PHYSICAL REVIEW D **107,** 104059 (2023)

Quantifying uncertainties in general relativistic magnetohydrodynamic codes

Pedro L. Espino,[1,2] Gabriele Bozzola,[3] and Vasileios Paschalidis[3,4]
[1]Department of Physics, The Pennsylvania State University, University Park, Pennsylvania 16802, USA
[2]Department of Physics, University of California, Berkeley, California 94720, USA
[3]Department of Astronomy, University of Arizona, Tucson, Arizona 85721, USA
[4]Department of Physics, University of Arizona, Tucson, Arizona 85721, USA

(Received 14 October 2022; accepted 1 May 2023; published 30 May 2023)

In this paper, we show that similar open-source codes for general relativistic (magneto)hydrodynamic [GR(M)HD] produce different results for key features of binary neutron star mergers. First, we present a new open-source version of the publicly available IllinoisGRMHD code that provides support for realistic, finite temperature equations of state. After stringent tests of our upgraded code, we perform a code comparison between GRHydro, IllinoisGRMHD, Spritz, and WhiskyTHC, which implement the same physics, but slightly different computational methods. The benefit of the comparison is that all codes are embedded in the Einstein Toolkit suite, hence their only difference is algorithmic. We find similar convergence properties, fluid dynamics, and gravitational waves, but different merger times, remnant lifetimes, and gravitational wave phases. Such differences must be resolved before the postmerger dynamics modeled with such simulations can be reliably used to infer the properties of nuclear matter especially in the era of precision gravitational wave astronomy.

DOI: 10.1103/PhysRevD.107.104059

I. INTRODUCTION

The detection of gravitational waves (GWs) from the merger of a binary neutron star (BNS) [1] has established the need for a clearer understanding of BNS systems. To gain insight into the physics of BNS mergers, observations associated with such events—be they in the GW or electromagnetic (EM) spectrum—must be analyzed with either simplified analytical models which parametrize the dominant physical phenomena in the system, e.g., [2–6] or with the use of accurate numerical relativity (NR) simulations. Parametric/phenomenological models of BNS mergers cannot reliably capture the physics of the most extreme stages of the merger, including the merger itself and the environment immediately following, where the matter is in a highly dynamical state and the spacetime curvature is the strongest. For a reliable, first-principles understanding of these stages of BNS mergers, the use of NR is the only recourse. Numerical simulations allow for the systematic isolation of different physical phenomena/physics, which provides a powerful tool for deducing the role of magnetic fields, e.g., [7–14], equation of state (EOS) effects, e.g., [15–24], and neutrino transport, e.g., [18,20,25,26].

There are many important physical effects to consider in simulations of BNS mergers (see [27–32] for reviews); such as finite temperature EOSs and magnetic fields. Regarding the use of realistic EOSs in BNS merger simulations, an important aspect is the consistent evolution of the electron fraction, as it provides crucial information for the description of nucleosynthesis processes that take part during and after merger, and is necessary to fully describe kilonova (KN) afterglows [33,34]. Additionally, the thermodynamic properties of the postmerger remnant and ejecta are important and not well-approximated by cold EOSs (see, e.g., [35,36]). Magnetic fields are also a crucial ingredient to consider in BNS merger simulations. In particular, an area which remains poorly understood is the effect of magnetic fields on ejecta properties. Large scale magnetic fields are important in the evolution of the postmerger remnant and in the kilonova signal associated with BNS mergers, as they are expected to power relativistic jets [13,37–40] and drive relativistic outflow [41–43]. The interplay between finite temperature EOSs and strong magnetic fields remains insufficiently characterized in BNS merger simulations. For example, realistic EOSs have been shown to produce lower-velocity ejecta when compared to simple analytic EOSs [18,23,31,44,45] while, on the other hand, simulations which account for magnetic effects suggest that, if magnetic fields are strong enough, shocked dynamical ejecta during merger could be boosted to higher velocities [10–14,46]. Most modern simulations of BNS mergers consider only some of the aforementioned physical effects, and there have only been a handful of simulations which consider all effects at once [10–14,47]. Even in the case of more complete simulations, all treatments of neutrino transport to-date must make some level of approximation since solving the full $(6+1)$-dimensional neutrino transport equation is not possible with current schemes and computational resources.

There are currently available several codes with support to different levels of physics. For instance, many state-of-the-art codes with microphysical EOS compatibility do not implement a constrained-transport treatment of magnetic fields. While variety in the numerical methods and codes used in the study of BNS mergers is a strong point from the perspective of understanding numerical systematic errors, key differences between codes can lead to a range of predictions for the relevant observables, depending on the physics included in simulations. For example, although there is general agreement in the ejecta properties predicted by different codes [17–19,23,25,45,48–50], dynamical ejecta masses range over $10^{-4}M_\odot \lesssim M_{\rm ej} \lesssim 10^{-2}M_\odot$ while speeds range over $0.1c \lesssim v_{\rm ej} \lesssim 0.3c$, depending on the total binary mass, EOS, and numerical methods considered. Simulations with even the highest resolutions can have numerical uncertainties of over 40% [48,51–53]. The need for accurate simulations and detailed estimates of the error budgets is becoming more and more urgent as future GW detectors are expected to be more sensitive than current systematic errors [54]. Future investigations of BNS mergers would ideally require a set of catch-all, accurate, open source general relativistic (magneto)hydrodynamic [GR(M)HD] codes which can reliably simulate BNS systems while considering as many microphysical phenomena as possible. At the very least, reliable future predictions using NR codes require detailed comparisons, including cross-checks of the results obtained using different codes and of the varied numerical methods used within the codes themselves.

In order to begin addressing each of these needs, we first present a new implementation of realistic, finite-temperature EOS support in IllinoisGRMHD[1] (as extensively discussed below), and then perform a systematic comparison between different open-source GR(M)HD codes: IllinoisGRMHD [56], GRHydro [57], Spritz [58], and WhiskyTHC [59–61]. These codes implement the same physics and have largely overlapping numerical methods, e.g., they are all housed within the Einstein Toolkit [62,63], thereby we can isolate and compare the impact on the evolution and uncover the existence of systematic errors arising only from the GR(M)HD codes. This allows us to test the robustness of certain results with respect to different computational choices. In most cases, such choices are about failsafes for when the main algorithms fail, as in the case of the conservative-to-primitive schemes used in the artificial atmosphere. In our comparison, we focused on BNS mergers and found that important quantities disagree across the codes. For example, codes do not agree on whether the remnant is stable or undergoes collapse very shortly after merger, which has profound implications for EM observations. This calls for more detailed studies and comparisons across different codes.

This paper is split in two parts. First, we discuss in detail the formalism employed by IllinoisGRMHD, the extensions required to reach feature-parity with the other codes, and the tests we performed. In particular, in Sec. II we review the basic equations that are relevant to BNS systems, and provide detail on their numerical treatment within IllinoisGRMHD. In Sec. III we describe our methods for implementing realistic, finite temperature EOS capability within IllinoisGRMHD. In Sec. IV we discuss stringent dynamical tests of the extended version of IllinoisGRMHD which cover a broad range of scenarios. Readers that are familiar with GRMHD simulations may skip this part and focus on the results. In the second part, starting from Sec. IV E, we discuss the results from our simulations of BNS mergers.

Our extension to IllinoisGRMHD to allow for the use of realistic, finite-temperature EOSs is public. The extensions we have made to IllinoisGRMHD are crucial for understanding the interplay between magnetic fields and the EOS in the postmerger environment, and is a first step to including neutrino transport schemes in the code. The current state of IllinoisGRMHD (including the extensions we describe here) opens up many avenues of investigation surrounding compact object mergers with strong magnetic fields. For instance, the new code capabilities will make possible the calculation of nucleosynthesis rates and help elucidate the role of microphysical, finite temperature EOSs in BNS mergers with strong magnetic fields, among many other interesting phenomena.

Throughout the work we use geometrized units, where $G = c = 1$, unless otherwise stated. In addition, in cases where we use logarithmic scales, we assume that $\log \equiv \log_{10}$, unless otherwise noted. All visualizations and postprocess analyses in this work were carried out with the KUIBIT software package [64].

II. BASIC EQUATIONS

Throughout this work we will predominantly work with numerical codes that solve the Einstein field equations,

$$G_{\mu\nu} = 8\pi T_{\mu\nu}, \tag{1}$$

(where $G^{\mu\nu}$ and $T^{\mu\nu}$ are the Einstein and stress-energy tensors, respectively), coupled to the equations of ideal relativistic (magneto)hydrodynamics. In particular, we focus on the use of IllinoisGRMHD to solve the equations of ideal relativistic magnetohydrodynamics. IllinoisGRMHD takes advantage of the Baumgarte-Shapiro-Shibata-Nakamura (BSSN) formulation [65–67] of the $3 + 1$ Arnowitt-Deser-Misner (ADM) formalism, which recasts the Einstein equations in the form of an initial-value problem (for more details, see textbooks on the subject, e.g. [68–71]), in which the spacetime line element is given by

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

where α is the lapse, β^i is the shift vector, $\gamma_{\mu\nu} = g_{\mu\nu} + n_\mu n_\nu$ represents the induced metric on spacelike hypersurfaces,

---

[1] Recently, [55] implemented a similar extension to IllinoisGRMHD.

and $n^\mu = (1/\alpha, -\beta^i/\alpha)$ is the future-pointing unit vector orthogonal to each spatial slice. The solution to Eq. (1) involves the evolution of the magnetohydrodynamic variables that appear on the right-hand sides of the ADM equations. An approach which is well suited to the $3+1$ formalism is the Eulerian (or Valencia) formulation of relativistic hydrodynamics [72,73]. In this formulation, the evolution equations for the relevant fluid variables arise from several conservation laws, including the continuity equation

$$\nabla_\mu(\rho_\mathrm{b} u^\mu) = 0, \tag{3}$$

where ρ_b is the rest mass density and u^μ is the fluid four-velocity, the lepton number conservation (when neutrino effects are ignored)

$$u^\mu \nabla_\mu (Y_\mathrm{e}) = 0, \tag{4}$$

which can be rewritten by use of Eq. (3) as

$$\nabla_\mu(\rho_b Y_\mathrm{e} u^\mu) = 0, \tag{5}$$

where $Y_\mathrm{e} \equiv n_\mathrm{e}/n_\mathrm{b}$ is the electron fraction and n_b (n_e) is the baryon (electron) number density, the conservation of stress energy

$$\nabla_\mu T^{\mu\nu} = 0, \tag{6}$$

and the homogeneous Maxwell's equations

$$\nabla_\nu F^{*\mu\nu} = 0, \tag{7}$$

where $F^{*\mu\nu} = \frac{1}{2}\varepsilon^{\mu\nu\alpha\beta}F_{\alpha\beta}$ is the dual to the electromagnetic field strength tensor $F_{\alpha\beta}$ and $\varepsilon^{\mu\nu\alpha\beta}$ is the rank-4 Levi-Civita symbol. The matter variables are evolved using Eqs. (3)–(7) once they have been cast in flux-conservative form,

$$\partial_t \mathbf{C} + \boldsymbol{\nabla}\cdot\mathbf{F} = \mathbf{S}, \tag{8}$$

where $\mathbf{C}$, $\mathbf{F}$, and $\mathbf{S}$ are vectors which contain the conservative, flux, and source terms, respectively. The vector of primitive variables $\mathbf{P}$

$$\mathbf{P} = \begin{bmatrix} \rho_\mathrm{b} \\ P \\ v^i \\ B^i \\ Y_\mathrm{e} \end{bmatrix}, \tag{9}$$

contains information on the physical state of the fluid, where P is the fluid pressure, $v^i = u^i/u^0$ is the fluid three-velocity and B^i are the spatial components of the magnetic field as measured by a normal observer. The conservative variables $\mathbf{C}$ are determined in terms of the primitive variables, the lapse α, and the metric as

$$\mathbf{C} = \begin{bmatrix} \rho_* \\ \tilde{\tau} \\ \tilde{S}_i \\ \tilde{B}^i \\ \tilde{Y}_e \end{bmatrix} = \begin{bmatrix} \alpha\sqrt{\gamma}\rho_\mathrm{b}u^0 \\ \alpha^2\sqrt{\gamma}T^{00} - \rho_* \\ (\rho_* h + \alpha u^0 b^2)u_i - \alpha\sqrt{\gamma}b^0 b_i \\ \sqrt{\gamma}B^i \\ \rho_* Y_\mathrm{e} \end{bmatrix}, \tag{10}$$

where $b^\mu = B^\mu_{(u)}/\sqrt{4\pi}$ [where $B^\mu_{(u)}$ is the magnetic field as measured by an observer in the fluid rest frame], γ is the determinant of the 3-metric, and $h = 1 + \epsilon + P/\rho_0$ is the specific enthalpy (where ϵ is the specific internal energy). We note that IllinoisGRMHD uses the coordinate three-velocity, $v^i \equiv u^i/u^0$, unlike many other evolution codes, which also adopt the Valencia formalism, but use the Eulerian three-velocity $v^i_{(n)} = u^i/(\alpha u^0) + \beta^i/\alpha$ [74,75], i.e., the velocity measured by normal observers. Additionally, we note that IllinoisGRMHD works with an ideal fluid stress-energy tensor of the form

$$T_{\mu\nu} = (\rho_\mathrm{b} h + b^2)u_\mu u_\nu + (P + b^2/2)g_{\mu\nu} - b_\mu b_\nu. \tag{11}$$

The evolution equations for the relevant fluid variables are determined by using the aforementioned conservation laws and casting them in the form of Eq. (8). It is useful to pay special attention to the evolution of the magnetic field, due to how it is treated in IllinoisGRMHD. Specifically, the evolution equation for the magnetic field is

$$\partial_t \tilde{B}^i + \partial_j(v^j\tilde{B}^i - v^i\tilde{B}^j) = 0, \tag{12}$$

where $\tilde{B}^i$ is the conservative variable corresponding to the magnetic field B^i, as defined in Eq. (10). The use of Eq. (12) to evolve the magnetic field results in terms that violate the no-monopole constraint ($\boldsymbol{\nabla}\cdot\mathbf{B} = 0$), which is addressed in IllinoisGRMHD by instead considering the evolution of the vector potential $\mathcal{A}_\mu = \Phi n_\mu + A_\mu$ (where A_i is the magnetic vector potential and Φ is electric scalar potential) and recovering the magnetic field as $\tilde{B}^i = \epsilon^{ijk}\partial_j A_k$. The evolution equation for A_i is

$$\partial_t A_i = \epsilon_{ijk} v^j \tilde{B}^k - \partial_i(\alpha\Phi - \beta^j A_j). \tag{13}$$

IllinoisGRMHD works in the generalized Lorenz gauge $\nabla_\mu \mathcal{A}^\mu = \xi n_\mu \mathcal{A}^\mu$ [76], where ξ is chosen such that the Courant-Friedrich-Lewy condition corresponding to it is satisfied at all times given the grid choices. In the following we provide additional descriptions of the algorithms employed within IllinoisGRMHD for the implementation of finite temperature EOS compatibility. We direct the reader to [56] for a detailed description of all of the additional algorithms employed within IllinoisGRMHD.

III. IMPLEMENTATION OF REALISTIC EQUATION OF STATE COMPATIBILITY WITHIN IllinoisGRMHD

The current open-source version of IllinoisGRMHD solves the equations of GRMHD by assuming simple, analytic EOSs such as polytropic EOSs; in these cases the thermal effects are typically approximated using a Γ-law EOS. With this approach, the pressure of the fluid is $P = P_{\rm cold} + (\Gamma_{\rm th} - 1)\rho_{\rm b}\epsilon_{\rm th}$, where $P_{\rm cold}$ is any analytic representation of the cold EOS and $\epsilon_{\rm th}$ is the thermal-specific energy density. These EOSs can only provide a qualitative understanding of the state of matter during a BNS merger [24,35]. However, efforts to model parametrically both the thermal and the cold component of the nuclear EOS in BNS mergers are under way, see, e.g., [24,77–80].

The implementation of realistic EOSs within IllinoisGRMHD allows us to understand, in a more detailed manner, the interplay between the EOS, thermal effects, and magnetic fields in these systems. Moreover, it is a crucial first step toward implementing additional important microphysics within IllinoisGRMHD, such as neutrino transport. The inclusion of realistic EOS capability within IllinoisGRMHD required two steps: the implementation of the evolution equation for the electron fraction and the implementation of algorithms which can perform the nontrivial inversion of the evolved conservative variables **C** to the physical primitive variables **P**, which we refer to as conservative-to-primitive inversion. In the following we discuss the implementation of the evolution equation for the electron fraction $Y_{\rm e}$ within our extended version of IllinoisGRMHD. We also discuss the implementation of state-of-the-art routines within IllinoisGRMHD for conservative-to-primitive inversion and present relevant tests. For the remainder of this work, we refer to the currently available version of IllinoisGRMHD as `OriginalIllinoisGRMHD` (abbreviated as `OIL`). Our extended version of IllinoisGRMHD will be referred to as `MicrophysicalIllinoisGRMHD` (abbreviated as `MIL`). In cases where we discuss features which are common between the two codes, we will refer to them jointly as the IllinoisGRMHD code. Algorithmically, `OIL` and `MIL` are identical, except for the changes highlighted in this work which are required for realistic EOS compatibility.

