Numerical simulations of interfaces in relativistic hydrodynamics

S T Millmore and I Hawke

Department of Mathematics, University of Southampton, Southampton SO17 1BJ, UK

Received 24 September 2009, in final form 28 October 2009
Published 15 December 2009
Online at stacks.iop.org/CQG/27/015007

Abstract
We consider models of relativistic matter containing sharp interfaces across which the matter model changes. These models will be relevant for neutron stars with crusts, phase transitions, or for viscous boundaries where the length scale is too short to be modelled smoothly. In particular we look at the numerical techniques that allow us to evolve stable interfaces, for the interfaces to merge, and for strong waves and shocks to interact with the interfaces. We test these techniques for ideal hydrodynamics in special and general relativity for simple equations of state, finding that simple level set-based methods extend well to relativistic hydrodynamics.

PACS numbers: 04.25.D−, 04.40.—b

(Some figures in this article are in colour only in the electronic version)

1. Introduction

Numerical simulations of compact objects such as neutron stars (NSs) are widely used to predict the nonlinear dynamics of, for example, binary merger or collapse [1]. Accurate and detailed simulations are thought necessary for constructing gravitational wave templates for use in detectors such as LIGO and VIRGO [2, 3]. The dynamics and wavesignal are in large part determined by the structure of the NS, which in turn is determined by the bulk equation of state (EOS) and the detailed microphysics in key, narrow, regions near the surface, crust and internal interfaces.

Most current models of NSs used in numerical simulations use single ideal perfect fluids (plus magnetic fields). When extending this model, the important physics may be different in specific regions of the NS (the surface and exterior, the crust and the core, regions of superconductivity or local phase transitions). Some effects, such as viscosity, will play a role only in localized regions. In all of these situations, there will be complex behaviour concentrated in a thin interface layer. The question is how these interfaces can be modelled accurately and practically in a numerical simulation.

Whether an interface can be simulated practically depends on the length scales involved, both in the underlying physics and in the simulation. In single fluid stars, the surface (interface
between fluid and vacuum) is genuinely sharp. Even in more realistic situations, the physical length scales may be extremely short. A more quantitative example is given by the Ekman layer in hot NS cores which is relevant for proto NS modelling. In this case, the length scale has been estimated by [4] to have the form $\delta \sim \left[ \frac{\eta}{(\rho\Omega)} \right]^{1/2}$ with the viscosity coefficient $\eta$ being highly temperature dependent, but with $\eta \approx 10^{13} - 10^{19} \text{ g cm}^{-1} \text{ s}^{-1}$ being reasonable for $T > 10^9 \text{ K}$. Working in cgs units and choosing $\eta \approx 10^{16} \text{ g cm}^{-1} \text{ s}^{-1}$ as a representative value leads to $\delta \sim 10^{-1/2} \Omega^{-1/2} \text{ cm}$ for $\rho \sim 10^{14} \text{ g cm}^{-3}$.

We then need to consider what physical length scales can be resolved in feasible numerical simulations. We will focus on finite difference simulations using current or near-future technology. We will concentrate on the number of timesteps, as we assume that a combination of local mesh refinement and parallelism will ensure that the spatial resolution can be made as small as required to resolve a given feature. Thus, the constraint on the length scales that can be resolved will arise from the length of time that we can simulate for, which will be determined by the number of timesteps required, and the given accuracy demanded. This timestepping issue is intrinsically serial—parallelism and local AMR cannot completely avoid it. We assume that it is practical to simulate for $N \sim 10^{10}$ timesteps and that we wish to cover 10 ms in physical time—sufficient for the late stages of a binary inspiral, or a few rotation periods of a NS. Then our finest timestep must be $\sim 10^{-12} \text{ s}$ which means, by the CFL criteria, that the smallest grid spacing must be $\sim 10^{-1} \text{ cm}$. Standard numerical methods smear interfaces and contact discontinuities with time [5] with the width of a sharp interface being spread over $N^{1/(p+1)}$ zones after $N$ timesteps, where $p$ is the order of accuracy of the method. Making the optimistic choice of $p = 4$, we see that an initially perfectly resolved interface will be smeared over $10^2$ zones with a physical width of $\sim 10 \text{ cm}$. By comparing with the order of magnitude estimates for the physical interface width given in the previous paragraph, we see that even with these optimistic assumptions the numerical simulation will be unable to distinguish numerical and physical effects for NSs with even moderate rotation rates.

This argument suggests that it is impractical and inefficient to model the detailed physics involved in these interface regions directly. However, we still need to incorporate these effects as best we can within our numerical simulations. To do this, we will model these interfaces as being genuinely sharp, that is, they have zero width. The additional physics will then be incorporated through the dynamical behaviour of the interfaces, and the boundary conditions imposed there.

In this paper, we will use a number of single ideal fluid components separated by sharp material interfaces. We will work in full general relativity using ideal hydrodynamics but restrict to 1 + 1 dimensions for simplicity. The model and a way to simulate it numerically is outlined in section 2. A specific numerical implementation is described in section 3. Finally we show a variety of tests illustrating the advantages and limitations of this simple implementation in section 4.

Throughout the paper, we use a signature of $(-,+,+,+)$ and work in geometric ($G = c = M_\odot = 1$) units.

2. Modelling sharp interfaces

2.1. The continuum model

The continuum model that we wish to simulate considers the spacetime as being composed of separate regions denoted $\Omega x$. On each region, the physical matter model (as described by a stress–energy tensor and equations of motion) may be different. Each region obeys the Einstein equations and any additional equations of motion required for the matter. In the
present paper, we restrict ourselves to vacuum or to single ideal relativistic perfect fluids with the stress–energy tensor

\[ T^{\mu\nu} = \rho_0 h u^\mu u^\nu + p g^{\mu\nu}, \]  

(2.1)

where \( \rho_0 \) is the rest-mass density, \( h \) is the specific enthalpy defined by

\[ h = \left(1 + \frac{p}{\rho_0} \right), \]  

(2.2)

with \( \epsilon \) the specific internal energy, \( p \) the pressure and \( u^\mu \) the 4-velocity.

