

PAPER

Extremely high-order convergence in simulations of relativistic stars

To cite this article: John Ryan Westernacher-Schneider 2021 *Class. Quantum Grav.* **38** 145003

View the [article online](#) for updates and enhancements.

You may also like

- [Active current gating in electrically biased conical nanopores](#)
Samuel Bearden, Erik Simpanen and Guigen Zhang
- [Imagining a Healthy Future: Cross-Disciplinary Design for Sustainable Community Development in Cange, Haiti](#)
Dustin Albright, Ufuk Ersoy, David Vaughn et al.
- [STELLAR NUCLEOSYNTHESIS IN THE HYADES OPEN CLUSTER](#)
Simon C. Schuler, Jeremy R. King and Lih-Sin The

Extremely high-order convergence in simulations of relativistic stars

John Ryan Westernacher-Schneider^{1,2,*} 

Department of Physics and Astronomy, Clemson University, 118 Kinard Laboratory,
Clemson, SC 29634-0978, United States of America

Department of Astronomy/Steward Observatory, The University of Arizona, 933 N.
Cherry Ave, Tucson, AZ 85721, United States of America

E-mail: wester5@clemson.edu

Received 27 January 2021, revised 26 April 2021

Accepted for publication 17 May 2021

Published 15 June 2021



CrossMark

Abstract

We provide a road towards obtaining gravitational waveforms from inspiraling material binaries with an accuracy viable for third-generation gravitational wave detectors, without necessarily advancing computational hardware or massively-parallel software infrastructure. We demonstrate a proof-of-principle $1 + 1$ -dimensional numerical implementation that exhibits up to 7th-order convergence for highly dynamic barotropic stars in curved spacetime, and numerical errors up to 6 orders of magnitude smaller than a standard method. Aside from errors associated with overshoot in high-order extrapolation from the interior of the star to the surface, there are no obvious fundamental obstacles to obtaining convergence of even higher order. The implementation uses a novel surface-tracking method, where the surface is evolved and high-order accurate boundary conditions are imposed there. Computational memory does not need to be allocated to fluid variables in the vacuum region of spacetime. We anticipate the application of this new method to full $3 + 1$ -dimensional simulations of the inspiral phase of compact binary systems with at least one material body. The additional challenge of a deformable surface must be addressed in multiple spatial dimensions, but it is also an opportunity to input more precise surface tension physics.

Keywords: neutron star, high-order convergence, stellar surface, vacuum regions, artificial atmosphere, 3G era, relativistic star

 Supplementary material for this article is available [online](#)

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

* Author to whom any correspondence should be addressed.

1. Introduction and motivation

Gravitational wave astronomy has recently exploded with activity [1–5]. The signal-to-noise ratio in current gravitational wave data [6, 7] does not yet challenge the accuracy of theoretical models of the signals coming from perturbation theory, phenomenology, and numerical relativity [8–13]. Therefore, those models are used to inform interpretations of the data [14, 15]. However, with future third-generation detectors [16, 17] or current detectors at design-sensitivity, the signal-to-noise ratio is expected to challenge current models [18–20]. It is therefore necessary to make substantial advancements in the accuracy of theoretical models, including numerical relativity simulations. Advancements are being pursued in perturbative calculations [21–32] as well as numerical simulations. Some efforts in the latter category involve innovations and optimizations of hardware [33–35], parallel-computing software [36–40], numerical methods [41–53], and novel physics formulations [54, 55].

In this work, we take advantage of an existing arbitrage opportunity to substantially increase the accuracy of simulations of relativistic stars with a negligible increase in computational cost. The price paid is a modest increase in code complexity. The opportunity exists because current, widely applied numerical methods for dealing with stellar surfaces and vacuum regions are an unnecessary and substantial source of numerical error and degradation of convergence rate. The standard method consists of filling the vacuum regions of spacetime with an artificial low density fluid atmosphere, maintaining it at every time step, and then evolving the fluid equations everywhere (including across stellar surfaces). Often a low-order accurate solver is used purposely in the vicinity of the surface (e.g. [56, 57]). Strong attempts to achieve high-order convergence in binary simulations yield convergence orders for the inspiral gravitational waveform of 2.5–3 [44, 47].¹ One recent advancement uses negative pressure artificial atmospheres which can be in hydrostatic equilibrium [54]. A recently demonstrated strategy to use an artificial atmosphere-style scheme with zero density, while importantly lowering errors associated with mass conservation and ejecta, does not substantially improve the bulk motion [53]. High-order convergence in simulations of isolated fluid bodies has been elusive, and is a significant obstacle for simulating such systems with an accuracy viable for future gravitational wave observation campaigns.

Our objective in this work is to improve the treatment of relativistic stellar surfaces in 1 spatial dimension, which is a necessary step towards high-order convergence in full $3 + 1$ dimensional numerical relativity simulations. Our strategy is to evolve the surface position and velocity, and impose free surface boundary conditions there. This moving boundary condition can be imposed at arbitrarily high order, in principle. *We demonstrate up to 7th-order accuracy formally for the fluid variables in a $1+1$ -dimensional toy model involving highly dynamic stars on curved spacetime.* Using this ‘surface-tracking’ method, we also demonstrate substantial improvements in the absolute level of numerical error even when the time-stepping algorithm is a standard 3rd-order one. These demonstrations are presented as a proof-of-principle for the method, with extensions to multiple spatial dimensions left for future work. The multi-dimensional moving boundary (the stellar surface) will also be deformable, which presents an additional technical challenge as well as an opportunity to input surface tension physics. The machinery for dealing with *rigid* moving boundaries in multiple dimensions was already presented in reference [59]. We expect that non-oscillatory interpolators will have to be used

¹ Note that reference [44] describes oscillations of equilibrium stars in simulations as being triggered by atmosphere effects. However, strictly speaking they are initially triggered by the use of an ill-balanced numerical scheme, whereas atmosphere effects sustain the oscillations subsequently. See e.g. reference [58] for an example of a well-balanced scheme.

in multiple dimensions in order to prevent excessive unphysical wrinkling of the surface, and this may be where surface tension physics enters.

An important design goal is that the surface-tracking method can be used with any interior scheme, which enables adoption of the method in all existing production codes. We anticipate that the method will work with any grid structure, although our demonstration is on a Cartesian one. Furthermore, there is absolutely no need to allocate computational memory to fluid variables in the vacuum regions, although taking advantage of this would increase the data structure complexity in the code.

Our surface boundary treatment is based on a history of development that in the past used the inverse of the original Lax–Wendroff procedure [60]. The Lax–Wendroff procedure involves writing time derivatives in terms of spatial derivatives using the equations of motion, and can be used to derive the one-step 2nd-order accurate Lax–Wendroff numerical method. Near computational boundaries, where spatial derivative stencils extend beyond the domain, it is advantageous to do the reverse: replace spatial derivatives with time derivatives. In the *inverse* Lax–Wendroff procedure, spatial derivatives normal to the boundary are replaced by time derivatives and tangential spatial derivatives using the equations of motion. The idea of using the Lax–Wendroff procedure in reverse was proposed in references [61, 62] for the analysis of numerical boundary conditions, and was extended to arbitrarily shaped boundaries in reference [63]. Inspired by reference [64], extensions to nonlinear conservation laws [59] and compressible non-relativistic inviscid fluids [65] were achieved. Significant simplification of the boundary treatment was presented in reference [66]. Linear stability of the inverse Lax–Wendroff treatment of boundaries has been established for up to 13th-order accurate methods [67, 68]. See the review in reference [69].

Schemes for tracking a fluid–vacuum interface have previously been reported. A numerical treatment of the vacuum Riemann problem was presented in references [70, 71]. Such work was built upon by reference [72], allowing the fluid–vacuum interface to contract, thereby enabling the simulation of oscillating stars with a surface-tracking method. This improved mass conservation relative to an artificial atmosphere scheme, but there was no attempt to render the surface treatment as high-order accurate.

In this work, we extend the work of reference [65] to relativistic inviscid fluids in $1 + 1$ -dimensional curved spacetime, and to boundary conditions appropriate for fluid–vacuum interfaces such as stellar surfaces. The treatment of a fluid–vacuum interface requires novel strategies since extrapolations into the vacuum region would in general yield unphysical values of the fluid variables. We find that we can implement the efficiency improvement of reference [66] even more aggressively, whereby we eliminate the inverse Lax–Wendroff procedure entirely. This yields an exceptionally simple scheme. The reason this is possible is likely due to the fact that stellar surfaces are characteristic surfaces (at least very nearly²), so one can be more relaxed regarding the treatment of ingoing versus outgoing information³. Extension to rigid surfaces in multiple spatial dimensions should follow straightforwardly using the general recipe described in reference [65], but as mentioned earlier, treating deformable surfaces will require additional development likely to involve non-oscillatory interpolations of the surface position and velocity. For technical reasons described in the body of the text (see section 3), we use a ‘liquid’ equation of state whereby the rest mass density takes on a small positive value when the pressure vanishes (thus the density is finite at the stellar surface). We consider only

² We use a liquid equation of state in this work, whereby the sound speed does not vanish at the stellar surface (but instead becomes very small).

³ If working with an equation of state such that the sound speed is non-negligible at the surface, one can expect to have to use the inverse Lax–Wendroff procedure for the ingoing first derivatives [66].

barotropic fluids, since we anticipate the future application of our methods to the inspiral phase of compact binary mergers involving at least one material body. The inspiral phase is expected to be well-approximated by barotropic fluid flow [73, 74]. The extremeness of the stellar evolutions that we study far exceeds what one expects during the inspiral phase of a binary. In a binary inspiral, stars are subjected to smooth, long-wavelength tidal deformations rather than the volatile sloshing motion that we consider in this work.

It is worth mentioning that the well-posedness of the fluid–vacuum interface is a difficult and unsettled mathematical problem. Some find benefits to using a liquid equation of state (e.g. reference [75]). For general reading on the problem, see for example references [75–80] and references therein. Numerically, we find that the free surface problem is quite stable when using a standard formulation of hydrodynamics.

In section 2, we describe our toy model environment in which our new method is demonstrated, including the characteristic structure of the system in section 2.1. We describe the boundary conditions used at the fluid–vacuum interfaces in section 3, as well our choice of liquid equation of state. The surface-tracking algorithm is described in detail in section 4. Implementation details for the interior scheme are presented in section 5. Numerical results are presented in section 6. We summarize and give some thoughts on future directions in section 7. Although it is not used, we present the inverse Lax–Wendroff procedure for first derivatives in appendix B, as well as the constraint on first derivatives that one would have to use in the inverse Lax–Wendroff procedure in appendix A. Various failure modes that we have encountered for the surface-tracking method, as well as some possible policies for dealing with them, are presented in appendix C (although we have found that the invocation of such error handling policies usually indicates a bug in the implementation, except for very extreme stellar evolutions).

