QUOKKA: A code for two-moment AMR radiation hydrodynamics on GPUs

Benjamin D. Wibking[1,2]⋆ and Mark R. Krumholz[1,2]
[1]*Research School of Astronomy & Astrophysics, Mount Stromlo Observatory, Cotter Road, Weston Creek, ACT 2611 Australia*
[2]*ARC Centre of Excellence for Astronomy in Three Dimensions (ASTRO-3D), Canberra ACT 2600, Australia*

ABSTRACT

We present QUOKKA, a new subcycling-in-time, block-structured adaptive mesh refinement (AMR) radiation hydrodynamics code optimised for graphics processing units (GPUs). QUOKKA solves the equations of hydrodynamics with the piecewise parabolic method (PPM) in a method-of-lines formulation, and handles radiative transfer via the variable Eddington tensor (VET) radiation moment equations with a local closure. We use the AMReX library to handle the adaptive mesh management. In order to maximise GPU performance, we combine explicit-in-time evolution of the radiation moment equations with the reduced speed-of-light approximation. We show results for a wide range of test problems for hydrodynamics, radiation, and coupled radiation hydrodynamics. On uniform grids in 3D ~~, we achieve a peak of 93~~ on a single GPU, our code achieves > 250 million hydrodynamic updates per second ~~per GPU, and 22~~, and almost 40 million radiation hydrodynamic updates per second ~~per GPU~~. For radiation hydrodynamics problems on uniform grids in 3D, our code ~~also~~ scales from 4 GPUs to 256 GPUs with an efficiency of 80 per cent. The code is publicly released under an open-source license on ⃝GitHub.

Key words: radiation hydrodynamics – numerical methods

1 INTRODUCTION

1.1 Methods for radiation-hydrodynamics

In many astrophysical systems, the radiation field carries a substantial portion of the total momentum and energy budget, and therefore must be included in any numerical simulation. However, simulating such systems, particularly at high resolution, presents a fundamental challenge in both physics and numerics. Part of this challenge is dimensionality: in full generality the radiation field is governed by the time-dependent equation of radiative transfer,

$$\frac{1}{c}\frac{\partial}{\partial t}I_\nu + \hat{\mathbf{n}} \cdot \nabla I_\nu = \eta_\nu - \kappa_\nu^{(\text{tot})}\rho I_\nu + \frac{1}{4\pi}\int \kappa_\nu^{(\text{sca})}\rho I_\nu \, d\Omega, \quad (1)$$

where I_ν is the radiation intensity at frequency ν travelling in the direction specified by the unit vector $\hat{\mathbf{n}}$, η_ν is the matter emissivity, ρ is the matter density, and $\kappa_\nu^{(\text{tot})}$ and $\kappa_\nu^{(\text{sca})}$ are the total and scatting specific opacities, respectively. This is a time-dependent integro-differential equation with six dimensions: three positions, two angles (parameterised by $\hat{\mathbf{n}}$), and the frequency. Full numerical solution of a problem of this dimensionality, at least if it must be done millions of times to run in tandem with a hydrodynamic (HD) or magnetohydrodynamic (MHD) simulation, remains out of reach for most applications.

Within the astrophysics community, there are two general approaches to numerical radiation hydrodynamics (RHD). One is characteristic methods, which solve equation 1 (or its time-independent form), but only along rays pointing back to particular sources (e.g., Abel & Wandelt 2002; Rijkhorst et al. 2006; Krumholz et al. 2007b) or rays randomly assigned by Monte Carlo (e.g., Fleck & Cummings

1971; Tsang & Milosavljević 2015). A second approach, which we will pursue here, is moment methods (e.g., Mihalas & Mihalas 1984; Castor 2004), whereby one takes moments of the transfer equation, thereby eliminating the angular dimensions of the problem.[1] This results in a series of moment equations that one can solve in lieu of solving the equation of radiative transfer directly, but at the price of introducing the need for a closure relation to approximate the higher moments that appear in the equations being solved. Some authors also combine moment and characteristic methods (e.g., Rosen et al. 2017).[2]

One of the simplest closures is the flux-limited dif-

---

[1] Fully eliminating the angular dependence requires some care, since in general for a moving medium the emissivity and opacity depend on direction, even if the medium itself interacts with light isotropically in its own rest frame. While one might guess velocity-dependent beaming effects are unimportant in non-relativistic problems, it turns out that one cannot formulate a consistent, energy-conserving theory of non-relativistic radiation-hydrodynamics without including them, at least to leading order (Mihalas & Klein 1982; Lowrie et al. 1999; Krumholz et al. 2007a). Systems of moment equations where the radiation moments are written in the lab frame but the emissivity and absorption are written in the comoving frame, where they can be assumed to be isotropic, are known as mixed-frame formulations. This is the most common approach in modern RHD codes.

[2] While characteristic and moment methods are the only ones widely used in astrophysics, in terrestrial applications (for example neutron transport calculations for nuclear reactor design) there are two other widely-used approaches to handle the angular dependence of the transfer equation. One is to discretise the unit sphere using weighted quadratures (the discrete ordinates, or S_n, method; e.g., Lathrop & Carlson 1964; Adams 1997). The other is to expand the angular integration in spherical harmonics (the P_n method; e.g., Modest 1989).

⋆ E-mail: ben.wibking@anu.edu.au (BDW)

fusion (FLD) approximation (Alme & Wilson 1973) (LeBlanc & Wilson 1970; Alme & Wilson 1973), which retains only the first moment equation, for the radiation energy density, and closes the system by assuming both that the time derivative of the radiation flux is zero and that the Eddington tensor, defined as the ratio of the radiation pressure tensor to the radiation energy density, has a fixed value. However, a long-understood deficiency of flux-limited diffusion is that it cannot cast shadows (e.g., Hayes & Norman 2003), a property that can make a critical difference in the dynamics of some semitransparent problems (e.g., Davis et al. 2014). A more accurate approximation is to evolve both the radiation energy density and the radiation flux, while still invoking a closure relation for the Eddington tensor; this is called a two-moment approach, since one solves for the first two moments of the radiation field. When the Eddington tensor is computed via a formal solution of the angle-dependent radiative transfer equation, we obtain the quasidiffusion or variable Eddington tensor (VET) method (Gol'din 1964). When retaining a local closure for the radiation pressure tensor in terms of the radiation energy density E and the flux F, we obtain a local VET method, commonly referred to as the M1 ('moment-one') method (Minerbo 1978; Levermore 1984; Dubroca & Feugeas 1999; González et al. 2007). There are a number of moment-based astrophysical RHD codes, implementing a wide variety of closures, in wide use, including Zeus (Turner & Stone 2001), FLASH (Fryxell et al. 2000), Orion (Krumholz et al. 2007a; Shestakov & Offner 2008; Li et al. 2021), Ramses (Commerçon et al. 2011; Rosdahl et al. 2013), Athena (Davis et al. 2012; Jiang et al. 2012), Enzo (Reynolds et al. 2009; Bryan et al. 2014), Castro (Zhang et al. 2011, 2013; Almgren et al. 2020), and Fornax (Skinner et al. 2019), to give a partial list.

While the use of moment methods removes the dimensionality problem, it leaves a second problem, which is the strong mismatch in signal speeds between radiation and sound (or MHD) waves, which in a non-relativistic system travel at far less than the speed of light. This mismatch renders simple explicit methods, as are commonly used for HD and MHD, impractically slow for radiative transfer, due to the tiny time steps that would be imposed by the Courant-Friedrichs-Lewy (CFL) condition. For this reason, numerical methods for RHD either use an implicit method for the radiation part of the problem (e.g., Zeus, Orion, some versions of Ramses and Athena) or adopt the reduced speed of light approximation (Gnedin & Abel 2001; Skinner & Ostriker 2013; Fornax, other versions of Ramses and Athena). The reduced speed of light approximation (RSLA) consists of replacing the speed of light c that defines the signal speed in the radiation moment equations with a lower speed $\hat{c}$, while keeping the terms that describe the rate of momentum and energy exchange between gas and radiation unchanged. The lower speed $\hat{c}$, while still substantially larger than the HD or MHD signal speeds, is close enough to those speeds to allow radiation time steps large enough to render explicit methods computationally feasible.

1.2 Why a new radiation hydrodynamics code?

In this paper we describe Quokka[3], a new code for RHD. In terms of the taxonomy introduced above, Quokka is a two-moment code that uses the RSLA to allow an explicit treatment of radiation transport. Quokka is also an adaptive mesh refinement (AMR) code, so it

begins with a base grid at uniform spatial resolution, but then dynamically adds higher-resolution grids as needed to achieve user-specified accuracy goals (Berger & Oliger 1984; Berger & Colella 1989). However, these features do not make Quokka unique: Orion and Ramses (among others) offer moment-based AMR RMHD, while Fornax uses RSLA on a dendritic (though not adaptive) grid.

The unique feature of Quokka is that it has been designed from the ground up to run efficiently on graphics processing units (GPUs). This design goal motivated our choice of both algorithms and low-level implementation details. While Quokka is not the first GPU hydrodynamics code in astrophysics (others include Gamer, Schive et al. 2010, 2018, Cholla, Schneider & Robertson 2015, Castro, Almgren et al. 2020, and ARK-RT, Bloch et al. 2021), nor even the first AMR GPU code, it is the first to feature two-moment AMR RHD on GPUs.

Bringing RHD to GPUs creates some unique challenges. Contemporary compute nodes are often limited by data bandwidth, both in terms of moving data between main memory and the CPU or GPU, and in terms of moving data between CPUs or GPUs. For this reason, implicit methods generally have poor scalability, due to the need for global communications during an implicit solve (see, e.g., Appendix E of Skinner et al. 2019). This imbalance between computation and communication is magnified on GPUs. Likewise, robust implicit methods require iterative sparse matrix solvers, which achieve lower peak efficiency on GPUs compared to CPUs due to their heavy use of indirect addressing and highly branching control flow. These considerations motivate our choice of an explicit RSLA method. They also motivate our choice of time integration strategy, which as we detail below has been designed to maximize computation (and therefore minimize the relative amount of communication) on each hydrodynamic timestep. We show that, with this strategy, we are able to achieve update computation rates of >~90 > 250 million zone updates per second per GPU for pure HD, and >~20 nearly 40 million for RHD. We also achieve $\geq 80\%$ parallel efficiency (compared to single-node performance) out to 256 GPUs. This combination of performance and scaling makes Quokka substantially faster than any other public RHD code.

The remainder of this paper is organized as follows. In Section 2 we introduce the set of equations that Quokka solves, and detail our numerical methods for solving them. In Section 3, we present a wide range of tests that demonstrate the accuracy and capabilities of the code. Section 4 covers our tests of code performance and scalability. In Section 5, we discuss the range of applicability of our methods, and our plans for application and future code expansions. Finally, the code itself, including all test problems, is freely available on ⚙ GitHub under an open-source license.

2 METHODS

2.1 Equations

We solve the equations of radiation hydrodynamics (Pomraning 1973; Mihalas & Mihalas 1984; Castor 2004) for an inviscid, nonrelativistic fluid in local thermodynamic equilibrium in the mixed-frame formulation, where the radiation variables are defined in an inertial frame (i.e., Eulerian simulation coordinates) and the radiation-matter interaction terms are written in the frame comoving with the fluid, with the transformations between the frames accounted for via the addition of radiation-matter exchange terms that depend explicitly on the ratio of fluid velocity to the speed of light, $\beta = v/c$. In this first version of Quokka we omit scattering, so that matter-radiation inter-

---

[3] Quadrilateral, Umbra-producing, Orthogonal, Kangaroo-conserving Kode for Astrophysics!

action is purely by emission and absorption. We write the equations as follows:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0 \,, \tag{2}$$

$$\frac{\partial (\rho \mathbf{v})}{\partial t} + \nabla \cdot (\rho \mathbf{v}\mathbf{v} + \mathsf{P}) = \mathbf{G} \,, \tag{3}$$

$$\frac{\partial E}{\partial t} + \nabla \cdot [(E + \mathsf{P})\mathbf{v}] = cG^0 \,, \tag{4}$$

$$\frac{\partial E_r}{\partial t} + \nabla \cdot \mathbf{F}_r = -cG^0 \,, \tag{5}$$

$$\frac{1}{c^2}\frac{\partial \mathbf{F}_r}{\partial t} + \nabla \cdot \mathsf{P}_r = -\mathbf{G} \,, \tag{6}$$

where ρ is the gas density, $\mathbf{v}$ is the gas velocity, E is the total energy density of the gas, $\mathsf{P} = \delta_{ij}P$ is the gas pressure tensor, E_r is the radiation energy density, F_r is the radiation flux, P_r is the radiation pressure tensor, $\nabla \cdot \rho \mathbf{v}\mathbf{v}$ denotes the sum $(\rho v_i v^j)_{,j}$, and G^i is the radiation four-force, with G^0 the time-like component and $\mathbf{G}$ consisting of the space-like components. In the mixed-frame formulation, the radiation four-force to order β is

$$-cG^0 = \rho(\kappa_P 4\pi B - \kappa_E c E_r) + \rho \kappa_F \left(\frac{\mathbf{v}}{c} \cdot \mathbf{F}_r\right) \,, \tag{7}$$

$$-\mathbf{G} = -\rho \kappa_F \frac{\mathbf{F}_r}{c} + \rho \kappa_P \left(\frac{4\pi B}{c}\right)\frac{\mathbf{v}}{c} + \rho \kappa_F \frac{\mathbf{v}\mathsf{P}_r}{c} \,, \tag{8}$$

where κ_F, κ_E, and κ_P are the flux-mean, energy-mean, and Planck-mean specific opacities evaluated in the comoving frame, B is the Planck function evaluated at the gas temperature, and $\mathbf{v}\mathsf{P}_r$ is the tensor contraction $v_j \mathsf{P}_r^{i,j}$ (Mihalas & Mihalas 1984). The latter two terms in the expression for $\mathbf{G}$ correspond to the relativistic work term of Krumholz et al. (2007a) and are only important in the regime $\beta\tau \gtrsim 1$ (where τ is a characteristic optical depth), to which we cannot apply the RSLA (as discussed below), so we neglect them. However, the term of order β in the expression for cG^0 corresponds to the work done by the radiation force on the gas and can be the dominant term for problems of interest.

To apply the RSLA to these equations, we first rewrite the radiation moment equations so that they have a factor of exactly $1/c$ next to each of the time derivatives:

$$\frac{1}{c}\frac{\partial E_r}{\partial t} + \nabla \cdot \left(\frac{\mathbf{F}_r}{c}\right) = -G^0 \,, \tag{9}$$

$$\frac{1}{c}\frac{\partial}{\partial t}\left(\frac{\mathbf{F}_r}{c}\right) + \nabla \cdot \mathsf{P}_r = -\mathbf{G} \,, \tag{10}$$

then we replace this $1/c$ factor with a factor of $1/\hat{c}$, where $\hat{c}$ is the reduced speed of light, and multiply through by factors of $\hat{c}$ to obtain the conservation law form of the reduced speed of light radiation moment equations (e.g., Skinner & Ostriker 2013):

$$\frac{\partial E_r}{\partial t} + \nabla \cdot \left(\frac{\hat{c}}{c}\mathbf{F}_r\right) = -\hat{c}G^0 \,, \tag{11}$$

$$\frac{\partial \mathbf{F_r}}{\partial t} + \nabla \cdot (c\hat{c}\,\mathsf{P}_r) = -c\hat{c}\,\mathbf{G} \,. \tag{12}$$

The maximum wave speed of this system of equations is bounded by $\hat{c}$ (as long as the flux satisfies causality, i.e. $F_r \leq cE_r$). As emphasised by Skinner & Ostriker (2013), all other factors of c remain unchanged, and, since the factors of c are unchanged on the right-hand side of the hydrodynamic equations, the reduced speed of light radiation hydrodynamic system does not conserve total energy or momentum for $\hat{c} \neq c$. When the left-hand side flux divergence terms are negligible, this nonconservation implies that the equilibrium temperature of the reduced speed of light system is slightly modified with

respect to the correct equilibrium temperature, implying that we cannot apply the RSLA to problems in the equilibrium diffusion limit in general (see section 3.2.3).