The full spacetime dynamics is described by the union of all the regions \( \Omega_X \). At and across the boundaries \( \partial\Omega_X \), the Einstein equations must be satisfied, and from the results of [6], we note that this should not introduce discontinuities to invariant spacetime quantities, and hence we expect no additional problems from the spacetime. Therefore, all that remains to describe the model is to give the conditions satisfied by the matter model at the boundaries \( \partial\Omega_X \), and the conditions specifying the location of the boundaries. In what follows, we shall use a standard 3 + 1 splitting approach, so the spacetime is foliated into slices. The spacetime regions \( \Omega_X \) and their boundaries become spacelike volumes within the slice.

For this paper we consider the simplest possible situation for the boundary and interface conditions. We will assume that there is force balance at the boundary and no diffusion through it. We will assume that there is no surface tension or other non-ideal effect acting on or within the boundary. This means that the boundary will advect with the neighbouring fluid velocities, and the boundary condition will depend on the pressure normal to it.

2.2. Numerical model

As noted above, we only need to specify the boundary location and the conditions on the matter model at and across the boundary. We will first discuss the conditions applied to the location of the boundary.

2.2.1. Boundary location. In order to impose a boundary condition at \( \partial\Omega_X \), we first need to know where the boundary is. We assume that the location is known in the initial slice. We need to find the best method of identifying the location of the boundary at any later time. We note that this method must be able to deal with complex changes in topology of any interface, particularly in merger simulations.

The standard technique for dealing with sharp features such as shocks and interfaces in a single perfect fluid is to capture them. In this approach, the interface is smeared over a (hopefully small) number of points, with the appropriate solution found in the continuum limit. This is incompatible with the approach we are taking here. As well as the length scale issues discussed in the introduction, there are serious numerical problems when the model changes at the interface and the physics of the mixture are not taken into account, as illustrated in appendix A.

The alternative is to explicitly track or locate the interface. This is a standard problem in numerical relativity, as event and apparent horizons must be located in numerical simulations (for a review see [7]). Of particular relevance to us are the techniques used for event horizons (see for example [8]) where the interface is implicitly tracked and complex changes of topology can be dealt with. The techniques used borrow substantially from those in Newtonian computational fluid dynamics which are precisely used to deal with the problem we are interested in here (see [9, 10] for detailed descriptions).
Explicitly, we will use a level set to track the boundary \( \partial \Omega_X \). That is, we introduce a scalar function \( \phi \) defined over the entire spacetime. The location of the boundary is implicitly given by those points where \( \phi \) vanishes, i.e. by

\[
\partial \Omega_X = \{ x : \phi(x) = 0 \}.
\]  

(2.3)

The condition defining the location of the boundary \( \partial \Omega_X \) then becomes (after the 3 + 1 split) an evolution equation for the scalar function \( \phi \). This evolution equation is arbitrary except at the boundary. The equation can therefore be chosen for modelling simplicity or numerical convenience, provided that it reproduces the correct behaviour at the boundary.

In this paper, we consider the boundary as being advected with the fluid, i.e. it will be Lie dragged by the 4-velocity \( u^\mu \). The obvious evolution equation for the level set is

\[
L_u \phi = 0. 
\]  

(2.4)

However, the evolution of the level set can then lead to complex behaviour for \( \phi \) away from the boundary, as noted by, e.g. [8, 9]. This complexity is irrelevant to the physical behaviour. Methods for ‘renormalizing’ the level set (which changes the behaviour of \( \phi \) away from the boundary) are typically used to avoid problems—the standard methods shown by [11] and [9], and used by, e.g. [8] work in arbitrary dimensions. In 1 + 1 dimensions, and when only a single boundary is present, these problems can be avoided by using a constant vector instead of the fluid 4-velocity. Obviously this constant vector must be chosen to match the velocity at the boundary, i.e. equation (2.4) becomes

\[
L_{u|_{\partial \Omega_X}} \phi = 0.
\]  

(2.5)

2.2.2. Boundary condition. To complete the description of the model, we need to describe the conditions imposed at the boundary. There are a number of properties that we would like these conditions to satisfy. We need the conditions to be compatible with the continuum model. We would like the conditions to approximate, as closely as possible, the additional microphysics. We need the boundary conditions to be practical in a numerical simulation; in particular, it must be possible to use them without introducing unphysical oscillations due to any discontinuities present. We would like the boundary conditions to be as simple as possible.

The condition that we study in this paper is the simplest successful condition proposed in Newtonian computational fluid dynamics; the Ghost Fluid condition proposed by [11]. It is intended to model the simple situation described above; two ideal fluids separated by an interface with no diffusion. It is simple and practical to implement, but is not correct in all circumstances; studies such as [12, 13] have shown that incorrect shock structures can be computed when strong shocks hit the interface between extremely different EOSs. We use the Ghost Fluid boundary condition to illustrate the general approach.

The Ghost Fluid approach considers the fluid on each spacetime region \( \Omega_X \) separately. To impose boundary conditions at \( \partial \Omega_X \), the fluid describing the matter within the spacetime region \( \Omega_X \) is artificially extended through the interface into the neighbouring region(s). This is analogous to the imposition of boundary conditions using ghost zones or ghost points. The boundary conditions are then imposed by providing values for the artificially extended fluid. In this approach a condition is given on the pressure, the velocity and the entropy, from which all other quantities can be recovered (given an EOS).

The precise conditions imposed at the interface are the continuity of the pressure and the normal velocity, and that the flow is isentropic. In principle, for a material interface, we would expect the pressure and normal component of the velocity to be continuous; entropy and the tangential velocity may jump. To follow this as closely as possible, the pressure and
normal velocity component are copied from the fluid in the neighbouring region $\Omega_1$, whilst the entropy and tangential velocity are extrapolated. This is illustrated in $1+1$ dimensions in figure 1.