Throughout, we use the negative-time metric signature $(-, +, +, \dots)$. Spacetime indices are denoted with letters at the beginning of the alphabet $\{a, b, c, \dots\}$. Spatial indices are denoted with letters beginning in the middle of the alphabet $\{I, j, k, \dots\}$. We use units in which $G = c = 1$.

2. Toy model

In this section, we describe a toy model which we use to demonstrate the surface-tracking methods. It consists of an isolated fluid body living in 1 spatial dimension, under the influence of a time-independent spacetime curvature which acts to maintain the body’s compactness. The spacetime is prescribed, non-dynamical, and not derived from any underlying theory. The toy model contains the minimal set of ingredients required for demonstrating high-order accurate simulations of a relativistic star in curved spacetime.

We consider a $1 + 1$ -dimensional Cartesian ‘star’ in a frozen spacetime with metric

$$ds^2 = -\alpha(x)^2 dt^2 + dx^2. \quad (2.1)$$

The spacetime is periodic in x with a length $L = 10$, and the lapse function is prescribed to be

$$\alpha(x) = \frac{2}{3} \left(1 - \frac{1}{2} \cos \left\{ \frac{2\pi}{L}(x - L/2) \right\} \right). \quad (2.2)$$

This lapse function goes to 1 at $x = 0$ and $x = L$, and reaches a minimum value of $1/3$ at $x = L/2$, thereby providing a confining gravitational potential centered at the position $x = L/2$. Note that this is a curved spacetime, since it has non-vanishing Ricci scalar $R = -(2/\alpha)\partial_x^2\alpha$. We intend that this lapse function only mimics a confining gravitational potential; it is not

derived from any underlying theory. In the language of a $1 + 1$ -dimensional decomposition of spacetime, the shift vanishes $\beta^x = 0$ and the spatial metric is Euclidean $\gamma_{xx} = \delta_{xx}$, thus in particular the spatial metric determinant is $\sqrt{\gamma} = 1$. This Cartesian $1 + 1$ -dimensional setup is more representative of a $3 + 1$ -dimensional Cartesian setup than a $1 + 1$ -dimensional spherically-symmetric one, because of the absence of singular coordinate behavior at the origin. Such singular coordinate behavior in spherical symmetry requires numerical care, and is a distraction from the main proof-of-principle we present in this work. These reasons motivate our choice of toy model spacetime equation (2.1).

We consider two formulations of barotropic fluid dynamics. One resembles the *Valencia* formulation [81, 82] because it is based on the equations of mass and momentum conservation (we simply refer to it as the Valencia formulation). The other formulation is based on mass conservation and the Hamiltonian Euler equation (we refer to it as the *Hamiltonian* formulation). The Hamiltonian Euler equation evolves the *canonical momentum* (rather than the momentum density). Both formulations are analytically equivalent for smooth solutions (in the absence vacuum boundaries [55]), but have different properties when discretized numerically. Both formulations use the conservation of rest mass density,

$$0 = \nabla_a (\rho u^a), \quad (2.3)$$

where ρ is the rest mass density and u^a is the fluid two-velocity. Rest mass conservation is an approximation of baryon number conservation in a relativistic setting. In the usual $1 + 1$ decomposition of the spacetime, this equation becomes

$$0 = \partial_t \mathcal{D} + \partial_x [\alpha \mathcal{D} v], \quad (2.4)$$

where v is the fluid speed as seen by Eulerian observers, $W = \alpha u^t = 1/\sqrt{1 - v^2}$ is the corresponding Lorentz factor, and $\mathcal{D} = \rho W$ is the relativistic rest mass density.

The Valencia formulation uses the covariant conservation of momentum,

$$0 = \nabla_a T^a_i, \quad (2.5)$$

where $T^a_i = \rho h u^a u_i + \delta^a_i P$, h is the specific enthalpy, P is the pressure, a is a spacetime index, and i is a spatial index. In a $1 + 1$ decomposition, this equation becomes

$$0 = \partial_t S + \partial_x [\alpha S v + \alpha P] + (\rho h W^2 - P) \partial_x \alpha, \quad (2.6)$$

where $S = \rho h W^2 v$. The Hamiltonian formulation instead uses the Hamiltonian Euler equation

$$0 = u^a [\partial_a (h u_b) - \partial_b (h u_a)], \quad (2.7)$$

which does not have any covariant derivatives because the derivatives occur in an antisymmetric combination. In the usual $1 + 1$ decomposition of the spacetime, this equation becomes

$$0 = \partial_t p + \partial_x H, \quad (2.8)$$

where $p = h W v$ is the canonical momentum, and $H = \alpha h W$ is the Hamiltonian⁴. See [54] for a more general discussion of this barotropic Hamiltonian formulation of fluid dynamics.

⁴ In an arbitrary frame, stationarity of the fluid configuration implies that a slightly different quantity, namely $\tilde{H} \equiv \alpha h / W$, is constant. This is because stationarity in a moving frame corresponds to vanishing Lagrangian time derivatives $D/Dt = \partial_t + \alpha v \partial_x$. When equation (2.10) is written in terms of D/Dt , stationarity implies that $\alpha h / W$ is constant in space.

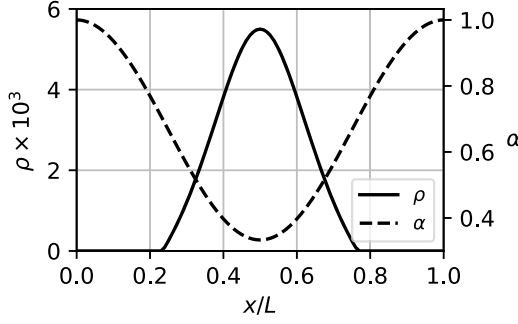


Figure 1. Rest mass density for an equilibrium configuration with $H = 0.7$, $\kappa = 100$, $\Gamma = 2$, and lapse function given by equation (2.2).

For ease of reference, we collect the equations of motion for the Valencia formulation here:

$$\begin{aligned} 0 &= \partial_t \mathcal{D} + \partial_x [\alpha \mathcal{D} v], \\ 0 &= \partial_t S + \partial_x [\alpha S v + \alpha P] + \partial_x \alpha (\rho h W^2 - P). \end{aligned} \quad (2.9)$$

And we collect the equations of motion for the Hamiltonian formulation here:

$$\begin{aligned} 0 &= \partial_t \mathcal{D} + \partial_x [\alpha \mathcal{D} v], \\ 0 &= \partial_t p + \partial_x H. \end{aligned} \quad (2.10)$$

The system is closed by an equation of state, e.g. $P = P(\rho)$ or $h = h(\rho)$.

A stationary fluid configuration in this spacetime is completely determined by setting $v = 0$ and specifying the value of its constant Hamiltonian H . The rest mass density profile is then recovered by inverting $H = H(\rho)$. A barotropic equation of state can be written as $h = h(\rho)$. Thus the recovery of ρ from a chosen value of H begins with obtaining h via $h = H/\alpha$, and then inverting $h = h(\rho)$. For the lapse function given in equation (2.2), the fluid has left and right surfaces when $H < 1$, which correspond to spatial locations x where $h(x) = 1$. When $H \geq 1$, the fluid extends across the entire spatial domain. In figure 1 we display the lapse function and resulting density profile for the choice $H = 0.7$, and a polytropic equation of state $P = \kappa \rho^\Gamma$ with $\kappa = 100$, and $\Gamma = 2$. This star has a central density $\rho_c = 5.5 \times 10^{-3}$, central energy density $\varepsilon_c \approx 8.7 \times 10^{-3}$, dynamical time $t_d = 1/\sqrt{\varepsilon_c} \approx 10$, and left and right surfaces at $x/L \approx 0.2341$ and $x/L \approx 0.7659$, respectively.

We also consider increasingly perturbed versions of this configuration, whereby the star is initialized with uniform motion $u^x/u^t = \alpha v \in \{0.01, 0.02, 0.04\}$. In these cases, the initial enthalpy is given by $h = \tilde{H}W/\alpha$ with $\tilde{H} = 0.7$, and the initial surface locations are correspondingly modified (but still satisfy $h = 1$). Note that the spacetime remains fixed, so that the resulting evolution is a star that sloshes back and forth in a fixed gravitational well.

Note that we use an equation of state that is not precisely $P = 100\rho^2$, but is very close (and described in section 3).

2.1. Characteristic structure

In this section we present the characteristic structure used in our numerical implementation. For a set of evolved variables $\mathbf{U}$, corresponding fluxes $\mathbf{F}$, sources $\mathbf{S}$, and lapse function $\alpha(x)$,

the equations of motion read

$$0 = \partial_t U + \partial_x [\alpha F] + S. \quad (2.11)$$

When seeking the characteristic structure of this system with respect to a chosen set of variables $\mathbf{q}$, one can use the chain rule as follows:

$$\begin{aligned} 0 &= \frac{\partial U}{\partial \mathbf{q}} \partial_t \mathbf{q} + \alpha \frac{\partial F}{\partial \mathbf{q}} \partial_x \mathbf{q} + \text{l.o.} \\ \rightarrow 0 &= \partial_t \mathbf{q} + \alpha \left(\frac{\partial U}{\partial \mathbf{q}} \right)^{-1} \frac{\partial F}{\partial \mathbf{q}} \partial_x \mathbf{q} + \text{l.o.}, \end{aligned} \quad (2.12)$$

where we have denoted by ‘l.o.’ terms that are lower order in the sense of characteristic analysis⁵, and assumed $\partial U / \partial \mathbf{q}$ is invertible. Equation (2.12) is the quasilinear form of the equations of motion with respect to the variables $\mathbf{q}$, and the Jacobian can be read off from equation (2.12) as the prefactor of $\partial_x \mathbf{q}$: $J \equiv \alpha (\partial U / \partial \mathbf{q})^{-1} \partial F / \partial \mathbf{q}$. If the fluxes and evolution variables can be written as explicit functions of the choice of variables $\mathbf{q}$, then the Jacobian is easy to compute.

We use the characteristic structure associated with two choices of variables: ‘safe’⁶ variables $\mathbf{q} = (h, W_V)^T$ and ‘flux’ variables $\mathbf{q} = \mathbf{F}$. For the flux variables, the Jacobian is the same as if $\mathbf{q} = \mathbf{U}$, namely

$$J = \alpha \frac{\partial \mathbf{F}}{\partial \mathbf{U}}, \quad (2.13)$$

which is readily computed in terms of primitive variables $\mathbf{p}$ using the chain rule⁷ and assuming the applicability of the inverse function theorem [83] for $\partial \mathbf{p} / \partial \mathbf{U}$ ⁸ (see equation (2.14) below, and set $\mathbf{q} = \mathbf{p}$).