Writing out the right-hand side terms explicitly, we obtain

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0 \,, \tag{13}$$

$$\frac{\partial (\rho \mathbf{v})}{\partial t} + \nabla \cdot (\rho \mathbf{v}\mathbf{v} + \mathsf{P}) = \rho \kappa_F \mathbf{F}_r/c \,, \tag{14}$$

$$\frac{\partial E}{\partial t} + \nabla \cdot [(E + \mathsf{P})\mathbf{v}] = -c\rho(\kappa_P a_r T^4 - \kappa_E E_r) - \rho \kappa_F \left(\frac{\mathbf{v}}{c} \cdot \mathbf{F}_r\right) \,, \tag{15}$$

$$\frac{\partial E_r}{\partial t} + \nabla \cdot \left(\frac{\hat{c}}{c}\mathbf{F}_r\right) = \hat{c}\rho\left(\kappa_P a_r T^4 - \kappa_E E_r\right) + \rho \kappa_F \left(\frac{\hat{c}}{c}\frac{\mathbf{v}}{c} \cdot \mathbf{F}_r\right) \,, \tag{16}$$

$$\frac{\partial \mathbf{F}_r}{\partial t} + \nabla \cdot (c\hat{c}\,\mathsf{P}_r) = -\hat{c}\rho\kappa_F \mathbf{F}_r \,. \tag{17}$$

These equations make no approximations about the frequency dependence of the radiation field. However, for computational tractability, in what follows we will approximate κ_F with the Rosseland mean opacity κ_R, which yields the correct radiation force in the diffusion limit, and approximate κ_E with the Planck mean opacity κ_P, which yields the correct energy absorption and emission in the optically-thin limit for fluids at rest (Mihalas & Mihalas 1984). However, we emphasise that the choice to set $\kappa_F \approx \kappa_R$ and $\kappa_E \approx \kappa_P$ is an additional approximation, and that others might be preferable depending on the physical system being simulated. In future work, we plan to address the limitations of these approximate grey opacities via an extension of our method to the multigroup solution of the radiation moment equations. Our present set of equations is sufficient for grey nonrelativistic radiation hydrodynamics in the semi-transparent regime, where we can neglect the 'relativistic work term' that is important only in the dynamic diffusion ($\beta\tau \gtrsim 1$) regime, as described earlier.

2.2 Solution method

We solve the system formed by equation 13–equation 17 using an operator split approach, whereby we first advance the hydrodynamic transport subsystem (Section 2.2.1), then the radiation transport subsystem (Section 2.2.2), and finally update the local coupling terms (Section 2.2.3). The first subsystem uses a single explicit update step, the second a set of subcycled explicit updates, and the third a purely local implicit update. We describe each of these steps below.

This update cycle operates within a Berger & Oliger (1984) / Berger & Colella (1989) adaptive mesh refinement (AMR) framework, whereby each spatial variable is represented by a volume average in each cell, on a rectangular, Cartesian grid. We cover the entire computational domain with a coarse grid with cell spacings $\Delta x_0, \Delta y_0, \Delta z_0$ in the x, y, and z directions; the grid spacings need not be the same, but for most applications we choose them to be the same. We denote this coarse grid level 0. We then dynamically add (or remove) additional, finer grids over parts of the domain in response to user-specified refinement criteria. We denote these additional levels 1, 2, . . ., with each grid on level l having cells a factor of 2 smaller than those on level $l-1$, so that the cell spacing on level l is $\Delta x_0/2^l$ in the x direction, and similarly for y and z. We use only factor of 2 refinements in order to minimize numerical glitches arising from the discontinuous change in resolution, which can arise especially in problems where shocks cross the coarse-fine mesh interface at an oblique angle (e.g., Fryxell et al. 2000). When adding

finer grids we conservatively interpolate the underlying coarse data (using linear interpolation for robustness, regardless of the spatial reconstruction used to compute the fluxes), and when removing finer grids we conservatively average down the fine data. Time steps on different AMR levels are sub-cycled, such that the time step on level l is $\Delta t_l = \Delta t_0/2^l$. At the end of every two time steps on level $l > 0$, we perform a synchronization step to ensure that we maintain machine-precision conservation for all conserved quantities (Section 2.2.4).

Our implementation of AMR in Quokka ~~is based on~~ uses the lower-level `AMRCore` interface provided by the AMReX library (Zhang et al. 2019; AMReX Development Team et al. 2021) for adaptive mesh grid generation and coarse/fine grid interpolation, domain decomposition, and parallel communication. In addition to solving the radiation hydrodynamics equations, Quokka itself handles the timestepping and mesh refinement criteria.

2.2.1 Hydrodynamics

For the solution of the hydrodynamic subsystem (equation 13–equation 15, omitting the matter-radiation coupling terms on the right hand sides), we adopt a method-of-lines (or semi-discrete) approach, discretizing the spatial variables while initially keeping the time variable continuous, thereby transforming the partial differential equations into a large set of ordinary differential equations that can be integrated in time using a standard ordinary differential equation (ODE) integrator ~~(Jameson et al. 1981)~~(Hyman 1979; Jameson et al. 1981). For the latter, we use the second-order strong stability preserving Runge-Kutta method (RK2-SSP; Shu & Osher 1988). Such an approach has been successfully employed in several recent astrophysical hydrodynamics codes (Skinner et al. 2019; Stone et al. 2020). ~~Somewhat surprisingly, we~~

We schematically write the timestep Δt used for the RK2-SSP integration as

$$\Delta t = C_0 \frac{\Delta x}{|\lambda|}, \tag{18}$$

where Δx is the minimum grid spacing, $|\lambda|$ is a maximum signal speed, and C_0 is a stability coefficient. Analysis of the stability polynomial of a Runge-Kutta integrator applied to the linearized hydrodynamics equations (Colella et al. 2011; McCorquodale & Colella 2011) yields a value for $|\lambda|$ of

$$|\lambda| = \max \sum_{d=1}^{D} \left[(\mathbf{v} \cdot \hat{\mathbf{e}}_{\mathbf{d}}) + c_s \right], \tag{19}$$

where $\mathbf{v}$ is the fluid velocity, $\hat{\mathbf{e}}$ is the unit vector in coordinate direction d, c_s is the adiabatic sound speed, and the maximum is taken over all cells. However, even in the linear case, such an eigenvalue analysis gives, in general, only a necessary condition for stability and not a sufficient condition (Reddy & Trefethen 1992). For this reason, we more conservatively estimate the value of $|\lambda|$ as

$$|\lambda| = D \max(|\mathbf{v}| + c_s), \tag{20}$$

and compute the timestep Δt on each AMR level as

$$\Delta t = \frac{C_0}{D} \frac{\Delta x}{\max(|\mathbf{v}| + c_s)}, \tag{21}$$

where we define the dimensionless factor C_0/D to be the CFL number so as to be consistent with its standard definition in one spatial dimension.[4] When written in this form, for both forward Euler and RK2-SSP, the maximum stable coefficient C_0 for a system of constant-coefficient, linear equations is $C_0 = 1$. The maximum stable CFL number in 3D for the RK2-SSP integrator is therefore $1/3$. We note that it is not sufficient to estimate $|\lambda|$ as $\max(|\mathbf{v}| + c_s)$, since the component-wise sum of the velocities may exceed the vector magnitude $|\mathbf{v}|$ and therefore violate the lower bound given by equation 19.[5]

We find that such a method-of-lines scheme is not stable when combining higher-order spatial reconstruction with forward Euler time integration~~, but~~. However, we find it is stable for timesteps ~~$O(\Delta x/(c_s + |v|))$~~ satisfying the above timestep criterion when used with higher-order (second-order or higher) Runge-Kutta methods. ~~This is consistent with the findings of Stone et al. (2020), who also find that the maximum stable timestep corresponding to the linear stability region of a given (second order or higher) Runge-Kutta integrator divided by the number of spatial dimensions.~~ We note that this stability problem with forward Euler is also found by Stone et al. (2020) in the method-of-lines implementation of Athena++, but does not appear for single-step integrators that average in time the reconstructed profiles of characteristic waves over the cell interfaces, as done in the original version of PPM (Colella & Woodward 1984).

As Skinner et al. (2019) notes, in contrast to fully-discrete unsplit hydrodynamic methods such as the corner transport upwind (CTU) method (Colella 1990), the coupling across corners of each cell is achieved via the use of a multi-stage time integrator, rather than via direct computation of fluxes from diagonal neighbors of each cell. While we are formally limited to a smaller timestep compared to the CTU method (due to the factor of $1/D$), our method may be more robust in practice, as the CTU integrator has been found to be unstable in supersonic turbulence with strong radiative cooling unless very small ($\lesssim 0.1$) CFL numbers are employed (Schneider & Robertson 2017).

We reconstruct the hydrodynamic variables on each face of each cell from the cell-average variables of the neighbouring cells. We perform this reconstruction using the piecewise parabolic method (PPM; Colella & Woodward 1984, hereafter CW84) using the primitive hydrodynamic variables (density, velocity, and pressure). As is standard, the conversion from conserved (density, momentum, and energy) to primitive variables is carried out assuming that the volume average and cell centered states are equivalent, which is an approximation accurate to $O(\Delta x^2)$. As noted by several authors, the PPM algorithm is therefore formally second-order accurate in spatial resolution.[6] After the primitive variables have been defined, for the reconstruction step proper, we use the standard interface-centered PPM stencil:

$$q_{j+1/2} = \frac{7}{12}(q_j + q_{j+1}) - \frac{1}{12}(q_{j+2} + q_{j-1}). \tag{22}$$

We follow the implementation of Stone et al. (2020) in re-grouping the above terms symmetrically with respect to the interface $i + 1/2$ so

---

[4] This is the same timestep criterion used in the Fornax code, with $D = 3$ and $\Delta x = \Delta r/\sqrt{2}$; see Eq. 37 of Skinner et al. 2019.

[5] As an example, consider the velocity vector with unit magnitude $|\mathbf{v}|$ and equal components $v_x = v_y = v_z$. Then each component $v_i = \sqrt{3}/3$ and the sum of components $\sum_{d=1}^{3} \mathbf{v} \cdot \hat{\mathbf{e}}_{\mathbf{d}} = \sqrt{3} \approx 1.732$.

[6] We note that there exist fully fourth-order versions of PPM (Felker & Stone 2018), but because fourth-order accuracy does not permit local source terms to be evaluated independently for each cell, we choose to implement a second-order method.

as to preserve exact symmetry in floating point arithmetic. ~~On a mesh with uniform spacing between cells along the direction perpendicular to the reconstructed face, this stencil is fourth-order accurate.~~

We do not perform the slope-limiting and contact steepening steps of CW84. We instead ~~enforce monotonicity of the reconstructed state by re-setting the interface state to lie between the values of the cells adjacent to the interface, following~~ prevent new extrema in the reconstructed states by limiting the interface states at the faces of a given cell to the minimum and maximum of the cell-average values of cell under consideration and its two neighbouring cells along the axis of reconstruction, similar to the monotonicity constraint introduced by Mignone et al. (2005). This is followed by the extrema detection and overshoot correction step within each cell as described by CW84. In this step, the parabola assumed to exist across each cell is examined. If an 'overshoot' (as defined by CW84) of the parabola is detected, we follow the original CW84 prescription of performing linear reconstruction on the side of the cell affected by the overshoot. If an extremum is instead detected, rather than forcing the reconstruction to a constant value across the cell as done by CW84, we revert to performing a linear reconstruction within the affected cell, following Balsara (2017). ~~Either of these~~ We note that these latter two steps only examine the interface values, and do not guarantee that the interface states lie within neighbouring cell-average values, and therefore the cell-average limiting carried out in the first step is not redundant. Any of these limiting steps may make the interface states discontinuous, with distinct states associated with each of the two cells adjacent to an interface~~, which reduces the order of accuracy of the stencil. Since the overall algorithm is a combination of second order and (usually) fourth order steps, the overall order of accuracy of PPM is often referred to as 'third order' (e.g., Stone et al. 2020). However, we note that there exist fully fourth-order extensions of PPM (Felker & Stone 2018), but these methods do not allow local source terms to be evaluated independently for each cell while maintaining fourth-order accuracy~~.

We also implement reconstruction based on a piecewise-linear method (PLM) using the monotonized-central (MC) slope limiter (Van Leer 1977). We use PPM reconstruction by default, but allow PLM reconstruction via a compile-time option.

In some cases, especially in underresolved strong shocks, the previous steps do not provide sufficient dissipation to avoid oscillations. This problem was recognized by CW84, who proposed a shock flattening procedure in combination with a small amount of artificial viscosity. We find that this shock flattening procedure is not sufficient in multidimensional problems. Instead, we follow Miller & Colella (2002), who generalize the CW84 shock-flattening procedure for multidimensional hydrodynamics. Using this latter method, we find that no artificial viscosity is needed and we do not include any in our implementation.

Finally, in order to compute the flux of mass, momentum and energy between cells, we use the HLLC Riemann solver with the 'primitive variable Riemann solver' wavespeeds and intermediate states (Toro 2013). We make the standard approximation that the face-average flux is the same as the face-centered flux, and therefore this step is also second-order accurate in spatial resolution. For each cell, the fluxes across each face are then added together to produce an unsplit spatial divergence term used by each stage of the Runge-Kutta integrator to advance the cell in time.

In multidimensional simulations, it has been long recognized that in strong grid-aligned shocks, the HLLC Riemann solver can unphysically amplify the so-called 'carbuncle' instability (Quirk 1994). In astrophysical problems, this is most often encountered in strong explosions. Implementing additional dissipation in the form of artificial viscosity (e.g., Gittings et al. 2008), the 'H-correction' (Sanders et al. 1998), or by adaptively switching to an HLL Riemann solver (Harten et al. 1983) for computing fluxes perpendicular to strong shocks (e.g., Quirk 1994; Skinner et al. 2019) are possible solutions to this issue. In future work we plan to implement an adaptive procedure to fix the carbuncle instability via the latter method.

Future work may also include implementing an adaptive method to reduce the order of reconstruction in order to preserve density and pressure positivity in near-vacuum regions, such as the multidimensional optimal order detection (MOOD) method of Clain et al. (2011). An alternative solution may be to adaptively switch to an exact (iterative) Riemann solver depending on the flow conditions (Toro 2013).

2.2.2 Radiation

We solve the radiation transport subsystem (equation 16–equation 17, again omitting the terms on the right-hand side) in a similar method-of-lines fashion. Our approach is most similar to that of Skinner et al. (2019), who also evolve the radiation moment equations with a time-explicit method-of-lines approach; however, they do not use either PPM reconstruction or a reduced speed of light. Because even with the RSLA the signal speed for the radiation subsystem is substantially larger than for the hydrodynamic subsystem, we evolve the former explicitly in time with several radiation timesteps per hydrodynamic timestep. In the regime of applicability of the RSLA, this approach allows a much more computationally efficient solution to the radiation moment equations, due to the fact that explicit methods have a greater arithmetic intensity per byte of data, have simple memory access patterns and control flows (compared to implicit solvers), and do not require global communication across the computational domain in order to advance the solution in time. All these features are greatly beneficial on GPUs, where the ratio of floating-point arithmetic performance to memory bandwidth is typically greater than on CPUs.

We carry out each radiation subcycle using the same RK2-SSP integrator (Shu & Osher 1988) that we use for hydrodynamics. We likewise use a finite volume representation of the radiation variables, with a PPM spatial reconstruction (or optionally, PLM) of the radiation energy density E_r and reduced flux $\mathbf{f} = \mathbf{F}_r/cE_r$; the only difference in our procedures for hydrodynamics and radiation is that for radiation we do not employ a shock flattening procedure. There can exist unphysical radiation shocks when using local closures, since in general such closures make the radiation subsystem nonlinear, but there is no applicable shock flattening procedure to suppress this effect. We carry out reconstruction in terms of the reduced flux $\mathbf{f}$ rather than the absolute flux $\mathbf{F}_r$ in order to suppress unphysical fluxes $|\mathbf{F}_r| > cE$. This is effective in 1D problems, but in multidimensional problems, the magnitude of the radiation flux may still exceed cE_r, which is an unphysical state in which local closures cannot compute the Eddington factor at all. Reducing the order of reconstruction to first order (piecewise constant) when the interface states violate this constraint helps but does not eliminate the issue in all cases. For the purpose of computing the local closure only, we use rescale the flux such that $|\mathbf{F}_r| = cE_r$ whenever $|\mathbf{F}_r| > cE_r$. For particularly difficult problems, especially in order to avoid unphysical instabilities in the propagation of non-grid-aligned optically-thin radiation fronts, we find that it is necessary to reconstruct the radiation variables using PLM reconstruction.