3. Implementation

3.1. Formulation

3.1.1. Hydrodynamics—test fluid, Minkowski spacetime. In order to best compare with standard relativistic hydrodynamics simulations, we use the conserved formulation of [14]. In $1+1$ Minkowski spacetime using Cartesian coordinates and the test-fluid approximation, this is written as

$$\partial_t \mathbf{q} + \partial_x \mathbf{f}(\mathbf{q}) = \mathbf{0},$$

where the conserved variables $\mathbf{q}$ and fluxes $\mathbf{f}$ are given in terms of the primitive variables introduced in section 2 by

$$\mathbf{q} = (D, S_x, \tau)^T = (\rho_0 W, \rho_0 h W^2 v_x, \rho_0 h W^2 - p - \rho_0 W)^T,$$

$$\mathbf{f} = (Dv^x, S_x v^x + p, (\tau + p)v^x)^T,$$

where we have introduced the 3-velocity

$$v^i = \frac{1}{\alpha} \left( \frac{u^i}{u^0} + \beta^i \right) = \frac{u^i}{u^0},$$

where the shift $\beta^i$ and lapse $\alpha$ resulting from the $3+1$ split take the trivial values in this case. We have also introduced the Lorentz factor

$$W = (1 - v_i v^i)^{-1/2}.$$
To close the system, an EOS is required. In this paper, we use the simple γ-law EOS
\[ p(\rho_0, \epsilon) = (\gamma - 1)\rho_0 \epsilon, \] (3.5)
where \( \gamma \) is the ratio of specific heats. Different regions of spacetime may have different EOSs which is modelled simply by providing different values of \( \gamma \).

The level set equation (2.4) becomes
\[ \partial_t \phi + v^i \partial_i \phi = 0. \] (3.6)
This equation is not in the conservation law form. Whilst it can be written in the conservation law form, this would have no advantage, as it is only the zeros of the level set function that have meaning. The non-conservative form of equation (3.6) is evolved directly.

3.1.2. Hydrodynamics—\( 1+1 \) spherical spacetime. In \( 1+1 \) spherical symmetry, we follow, e.g. [15–17] in using polar-areal coordinates. However, in order that we can straightforwardly extend this work to higher dimensions, we use the formulation of [14], which leads to slight differences of notation. The line element is written as
\[ ds^2 = -\alpha^2(t, r) dt^2 + a^2(t, r) dr^2 + r^2(d\theta^2 + \sin^2\theta d\phi^2). \] (3.7)
The 3-velocity becomes
\[ v^i = \frac{u^i}{a\alpha}. \] (3.8)
The conservation law form, when written to match the 3+1 form of [14], is
\[ \partial_t (aq) + \frac{1}{r^2} \partial_r (aq f(q)) = s(q), \] (3.9)
where the variables, fluxes and sources are
\[ q = (D, S_r, \tau)^T, \]
\[ f = (Dv^r, S_r v^r + p, (\tau + p) v^r)^T, \]
\[ s = aa \left(0, -\frac{a^2 m}{r^2} (S_r v^r + \tau + p + D) + \frac{2p}{r} + \frac{m}{r^2} S_r\right)^T, \] (3.10)
where the spacetime quantity \( m = \frac{\tau}{r} (1 - \alpha^{-2}) \) is the mass aspect function. The spacetime quantities satisfy the equations
\[ \partial_r a = -4\pi raa S_r, \] (3.11a)
\[ \partial_r a = a^3 \left[4\pi r (\tau + D) - \frac{m}{r^2}\right], \] (3.11b)
\[ \partial_r \log(\alpha) = a^2 \left[4\pi r (S_r v^r + p) + \frac{m}{r^2}\right]. \] (3.11c)
We will evolve \( a \) using equation (3.11a) and solve for the lapse using the slicing condition (3.11c), whilst the error in the Hamiltonian constraint (3.11b) will be used to check the accuracy of the simulations.

To reduce problems near the coordinate singularity at the origin, the equations are rewritten as
\[ \partial_t (aq) + 3\partial_i (aa r^2 f^{(1)}(q)) + \partial_i (aa f^{(2)}(q)) = \bar{s}(q), \] (3.12)
where the revised fluxes and sources are
\[ f^{(1)} = (Dv', S_v v', (\tau + p)v')^T, \]
\[ f^{(2)} = (0, p, 0)^T, \]
\[ s = \alpha a \left( 0, -\frac{a^2 m}{r^2} (S_v v' + \tau + p + D), -\frac{m}{r^2} S_v \right)^T. \]

The level set equation (2.4) becomes
\[ \partial_t \phi + \alpha v' \partial_r \phi = 0. \]  

As in flat space, this non-conservative form is evolved directly.

### 3.2. Numerical methods

To evolve in time, the method of lines is used with a second order Runge–Kutta algorithm.

#### 3.2.1. Hydrodynamics

To illustrate the utility of the approach and the Ghost Fluid boundary condition we have used standard high-resolution shock capturing (HRSC) methods to solve the hydrodynamics equations (3.1) and (3.12). The domain is split into cells, and the conserved variables are updated using
\[ \frac{d}{dt} \tilde{q}_i = \frac{1}{\Delta x_i} [\tilde{f}_{i-1/2} - \tilde{f}_{i+1/2}]. \]

Following, e.g. [18], we concentrate on reconstruction–evolution methods, where the variables are reconstructed to cell boundaries and a Riemann problem (approximately) solved to find the inter-cell flux \( \tilde{f}_{i+1/2} \). For reconstruction, we use slope-limited TVD methods with van Leer’s monotized-centred (MC) limiter [19] or the PPM method [20, 21]. For simplicity, we use the HLLE flux formula [22] in what follows, although we have seen no qualitative difference when the Marquina flux formula [23] is used instead.