For the safe variables, we use the full expression for the Jacobian,

$$J = \alpha \left(\frac{\partial \mathbf{U}}{\partial \mathbf{q}} \right)^{-1} \frac{\partial \mathbf{F}}{\partial \mathbf{q}}. \quad (2.14)$$

The matrices $\partial \mathbf{U} / \partial \mathbf{q}$ and $\partial \mathbf{F} / \partial \mathbf{q}$ are again readily computed in terms of primitive variables $\mathbf{p}$ via $\partial \mathbf{U} / \partial \mathbf{q} = (\partial \mathbf{U} / \partial \mathbf{p})(\partial \mathbf{q} / \partial \mathbf{p})^{-1}$ and $\partial \mathbf{F} / \partial \mathbf{q} = (\partial \mathbf{F} / \partial \mathbf{p})(\partial \mathbf{q} / \partial \mathbf{p})^{-1}$, which assumes the applicability of the inverse function theorem for $\partial \mathbf{q} / \partial \mathbf{p}$.

The Jacobian J , its eigenvalues $\lambda_{\pm}$, and its left and right eigenvectors $\mathbf{L}_{\pm}, \mathbf{R}_{\pm}$ are provided in table 1 for the safe variables and flux variables in the Valencia and Hamiltonian formulations. The speed of sound is $c_s = \sqrt{\partial P / \partial (\rho h)}$, which can be rewritten (for general h) as $c_s = \sqrt{(1/h) \partial \rho P}$ or as $c_s = \sqrt{(\rho/h) \partial \rho h}$. In our numerical implementation we make use of local characteristic variables defined by

$$\mathbf{V} = \begin{bmatrix} L_{+,1} & L_{+,2} \\ L_{-,1} & L_{-,2} \end{bmatrix}_X \mathbf{q}. \quad (2.15)$$

Where $\mathbf{L}_{\pm} = (L_{\pm,1}, L_{\pm,2})$, and the matrix of left eigenvectors is evaluated at the specific point $x = X$, where the characteristic decomposition is being performed.

⁵ For example, terms proportional to S and $\partial_x \alpha$.

⁶ The name ‘safe’ is explained in section 3.

⁷ i.e. $\partial \mathbf{F} / \partial \mathbf{U} = (\partial \mathbf{F} / \partial \mathbf{q}) \cdot \partial \mathbf{q} / \partial \mathbf{U}$.

⁸ i.e. $\partial \mathbf{p} / \partial \mathbf{U} = (\partial \mathbf{U} / \partial \mathbf{p})^{-1}$.

Table 1. Collection of fluid characteristic information with respect to several sets of variables.

Safe variables	Valencia flux variables	Hamiltonian flux variables
$J = \frac{\alpha}{1-v^2c_s^2} \begin{bmatrix} v(1-c_s^2) & hc_s^2/W^3 \\ 1/(hW) & v(1-c_s^2) \end{bmatrix}$	$J = \frac{\alpha}{1-v^2c_s^2} \begin{bmatrix} v(v^2-c_s^2) & 1/(hW^3) \\ \frac{h}{W}(v^2-c_s^2) & v(2-v^2-c_s^2) \end{bmatrix}$	$J = \frac{\alpha}{1-v^2c_s^2} \begin{bmatrix} v(1-c_s^2) & \rho/(hW^2) \\ hc_s^2/(\rho W^2) & v(1-c_s^2) \end{bmatrix}$
$\lambda_{\pm} = \alpha \frac{v \pm c_s}{1 \pm vc_s}$	$\lambda_{\pm} = \alpha \frac{v \pm c_s}{1 \pm vc_s}$	$\lambda_{\pm} = \alpha \frac{v \pm c_s}{1 \pm vc_s}$
$L_{\pm} = (\pm \frac{W}{hc_s}, 1)$	$L_{\pm} = (hW(-v \pm c_s), 1)$	$L_{\pm} = (\pm \frac{hc_s}{\rho}, 1)$
$R_{\pm} = (\pm \frac{hc_s}{W}, 1)^T$	$R_{\pm} = (\frac{1}{hW(v \pm c_s)}, 1)^T$	$R_{\pm} = (\pm \frac{\rho}{hc_s}, 1)^T$

∞

3. Vacuum boundary conditions

For concreteness we focus on the right boundary of our one-dimensional star. The left boundary description is analogous. The boundary conditions appropriate for the right stellar surface are

$$\begin{aligned} 3P(x_v) &\equiv \bar{P} = 0, \\ v(x_v) &\equiv \bar{v} = \lim_{x \rightarrow x_v^-} v(x). \end{aligned} \quad (3.1)$$

We denote evaluation at the surface with an overbar. The vanishing pressure condition implies $\bar{h} = 1$. We assume a liquid equation of state whereby the rest mass density takes a positive value $\bar{\rho}$ when the pressure vanishes, i.e. $P(\bar{\rho}) = 0$ and $\bar{\rho} > 0$. The use of a liquid equation of state greatly simplifies technical details in the description of the stellar surface, because it allows one to work directly at the surface rather than in a neighborhood of it (essentially by avoiding repeated divisions by zero there). An example of such an equation of state is that of a generalized polytrope,

$$h = 1 - a + \frac{\kappa\Gamma}{\Gamma - 1} \rho^{\Gamma-1}, \quad (3.2)$$

with $a = \text{constant} > 0$. The corresponding formula for the density is

$$\rho = \left(\frac{h - 1 + a}{\kappa(1 + n)} \right)^n, \quad (3.3)$$

where $\Gamma = 1 + 1/n$. The pressure vanishes when the enthalpy $h = 1$, implying that

$$\bar{\rho} = \left(\frac{a}{\kappa(1 + n)} \right)^n. \quad (3.4)$$

By taking $a \ll 1$, this equation of state deviates negligibly from the usual polytrope except in the vicinity of the surface of a star. In this work, we set $\bar{\rho} = 10^{-13}$, which is roughly 10 orders of magnitude below the central density of the star. We found no issue going even lower, and our results on convergence are not sensitive to the choice. The specific evolutions we obtain are also not sensitive to the choice of $\bar{\rho}$, provided it is chosen to be sufficiently small (a similar level as with artificial atmosphere prescriptions, i.e. at least 6 or 7 orders of magnitude below the central density of the star).

In past work, it was found that high-order numerical treatment of boundary conditions can be achieved stably as long as the inverse Lax–Wendroff procedure is followed for ingoing first derivatives [66], with all ingoing higher derivatives simply being extrapolated to the boundary. For a stellar surface, we find we are able to be even more aggressive; we can extrapolate *all* ingoing derivatives, and only impose the free surface boundary conditions on the undifferentiated fluid variables. This makes the implementation especially efficient. The greater stability we find for the stellar surface problem is probably because the surface is almost a characteristic boundary ($c_s \ll 1$). If the ingoing and outgoing characteristic speeds were very different, extrapolation of ingoing variables to the boundary would be numerically more dangerous. If the sound speed is not small at the surface, one can expect to have to follow the inverse Lax–Wendroff procedure for at least the ingoing first derivatives, as in [66]. In appendix A we have derived the constraint on first spatial derivatives of the fluid variables at the stellar surface, which would be necessary for the inverse Lax–Wendroff procedure. We also describe how to derive constraints on higher order derivatives.

As in reference [54], we extend the equation of state to negative pressures $h < 1$ via a piecewise attachment to $h(\rho)$. This is why we call the variables $\mathbf{q} = (h, Wv)^T$ ‘safe’, i.e. if extrapolations result in a value of the enthalpy less than 1 (but greater than 0), the equation of state supports it. The variable Wv is also safe (compared to v) because its physically-admissible range is $(-\infty, \infty)$.

4. Surface tracking

In this section, we describe our high-order accurate tracking algorithm for the fluid–vacuum interface. We follow reference [65] closely, and adapt their work to the relativistic case and to boundary conditions which are appropriate for the fluid–vacuum interface. We also combine the method with an efficiency improvement presented in [66], which we implement even more aggressively. The basis of the original approach is the inverse Lax–Wendroff method, which replaces spatial derivatives at the boundary with time derivatives, and was presented for nonlinear conservation laws originally in [59].⁹ But our implementation does not use the inverse Lax–Wendroff method at all, making it simpler to implement. For concreteness and ease of presentation, we use a 5th-order accurate scheme [66] in what follows, but other orders of accuracy can be obtained with obvious modifications (which we point out along the way).

First, it is instructive to give a few words about the time update procedure. At time t^n , where the exponent denotes a time level index, we assume the last interior point on the right side of the star is located at position x_R . Everything that follows works similarly for the left surface. The surface at time t^n is located at x_v such that $x_R \leq x_v < x_{R+1}$, and x_{R+1} is a vacuum point. The surface may or may not cross a grid point during the update (in a sub-step of the update, or after the entire update), either engulfing a new grid point or receding past one. If the surface engulfs the new grid point x_{R+1} in any sub-step of the update, values for the fluid variables are not assigned there until the full time step is complete, and only if the surface position at time t^{n+1} has engulfed the point x_{R+1} .¹⁰ If the surface recedes past the grid point x_R during any sub-step or after the entire update, strictly speaking one should anticipate this at the beginning of the time update, and then only evolve the points that are interior to the star during all sub-steps and at time t^{n+1} , i.e. $\{\dots, x_{R-3}, x_{R-2}, x_{R-1}\}$. However, rather than devise such an anticipation criterion, we instead drop the point x_R in the middle of the time update if the surface recedes past it on any sub-step of the time update (e.g. Runge–Kutta sub-steps), and continue to exclude that point for the remainder of the update. If x_R is still an interior point at time t^{n+1} , then we repopulate the fluid variables there at that stage.

The fluid variables at the interior points near the surface, which we collect as $X_{\text{int}} \equiv \{x_{R-4}, x_{R-3}, x_{R-2}, x_{R-1}, x_R\}$,¹¹ are assumed to have been updated (and possibly defined anew for x_R) from time t^{n-1} to time t^n , or they are provided by initial data if the initialization time is t^n . We will be assigning values of the split fluid fluxes $f^\pm$ (defined in section 5, and we denote simply as f in this section) at ghost points located at $X_{\text{ghost}} \equiv \{x_{R+1}, x_{R+2}\}$ ¹² via a Taylor

⁹ Note there is a sign typo in their equation [59]. It should read $\beta_2 = (61u_0^2 + 160u_1^2 + 74u_0u_2 + 25u_2^2 - 196u_1u_0 - 124u_1u_2)/12$.

¹⁰ It is possible that the surface could engulf a grid point or recede past a grid point in a sub-step of the update, but not after the entire update.

¹¹ If using the 7th-order scheme, then $X_{\text{int}} = \{x_{R-6}, x_{R-5}, x_{R-4}, x_{R-3}, x_{R-2}, x_{R-1}, x_R\}$.