A. Evolution of the electron fraction

In flux-conservative form, the electron fraction evolution equation is

$$\partial_t(\tilde{Y}_{\rm e}) + \partial_j(v^j\tilde{Y}_{\rm e}) = 0. \tag{14}$$

With the inclusion of Eq. (14), the full set of equations solved within `MIL` is

$$\partial_t\begin{bmatrix}\rho_* \\ \tilde{Y}_{\rm e} \\ \tilde{\tau} \\ \tilde{S}_i \\ A_i\end{bmatrix} + \partial_j\begin{bmatrix}\rho_* v^j \\ \tilde{Y}_{\rm e}v^j \\ \alpha^2\sqrt{\gamma}T^{0j} - \rho_* v^j \\ \alpha\sqrt{\gamma}T^j_i \\ \alpha\Phi - \beta^j A_j\end{bmatrix} = \begin{bmatrix}0 \\ 0 \\ s \\ \frac{1}{2}\alpha\sqrt{\gamma}T^{\alpha\beta}g_{\alpha\beta,i} \\ \epsilon_{ijk}v^j\tilde{B}^k\end{bmatrix}, \tag{15}$$

where

$$s = \alpha\sqrt{\gamma}[(T^{00}\beta^i\beta^j + 2T^{0i}\beta^j + T^{ij})K_{ij} - (T^{00}\beta^i + T^{0i})\partial_i\alpha], \tag{16}$$

and K_{ij} is the extrinsic curvature. The evolution of $\tilde{Y}_{\rm e}$ follows that of the other conservative variables, which begins with the determination of initial conditions. Presently, we allow for $Y_{\rm e}$ to be initialized in two possible ways:

1. Linear profile: we set $Y_{\rm e} = \Upsilon\rho_{\rm b}$ (where Υ is a constant that ensures proper dimensionality), such that the electron fraction profile is linear with respect to $\rho_{\rm b}$, in order to consider a simple profile where gradients are nonzero throughout the solution grid. This initialization is unphysical and only useful for testing the advection of $Y_{\rm e}$ in situations without realistic EOSs, i.e., passive advection of the electron fraction.
2. β-equilibrium profile: we set $Y_{\rm e}(\rho_{\rm b})$ according to the conditions for β-equilibrium in cold neutron star (NS) matter. This initial profile is suitable for realistic descriptions of isolated stars as well as for binaries that are initially separated at large enough distances such that the components are cold. All of the BNS initial data considered in our tests are built for quasiequilibrium systems, in which the assumption that the components are cold, β-equilibrated stars is well justified. We use the same assumption of cold β-equilibrium to initialize all other hydrodynamical variables at $t = 0$.

Once the initial data are specified for the primitive variables at all grid points, the conservative variables are obtained through the simple algebraic relations provided in Eq. (10), which provides **C** at all grid points. For the evolution of **C**, we employ three ghost-zones at the outer boundary of each adaptive mesh refinement (AMR) grid. We fill all buffer zones at the refinement level boundary with data interpolated from neighboring rougher or finer levels of refinement using standard prolongation or restriction operators, respectively.

Filling in the buffer zones in this manner results in **C** being either prolongated or restricted after calculation [56]. To ensure consistency between **C** and **P**, the primitives are recovered from the conservatives using a root-finding algorithm at $t = 0$, which we discuss in Sec. III B. Next, we evaluate the flux term **F** in preparation for the next time step. To this end, we must reconstruct the primitives between grid points (i.e., at cell interfaces). IllinoisGRMHD employs the piecewise parabolic method (PPM) [81] for primitives reconstruction. Reconstruction is used to evaluate the primitives on the left and right interfaces of all grid points, $\mathbf{P}_{L,R}$, in all directions. These interface values are then used to calculate the corresponding conservative variables at cell interfaces, which are in turn used to calculate the flux term in Eq. (14) using a second-order, finite-volume, high-resolution shock-capturing scheme. The handling of fluxes at grid interfaces $\mathbf{F}_{L,R}$ requires a solution to a Riemann problem. IllinoisGRMHD employs the standard Harten-Lax-vanLeer (HLL) [82] approximate Riemann solver, where for a given direction the electron fraction flux is given by

$$F^{\mathrm{HLL}}(Y_{\mathrm{e}}) = \frac{c^{-}F_{R} + c^{+}F_{L} - c^{+}c^{-}(\tilde{Y}_{\mathrm{e},R} - \tilde{Y}_{\mathrm{e},L})}{c^{+} + c^{-}}, \quad (17)$$

where $c^{\pm} = \pm \max(0, c^{R}_{\pm}, c^{L}_{\pm})$ and $c^{L,R}_{\pm}$ are the maximum $(+)$ and minimum $(-)$ characteristic speeds at the left (L) and right (R) cell interfaces (see [56,83] for further algorithmic details). The derivatives of the fluxes are then determined and summed independently for each direction. For instance, the flux along the x-direction takes the form

$$(\partial_{x}F^{x})_{ijk} = \frac{F^{HLL,x}_{i+\frac{1}{2}jk}(Y_{\mathrm{e}}) - F^{HLL,x}_{i-\frac{1}{2}jk}(Y_{\mathrm{e}})}{\Delta x}. \quad (18)$$

The flux along the y- and z-directions take a similar form, but we instead consider finite differencing along the j and k indices, respectively.

The evolution equation for Y_{e} does not include source terms in the absence of neutrinos, so the right-hand side of Eq. (18) is then passed to the method of lines (`MoL`) thorn, which integrates the conservative variable $\tilde{Y}_{\mathrm{e}}$ forward in time. At this point, the updated conservative variables would be known at all grid points except the outer boundary. The next step is to recover the primitive variables given these evolved conservative variables (see Sec. III B). After the primitives have been recovered, they are checked for physicality and marginally modified if they are outside of their physical ranges [84]. For example, in the case of electron fraction, we check that

$$Y_{\mathrm{e,lower}} \leq Y_{\mathrm{e}} \leq Y_{\mathrm{e,upper}}, \quad (19)$$

where $Y_{\mathrm{e,lower}}$ ($Y_{\mathrm{e,upper}}$) corresponds to the lowest (highest) value for Y_{e} available in an EOS table. Next, outer boundary conditions are placed on the recovered primitives to fill the necessary three ghost zones in each direction. We apply zero-derivative outflow outer boundary conditions as described in [56]. Up to this point, the primitives **P** are known at the new time step on all grid points. The final step we take is to recompute the conservatives on all grid points using Eq. (10) for consistency between **P** and **C**, and the evolution algorithm is allowed to proceed.

B. Conservative-to-primitive solvers

At every step of the evolution an inversion from the evolved conservative variables **C** to the physical primitive variables **P** is required to know the state of the fluid. Eq. (10) presents a system of nonlinear algebraic expressions which can be solved for nine relevant fluid variables (ρ_{b}, v^{i}, B^{i}, Y_{e}, and either h or ϵ). These nine variables, along with an EOS, in turn provide all of the information required to determine **P**, along with other fluid variables of interest. For example, a solution to Eq. (10) can provide the five main variables (ρ_{0}, v_{i}, ϵ) and, trivially, B_{i} and Y_{e}. We can then determine the remaining variable, P with the use of an EOS and incidentally obtain information on other physical variables such as the temperature T and specific entropy s_{b}. As we require a solution for five *main* variables in order to determine **P**, the primitives inversion problem is fundamentally a nontrivial 5D problem that cannot be solved analytically. 5D schemes which solve Eq. (10) were originally implemented in early MHD codes such as HARM [85]. However, these schemes were eventually found to be inefficient and inaccurate, which led to the development of methods which solve for only two auxiliary variables [86] and thereby reduce the dimensionality of the problem to 2D. The dimensionality of the problem can be further reduced to 1D; modern 1D algorithms which provide reliable and efficient solutions have been developed [17,87] and are widely used in GR(M)HD codes.

In order to consider strongly magnetized systems which include realistic descriptions of the dense matter EOS, we implement state-of-the-art conservative-to-primitive solvers within `MIL`. Our implementation includes porting the solvers discussed in [88] for use in the Einstein Toolkit, packaged within a new thorn `ConservativeToPrimitive` which can interface with `MIL` and its associated thorns, but also works as a standalone thorn which can be used with other GR(M) HD codes that operate within the Cactus infrastructure. We focus on a subset of the solvers implemented in [88], due to their reliability, speed, and algorithmic similarity to the original solvers used in `OIL` (see [89] for another possible approach). In particular we focus on the 2D solver of [86] (which we label `Noble`), the 1D solver of [17] (which we label `Palenzuela`), and the 1D solver of [87] (which we label `Newman`). We note that the `Noble` solver uses the same algorithm as the solvers in the original version of IllinoisGRMHD and that other GRMHD codes with realistic EOS capability rely on the `Palenzuela`

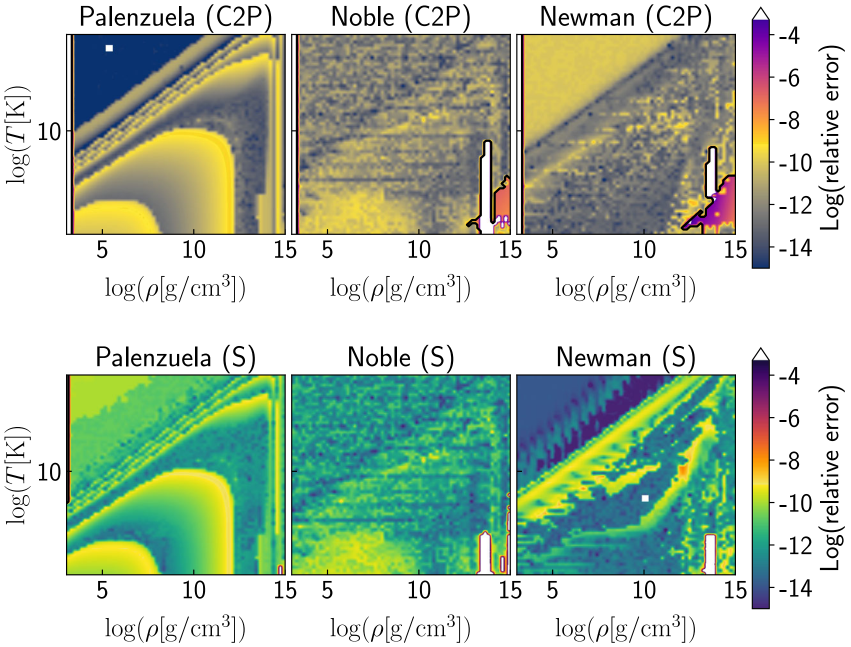

FIG. 1. Top panel: Relative error in the recovery of primitives for a selection of the solvers originally implemented in [88] and ported to the Einstein Toolkit within the `ConservativeToPrimitive` thorn (labeled C2P). We test the LS220 EOS [92,93] with three of the available solvers. For all tests, we fix a subset of the primitives and provide random initial data for the remainder. Specifically, we fix $P_{\mathrm{mag}}/P = 10^{-3}$, $W = 2$, and $Y_{\mathrm{e}} = 0.1$ while scanning the physically allowed range for T and ρ_{b}. Bottom panel: Same as the top panel, but using the original implementation of each solver in the code packaged with [88] (labeled S).

algorithm [58,90,91]. We refer the reader to [88] for a review of the algorithms used in each solver within `ConservativeToPrimitive`.[2]

We employ a set of preliminary tests to confirm that our port of these solvers to the Einstein Toolkit behaves as intended. In particular, we test the aforementioned solvers using the same tests as [88] where a set of primitives **P** are initialized, randomly perturbed, used to calculate a set of conservatives **C**, and finally recovered into a new set **P**'. The primitives recovery for each solver is then assessed by considering the relative error between the original set **P** and the recovered set $\mathbf{P}'$ (along with other diagnostics including the number of interpolation calls to the EOS table and number of algorithm iterations). In these tests, a subset of the primitives is varied over the physically allowed range while holding others constant. We focus on the case of the LS220 realistic, finite temperature, tabulated EOS [92,93] and employ tests where we prescribe the ratio of magnetic to fluid pressure $P_{\mathrm{mag}}/P = b^2/(2P) = 0.001$, the Lorentz factor $W = 2$, and the electron fraction $Y_{\mathrm{e}} = 0.1$, while scanning over the allowed range of rest mass density ρ_{b} and temperature T for this EOS. For each value of ρ_{b} and T that we consider, we determine the relative error between the original and recovered set of primitives. We present the results for this set of tests in the top panel of Fig. 1. As a comparison, we also show the results of these tests using the implementation of the solvers within the code of [88] in the bottom panel of Fig. 1.

For all tests, we find very good agreement between the original solvers and our implementation of them within the Einstein Toolkit. Points on Fig. 1 which appear in white indicate the failure of a given solver to recover a consistent set of primitives. Points at which the solvers fail which are surrounded by successful recovery are usually due to the random perturbations we induce on **P** before recovery, which occurs for both the original code of [88] and our implementation within `ConservativeToPrimitive` (see the top-left and lower-right panels of Fig. 1). However, there are regions of variable space where solvers consistently fail regardless of the initial perturbations (in both our implementation of the solvers within

---

[2] We point out that [88] appears to have typographical errors in the algorithm description for the `Newman` solver, when compared to the original paper of [87]. In particular, there is a difference in the calculation of the auxiliary variable $\mathcal{M}^2$ [see Eq. (47) of [88] as compared to Eq. (4.7) of [87]], where [87] correctly calculates it as $\mathcal{M}^2 = m_i m^i = \tilde{S}_i \tilde{S}^i$, where $\tilde{S}_i$ is the conservative variable associated with the momentum density which appears in the left-hand side of Eq. (15). Equation (47) of [88] inconsistently calculates this variable as $\mathcal{M}^2 = (B^i v_i)^2/\sqrt{\gamma}$, where γ is the determinant of the 3-metric γ_{ij}, B^i is the magnetic field, and v_i is the fluid three-velocity. We also note that the first term of Eq. (55) in [88] misses a factor of the auxiliary variable a, when compared to the analogous Eq. (5.11) in [87]. We point out that despite these typographical errors, the numerical implementation of the `Newman` solver within the code of [88] is consistent with the correct algorithmic steps presented in [87].

`ConservativeToPrimitive` and the original implementation in [88]). The most reliable solver across variable space is the `Newman` solver although, as indicated by the rightmost panels of Fig. 1, it may fail for cold matter ($T \lesssim 10$ MeV) in the high-density regime $\rho_\mathrm{b} \approx 10^{14}\,\mathrm{g\,cm^{-3}}$, which is a highly relevant area of variable space for BNS mergers. It is important to note that regions where a given solver may fail are often well covered by at least one of the other solvers. Therefore, in practice the best use of these solvers should employ a hybrid approach, where a solver which fails in a given region of variable space is substituted by a solver which provides reliable solutions in that same region. In the following we discuss the choice of solvers we employ moving forward.

C. Solution algorithm

The preliminary tests presented in Fig. 1 suggest that some solvers are better suited for systems involving neutron stars with realistic EOSs over others. Nevertheless, we note that the use of *any* solver is still likely to produce robust solutions. Therefore, the chief property that we use in determining the primary solver to utilize is the performance speed and agreement with the results of the `OIL` code. For realistic EOSs, the bottleneck in limiting solver speed is typically the number of times the solver must interpolate the EOS. Among the solvers in `ConservativeToPrimitive`, the `Noble` solver requires the fewest such EOS calls, resulting in the most efficient solutions [88]. Nevertheless, the `Palenzuela` and `Newman` solvers require a similar number of EOS calls to `Noble`, and the `Palenzuela` solver has the added benefit of strong agreement with `OIL` and of reliable coverage across densities and temperatures, as suggested by the leftmost panels of Fig. 1. Due to the reliability and agreement with `OIL`, and because the performance is not much worse than other solvers, we consider the `Palenzuela` solver as the optimal choice for our primary solver. In the remainder of this work, we adopt the following solution algorithm for conservative-to-primitive inversion in `MIL`

1. We employ the `Palenzuela` solver in our first attempt at a solution. We find that the `Palenzuela` solver is robust for both analytic and microphysical EOSs and comparably efficient to other solvers.
2. If the `Palenzuela` solver fails, as it is disposed to for highly relativistic matter, we try to recover a solution with either the `Newman` or `Noble` solvers. The `Noble` solver is robust for highly magnetized flows and highly efficient, but begins to fail at high Lorentz factors. The `Newman` solver is reliable for both highly magnetized and highly relativistic flows.
3. If all of the aforementioned solvers fail we impose a fix of the velocities, as done in the `OIL` code [56], which assumes a cold EOS ($T = 0.1$ MeV); we refer to this fix as the "Font" fix, which has been generalized from its original form in `OIL` for generic EOSs.
4. Finally, if all other recovery attempts have failed, it is likely that the fluid element is at low densities, so we set the recovered primitives to values which are sensible for atmospheric conditions [cold temperature $T_\mathrm{atm} = 0.1$ MeV, low density $\rho_{b,\mathrm{atm}} \leq 10^{-7}\rho_{b,\mathrm{max}}$, fixed electron fraction $Y_{e,\mathrm{atm}} = 0.25$, and a pressure which is consistent with the EOS $P_\mathrm{atm} = P(\rho_{b,\mathrm{atm}}, T_\mathrm{atm}, Y_{e,\mathrm{atm}})$]. We note that the other open-source codes we consider, with the exception of GRHydro which uses the same treatment as `MIL`, only fix atmospheric values of ρ_b and T, but not Y_e. We emphasize, however, that the values of the primitive variables in the atmosphere are generally dynamically unimportant, and are set for numerical stability.

We find that the above solution algorithm leads to a robust and efficient evolution across the physically allowed range of variables for a finite temperature, realistic EOS. We note that the order of solvers used is completely customizable at run time and can be changed dynamically for any simulation, and that any ordering of the `Palenzuela`, `Noble`, and `Newman` solvers results in a reliable solution. The order of solvers chosen here is optimal when considering efficiency and reliability for finite temperature, tabulated EOSs. In the remainder of this work, we utilize the `MIL` code with the above solution algorithm, unless otherwise noted.

IV. DYNAMICAL TESTS OF MIL AND CODE COMPARISONS

A. Summary of tests, initial data, and grid hierarchies

In this section we focus on code comparisons between `MIL` and other open-source codes GR(M)HD codes. As tests of `MIL`, we employ a number of evolutions for which the system behavior is well understood. Where relevant, we compare the results of the `MIL` code to other publicly available codes with similar capabilities. In particular we compare with the GRHydro [57], Spritz [58], WhiskyTHC [59–61], and `OIL` [56] codes.

We present the results of these tests as follows:

1. G2TOV: We first discuss tests which employ a Tolman-Oppenheimer-Volkoff (TOV) star built with a $\Gamma = 2$ polytrope, which we label test G2TOV. For test G2TOV we compare the solution to results obtained with the GRHydro code. This test was designed to mirror those considered in [56].
2. LSTOV: Next, we discuss tests where we consider a magnetized TOV star constructed using the finite temperature, tabulated LS220 EOS [92,93], which we label test LSTOV. For this test we compare our solutions to those obtained with the publicly available Spritz code, which uses an implementation of the `Palenzuela` solver for finite temperature EOSs.