#### 3.2.2. Level set

The evolution of the level set (equation (3.6) or (3.14)) cannot use the same numerical methods as the equation is not in the conservation law form. Instead we shall follow the methods outlined in [9], evolving the level set equation as a Hamilton–Jacobi equation. General high-order methods for evolving Hamilton–Jacobi equations can be found, for example, in [24].

The general method for Hamilton–Jacobi equations of the form
\[ \partial_t \phi + H(\partial_x \phi) = 0 \]
that works even when the level set \( \phi \) is not differentiable is to solve
\[ \frac{d}{dt} \phi_i + \hat{H} \left( D^+_x \phi_i, D^-_x \phi_i \right) = 0, \]
where \( \phi_i \) is the numerical value of \( \phi \) at gridpoint \( x_i \), \( \hat{H} \) the numerical Hamiltonian, assumed Lipschitz continuous, and \( D^+_x \) appropriate winded approximations of \( \partial_x \phi \).

It is simple to use ENO or WENO methods to find the appropriate \( D^+_x \) operators (explicit expressions for low order are given by [9]). In the \( 1+1 \) examples studied here, it has been sufficient to use first order accurate upwind differencing. We would not expect this to be sufficient in more complex examples or in higher dimensions.

We have also used the simple Lax–Friedrichs approximation to the Hamiltonian
\[ \hat{H}(u_+, u_-) = H \left( \frac{u_+ + u_-}{2} \right) - \hat{\alpha} \left( \frac{u_+ - u_-}{2} \right), \]
where $\hat{a}$ is the maximum absolute value of $H$ over the entire domain. In higher dimensions it may be costly to compute $\hat{a}$ over the whole domain, so a local Lax–Friedrichs method (where the maximum value of $\hat{a}$ over the numerical stencil is used) can be used instead.

### 3.2.3. Ghost Fluid boundary condition

Figure 1 has already been given to illustrate how the Ghost Fluid boundary condition is implemented. To be explicit, we consider a numerical grid $\{x_i\}$ on which the level set $\phi_i$ changes sign between points $I, I+1$. We apply the Ghost Fluid boundary condition to fluid 1, which covers the domain to the left of the interface, using where necessary information from fluid 2, which covers the domain to the right of the interface. We will apply the boundary condition to at least $p+1$ ‘ghost’ points extending from $x_{I+1}$ to $x_{I+p+1}$, where $p$ is the stencil size of the numerical method. The reason for the use of $p+1$ ghost points is that at least one additional point is needed if the interface should move one point to the right during the evolution step.

The normal component of the velocity and the pressure of fluid 1, $v^{(1)}$ and $p^{(1)}$ respectively, are copied from fluid 2. The entropy is extrapolated from fluid 1. In the Ghost Fluid method, zero-order extrapolation normal to the interface is used. In the 1 + 1-dimensional cases here, the extrapolation is a simple copy. Explicitly we have

\[
\begin{align*}
v_j^{(1)} &= v_j^{(2)}, & j &= I+1, \ldots, I+p+1, \\
p_j^{(1)} &= p_j^{(2)}, & j &= I+1, \ldots, I+p+1, \\
s_j^{(1)} &= s_j^{(1)}, & j &= I+1, \ldots, I+p+1.
\end{align*}
\]

All other hydrodynamical quantities are found from the pressure, entropy and the EOS.

### 3.2.4. The evolution step

In simple cases such as flat space or where free evolution of spacetime quantities is used, the evolution step is now simply a matter of updating both the fluid and level set. However, should key spacetime quantities (such as $a$ for the case given in section 3.1.2) be updated using non-local equations (for example, the constraint (3.11b) could be used to give $a$) then the order of the update becomes important. For this reason, we ensure that we update the level set using the methods of section 3.2.2 before the hydrodynamical quantities can be updated using the methods of section 3.2.1.

We represent a complete evolution step, assuming a first-order time-stepping algorithm as follows.

1. Given data $\{\hat{q}_n^i\}$ for the hydrodynamical variables and $\{\phi_n^i\}$ for the level set and timestep $n$.
2. Find interface locations $\{x^i_n : \phi = 0\}, j = 0, \ldots, N^n$, where $N^n - 1$ is the number of interfaces at time $t^n$. For simplicity, we denote the boundaries of the computational domain by $x^0_n$ and $x^{N^n}_n$, respectively.
3. If there is only one genuine interface, approximate the velocity $v$ there.
4. Compute the updated level set $\phi^{n+1}$, either using the interface velocity approximated in the previous step, or using the velocity of the fluid at each point.
5. Compute the new interface locations $\{\phi_j^{n+1} : \phi = 0\}, j = 0, \ldots, N^{n+1}$.
6. For each domain $\Omega^j_n = [x_j^n, x_{j+1}^n], j = 0, \ldots, N^n - 1$:
   - (a) Check the domain still exists at the next time step, $\Omega_j^{n+1} \neq \emptyset$. If it does not, go to the next domain.
   - (b) Apply the Ghost Fluid boundary condition to both boundaries, extending the domain by a sufficient number of points, as given in section 3.2.3.
(c) Compute the inter-cell fluxes on this extended domain.
(d) Restrict the updated variables to the domain $\Omega_{n+1} = [x_n^{n+1}, x_{n+1}^{n+1}]$.
(e) Compute primitive variables and apply physical or symmetry boundary conditions as required.

Extending this to higher order in time is trivial with method of lines type methods.

4. Tests

The tests used here are intended to see whether the Ghost Fluid method works in relativistic situations as it does in Newtonian theory. This will indicate whether this is a viable approach for numerical simulations of more complex models of NS interactions such as mergers. The simplest test is a check that the method would keep a single non-interacting material interface stable. This is provided in appendix A as part of an illustration of why capturing an interface is not straightforward numerically, as discussed in section 2.2.1.

We start by looking at the interaction between strong nonlinear features and interfaces in flat space, using the model of section 3.1.1. These ensure that no spurious oscillations are introduced and see how accurately the method captures transient and long-lived smooth features. We then move on to full GR tests, where we can assess the accuracy and effectiveness with simple spherically symmetric NS models, both stable and nonlinearly perturbed.