¹² If using the 7th-order scheme, then $X_{\text{ghost}} = \{x_{R+1}, x_{R+2}, x_{R+3}\}$.

expansion at the fluid–vacuum interface x_v :

$$f_i = \sum_{k=0}^4 \frac{(x_i - x_v)^k}{k!} \bar{f}^{(k)}, \quad (4.1)$$

where $\bar{f}^{(k)}$ is the k th-order derivative of f approximated at $(5 - k)$ th order¹³ (denoted with the superscript $^{(k)}$) evaluated at the surface (denoted with the overbar).

We transform to *local* characteristic variables at the point x_R in order to control the outgoing versus ingoing fluid information. This is done separately for the unsplit fluxes F and the safe variables q from section 2. Using the characteristic structure from table 1, the local characteristic variable transformation for any set of variables (represented schematically as Q) is

$$QV_i = \begin{bmatrix} L_{+,1} & L_{+,2} \\ L_{-,1} & L_{-,2} \end{bmatrix}_{x_R} Q_i. \quad (4.2)$$

Notice that the matrix of left eigenvectors is evaluated at the point x_R , so by taking the derivative of both sides, we see that derivatives of the variables also transform in the same way:

$$QV_i^{(k)} = \begin{bmatrix} L_{+,1} & L_{+,2} \\ L_{-,1} & L_{-,2} \end{bmatrix}_{x_R} Q_i^{(k)}. \quad (4.3)$$

After obtaining local characteristic variables corresponding to the unsplit fluxes F and the safe variables q , we fit a 5th-order Lagrange polynomial¹⁴ to the available grid information near the surface, at locations X_{int} .¹⁵ Derivatives approximated at $(5 - k)$ th order (up to $k = 4$),¹⁶ $QV_i^{(k)}$, are obtained by analytic derivatives of the Lagrange polynomial fits. If the solution near the surface is expected to be non-smooth, a weighted essentially non-oscillatory (WENO) polynomial may be more appropriate [66]. But we find that the bare Lagrange polynomials work quite well¹⁷. In addition to denoting evaluation at the surface with an overbar, e.g. $\bar{Q}$, we denote extrapolated values with a star superscript, e.g. Q^* .

We extrapolate all the local characteristic variables to the surface, thereby obtaining $q\bar{V}^{(k)}$ and $F\bar{V}^{(k)}$. The only exceptions are the ingoing variables for $k = 0$, $q\bar{V}_-^{(0)}$ and $F\bar{V}_-^{(0)}$, which are instead effectively fixed by the free surface boundary condition $\rho|_{x_v} = \bar{\rho}$. For example, we solve for $\bar{v}$ using $q\bar{V}_+^*$ and using equation (4.2) evaluated at the surface:

$$q\bar{V}_+^* = \left(\frac{W}{hc_s} \right)_R + \bar{W}\bar{v}. \quad (4.4)$$

In this expression, the only unknown is $\bar{v}$ and one can always solve for it. Thus, we obtain $\bar{v}$ from equation (4.4), and $\bar{\rho}$ is the (constant) surface density value determined by the liquid equation of state. This yields $\bar{q}^{(0)}$, from which $\bar{F}^{(0)}$ can easily be computed.

¹³ $(7 - k)$ th order for a 7th-order scheme.

¹⁴ 7th-order Lagrange polynomial for a 7th-order scheme.

¹⁵ Note that overshooting would become an increasing issue with increasingly high-order Lagrange polynomial fits. Using WENO polynomials instead might counteract such an issue.

¹⁶ $(7 - k)$ th order up to $k = 6$ for a 7th-order scheme.

¹⁷ On physical grounds, fluid–vacuum interfaces do not technically support shockwaves because the Rankine–Hugoniot jump conditions are not satisfied there [70, 84]. But that fact does not exclude the possibility of gradients near the surface that are so large as to present a practical numerical problem for simple Lagrange polynomial extrapolation.

In this way, we obtain $\bar{q}^{(k)}$ and $\bar{F}^{(k)}$, from which we form the split fluxes $\bar{f}_{\pm}^{(k)}$ from section 5. Flux values are then obtained at any points where they are needed using the Taylor expansion equation (4.1). If populating fluid variables at interior points, we do so with the equivalent Taylor expansion for the safe variables q .

As per the style in references [59, 65, 66], we now summarize the procedure for populating fluxes at ghost points and possibly populating fluid variables at interior points:

- (a) Compute the local characteristic variables ${}^q V_i$ and ${}^F V_i$ at points X_{int} using equation (4.2).
- (b) Fit Lagrange polynomials to ${}^q V_i$ and ${}^F V_i$ at points X_{int} , then, using those polynomials, compute derivatives ${}^q V^{(k)}$ and ${}^F V^{(k)}$ and extrapolate to the surface to obtain ${}^q \bar{V}^{*(k)}$ and ${}^F \bar{V}^{*(k)}$.
- (c) Obtain $\bar{q}^{(k)}$ and $\bar{F}^{(k)}$ by inverting equation (4.3).
- (d) Treat the $k = 0$ case specially as done in the example surrounding equation (4.4), thereby obtaining $\bar{q}^{(0)}$ and by extension $\bar{F}^{(0)}$.¹⁸
- (e) Solve for $\bar{q}^{(k)}$ and $\bar{F}^{(k)}$ for $k = 1, 2, 3, 4$ from ${}^q \bar{V}^{*(k)}$ and ${}^F \bar{V}^{*(k)}$ by inverting equation (4.3).
- (f) Populate split fluxes $f_{\pm}$ wherever they are needed, for example X_{ghost} .
- (g) If needed, populate q_i at any interior points where values do not exist or must be replaced (see the discussion at the beginning of this section regarding the time update procedure, and see appendix C for policies to deal with failure modes).

The surface is evolved by solving the equation $\partial_t x_v = \bar{\alpha} \bar{v}$ with a time-stepping algorithm (e.g. Runge–Kutta), using the values of $\bar{v}$ obtained above at each time sub-step. Note the right-hand side is the advective velocity of the surface $(u^x/u^t)|_{x_v}$, which is the actual velocity one sees the surface move with on a fixed grid. In this work, the lapse function is known analytically, but in fully general relativistic simulations one would have to interpolate it to the surface location with an order of accuracy that is consistent with the rest of the methods.

5. Implementation

In this section we present our numerical discretization. In appendix C we describe various failure modes that we have encountered, together with policies for dealing with them.

We use a uniform grid over a domain of length L with grid spacing Δx . The system of equations of motion in semi-discrete form appear schematically in a flux-conservative form as

$$\partial_t U_i = -\frac{1}{\Delta x} (F_{i+1/2} - F_{i-1/2}) + S_i. \quad (5.1)$$

The evolved variables are represented by U , the fluxes are represented by F , and any source terms that may be present are represented by S . Spatial locations are indicated with the index i . We discretize spatial derivatives using 5th- or 7th-order upwind-biased finite differences, together with a global Lax Friedrichs flux split (see e.g. [85]). The upwind-biasing is necessary for numerical stability, and the flux split technique allows upwinding to be applied to a system with multiple signal speeds. The flux split is $F = f^+ + f^-$ where

$$f^{\pm} = \frac{1}{2} (F \pm \max(|\lambda|) U). \quad (5.2)$$

¹⁸ One could obviously use the safe variables and their derivatives, $\bar{q}^{(k)}$, to obtain the fluxes and their derivatives, $\bar{F}^{(k)}$, using analytic derivative formulae for F . But those formulae are very cumbersome for high-order derivatives, so we find it easier to extrapolate the fluxes separately.

The eigenvalues of the flux Jacobian are represented by λ , and the maximum is taken over the entire grid. By adding such terms $\pm \max(|\lambda|)U$, the eigenvalues of the Jacobians of $f^\pm$ are respectively semi-positive and semi-negative, allowing us to separately apply the appropriate upwind finite difference operator to each split flux. Given values of the flux $f^\pm$ at grid points (integer index i), 5th- and 7th-order left-biased approximations for $f_{i+1/2}^+$ and right-biased approximations for $f_{i-1/2}^-$ [57] are given by

$${}^5f_{i\pm 1/2}^\pm = \frac{1}{60}(2f_{i\mp 2}^\pm - 13f_{i\mp 1}^\pm + 47f_i^\pm + 27f_{i\pm 1}^\pm - 3f_{i\pm 2}^\pm) \quad (5.3)$$

$$\begin{aligned} {}^7f_{i\pm 1/2}^\pm = & -\frac{1}{140}f_{i\mp 3}^\pm + \frac{5}{84}f_{i\mp 2}^\pm - \frac{101}{420}f_{i\mp 1}^\pm + \frac{319}{420}f_i^\pm \\ & + \frac{107}{210}f_{i\pm 1}^\pm - \frac{19}{210}f_{i\pm 2}^\pm + \frac{1}{105}f_{i\pm 3}^\pm. \end{aligned} \quad (5.4)$$

The discretization (5.1) thus reads

$$\partial_t U_i = -\frac{1}{\Delta x} \left({}^s f_{i+1/2}^+ + {}^s f_{i+1/2}^- - {}^s f_{i+1/2}^+ - {}^s f_{i+1/2}^- \right) + S_i, \quad (5.5)$$

where s is the order of accuracy of the flux.

For time stepping we use a total variation diminishing Runge–Kutta scheme of 3rd-order accuracy. This is a widely used and robust scheme for computational fluid dynamics, owing to its suppression of spurious ringing near discontinuities (e.g. shockwaves). For a system written schematically as $\partial_t U = \mathcal{L}(U)$, the time stepping reads

$$\begin{aligned} U^{\{1\}} &= U^n + \Delta t \mathcal{L}(U^n) \\ U^{\{2\}} &= \frac{3}{4}U^n + \frac{1}{4} \left(U^{\{1\}} + \Delta t \mathcal{L}(U^{\{1\}}) \right) \end{aligned} \quad (5.6)$$

$$U^{n+1} = \frac{1}{3}U^n + \frac{2}{3} \left(U^{\{2\}} + \Delta t \mathcal{L}(U^{\{2\}}) \right), \quad (5.7)$$

where (U^n, U^{n+1}) are respectively the values of U at time levels $(n, n+1)$, and $(U^{\{1\}}, U^{\{2\}})$ are values at intermediate steps. For the purposes of simplicity in demonstrating high-order convergence, we obtain s th-order accuracy in time by using a time step $\Delta t \propto \Delta x^{s/3}$ [59]. This means the time step decreases fast enough as resolution is increased such that s th-order accuracy in time is achieved. In future applications, we do not suggest using this modification of the 3rd-order Runge–Kutta scheme, because there is still a comparable decrease in numerical error with the standard scheme (see figure 6 below). The surface position is evolved according to $\partial_t x_v = (\alpha v)|_{x_v}$ using the same Runge–Kutta time stepping as all other variables.