One drawback to upwind finite volume methods for radiation transport is that in naive form, they do not give the correct behavior for diffusive regions where the optical depth per cell is much greater than

unity. This failure occurs because numerical diffusion dominates over physical diffusion when using upwind methods when the mean free path of photons is not resolved (Lowrie & Morel 2001). One common approach to fix this incorrect behavior is to modify the Riemann solver in the optically thick regime to reduce the upwind bias of the spatial derivative (Audit et al. 2002; Skinner et al. 2019; Mezza-cappa et al. 2020). However, this can lead to violations of causality (i.e., $|\mathbf{F}_r| > cE_r$) when the radiation flux is in the streaming regime (Audit et al. 2002), which occurs especially at discontinuities in the opacity between optically-thin cells and optically-thick cells. The only apparent fix for this problem, which we adopt, is to disable the optical-depth correction in the Riemann solver for those cells where it produces a causality-violating state. We find that this condition is only activated when $f \to 1$, so it may not qualitatively affect the solution. However, we also advocate refining on the gradient in the optical depth per cell in order to resolve the boundary layers in such situations whenever it is computationally feasible.

For computing the flux of radiation quantities between cells, we use an HLL Riemann solver, with wavespeeds computed assuming the Eddington factors are fixed at the beginning of the timestep (Balsara 1999). This approach allows us to substitute different closure relations for the Eddington factors without requiring a modification of the Riemann solver, unlike previous implementations that are restricted to a single local closure (e.g., González et al. 2007; Skinner & Ostriker 2013; Skinner et al. 2019). In principle, we could even use Eddington tensors computed via a short characteristics formal solution of the radiative transfer equation (e.g., Davis et al. 2012), but we leave exploration of a non-local VET method to future work.

Our default closure relation for the Eddington tensor is the Lever-more (1984) closure, which is derived by assuming that the radiation field is isotropic in some (unknown) reference frame and then computing a Lorentz transform from this reference frame to one in which the reduced flux $\mathbf{f}$ matches the value in the cell under consideration. This procedure leads to a radiation pressure tensor (e.g., González et al. 2007; Rosdahl et al. 2013; Skinner & Ostriker 2013)

$$\mathsf{P}_r = \left(\frac{1 - \chi}{2} \mathsf{I} + \frac{3\chi - 1}{2} \hat{\mathbf{n}}\hat{\mathbf{n}} \right) E_r \tag{23}$$

where I is the identity tensor, and the Eddington factor χ and the flux direction cosine $\hat{\mathbf{n}}$ are

$$\chi = \frac{3 + 4f^2}{5 + 2\sqrt{4 - 3f^2}}, \tag{24}$$

$$\hat{\mathbf{n}} = \frac{\mathbf{F}_r}{|\mathbf{F}_r|}. \tag{25}$$

When the radiation flux is exactly zero, we drop the direction-dependent term in equation 23. By considering a coordinate system where the radiation flux is aligned with a coordinate axis, we see that χ is the component of the Eddington tensor in the direction of the radiation flux.

We emphasise that this is only one possible choice of closure, and a variety of alternative local closures exist (e.g., Minerbo 1978; Levermore & Pomraning 1981). We refer readers to Janka (1992) and Koerner & Janka (1992) for systematic comparisons to angle-dependent transport solutions for neutrinos, and Olson et al. (2000) for comparisons to photon solutions. Because of its prominence in the neutrino transport literature, as well as marginally favorable performance on some test problems, we also provide an implementation of the Minerbo (1978) closure in addition to the default Levermore (1984) option. However, users can also implement any local closure of their choice simply by providing an implementation of a function that maps from the reduced flux $\mathbf{f}$ to the Eddington factor χ

for their preferred closure. Doing so does not come at any cost in computational performance.

2.2.3 Matter-radiation coupling

Following the computation of the hyperbolic part of the radiation sub-system, we use an implicit method to evaluate the source terms (those appearing on the right-hand sides of equation 13–equation 17) for both the radiation and hydrodynamic subsystems; this update occurs once per radiation subcycle, and thus several times per hydrodynamic step. Since there are no spatial derivatives in these terms, each cell can be updated independently.

The radiation-matter coupling update occurs in three steps. The first is to handle the energy source terms $c\rho(\kappa_P a_r T^4 - \kappa_E E_r)$ that appear in equation 15 and equation 16. In the regime of problems to which we can apply the RSLA this term is often the stiffest, and we therefore update it using the backward-Euler implicit method of Howell & Greenough (2003), specialized to the case of a single material and extended to include a reduced speed of light. Let $E_g = E - \rho v^2/2$ be the gas internal energy, and let $E_g^{(t)}$ and $E_r^{(t)}$ be the gas internal energy and radiation energy at the end of the hyperbolic update, where the superscript (t) indicates quantities evaluated at this point in the update cycle. We compute the new gas internal energy $E_g^{(t+1)}$ and radiation energy $E_r^{(t+1)}$, where $(t + 1)$ indicates the state after accounting for the exchange term, by solving the implicit system

$$0 = F_G \equiv \left(E_g^{(t+1)} - E_g^{(t)} \right) + \left(\frac{c}{\hat{c}} \right) R^{(t+1)} \tag{26}$$

$$0 = F_R \equiv \left(E_r^{(t+1)} - E_r^{(t)} \right) - (R + S)^{(t+1)}, \tag{27}$$

where

$$R \equiv \Delta t \rho \kappa_P (4\pi B - \hat{c} E_r), \tag{28}$$

Δt is the radiation substep timestep, and S is an optional source term that we include to allow, for example, addition of radiation by stellar sources. The quantities F_G and F_R are the residual errors in the gas energy and radiation energy, respectively.

To solve this system via Newton-Raphson iteration, we require the Jacobian matrix, the elements of which are

$$\frac{\partial F_G}{\partial E_g} = 1 + \left(\frac{c}{\hat{c}} \right) \frac{\partial R}{\partial E_g}, \tag{29}$$

$$\frac{\partial F_G}{\partial E_r} = -c\Delta t \rho \kappa_P, \tag{30}$$

$$\frac{\partial F_R}{\partial E_g} = -\frac{\partial R}{\partial E_g}, \tag{31}$$

$$\frac{\partial F_R}{\partial E_r} = 1 + \hat{c}\Delta t \rho \kappa_P, \tag{32}$$

where

$$\frac{\partial R}{\partial E_g} = \frac{\rho \Delta t}{C_v} \left[\kappa_P \frac{\partial B}{\partial T} + \frac{\partial \kappa_P}{\partial T} \left(4\pi B - \hat{c} E_r^{t+1} \right) \right], \tag{33}$$

and C_v is the gas total heat capacity at constant volume. From the Jacobian, we can write the change in radiation and gas temperature for each iterative update as

$$\Delta E_r = -\frac{F_R + \eta F_G}{\frac{\partial F_R}{\partial E_r} + \eta \frac{\partial F_G}{\partial E_r}}, \tag{34}$$

$$\Delta E_g = -\frac{F_G + \Delta E_r \frac{\partial F_G}{\partial E_r}}{\frac{\partial F_G}{\partial E_g}}. \tag{35}$$

where $\eta \equiv -(\partial F_R/\partial E_g)(\partial F_G/\partial E_g)^{-1}$. We repeatedly apply equation 34 and equation 35 to the radiation and gas energies until the system converges. Howell & Greenough (2003) leave unspecified the convergence criteria they use for their solver. After experimenting with several possibilities, we decide to stop the Newton-Raphson iterations when the residuals F_R and F_G satisfy

$$\left| \frac{F_G}{E_{\text{tot}}} \right| < \epsilon \text{ and} \tag{36}$$

$$\left| \frac{c}{\hat{c}} \frac{F_R}{E_{\text{tot}}} \right| < \epsilon , \tag{37}$$

where

$$E_{\text{tot}} \equiv E_g^{(t)} + \frac{c}{\hat{c}} \left(E_r^{(t)} + S \right) . \tag{38}$$

When $\hat{c} = c$, E_{tot} is the total (internal gas plus radiation) energy at the end of the timestep. By default, the relative tolerance ϵ is set to 10^{-10}. We find that larger tolerances produce unacceptably inaccurate solutions for many problems. In especially stiff problems, it may be necessary to reduce the tolerance to the order of machine precision for double-precision floating point arithmetic ($\sim 10^{-15}$). If the solver exceeds a specified maximum number of iterations (400 by default) without converging, the code prints an error message and exits. Convergence failure usually occurs only when the intial timestep has not been sufficently reduced compared to the CFL timestep at the start of a simulation.

Once the Newton-Raphson iterations have converged, we have obtained the updated gas internal energy and radiation energy, and we proceed to the next two steps of updating the coupling terms. We first update the radiation and gas momenta, accounting for the coupling term $\rho \kappa_F \mathbf{F}_r/c$. To do so, we compute the flux mean opacity κ_F using the updated gas temperature. Then, following Skinner et al. (2019), we use a backward-Euler discretization of the radiation flux source term (modified to include a reduced speed of light):

$$\mathbf{F}_r^{(t+1)} = \frac{\mathbf{F}_r^{(t)}}{1 + \rho \kappa_F \hat{c} \Delta t} . \tag{39}$$

In order to ensure momentum conservation when $\hat{c} = c$, we apply the difference in radiation flux in an equal and opposite manner to the gas momenta (as advocated by Skinner et al. 2019):

$$\Delta \mathbf{F}_r \equiv \mathbf{F}_r^{(t+1)} - \mathbf{F}_r^{(t)} , \tag{40}$$

$$(\rho \mathbf{v})^{(t+1)} = (\rho \mathbf{v})^{(t)} - \frac{\Delta \mathbf{F}_r}{\hat{c} c} . \tag{41}$$

The final step is to compute the work done by the radiation force on the gas. Since we are evolving the conserved variables, this term cannot be computed explicitly as written in equation 16 without causing a significant error in the gas internal energy when the radiation force is stiff. We instead compute this term as the difference in gas kinetic energy over the timestep ΔE_{kin}, then add this quantity to the total gas energy and subtract this quantity from the radiation energy:

$$E^{(t+1)} \leftarrow E_g^{(t+1)} + \left(E_{\text{kin}}^{(t)} + \Delta E_{\text{kin}} \right) , \tag{42}$$

$$E_r^{(t+1)} \leftarrow E_r^{(t+1)} - \left(\frac{\hat{c}}{c} \right) \Delta E_{\text{kin}} , \tag{43}$$

where $E^{(t+1)}$ denotes the total gas energy at the end of the timestep. This completes the update for all radiation-matter coupling terms.

2.2.4 Level synchronization procedure

As explained by Berger & Colella (1989), in an AMR calculation it is necessary to adjust the solution on the coarse AMR level following the solution on any refined level in order to maintain conservation of the evolved quantities (e.g., mass, momentum, energy). For hyperbolic equations evolved explicitly in time, this is traditionally done by saving the flux at the coarse-fine grid boundary in a 'flux register' for both the flux computed on the fine level and the flux computed on the coarse level. In general, these fluxes are different due to the differing stencil used on the coarse and fine levels, and without correction, this would lead to a loss of conservation of energy (and any other conserved quantities). The flux register stores this mismatch, and in the synchronization step, adds the missing mass, momentum, or energy to the cells on the coarse level immediately adjacent to the coarse-fine boundary.

As noted by Howell & Greenough (2003), an implicit radiation update has additional difficulties in ensuring conservation, since radiation can propagate much further than a single grid cell on the coarse grid. Our radiation update is fully explicit, but we would like to advance each AMR level on the hydrodynamic timescale, rather than on the radiation timescale, so we have a similar long-range signal propagation difficulty. Rosdahl et al. (2013) outline three possible solutions to the problem: i) perform the radiation solve after each coarse hydrodynamic step, keeping subcycling-in-time on refined levels (which would be very inaccurate), ii) use a single global timestep for all AMR levels, which allows one to advance the radiation solution on all levels in each radiation substep (which would be very computationally expensive, since in our applications of interest, the global timestep is typically limited by the timestep of the highest-resolution level) or iii) restrict the timestep for each level to the minimum of the radiation and hydrodynamic timesteps. In our code, we set the coarse timestep such that the number of radiation substeps per level is limited to a maximum value $N_{\text{sub,max}}$ in order to minimize the signal propagation distance from the coarse-fine boundaries. The flux mismatch at the coarse-fine boundaries is added to the immediately adjacent cells on the coarse grid at the end of each level advance. When $N_{\text{sub,max}} = 1$, our solution is identical to the flux synchronization method used in the Ramses AMR code (Rosdahl et al. 2013). However, as a default we set the parameter $N_{\text{sub,max}}$ to 10, which appears to be sufficient to avoid significant discontinuities in the radiation energy and flux at coarse-fine boundaries, but still allows for significant subcycling and thus a substantially lower computational cost. We use this value for all test problems shown in this work, but users are able to set this parameter as desired for either greater efficiency or greater consistency at refinement boundaries. When this parameter is too large, however, it is possible for the coarse level to fail to maintain positivity of the radiation energy or causality of the radiation flux.

3 TEST PROBLEMS

We now proceed to describe a series of tests that we have conducted to verify Quokka's accuracy and convergence characteristics, starting with tests of the hydrodynamic subsystem (Section 3.1), followed by tests of the radiation transport and radiation-matter exchange subsystems (Section 3.2), and concluding with tests of coupled radiation hydrodynamics (Section 3.3).

Additional example problems and an automated test suite of 20 test problems with checks against exact solutions are included with Quokka's source code. We run this test suite for each commit and pull request in our GitHub repository. While continuous integration tests such as ours cannot guarantee bug-free software, this practice has flagged and prevented the introduction of several bugs during the development of Quokka. In order to maintain high software quality,

we also run the commercial static code analyzer SonarQube[7] on every commit and pull request.

3.1 Hydrodynamics

For all our hydrodynamics tests, we disable the radiation portion of the code. These tests evaluate the hydrodynamic transport solver in isolation.

3.1.1 Sound wave

We compute the propagation of a sound wave in one dimension in order to measure the convergence of our numerical method to the exact solution as a function of spatial resolution, following the test described by Stone et al. (2008). With $\mathbf{U}$ denoting the vector of conserved variables $(\rho, \rho v_x, \rho v_y, \rho v_z, E)$, we initialize the simulations with the initial state $\mathbf{U}_0 + \delta\mathbf{U}$, where

$$\delta\mathbf{U} = A\mathbf{R}\sin(2\pi x) \tag{44}$$

where $\mathbf{R} = (1, -1, 1, 1, 1.5)$ is the right eigenvector of the linearized hydrodynamic system, and $\mathbf{U}_0$ is the background state with density $\rho = 1$, velocity $\mathbf{v} = 0$, and pressure $P = 1/\gamma$. We set the adiabatic index $\gamma = 5/3$ and the wave amplitude $A = 10^{-6}$. We simulate a periodic domain $x = 0$ to 1 and evolve the system for one wave period, allowing us to compute the error of the solution by direct comparison of the initial conditions and the final state of the simulation. We define the error vector

$$(\Delta\mathbf{U})_k = \frac{1}{N_x}\sum_i \left| U_{i,k} - U_{i,k}^0 \right| \tag{45}$$

where k denotes a component of each state vector, $\mathbf{U_i}$ is the vector of conserved variables in cell i at the final timestep, and $\mathbf{U_i}^0$ is the vector of conserved variables in cell i in the initial conditions. Each component $|\Delta U|_k$ is therefore the L_1 norm of the error of a component of the solution state. We assess the accuracy of the solution based on the root-mean-square (rms) of the components of this error vector, denoted $||\Delta\mathbf{U}||$.

We run simulations using PPM reconstruction and a CFL number of 0.1, using grid sizes from $N_x = 16$ to $N_x = 1024$. We show the error norm as a function of resolution in Figure 1. For $N_x = 16$, we obtain $||\Delta\mathbf{U}|| = 1.0 \times 10^{-7}$, for $N_x = 128$, we obtain $||\Delta\mathbf{U}|| = 1.6 \times 10^{-9}$, and for $N_x = 1024$, we obtain $||\Delta\mathbf{U}|| = 1.7 \times 10^{-11}$. The results for our code are in excellent agreement with those from the ATHENA hydrodynamic solver (Figure 7 of Stone et al. 2008). The N_x^{-2} scaling of the error norm indicates that our hydrodynamic solver converges at second order in spatial resolution, as expected from the formal order of accuracy of the method.

