Most, and L. Rezzolla, Phys. Rev. D **104**, 024057 (2021), 2103.09911.

[19] Lorene, Langage Objet pour la RElativité NumériquE, URL http://www.lorene.obspm.fr.

[20] P. Grandclement, Phys. Rev. D **74**, 124002 (2006), [Erratum: Phys.Rev.D 75, 129903 (2007)], gr-qc/0609044.

[21] K. Taniguchi, T. W. Baumgarte, J. A. Faber, and S. L. Shapiro, Phys. Rev. D **74**, 041502 (2006), gr-qc/0609053.

[22] K. Taniguchi, T. W. Baumgarte, J. A. Faber, and S. L. Shapiro, Phys. Rev. D **75**, 084005 (2007), gr-qc/0701110.

[23] K. Taniguchi, T. W. Baumgarte, J. A. Faber, and S. L. Shapiro, Phys. Rev. D **77**, 044003 (2008), 0710.5169.

[24] K. Kyutoku, M. Shibata, and K. Taniguchi, Phys. Rev. D **79**, 124018 (2009), 0906.0889.

[25] K. Kyutoku, H. Okawa, M. Shibata, and K. Taniguchi, Phys. Rev. D **84**, 064018 (2011), 1108.1189.

[26] K. Kyutoku, K. Kawaguchi, K. Kiuchi, M. Shibata, and K. Taniguchi, Phys. Rev. D **103**, 023002 (2021), 2009.03896.

[27] T. Assumpcao, L. R. Werneck, T. P. Jacques, and Z. B. Etienne (2021), 2111.02424.

[28] W. Tichy, Class. Quant. Grav. **26**, 175018 (2009), 0908.0620.

[29] W. Tichy, Phys. Rev. D **86**, 064024 (2012), 1209.5336.

[30] W. Tichy, A. Rashti, T. Dietrich, R. Dudi, and B. Brügmann, Phys. Rev. D **100**, 124046 (2019), 1910.09690.

[31] N. L. Fischer and H. P. Pfeiffer, Phys. Rev. D **105**, 024034 (2022), 2108.05826.

[32] N. L. Vu et al., Phys. Rev. D **105**, 084027 (2022), 2111.06767.

[33] H. P. Pfeiffer, L. E. Kidder, M. A. Scheel, and S. A. Teukolsky, Comput. Phys. Commun. **152**, 253 (2003), gr-qc/0202096.

[34] F. Foucart, L. E. Kidder, H. P. Pfeiffer, and S. A. Teukolsky, Phys. Rev. D **77**, 124051 (2008), 0804.3787.

[35] N. Tacik et al., Phys. Rev. D **92**, 124012 (2015), [Erratum: Phys.Rev.D 94, 049903 (2016)], 1508.06986.

[36] N. Tacik, F. Foucart, H. P. Pfeiffer, C. Muhlberger, L. E. Kidder, M. A. Scheel, and B. Szilágyi, Class. Quant. Grav. **33**, 225012 (2016), 1607.07962.

[37] M. Ansorg, B. Bruegmann, and W. Tichy, Phys. Rev. D **70**, 064011 (2004), gr-qc/0404056.

[38] M. Ansorg, Phys. Rev. D **72**, 024018 (2005), gr-qc/0505059.

[39] B. Khamesra, M. Gracia-Linares, and P. Laguna, Class. Quant. Grav. **38**, 185008 (2021), 2101.10252.

[40] I. Rácz and J. Winicour, Phys. Rev. D **91**, 124013 (2015), 1502.06884.

[41] I. Rácz, Class. Quant. Grav. **33**, 015014 (2016), 1508.01810.

[42] I. Rácz and J. Winicour, Class. Quant. Grav. **35**, 135002 (2018), 1712.03294.

[43] K. Csukás and I. Rácz, Class. Quant. Grav. **37**, 155006 (2020), 1911.02900.

[44] H. R. Rüter, D. Hilditch, M. Bugner, and B. Brügmann, Phys. Rev. D **98**, 084044 (2018), 1708.07358.

[45] A. Rashti, rashti-alireza/cpi: v3.0 (2021), URL https://doi.org/10.5281/zenodo.5495885.

[46] J. P. Boyd, Chebyshev and Fourier Spectral Methods (Springer-Verlag, Berlin New York, 2000), ISBN 978-3-540-51487-9, URL http://books.google.com/books?hl=en&lr=&id=lEWnQWyzLQYC&oi=fnd&pg=PR10&dq=Chebyshev+and+Fourier+Spectral+Methods&ots=WRdKyCo9nt&sig=IO1U70ic8UM3-CDhwYegnCuh7fE.

[47] P. Grandclement and J. Novak, Living Rev. Rel. **12**, 1 (2009), 0706.2286.

[48] Y. Saad, Iterative Methods for Sparse Linear Systems (Society for Industrial and Applied Mathematics (SIAM, 3600 Market Street, Floor 6, Philadelphia, PA 19104, Philadelphia, Pa, 2003), ISBN 9780898718003.

[49] R. Burden, Numerical analysis (Brooks/Cole, Cengage Learning, Boston, MA, 2011), ISBN 9780538733519.

[50] L. R. Taylor, W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, Numerical Recipes: The Art of Scientific Computing, vol. 56 (Cambridge University Press, Cambridge, UK New York, 1987), ISBN 978-0521880688.