(3) G2BNS: Next, we test our code in the case of a BNS merger, assuming a $\Gamma = 2$ polytropic EOS. We compare the solution against the results of the GRHydro, Spritz, WhiskyTHC, and `OIL` codes. We consider both magnetized and unmagnetized systems, and refer to this test as G2BNS.

(4) LSBNS: Finally, we consider a BNS built using the LS220 EOS. Along with the `MIL` code, we simulate this system with the GRHydro, and WhiskyTHC codes. The binary configuration is built such that there is a transient hypermassive neutron star (HMNS) post-merger remnant, which allows us to compare properties of the merger remnant between codes. We also consider evolution with the Spritz code, but we do not include those results, because the public version of the code is still under development for cases with tabulated EOS support in the context of binary neutron star mergers. We label this test as LSBNS.

The use of `MIL` and `ConservativeToPrimitive` requires the initialization of three new required variables when evolving finite temperature, realistic EOSs. Namely, we must initialize the specific internal energy ϵ, temperature T, and electron fraction Y_e such that they are consistent with the initial data. For tests which employ a cold polytropic EOS we initialize the specific internal energy as

$$\epsilon = \frac{P}{(\Gamma - 1)\rho_0}. \tag{20}$$

For polytropic EOSs, the initialization of T and Y_e are independent to that of ϵ and do not play a role in the fluid evolution, so we simply initialize Y_e based on the first type of profile listed in Sec. III A and fix $T = 0.1$ MeV, although the temperature is a passive variable in such tests. For tests which employ realistic, finite temperature EOSs, we are interested in equilibrium initial data, which requires cold, uniform temperatures and β-equilibrium. To construct cold, β-equilibrium initial data, we build barotropic tables that provide $P(\rho_\mathrm{b})$. Due to the fact that `ConservativeToPrimitive` makes use of EOS tables in the format of the `StellarCollapse` repository [92] via the EOS driver thorn `EOS_Omni`, we construct cold, β-equilibrium tables required for initial data using the `StellarCollapse` tables. The `StellarCollapse` tables provide several fluid variables (including the pressure P, specific internal energy ϵ, and constituent chemical potentials μ_i, where $i = p, n, e$ is an index over the particle species for protons, neutrons, and electrons, respectively) at discrete values of the rest mass density ρ_b, electron fraction Y_e, and temperature T. Each of the fluid variables may be interpolated to arbitrary values of the triplet $(\rho_\mathrm{b}, Y_\mathrm{e}, T)$. To extract the cold, β-equilibrated, barotropic functions required for initial data ($P(\rho_\mathrm{b})$, and $\epsilon(\rho_\mathrm{b})$), we take the following steps:

(1) We fix the value of $T = T_\mathrm{cold} = 0.1$ MeV. The lowest possible temperature values in these tables are typically $T = 0.01 - 0.05$ MeV, but entries near the table limits are not finely sampled, which often leads to interpolation errors in the primitives recovery process. As such, we employ a finite temperature which is still cold compared to the Fermi energy but avoids the table boundaries.

(2) For a given value of $\rho_\mathrm{b} = \rho_{b,\mathrm{curr}}$, we scan the available range of Y_e and evaluate the chemical potentials relevant for beta equilibrium [$\mu_i(Y_\mathrm{e}, \rho_{b,\mathrm{curr}}, T_\mathrm{cold})$].

(3) We employ root-finding to locate the value of Y_e which corresponds to β-equilibrium, such that

$$\mu_n - \mu_p - \mu_e = 0. \tag{21}$$

At this value of $Y_\mathrm{e} = Y_{\mathrm{e},\beta}$, we record the pressure and specific internal energy. We continue the algorithm from step 2 above, scanning the available range of ρ_b and thereby constructing tabulated functions for the barotropic pressure P and specific internal energy ϵ as functions of the rest mass density ρ_b for β-equilibrium matter.

In Table I we present the grid hierarchy corresponding to each dynamical test we consider. For each case we list the test name, the finest-level grid resolution dx_fin, the number of refinement levels (RL), the number of grid-points per NS radius (GP), the half-side length of each nested grid (HSL) listed from finest to coarsest, and the evolution codes used. In all cases we evolve the spacetime using the `McLachlan` spacetime evolution code within the Einstein Toolkit (via the `ML_BSSN` thorn) [94,95], unless otherwise noted. `ML_BSSN` solves the Einstein equations within the BSSN formulation of the ADM $3+1$ formalism. Our gauge conditions consist of the "$1 + \log$" slicing condition for the lapse [96] and the "gamma-driver" condition for the shift [97]. Integration in time is carried out using a fourth-order accurate Runge-Kutta (RK4) scheme, using the `MoL` thorn, with a Courant factor of 0.5, unless otherwise stated.

B. Diagnostics

We employ several diagnostics to assess the quality of our simulations. In cases where several resolution simulations are considered and some solution is known, we monitor convergence of the solution by calculating the convergence order n_con as

$$n_\mathrm{con} = \frac{\log\left(\frac{L_i - L_0}{L_j - L_0}\right)}{\log\left(\Delta x_i / \Delta x_j\right)}, \tag{22}$$

where L_0 is the value that the solution should approach in the continuum limit, i is a label for a grid of a given

TABLE I. Grid hierarchy and simulation settings for the tests presented in Sec. IV, in the case of the lowest-resolution grids employed. For each case we list the test name, the finest-level grid resolution dx_{fin}, the number of refinement levels (RL), the number of grid points per NS radius (GP), and the half-side length of each nested grid (HSL) listed from finest to coarsest. We also list the evolution codes (Evol. code), Riemann solver (Riemann sol.), reconstruction scheme (Recon. scheme), and conservative-to-primitive error tolerance (C-to-P tol.) used in each simulation. In terms of the Riemann solver, we use either the Harten-Lax-vanLeer (HLL) [82] or Harten-Lax-vanLeer-Einfeldt (HLLE) [98], depending on the algorithm available in each code. In all cases we use the piecewise parabolic method (PPM) for primitives reconstruction, except for when we employ the WhiskyTHC code in which case use MP5.

Test	dx_{fin}	RL	GP	HSL ($M_\odot$)	Evol. code	Riemann sol.	Recon. scheme	C-to-P tol.
G2TOV	0.25	1	32	(10.0)	`MIL`	HLL	PPM	10^{-8}
					GRHydro	HLLE	PPM	10^{-8}
LSTOV	0.111	3	64	(8.9, 17.8, 35.6)	`MIL`	HLL	PPM	10^{-8}
					Spritz	HLLE	PPM	10^{-8}
G2BNS	0.155	7	40	(13.0, 17.9, 26.1, 60.0, 120.0, 240.0, 396.0)	`OIL`	HLL	PPM	10^{-8}
					`MIL`	HLL	PPM	10^{-8}
					Spritz	HLLE	PPM	10^{-8}
					GRHydro	HLLE	PPM	10^{-8}
					WhiskyTHC	HLLE	MP5	10^{-8}
LSBNS	0.14	7	50	(8.7, 17.4, 34.8, 69.6, 139.2, 278.4, 556.8)	`MIL`	HLL	PPM	10^{-8}
					GRHydro	HLLE	PPM	10^{-8}
					WhiskyTHC	HLLE	MP5	10^{-8}

resolution, L_i corresponds to the solution on grid i, and Δx_i corresponds to the grid resolution for grid i. For all cases involving fluid variables we typically consider the reference solution L_0 in Eq. (22) to be the initial data itself. The use of such a reference solution can prove troublesome in cases with unreliable or faulty initial data. In several of the cases we consider, we find a lack of convergence toward the continuum solution even at the start of the simulation, suggesting nonconvergent initial data. In such cases we consider the *self*-convergence of a given solution, which is demonstrated when

$$(L_{\mathrm{CR}} - L_{\mathrm{HR}})\left[\left(\frac{\Delta x_{\mathrm{MR}}}{\Delta x_{\mathrm{HR}}}\right)^n - 1\right] = (L_{\mathrm{MR}} - L_{\mathrm{HR}})\left[\left(\frac{\Delta x_{\mathrm{CR}}}{\Delta x_{\mathrm{HR}}}\right)^n - 1\right], \tag{23}$$

where n is the expected convergence order, L is the solution on a given grid, Δx is the grid resolution for a given grid, and the labels CR, MR, and HR correspond to the canonical, medium, and high resolution grids, respectively. We refer to quantities scaled according to Eq. (23) using the symbol $c[L]$, for quantity L.

We monitor the maximum rest mass density $\rho_{b,\max}$ and minimum of the lapse function $\alpha_{\min}$ to qualitatively assess stability and collapse. We also consider spacetime quantities, such as the Hamiltonian constraint violations, in order to assess convergence of the solution. We consider 2D snapshots of several fluid quantities, including the rest mass density ρ_{b}, temperature T, and different components of the magnetic field strength B_i, where relevant. When considering rotational properties of the fluid, we compute the angular velocity in the equatorial plane as

$$\Omega = \frac{Xv^y - Yv^x}{\varpi^2}, \tag{24}$$

where $\varpi \equiv \sqrt{X^2 + Y^2}$, $X \equiv (x - x_{\mathrm{com}})$, $Y \equiv (y - y_{\mathrm{com}})$, x and y are the Cartesian coordinates, and x_{com} and y_{com} correspond to the coordinates of the Newtonian center-of-mass.

In cases where we consider gravitational radiation, we extract GWs using the Newman-Penrose formalism [99,100], with focus on the $s = -2$ spin-weighted spherical harmonic decompositions of the Newman-Penrose scalar Ψ_4. The coefficients of the spin-weighted decomposition are labeled $\Psi_4^{l,m}$, where l and m are the degree and order for the spherical harmonics. We extract $\Psi_4^{l,m}$ from the numerical solution at fixed concentric spheres with increasing coordinate radii $r_{\mathrm{ex}} = \eta$km, where η takes on several discrete values $70 \lesssim \eta \lesssim 450$. When extracting GWs, we ensure to use the values of $\Psi_4^{l,m}$ which are in the wave zone. Where relevant, we compute the GW strain h from

$$\Psi_4 = \ddot{h}_+ - i\ddot{h}_\times, \tag{25}$$

using the fixed-frequency integration method [95]. We identify the time of merger t_{mer} as the time when the GW amplitude reaches an absolute maximum during the merger.

C. G2TOV: Test using a TOV star with polytropic EOS

We first consider the case of TOV initial data constructed using a $\Gamma = 2$ polytropic EOS. We construct initial data

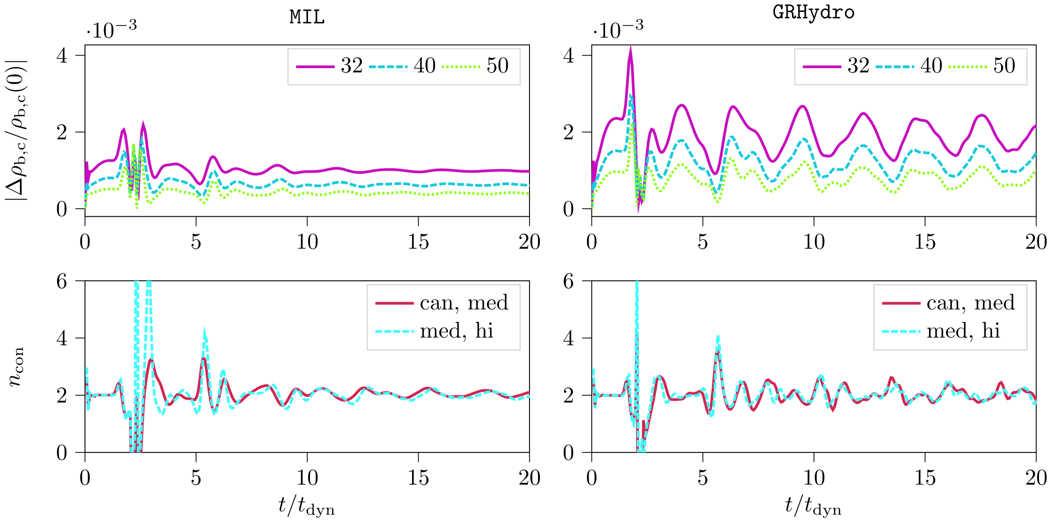

FIG. 2. Left panel: Results of test G2TOV in the case of evolution with the `MIL` code. In the top panel we show the relative change in central rest mass density compared to the value at $t = 0$, where $\Delta\rho_{\rm b,c} \equiv \rho_{\rm b,c}(t) - \rho_{\rm b,c}(0)$. We depict results for grid resolutions that employ 32, 40, and 50 grid points per NS radius using solid magenta, dashed blue, and dotted green lines, respectively. In the bottom panel we show the convergence rate, calculated using Eq. (22), for the quantity depicted in the top panel in cases where we compare the canonical/medium and medium/high resolution results using solid red and dashed blue lines, respectively. Right panel: same as left panels but for the GRHydro code.

using the `TOVSolver` thorn within the Einstein Toolkit with a central rest mass density of $\rho_{\rm b,c} = 0.00129$ and polytropic constant of $\kappa = 100$. We consider a static spacetime and do not employ mesh refinement (i.e., the test is carried out in the Cowling approximation). This is the only test we consider under the Cowling approximation. The half-side length of the solution grid extends to $1.25R_{\rm NS}$, where $R_{\rm NS} \approx 8M_\odot$ is the NS radius. Although the outer grid boundary for test G2TOV is significantly close to the surface of the stellar solution, we emphasize that test G2TOV is done in the Cowling approximation, such that the spacetime is static. We find that the regions near the grid boundary remain atmospheric at all grid resolutions tested, and because the spacetime is static, we do not worry about radiation reflection from the grid boundaries affecting the fluid solution. Moreover, for this test we are mainly concerned about the convergence of the solution on the central grid point which, as suggested by Fig. 2, changes by at most ~0.4% and 0.2% for the simulations employing GRHydro and `MIL`, respectively.

Our canonical, medium, and high resolution grids employ grid spacings such that there are 32, 40, and 50 grid points covering the NS radius, respectively. We evolve with the `MIL` code and with the GRHydro code as a comparison. In the case of the GRHydro evolution we employ the Harten-Lax-vanLeer-Einfeldt (HLLE) [98] Riemann solver and PPM reconstruction for maximum algorithmic overlap with `MIL`.

In the top panel of Fig. 2 we show the evolution of the relative change in the central rest mass density, where $\Delta\rho_{\rm b,c} \equiv \rho_{\rm b,c}(t) - \rho_{\rm b,c}(0)$ over a timescale of $20t_{\rm dyn}$, where

$$t_{\rm dyn} = 1/\sqrt{\rho_{\rm b,c}(0)} \tag{26}$$

is the dynamical time. We show results for the low-, medium-, and high-resolution grids, using solid magenta, dashed blue, and dotted green lines, respectively. We find that the evolution of a polytropic TOV star proceeds as expected, with oscillations in the rest mass density not exceeding ~0.2% and decreasing over time as the model settles. In the lower panel of Fig. 2 we also depict the convergence order of $\Delta\rho_{\rm b,c}/\rho_{\rm b,c}(0)$, with comparisons of the canonical/medium and medium/high resolution results depicted with the solid red and dashed blue lines, respectively. As we increase the grid resolution we find that the quantity $\Delta\rho_{\rm b,c}$ converges to zero at the expected second-order rate. In the right panel of Fig. 2 we show the results of the G2TOV test in the case of evolution with the GRHydro code. We find behavior in the GRHydro code which is consistent with that of the `MIL` code. When evolving with GRHydro we find that oscillations in the rest mass density are of larger amplitude when compared to the `MIL` code, but the solution nonetheless converges at the expected rate.

D. LSTOV: Test using a cold, magnetized TOV star with realistic EOS

In this set of tests we consider a cold, magnetized TOV star built using a realistic EOS. We construct initial data with several codes as a way to understand the systematic error introduced at the level of the initial data. In particular, we use the Cook code [101], the `RNSID` code [102,103] packaged within the Einstein Toolkit, and the LORENE code [104]. In the following we focus on initial data constructed with the Cook and LORENE codes, as the `RNSID` code produces similar results. We construct initial data for a cold, barotropic, β-equilibrated EOS obtained using the LS220 EOS. The initial configuration has a central energy (rest mass) density of $0.8 \times 10^{15}\ \mathrm{g\,cm^{-3}}$ ($0.74 \times 10^{15}\ \mathrm{g\,cm^{-3}}$), ADM (rest) mass of $1.4M_{\odot}(1.55M_{\odot})$, and circumferential radius of $R_{\mathrm{NS}} = 12.69$ km. We employ three levels of fixed mesh refinement, with the innermost, finest refinement level extending to $1.5R_{\mathrm{NS}}$, and every subsequent level extending to twice the distance of the adjacent finer level. For all tests we employ a finest-level grid resolution of $dx_{\mathrm{fin}} = R_{\mathrm{NS}}/64$ (which we refer to as canonical resolution), such that there are 64 grid points covering the NS radius. We superimpose a purely poloidal magnetic on nonmagnetized initial data which is confined to the interior of the NS, using the `Seed_Magnetic_Fields_BNS` code within the Einstein Toolkit. The field structure approximately corresponds to that generated by a current loop [105], with the toroidal component of the vector potential taking the form

$$A_{\phi} = \frac{\pi r_0^2 I_0 \varpi^2}{(r_0^2 + r^2)^{3/2}} \left(1 + \frac{15 r_0^2 (r_0^2 + \varpi^2)}{8(r_0^2 + r^2)^2}\right), \qquad (27)$$

where r_0 is the current loop radius, I_0 is the loop current, $r^2 = \varpi + z^2$, $\varpi = (x - x_{\mathrm{NS}})^2 + (y - y_{\mathrm{NS}})^2$, and x_{NS} and y_{NS} are the initial coordinates of the NS center-of-mass. This choice of magnetic field exhibits a $1/r^3$ falloff outside the NS, as expected for a dipole field, but as mentioned above we do not extend the field to the exterior; the `Seed_Magnetic_Fields_BNS` code employs a cutoff pressure P_{cut} below which the vector potential is set to zero, which we set to $P_{\mathrm{cut}} \approx 0.01 P_{\mathrm{max}}(0)$ [where $P_{\mathrm{max}}(0)$ is the maximum fluid pressure at the start of the simulation]. The initial field strength is such that $B_{z,\mathrm{max}} \approx 1.2 \times 10^{17}$ G, which corresponds to a ratio of magnetic to fluid pressure of $b^2/2P \approx 0.001$, where $b^2 \equiv b^{\mu} b_{\mu}$.