3.1.2 Contact discontinuity

The HLLC Riemann solver has the property that it can resolve an isolated stationary contact discontinuity with infinite resolution (Toro 2013). The HLL solver, on the other hand, introduces a large amount of numerical diffusion for this problem (see Figure 10.20 of Toro 2013). To verify that our hydrodynamic implementation can maintain a perfect contact discontinuity, we simulate a system where the initial conditions have a left and right state separated with a discontinuity
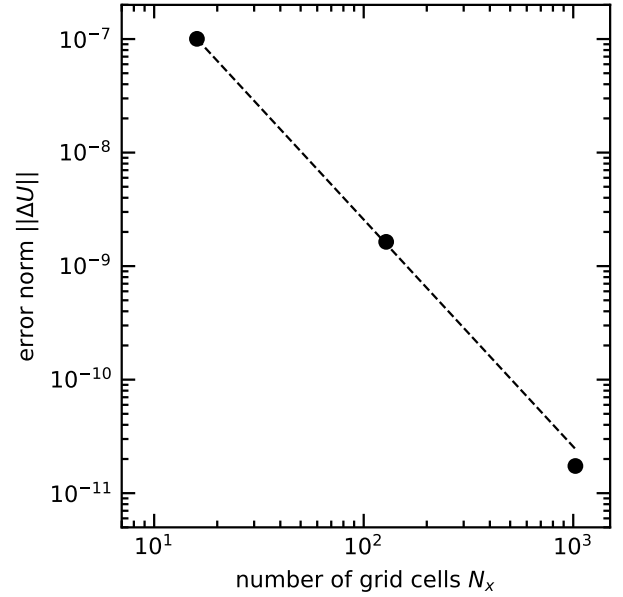
Figure 1. The error $||\Delta\mathbf{U}||$ in the solution (equation 45) for a sound wave as a function of spatial resolution per wavelength N_x. Black circles show numerical results, and the dashed line is a power law that scales as N_x^{-2} normalised to the observed error at the smallest N_x.

at $x = 0.5$. The left state is $\rho_L = 1.4$, $p_L = 1.0$, and the right state is $\rho_R = 1.0$ and $p_R = 1.0$. Since this is a pure contact discontinuity, the solution should not evolve from the initial state. We set the velocity to zero, and use an adiabatic index $\gamma = 1.4$. We evolve the solution numerically until $t = 2$. The error with respect to the correct solution is exactly zero.

3.1.3 Stationary shock tube

Our next test is a stationary shock tube, which we set up using the parameters suggested on the website of F.X. Timmes[8]. This shock tube problem is substantially more difficult to solve than the standard Sod (1978) shock tube test due to the larger jump in pressure and density at the discontinuity. We initialize left and right states with a discontinuity at $x_0 = 2$, with the left state $\rho_L = 10$, and $p_L = 100$, and the right state $\rho_R = 1$ and $p_R = 1$. The initial velocity is zero. We evolve the solution using a CFL number of 0.1 until $t = 0.4$ on a grid of 1000 cells on the domain $[0, 5]$. We use a small CFL number since the wave structure at the discontinuity creates waves that propagate faster than the linearized Roe eigenvalues would predict.

We show QUOKKA's results for this test in Figure 2. As for the sound wave test, we compute the L_1 error norm for each of the conserved variables, and then compute the root-mean-square of those error norms. The rms L_1 error norm divided by the rms norm of the exact solution is 1.12×10^{-3}. Inspecting the solution in Figure 2, we see that the agreement between the exact solution and the numerical solution is very good. The only noticeable differences are small oscillations near discontinuities in the derivative of the solution at $x \approx 2.4$ and near the density discontinuity at $x \approx 3.6$. We find that the shock flattening method of Miller & Colella (2002) is essential
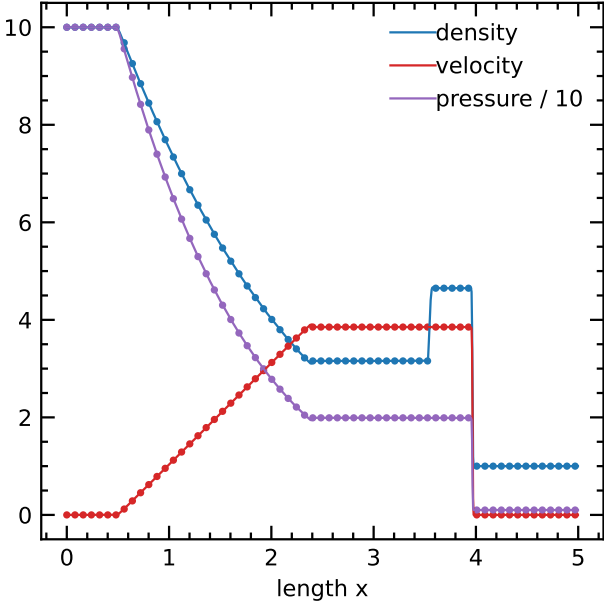
Figure 2. Density, velocity, and pressure profiles for the stationary shock tube test (Section 3.1.3). Lines show the QUOKKA simulation result. For comparison, we show the exact solutions for density, velocity, and pressure as circles of the same color.

to produce a reasonable solution to this problem. Without it, we find unacceptably large post-shock oscillations (not shown).

3.1.4 'LeBlanc' test

We next carry out the 'LeBlanc' shock tube test, originally published by Benson (1992) and further described by Pember & Anderson (2001). In this problem, we initialize a left state with $\rho_L = 1$ and $p_L = \frac{2}{3} \times 10^{-1}$, and a right state with $\rho_R = 10^{-3}$ and $p_R = \frac{2}{3} \times 10^{-10}$. We set the initial velocity to zero and use an adiabatic index $\gamma = 5/3$. This is an extreme shock tube that far exceeds any shock that may be encountered in any conceivable application, featuring a pressure jump of nine orders of magnitude, and is therefore an excellent test problem. We evolve this simulation until $t = 6$ using a grid of 2000 cells and a CFL number of 0.1. The resulting state is shown in Figure 3. Pember & Anderson (2001) highlight the difficulty of obtaining the correct specific internal energy in the solution for this test, but we find that QUOKKA produces the correct shock location and specific internal energy, with the exception of a small overshoot at the shock location. The use of shock flattening is essential for this problem. Overall, the performance of our code on this problem is excellent.

3.1.5 Wave-shock interaction (Shu-Osher) test

We show the Shu-Osher test in Figure 4. Following the description of Shu & Osher (1989), the initial conditions are, on the left side,

$$\rho_L(x) = 3.857143,\tag{46}$$
$$v_L(x) = 2.629369,\tag{47}$$
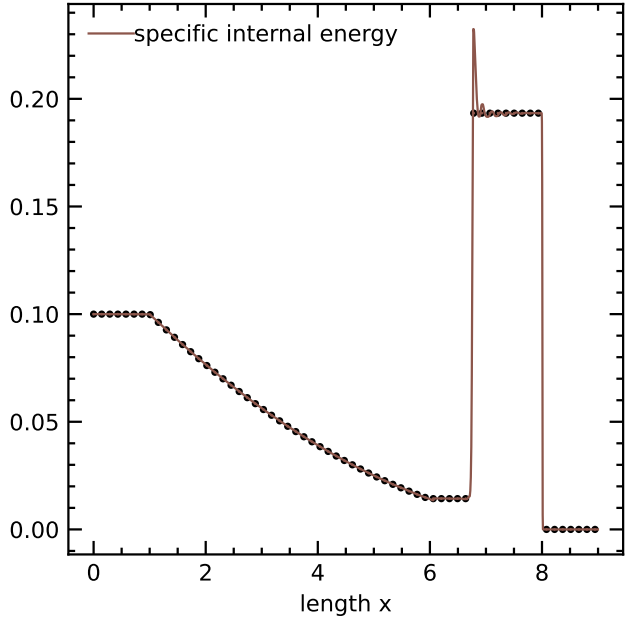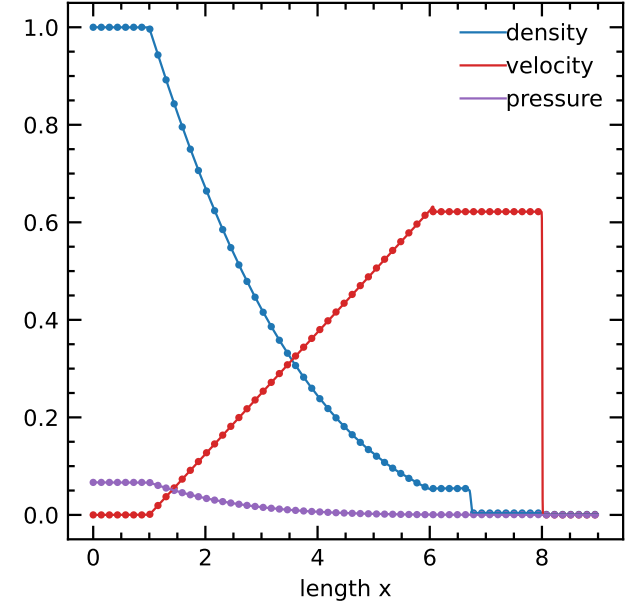$$P_L(x) = 10.33333,\tag{48}$$

Figure 3. *Upper:* Density, velocity, and pressure profiles for the 'LeBlanc' shock tube test problem (Section 3.1.4). *Lower:* The specific internal energy. In both panels, solid lines show the QUOKKA simulation result, and circles in corresponding colours show the exact result.

and the right-hand state is

$$\rho_R(x) = 1 + 0.2 \sin(5x),\tag{49}$$
$$v_R(x) = 0,\tag{50}$$
$$P_R(x) = 1.\tag{51}$$

We compute a reference solution using ATHENA++ (Stone et al. 2020) with the VL2 integrator, PPM reconstruction in the characteristic variables, and the HLLC Riemann solver on a grid of 1600 cells. Our solution is computed using PPM reconstruction (in the primitive variables), the RK2-SSP integrator, and the HLLC Riemann
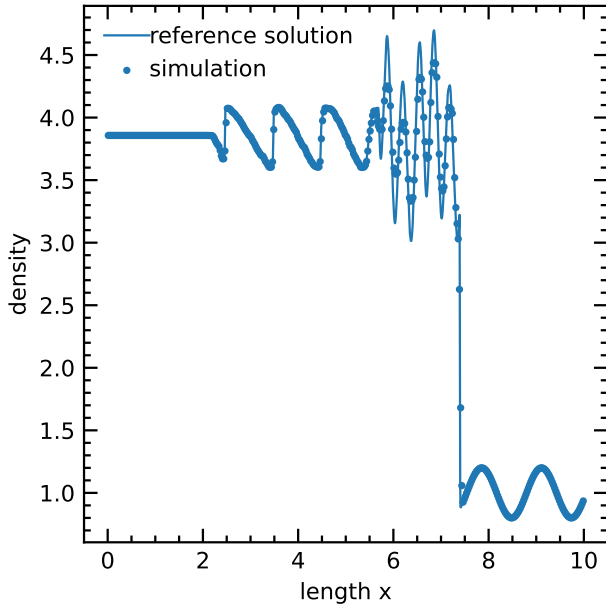
Figure 4. Density as a function of position for the Shu-Osher wave-shock interaction problem (Section 3.1.5). Here we show a comparison between the Quokka solution (shown as circles) and a reference solution (shown as a solid line) computed using Athena++ (Stone et al. 2020).

Figure 5. Density profiles for the slow-moving shock test problem (Section 3.1.6). We show a comparison between the Quokka (solid line) and exact solutions (circles).

solver on a grid of 400 cells. The agreement is very good, with comparable resolution of the high-frequency features to the third-order essentially-non-oscillatory (ENO) scheme of Shu & Osher (1989). When PLM reconstruction is used instead for the same number of grid cells, the high-frequency features are aliased (not shown; see also Figure 14 of Shu & Osher 1989), indicating a significantly higher effective resolution for PPM-based methods even in the presence of shocks.

3.1.6 Slow-moving shock

We show a slow-moving shock in Figure 5 using the parameters from Jin & Liu (1996), where $\rho_L = 3.86$, $(\rho v)_L = -3.1266$, and $E_L = 27.0913$, and the right-side state $\rho_R = 1.0$, $(\rho v)_R = -3.44$, and $E_R = 8.4168$, with $\gamma = 1.4$. This corresponds to the shock jump moving to the right with a velocity $v_{\text{shock}} = 0.1096$. For a CFL number of 0.2, this corresponds to the shock taking ~ 250 timesteps to move across a single cell. This may not be a common scenario for our applications, but it may occur in a protostellar accretion shock, for instance. The quality of the solution is again significantly improved by shock flattening. The post-shock oscillations for slow-moving shocks may still be present with first-order reconstruction (Jin & Liu 1996; Lee 2011), so it is difficult to completely eliminate. We also find that adding a small amount of artificial viscosity does not significantly reduce the oscillations. A modification to PPM reconstruction based on a characteristic wave decomposition succeeds in significantly reducing this oscillation (Lee 2011), which we may consider implementing in a future version of the code.
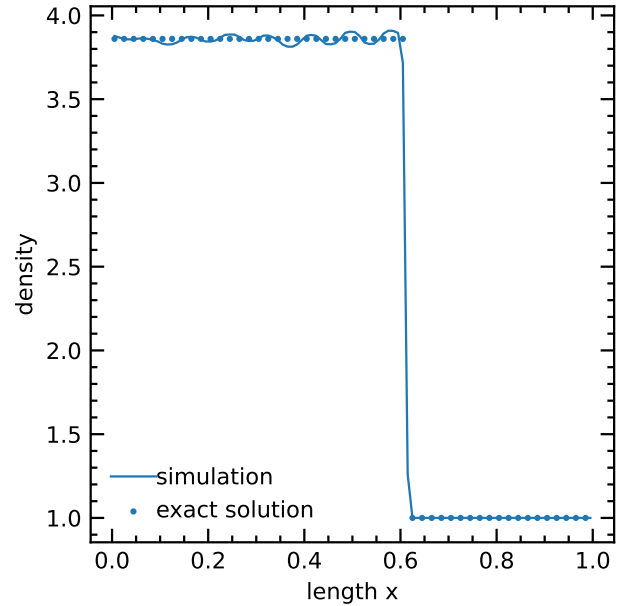
3.1.7 Strong rarefaction

We next test the performance of our code on the 1-2-3 problem of Einfeldt et al. (1991), which features a strong rarefaction and is designed to induce failures in approximate Riemann solvers. The initial conditions consist of left and right states with equal density and energy, $\rho_L = \rho_R = 1$ and $E_L = E_R = 3$, and equal magnitude but oppositely-directed velocities, $(\rho v)_L = -2$, $(\rho v)_R = 2$. We evolve the system to $t = 0.15$, using a CFL number 0.8 and a grid of $N_x = 100$ cells, and show the resulting state in Figure 6. We obtain the exact solution to which we compare the Quokka result using an exact Riemann solver. We find that the solution for the density profile is very close to the exact solution, except for a small discrepancy at the lowest density near $x = 0.5$. However, the most difficult aspect of this problem is obtaining the correct specific internal energy. Our results compare favorably with other solutions obtained with approximate Riemann solvers, where factors of two or three errors are obtained near $x \approx 0.5$ (Toro 2013). Obtaining the correct specific internal energy in the lowest density part of the flow may require the adaptive use of an exact Riemann solver in near-vacuum regions. Nonetheless, our code is stable and well-behaved for this problem.