We compare our high-order methods with a standard finite volume scheme using the Harten–Lax–van Leer (HLL) flux formula [86] and a standard artificial atmosphere with constant density 10^{-13} and a standard polytropic equation of state $P = 100\rho^2$ everywhere. We use the barotropic Valencia scheme, which is described in [54].

6. Numerical results

In what follows, we use the Valencia formulation given by equation (2.9). We comment on using the Hamiltonian formulation in section 6.1.

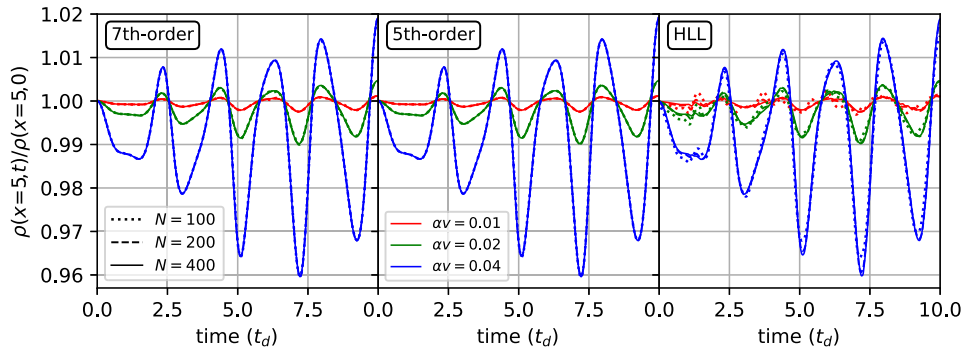


Figure 2. Density evolution at the center of the grid $x = 5$. Left, center, and right respectively show evolutions using the 7th-order, 5th-order, and HLL methods, all with CFL factor 1.0. Three resolutions $N = \{101, 201, 401\}$ are displayed using different line styles, and three initial uniform advective velocities $\alpha v = \{0.01, 0.02, 0.04\}$ are displayed using different colors. The dynamical time of the star is taken to be exactly $t_d = 10$.

Some properties of our simulations are displayed in figures 2 and 3, which evolve for a total duration of 10 dynamical times t_d (where we take t_d to be exactly 10, although a more precise value according to the central energy density is $t_d = 1/\sqrt{\epsilon_c} \approx 10.72$). The dynamical time scale is a gravitational time scale for the star, and it is a relevant scale because it is similar to the hydrodynamic time scale for the star near equilibrium. Figure 2 shows the evolution of the rest mass density at the center of the grid, $x = 5$, on a fractional basis relative to its initial value. The HLL method with artificial atmosphere (right) is shown along with our finite difference 5th-order (middle) and 7th-order (left) methods with surface tracking. Cases with three initial advective velocities $\alpha v \in \{0.01, 0.02, 0.04\}$ are displayed using different colors, while we omit the equilibrium case $\alpha v = 0$ because it appears uneventful in this plot and clutters the visual inspection. Three different resolutions corresponding to a number of grid points across the whole domain of $N \in \{100, 200, 400\}$ are shown using different line styles. The star initially spans roughly 53% of the domain, and its span fluctuates as it sloshes around in the fixed spacetime. As the initial velocity is increased, the density oscillations increase correspondingly. It is qualitatively evident that the HLL method with artificial atmosphere (right) is slower to converge, because the curves for different resolutions do not overlap well in comparison to the surface-tracking methods (left and middle). The disparity in convergence and absolute level of error are presented quantitatively in several figures described below.

For the surface-tracking method, the evolution history of the surface positions and velocities are available. These are shown in figure 3, where surface positions x_v are displayed in the left column and surface velocities v_v are displayed in the right column¹⁹. We only display the 7th-order method in this figure, but we display cases with all initial velocities $\alpha v \in \{0.00, 0.01, 0.02, 0.04\}$. In figure 5 below, which shows numerical errors over time, spikes in the numerical error correspond to maximal compression of the left and right sides of the star (since this yields maximal gradients at the surfaces). For the right surface, maximal gradients occur when x_v reaches locally minimal values, whereas for the left surface, maximal

¹⁹ Note that the actual surface motion would be given by the advective velocity, $\alpha|_{x_v} v_v$.

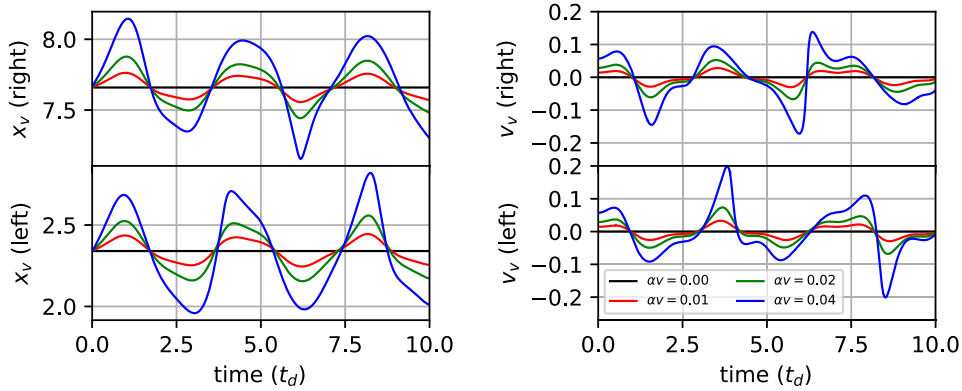


Figure 3. Left and right surface positions (left column) and surface velocities (right column) over time, for resolution $N = 400$ using the 7th-order method with Courant factor 1.0. Cases with four initial uniform advective velocities $\alpha v = \{0.00, 0.01, 0.02, 0.04\}$ are displayed using different colors. The dynamical time of the star is taken to be exactly $t_d = 10$.

gradients occur when x_v reaches locally maximal values (compare figure 5 to the left column of figure 3).

Next we present the convergence properties of the different methods on a quantitative basis. We quantify numerical errors by evaluating the residuals of the equations of motion, namely the local conservation of rest mass $|\nabla_a(\rho u^a)|$ and momentum $|\nabla_a T^{ai}|$ (both of which would be zero given analytic solutions). The numerically-obtained fluid variables are plugged into these residuals, and derivatives are computed using centered finite difference operators with an order of accuracy greater than the expected order of convergence (so as not to contaminate the measurement of convergence with the errors associated with such derivative evaluations). At the edges of the star, the residuals are not evaluated because the spatial derivative stencils extend into the vacuum regions. Similarly for the time derivatives at the temporal beginning and end of the simulations; residuals are not evaluated there because time derivative stencils extend beyond the computed data. We present the residuals on a space-and-time integrated basis versus resolution in figure 4, as well as on a space integrated basis versus time in figure 5. Generally speaking, we present a resolution range corresponding to $N \in [100, 400]$. $N = 100$ is slightly outside of the convergent regime for some initial velocities, whereas machine precision limits our ability to measure convergence for the surface-tracking methods when $N \gtrsim 400$.

In figures 4 and 5, the finite difference orders used to evaluate the residuals for the 7th-order, 5th-order, and HLL methods are respectively 8th-order, 6th-order, and 4th-order. The time steps used are also not the same. In all methods, the Courant–Friedrichs–Lewy factor is 1 for the lowest resolution corresponding to $N = 100$. But for the 7th- and 5th-order cases, the Courant factor is respectively $(\Delta x/0.1)^{7/3-1}$ and $(\Delta x/0.1)^{5/3-1}$, where Δx is the spatial resolution $\Delta x = L/N$ (these choices maintain 7th- and 5th-order accuracy in time, even though we use a 3rd-order accurate Runge–Kutta time-stepping method). For these reasons, it is not fair to compare the absolute levels of error between the three methods in figures 4 and 5. However, it is fair to compare the convergence orders.

Note that it is more delicate to measure convergence in the case of an equilibrium star, $\alpha v = 0$, because as grid points are added to the domain, the distance between the outermost interior points and the respective surfaces does not decrease monotonically, which then reflects

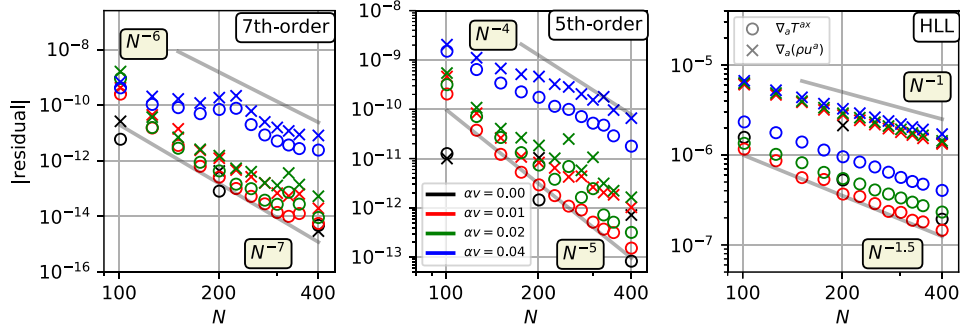


Figure 4. Global convergence rates for the 7th-order scheme (left), 5th-order scheme (middle), and standard finite volume scheme using the HLL flux and an artificial atmosphere (right). Cases with four initial uniform advective velocities $\alpha v = \{0.00, 0.01, 0.02, 0.04\}$ are displayed using different colors. The momentum and mass residuals are displayed with circle and cross tick marks. The residuals are averaged over ~ 10 dynamical times and over the interior of the star. Power-law trends are indicated with gray lines. Comparing the three methods with each other is not meaningful in this figure, except for the convergence rates, because different time steps are used and the residuals are evaluated with different order finite difference operators (8th, 6th, and 4th, respectively for left, middle, and right). For the equilibrium star $\alpha v = 0$, we only display $N \in \{100, 200, 400\}$ for reasons explained in the text.

upon the residuals as large fluctuations versus N .²⁰ For this reason, we display resolutions which are only integer multiples of $N = 100$, namely $N \in \{100, 200, 400\}$, which yields meaningful measurements of convergence because grid point positions are never removed as N doubles, only added. In non-equilibrium cases $\alpha v \neq 0$, this issue is less pronounced because the surface moves across grid points anyway; the numerical error associated with the initial position of the star on the grid is less dominant than the errors associated with the stellar motion. We believe this concern is a peculiarity of the one-dimensional case because there are only two outermost interior points. In multiple dimensions, there will be many outermost interior points with a well-populated range of shortest distances to the surface, thereby contributing a diverse range of numerical errors to the total error budget. In aggregate, such errors would not fluctuate drastically as the position of the star shifts slightly with respect to increasingly larger grids.