As a comparison, we also show the evolution of this model with the Spritz code [58], which uses an implementation of the `Palenzuela` solver for finite temperature, tabulated EOSs. We employ the Spritz code with PPM reconstruction and the HLLE Riemann solver for maximum algorithmic overlap with IllinoisGRMHD. In Fig. 3 we show the evolution of the central rest mass density $\rho_{\mathrm{b,c}}$ (top panels), maximum temperature T_{max} (center panels using dashed lines), central temperature T_c (center panels using solid lines), and maximum magnetic field components (lower panel) as functions of time. We depict results corresponding to `MIL` and Spritz using magenta and blue lines, respectively. In the top left, center left, and lower panels of Fig. 3 we focus on cases where initial data was constructed with the Cook code. We find that the evolution of the rest-mass density is qualitatively the same between the `MIL` the Spritz codes, exhibiting oscillations of at most $\sim$1%. The configuration shows an initial relatively large oscillation in the rest mass density of $\gtrsim$1% which peaks at $t \approx 1\text{–}2t_{\mathrm{dyn}}$. This initial large oscillation is likely due to the creation of a cold barotropic EOS from the finite temperature LS220 EOS table as previously discussed, which requires interpolation and root-finding routines (we note that similar oscillations happens in test cases with unmagnetized initial configurations, which suggests that the superimposed magnetic field is likely not the culprit behind the oscillations). It is possible that the use of a cold EOS created from a finite temperature EOS results in initial data with percent-level numerical errors, which is consistent with the oscillations of amplitude $\sim 1.01\rho_{\mathrm{b,c}}(0)$ shown in the top left panel of Fig. 3. We note that initial data produced with the Cook code uses a compactified radial grid (which is also true for both `RNSID` and LORENE) in spherical polar coordinates, and as such interpolation of the initial data onto the Cartesian grids of the Cactus infrastructure results in additional numerical errors in the initial configuration. As the initial configuration settles to the nearest equilibrium, it begins to exhibit smaller oscillations in the rest mass density of $\sim$0.5%. These oscillations decay over a timescale of $t \approx 20t_{\mathrm{dyn}}$. In the center left panel of Fig. 3, we show the evolution of the maximum and central temperatures using dashed and solid lines, respectively, for initial data constructed with the Cook code. The initial configuration is cold for all solvers tested, with a central (maximum) temperature of $T_{\mathrm{c}} \approx 0.1$ MeV ($T_{\mathrm{max}} \approx 0.5$ MeV). We note that initial data is constructed for an isothermal star at $T = 0.1$ MeV. However, once the initial data for ρ_b, ϵ, Y_{e}, and P are interpolated onto the solution grid, we recover the temperature for self-consistency among the fluid variables, which leads to errors in the temperature profile at $t = 0$, near the stellar surface. Early in the simulations, regardless of the evolution code used, the model develops a warm atmosphere, which becomes increasingly warm as the star undergoes small oscillations. The central configuration remains cold as the temperature of the atmosphere increases and saturates to $T_{\mathrm{max}} \approx 8$ MeV. In the lower panel of Fig. 3 we also depict the maximum value of the magnetic field for each simulation (along the z-direction), again focused on the case with initial data constructed with the Cook code. We find that for test LSTOV the `MIL` code preserves the initial magnetic field strength for all field components to a better extent than the Spritz code. The largest change in the magnetic field strength is seen in B_z, with a maximal change of at most 7% for evolutions with the `MIL` code, but a large oscillation of close to 25% for evolutions with the Spritz code. For test LSTOV while evolving with the Spritz code, we find that the large oscillation observed in B_z for evolutions with the Spritz code persists at higher grid resolutions as well, with changes in $B_{z,\mathrm{max}}$ of up to 25% for simulations employing a $\sim$60% finer grid than that

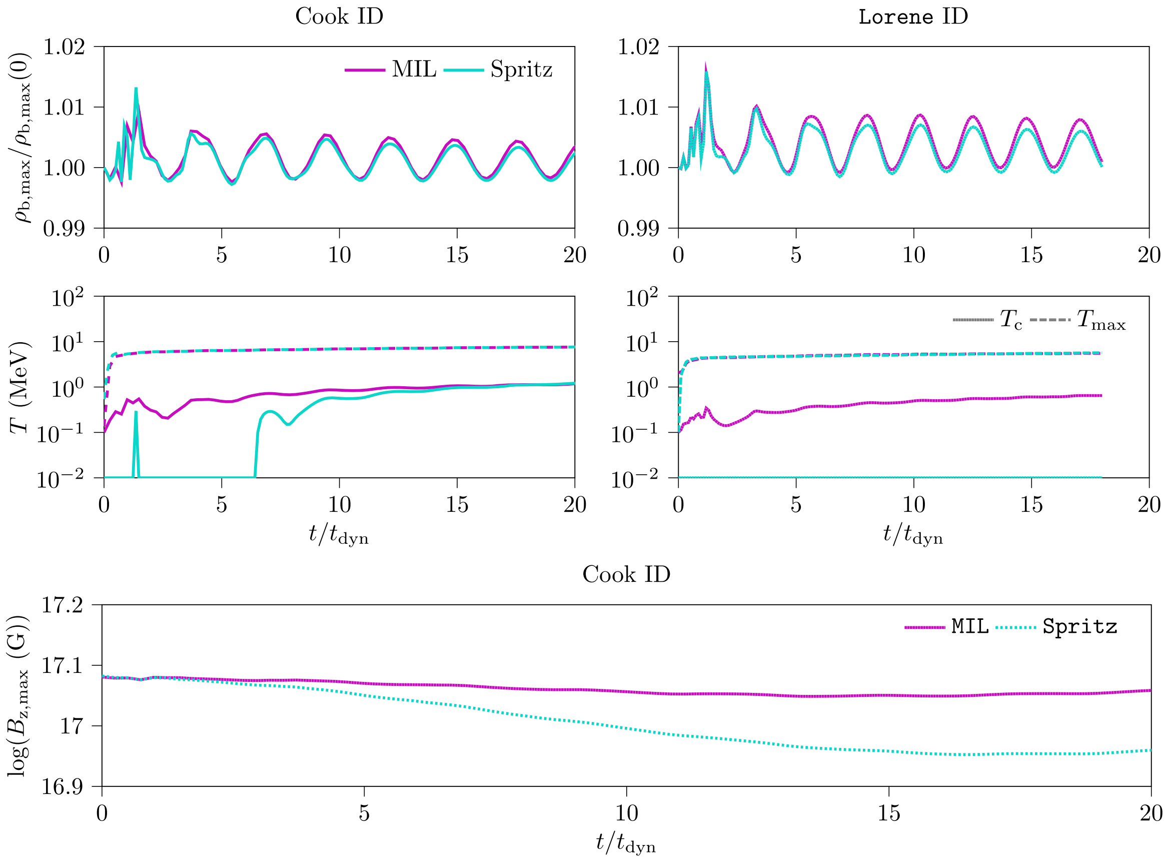

FIG. 3. Top panel: Central rest mass density $\rho_{\mathrm{b,c}}$ scaled by the value at $t = 0$ as a function of time for test LSTOV in the case of evolution with the `MIL` and Spritz codes; we show the results corresponding to evolution with the `MIL` and Spritz codes using magenta and blue lines, respectively. The left and right panels correspond to simulations using initial data constructed by the Cook *et al.* code [101] and the LORENE code [104], respectively. Center panel: Central temperature T_{c} (solid lines) and maximum temperature T_{max} (dashed lines) as functions of time, we use the same color scheme and left/right panel ordering as the top panel. Lower panel: Maximum value of the magnetic field components B_z; we show the results corresponding to evolution with the `MIL` and Spritz codes using magenta solid and blue dotted lines, respectively.

presented in Fig. 3 (such that there are 100 gridpoints covering the NS radius). However, we find that this initial large oscillation of the maximum magnetic field strength decreases for weaker initial magnetic field strengths, with initial values $B_{z,\mathrm{max}} \approx 1 \times 10^{15}$ G leading to an oscillation of approximately 1%. In the next section, we discuss the conservation of the magnetic field strength and structure in the context of a binary neutron star inspiral and discuss potential causes for the large change in magnetic field strength and structure for evolutions with the Spritz code. We note that the definition of the magnetic field differs by a factor of $\sqrt{4\pi}$ between Spritz and IllinoisGRMHD, such that $B^i_{\texttt{Spritz}} = \sqrt{4\pi} B^i_{\texttt{IllinoisGRMHD}}$, which has been accounted for in all tests.

To test the role of initial data on the evolution of the model, we consider the same initial configuration but instead built with the LORENE and `RNSID` codes. We consider a model with the same central rest mass density and employ the same grid hierarchy as in the case considering initial data constructed with the Cook code. In the right panel of Fig. 3 we present results of the evolution with the `MIL` and Spritz codes using magenta and blue lines, respectively, with focus on the case of initial data built with the LORENE code. In all cases considered, we observe initial oscillations of the rest mass density which do not exceed 2% and subsequent oscillations which are at the level of ~1%. The central panel of Fig. 3 shows the difference in the evolution of the maximum temperature T_{max} (which occurs near the atmospheric low-density regions of the star) and of the central temperature T_{c}. When using ID constructed with the LORENE code, we find that both T_{max} and T_{c} remain lower, when comparing to cases that employ ID constructed with the Cook code. We find that evolution with the Spritz code results in a central region which remains cold ($T \leq 0.1$ MeV, shown using the blue solid line in the lower-right panel of Fig. 3) throughout the simulation, whereas the `MIL` code produces slightly warmer central regions of up to $T_{\mathrm{c}} = 0.3$ MeV. We note that in the `MIL` code we impose a lower bound on the temperature of 0.1 MeV to avoid EOS interpolations near

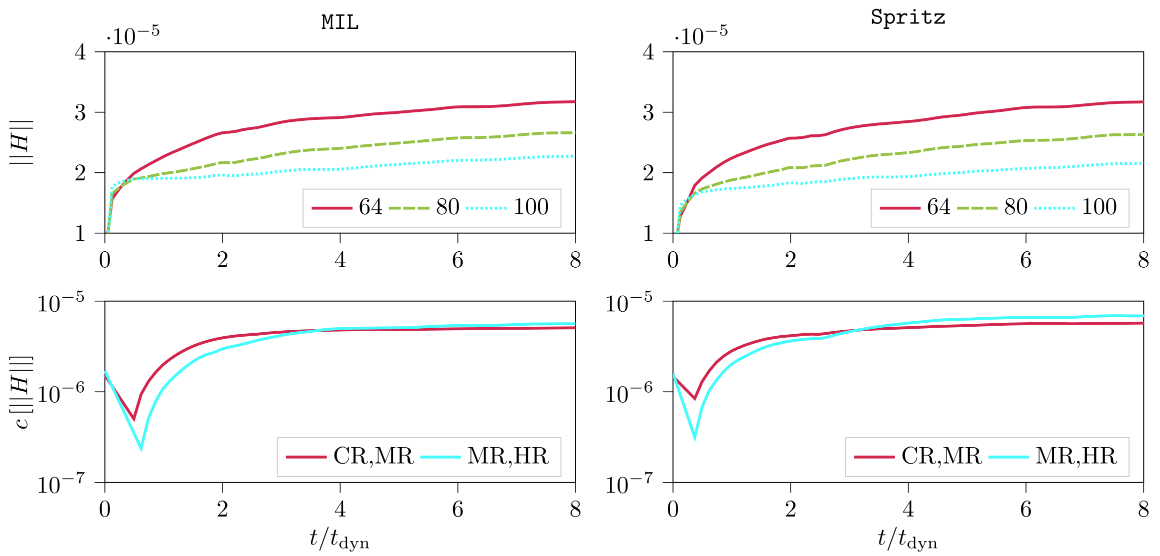

FIG. 4. Left panel: Results of test LSTOV in the case of evolution with the `MIL` code. We depict the convergence of the L2 norm of the Hamiltonian constraints $\|H\|$ in the case of a magnetized TOV star built using the LS220 EOS with the LORENE code, evolved with the `MIL` code and `Palenzuela` solver. The top panel depicts $\|H\|$ for the low-, medium-, and high-resolution grids using solid red, dashed green, and dotted blue lines, respectively. The bottom panel depicts $\|H\|$ scaled according to second-order convergence while comparing the canonical and medium (CR, MR) and medium and high (MR, HR) resolution results using red and blue lines, respectively. Right panel: Same as the left panel but in the case of evolution with the Spritz code.

the table limits. Such a low-temperature floor is not imposed in the Spritz code, which may be the reason for the slight difference in $T_{\rm c}$ between the two codes. A central temperature of $T_{\rm c} = 0.3$ MeV is still significantly cold from a nuclear perspective and tests of TOV star evolution without imposing the lower bound on T results in a temperature profile which is consistent with the Spritz evolutions. The evolution of the rest mass density and temperature presented in Fig. 3 is typical of stable TOV stars. The fact that a realistic TOV star remains stable and cold over a timescale of $t \approx 20 t_{\rm dyn}$ and the qualitatively similar behavior between the `MIL` and Spritz codes indicates that our implementation of the solvers within `ConservativeToPrimitive` is reliable. Tests with `RNSID` initial data produce very similar results to those presented in Fig. 4.

We consider the models constructed using the Cook code for a resolution study. We evolve the Cook initial data in two additional higher-resolution simulations with both the `MIL` and Spritz codes. For medium (high) resolution tests, we employ grid resolutions of $dx_{\rm med} = dx_{\rm can}/1.25$ ($dx_{\rm hi} = dx_{\rm can}/1.25^2$) such that there are 80 (100) grid points covering the NS radius, where $dx_{\rm can} = dx_{\rm fin}$ as listed in Table I for the LSTOV test. In the top panel of Fig. 4 we show the L2 norm of the Hamiltonian constraints $\|H\|$ for the canonical (solid red lines), medium (dashed green lines) and high (dotted blue lines) resolution grids, in cases where we evolve with the `MIL` (left panel) and Spritz (right panel) codes. We find that $\|H\|$ is generally small and converges toward zero (the expected value in the continuum limit) for higher-resolution grids. We also note that the evolution of $\|H\|$ is in agreement between the `MIL` and Spritz codes. However, we find that $\|H\|$ converges toward zero at a rate which is slower than expected, for both evolution codes considered. Specifically, we find a linear convergence rate of $\|H\|$ toward zero. We note that the expected convergence rate of the evolutions we consider is second order, and that second-order convergence is demonstrated for polytropic initial data (see Fig. 2). It is likely that the lower-order convergence observed for tests employing realistic EOSs is due to errors in the initial data. In particular, the initial data codes we consider make use of compactified spherical polar coordinate grids, which are not the same as the evolution grids. Numerical errors are therefore inevitable when interpolating solutions from their original compactified coordinates to the Cartesian grids employed within the Einstein Toolkit. We point out that the evolution of magnetized TOV stars presented in this section is consistent with stable evolution, and coincides with the results presented in [58] for Spritz. As there are likely errors at the level of the initial data when using realistic EOSs we instead consider the *self*-convergence of the solution. In the lower panels of Fig. 4 we show $\|H\|$ scaled according to second-order convergence [i.e., $n = 2$ in Eq. (23)] for test LSTOV while comparing the canonical and high resolution (solid red lines) and medium and high resolution solutions (dashed blue lines). We again show results for evolution with the `MIL` and Spritz codes using the left and right panels, respectively. The lower panels of Fig. 4 demonstrate that the solution exhibits self-convergence at the expected order.

We also consider the same test of convergence in cases with initial configurations built with the LORENE and `RNSID` codes and find similar results to those presented in Fig. 4. We point out that the construction of TOV initial data is algorithmically very similar between the Cook, `RNSID`, and LORENE codes, as they all make use of compactified spherical coordinate grids. As such, interpolation errors that results from a change in the coordinate system will arise in all tested cases.

E. G2BNS: Test of a BNS with a gamma-law EOS

As an additional test of the solvers implemented within `ConservativeToPrimitive`, we consider the evolution of a BNS system built assuming a polytropic EOS. We consider this test because the `OIL` code can reliably simulate such a system and because it provides a stringent test of the `MIL` code new algorithms in a highly relativistic scenario. We assume a polytropic constant of $\kappa = 123.61$ and adiabatic index of $\Gamma = 2$. We construct initial data for an equal mass BNS with an orbital separation of 45 km using the LORENE code; the ADM mass and radius of each binary component are $M_{\rm NS} = 1.4M_\odot$ and $R_{\rm NS} = 10.36$ km, respectively. During the evolution, we assume a Γ-law EOS with adiabatic index $\Gamma = 2$. In this test we employ AMR, with two sets of nested boxes centered on each NS and each employing seven levels of refinement. We employ an additional set of nested boxes with seven levels of refinement centered at the origin of the solution to reliably resolve the BNS merger remnant. The outer grid boundary is located at $R_{\rm out} = 582.12\ {\rm km} \approx 38.4R_{\rm NS}$ and the half-side length of the finest level around each star extends to $\sim 1.25R_{\rm NS}$ with each subsequent coarser level extending to $\eta R_{\rm NS}$ where $\eta \in (1.7, 2.5, 5.8, 11.6, 23.2)$. The finest-level grid resolution is set to $dx_{\rm fin} \approx R_{\rm NS}/40$, such that we initially resolve each binary component with at least 40 grid points along the radius. For test G2BNS, we consider both an unmagnetized, and magnetized scenario, because not all of the evolution codes considered can handle a constrained-transport evolution of the magnetic field. In cases where a magnetic field is also considered, we seed purely poloidal fields which are confined to the interior of each NS, with vector potential of the form Eq. (27), such that the initial maximum field strength is $B_{\rm max} \approx 2.3 \times 10^{15}$ G.