3.1.8 Kelvin-Helmholtz instability

To show the performance of Quokka in two dimensions, we simulate the Kelvin-Helmholtz instability created by counter-propagating gas flows with a shear layer between them. The purpose of this test is to illustrate the ability of the code to maintain the contact discontinuity between the flows as Kelvin-Helmholz rolls develop, even as we add deep AMR nesting. Following Stone et al. (2020), we use a two-dimensional periodic box on the domain [0, 1] along each axis with
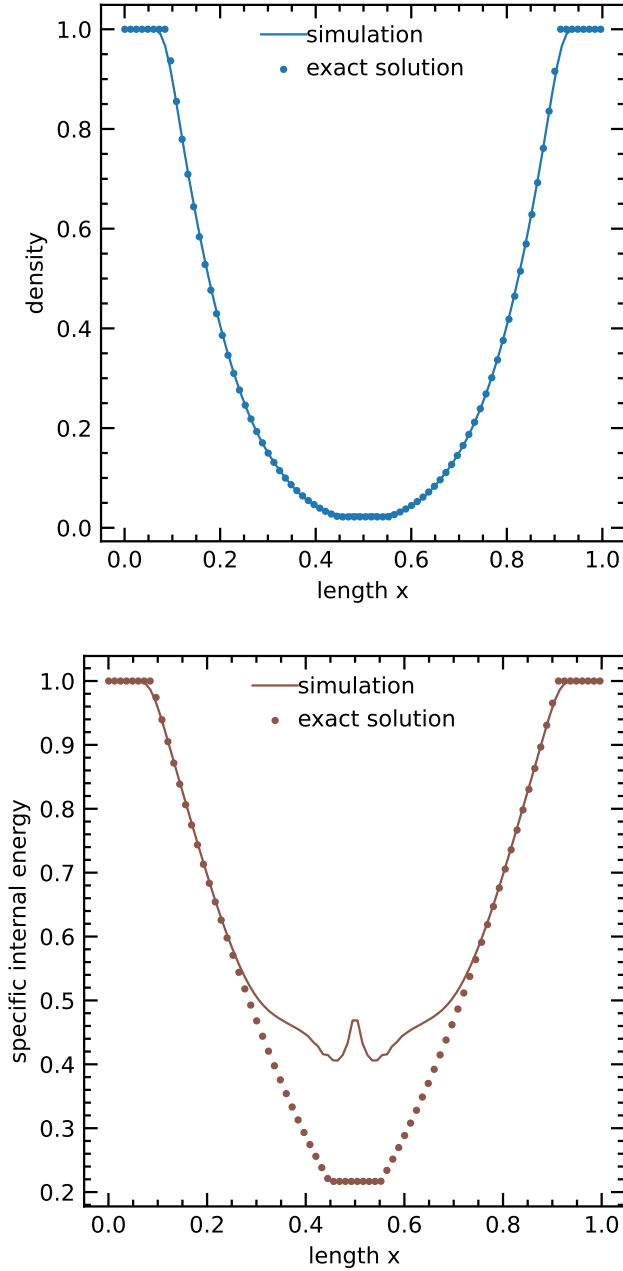
Figure 6. Profiles of density (top) and specific internal energy (bottom) for the strong rarefaction test problem (Section 3.1.7). In both panels we show a comparison between the QUOKKA solution computed using the default HLLC Riemann solver (solid line) and a solution computed using an exact, iterative Riemann solver (circles).

density and velocity given by:

$$\rho = 1.5 - 0.5 \tanh(\tilde{y}/L) \,, \tag{52}$$

$$v_x = 0.5 \tanh(\tilde{y}/L) \,, \tag{53}$$

$$v_y = A \cos(4\pi(x - x_0)) \, \exp(-\tilde{y}^2/\sigma^2) \,, \tag{54}$$

where $x_0 = 0.5$, $y_0 = 0.5$, $\tilde{y} = |y - y_0| - 0.25$, the shearing layer thickness $L = 0.01$, $\sigma = 0.2$, and perturbation amplitude $A = 0.01$. The initial pressure is uniform with $P = 2.5$ and we adopt an adiabatic index $\gamma = 1.4$. We enable AMR, with cells tagged for refinement if the

relative density gradient on either side of the cell in either direction exceeds 0.2, and we allow up to four levels of refinement on top of a base grid size of 2048^2. Thus the peak resolution of the calculation is $32,768^2$. Each local AMR grid has a uniform size of 128^2. We evolve the system to $t = 1.5$ with a CFL number of 0.4, and show the resulting numerical solution in Figure 7. We are able to carry out this calculation on a single GPU in ~ 4.5 hours of wallclock time. While there appears to be no converged solution to this problem without explicit dissipation, we find that our hydrodynamic solver is able to resolve the Kelvin-Helmholz rolls with very little dissipation and with significant small-scale structure caused by secondary instabilities, as expected for inviscid simulations (Lecoanet et al. 2016). There are no visible artifacts at resolution boundaries.

3.1.9 Liska-Wendroff Implosion

We next present our results for the so-called Liska-Wendroff implosion test (Hui et al. 1999; Liska & Wendroff 2003). This problem consists of the square domain $[0, 0.3]^2$, with an inner region $x + y \leq 0.15$ and an exterior region where $x + y > 0.15$ for an ideal gas with adiabatic index $\gamma = 1.4$. The inner region has initial density $\rho = 0.125$ and pressure $P = 0.14$ and the outer region begins with density $\rho = 1$ and pressure $P = 1$. We simulate the subsequent evolution to $t = 2.5$ on a uniform grid of 1024^2 cells with reflecting boundary conditions with a CFL number of 0.4. These initial conditions lead to a shock directed toward the origin, which is then reflected many times by the upper and right walls before finally converging in a jet traveling away from the origin along the diagonal $x = y$, as shown in Figure 8. Liska & Wendroff (2003) note that only codes that discretely preserve symmetry between x- and y-directions successfully produce the jet. In order to recover the jet in QUOKKA, we found it necessary to code the RK2-SSP integrator so that the fluxes in the x- and y-direction are added in an exactly symmetrical manner for each stage of the update. Additionally, when running the problem on NVIDIA GPUs, we preserve this symmetry only if we disable fused multiply-add (FMA) operations via the `nvcc` compiler option `fmad=false`, since the compiler otherwise breaks the symmetry expressed in the source code between the x- and y-direction fluxes. With this compiler option, QUOKKA exactly preserves symmetry along the diagonal and successfully recovers the jet.

3.2 Radiation

For our radiation tests we disable the hydrodynamic part of the code and only use the radiation transport and gas-radiation exchange updates. These tests evaluate the accuracy of those portions of the code.

3.2.1 Marshak wave

We next compute a Marshak wave (Marshak 1958). The problem consists of a uniform gas with a constant density $\rho = 10 \, \mathrm{g \, cm^{-3}}$ and constant opacities $\kappa_P = \kappa_R = 577 \, \mathrm{cm^2 \, g^{-1}}$. The gas has a uniform initial temperature of 10^4 K, but at $t = 0$ we impose on the left-hand side of the domain a boundary condition consisting of a half-isotropic flux with a radiation temperature of 3.481334×10^6 K. The radiation drives a wave of heat into the gas. Following Su & Olson (1996), we set the gas heat capacity at constant volume C_v so a functional form that makes it possible to linearize the matter-radiation coupling terms, and thus obtain a semi-analytic solution:

$$C_v \equiv \frac{\partial E_{\mathrm{int}}}{\partial T} = \alpha T^3 \,, \tag{55}$$
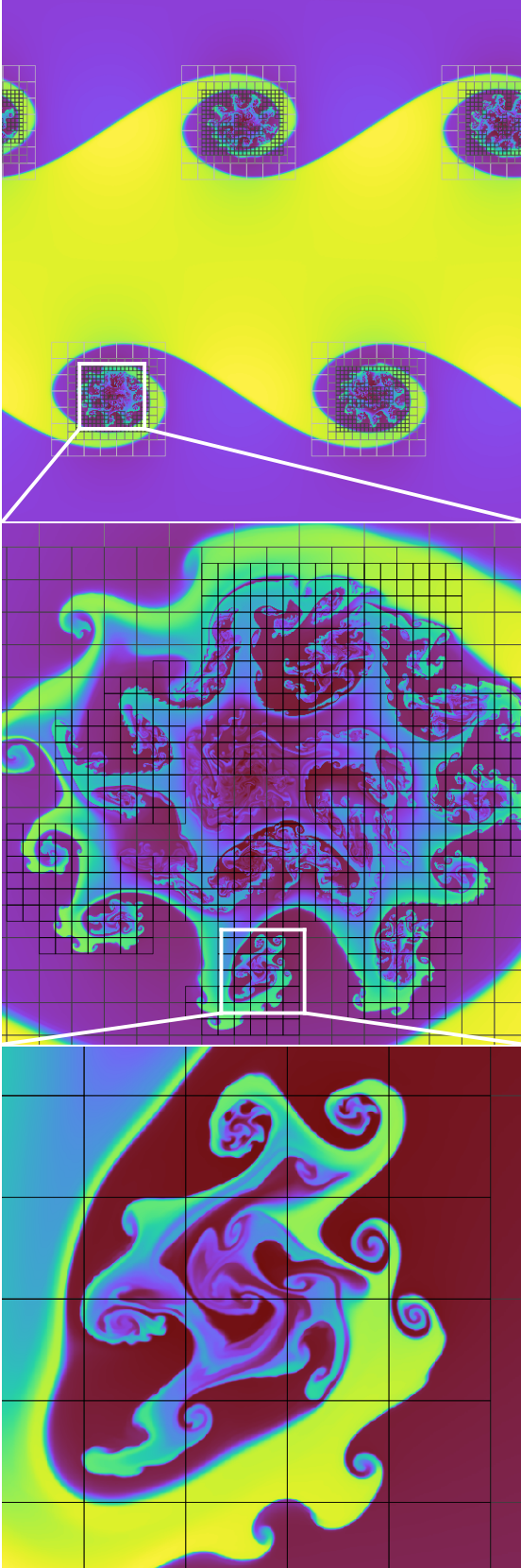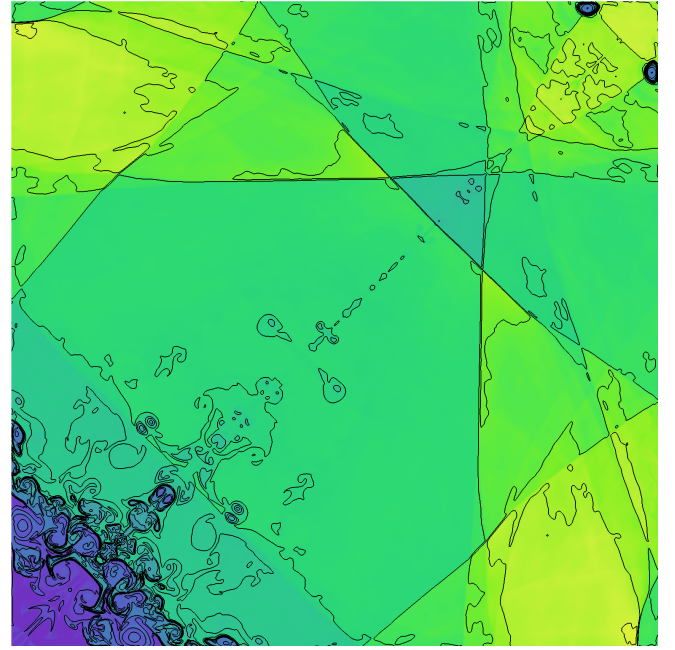
Figure 8. The two-dimensional implosion test (Liska & Wendroff 2003) on a 1024^2 grid at $t = 2.5$. The density is shown with 16 equally-spaced contours between 0.4 and 1.1, with the colormap showing the density over the same range. A thin jet shoots along the *x*-*y* diagonal. The solution is exactly symmetric.

where $E_{\text{int}} = (\alpha/4)\,T^4$, $\alpha = 4a_r/\epsilon$ and $\epsilon = 1$. With this heat capacity, Su & Olson (1996) obtain a semi-analytic quadrature solution of the radiation diffusion equation for this problem as a function of ϵ. We evolve the solution until time $t = \tau/(\epsilon c \rho \kappa)$ where $\tau = 10$, using a simulation domain on the interval $[0\,\text{cm}, 3.466205 \times 10^{-3}\,\text{cm}]$ resolved by grid of $N_x = 400$ cells. We note that this implies an optical depth per cell of $\tau_{\text{cell}} \approx 0.05$, so this problem does not test the accuracy of our code in the asymptotic diffusion limit (where $\tau_{\text{cell}} \gg 1$; the accuracy in this limit is instead tested via the radiation pressure tube problem in Section 3.3.1). We do not use a reduced speed of light for this test.

Since we solve the moment equations, rather than just the diffusion equation, we do not expect our numerical solution to agree with the Su & Olson solution at the leading edge of the wave, where our code respects causality and restricts the propagation speed of the wave to c; this constraint is violated in the diffusion approximation that Su & Olson adopt. However, we can still compare to their solution in the region where $F_r \ll cE_r$ and diffusion is a good approximation. In this region, we obtain excellent agreement with the semi-analytic solution, as shown in Figure 9. Note that the difference between our numerical solution and the "exact" solution at $x \gtrsim 3 \times 10^{-3}$ cm is not an error in our solution. Rather, it is a result of our code properly capturing the finite speed of light, while the semi-analytic solution does not.

3.2.2 Su-Olson problem

We next compute a problem involving radiation penetrating a cold medium but with an internal radiation source rather than a radiation source at the boundary. This problem is defined in dimensionless units where $a_r = c = 1$, with opacities $\kappa_P = \kappa_R = 1$, a constant
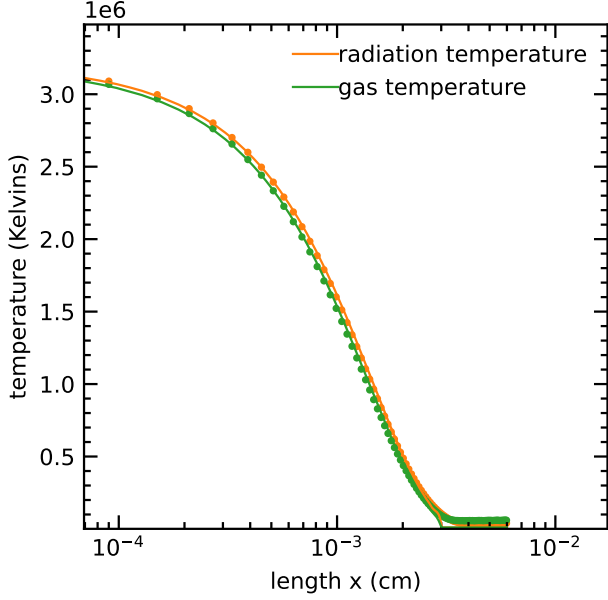
Figure 7. A simulation of the Kelvin-Helmholz instability with 4 levels of refinement. The top panel shows the full simulation domain, and the lower two panels show-successive zoom-ins on parts of the domain. Grid boundaries are shown for levels $l \geq 2$. Color shows density.

Figure 9. A Marshak wave test problem (Section 3.2.1). The radiation temperature and gas temperature computed by QUOKKA are shown as solid lines, while the analytic solution for the diffusion approximation is shown as circles.

Figure 10. The Su-Olson test problem (Section 3.2.2). The numerical solution is the solid line, with the exact diffusion solution shown as circles and the exact transport solution shown as crosses.

density $\rho = 1$, and a radiation source

$$S(x,t) = \begin{cases} Q\, a_r T_H^4 & 0 \leq x < x_0 \text{ and } t < t_0 \, , \\ 0 & x \geq x_0 \text{ or } t \geq t_0 \, , \end{cases} \tag{56}$$

where we have a normalisation factor $Q = (2x_0)^{-1}$, radiation source temperature $T_H = 1$, and spatial extent of the source $x_0 = 0.5$ and temporal extent $t_0 = 10$. The initial radiation and gas energies are zero in the idealized problem, but we set them to 10^{-10} in our simulation since the radiation solver requires nonzero gas and radiation energies. The gas velocity is zero. We adopt reflecting boundary conditions on the domain $[0, 30]$ on a grid of $N_x = 1500$ cells. We do not reduce the speed of light for this test.

When using the heat capacity given by equation 55, a semi-analytic solution of the angle-dependent transport equation may be obtained with a Fourier-Laplace transform (Su & Olson 1997). This solution assumes that $\mathbf{v} = 0$ at all times, so we drop all v/c terms for this problem. We show our numerical solution using CFL number 0.4 at time $t = 10$ in Figure 10; for comparison, we also show the exact transport solution and the exact diffusion solution. We find that with the Levermore (1984) closure, we obtain a solution in between the diffusion solution and the transport solution. While it makes little difference at $t = 10$, we find better agreement with the transport solution at earlier times when using the Minerbo (1978) closure (not shown). In this problem, some regions near the internal radiation source (located at $0 \leq x < 0.5$) have Eddington factors $\chi < 1/3$, which cannot be represented by any local closure of the form given by equation 23. Nonetheless, we obtain a solution that is more accurate than one would obtain by using a radiation diffusion equation.