4.1. Flat space tests

4.1.1. Shock–interface interaction. Two types of test are considered. Unless otherwise stated the domain is $x \in [0, 1]$. In the first a stable material interface is set up in the middle of the domain across which the EOS changes due to a change in $\gamma$. A shock is injected from the left edge of the domain. By time $t = 1$, the shock has interacted with the interface, being reflected and transmitted. In the mild version of this test, which is a modified relativistic analogue of test B of [11], the precise initial data are

$$\begin{align*}
\rho_0 &= 1.3346, \quad v = 0.1837, \quad p = 1.5, \quad \gamma = 1.4 \quad x < 0.05 \\
\rho_0 &= 1, \quad v = 0, \quad p = 1, \quad \gamma = 1.4 \quad 0.05 \leq x < 0.5 \\
\rho_0 &= 0.1379, \quad v = 0, \quad p = 1, \quad \gamma = 1.67 \quad x \geq 0.5.
\end{align*}$$

To illustrate the results when a stronger shock is injected, we also look at the initial data

$$\begin{align*}
\rho_0 &= 10.2384, \quad v = 0.9411, \quad p = 50, \quad \gamma = 1.4 \quad x < 0.05 \\
\rho_0 &= 1, \quad v = 0, \quad p = 1, \quad \gamma = 1.4 \quad 0.05 \leq x < 0.5 \\
\rho_0 &= 0.1379, \quad v = 0, \quad p = 1, \quad \gamma = 1.67 \quad x \geq 0.5.
\end{align*}$$

In this case, the domain is $x \in [0, 3]$ and the test is run to $t = 2$ in order to clearly separate the transmitted shock and the interface.

Results from the milder test are shown in figure 2. For simplicity, we use TVD reconstruction with the HLLE Riemann solver. The transmitted shock and reflected rarefaction are clearly seen, with the results converging towards the exact solution. The material interface has propagated to the right due to the interaction, and this is also captured accurately. No spurious oscillations have been generated either at the shock or at the material interface.

Results for the very strong shock case are shown in figure 3. The qualitative features are the same as in the milder case. However, the higher shock velocity leads to a problem with performing this test at lower resolution. The ‘start-up’ error (see, e.g. [25] for an explanation...
Figure 2. Results from an isolated shock hitting a contact discontinuity (‘test B’ from [11]). The transmitted shock is very obvious in the velocity and pressure profiles, and convergence of the solution at this point can be seen. The reflected rarefaction wave is considerably smaller, but can still be seen. The contact discontinuity, shown by the dashed line, is captured correctly. TVD reconstruction using the MC limiter is used, and only 100 points are shown in each plot.

of this purely technical problem) from the discontinuous initial data is much stronger than in the mild case, and it follows the leading shock extremely closely. As the coordinate separation between the transmitted shock and the interface is so small, due to the high shock velocity, the trailing start-up error continues to interact with the interface leading to significant distortion of the result. In order to minimize this problem, which would not appear from realistic initial data, we show results computed with sufficiently high resolution that the start-up error can be removed. Here it is clear that the method continues to work well even with high-velocity shocks.

A considerably more difficult test that does not have an exact solution was suggested by [26]. This has a ‘slab’ of helium/air mixture initially at rest in air. A moderate shock then
strikes the slab, with the resulting shocks and rarefaction interacting multiple times. This is a 1d restriction of the shock–bubble interaction used in the original Ghost Fluid paper \[11\]. The relativistic analogue used here has initial data

\[
\begin{align*}
\rho_0 &= 1.37795, \quad v = 0.17933, \quad p = 1.57, \quad \gamma = 1.4 \quad x < 0.25 \\
\rho_0 &= 1, \quad v = 0, \quad p = 1, \quad \gamma = 1.4 \quad 0.25 \leq x < 0.45 \\
\rho_0 &= 0.138, \quad v = 0, \quad p = 1, \quad \gamma = 1.67 \quad 0.45 \leq x < 0.55 \\
\rho_0 &= 1, \quad v = 0, \quad p = 1, \quad \gamma = 1.4 \quad x \geq 0.55.
\end{align*}
\]

Figure 3. Results from a strong isolated shock ($v \simeq 0.94$) hitting a contact discontinuity, extending the test shown in figure 2. The main difference with the milder test of figure 2 is the extremely narrow (coordinate) gap between the transmitted shock and the interface, which by $t = 2$ has moved a considerable distance across the domain; this necessitates the high resolutions used here to see each separate wave clearly. TVD reconstruction using the MC limiter is used, and only 200 points are shown in each plot.
The test is run to $t = 0.8$.

Results for this more complex interaction are shown in figure 4. TVD reconstruction using the MC limiter is used. The strong interacting features are captured, with the interfaces moving independently in reaction to the shocks. No oscillations occur, and the result clearly approaches the reference solution as the resolution is increased.

4.1.2. Perturbed shock. A more complex example is that of a shock perturbed by a strong, smooth feature. This test, suggested by [27] and used, e.g. by [28], does not have an exact
solution, so it is compared to a reference solution computed with 128000 points. We modify this test to include an interface, using the initial data

$$\begin{align*}
\rho_0 &= 5, \quad v = 0, \quad p = 50, \quad \gamma = 1.4 \quad x < 0.5 \\
\rho_0 &= 2 + 0.3 \sin(50x), \quad v = 0, \quad p = 5, \quad \gamma = 1.67 \quad x \geq 0.5.
\end{align*}$$

Again the domain is $x \in [0, 1]$ and the test is run to time $t = 0.35$.

Results for this test are shown in figures 5 and 6. It is clear that the results are converging to the correct solution irrespective of the numerical method used. No spurious oscillations are introduced. However, it appears that there is a slight discrepancy at the material interface. In particular, the edges of the smooth feature do not appear quite correct. We investigate this with another test.