We simulate the system with the use of `MIL` and track the evolution of the rest mass density $\rho_{\rm b}$, fluid three-velocity v^i, and the magnetic field B^i, when relevant. We approximate the merger time $t_{\rm mer}$ by considering the time when the rest mass density reaches a maximum during the evolution. We compare to several other open-source codes in the literature, including the `OIL`, GRHydro, WhiskyTHC, and Spritz codes. Similar to other tests, we employ all evolution codes considered with the HLL/HLLE Riemann solver and PPM reconstruction for a suitable comparison to IllinoisGRMHD. WhiskyTHC case we use MP5 reconstruction [106]. We also note that WhiskyTHC uses an internal rescaling of the rest mass density, and as such the initial conditions between WhiskyTHC and other cases is not identical. However, this internal rescaling of the rest mass density results in a maximal relative difference of less than 0.4% between cases.

1. Unmagnetized BNS

In cases where we do not consider the evolution of the magnetic field, we are free to compare the results provided by all of the open-source GRHD codes considered, as they are all able to handle an unmagnetized BNS system built with a polytropic EOS. As such, we compare the results of simulations using the `MIL`, `OIL`, GRHydro, WhiskyTHC, and Spritz codes. We emphasize that the numerical grids, initial data, reconstruction scheme, conservative-to-primitive schemes, and flux treatment for all of these simulations are chosen for maximum overlap, with the latter three likely having differences in implementation between codes. The comparison of results presented in this section was the closest scenario to a one-to-one comparison between open-source numerical codes that we could obtain.

In the left panel of Fig. 5 we show the maximum rest mass density in the simulations for the unmagnetized case of test G2BNS. We show results for the GRHydro (solid purple line), `OIL` (dashed magenta line), `MIL` with and without[3] the use of the solvers within `ConservativeToPrimitive` (dash-dotted golden and dotted orange lines, respectively), Spritz (dash-dash-dotted line), and WhiskyTHC (blue dash-dash-dot-dotted line) codes. We find that during the inspiral, all codes behave qualitatively the same, with the maximum rest mass density exhibiting oscillations of $\lesssim 2\%$ for the first $t \approx 5$ ms of the inspiral. We find that the merger time for these simulations all fall within 9.81 ± 0.1 ms, resulting in a difference of at most 2%. On the other hand, we find significant deviations of the simulation results after the merger. Each code considered results in different size oscillations of the rest mass density for the first 10 ms after merger. The GRHydro code results in the largest postmerger oscillations in $\rho_{\rm b,max}$, and is the only case where $\rho_{\rm b,max}$ shows a tendency to grow. In all cases besides GRHydro, the evolution of $\rho_{\rm b,max}$ is qualitatively similar, and the post-merger value of $\rho_{\rm b,max}$ oscillates between $0.9\rho_{\rm b,max}(0) - \rho_{\rm b,max}(0)$. We find qualitatively similar GW emission in all cases, but the frequency corresponding to peak GW emission shows a spread of at most $\sim 15\%$ between cases. Given that all codes use similar flux treatments (we note that in all other tested codes, the closest Riemann solver to the HLL solver

---

[3] We consider an evolution with the `MIL` code that does not employ the solvers within `ConservativeToPrimitive`, and instead uses the original solvers within the `OIL` code. We label this test as $\texttt{MIL}_{\rm noC2P}$ in Fig. 5. The fact that the $\texttt{MIL}_{\rm noC2P}$ evolution proceeds identically to the `OIL` case ensures that other changes made to the code (e.g., the changes that allow for passive $Y_{\rm e}$ advection) do not affect the dynamics.

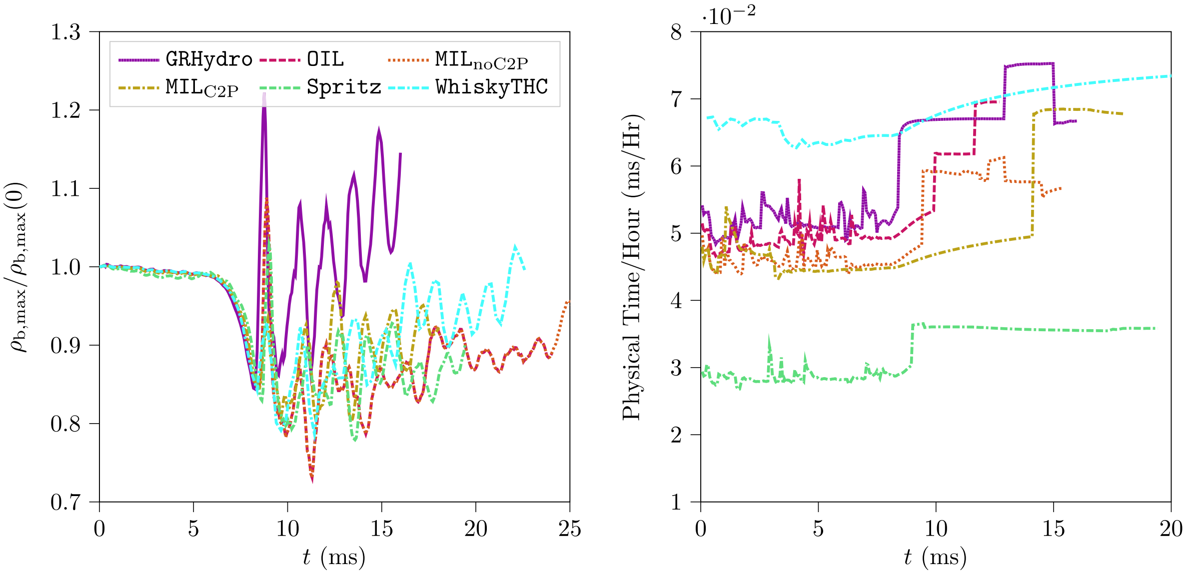

FIG. 5. Left panel: Maximum rest mass density ρ_{b}, scaled by the value at $t = 0$, for an unmagnetized BNS assuming a $\Gamma = 2$ polytropic EOS. We depict the results corresponding to the GRHydro (solid purple line), `OIL` (dashed magenta line), `MIL` with and without the use of the solvers within `ConservativeToPrimitive` (dash-dotted golden and dotted orange lines, respectively), Spritz (dash-dash-dotted line), and WhiskyTHC (blue dash-dash-dot-dotted line) codes. Right panel: Same format as the left panel, but instead depicting the physical time produced by each simulation per hour of run time, across 224 `Intel Haswell V3` cores. We use identical initial data, grid structures, reconstruction schemes, conservative-to-primitive error tolerance, and computational resources in all cases, as well as similar flux treatments. The tests presented here are as close as possible to a one-to-one comparison between codes that we could achieve.

adopted in IllinoisGRMHD is the positivity-preserving HLLE solver [98], which is what we adopt in evolutions with Spritz, GRHydro, and WhiskyTHC) and reconstruction schemes (in the case of WhiskyTHC, we use MP5 while in all other cases PPM is used), along with identical numerical grid, initial data, conservative-to-primitive error tolerance, and spacetime evolution scheme, it is unclear why the GRHydro code deviates so strikingly from all other codes, and why there is a general lack of quantitative agreement in the postmerger evolution between most codes. Key differences between codes may arise in the implementation of conservative-to-primitive algorithms, which may be the chief reason for discrepancies between codes. We leave the investigation of algorithmic implementations within open-source GRHD codes, and a more detailed code comparison, to future work.

In the right panel of Fig. 5 we show a measure of the performance for each code. Specifically, we show the amount of physical time produced per hour for each simulation, using the same amount of computational resources (specifically, 224 `Intel Haswell V3` cores). We find that, for this case study, the WhiskyTHC code performs the fastest (producing around 0.07 ms/hour throughout the inspiral and in the postmerger phase, respectively), while the Spritz code performs the slowest (producing around 0.03 ms/hour during the inspiral and 0.035 ms/hour in the postmerger phase, respectively). All other codes perform similarly, with 0.05 ms/hour during the inspiral and 0.07 ms/hour in the postmerger. We note that the hierarchy suggested by the right panel of Fig. 5 is not a final verdict on the performance of each code, and only demonstrates the performance for the particular case considered. We find that changes to the specific settings governing the conservative-to-primitive inversion and atmosphere control can significantly impact the performance of the codes considered. For example, we find that the conservative-to-primitive routines within Spritz often perform as reliably as in other codes with the use of fewer algorithmic iterations, which serves to speed up the code.

2. *Magnetized BNS*

In cases where we *do* consider a magnetic field for test G2BNS, we compare the evolution provided by the `MIL` code to that provided by the `OIL` and Spritz codes (note that the Spritz code uses an implementation of the `Noble` solver for polytropic EOSs). In the top panel of Fig. 6 we show equatorial snapshots of the rest mass density for test G2BNS at time $t \approx 7.3$ ms, corresponding to first contact between the binary components. From left to right in Fig. 6 we show snapshots for evolution with the `OIL`, `MIL`, and Spritz codes. In the rightmost panel of Fig. 6 we also show a snapshot corresponding to the `MIL` code in the case where we use an alternative set of reconstructed variables to calculate the specific enthalpy h (labeled $\mathtt{MIL}_{\mathrm{b}}$), as will be discussed in further detail below. We find that all codes considered produce very similar results as far as the merger

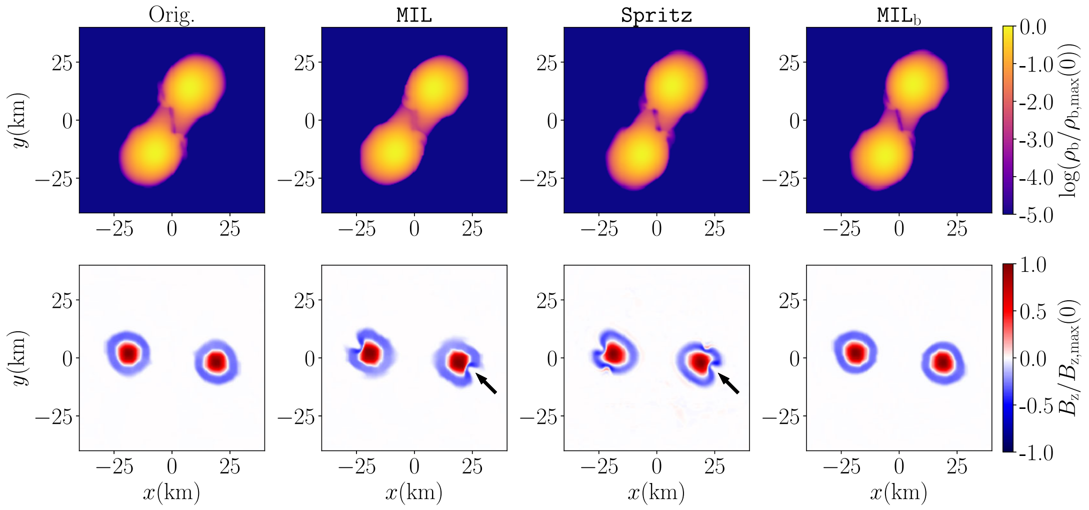

FIG. 6. Top panel: Equatorial snapshots of the rest mass density ρ_b (scaled by the maximum value at $t = 0$) for test G2BNS at $t \approx 7.3$ ms. From left to right we show results for the `OIL`, `MIL`, and Spritz codes. The rightmost panel depicts snapshots for an evolution using the `MIL` code wherein we use an alternative calculation of the specific enthalpy h [see Eqs. (28) and (29)]. Bottom panel: Same as the top panel, but showing the equatorial snapshots of the z-component of the magnetic field B_z (scaled by the maximum value at $t = 0$) for test G2BNS$_{\rm mag}$ at $t \approx 3.54$ ms. We point out common features that develop in the magnetic field structure using black arrows.

time and features during inspiral are concerned. We find that the merger times agree to within 2%, specifically $t_{\rm mer} = 47.85 t_{\rm dyn}$, $t_{\rm mer} = 47.85 t_{\rm dyn}$, and $t_{\rm mer} = 49.04 t_{\rm dyn}$ for the `OIL`, `MIL` and Spritz cases, respectively. At a time close to and preceding merger, pictured in the top panel of Fig. 6, the main difference between the results of each code can be observed in the region between the binary components, likely due to how low density, near-atmosphere regions are treated in the conservative-to-primitive solvers implemented in each code.

In the lower panel of Fig. 6 we show snapshots of the z-component of the magnetic field during the inspiral at time $t \approx 3.54$ ms. Moving from left to right, we again show results for the `OIL`, `MIL`, and Spritz codes; the rightmost panel corresponds to the `MIL` code with primitive reconstruction that agrees with the `OIL` code [see Eq. (28)], which we label `MIL`$_b$. Specifically, in `OIL` the calculation of specific enthalpy in the flux computation uses an analytic expression for ϵ, based on a Γ-law EOS. In the case where the adiabatic index for the thermal component of the EOS is $\Gamma_{\rm th} = 2$, the specific enthalpy becomes

$$h = 1 + \epsilon(P_{\rm r}) + \frac{P_{\rm r}}{\rho_{\rm b,r}} = 1 + \frac{2P_{\rm r}}{\rho_{\rm b,r}}, \tag{28}$$

where $P_{\rm r}$ and $\rho_{\rm b,r}$ are the reconstructed pressure and rest mass density, respectively. On the other hand, the solution within `MIL` and the Spritz codes calculate these specific enthalpies based on the reconstructed values of ϵ

$$h = 1 + \epsilon_{\rm r} + \frac{P(\epsilon_{\rm r}, Y_{\rm e,r}, \rho_{\rm b,r})}{\rho_{\rm b,r}}, \tag{29}$$

where $P(\epsilon_{\rm r}, Y_{\rm e,r}, \rho_{\rm b,r})$ is the pressure for a generic EOS based on the reconstructed values of the specific internal energy $\epsilon_{\rm r}$, electron fraction $Y_{\rm e,r}$, and rest mass density $\rho_{\rm b,r}$. In other words, the `OIL` code uses the reconstructed pressure to calculate h, whereas the `MIL` and Spritz codes use the reconstructed specific internal energy. The calculation of h based on Eq. (29) is required for self-consistency among the fluid variables in the case of generic EOSs, but in the case of analytic EOSs, it is preferable to proceed with calculation of h based on Eq. (28) for accuracy. The differences in use of reconstructed variables plays a role in the preservation of the magnetic field structure. We find that the `OIL` code tends to advect the magnetic field structure accurately for longer times when compared to the other tested codes. The `MIL` and Spritz evolutions all produce similar features in the magnetic field near the low density parts of the star. In particular, solutions employing generic solvers, which are compatible with realistic EOSs, produce "kinks" in the magnetic field structure, as highlighted by the arrows in the lower panel of Fig. 6. We note that, despite the artifacts seen in the magnetic field structure for solvers compatible with realistic EOSs, the magnetic field is generally properly advected during inspiral in all cases. Moreover, the sizes of the artifacts decrease (and the time at which they appear increases) with the use of higher-resolutions grids. The agreement of `MIL` with the well-tested Spritz and `OIL` codes suggests that the fluid and magnetic field solutions these codes provide are reliable for highly dynamical systems. For the sake of consistency with the `OIL` code, we adopt the same methods within `MIL` for calculating specific enthalpies in cases where we consider analytic EOSs. We find that this results in a field structure that is conserved throughout more of the inspiral, as illustrated by the disappearance of the "kinks" in the field structure in the rightmost snapshot in Fig. 6, wherein we use the same method for calculating specific enthalpies as in the `OIL` code. In other words, moving forward, within the `MIL` code we calculate h based on Eqs. (28) and (29) for analytic and finite temperature EOSs, respectively.

F. LSBNS: Tests using a BNS with finite temperature equations of state

In this section we consider the evolution of BNS systems built using a finite-temperature EOS. Specifically, we consider the LS220 EOS and construct initial data for an unmagnetized, equal-mass BNS with ADM mass $M_{\rm ADM} = 2.7 M_\odot$, and initial separation of 45 km. Our evolution grid extends to approximately 830 km at the outer boundary and consists of three additional sets of six nested cubes (corresponding to seven levels of refinement); we center one set of cubes at the origin of the solution grid, and the other two sets of cubes are used to track the center of each neutron star during the inspiral. The grid resolution of each nested cube is half that of the adjacent larger cube, and the finest-level grid resolution is set such that each NS is resolved with at least 50 grid points per radius during the inspiral; generally, the grid resolution of a given refinement level is $dx_l = 2^{(7-l)} R_{\rm NS}/50$, where $l = 7(1)$ labels the finest (coarsest) resolution grid and $R_{\rm NS} \approx 10.31$ km. We note that although these grid resolutions are lower than the standard used for production-level runs, such grid resolutions are routinely considered in the context of BNS mergers as low-resolution simulations which are useful in convergence studies [107,108]. We employ these grids because we were predominantly interested in comparisons of different code features. In the context of BNS mergers, we leave comparisons using high-resolution simulations and a full convergence study to future work, since each of the codes we consider has been tested extensively for convergence. The half-side length of the finest-resolution grid is set to $r_{\rm HSL,7} = 1.25 R_{\rm NS} \approx 12.95$ km; generally, the half-side length corresponding to a given refinement level is $r_{\rm HSL,l} = 1.25 R_{\rm NS} 2^{(7-l)}$. In the case of the `MIL` code, we use the HLL Riemann solver. In all other cases we use the HLLE Riemann solver. We use PPM for primitive reconstruction for all evolution codes considered except WhiskyTHC, in which case we use MP5.

As a way to compare the different open-source GR(M) HD codes currently available, we consider the evolution of this system with use of the `MIL`, WhiskyTHC, and GRHydro codes. We present the maximum rest mass density for each evolution in Fig. 7, using a purple solid line, orange dashed line, and green dotted line for the case of `MIL`, WhiskyTHC, and GRHydro, respectively.