3.2.3 Radiation-matter energy exchange

We next isolate the implicit matter-radiation energy exchange solver by solving a problem with no transport. Following Turner & Stone
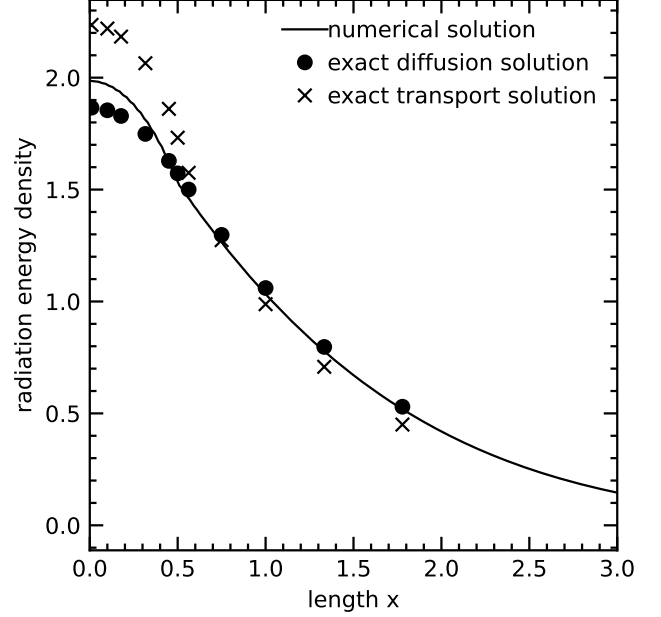
(2001), we set up a uniform domain with periodic boundary conditions, where the gas and radiation are initially out of thermal equilibrium. The initial radiation energy density is $E_r = 10^{12} \text{ erg cm}^{-3}$ and, the initial gas energy density is $E_g = 10^2 \text{ erg cm}^{-3}$. The density $\rho = 10^{-7} \text{ g cm}^{-3}$ and the specific opacity $\kappa_P = 1.0 \text{ cm}^2 \text{ g}^{-1}$. Rather than using a constant heat capacity (as Turner & Stone 2001 do) we use the heat capacity given by equation 55, which allows us to obtain an algebraic solution for the matter temperature T as a function of time t:

$$T^4 = \left(T_0^4 - \frac{\hat{c}}{c} \tilde{E}_0 \right) \exp\left[-\frac{4}{\alpha} \left(a_r + \frac{\hat{c}}{c} \frac{\alpha}{4} \right) \kappa \rho c t \right] + \frac{\hat{c}}{c} \tilde{E}_0 \, . \tag{57}$$

where $E_0 = E_g + (c/\hat{c})\, E_r$ and $\tilde{E}_0 = E_0\, [a_r + (\hat{c}/c)(\alpha/4)]^{-1}$ are constant as a function of time. Taking the limit $t \to \infty$, we immediately see that the equilibrium temperature T_{eq} is modified whenever $\hat{c} \neq c$, contrary to previous claims in the literature:

$$T_{\text{eq}}^4 = \frac{\hat{c}}{c} \tilde{E}_0 = \frac{\hat{c}}{c} E_0 \left[a_r + \left(\frac{\hat{c}}{c} \right) \frac{\alpha}{4} \right]^{-1} \, . \tag{58}$$

Fundamentally, this occurs whenever RSLA is employed (and $\hat{c} \neq c$) because the quantity $E_0 = E_g + (c/\hat{c})\, E_r$ is conserved in this problem, *not* the total energy $E_{\text{tot}} = E_g + E_r$. This is a generic failing of the RSLA, which does not conserve total energy. However, in practice when the boundary conditions are such that the quantity E_0 is *not* conserved, the physically correct steady-state solution may still be obtained – this is the situation for all the radiation-hydrodynamics test problems we present in Section 3.3, and is also the situation for most applications of interest.

To test QUOKKA's ability to recover the analytic solution, we simulate the problem using a constant timestep $\Delta t = 10^{-8}$ s until $t = 10^{-2}$ s. We use a reduced speed of light $\hat{c} = 0.1c$. We show the time evolution of the matter temperature in Figure 11 both when using RSLA and without. We find that the numerical solution agrees with the exact solution to better than one part in 10^5 at each timestep for
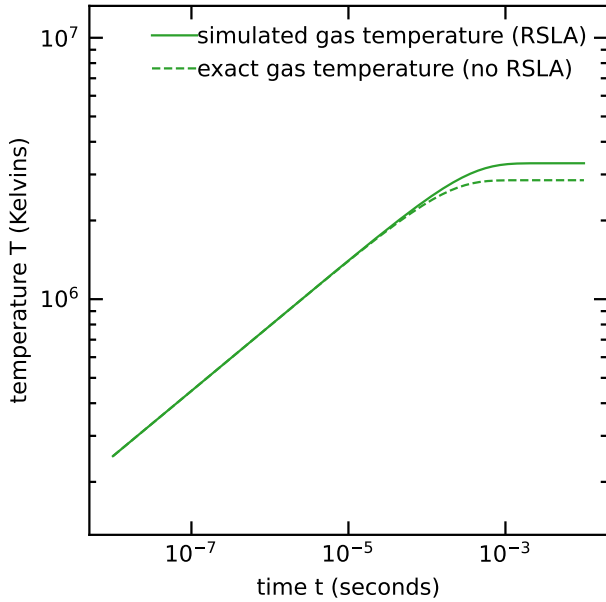
Figure 11. Time evolution of the gas temperature in the radiation-matter coupling test (Section 3.2.3). The simulated gas temperature using RSLA is shown as the solid line. The exact solution for the physically-correct gas temperature (i.e., without RSLA; equation 57) is shown as the dashed line.

both cases. The RSLA equilibrium temperature is approximately 20 per cent higher than the physically correct equilibrium temperature.

3.2.4 Shadow test

We next illustrate the performance of our radiation solver in two dimensions with a shadowing test based on that of Hayes & Norman (2003); our test differs from theirs only in that we use planar $x - y$ geometry instead of cylindrical $r - z$ coordinates. The goal of this test is evaluate how well an RHD scheme casts sharp shadows, recovering the geometric optics limit that should prevail when the optical depth is low. This problem consists of a rectangular domain of 1.0 cm by 0.12 cm with a streaming radiation source incident from the left boundary and an outflow boundary condition on the right. The lower boundary is reflecting, and the upper boundary allows outflow. In the middle lies an optically-thick cylinder. The initial gas and radiation temperatures are 290 K and the incident flux has a radiation temperature of 1740 K. The background has a density of $\rho_{\text{bg}} = 10^{-3}$ g cm^{-3}, and the cylinder has a density of $\rho_{\text{cl}} = 1.0$ g cm^{-3}. The gas has an opacity $\rho\kappa = (\rho/\rho_{\text{bg}})^2 \, 0.1$ cm^{-1}, a mean molecular weight $\mu = 10 \, m_{\text{H}}$, and an adiabatic index $\gamma = 5/3$. We allow two levels of mesh refinement on top of a base grid of 280×80 cells, tagging cells for refinement when the relative gradient in radiation energy density exceeds 0.1. The problem is evolved until $t = 5 \times 10^{-11}$ s with a CFL number of 0.4. We do not reduce the speed of light for this test. For this problem, we find it necessary to reduce the relative tolerance of the implicit matter-radiation coupling solver to 10^{-15}. Otherwise, there are unphysical radiation shocks reflected from the cylinder.

We show the radiation temperature at the end of the simulation in Figure 12. After the beam of light has crossed the domain, we find a sharp shadow cast behind the cylinder, as one would expect

physically. However, there are some residual artefacts from a transient beam of light that initially curved around the cylinder and reflected against the lower boundary, as seen in the low-temperature shock-like features within the shadow near the right edge of the domain. This appears to be an unavoidable artefact of using a local VET closure. (This does not occur when using the Eddington tensor obtained from the geometric optics limit, i.e. $f_{xx} = 1$.) Overall, this test shows that QUOKKA produces qualitatively correct results for semitransparent problems.

3.2.5 Beam test

We next test our code on a beam or 'searchlight' test involving streaming radiation propagating without any absorption, adopting parameters from González et al. (2007) with the only modification in that we move the beam to the lower left corner of the box. The domain is a square box $[0 \, \text{cm}, 2 \, \text{cm}]^2$ with constant density $\rho = 1.0$ g cm^{-3}, gas and radiation temperature $T = 300$ K, and zero opacity. A beam of radiation enters the domain at a 45 deg angle from the lower left corner ($x < 0.0625$ cm or $y < 0.0625$ cm) with a radiation temperature of 1000 K. We use AMR with a base grid of 128^2 and two levels of refinement to simulate this problem, refining wherever the relative gradient of the radiation energy density exceeds 0.1. For this problem, we use PLM reconstruction for the radiation variables in order to avoid oscillations near the leading edge of the beam. We use a CFL number of 0.4. We show the radiation energy density at time $t = 1.172 \, (L/c)$, where L is the box size, in Figure 13. The beam stays relatively narrow as it crosses the box, but at the leading edge of the beam, we see there is a transient bow shock-like feature which is due to our use of a local VET closure. Our code performs reasonably well on this problem, showing only a small amount of diffusion of the beam as it propagates. The bow shock feature appears to be an unphysical 'radiation shock' that can occur due to the nonlinear behavior of non-constant local VET closures. For instance, it can be shown that the radiation moment equations without source terms with the Levermore (1984) closure are mathematically identical to the hydrodynamic equations of an ultrarelativistic gas (Hanawa & Audit 2014). After the leading edge of the beam has crossed the box, the bow shock feature leaves the box and a narrow beam of light remains.

3.3 Radiation hydrodynamics

Our final suite of tests use the full suite of physics in QUOKKA, and involve coupled radiation and hydrodynamics.

3.3.1 Radiation pressure tube

Our first radiation-hydrodynamic test is the the radiation pressure tube problem of Krumholz et al. (2007a). This problem is designed to show that the radiation pressure gradient can stably balance the gas pressure gradient both in the regime where radiation pressure dominates and the in the regime where gas pressure dominates for a problem where the optical depth is sufficiently large that the radiation is in the equilibrium diffusion regime. We adopt the opacities $\kappa_P = \kappa_R = 100$ cm^2 g^{-1}, mean molecular weight $\mu = 2.33 \, m_H$, and adiabatic index $\gamma = 5/3$. The exact steady-state solution in the diffusion approximation is given by the solution to the differential
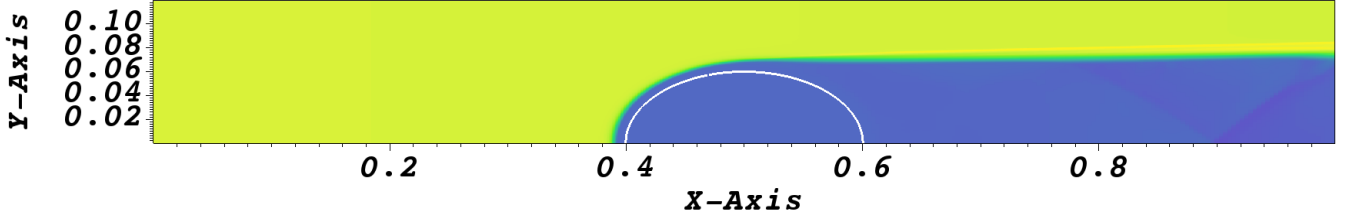
Figure 12. Results for the shadow test (Section 3.2.4). Color shows the radiation temperature. The solid line shows a density contour indicating the position of the 'cloud.'
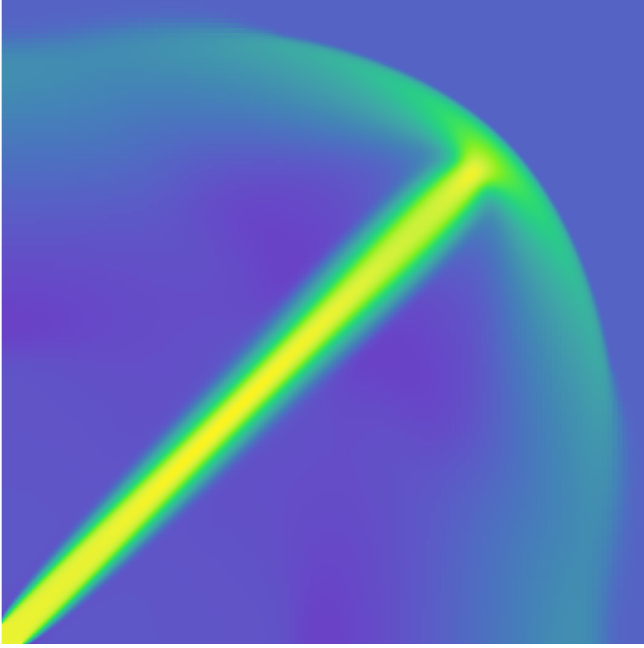

Figure 13. The beam test in vacuum (Section 3.2.5). Colour shows the logarithm of the radiation energy density.

equations

$$\frac{d\rho}{dx} = -\frac{\mu}{k_B T}\left(\frac{k_B}{\mu}\rho + \frac{4}{3}a_r T^3\right)\frac{dT}{dx},\tag{59}$$

$$\frac{d^2 T}{dx^2} = -\frac{3}{T}\left(\frac{dT}{dx}\right)^2 + \frac{1}{\rho}\frac{d\rho}{dx}\frac{dT}{dx},\tag{60}$$

where the left-side temperature, density, and density gradient are $T_0 = 2.75 \times 10^7$ K, $\rho_0 = 1.0\,\mathrm{g\,cm^{-3}}$, and $d\rho_0/dx = 0.005\,\mathrm{g\,cm^{-4}}$. We solve this equation on the domain $[0\,\mathrm{cm}, 128\,\mathrm{cm}]$ in order to obtain the initial conditions for this problem. The left and right side initial conditions are adopted as Dirichlet boundary conditions for our simulation. The reduced speed of light $\hat{c}$ is set to $10\,c_{s,0} \approx 4.03 \times 10^8\,\mathrm{cm\,s^{-1}}$, where $c_{s,0}$ is the sound speed at the left boundary.

After evolving for a sound crossing time $t = L_x/c_{s,0} \approx 3.177 \times 10^{-6}$ s with a CFL number of 0.4 on a grid of 128 cells, we obtain the numerical solution shown in Figure 14. We note that these parameters imply an optical depth per cell of $\tau_{\mathrm{cell}} \sim 10^2$, so this problem tests the accuracy of our numerical methods in the asymptotic diffusion regime where $\tau_{\mathrm{cell}} \gg 1$. Our numerical solution agrees with the initial conditions (obtained from the exact diffusion solution) to better than 0.2 per cent. Since the boundary conditions do not require con-
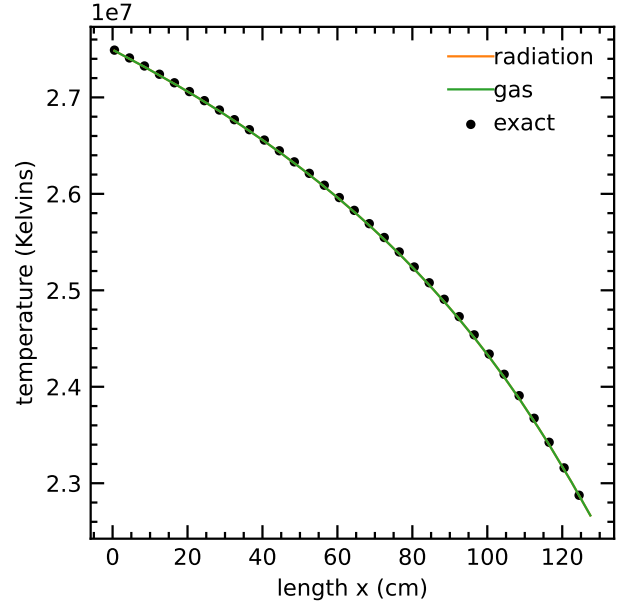

Figure 14. Temperature profiles for the radiation pressure tube test (Section 3.3.1). The radiation and gas temperatures are both shown, but only the latter line is visible because the two temperatures are nearly identical. The temperature for the exact diffusion solution is shown in the black circles. The simulated and exact temperatures agree to within 0.2 per cent.

servation of the quantity E_0 (see Section 3.2.3), we find that we are able to obtain the physically correct solution even when $\hat{c} \neq c$.