In figure 4, the mass and momentum residuals are displayed using different tick marks, cases with different initial advective velocities $\alpha v \in \{0.00, 0.01, 0.02, 0.04\}$ are displayed using different colors, and gray solid lines are reference power-law trends. We observe 7th- and 5th-order convergence (left and middle) for low values of the initial advective velocity αv . For the most extreme case with $\alpha v = 0.04$, the convergence order degrades by roughly 1, and it becomes apparent that the convergent regime is roughly $N \geq 200$. By contrast, the standard HLL method with artificial atmosphere yields the widely observed 1.5 convergence order, with a noticeable degradation to 1 for the mass residual. The degradation of the mass residual is also noticeable for the 5th-order method, but appears less pronounced for the 7th-order method. The explanation for this is unclear.

²⁰ Although if one appropriately shifts the position of the star with respect to the grid as points are added, then the distance between the outermost interior points and the respective surfaces would decrease monotonically.

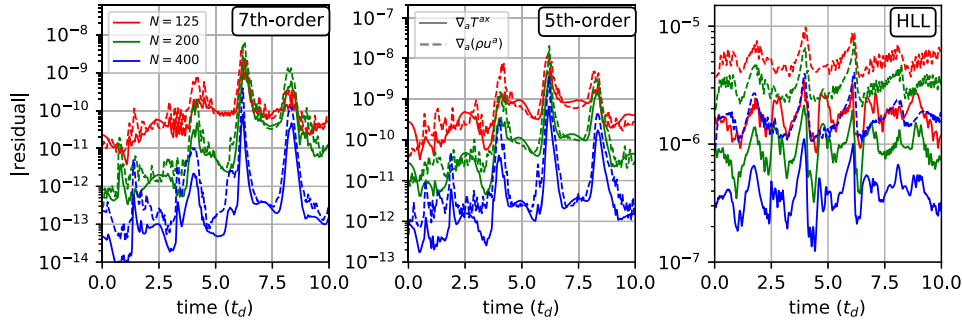


Figure 5. Spatially-averaged residuals over time for the 7th-order scheme (left), 5th-order scheme (middle), and standard finite volume scheme using the HLL flux and an artificial atmosphere (right). Only the most extreme case with initial uniform advective velocity $\alpha v = 0.04$ is displayed, which exhibits the clearest spikes in residuals (and degradation of convergence order) when surface gradients and acceleration of the surface become large. A selection of resolutions is displayed, corresponding to a number of grid points $N = \{125, 200, 400\}$. The momentum and mass residuals are displayed with solid and dashed lines. The residuals are averaged over the interior of the star, and the dynamical time is taken to be exactly $t_d = 10$. The curves have been gently smoothed with a Gaussian kernel of width $0.2 \times t_d$, in order to improve the visual presentation. Caution should be used when comparing the three methods using this figure, because different time steps are used and the residuals are evaluated with different order finite difference operators (8th, 6th, and 4th, respectively for left, middle, and right).

In figure 5 we show the spatially integrated residuals as a function of time. The mass and momentum residuals are displayed using different line styles, and a selection of three resolutions corresponding to $N \in \{125, 200, 400\}$ are displayed using different colors. Only the most extreme initial velocity is shown, $\alpha v = 0.04$, because this case most clearly illustrates the spiking of numerical errors coinciding with the local maxima of surface gradients (compare with figure 3). In some instances, for example comparing $N \in \{125, 200\}$ for the 7th-order method (left), the residual does not settle down to previous levels following a spike (look at time $\sim 6.25t_d$). This behavior, as well as the spiking in residuals themselves, are what account for the degradation of global convergence order observed in figure 4 for the most extreme case $\alpha v = 0.04$. The residuals in the HLL method with artificial atmosphere also exhibit spikes at these times, although they are somewhat less pronounced.

In order to make a fair comparison between the different methods on the basis of absolute level of residuals, in figure 6 we repeat the same style of plots as in figure 4, except we use the same Courant factor of 1 across all cases and we use the same order finite difference operator (8th-order) when evaluating all residuals. The absolute level of error as measured by these residuals is roughly 5 to 6 times smaller for the surface-tracking methods than for the HLL method with artificial atmosphere. Furthermore, for the surface-tracking methods, we observe that the convergence order degrades to 3 as one moves either to larger αv or larger N . This reflects the dominance of the numerical error associated with time stepping, which with fixed Courant factor decreases as 3rd order for our Runge–Kutta method. This figure shows that one does not necessarily need a time-stepping method higher than 3rd order to achieve the benefits of low numerical error. However, higher order strong stability preserving time-stepping methods exist [87] which could be utilized to achieve an overall convergence order of 5 or 7, for example.

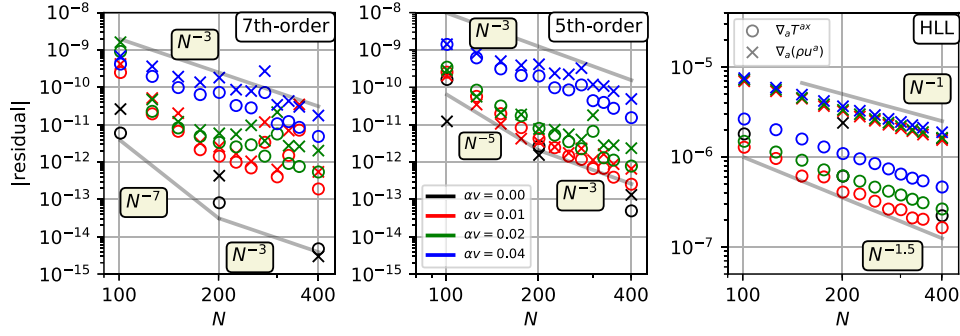


Figure 6. Global convergence rates with a fixed Courant–Friedrichs–Levy factor of 1.0 for the 7th-order scheme (left), 5th-order scheme (middle), and standard finite volume scheme using the HLL flux and an artificial atmosphere (right). Cases with four initial uniform advective velocities $\alpha v = \{0.00, 0.01, 0.02, 0.04\}$ are displayed using different colors. The momentum and mass residuals are displayed with circle and cross tick marks. The residuals are averaged over ~ 10 dynamical times and over the interior of the star. Power-law trends are indicated with gray lines. Comparing the three methods with each other is meaningful in this figure, because the same time stepping is used and the residuals are evaluated with the same (8th-order) finite difference operator. The convergence order for the 7th-order and 5th-order methods quickly drops to 3rd-order as time step errors become dominant, reflecting the 3rd-order Runge Kutta time-stepping method. For the equilibrium star $\alpha v = 0$, we only display $N \in \{100, 200, 400\}$ for reasons explained in the text.

Lastly, we demonstrate long-term stability of each method in figure 7. We show only the most extreme case $\alpha v = 0.04$, which is the most demanding test for long-term stability. We use a single resolution corresponding to $N = 150$, which is roughly the lowest number of grid points which yields stable evolution over 100 dynamical times. The left plot shows the mass and momentum residuals over time, and stability is evidenced by the absence of any apparent secular drift in residuals. The right plot shows the rest mass density evolution at the center of the grid $x = 5$, on a fractional basis with respect to its initial value. The HLL method with artificial atmosphere has a noticeable secular decay in central density oscillations relative to the surface-tracking 7th- and 5th-order methods, meaning that the latter methods are less dissipative at the same grid resolution. See supplemental material at (<https://stacks.iop.org/CQG/38/145003/mmedia>) for a video of the long-term evolution.

6.1. Hamiltonian formulation

In this subsection we briefly comment on the possibility of using the Hamiltonian formulation with the surface-tracking methods. It was found in [54] that surface stability issues were a persistent problem when using the HLL method with an artificial atmosphere, and this problem was solved by using a hybrid scheme whereby the Valencia formulation is used at the last interior point of the star and beyond. Our tests in this work indicate that it is possible to use the pure Hamiltonian formulation, and the convergence properties are similar to the Valencia formulation. However, the stability is greatly diminished. Failure modes are encountered more often, requiring the invocation of error handling policies, whereas with the Valencia formulation we almost never encounter a failure mode (unless the initial velocity of the star is roughly $\alpha v = 0.6$ or greater, and even then one expects failure modes to be avoided with a shock-capturing scheme since large surface gradients would be more well-tolerated). We were not

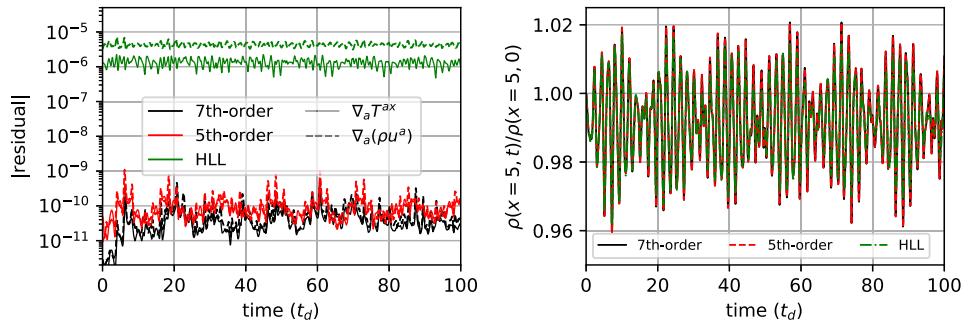


Figure 7. This figure shows long-term stability over 100 dynamical times. Spatially-averaged residuals over time for the 7th-order, 5th-order, and standard finite volume scheme using the HLL flux and an artificial atmosphere (left). Only the most extreme case with initial uniform advective velocity $\alpha v = 0.04$ is displayed. One resolution is displayed ($N = 150$), corresponding to roughly the lowest resolution for which long-term stability is achieved using all methods. The momentum and mass residuals are displayed with solid and dashed lines. The residuals are averaged over the interior of the star, and the dynamical time is taken to be exactly $t_d = 10$. The curves have been gently smoothed with a Gaussian kernel of width t_d , in order to improve the visual presentation. Different time steps are used and the residuals are evaluated with different order finite difference operators (8th, 6th, and 4th, respectively for 7th-order, 5th-order, and HLL methods). Central density evolutions from the same simulations are displayed on the right.

able to evolve any moving star cases stably for greater than about 30 dynamical times with the Hamiltonian formulation. The stability issues are again surface-related, which probably stem from the fact that the weak solutions for a fluid–vacuum interface in the Hamiltonian formulation support shockwaves [55] (whereas in the Valencia formulation only rarefaction waves are supported [70, 84]). Thus, in future applications of the Hamiltonian formulation with our surface-tracking methods, we expect it to be advisable and perhaps necessary to once again hybridize the formulation with Valencia at the outermost interior points, as done in reference [54].