We identify the merger time $t_{\rm mer}$ as the time where the $l = 2$, $m = 2$ mode of the dominant polarization of the GW strain peaks; for reference, we show these GWs (specifically the Newman-Penrose scalar ψ_4 scaled by the system ADM mass and extraction radius of $r = 440$ km) in the lower panel of Fig. 7. We find merger times of $t_{\rm mer} = 13.57$ ms, $t_{\rm mer} = 11.72$ ms, and $t_{\rm mer} = 12.14$ ms in the case of the `MIL`, WhiskyTHC, and GRHydro evolutions, respectively, which results in an approximately 13.6% relative difference, at most, in merger times between cases. In Fig. 8 we show the characteristic strain [109] extracted from the simulations in test LSBNS, assuming a source at a distance of 10 Mpc. We find very similar inspiral GW signals from all codes considered, with relative differences in the phase of the gravitational wave strains of at most ∼2%. We find that the largest discrepancy between GW signals arises in the immediate postmerger environment. The postmerger environment is when the matter is in a highly turbulent and dynamical state; in this stage of the merger that the relative difference in the phase of the GW strain can reach as high ∼20% (when comparing the `MIL` and WhiskyTHC cases), but is typically closer to ∼10%. After this highly dynamical state, as the remnant begins to settle, all codes again produce very similar GW signals. The frequency corresponding to peak postmerger GW emission $f^{2,2}_{\rm peak}$ is similar between all codes, with the `MIL`, WhiskyTHC, and GRHydro evolutions resulting in $f^{2,2}_{\rm peak} = 3.11$ kHz, $f^{2,2}_{\rm peak} = 3.06$ kHz, and $f^{2,2}_{\rm peak} = 3.13$ kHz.

In the top, middle, and lower panels of Fig. 9 we show snapshots of the rest mass density and temperature in the

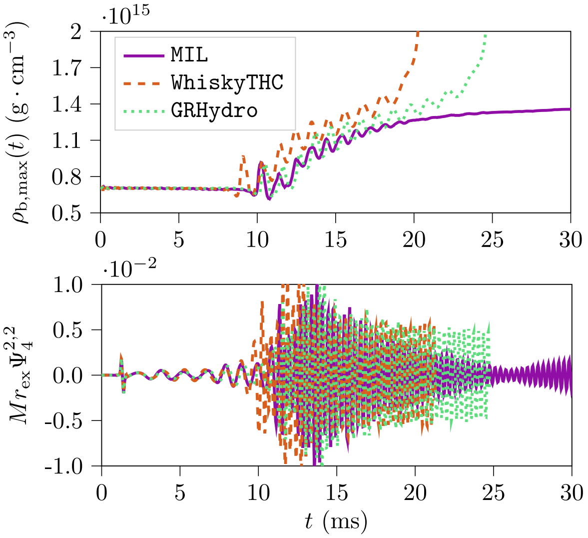

FIG. 7. Top panel: Maximum rest mass density $\rho_{\rm b,max}$ as a function of time for a BNS merger simulation using the LS220 EOS (test LSBNS). We depict the evolution of $\rho_{\rm b,max}$ in the case of the `MIL`, WhiskyTHC, and GRHydro codes using solid purple, dashed orange, and dotted green lines, respectively. Bottom panel: Same as the top panel, but showing the dominant component of the Newman-Penrose scalar as a measure of the GWs, scaled by the ADM mass M and extraction radius $r_{\rm ex} = 440$ km.

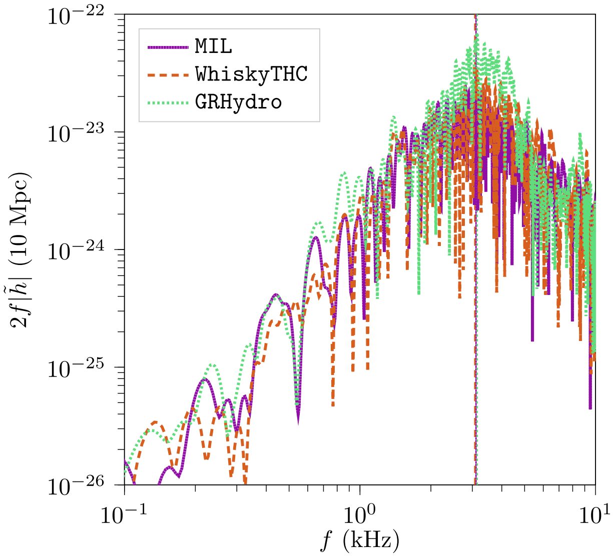

FIG. 8. GW signals corresponding to test LSBNS. We depict the characteristic strain at a distance of 10 Mpc from the source in the case of the `MIL`, WhiskyTHC, and GRHydro codes using solid purple, dashed orange, and dotted green lines, respectively. For a better comparison, we use the same duration signals in all cases, corresponding to times ranging from $t \approx 10$ ms to $t \approx 20$ ms (with signal duration $\tau_{\mathrm{signal}} \approx 10$ ms), which allows us to focus on the postmerger signals. The vertical lines of the same color and pattern mark the peak frequency for each case.

equatorial plane at key points during the merger for the case of `MIL`, WhiskyTHC, and GRHydro, respectively. For each evolution code considered we depict four snapshots, ordered by time from left to right at different stages of the merger corresponding to times $t \in (4.25, 8.37, 12.70, 18.88)$ ms. In the left-most panels in Fig. 9 we depict the system at a time during the inspiral. We generally find that the binary components remain cold during the inspiral, with typical temperatures inside the stars of $T \lesssim 1-2$ MeV. Temperatures of $\mathcal{O}(1\ \mathrm{MeV})$ within the stars are expected during the inspiral [36], and may be due to the growth of numerical error seeded at the level of initial data or interpolation errors from the EOS tables. We find that the stellar atmospheres become significantly warm as the stars inspiral, reaching temperatures of $T_{\mathrm{atmo}} \approx 5-10$ MeV before even the first orbit is complete (note that we employ a density cutoff to define the atmosphere, such that all matter with below $\rho_{\mathrm{b,atmo}} \approx 6.17 \times 10^{7}\ \mathrm{g\,cm^{-3}} \approx 10^{-7}\rho_{\mathrm{b,max}}(0)$ is set to the atmosphere as detailed in Sec. III B). This temperature profile (cold temperatures of $T \approx 1$ MeV within the stars and higher temperatures $T_{\mathrm{atmo}} \approx 10$ MeV in the atmospheres) is maintained during the entire inspiral up to merger, as depicted by the leftmost panels in Fig. 9. We find that, during the inspiral, the warmest temperatures develop in the case of the GRHydro and WhiskyTHC evolutions, reaching as high as $T \approx 5$ MeV in the bulk of the star and $T_{\mathrm{atmo}} \approx 15$ MeV in the atmosphere. These cases also exhibit warm, low-density clouds which develop around the binary components throughout the inspiral. However, the atmosphere is dynamically unimportant.

In the second-from-left column in Fig. 9 we depict the systems at a time near merger. At this point in the evolution the first significant shock heating happens as the cores of the two stars collide. In all cases, temperatures in the shear layer between the two stellar cores can climb to $T \approx 30-40$ MeV. As the two cores continue to orbit and merge, the warm shocked material is redistributed toward the outside of the merger remnant, resulting in a warm envelope of $T \approx 10$ MeV surrounding the merged core which can reach temperatures of $T \approx 40$ MeV. We depict such states in the third column of Fig. 9, which roughly correspond to a few milliseconds after merger. Finally, in the rightmost panels of Fig. 9 we show the state of the merger remnants at a time corresponding to approximately 7–9 ms after merger. As the merger remnant settles, the final configurations approach a warm central object with temperatures $T \approx 10$ MeV. We find that in all cases this central object is surrounded by a distribution of matter which resembles a ring of hot material (with temperatures ranging $T \approx 40-50$ MeV), which becomes cooler with increased radial distance from the central configuration (dropping to temperatures $T \approx 10-20$ MeV). Neutrinos are expected to play a significant role not only in determining the composition of the matter during BNS mergers, but potentially also in determining the postmerger temperature evolution. At temperatures above $T \approx 10$ MeV, thermal neutrinos are expected to be produced, and their energy spectrum is affected by the local temperature and density, which in turn determines the locations of neutrino decoupling [36,110]. Simulations employing state-of-the-art, high-accuracy neutrino transport schemes suggest that neutrinos could remain sufficiently trapped in regions near and inside the remnant [111]. This results in a neutrino trapped gas that converts thermal energy in the fluid into radiation energy carried by the neutrinos [112], and could result in an over 10% change in the temperature of dense regions in the remnant, when compared to simulations that employ lower-accuracy neutrino transport schemes [111]. Changes in the thermal profile in turn affect the mass distribution of the remnant, which could have implications for the GW spectrum associated with the postmerger evolution [36]. Presently, `MIL` does not account for neutrinos and as such we cannot capture these effects in the simulations presented in this work. Future code development of `MIL` is expected to include neutrino transport beginning with neutrino leakage by interfacing with the `ZelmaniLeak` code [113] and moving forward from there to more accurate neutrino transport schemes. We leave a full code comparison between different GR (M)HD codes, including the effects of neutrino transport, to future work.

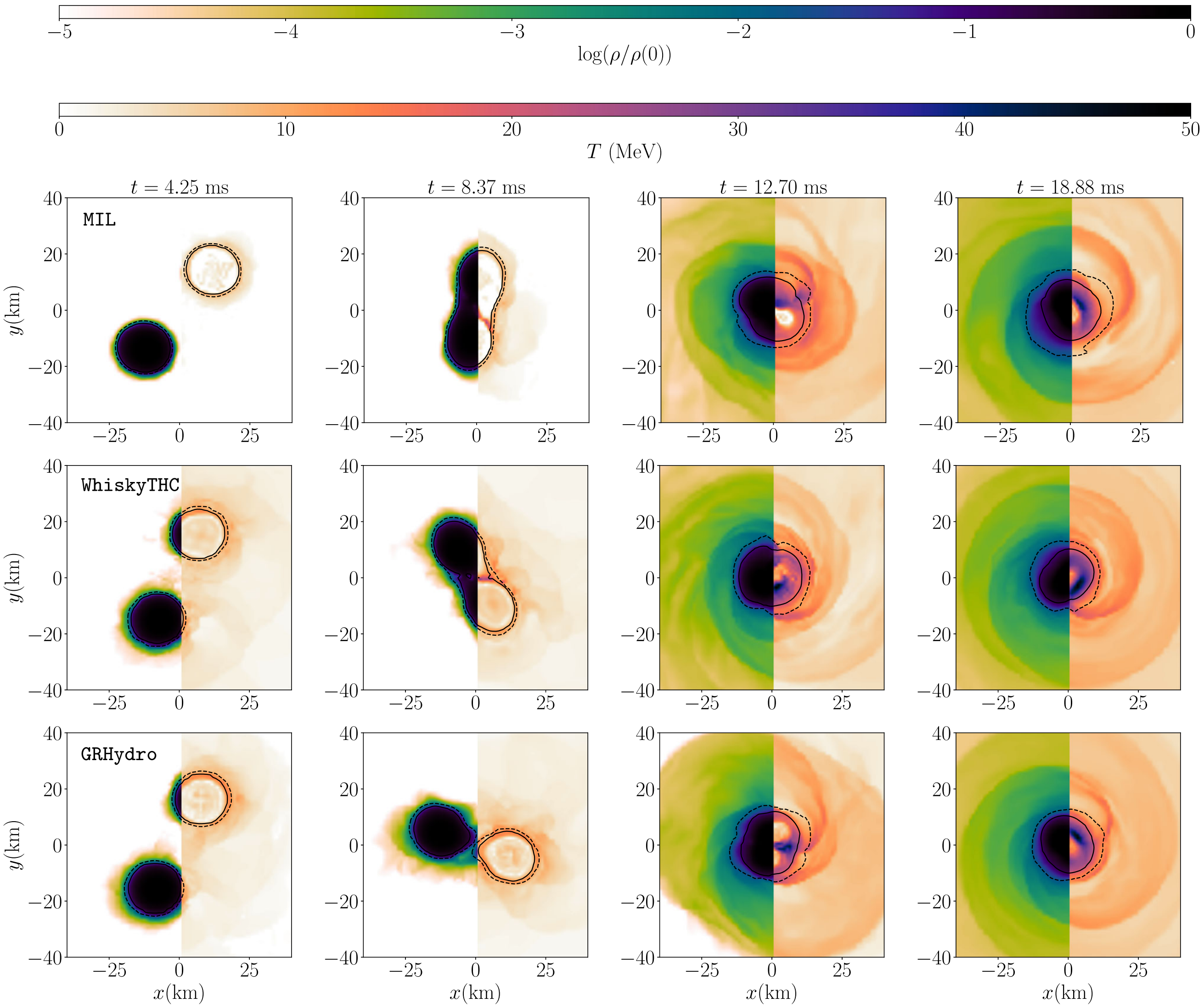

FIG. 9. Top row: Equatorial snapshots of the rest mass density (left half) and temperature (right half) for test LSBNS, in the case of evolution with the `MIL` code. From left to right we depict snapshots at times $t = 5.32$ ms, $t = 8.37$ ms, $t = 12.70$ ms, and $t = 18.88$ ms, respectively. The top (bottom) colorbar corresponds to the rest mass density (temperature). Middle row: same as top tow but in the case of evolution with the WhiskyTHC code. We outline regions above which the rest mass density takes on values of $\rho_b = 10^{13}\ \mathrm{g\,cm^{-3}}$ and $\rho_b = 10^{14}\ \mathrm{g\,cm^{-3}}$ using dashed and solid black contours, respectively. Bottom row: same as top row but in the case of evolution with the GRHydro code.

For a clear view of remnant properties, we consider rest mass density cutoffs which allow us to identify regions of the remnant. We identify the remnant itself and its core as regions in the postmerger environment above $\rho_b \geq 10^{13}\ \mathrm{g\,cm^{-3}}$ and $\rho_b \geq 10^{14}\ \mathrm{g\,cm^{-3}}$, respectively [10]. We highlight the remnant and its core using dashed and solid black lines in Fig. 9, respectively. We find that, in all cases, immediately following the merger a dense core is formed near the origin which typically extends to coordinate distance of ~15 km from the origin and is surrounded by lower density material which extends to coordinate distance ~20 km. This HMNS remnant is surrounded by a low-density, disklike structure.

The two key factors that determine the final fate of a BNS merger remnant are the remnant rest mass and the maximum rest mass of a uniformly rotating NS (i.e., the "supramassive" limit mass) M_{supra} for the given EOS. We note that, for hadronic EOSs like the ones considered in this work, the value of M_{supra} is closely related to the value of the maximum rest mass of a nonrotating star M_{TOV} and is expected to be about 20% larger $M_{\mathrm{supra}} \approx 1.2 M_{\mathrm{TOV}}$ [101,114–117] (although generally this mass increase can reach up to about 35% [118]); in the case of the LS220 EOS, $M_{\mathrm{supra}} = 2.83 M_{\odot}$, and the total system rest mass in test LSBNS is $M_b = 2.89 M_{\odot}$. One of the main sources of support against gravitational collapse in BNS merger remnants comes from the large amounts of differential rotation in the system (thermal support can also contribute significantly against gravitational collapse [36]). Differential rotation can greatly increase the maximum mass a star can support [119–121], but, as differential rotation is removed from the system (for example, via braking by magnetic fields [11,122]), the remnant will approach a uniformly rotating NS. If the total remnant mass is below $M_{\rm supra}$ for a given EOS, we may expect a long-lived NS remnant [123]. On the other hand, for systems with $M_{\rm tot} > M_{\rm supra}$, we may expect collapse. In all cases, we observe a transient HMNS remnant, as expected for this total binary mass and EOS. A key difference in the postmerger evolution for the simulations we consider is in the survival time of the HMNS remnant. In the `MIL` case, we find that the remnant does not collapse to a black hole (BH) by the end of evolution ($t_{\rm fin} \sim 40$ ms). On the contrary, the remnant collapses after $t_{\rm coll} \approx 21.6$ ms and $t_{\rm coll} \approx 25.2$ ms in the case of the WhiskyTHC and GRHydro evolutions, respectively (demonstrated by the sharp rise in the rest mass density in Fig. 7 and corroborated by the collapse of the lapse function α in both cases).

The remnant survival time $\tau_{\rm remnant}$ is an important quantity which significantly affects several observables associated with BNS mergers. For instance, neutrino irradiation from a metastable HMNS may affect ejecta properties, which would in turn affect the KN signal [123–125]. Additionally, $\tau_{\rm remnant}$ is expected to affect the nucleosynthesis in accretion disk outflows, with shorter HMNS lifetimes resulting in relatively low lanthanide and actinide abundances [126] due to the shorter neutrino irradiation times [124,127–129]. The significant differences in remnant survival times $\tau_{\rm remnant}$ exhibited by each code suggests that additional studies must be carried out which rely on the value of $\tau_{\rm remant}$ extracted from NR simulations. The remnant survival time may sensitively depend on the EOS [130,131], neutrino treatment (relevant for cooling and emergent viscous effects) [132–134], magnetic field treatment (relevant for angular momentum transport) [11,122,134,135], and (as suggested by the tests presented here) differences in the implementation of algorithmically equivalent numerical methods. Additional, in-depth studies remain to be carried to systematically understand all of the aforementioned effects on $\tau_{\rm remnant}$.

The survival time of the remnant can also play a role in the development of different fluid instabilities. Fluid instabilities that uniquely develop in the postmerger environment—such as the one-arm spiral instability and magneto-rotational instability (MRI)—would be arrested upon gravitational collapse to a black hole. For instance, a potential observable signature which arises from the one-arm spiral instability [136,137] is the long-lived, sustained powering of the $l = 2$, $m = 1$ GW mode [138], which typically has half the characteristic frequency of the initially dominant but decaying $l = 2$, $m = 2$ GW mode [138,139]. Over relatively long timescales, and as long as the postmerger massive neutron star remnant has not collapsed to a black hole, the one-arm spiral mode can continuously power the emission of the $l = 2$, $m = 1$ GW mode [138]; simulations which produce shorter remnant survival times may not allow the one-arm spiral instability to develop. In Fig. 10 we show the $l = 2$, $m = 2$ (using solid lines) and $l = 2$, $m = 1$ (using dashed lines) GW modes for test LSBNS. In all simulations we see increased emission of the $m = 1$ mode at a time close to merger, suggesting the potential early development of the one-arm spiral instability. In the WhiskyTHC and GRHydro evolutions, the amplitude of the $m = 1$ mode remains roughly constant until collapse, at which point the emission in this mode ceases. However, in the case of the `MIL` evolution, which has not collapsed by the end of simulation, GW emission in the $m = 1$ mode continues and is further amplified, while emission in the $m = 2$ mode decays. Due to the different remnant collapse times, the $m = 1$ mode is allowed to continue growing in the `MIL` evolution, but not in the WhiskyTHC and GRHydro evolutions.