[51] in Table of Integrals, Series, and Products, edited by D. Zwillinger, V. Moll, I. S. Gradshteyn, and I. M. Ryzhik (Academic Press, Boston, 2014), pp. 25–62, eighth edi ed., ISBN 978-0-12-384933-5, URL http://www.sciencedirect.com/science/article/pii/B9780123849335000011.

[52] T. A. Davis, Direct Methods for Sparse Linear Systems (Society for Industrial and Applied Mathematics, Philadelphia, 2006), ISBN 978-0-89871-613-9, URL http://epubs.siam.org/doi/book/10.1137/1.9780898718881.

[53] T. Frankel, The geometry of physics : an introduction (Cambridge University Press, Cambridge New York, 2012), ISBN 1107602602.

[54] T. A. Davis, ACM Transactions on Mathematical Software **30**, 196 (2004), ISSN 00983500, URL https://doi.org/10.1145/992200.992206.

[55] C. Ronchi, R. Iacono, and P. S. Paolucci, Journal of Computational Physics **124**, 93 (1996), ISSN 00219991, URL https://linkinghub.elsevier.com/retrieve/pii/S0021999196900479.

[56] H. P. Pfeiffer and J. W. York, Jr., Phys. Rev. D **67**, 044022 (2003), gr-qc/0207095.

[57] J. W. York, Jr., Phys. Rev. Lett. **82**, 1350 (1999), gr-qc/9810051.

[58] G. B. Cook and H. P. Pfeiffer, Phys. Rev. D **70**, 104016 (2004), gr-qc/0407078.

[59] S. Brandt and B. Bruegmann, Phys. Rev. Lett. **78**, 3606 (1997), gr-qc/9703066.

[60] E. Gourgoulhon, 3 + 1 Formalism in General Relativity, July (Springer, Berlin New York, 2011), ISBN 978-3-642-24525-1.

[61] W. Tichy, Rept. Prog. Phys. **80**, 026901 (2017), 1610.03805.

[62] J. S. Read, B. D. Lackey, B. J. Owen, and J. L. Friedman, Phys. Rev. D **79**, 124032 (2009), 0812.2163.

[63] M. Campanelli, C. O. Lousto, Y. Zlochower, B. Krishnan, and D. Merritt, Phys. Rev. D **75**, 064030 (2007), gr-qc/0612076.

[64] B. Brügmann, J. A. González, M. D. Hannam, S. Husa, and U. Sperhake, in High Performance Computing in Science and Engineering, Garching/Munich 2007, edited by S. Wagner, M. Steinmetz, A. Bode, and M. Brehm (Springer Berlin Heidelberg, Berlin, Heidelberg, 2009), pp. 3–18, ISBN 978-3-540-69182-2.

[65] M. Thierfelder, S. Bernuzzi, and B. Bruegmann, Phys. Rev. D **84**, 044012 (2011), 1104.4751.

[66] T. Dietrich, S. Bernuzzi, M. Ujevic, and B. Brügmann, Phys. Rev. D **91**, 124041 (2015), 1504.01266.

[67] S. Bernuzzi and T. Dietrich, Phys. Rev. D **94**, 064062 (2016), 1604.07999.

[68] S. V. Chaurasia, T. Dietrich, and S. Rosswog (2021), 2107.08752.

[69] L. Blanchet, Living Rev. Rel. **17**, 2 (2014), 1310.1528.

[70] K.-W. Lo and L.-M. Lin, Astrophys. J. **728**, 12 (2011), 1011.3563.

[71] J. D. Brown, O. Sarbach, E. Schnetter, M. Tiglio, P. Diener, I. Hawke, and D. Pollney, Phys. Rev. D **76**, 081503 (2007), 0707.3101.

[72] J. A. Faber, T. W. Baumgarte, Z. B. Etienne, S. L. Shapiro, and K. Taniguchi, Phys. Rev. D **76**, 104021 (2007), 0708.2436.

[73] J. D. Brown, Phys. Rev. D **77**, 044018 (2008), 0705.1359.

[74] G. Reifenberger and W. Tichy, Phys. Rev. D **86**, 064003 (2012), 1205.5502.

[75] Sxs gravitational waveform database, https://data.black-holes.org/waveforms/catalog.html.

[76] I. Ruchlin, J. Healy, C. O. Lousto, and Y. Zlochower, Phys. Rev. D **95**, 024033 (2017), 1410.8607.

[77] In practice we use normal vector on the NS's surface instead of $D_i\rho_0$

[78] Please note that we do not solve all coupled equations at once, i.e., each elliptic equation is solved separately one after another. In other words, while solving each equation for the field, the other coupled fields are treated like fixed source terms. However, since we use an iterative procedure we eventually find the solution to the complete system of coupled equations.

[79] To determine the value of the relaxation parameter λ here, we have conducted numerical experiments, with the goal of finding a λ such that our iteration algorithm converges quickly enough. Below we use this kind of relaxation also when we update or adjust other quantities, albeit with different values.

[80] $E_b = M_{\mathrm{ADM}} - M_{\mathrm{Chr}} - M_{\mathrm{NS}}$, where M_{Chr} is defined at Eq. 69 and M_{NS} is the gravitational mass of the NS in

isolation.