Figure 5. Results for a test with a shock tube test with a sine wave density profile for 200 points. The higher accuracy when using PPM is evident in the post shock density profile, with the sine wave being captured more accurately. The solid line shows a reference solution computed using 128000 points.
4.1.3. Moving sine wave. A final flat space test is adapted from a Newtonian test suggested by [29]. A sine wave is advected in a surrounding flow. The material interfaces should be stable, and so the evolution should be trivial. The initial data used here is

\[
\begin{align*}
\rho_0 &= 1, & v &= 0.5, & p &= 1, & \gamma &= 1.4 & x < 0.16 \\
\rho_0 &= 1 + 0.3 \sin(50(x - 0.16)), & v &= 0.5, & p &= 1, & \gamma &= 1.67 & 0.16 \leq x < 0.537 \\
\rho_0 &= 1, & v &= 0.5, & p &= 1, & \gamma &= 1.4 & x \geq 0.537.
\end{align*}
\]

Again the domain is \( x \in [0, 1] \) and the test is run to time \( t = 0.4 \).

Results for this test are shown in figures 7 and 8. The majority of the features are well captured and no spurious oscillations are introduced at the material interfaces. It is clear, however, that the accuracy suffers near the interfaces due to the strong gradients there, and the overly simple Ghost Fluid boundary condition. Measuring the 1-norm of the errors in density...
Figure 7. An advected sine wave, evolved with a resolution of 200 points. Since the initial density profile is simply advected with the fluid velocity, an exact solution exists for this test. The differences between the two reconstruction methods are evident, with the PPM test more accurately capturing the sine wave.

Figure 8. An advected sine wave, evolved with a resolution of 800 points. The sine wave is now captured well with both techniques, though at the interface between the fluids there is still some error. Comparison with figure 7 shows the convergence of this error, though this convergence is slow (with an $\infty$-norm convergence order of 0.7 for both reconstruction methods).

gives a convergence order of 1.5 for the MC Limiter, and 1.7 for PPM. The $\infty$-norm for both reconstruction methods gives a convergence order of 0.7, clearly dominated by the errors at the interface. We will see in the next section that in simple GR simulations modelling neutron stars this is not a major concern.

4.2. GR tests

In this section, we look at tests in nonlinear general relativity, focusing on tests relevant for neutron stars. We use the system of equations outlined in section 3.1.2. All of the tests look at static or nonlinearly perturbed TOV stars in spherical symmetry. The initial data are always constructed with a polytropic EOS

$$p = K \rho^\gamma,$$

(4.6)

where $K$ is the polytropic constant and evolved with the ideal fluid EOS of equation (3.5), where $\gamma$ may vary in space.
4.2.1. Static stars. As a reference solution, we construct and evolve a standard static single-component star. The initial data are determined by the initial central density \( \rho_0(t = 0, r = 0) \) denoted \( \rho_c \), and the EOS. Here we use the values

\[
\rho_c = 1.28 \times 10^{-3}, \quad \gamma = 2, \quad K = 100. \tag{4.7}
\]

We then evolve this using the TVD method as described in section 3.2. The errors in the evolution are shown in figure 9. Second-order convergence is seen over the majority of space at late time. However, near the surface of the star convergence is lost, both due to the ‘clipping’ inherent in HRSC methods, and due to the atmosphere algorithm used here. At the first point below a sufficiently small density or pressure, all exterior points are set to a stationary \((v = 0)\), small polytropic atmosphere value. This allows for stable simulations without significantly affecting the interior evolution, but is clearly artificial. This is the dominant factor in reducing the order of convergence of the norms.

As a first test of the accuracy of the Ghost Fluid algorithm in GR, we then introduce a ‘trivial’ interface into the TOV star at coordinate location \( r = 3.015 \). On either side of this interface, the EOS is the same as that given in equation (4.7). We then evolve, imposing the Ghost Fluid boundary condition at the interface. It should be noted that in this simulation, although the level set is evolved, the interface inferred from the zero of the level set remains within the same grid cell at all times.

The results of this test are practically indistinguishable to the reference solution of figure 9, indicating that the Ghost Fluid method makes virtually no difference to the accuracy of the results. The convergence properties are nearly identical, and the effect of the interface is very difficult to detect in the errors. We can in addition check the effect of the interface on the radial 3-velocity component, which is clearly continuous across the interface and affected
only slightly by minor reflections between the surface and the interface. In addition, we can look at the change in the entropy near the interface, where the jump caused by the boundary condition is negligible at the $\sim 10^{-7}$ level.

We then move on to look at static stars containing a genuine interface between two different components. We have studied a number of different configurations, with the qualitative behaviour being similar in all cases. As an illustration, we use the initial data

$$
\begin{aligned}
\rho_c &= 6 \times 10^{-4}, \quad \gamma = 2, \quad K = 100, \quad r < 3.015 \\
\gamma &= \frac{5}{3}, \quad K = 11.17, \quad r \geq 3.015.
\end{aligned}
$$

(4.8)

This ensures the TOV is stable, and that pressure and its first derivative are continuous at the interface. These initial data are illustrated in figure 10.

The result of the evolution is shown in figure 11. In this case, we do see an effect in the Hamiltonian constraint from the interface at $r = 3.015$. This leads to larger errors within the star (at fixed resolution). However, these errors appear to converge at the second order (except for the points directly next to the interface). This means that, for these resolutions, the convergence of the constraint error is actually better than for the reference solution shown in figure 9. This is because the error at the surface, which does not converge at second order due to the atmosphere algorithm and the inherent properties of HRSC methods, contributes less (relatively) to the total error. In contrast, the errors in the density are (relatively) larger at the surface as we are using a lower central density and a softer EOS for the outer fluid. Hence, at late times as the surface errors increase, the effects are seen in the errors of the density, which drifts away from perfect second order convergence.