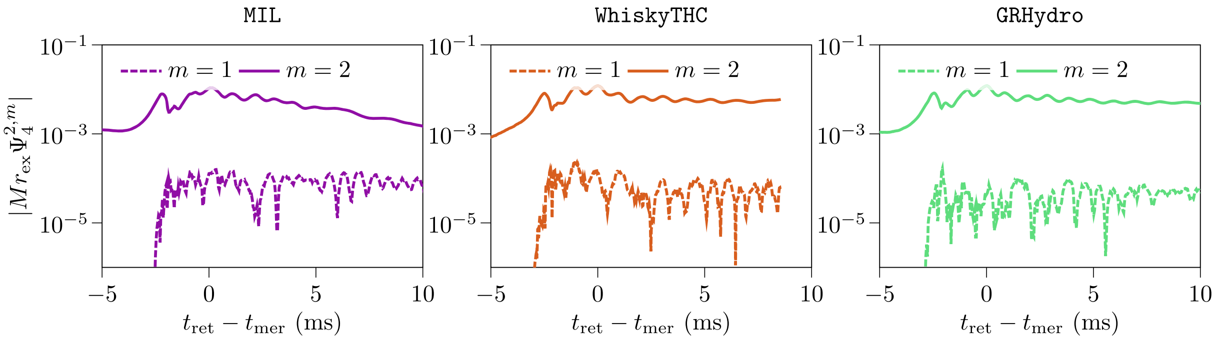

FIG. 10. Dominant coefficients of the spin-weighted decomposition of the Newman-Penrose scalar $\Psi_4^{2,m}$ (scaled with the extraction radius $r_{\rm ex}$ and ADM mass M) as a function of retarded time $t_{\rm ret}$ (shifted by the time of merger $t_{\rm mer}$) for the $l = 2$, $m = 1$ (dashed lines) and $l = 2$, $m = 2$ (solid lines) GW modes for test LSBNS. In the left, center, and right panels we show results for evolution with `MIL` (purple lines), WhiskyTHC (orange lines), and GRHydro (green lines), respectively.

Another set of instabilities that is relelvant in the postmerger environment, and are thereby affected by the remnant collapse time, are magnetic instabilities. In particular, the MRI can develop in the strongly differentially rotating environments following a BNS merger [9,140–144]. As long as differential rotation with an appropriate profile is sustained in a massive neutron star postmerger remnant, the MRI may develop and produce exponential magnetic field amplification (see [32,145] and references therein). To consider the amount of differential rotation in the postmerger remnants depicted in Fig. 9 we calculate the approximate angular velocity in the orbital plane $\Omega = v^{\phi}$, using Eq. (24). In Fig. 11 we show the radial profile of the angular velocity $\Omega(\varpi)$, averaged along the azimuthal direction and in time, for test LSBNS in the case of evolution with the `MIL` code (depicted using purple lines and contours) and the GRHydro code (depicted using green lines and contours). We time-average over a time window of approximately 2 ms ranging from $t \approx 22$ ms to $t \approx 24$ ms for both simulations. We focus on a fixed time window leading up to the gravitational collapse of the GRHydro model. We find that the remnants produced in each simulation still exhibit significant amounts of differential rotation well after the time of merger. The supramassive limit for EOS LS220 falls below the total system mass. Therefore, we may reasonably expect a delayed collapse once the configuration is no longer supported by differential rotation. The timescale on which differential rotation is removed in BNS merger remnants is typically $\sim\mathcal{O}(100\ \mathrm{ms})$ [12,146], which is significantly longer than the timescales probed by our dynamical simulations. However, such arguments assume that typically a large fraction of the angular momentum must be lost or transport for collapse to proceed, which is not generally the case. Therefore, although we may expect the same final remnants—a black hole—in all of our simulations for test LSBNS, the timescales over which this happens varies significantly across codes. We note that similar amounts of differential rotation are observed in the GRHydro case, suggesting that it was not the lack of differential rotation support which led to the collapse of the postmerger remnants in that case. As the massive NS remnants for both the `MIL` and GRHydro cases exhibit a significant amount of differential rotation until the time of collapse in the GRHydro case, we may expect the MRI to develop, if sufficiently resolved, in analogous simulations which include magnetic fields. However, our results suggest that the MRI could have a significantly longer time to develop in evolutions with the `MIL` code, due to a potentially significantly longer remnant lifetime than we can probe. On the other hand, the magnetic field can also reduce or prolong the lifetime of the remnant. The comparison cases presented in this section suggest that understanding the error introduced by using different numerical methods in NR simulations of BNS mergers is crucially important to the questions of magnetic field amplification mechanisms, fluid instabilities, and observables associated with such systems. Finally, we note that grid resolution can significantly impact the survival times of remnants (see [40] and references therein). We leave a full investigation of the effect of magnetic fields, grid resolution, and numerical schemes to future work.

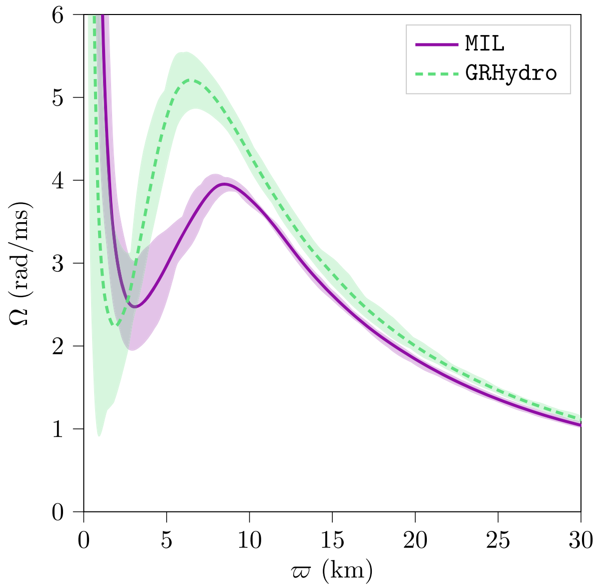

FIG. 11. Time-averaged, azimuthally-averaged angular velocity as a function of radial distance on the equatorial plane for the case where we use the `MIL` (purple solid line) and GRHydro (dashed green line) codes in test LSBNS. We time-average over a period of approximately 2 ms, ranging from simulation times of $t \approx 22$ ms to $t \approx 24$ ms. The purple and green contours depicts the range of values for the instantaneous angular velocity in the time window we consider for the `MIL` and GRHydro evolutions, respectively. The profiles indicate that the remnant is differentially rotating at the time leading up to gravitational collapse.

V. CONCLUSION AND OUTLOOK

In this work we have demonstrated the capabilities of the `MIL` code, a code which builds upon the current open source version of IllinoisGRMHD by including necessary changes for microphysical EOS compatibility. We outlined the algorithmic structure of IllinoisGRMHD and expanded on the additions that `MIL` provides. Specifically, we discussed the implementation of the electron fraction advection equation and the implementation of new state-of-the-art conservatives-to-primitives solvers within the code. The main products of these aforementioned additions are the `MIL` code itself and a new standalone thorn from primitives recovery, `ConservativeToPrimitive`, which is in principle able to interface with any GR(M)HD code that uses the Cactus infrastructure. Over a barrage of tests we established that the `MIL` code works as well as other publicly available GR(M)HD codes with similar capabilities. Crucially, for the systems considered, the `MIL` code agrees to a large extent with the Spritz code GRHydro, and WhiskyTHC codes.

Throughout the work we considered the numerical solutions provided by `MIL` for a wide range of astrophysical relevance. In particular, we consider several TOV and BNS configurations, all of which provide stringent tests of the `MIL` code. Where relevant, we compare with other open source GR(M)HD codes. We find general agreement and similar convergence properties to the GRHydro, Spritz, and WhiskyTHC codes, as well as to the current open source version of IllinoisGRMHD (referred to as `OIL` throughout the work). In some cases we find improved performance for `MIL` when compared to GRHydro (a property which is carried over from the `OIL` code as explored in [56]). We find general quantitative agreement between all codes considered in scenarios where the matter remains at low temperatures throughout the evolution. The largest differences observed between the results of each code appear during highly-dynamical scenarios such as during BNS mergers and in the immediate postmerger evolution. Although we find agreement in the postmerger peak frequency of the characteristic strain, other quantitative differences arise in these highly dynamical scenarios, likely due to the differences in the implementation of conservative-to-primitive algorithms. For example, some differences we highlight include: (1) the GRHydro and WhiskyTHC codes leading to higher temperatures than the `MIL` code in a BNS merger scenario, typically by up to $\Delta T \approx 10$ MeV; (2) merger times which differ by approximately 10% between codes; (3) significantly different survival times of the postmerger remnant; (4) large oscillations in the rest mass density for GRHydro when comparing to other codes, and general lack of quantitative agreement in the postmerger evolution; and (5) better preservation of the initial magnetic field structure and strength during the inspiral in the `OIL` and `MIL` codes when compared to the Spritz code.

Given that all codes considered allow some freedom in the flux treatment and reconstruction scheme, we ensure in all our tests that these two aspects of the codes have maximal algorithmic overlap. However, even if two codes employ the same algorithm, the implementation methods will be different in each case. These differences in algorithmic implementation may lead to the quantitative differences noted throughout the work. Despite the quantitative differences highlighted, we emphasize that all codes nevertheless produce qualitatively consistent (magneto)hydrodynamics evolutions and result in quantitatively similar gravitational waves. We note that the `MIL` code provides additional freedom over GRHydro, Spritz, and WhiskyTHC in the primitives inversion algorithm used during any given simulation through the thorn `ConservativeToPrimitive`, allowing for suitable comparisons with different codes and the ability to use the optimal conservative-to-primitive solver for the problem at hand. Additionally, the use of several solvers within `ConservativeToPrimitive` for a given simulation allows for a robust and efficient primitives recovery solution in the case of finite temperature EOSs, regardless of the variable space probed.

The updates included in `MIL` make possible the investigation of important, open questions surrounding BNS mergers. In particular, `MIL` will allow us to investigate the interplay between the effects of magnetic fields and microphysical and finite temperature EOSs in BNS mergers, the calculation of nucleosynthesis yields associated with BNS mergers, and provides the groundwork for the inclusion of neutrino effects within IllinoisGRMHD. Future work utilizing the full capabilities of `MIL` will investigate all of the aforementioned phenomena. For instance, the tests presented in Sec. IV F will be expanded upon for a full investigation of the interplay between magnetic field and realistic EOS effects. Additionally, because the `MIL` code provides a dynamical description of several fluid and thermodynamic quantities—including the rest mass density, temperature, specific entropy, and electron fraction—its output may be used with nuclear reaction network codes such as `SkyNet` [147] which only require such quantities as input. In turn, nuclear reaction networks provide insight into the nucleosynthesis rates associated with different stages of BNS mergers. Full GRMHD simulations of BNS mergers with magnetic field constrained transport have not received much attention in the context of nucleosynthesis, and this is now possible with `MIL`. Finally, because the electron fraction is known at all times during evolutions with `MIL`, the code may be expanded further to include neutrino effects. For instance, one may include neutrino leakage by interfacing `MIL` with the `ZelmaniLeak` thorn [113], or include more accurate descriptions of neutrino transport that move beyond leakage beyond leakage [26,148–151]. Such an extension to `MIL` will help elucidate the combined effect of magnetic and neutrino driven outflows after a BNS merger. Specifically, neutrino effects are expected to have a significant role on the electron fraction of the outflow [151]. The constrained transport magnetic field evolution and finite temperature description of `MIL` coupled with accurate neutrino transport could provide some of the most detailed insight into the physics following a BNS merger. In addition to the aforementioned changes, we also hope to implement a refluxing scheme in `MIL` [152,153], or to interface `MIL` with the `Refluxing` thorn[4] as has been done for the GRHydro and WhiskyTHC codes. Although conservation of ADM mass and momentum is achieved to within truncation error in IllinoisGRMHD [83], the implementation of a refluxing scheme could improve the conservation of relevant fluid quantities. We note that another recent extension to IllinoisGRMHD similar to ours (albeit using alternative methods and implementations) was considered in [55]. A detailed comparison of the codes

---

[4] https://bitbucket.org/dradice/refluxing/src/master/.

considered in this work along with the results presented in [55] is a crucial step toward understanding systematic errors in NR simulations of BNS mergers. We hope to consider these and other investigations in future work with the `MIL` code. We conclude the present work with a note about the availability of the `MIL` code. The source code for `MIL`, along with the supporting thorns, is publicly available at https://github.com/pilambdaepsilon/UAThorns.We also provide examples of parameter files, initial data, and equation of state files for all of the examples considered in this work.

ACKNOWLEDGMENTS

This work was supported by NSF Grants No. PHY-1912619 and No. PHY-2145421 to the University of Arizona. P. E. also acknowledges support from NSF Grant No. PHY-2020275 [Network for Neutrinos, Nuclear Astrophysics, and Symmetries (N3AS)]. During completion of this work, P. E. was in part supported by the Marshall Foundation Dissertation Scholarship. G. B. is supported by NASA Grant No. 80NSSC20K1542 to the University of Arizona. We wish to thank Bruno Giacomazzo, Federico Cipolletta, and David Radice for discussions. Simulations were in part performed at the `Comet` and `Expanse` clusters at SDSC, and the `Stampede2` cluster at Texas Advanced Computing Center (TACC) through Extreme Science and Engineering Discovery Environment (XSEDE) Grant No. TG-PHY190020. Simulations were also performed on the ELGato, Ocelote, and PuMA clusters at the University of Arizona. KUIBIT [64] is based on NumPy [154], SciPy [155], and H5PY [156].

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

[2] G. Raaijmakers *et al.*, Astrophys. J. **922**, 269 (2021).

[3] P. J. Easter, S. Ghonge, P. D. Lasky, A. R. Casey, J. A. Clark, F. H. Vivanco, and K. Chatziioannou, Phys. Rev. D **102**, 043011 (2020).

[4] C. Barbieri, O. S. Salafia, M. Colpi, G. Ghirlanda, and A. Perego, Astron. Astrophys. **654**, A12 (2021).

[5] M. Breschi, R. Gamba, and S. Bernuzzi, Phys. Rev. D **104**, 042001 (2021).

[6] M. Nicholl, B. Margalit, P. Schmidt, G. P. Smith, E. J. Ridley, and J. Nuttall, Mon. Not. R. Astron. Soc. **505**, 3016 (2021).

[7] V. Paschalidis, M. Ruiz, and S. L. Shapiro, Astrophys. J. Lett. **806**, L14 (2015).

[8] M. Ruiz, S. L. Shapiro, and A. Tsokaros, Phys. Rev. D **97**, 021501 (2018).

[9] M. Ruiz, R. N. Lang, V. Paschalidis, and S. L. Shapiro, Astrophys. J. **824**, L6 (2016).

[10] K. Kiuchi, K. Kyutoku, Y. Sekiguchi, and M. Shibata, Phys. Rev. D **97**, 124039 (2018).

[11] M. Shibata, S. Fujibayashi, and Y. Sekiguchi, Phys. Rev. D **103**, 043022 (2021).

[12] R. Ciolfi and J. V. Kalinani, Astrophys. J. Lett. **900**, L35 (2020).

[13] P. Mosta, D. Radice, R. Haas, E. Schnetter, and S. Bernuzzi, Astrophys. J. Lett. **901**, L37 (2020).

[14] R. Ciolfi, Mon. Not. R. Astron. Soc. **495**, L66 (2020).

[15] E. B. Abdikamalov, H. Dimmelmeier, L. Rezzolla, and J. C. Miller, Mon. Not. R. Astron. Soc. **394**, 52 (2009).

[16] A. Bauswein, T. W. Baumgarte, and H. T. Janka, Phys. Rev. Lett. **111**, 131101 (2013).

[17] C. Palenzuela, S. L. Liebling, D. Neilsen, L. Lehner, O. Caballero, E. O'Connor, and M. Anderson, Phys. Rev. D **92**, 044045 (2015).

[18] Y. Sekiguchi, K. Kiuchi, K. Kyutoku, and M. Shibata, Phys. Rev. D **91**, 064059 (2015).

[19] L. Lehner, S. L. Liebling, C. Palenzuela, O. L. Caballero, E. O'Connor, M. Anderson, and D. Neilsen, Classical Quantum Gravity **33**, 184002 (2016).

[20] D. Radice, A. Perego, K. Hotokezaka, S. A. Fromm, S. Bernuzzi, and L. F. Roberts, Astrophys. J. **869**, 130 (2018).

[21] E. R. Most, L. J. Papenfort, V. Dexheimer, M. Hanauske, S. Schramm, H. Stoecker, and L. Rezzolla, Phys. Rev. Lett. **122**, 061101 (2019).

[22] T. Dietrich, D. Radice, S. Bernuzzi, F. Zappa, A. Perego, B. Brügmann, S. V. Chaurasia, R. Dudi, W. Tichy, and M. Ujevic, Classical Quantum Gravity **35**, 24LT01 (2018).

[23] T. Vincent, F. Foucart, M. D. Duez, R. Haas, L. E. Kidder, H. P. Pfeiffer, and M. A. Scheel, Phys. Rev. D **101**, 044053 (2020).

[24] C. Raithel, V. Paschalidis, and F. Özel, Phys. Rev. D **104**, 063016 (2021).

[25] V. Nedora, S. Bernuzzi, D. Radice, A. Perego, A. Endrizzi, and N. Ortiz, Astrophys. J. Lett. **886**, L30 (2019).

[26] F. Foucart, M. D. Duez, F. Hebert, L. E. Kidder, H. P. Pfeiffer, and M. A. Scheel, Astrophys. J. **902**, L27 (2020).

[27] V. Paschalidis, Classical Quantum Gravity **34**, 084002 (2017).

[28] L. Baiotti and L. Rezzolla, Rep. Prog. Phys. **80**, 096901 (2017).