3.3.2 Optically-thin radiation-driven wind

In order to test the radiation-gas momentum coupling in the optically-thin limit, we next simulate a radiation-driven wind in the limit of very low optical depth. We consider an isothermal gas with sound speed $c_T = 0.2\,\mathrm{km\,s^{-1}}$, with constant opacities $\kappa_P = 0$ and $\kappa_R = 5\,\mathrm{cm^2\,g^{-1}}$. A flux of radiation $F_{r,0}$ enters the computational domain from the left side at $x = 0$, inducing an acceleration $a_0 = \kappa_R F_{r,0}/c$ in the gas; we choose the density of the gas low enough that the optical depth is negligible, so the flux and acceleration are constant across the domain. Consider a fluid parcel moving at Mach number $\mathcal{M}_0$ at $x = 0$, where the radiation flux enters the domain. Integrating the gas momentum equation with respect to position x yields a Bernoulli
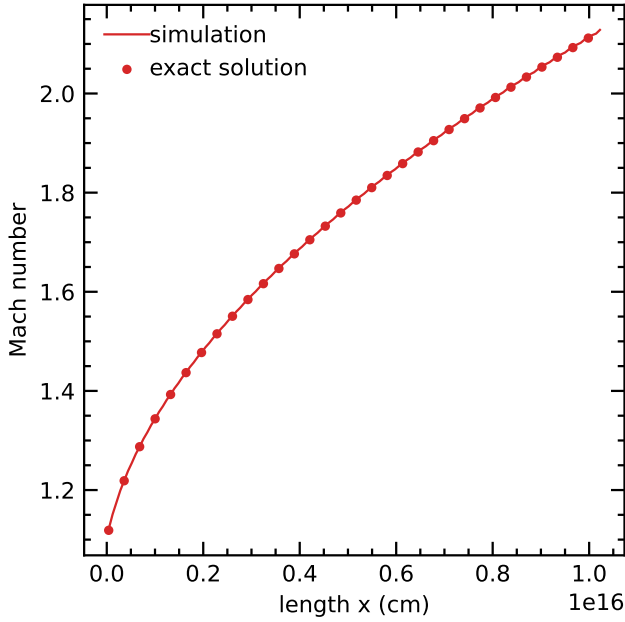
Figure 15. Mach number as a function of position for a radiation-driven wind (Section 3.3.2). The solid line is the QUOKKA numerical solution, while the points show the exact solution given by equation 61.
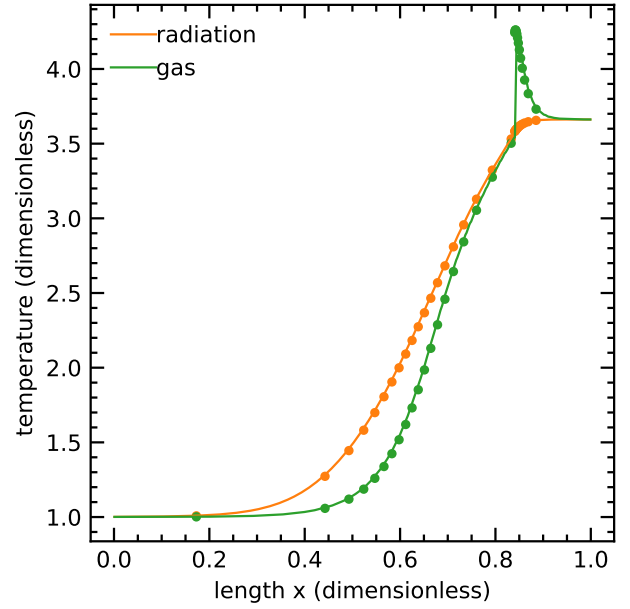

Figure 16. Radiation and matter temperatures in a subcritical radiative shock with $\mathcal{M} = 3$ (Section 3.3.3). The simulation results as shown as solid lines, while the exact steady-state solution is shown as circles.

equation for the Mach number $\mathcal{M}$ as a function of position

$$\frac{1}{2}\mathcal{M}_0^2 = \frac{1}{2}\mathcal{M}^2 + \log\left(\frac{\mathcal{M}_0}{\mathcal{M}}\right) - \left(\frac{x}{L}\right), \tag{61}$$

where $L = c_T^2/a_0$ is the characteristic acceleration length of the problem. If the density at $x = 0$ is ρ_0, then from conservation of mass, in steady state the density as a function of position is $\rho = (\mathcal{M}_0/\mathcal{M})\rho_0$.[9] For our test we choose $\mathcal{M}_0 = 1.1$, and we set $\rho_0 = 3.897212 \times 10^{-19}$ g cm^{-3}; for our chosen value of κ_R, this yields an optical depth $\tau = 10^{-6}$ from $x = 0$ to $x = L$.

To simulate this problem, we set up a domain from $x = 0$ to $x = L$, resolved by $N_x = 128$ cells. We initialize the system with the exact solution (equation 61), and also use the exact solution to impose Dirichlet boundary conditions on the density, velocity, and radiation flux. We evolve the system until $t = 10\,(L/c_T)$ using a CFL number of 0.4; for this calculation we use an isothermal Riemann solver. We set the reduced speed of light to $\hat{c} = 10\mathcal{M}_1 c_T$, where $\mathcal{M}_1$ is the Mach number at the right-side boundary. In Figure 15, we show the exact solution for the Mach number (circles) compared to the solution produced by QUOKKA (solid line), finding excellent agreement.

3.3.3 Subcritical radiative shock

We next simulate a subcritical radiative shock, following the set-up used by Skinner et al. (2019) with the dimensionless parameters for the Mach $\mathcal{M} = 3$ example given by Lowrie & Edwards (2008). We scale to cgs units with the opacities $\kappa_P = \kappa_R = 577$ cm^2 g^{-1} (1 g cm^{-3}/ρ), mean molecular weight $\mu = m_H$, and

adiabatic index $\gamma = 5/3$. The left-side state consists of $\rho_L = 5.69$ g cm^{-3}, velocity $v_L = 5.19 \times 10^7$ cm s^{-1}, and temperature (gas and radiation) $T_L = 2.18 \times 10^6$ K. The right-side state is $\rho_R = 17.1$ g cm^{-3}, $v_R = 1.73 \times 10^7$ cm s^{-1}, and $T_R = 7.98 \times 10^6$ K. These states are also used as Dirichlet boundary conditions for the simulation. In order to exactly match the assumptions used in the semi-analytic solution of Lowrie & Edwards (2008), we use the Eddington approximation (i.e., $\mathsf{P}_r = (1/3)E_r\mathsf{I}$) to close the radiation pressure tensor for this problem.[10] Following Skinner et al. (2019), we use a reduced speed of light $\hat{c} = 10(v_L + c_{s,L})$, where $c_{s,L}$ is the adiabatic sound speed of the left-side state. We use a CFL number of 0.4 and evolve until $t = 10^{-9}$ s on a grid of 512 cells on the domain $[0\,\mathrm{cm}, 0.01575\,\mathrm{cm}]$, with the discontinuity placed between the left- and right-side states $x_0 = 0.0130$ cm. The shock drifts 1.5 per cent of the domain length to the right from the location of the initial discontinuity, which may be due to a combination of the initial numerical transient and our use of the asymptotic states as boundary conditions, rather than the exact states expected at a finite distance from the shock location. This makes the steady-state location of the shock on the simulation grid not well-defined. After accounting for this drift, the agreement between the numerical and semi-analytic solution is excellent, as shown in Figure 16. We find the that the relative error of the gas temperature in L_1 norm is 0.4 per cent, which is at least as good as the solution of Skinner et al. (2019) for the same spatial resolution. In this problem, we find that using shock flattening is essential to obtain a non-oscillatory temperature structure for the Zel'dovich spike (the gas temperature discontinuity shown in Figure 11; Zel'dovich & Raizer 1967).

---

[9] We neglect gravitational forces in this problem, but we note that our solution is formally equivalent to the Eddington ratio $\eta_{\mathrm{Edd}} = 2$ case of the plane-parallel radiation-inhibited Bondi accretion problem of Skinner & Ostriker (2013).

[10] We provide a PYTHON code that computes the semi-analytic solution for radiative shocks using the Eddington approximation (Lowrie & Edwards 2008) in our ⌂ GitHub repository.

3.3.4 Radiation-driven dust shell

As a final example, we consider a non-steady-state radiation hydrodynamics problem: the radiation-driven dust shell problem from Skinner & Ostriker (2013, hereafter SO13), consisting of an initial shell of dusty gas placed at radius r_0 with the radial density profile:

$$\rho(r) = \frac{M_{\rm sh}}{4\pi r^2 \sqrt{2\pi\sigma_{\rm sh}^2}} \exp\left(-\frac{(r-r_0)^2}{2\sigma_{\rm sh}^2}\right), \qquad (62)$$

where $M_{\rm sh}$ is the mass of the shell and $\sigma_{\rm sh}$ is the thickness of the shell. We place a point-like source of radiation, representing a central star, at $r = 0$. The radiation source is smoothed so that it can be resolved on the computational grid, using a Gaussian profile of the form

$$j(r) = \frac{L_\star}{(2\pi\sigma_\star^2)^{3/2}} \exp\left(-\frac{r^2}{2\sigma_\star^2}\right) \qquad (63)$$

where $L_\star$ is the luminosity of the source and $\sigma_\star$ is a smoothing parameter defining the spatial extent of the source. Under the thin-shell approximation and neglecting gas pressure forces, SO13 obtain an equation of motion for the shell. Starting from rest, the resulting shell velocity, written in terms of the shell Mach number $\mathcal{M}_{\rm sh}$, is

$$\mathcal{M}_{\rm sh} \equiv \frac{dR}{dT} = \sqrt{2}\mathcal{M}_0\sqrt{1 - \frac{1}{R}}. \qquad (64)$$

with dimensionless radius $R \equiv r/r_0$, dimensionless time $T \equiv t/t_0$, characteristic time $t_0 = r_0/c_T$, reference sound speed c_s, and reference Mach number $\mathcal{M}_0$:

$$\mathcal{M}_0 = c_s^{-1}\sqrt{\frac{L_\star \kappa_R}{4\pi r_0 c}}. \qquad (65)$$

Following the parameters used by SO13, we set $\kappa_P = \kappa_R = 20\,{\rm cm}^2\,{\rm g}^{-1}$, $c_s = 2\,{\rm km\,s}^{-1}$, $r_0 = 5\,{\rm pc}$, $M_{\rm sh} = 5 \times 10^5\,{\rm M}_\odot$, and $L_\star = 2 \times 10^{42}\,{\rm erg\,s}^{-1}$. We note that c_s is only a reference sound speed and does not change the thin-shell solution since pressure forces are assumed to be negligible. We adopt values of $\sigma_\star = 0.3 r_0$ and $\sigma_{\rm sh} = 0.3 r_0/(2\sqrt{2\log 2})$.

We initialize our simulation of this problem using the density profile (equation 62) and the quasi-static radiation energy and flux derived by SO13. We initialize the gas temperature in equilibrium with the radiation temperature. A density floor is set at $\rho_{\rm floor} = 10^{-8}\rho_0$, where $\rho_0 = M_{\rm sh}/(4\pi r_0^3/3)$. A pressure floor is likewise set at $P_{\rm floor} = 10^{-8}P_0$, where $P_0 = \gamma\rho_0 c_s^2$. We use an adiabatic equation of state with $\gamma = 5/3$ and a mean molecular weight $\mu = 2.33 m_H$, where m_H is the mass of the hydrogen atom. We use a uniform grid of 128^3 and PLM reconstruction for both hydrodynamic and radiation variables. Following SO13, we reduce the speed of light to $\hat{c} = 860\,c_s$. The simulation is evolved until $t = 0.125\,t_0$ using a CFL number of 0.3. The shell velocity as a function of time is shown in Figure 17. Our simulation has values slightly lower than expected from the thin-shell solution, whereas SO13 find simulated shell velocities slightly higher than the thin shell solution. Exact agreement cannot be expected since SO13 do not specify their values for the parameters $\sigma_\star$ and $\sigma_{\rm sh}$ and differences in implementation details of our radiation hydrodynamic solvers. Overall, we find very good agreement between the simulation (shown as crosses) and the thin-shell solution (solid line).

4 PERFORMANCE AND SCALING

The entire motivation for QUOKKA is to achieve high performance on RHD problems run on GPUs. We therefore next test the performance
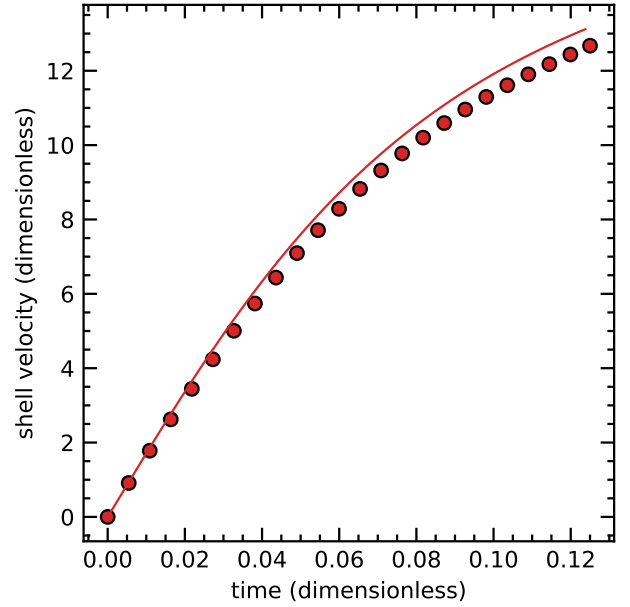

Figure 17. Shell velocity as a function of dimensionless time $T = t/t_0$ in the radiation-driven dust shell test (Section 3.3.4). The exact thin-shell solution is shown as the solid line, while the circles show the mass-weighted shell velocity from the simulation.

and scaling of the code. All the tests we present were performed on the Gadi supercomputer at the National Computational Infrastructure[11], using the gpuvolta nodes. Each node has 2 24-core Intel Xeon Platinum 8268 (Cascade Lake) 2.9 GHz CPUs and 4 Nvidia Tesla Volta V100-SXM2-32GB GPUs connected to each other in an all-to-all topology with NVLink 2.0. Nodes are coupled via HDR InfiniBand in a Dragonfly+ topology.

4.1 Weak scaling

We first demonstrate that QUOKKA has excellent parallel scaling efficiency when keeping the number of computational cells fixed per GPU (referred to as *weak scaling*). For our first test of weak scaling, we show the scalability of the hydrodynamics solver on uniform grids, disabling mesh refinement and radiation. We simulate a Sedov-Taylor blast wave (Sedov 1959; Taylor 1946) in a 3D periodic box on the domain $[-1, 1]$ in each coordinate direction. The initial conditions consist of a spherical region of high pressure $P = 10$ for radii $r < 0.1$ and low pressure $P = 0.1$ for $r \geq 0.1$, with a uniform density of $\rho = 1$ and zero velocity, for an ideal gas with adiabatic index $\gamma = 5/3$.

We run with a varying number of GPUs with two 256^3 grids per GPU, increasing the resolution of our simulation as we extend to greater numbers of GPUs. However, a power-of-two resolution increase does not easily map onto a jump from one GPU to four GPUs, so the single-GPU simulation only uses a grid size of 256^3. The grid size of the simulations therefore ranges from 256^3 (for 1 GPU) to 2048^3 (for 256 GPUs). We set the AMREX domain decomposition parameters `blocking_factor` and `max_grid_size` to a value of 128, leading the computational grid to be decomposed into arrays of

---

[11] https://nci.org.au/our-systems/hpc-systems

Nodes	GPUs	Mzones/GPU/s	Scaling efficiency	Grid size
0.25	1	93.90	–	256^3
1	4	62.53	100.00	512^3
8	32	56.16	89.82	1024^3
64	256	55.11	88.13	2048^3

Table 1. Weak scaling efficiency for hydrodynamics with PPM reconstruction as a function of the number of GPUs for a Sedov blast wave with periodic boundary conditions.

Nodes	GPUs	Mzones/GPU/s	Scaling efficiency	Grid size
0.25	1	166.18	–	256^3
1	4	87.18	100.00	512^3
8	32	75.83	86.97	1024^3
64	256	68.39	78.44	2048^3

Table 2. Weak scaling efficiency for hydrodynamics with PLM reconstruction as a function of the number of GPUs for a Sedov blast wave with periodic boundary conditions.

size 128^3. (We also tested local grid sizes of 256^3 but found only a few per cent performance improvement on this problem.) We use one MPI rank per GPU for all simulations. The CFL number is 0.25 and we evolve for 100 timesteps for each simulation. We assess performance by counting the total number of cell-updates and dividing by the number of GPUs in order to obtain the performance figure-of-merit in the units of 1 million cells (or zones) per timestep per GPU per second (Mzones/GPU/s). We report the results in Table 1.