4.2.2. Perturbed star. The tests in section 4.2.1 show that there is no problem in extending the level set method augmented with the Ghost Fluid boundary condition to GR when studying simple solutions. However, we really require that these methods will work in strongly nonlinear
situations which will include interface motion and shock formation and interaction. In order to look at this in a 1+1 context, we take a two-component TOV star and give a strong nonlinear perturbation designed both to move the interface and to produce shocks propagating within the fluid. Due to the limitations of our atmosphere algorithm, the results will not be trustworthy when the shocks start to reach the surface of the star, so we keep the evolutions relatively short. However, these evolutions are sufficient to validate the approach that we use here.

The initial data used are based on a two-component TOV star with parameters

\[
\rho_i = \begin{cases} 
1.28 \times 10^{-3}, & y = 2, \quad K = 100, \quad r < 3.015 \\
1.9, & y = 1.9, \quad K = 51.57, \quad r \geq 3.015.
\end{cases}
\]  

(4.9)

The interior is then perturbed using

\[
\rho_0 = (\rho_i)_{\text{TOV}}(1 + h(r)), \quad p = p_{\text{TOV}}(1 + h(r)),
\]

(4.10)

where

\[
h(r) = \begin{cases} 
\frac{1}{20}(1 - \tanh[50(r - 2)]) & r < 2.5, \\
0 & r \geq 2.5.
\end{cases}
\]

(4.11)

This is illustrated in the left panel of figure 12. The addition of a large amount of matter near the origin means that the spacetime is no longer in equilibrium. In particular, it means that the outer, unperturbed, layers should initially fall in, as the enclosed gravitational mass has been increased, modifying the spacetime in the unperturbed region. However, for our purposes the main point of the strong perturbation is that it leads to a very steep feature near \( r \approx 2 \) that will form a shock and move the interface. As seen in the right panel of figure 12, there is an additional shock formed by the ‘ingoing’ piece of the initial perturbation reflecting from the origin; at the time of this plot, it has not reached the interface.

Figure 11. Errors introduced in evolving a two-component star. This should be compared to the reference solution in figure 9. The interface at \( r \approx 3 \) now has a visible effect in the errors, but these are still manageable and converge. Note that at these resolutions the convergence of the norm of the constraints is actually better than in the reference solution, as the errors in the interior (which converge at second order) are larger due to the non-trivial interface, whilst those near the surface (which converge at less than second order) are smaller due to the softer EOS. The errors in the density show good convergence until late times when the errors near the surface become sufficiently large.
Figure 12. The left panel shows the initial data for the nonlinearly perturbed two-component star. The interior perturbation near $r \simeq 2$ is not quite a shock but will trigger an outgoing wave that steepens into a shock before passing through the interface at $r \simeq 3$. It will also produce a strong ingoing wave that will reflect from the origin, producing another shock. The right panel shows the evolved density and pressure at a time when the initial outgoing shock has passed through the interface but before the reflected shock has had time to reach it.

Figure 13. Constraint errors in the evolution of the nonlinearly perturbed star. The curves are scaled for second-order convergence, which is seen away from the surface and the shocks. Reasonable convergence is seen at a resolution of 80 points, of which $\simeq 40$ points cover the NS, from which the accuracy of these simple methods in current 3D simulations can be inferred.

The constraint errors in the evolution of the nonlinearly perturbed star are shown in figure 13. As expected, we now see a large number of features indicated on the plot. As
in the reference solution in figure 9 we see the error from the surface, which is sizeable and does not converge cleanly at second order. As in the static two-component test in figure 10, we see an error at the interface, which in this case is relatively small and converges except at a few points directly neighbouring the interface. Finally we see strong features at the two shocks where, due to the nature of HRSC methods, the order of accuracy is degraded to first order.

Figure 14 shows the motion of the interface and also shows that no spurious oscillations occur due to the interaction between the outgoing shock and the interface. The left panel shows the velocity at a time when the outgoing shock has passed through the interface located at $r \approx 3$. The velocity is continuous with no spurious oscillations, despite the interaction with the shock. The right panel shows the motion of the interface. Initially it falls in with the bulk motion of the unperturbed matter, due to the interior perturbation modifying the spacetime, as noted above. It then interacts with the shock, being pushed outwards and jumping between cells, before starting to fall back again, with the additional interaction with the reflected shock at $t \approx 18$ being insufficient to stop its inward motion.

5. Conclusions

In this paper, we have considered models of relativistic matter containing sharp interfaces and their particular importance and implementation for numerical simulations. We expect that simulations of realistic models of neutron stars, as required for gravitational wave template calculations, will contain features that will be too sharp to be resolved as a smooth feature using reasonable numerical resolution. It follows that models incorporating sharp interfaces and techniques for evolving them cleanly will be required.
The use of level set techniques to capture and evolve the interfaces that might form and disappear through mergers is already well known in numerical relativity, particularly in the context of event horizon finders. Their use here is natural and simple, and should allow the techniques investigated here to easily be extended to more spatial dimensions.

However, the description of the interface location is not sufficient. We must impose a boundary condition at the interface, which will complete the description of the continuum model. The Ghost Fluid boundary condition used here has the advantage of simplicity, which makes it ideal for a proof-of-principle test such as this. However, it is questionable if this condition is sufficient for all situations, and more advanced techniques may be required with more complex EOSs than those considered here.

The results of section 4 illustrate the advantages of the combined level set and Ghost Fluid method. It is simple, and for strong shocks or smooth flow interacting with interfaces it gives good results. We have shown that it directly extends to GR, and that for mildly perturbed neutron star tests the results are stable, convergent and as accurate as the numerical method employed. However, when there is a long-lasting, strong feature that interacts with the interface, the test shown in section 4.1.3 illustrates the limitations of the technique. This may indicate the need for better boundary conditions at the interface before high accuracy simulations for gravitational wave extraction can be attempted for complex neutron star models.