7. Summary

Future gravitational wave detections will have signal-to-noise ratios which challenge the accuracy of current state-of-the-art waveform models calibrated to numerical relativity simulations. In numerical relativity simulations of compact binaries involving at least one material body, the standard way that stellar surfaces are treated (using an artificial atmosphere) is a significant source of error and the primary obstacle preventing high-order convergence during the inspiral phase of the binary. We presented a new numerical method which tracks and evolves the stellar surface, and imposes free surface boundary conditions there with an arbitrarily high order of accuracy (in principle). Our surface-tracking method is independent of the scheme used in the interior of the star, which should allow wide adoption of our method. We demonstrated this new method in a $1 + 1$ -dimensional proof-of-principle numerical implementation, whereby a relativistic one-dimensional star with a barotropic liquid equation of state undergoes violent sloshing motion inside of a fixed gravitational potential. The method performs very well in comparison to a standard finite volume scheme with an artificial atmosphere, both in convergence rate as well as a decrease in absolute numerical errors by 5–6 orders of magnitude at

the same resolution. In the future, we anticipate that this method will be applied to simulate the inspiral phase of material compact binaries in full 3 + 1-dimensional numerical relativity (which is expected to be barotropic fluid flow to a high degree of accuracy).

A challenge for multiple dimensions, which we have not addressed in this work, is dealing not just with a moving boundary (the surface), but with a deformable one. We anticipate that non-oscillatory interpolations of the surface position will be necessary to prevent excessive unphysical wrinkling of the stellar surface in multiple dimensions. However, this is not merely a technical challenge; it is also an opportunity to input more precise surface tension physics into numerical simulations, which may be accessible observationally with future gravitational wave detectors.

We do not believe that extending our surface-tracking method to baroclinic liquids would be a significant challenge, and there are no obvious impediments to using a realistic, tabulated equation of state. However, extending our method to the merger phase of compact binaries, whereby matter is being stripped and ejected, would be very challenging. From a geometrical standpoint, the number of distinct fluid–vacuum interfaces during the merger phase would undergo a rapid bifurcation cascade, with each surface undergoing potentially very large shape variations, which would require an exceptionally robust surface-tracking algorithm.

Acknowledgments

We thank Charalampos Markakis, Vasileios Paschalidis, Frans Pretorius, Sirui Tan, and Chi-Wang Shu for discussions, as well as Nikolaos Tsakiris and Nick Stergioulas for inspiration [72] on using a surface-tracking algorithm. This research is supported by National Science Foundation Grant No. PHY-1912619 at the University of Arizona.

Data availability statement

The data that support the findings of this study are available upon reasonable request from the authors.

Appendix A. Surface derivative constraint in the toy model

Since the fluid flow in the vicinity of the surface matches the surface motion, any quantity that takes on a persistent value at the stellar surface will have a vanishing Lagrangian time derivative there $D/Dt = \partial_t + (u^x/u^t)\partial_x$ [88], where in our toy model $u^x/u^t = \alpha v$. For example, the pressure is persistently zero at the surface, and therefore satisfies

$$\frac{DP}{Dt} = \partial_t P + \alpha v \partial_x P = 0, \quad (\text{A1})$$

at the surface. The density is persistently equal to $\bar{\rho}$, defined by $P(\bar{\rho}) = 0$, thus we also have $D\rho/Dt = 0$. Similarly, we have $D(\rho h)/Dt = 0$ at the surface.

We can derive a condition on the spatial derivatives of the fluid variables at the surface using these vanishing Lagrangian time derivatives and the fluid equations. Write the rest mass conservation equation (2.4) in terms of the Lagrangian time derivative,

$$\begin{aligned}
0 &= \partial_t(\rho W) + \partial_x(\alpha \rho W v) \\
&= \frac{D}{Dt}(\rho W) + \rho W \partial_x(\alpha v) \\
&= \rho \frac{DW}{Dt} + W \frac{D\rho}{Dt} + \rho W \partial_x(\alpha v).
\end{aligned} \tag{A2}$$

Evaluation at the surface yields

$$0 = \bar{\rho} \frac{\overline{DW}}{Dt} + \bar{\rho} \overline{W \partial_x(\alpha v)}. \tag{A3}$$

Note that $DW/Dt = W^3 v Dv/Dt$. The energy equation $\nabla_a T^{at}$ for our toy model expressed in a $1 + 1$ decomposition of spacetime is [89]

$$\begin{aligned}
0 &= \partial_t(\rho h W^2 - P) + \partial_x(\alpha \rho h W^2 v) + \rho h W^2 v \partial_x \alpha \\
&= \partial_t(\rho h W^2) - \frac{DP}{Dt} + \alpha v \partial_x P + \partial_x(\alpha \rho h W^2 v) + \rho h W^2 v \partial_x \alpha,
\end{aligned} \tag{A4}$$

where in the second line we made the replacement $-\partial_t P = -DP/Dt + \alpha v \partial_x P$. Writing the remaining Eulerian time derivatives in terms of Lagrangian ones, we obtain

$$0 = W^2 \frac{D(\rho h)}{Dt} + \rho h 2W \frac{DW}{Dt} - \frac{DP}{Dt} + \alpha v \partial_x P + \rho h W^2 \partial_x(\alpha v) + \rho h W^2 v \partial_x \alpha. \tag{A5}$$

Evaluation at the surface causes $D(\rho h)/Dt$ and DP/Dt to vanish and $\bar{h} = 1$:

$$0 = \bar{\rho} 2 \bar{W} \frac{\overline{DW}}{Dt} + \bar{\alpha} \bar{v} \overline{\partial_x P} + \bar{\rho} \bar{W}^2 \overline{\partial_x(\alpha v)} + \bar{\rho} \bar{W}^2 \bar{v} \overline{\partial_x \alpha}. \tag{A6}$$

Eliminating $\overline{DW}/Dt$ using equations (A6) and (A3) yields the constraint on spatial derivatives of the fluid variables:

$$\bar{v} \overline{\partial_x P} = \bar{\rho} \bar{W}^2 \overline{\partial_x v}. \tag{A7}$$

This constraint equation (A7) must be satisfied at a stellar surface satisfying the boundary conditions (3.1). Note we can rewrite $\partial_x v$ in terms of the primitive-like variable $\partial_x(Wv)$ via $\partial_x(Wv) = W^3 \partial_x v$. To gain an intuition for what the constraint equation (A7) means, note first that for an equilibrium star ($\partial_x(\alpha v) = 0$) in the rest frame $v = 0$, $\overline{\partial_x P}$ is left unconstrained (as one would expect, since there are many possible equilibrium stars with varying $\overline{\partial_x P}$). For an equilibrium star ($\partial_x(\alpha v) = 0$) in a boosted frame $v \neq 0$, equation (A7) expresses the condition of hydrostatic equilibrium at the surface in that frame.

An interesting implication of equation (A7) is that one cannot initialize a star in an arbitrary but seemingly reasonable manner and expect it to have a free surface. For example, initializing an equilibrium density profile with $\overline{\partial_x P} \neq 0$ but setting $v = \text{constant} > 0$ results in a violation of equation (A7). In order for the initial data to satisfy the free surface vacuum boundary conditions (3.1), care must be taken to enforce equation (A7) as well. Combining instead rest mass conservation $\nabla_a(\rho u^a) = 0$ with the Euler equation $\nabla_a T_i^a = 0$ yields the same constraint as equation (A7).

Appendix B. Inverse-Lax Wendroff procedure for first derivatives

Denote k th-order spatial derivatives of quantities with a superscript, e.g. $q^{(k)}$. Note that the matrix of left eigenvectors in equation (4.2) is evaluated at the point x_R , whereas V and q are evaluated at arbitrary points x_i . Thus, derivatives of the fluid variables also obey this relation because the derivative operator treats the matrix of left eigenvalues at x_R as constant. At the surface, using the safe variables from section 2.1, we obtain

$$\bar{V}^{(k)} = \begin{bmatrix} W/(hc_s) & 1 \\ -W/(hc_s) & 1 \end{bmatrix}_R \bar{q}^{(k)}. \quad (\text{B1})$$

With extrapolation of the outgoing first-order derivative, we have

$$\bar{V}_+^{*(1)} = \left(\frac{W}{hc_s} \right)_R \bar{\partial}_x h + \bar{\partial}_x(Wv), \quad (\text{B2})$$

where there are two unknowns $\bar{\partial}_x h$ and $\bar{\partial}_x(Wv)$. We can eliminate one unknown using the constraint equation (A7), which we write in terms of $\bar{\partial}_x \rho = \bar{\partial}_h \rho \bar{\partial}_x h = (\bar{\rho}/\bar{c}_s^2) \bar{\partial}_x h$ and $\bar{\partial}_x(Wv)$ as

$$\bar{W} \bar{v} \bar{\partial}_x h = \bar{\partial}_x W v. \quad (\text{B3})$$

Plugging into equation (B2) gives

$$\bar{V}_+^{*(1)} = \left[\left(\frac{W}{hc_s} \right)_R + \bar{W} \bar{v} \right] \bar{\partial}_x h, \quad (\text{B4})$$

which we can use to solve for $\bar{\partial}_x h$ and then recover $\bar{\partial}_x(Wv)$ using equation (B3). This expression is not invertible for $\bar{\partial}_x h$ when $\left(\frac{W}{hc_s} \right)_R + \bar{W} \bar{v} = 0$. In this unlikely scenario, one can instead extrapolate both characteristic variables to obtain $V_\pm^{*(1)}$, and then invert equation (B1) to obtain q^* .

This would constitute the inverse Lax–Wendroff procedure for the first derivatives, but we find it to be unnecessary. For derivatives higher than first order, in principle one can derive constraints on higher-order spatial derivatives of the fluid variables following a similar strategy as we did for equation (A7). For example, one can use the fact that $\bar{D}^k P / \bar{D} t^k = \bar{D}^k \rho / \bar{D} t^k = \bar{D}^k \rho h / \bar{D} t^k = 0$. But this is numerically unnecessary, and with a dynamical spacetime would actually involve time derivatives of the metric, leading to very cumbersome constraints on the fluid derivatives at the surface.

Appendix C. Failure modes and error-handling policies

There are failure modes we have encountered during the development of our surface-tracking method. They all involve the density becoming negative at interior stellar points close to the surface.

In the event that the density at the last interior point becomes negative or zero, it is repopulated using an interpolation of the same order as the rest of the methods (e.g. 5th- or 7th-order). If the interpolated replacement also has a negative density, then the surface position is manually receded to some fraction A of a grid separation Δx away from the last interior positive density point. This failure mode might occur, for example, if the surface position is a tiny distance away from the last interior point, such that the density at the last interior point is very small.

If the density at multiple consecutive last interior points (e.g. the last two points) becomes negative or zero, then the surface position is reset to some fraction of a grid separation $A\Delta x$ away from the last positive density point, and the points at which the density became negative are therefore deemed to be in the vacuum region. One could alternatively attempt to repopulate the points with physical values, but we interpret this failure mode as an indication that the surface evolution has become too inconsistent with the fluid.