[29] M. D. Duez and Y. Zlochower, Rep. Prog. Phys. **82**, 016902 (2019).

[30] D. Radice, S. Bernuzzi, and A. Perego, Annu. Rev. Nucl. Part. Sci. **70**, 95 (2020).

[31] S. Bernuzzi, Gen. Relativ. Gravit. **52**, 108 (2020).

[32] R. Ciolfi, Gen. Relativ. Gravit. **52**, 59 (2020).

[33] D. Kasen, N. R. Badnell, and J. Barnes, Astrophys. J. **774**, 25 (2013).

[34] J. Barnes and D. Kasen, Astrophys. J. **775**, 18 (2013).

[35] L. Rezzolla, E. R. Most, and L. R. Weih, Astrophys. J. **852**, L25 (2018).
[36] C. Raithel, V. Paschalidis, and F. Ozel, Phys. Rev. D **104**, 063016 (2021).
[37] R. D. Blandford and R. L. Znajek, Mon. Not. R. Astron. Soc. **179**, 433 (1977).
[38] M. H. P. M. Van Putten, Science **284**, 115 (1999).
[39] N. Bucciantini, B. D. Metzger, T. A. Thompson, and E. Quataert, Mon. Not. R. Astron. Soc. **419**, 1537 (2012).
[40] M. Ruiz, R. N. Lang, V. Paschalidis, and S. L. Shapiro, Astrophys. J. **824**, L6 (2016).
[41] D. M. Siegel, R. Ciolfi, and L. Rezzolla, Astrophys. J. Lett. **785**, L6 (2014).
[42] R. Fernandez, A. Tchekhovskoy, E. Quataert, F. Foucart, and D. Kasen, Mon. Not. R. Astron. Soc. **482**, 3373 (2019).
[43] B. D. Metzger, T. A. Thompson, and E. Quataert, Astrophys. J. **856**, 101 (2018).
[44] S. Wanajo, Y. Sekiguchi, N. Nishimura, K. Kiuchi, K. Kyutoku, and M. Shibata, Astrophys. J. Lett. **789**, L39 (2014).
[45] A. Perego, D. Radice, and S. Bernuzzi, Astrophys. J. Lett. **850**, L37 (2017).
[46] D. Radice, A. Perego, K. Hotokezaka, S. Bernuzzi, S. A. Fromm, and L. F. Roberts, Astrophys. J. Lett. **869**, L35 (2018).
[47] L. Sun, M. Ruiz, S. L. Shapiro, and A. Tsokaros, Phys. Rev. D **105**, 104028 (2022).
[48] M. Shibata, S. Fujibayashi, K. Hotokezaka, K. Kiuchi, K. Kyutoku, Y. Sekiguchi, and M. Tanaka, Phys. Rev. D **96**, 123012 (2017).
[49] L. Bovard, D. Martin, F. Guercilena, A. Arcones, L. Rezzolla, and O. Korobkin, Phys. Rev. D **96**, 124005 (2017).
[50] D. Radice, A. Perego, S. Bernuzzi, and B. Zhang, Mon. Not. R. Astron. Soc. **481**, 3670 (2018).
[51] F. Foucart, E. O'Connor, L. Roberts, L. E. Kidder, H. P. Pfeiffer, and M. A. Scheel, Phys. Rev. D **94**, 123016 (2016).
[52] D. Radice, A. Perego, K. Hotokezaka, S. Bernuzzi, S. A. Fromm, and L. F. Roberts, Astrophys. J. **869**, L35 (2018).
[53] S. Bernuzzi *et al.*, Mon. Not. R. Astron. Soc. **497**, 1488 (2020).
[54] M. Pürrer and C.-J. Haster, Phys. Rev. Res. **2**, 023151 (2020).
[55] L. R. Werneck *et al.*, Phys. Rev. D **107**, 044037 (2023).
[56] Z. B. Etienne, V. Paschalidis, R. Haas, P. Moesta, and S. L. Shapiro, Classical Quantum Gravity **32**, 175009 (2015).
[57] P. Mosta, B. C. Mundim, J. A. Faber, R. Haas, S. C. Noble, T. Bode, F. Loffler, C. D. Ott, C. Reisswig, and E. Schnetter, Classical Quantum Gravity **31**, 015005 (2014).
[58] B. Giacomazzo, F. Cipolletta, J. Kalinani, R. Ciolfi, L. Sala, B. Giudici, and E. Giangrandi, The Spritz code (1.0.0) (Zenodo, 2020), 10.5281/zenodo.3689752.
[59] D. Radice and L. Rezzolla, Astron. Astrophys. **547**, A26 (2012).
[60] D. Radice, L. Rezzolla, and F. Galeazzi, Mon. Not. R. Astron. Soc. **437**, L46 (2013).
[61] D. Radice, L. Rezzolla, and F. Galeazzi, Classical Quantum Gravity **31**, 075012 (2014).
[62] F. Loffler, J. Faber, E. Bentivegna, T. Bode, P. Diener, R. Haas, I. Hinder, B. C. Mundim, C. D. Ott, E. Schnetter, G. Allen, M. Campanelli, and P. Laguna, Classical Quantum Gravity **29**, 115001 (2012).
[63] Z. Etienne *et al.*, The Einstein Toolkit (2021), to find out more, visit http://einsteintoolkit.org.
[64] G. Bozzola, J. Open Source Software **6**, 3099 (2021).
[65] T. Nakamura, K. Oohara, and Y. Kojima, Prog. Theor. Phys. Suppl. **90**, 1 (1987).
[66] M. Shibata and T. Nakamura, Phys. Rev. D **52**, 5428 (1995).
[67] T. W. Baumgarte and S. L. Shapiro, Phys. Rev. D **59**, 024007 (1999).
[68] M. Alcubierre, *Introduction to 3+1 Numerical Relativity* (Oxford University Press, UK, 2008).
[69] C. Bona and C. Palenzuela-Luque, *Elements of Numerical Relativity* (Springer, Berlin, Heidelberg, 2005), Vol. 673.
[70] T. W. Baumgarte and S. L. Shapiro, *Numerical Relativity: Solving Einstein's Equations on the Computer* (Cambridge University Press, Cambridge, England, 2010).
[71] M. Shibata, *Numerical Relativity* (World Scientific Publishing Co., Singapore, 2016).
[72] J. A. Font, Living Rev. Relativity **11**, 7 (2008).
[73] L. Rezzolla and O. Zanotti, *Relativistic Hydrodynamics* (Oxford University Press, Oxford, 2013).
[74] F. Banyuls, J. A. Font, J. M. A. Ibanez, J. M. A. Marti, and J. A. Miralles, Astrophys. J. **476**, 221 (1997).
[75] L. Anton, O. Zanotti, J. A. Miralles, J. M. Marti, J. M. Ibanez, J. A. Font, and J. A. Pons, Astrophys. J. **637**, 296 (2006).
[76] B. D. Farris, R. Gold, V. Paschalidis, Z. B. Etienne, and S. L. Shapiro, Phys. Rev. Lett. **109**, 221102 (2012).
[77] E. R. Most and C. A. Raithel, Phys. Rev. D **104**, 124012 (2021).
[78] C. A. Raithel and V. Paschalidis, Phys. Rev. D **106**, 023015 (2022).
[79] C. A. Raithel, P. Espino, and V. Paschalidis, Mon. Not. R. Astron. Soc. **516**, 4792 (2022).
[80] J. Fields, A. Prakash, M. Breschi, D. Radice, S. Bernuzzi, and A. d. S. Schneider, arXiv:2302.11359.
[81] P. Colella and P. R. Woodward, J. Comput. Phys. **54**, 174 (1984).
[82] A. Harten, P. D. Lax, and B. V. Leer, SIAM Rev. **25**, 35 (1983).
[83] Z. B. Etienne, Y. T. Liu, and S. L. Shapiro, Phys. Rev. D **82**, 084031 (2010).
[84] Z. B. Etienne, Y. T. Liu, V. Paschalidis, and S. L. Shapiro, Phys. Rev. D **85**, 064029 (2012).
[85] C. F. Gammie, J. C. McKinney, and G. Toth, Astrophys. J. **589**, 444 (2003).
[86] S. C. Noble, C. F. Gammie, J. C. McKinney, and L. Del Zanna, Astrophys. J. **641**, 626 (2006).
[87] W. I. Newman and N. D. Hamlin, SIAM J. Sci. Comput. **36**, B661 (2014).
[88] D. M. Siegel, P. Mosta, D. Desai, and S. Wu, Astrophys. J. **859**, 71 (2018).
[89] W. Kastaun, J. V. Kalinani, and R. Ciolfi, Phys. Rev. D **103**, 023018 (2021).
[90] E. R. Most, L. J. Papenfort, and L. Rezzolla, Mon. Not. R. Astron. Soc. **490**, 3588 (2019).

[91] J. V. Kalinani, R. Ciolfi, W. Kastaun, B. Giacomazzo, F. Cipolletta, and L. Ennoggi, Phys. Rev. D **105**, 103031 (2022).
[92] E. O'Connor and C. D. Ott, Classical Quantum Gravity **27**, 114103 (2010).
[93] J. M. Lattimer and D. F. Swesty, Nucl. Phys. **A535**, 331 (1991).
[94] D. Brown, P. Diener, O. Sarbach, E. Schnetter, and M. Tiglio, Phys. Rev. D **79**, 044023 (2009).
[95] C. Reisswig, C. D. Ott, U. Sperhake, and E. Schnetter, Phys. Rev. D **83**, 064008 (2011).
[96] C. Bona, J. Masso, E. Seidel, and J. Stela, Phys. Rev. Lett. **75**, 600 (1995).
[97] M. Alcubierre, B. Bruegmann, P. Diener, M. Koppitz, D. Pollney, E. Seidel, and R. Takahashi, Phys. Rev. D **67**, 084023 (2003).
[98] B. Einfeldt, SIAM J. Numer. Anal. **25**, 294 (1988).
[99] E. Newman and R. Penrose, J. Math. Phys. (N.Y.) **3**, 566 (1962).
[100] R. Penrose, Phys. Rev. Lett. **10**, 66 (1963).
[101] G. Cook, S. Shapiro, and S. Teukolsky, Astrophys. J. **422**, 227 (1994).
[102] N. Stergioulas and J. L. Friedman, Astrophys. J. **444**, 306 (1995).
[103] T. Nozawa, N. Stergioulas, E. Gourgoulhon, and Y. Eriguchi, Astron. Astrophys. Suppl. Ser. **132**, 431 (1998).
[104] E. Gourgoulhon, P. Grandclement, J.-A. Marck, J. Novak, and K. Taniguchi, LORENE: Spectral methods differential equations solver (2016), ascl:1608.018.
[105] V. Paschalidis, Z. B. Etienne, and S. L. Shapiro, Phys. Rev. D **88**, 021504 (2013).
[106] A. Suresh and H. Huynh, J. Comput. Phys. **136**, 83 (1997).
[107] Y. Sekiguchi, K. Kiuchi, K. Kyutoku, and M. Shibata, Prog. Theor. Exp. Phys. **2012**, 01A304 (2012).
[108] K. Hayashi, S. Fujibayashi, K. Kiuchi, K. Kyutoku, Y. Sekiguchi, and M. Shibata, Phys. Rev. D **106**, 023008 (2022).
[109] C. J. Moore, R. H. Cole, and C. P. L. Berry, Classical Quantum Gravity **32**, 015014 (2015).
[110] M. Cusinato, F. M. Guercilena, A. Perego, D. Logoteta, D. Radice, S. Bernuzzi, and S. Ansoldi, arXiv:2111.13005.
[111] F. Zappa, S. Bernuzzi, D. Radice, and A. Perego, arXiv:2210.11491.
[112] A. Perego, S. Bernuzzi, and D. Radice, Eur. Phys. J. A **55**, 124 (2019).
[113] C. D. Ott, E. Schnetter, A. Burrows, E. Livne, E. O'Connor, and F. Loffler, J. Phys. Conf. Ser. **180**, 012022 (2009).
[114] G. Cook, S. Shapiro, and S. Teukolsky, Astrophys. J. **424**, 823 (1994).
[115] J.-P. Lasota, P. Haensel, and M. A. Abramowicz, Astrophys. J. **456**, 300 (1996).
[116] I. A. Morrison, T. W. Baumgarte, and S. L. Shapiro, Astrophys. J. **610**, 941 (2004).
[117] C. Breu and L. Rezzolla, Mon. Not. R. Astron. Soc. **459**, 646 (2016).
[118] G. Bozzola, P. L. Espino, C. D. Lewin, and V. Paschalidis, Eur. Phys. J. A **55**, 149 (2019).
[119] M. Ansorg, D. Gondek-Rosinska, and L. Villain, Mon. Not. R. Astron. Soc. **396**, 2359 (2009).
[120] G. Bozzola, N. Stergioulas, and A. Bauswein, Mon. Not. R. Astron. Soc. **474**, 3557 (2018).
[121] P. L. Espino and V. Paschalidis, Phys. Rev. D **99**, 083017 (2019).
[122] S. L. Shapiro, Astrophys. J. **544**, 397 (2000).
[123] B. Margalit and B. D. Metzger, Astrophys. J. Lett. **850**, L19 (2017).
[124] B. D. Metzger and R. Fernandez, Mon. Not. R. Astron. Soc. **441**, 3444 (2014).
[125] E. M. Holmbeck, A. Frebel, G. C. McLaughlin, R. Surman, R. Fernandez, B. D. Metzger, M. R. Mumpower, and T. M. Sprouse, Astrophys. J. **909**, 21 (2021).
[126] J. Lippuner, R. Fernandez, L. F. Roberts, F. Foucart, D. Kasen, B. D. Metzger, and C. D. Ott, Mon. Not. R. Astron. Soc. **472**, 904 (2017).
[127] D. Martin, A. Perego, A. Arcones, O. Korobkin, and F.-K. Thielemann, Proc. Sci. NICXIII2015 (**2015**) 120 [arXiv:1509.07628].
[128] D. Martin, A. Perego, A. Arcones, F.-K. Thielemann, O. Korobkin, and S. Rosswog, Astrophys. J. **813**, 2 (2015).
[129] J. M. Miller, T. M. Sprouse, C. L. Fryer, B. R. Ryan, J. C. Dolence, M. R. Mumpower, and R. Surman, Astrophys. J. **902**, 66 (2020).
[130] K. Hotokezaka, K. Kiuchi, K. Kyutoku, T. Muranushi, Y.-i. Sekiguchi, M. Shibata, and K. Taniguchi, Phys. Rev. D **88**, 044026 (2013).
[131] M. Lucca and L. Sagunski, J. High Energy Astrophys. **27**, 33 (2020).
[132] M. D. Duez, Y. T. Liu, S. L. Shapiro, and B. C. Stephens, Phys. Rev. D **69**, 104030 (2004).
[133] M. Shibata and K. Kiuchi, Phys. Rev. D **95**, 123003 (2017).
[134] C. Palenzuela, S. Liebling, and B. Miñano, Phys. Rev. D **105**, 103020 (2022).
[135] D. Radice, Astrophys. J. Lett. **838**, L2 (2017).
[136] J. M. Centrella, K. C. B. New, L. L. Lowe, and J. D. Brown, Astrophys. J. Lett. **550**, L193 (2001).
[137] M. Saijo, T. W. Baumgarte, and S. L. Shapiro, Astrophys. J. **595**, 352 (2003).
[138] V. Paschalidis, W. E. East, F. Pretorius, and S. L. Shapiro, Phys. Rev. D **92**, 121502 (2015).
[139] D. Radice, S. Bernuzzi, and C. D. Ott, Phys. Rev. D **94**, 064011 (2016).
[140] D. M. Siegel, R. Ciolfi, A. I. Harte, and L. Rezzolla, Phys. Rev. D **87**, 121302 (2013).
[141] K. Kiuchi, K. Kyutoku, Y. Sekiguchi, M. Shibata, and T. Wada, Phys. Rev. D **90**, 041502 (2014).
[142] J. D. Kaplan, C. D. Ott, E. P. O'Connor, K. Kiuchi, L. Roberts, and M. Duez, Astrophys. J. **790**, 19 (2014).
[143] J. Guilet, A. Bauswein, O. Just, and H.-T. Janka, Mon. Not. R. Astron. Soc. **471**, 1879 (2017).
[144] V. Paschalidis, Classical Quantum Gravity **34**, 084002 (2017).
[145] V. Paschalidis and N. Stergioulas, Living Rev. Relativity **20**, 7 (2017).
[146] R. Ciolfi, W. Kastaun, J. V. Kalinani, and B. Giacomazzo, Phys. Rev. D **100**, 023005 (2019).
[147] J. Lippuner and L. F. Roberts, Astrophys. J. Suppl. Ser. **233**, 18 (2017).

[148] V. Nedora, S. Bernuzzi, D. Radice, B. Daszuta, A. Endrizzi, A. Perego, A. Prakash, M. Safarzadeh, F. Schianchi, and D. Logoteta, Astrophys. J. **906**, 98 (2021).

[149] V. Nedora, F. Schianchi, S. Bernuzzi, D. Radice, B. Daszuta, A. Endrizzi, A. Perego, A. Prakash, and F. Zappa, Classical Quantum Gravity **39**, 015008 (2022).

[150] L. R. Weih, A. Gabbana, D. Simeoni, L. Rezzolla, S. Succi, and R. Tripiccione, Mon. Not. R. Astron. Soc. **498**, 3374 (2020).

[151] D. Radice, S. Bernuzzi, A. Perego, and R. Haas, Mon. Not. R. Astron. Soc. **512**, 1499 (2022).

[152] A. Mignone, C. Zanni, P. Tzeferacos, B. van Straalen, P. Colella, and G. Bodo, Astrophys. J. Suppl. Ser. **198**, 7 (2012).

[153] D. C. Collins, H. Xu, M. L. Norman, H. Li, and S. Li, Astrophys. J. Suppl. Ser. **186**, 308 (2010).

[154] C. R. Harris *et al.*, Nature (London) **585**, 357 (2020).

[155] P. Virtanen *et al.*, Nat. Methods **17**, 261 (2020).

[156] A. Collette, *Python and HDF5* (O'Reilly, Sebastopol, California, 2013).