Acknowledgments

IH was supported by the STFC rolling grant PP/E001025/1 and Nuffield Foundation grant NAL/32622. SM was supported by an EPSRC studentship. This work was supported by CompStar, a Research Networking Programme for the European Science Foundation.

Appendix A. Colour functions

At first glance, it would appear that the simple tests shown in this paper are solved with an excessively complex numerical method. From the simplest point of view, our system of equations describing relativistic hydrodynamics has simply been augmented with an equation of motion for the EOS parameter $\gamma$. This can be described by a ‘colour function’ [25] and advected using standard HRSC methods if written in the form

$$\partial_t (\rho_0 \gamma) + \partial_i (\rho_0 \gamma v^i) = 0.$$  (A.1)

In fact any parameter $\kappa$ required by the EOS can be advected using an equation of this form.

However, it is well known in the computational fluid dynamics literature that this naive approach fails for the simplest interfaces. Figure A1 shows this approach failing for a stable material interface. The interface is set at $x = 0.5$, with initial data

$$(\rho_0, v, p, \gamma) = \begin{cases} (1, 0.1, \frac{2}{3}, \frac{5}{3}) & x < 0.5, \\ (\frac{1}{2}, 0.1, \frac{2}{3}, \frac{4}{3}) & x \geq 0.5, \end{cases}$$  (A.2)

leading to a stable interface propagating to the right. The standard conservation law form, as in section 3.1.1, augmented with the colour function equation (A.1), is evolved using the simple TVD HRSC method used in the rest of the paper. This is considerably simpler than the level set method, but as can be seen from the figure fails utterly to deal with this simple interface. Spurious oscillations are introduced. These oscillations are sizeable and do not converge with resolution. Clearly there is a failure of the approach, which is due to the formulation of the
A stable contact discontinuity, initially at $x = 0.5$, with the fluid to the left having $\gamma = 5/3$ and the fluid to the right having $\gamma = 4/3$ is advected to the right with speed 0.1. The initial data are set such that the interface should propagate unchanged. The plot shows this initial data evolved using two resolutions and standard HRSC methods to time $t = 0.2$ using the colour function approach, with the exact solution given by the solid line. Spurious oscillations are introduced at the interface. These oscillations do not converge with resolution, indicating a failure of the approach.

The reasons for the failure of the colour function approach have been studied in detail by a number of authors. As explained by [30], the key is that the inherent ‘smearing’ of the HRSC scheme will effectively ‘mix’ the fluids either side of the interface in a narrow region. The detailed way that the mixing occurs will depend on the resolution and the numerical method, but it will always occur. However, without imposing further conditions it is clear that this mixing will not be thermodynamically consistent. That is, the numerical mixing will lead to spurious generation of entropy at the interface, creating the oscillations observed. Whilst it is possible to construct numerical methods that are thermodynamically consistent (e.g. [26]; for
example, for the EOS used here evolving the EOS parameter $\kappa = 1/(\gamma - 1)$ is consistent, as shown by [30]), this may not always be straightforward, and the problem of the length over which the interface is smeared, as described in the introduction, remains.

References

[1] Font J A 2008 Living Rev. Relativ. 11 http://www.livingreviews.org/lrr-2008-7
[2] Abbott B P et al 2009 Rep. Prog. Phys. 72 076901
[3] Acierno F et al 2006 Class. Quantum Grav. 23 S635
[4] Andersson N, Comer G L and Glampedakis K 2005 Nucl. Phys. A 763 212
[5] Harten A 1978 Math. Comput. 32 363
[6] Barnes A P, Lefloch P G, Schmidt B G and Stewart J M 2004 Class. Quantum Grav. 21 5043
[7] Thornburg J 2007 Living Rev. Relativ. 10 http://www.livingreviews.org/lrr-2007-3
[8] Diener P 2003 Class. Quantum Grav. 20 4901
[9] Osher S and Fedkiw R 2003 Level Set Methods and Dynamic Implicit Surfaces (Applied Mathematical Sciences vol 153) (Berlin: Springer)
[10] Sethian J 1996 Level Set Methods and Fast Marching Methods (Cambridge: Cambridge University Press)
[11] Fedkiw R P, Aslam T, Merriman B and Osher S 1999 J. Comput. Phys. 152 457
[12] Liu T G, Khou B C and Yeo K S 2003 J. Comput. Phys. 190 651
[13] Liu T, Khou B and Wang C 2005 J. Comput. Phys. 204 193
[14] Banyuls F, Font J A, Ibáñez J M, Martí J M and Miralles J A 1997 Astrophys. J. 476 221
[15] Romero J V, Ibáñez J M, Martí J M and Miralles J A 1996 Astrophys. J. 462 839
[16] Neilsen D W 1999 PhD thesis The University of Texas at Austin
[17] Noble S C 2003 PhD thesis The University of Texas at Austin
[18] Hawke I, Löffler F and Nerozzi A 2005 Phys. Rev. D 71 104006
[19] van Leer B J 1973 Lecture Notes in Physics 18 163
[20] Colella P and Woodward P R 1984 J. Comput. Phys. 54 174
[21] Martí J M and Müller E 1996 J. Comput. Phys. 123 1
[22] Einfeldt B 1988 SIAM J. Numer. Anal. 25 294
[23] Donat R and Marquina A 1996 J. Comput. Phys. 125 42
[24] Shu C W 1999 High-Order Methods for Computational Physics ed T J Barth and H Deconinck (Berlin: Springer)
[25] Leveque R 2002 Finite Volume Methods for Hyperbolic Problems (Cambridge: Cambridge University Press)
[26] Wang S-P, Anderson M H, Oakley J G, Corradini M L and Bonazza R 2004 J. Comput. Phys. 195 528
[27] Dolezal A and Wong S S M 1995 J. Comput. Phys. 120 266
[28] Del Zanna L and Bucciantini N 2002 A&A 390 1177
[29] Lombard B and Donat R 2005 SIAM J. Sci. Comput. 27 208
[30] Abgrall R and Karni S 2001 J. Comput. Phys. 169 594