Note that these failure modes almost never occur for the Valencia formulation. For the Hamiltonian formulation, these failure modes occur often, thus such evolutions are unsupportable without the error handling policies. When using the Valencia formulation, the failure modes above usually indicate an implementation bug. Generally speaking, the latter failure mode (negative density at multiple consecutive points) is more severe.

ORCID iDs

John Ryan Westernacher-Schneider  <https://orcid.org/0000-0002-3047-7200>

References

- [1] Aasi J *et al* 2015 *Class. Quantum Grav.* **32** 074001
- [2] Acernese F *et al* 2014 *Class. Quantum Grav.* **32** 024001
- [3] Abbott B P *et al* 2016 *Phys. Rev. Lett.* **116** 061102
- [4] Abbott B P *et al* 2017 *Phys. Rev. Lett.* **119** 161101
- [5] Abbott B *et al* 2019 *Phys. Rev. X* **9** 031040
- [6] Abbott B P *et al* 2017 *Class. Quantum Grav.* **34** 104002
- [7] Haster C-J, Wang Z, Berry C P L, Stevenson S, Veitch J and Mandel I 2016 *Mon. Not. R. Astron. Soc.* **457** 4499
- [8] Buonanno A and Damour T 1999 *Phys. Rev. D* **59** 084006
- [9] Husa S, Khan S, Hannam M, Pürrer M, Ohme F, Forteza X J and Bohé A 2016 *Phys. Rev. D* **93** 044006
- [10] Boyle M *et al* 2019 *Class. Quantum Grav.* **36** 195006
- [11] Dietrich T *et al* 2018 *Class. Quantum Grav.* **35** 24LT01
- [12] Haas R *et al* 2016 *Phys. Rev. D* **93** 124062
- [13] Kiuchi K, Kawaguchi K, Kyutoku K, Sekiguchi Y, Shibata M and Taniguchi K 2017 *Phys. Rev. D* **96** 084060
- [14] Abbott B P *et al* 2016 *Phys. Rev. X* **6** 041014
- [15] Abbott B *et al* 2019 *Phys. Rev. X* **9** 011001
- [16] Abbott B P *et al* 2017 *Class. Quantum Grav.* **34** 044001
- [17] Reitze D *et al* 2019 arXiv:1907.04833
- [18] Samajdar A and Dietrich T 2018 *Phys. Rev. D* **98** 124030
- [19] Samajdar A and Dietrich T 2019 *Phys. Rev. D* **100** 024046
- [20] Brown D 2021 *Bull. Am. Phys. Soc.* <https://meetings.aps.org/Meeting/APR21/Session/X05.2>
- [21] Goldberger W D and Rothstein I Z 2006 *Phys. Rev. D* **73** 104029
- [22] Galley C R and Tiglio M 2009 *Phys. Rev. D* **79** 124027
- [23] Blanchet L 2014 *Living Rev. Relativ.* **17** 2
- [24] Bini D, Damour T and Geralico A 2020 *Phys. Rev. D* **102** 024062
- [25] Bini D, Damour T and Geralico A 2020 *Phys. Rev. D* **102** 024061
- [26] Nagar A and Retegno P 2019 *Phys. Rev. D* **99** 021501
- [27] Blümlein J, Maier A, Marquard P and Schäfer G 2020 arXiv:2003.01692
- [28] Landry P and Poisson E 2015 *Phys. Rev. D* **91** 104018
- [29] Landry P 2017 *Phys. Rev. D* **95** 124058
- [30] Pani P, Gualtieri L and Ferrari V 2015 *Phys. Rev. D* **92** 124003
- [31] Pani P, Gualtieri L, Maselli A and Ferrari V 2015 *Phys. Rev. D* **92** 024010
- [32] Steinhoff J, Hinderer T, Buonanno A and Taracchini A 2016 *Phys. Rev. D* **94** 104028

- [33] Lim D-J, Anderson T R and Shott T 2015 *Omega* **51** 128
- [34] Stevens R, Ramprakash J, Messina P, Papka M and Riley K 2019 Aurora: Argonne's next-generation exascale supercomputer *Tech. Rep.* (Argonne National Laboratory (ANL))
- [35] Narasimhamurthy S *et al* 2019 *Parallel Comput.* **83** 22
- [36] Kidder L E *et al* 2017 *J. Comput. Phys.* **335** 84
- [37] Palenzuela C, Miñano B, Viganò D, Arbona A, Bona-Casas C, Rigo A, Bezares M, Bona C and Massó J 2018 *Class. Quantum Grav.* **35** 185007
- [38] Zhang W *et al* 2019 *J. Open Source Softw.* **4** 1370
- [39] Fernando M, Neilsen D, Lim H, Hirschmann E and Sundar H 2019 *SIAM J. Sci. Comput.* **41** C97
- [40] Neilsen D, Fernando M, Sundar H and Hirschmann E 2019 A scalable framework for adaptive computational general relativity on heterogeneous clusters *Proc. ACM Int. Conf. on Supercomputing* pp 1–12
- [41] Radice D, Rezzolla L and Galeazzi F 2013 *Mon. Not. R. Astron. Soc.* **437** L46
- [42] Miller J M and Schnetter E 2016 *Class. Quantum Grav.* **34** 015003
- [43] Bugner M, Dietrich T, Bernuzzi S, Weyhausen A and Brügmann B 2016 *Phys. Rev. D* **94** 084004
- [44] Bernuzzi S and Dietrich T 2016 *Phys. Rev. D* **94** 064062
- [45] Felker K G and Stone J M 2018 *J. Comput. Phys.* **375** 1365
- [46] Foucart F, Duez M, Gudinias A, Hébert F, Kidder L, Pfeiffer H and Scheel M 2019 *Phys. Rev. D* **100** 104048
- [47] Most E R, Papenfort L J and Rezzolla L 2019 *Mon. Not. R. Astron. Soc.* **490** 3588
- [48] Fambri F, Dumbser M, Köppel S, Rezzolla L and Zanotti O 2018 *Mon. Not. R. Astron. Soc.* **477** 4543
- [49] Fambri F 2020 *Arch. Comput. Methods Eng.* **27** 199
- [50] Köppel S 2018 *J. Phys.: Conf. Ser.* **1031** 012017
- [51] Ruchlin I, Etienne Z B and Baumgarte T W 2018 *Phys. Rev. D* **97** 064036
- [52] Ruchlin I, Etienne Z B and Baumgarte T W 2018 *SENr: Simple, Efficient Numerical Relativity Astrophysics Source Code Library*
- [53] Poudel A, Tichy W, Brügmann B and Dietrich T 2020 *Phys. Rev. D* **102** 104014
- [54] Westernacher-Schneider J R, Markakis C and Tsao B J 2020 *Class. Quantum Grav.* **37** 155005
- [55] Westernacher-Schneider J R 2020 arXiv:2004.15000
- [56] Radice D, Rezzolla L and Galeazzi F 2014 *Class. Quantum Grav.* **31** 075012
- [57] Guercilena F, Radice D and Rezzolla L 2017 *Comput. Astrophys. Cosmol.* **4** 3
- [58] Xing Y and Shu C-W 2013 *J. Sci. Comput.* **54** 645
- [59] Tan S and Shu C-W 2010 *J. Comput. Phys.* **229** 8144
- [60] Lax P and Wendroff B 1960 *Commun. Pure Appl. Math.* **13** 217
- [61] Goldberg M and Tadmor E 1978 *Math. Comput.* **32** 1097
- [62] Goldberg M and Tadmor E 1981 *Math. Comput.* **36** 603
- [63] Lombard B, Piraux J, Gélis C and Virieux J 2008 *Geophys. J. Int.* **172** 252
- [64] Huang L, Shu C-W and Zhang M 2008 *J. Comput. Math.* **26** 336
- [65] Tan S and Shu C-W 2011 *J. Comput. Phys.* **230** 6023
- [66] Tan S, Wang C, Shu C-W and Ning J 2012 *J. Comput. Phys.* **231** 2510
- [67] Vilar F and Shu C-W 2015 *ESAIM: Math. Modelling Numer. Anal.* **49** 39
- [68] Li T, Shu C-W and Zhang M 2016 *J. Comput. Appl. Math.* **299** 140
- [69] Shu C-W and Tan S 2017 *Handbook of Numerical Analysis* vol 18 (Amsterdam: Elsevier) pp 23–52
- [70] Munz C-D 1994 *Math. Methods Appl. Sci.* **17** 597
- [71] Munz C-D, Schneider R and Gerlinger O 1994 *Numerical Methods for the Navier–Stokes Equations* (Berlin: Springer) pp 181–90
- [72] Tsakiris N 2007 A vacuum tracking method for oscillating stars *Master's Thesis* Aristotle University of Thessaloniki
- [73] Rieutord M, Dubrulle B and Gourgoulhon E 2006 *Eur. Astron. Soc. Publ. Ser.* **21** 43
- [74] Friedman J L and Stergioulas N 2013 *Rotating Relativistic Stars* (Cambridge: Cambridge University Press)
- [75] Oliynyk T A 2019 arXiv:1907.08192
- [76] Christodoulou D and Lindblad H 2000 *Commun. Pure Appl. Math.* **53** 1536
- [77] Lindblad H 2005 *Ann. Math.* **162** 109
- [78] Jang J and Masmoudi N 2009 *Commun. Pure Appl. Math.* **62** 1327
- [79] Hadžić M, Shkoller S and Speck J 2019 *Commun. PDE* **44** 859
- [80] Ginsberg D 2019 *J. Hyperbolic Differ. Equ.* **16** 401

- [81] Martí J M, Ibáñez J M and Miralles J A 1991 *Phys. Rev. D* **43** 3794
- [82] Font J A, Ibáñez J M, Marquina A and Martí J M 1994 *Astron. Astrophys.* **282** 304
- [83] Spivak M 2018 *Calculus on Manifolds: A Modern Approach to Classical Theorems of Advanced Calculus* (Boca Raton, FL: CRC Press)
- [84] Toro E F 2013 *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction* (Berlin: Springer)
- [85] Shu C-W 1998 *Advanced Numerical Approximation of Nonlinear Hyperbolic Equations* (Berlin: Springer) pp 325–432
- [86] Harten A, Lax P D and Leer B v 1983 *SIAM Rev.* **25** 35
- [87] Gottlieb S, Ketcheson D I and Shu C-W 2009 *J. Sci. Comput.* **38** 251
- [88] Baumgarte T W and Shapiro S L 2010 *Numerical Relativity: Solving Einstein's Equations on the Computer* (Cambridge: Cambridge University Press)
- [89] Alcubierre M 2008 *Introduction to 3 + 1 Numerical Relativity (International Series of Monographs on Physics)* (Oxford: Oxford University Press)