We find a $\approx 30\%$ drop in performance per GPU when going from 1 GPU to 4 GPUs, corresponding to using all 4 GPUs on a single node of the compute cluster. We hypothesize that this is due to the limited communication bandwidth between GPUs on a node, which is limited here by the bandwidth of the PCI-Express bus. Similar scaling behavior is observed when running the K-Athena hydrodynamics code on GPUs (Grete et al. 2019), which uses an entirely different GPU programming framework and domain decomposition implementation. Nor is the phenomenon unique to GPU codes: For intra-node scaling on CPUs, Stone et al. (2020) report a similar decrease in performance when going from one CPU to all the CPUs on a node for Athena++, which they also attribute to limitations of memory bandwidth. Regardless of the origin of the slow down However, significantly different scaling behavior is observed when running the K-Athena hydrodynamics code on GPUs (Grete et al. 2019) on the Summit supercomputer[12], finding a 99 per cent weak scaling efficiency going from 1 GPU to 6 GPUs on a single node, so there may be some inefficiency in our current GPU-to-GPU communication method. We find that using CUDA-aware MPI does not improve performance for our code. However, we observe very little further degradation in performance per GPU when going from 1 node (4 GPUs) to 64 nodes (256 GPUs), yielding a parallel efficiency of 88 per cent on 64 nodes when compared to running on 1 node. We could not run on larger numbers of GPU nodes due to job size limitations, but we expect scaling to continue to thousands of GPUs based on the parallel scaling observed for other GPU hydrodynamics codes based on AMReX, such as Castro (Almgren et al. 2020).

[12] https://www.olcf.ornl.gov/olcf-resources/compute-systems/summit/

Nodes	GPUs	Mzones/GPU/s	Scaling efficiency	Grid size
0.25	1	22.68	–	256^3
1	4	10.00	100.00	512^3
8	32	8.83	88.30	1024^3
64	256	8.00	80.00	2048^3

Table 3. Weak scaling efficiency for radiation hydrodynamics as a function of the number of GPUs for the radiation-driven shell test (Section 3.3.4) with periodic boundary conditions.

We next test the scaling behaviour for full radiation hydrodynamics solver on uniform grids. Table 3 lists the performance per GPU and parallel efficiency measured with respect to single-node performance for the radiation-driven shell test problem run for 50 timesteps. Since we have many radiation substeps per hydrodynamic step (set here to 10; see Section 2.2.4), the performance metric in units of Mzones/GPU/s is lower by a factor comparable to but somewhat smaller than the number of radiation substeps per hydro step; a single radiation update is slightly less costly than a single hydrodynamic update. In this case, we observe a steeper drop in performance when going from 1 GPU to 4 GPUs (approximately a factor of 2). The lower parallel efficiency is not surprising, since *each* radiation substep requires communicating boundary conditions between grids, so the amount of inter-GPU communication per hydro timestep increases significantly for radiation hydrodynamics. Nonetheless, as is the case for hydrodynamics, there is little additional performance penalty when scaling from 1 node to 64 nodes. We measure a parallel efficiency in this case of 80 per cent.

Finally, we point out that absolute speed of Quokka is excellent. Comparison between CPU and GPU codes is non-trivial, since it obviously depends on the CPU-to-GPU ratio on a particular compute platform. However, it is worth pointing out that Quokka's update rate per core (normalised by the number of CPU cores per compute node) for *radiation*-hydrodynamics on GPU is comparable to or better than Athena++'s for *hydrodynamics* on CPU.

4.2 Strong scaling with AMR

Many applications of interest will seek to minimize either the total runtime of the simulation or the total node-hours used for a simulation for a problem of a fixed size. Additionally, most applications we are interested in will benefit from or require the use of AMR. We therefore test the ability of Quokka to scale an AMR radiation hydrodynamic simulation of fixed size to larger numbers of GPUs in order to either minimize total runtime or total node-hours (referred to as *strong scaling*). For this test, we initialize the radiation-driven shell problem (Section 3.3.4) on a base grid of 256^3 cells with two levels of mesh refinement based on the relative gradient in the gas density. We run each simulation for 50 timesteps, with a CFL number of 0.3 and PLM reconstruction for both hydrodynamics and radiation. We set the AMReX domain decomposition parameters `blocking_factor` set to 32 and `max_grid_size` set to a value of 128, so that all grids are between 32^3 and 128^3 in size, with possible non-cubic grids at intermediate sizes. The number of GPUs used for each simulation is varied, scaling from 1 node (4 GPUs) to 8 nodes (32 GPUs). This is a particularly stringest test, since the level-by-level AMR timestepping requires that each level be computed separately, limiting the amount of parallelism that can be distributed across GPUs. There is also additional communication overhead when AMR is enabled compared to a single-level uniform grid simulation. We

Nodes	GPUs	Mzones/GPU/s	$\left\langle \frac{\text{Cells}}{\text{GPU}} \right\rangle$	Scaling efficiency	Speedup
1	4	4.89	421.8^3	100.00	1.00x
2	8	4.28	334.8^3	87.40	1.75x
4	16	3.27	265.7^3	66.92	2.68x
8	32	2.66	210.9^3	54.42	4.35x

Nodes	GPUs	$\left\langle \frac{\text{Cells}}{\text{GPU}} \right\rangle_{l=0}$	$\left\langle \frac{\text{Cells}}{\text{GPU}} \right\rangle_{l=1}$	$\left\langle \frac{\text{Cells}}{\text{GPU}} \right\rangle_{l=2}$
1	4	161.3^3	259.1^3	376.7^3
2	8	128.0^3	205.6^3	299.0^3
4	16	101.6^3	163.2^3	237.3^3
8	32	80.6^3	129.5^3	188.4^3

Table 4. Strong scaling efficiency for radiation hydrodynamics as a function of the number of GPUs for the radiation-driven shell test (Section 3.3.4) with periodic boundary conditions on a base grid of 256^3 cells and 2 levels of refinement. The number of cells per GPU is computed as an average over all timesteps.

show the scaling results in Table 4. Comparing Table 3 and Table 4, the performance per GPU for a single node is lower than that of a uniform grid simulation by ≈ 50 per cent. (A similar, although somewhat smaller, overhead when enabling AMR is also observed with CPU codes, e.g., ATHENA++; Stone et al. 2020). The scaling efficiency is reasonable for 2 and 4 nodes (67 per cent for 4 nodes), but drops significantly at 8 nodes to 54 per cent parallel efficiency. We hypothesize that this is due to the small number of cell-updates per GPU once 32 GPUs are in use for this problem (approximately $211^3 \approx 9.4 \times 10^6$ cells/GPU). We find that performance on a single GPU is significantly diminished for uniform-grid problems smaller than 256^3, so this performance drop may be largely due to the inability to use all GPU hardware threads when the amount of work per GPU is small. Similar GPU performance behavior is observed when running K-ATHENA on GPUs for varying problem sizes per GPU (Grete et al. 2019). This effect is also magnified by the sequential nature of the level-by-level timestepping. For level $l = 1$, the number of cells per GPU drops below 256^3 for 8 GPUs, and for level $l = 2$, it drops below 256^3 for 16 GPUs. High scaling efficiency is obtained before reaching these thresholds, so it appears that reasonable performance on GPUs may be obtained with AMR when all refinement levels have at least 256^3 cells per GPU on average. In general, obtaining the best possible GPU performance may require an adjustment to the mesh refinement parameters usually used when running on CPUs. For self-gravitating problems, scaling may be aided by the self-similar nature of gravitational collapse, leading to an approximately equal number of cells on each refinement level for appropriate refinement criteria (see discussion in Stone et al. 2020).

5 DISCUSSION AND CONCLUSIONS

We conclude by discussing some of the limitations of QUOKKA as it currently exists, and our plans for future expansions of the code that will address at least some of these.

5.1 Range of applicability

Our method is limited in its range of applicability due to the use of a reduced speed of light. In the streaming limit, $\hat{c}$ may be chosen so

that it is larger than the fastest radiation-driven fronts (e.g., ionization fronts, some Marshak waves) in order to maintain the correct dynamics (Gnedin & Abel 2001). In the diffusion limit, $\hat{c}$ may be chosen so that it is larger than the effective diffusion speed $\sim c/\tau$. In this regime, we see that energy non-conservation may cause the equilibrium temperature in a closed box to differ from the physically correct equilibrium temperature (Section 3.2.3), but in most astrophysical RHD applications, the boundary conditions are not those of a closed system, and in this case our method recovers correct solutions without difficulty (e.g., Section 3.3.3). However, as emphasized by Skinner & Ostriker (2013), there is no constant choice of $\hat{c}$ that enables one to preserve the ordering of the hydrodynamic signal speed $c_s + |\mathbf{v}|$, the dynamic diffusion speed $\sim |\mathbf{v}| + c/\tau$, and the reduced speed of light $\hat{c}$ such that $v_{\text{hydro}} \ll v_{\text{diffusion}} \ll \hat{c}$. Extensions to the reduced speed of light method are possible that enable qualitatively correct behavior in a larger parameter space, but we leave their implementation to future work (Wibking, et al., in prep.).

5.2 Future extensions

There are several ways in which our code may be extended to include more physics or more accurate radiation transport. The easiest additional radiative process to include is monoenergetic, isotropic scattering, with a straightforward extension for moment methods (e.g., Jiang et al. 2012). Also relatively straightforward would be an extension to include coarse frequency dependence of the radiation via a multigroup extension of our radiation-matter coupling implicit solver. Even with a relatively small number of energy groups, many additional applications would be possible, including observational comparisons.

In order to improve the accuracy of the solution, one might also eschew local closures entirely and substitute a non-local closure for the Eddington tensor based on solution of the discrete ordinates (S_N) equations (e.g. Davis et al. 2012; Jiang et al. 2012), or using our moment method as a nonlinear preconditioner to accelerate the convergence of the thermal emission term in the S_N equations themselves (Park et al. 2012). The latter is an attractive option especially when used in combination with photon-conserving spatial discretizations of the S_N equations (Adams 1997; Adams 2001).

In the near future, we plan to add support for self-gravity with AMReX's geometric multigrid solver for GPUs (Zhang et al. 2019), sink and star particles for star cluster simulations (e.g., Krumholz et al. 2004; Offner et al. 2009), and optically-thin line cooling for the interstellar medium. These additions will enable simulations of the interstellar medium, galactic winds, and star clusters, among others.

Radiation hydrodynamics codes like ours will enable the widespread use of more accurate radiation transport methods and an ever-greater dynamic range in both space and time. As we approach the era of exascale supercomputers, we see a bright future for AMR radiation hydrodynamics on GPU architectures.

the National Computational Infrastructure (NCI), which is supported by the Australian Government.

Software: AMReX (AMReX Development Team et al. 2021), matplotlib (Hunter 2007), numpy (Harris et al. 2020), VisIt (Childs et al. 2012), yt (Turk et al. 2011).

DATA AVAILABILITY

The complete source code of Quokka, including the source code for all test problems shown in this work, is hosted in this public ⌂GitHub repository.

REFERENCES

AMReX Development Team et al., 2021, AMReX-Codes/amrex: AMReX 21.09, doi:10.5281/zenodo.5363443, https://doi.org/10.5281/zenodo.5363443
Abel T., Wandelt B. D., 2002, MNRAS, 330, L53
Adams M. L., 1997, Trans. Theory Stat. Phys., 26, 385
Adams M. L., 2001, Nuclear Science and Engineering, 137, 298
Alme M. L., Wilson J. R., 1973, ApJ, 186, 1015
Almgren A., et al., 2020, Journal of Open Source Software, 5, 2513
Audit E., Charrier P., Chièze J. P., Dubroca B., 2002, arXiv e-prints, pp astro–ph/0206281
Balsara D. S., 1999, J. Quant. Spectrosc. Radiative Transfer, 61, 617
Balsara D. S., 2017, Living Reviews in Computational Astrophysics, 3, 2
Benson D. J., 1992, Journal of Computational Physics, 100, 143
Berger M. J., Colella P., 1989, J. Comp. Phys., 82, 64
Berger M. J., Oliger J., 1984, J. Comp. Phys., 53, 484
Bloch H., Tremblin P., González M., Padioleau T., Audit E., 2021, A&A, 646, A123
Bryan G. L., et al., 2014, ApJS, 211, 19
Castor J. I., 2004, Radiation Hydrodynamics. Cambridge University Press, Cambridge, UK
Childs H., et al., 2012, in , High Performance Visualization–Enabling Extreme-Scale Scientific Insight. pp 357–372
Clain S., Diot S., Loubère R., 2011, J. Comp. Phys., 230, 4028
Colella P., 1990, J. Comp. Phys., 87, 171
Colella P., Woodward P. R., 1984, J. Comp. Phys., 54, 174
Colella P., Dorr M. R., Hittinger J. A. F., Martin D. F., 2011, Journal of Computational Physics, 230, 2952
Commerçon B., Teyssier R., Audit E., Hennebelle P., Chabrier G., 2011, A&A, 529, A35+
Davis S. W., Stone J. M., Jiang Y.-F., 2012, ApJS, 199, 9
Davis S. W., Jiang Y.-F., Stone J. M., Murray N., 2014, ApJ, 796, 107
Dubroca B., Feugeas J., 1999, Academie des Sciences Paris Comptes Rendus Serie Sciences Mathematiques, 329, 915
Einfeldt B., Roe P. L., Munz C. D., Sjogreen B., 1991, Journal of Computational Physics, 92, 273
Felker K. G., Stone J. M., 2018, J. Comp. Phys., 375, 1365
Fleck Jr. J. A., Cummings Jr. J. D., 1971, J. Comp. Phys., 8, 313
Fryxell B., et al., 2000, ApJS, 131, 273
Gittings M., et al., 2008, Computational Science and Discovery, 1, 015005
Gnedin N. Y., Abel T., 2001, New Astron., 6, 437
Gol'din V., 1964, USSR Computational Mathematics and Mathematical Physics, 4, 136
González M., Audit E., Huynh P., 2007, A&A, 464, 429
Grete P., Glines F. W., O'Shea B. W., 2019, arXiv e-prints, p. arXiv:1905.04341
Hanawa T., Audit E., 2014, J. Quant. Spectrosc. Radiative Transfer, 145, 9
Harris C. R., et al., 2020, Nature, 585, 357
Harten A., Lax P. D., Leer B. v., 1983, SIAM Review, 25, 35
Hayes J. C., Norman M. L., 2003, ApJS, 147, 197
Howell L. H., Greenough J. A., 2003, J. Comp. Phys., 184, 53

Hui W. H., Li P. Y., Li Z. W., 1999, Journal of Computational Physics, 153, 596
Hunter J. D., 2007, Computing in Science & Engineering, 9, 90
Hyman J. M., 1979, in Advances in Computer Methods for Partial Differential Equations - III. pp 313–321
Jameson A., Schmidt W., Turkel E., 1981, in AIAA.
Janka H. T., 1992, A&A, 256, 452
Jiang Y.-F., Stone J. M., Davis S. W., 2012, ApJS, 199, 14
Jin S., Liu J.-G., 1996, J. Comp. Phys., 126, 373
Koerner A., Janka H. T., 1992, A&A, 266, 613
Krumholz M. R., McKee C. F., Klein R. I., 2004, ApJ, 611, 399
Krumholz M. R., Klein R. I., McKee C. F., Bolstad J., 2007a, ApJ, 667, 626
Krumholz M. R., Stone J. M., Gardiner T. A., 2007b, ApJ, 671, 518
Lathrop K. D., Carlson B. G., 1964, doi:10.2172/4666281
LeBlanc J. M., Wilson J. R., 1970, ApJ, 161, 541
Lecoanet D., et al., 2016, MNRAS, 455, 4274
Lee D., 2011, in Pogorelov N. V., Audit E., Zank G. P., eds, Astronomical Society of the Pacific Conference Series Vol. 444, 5th International Conference of Numerical Modeling of Space Plasma Flows (ASTRONUM 2010). p. 236
Levermore C. D., 1984, J. Quant. Spectrosc. Radiative Transfer, 31, 149
Levermore C. D., Pomraning G. C., 1981, ApJ, 248, 321
Li P. S., et al., 2021, J. Open Source Software
Liska R., Wendroff B., 2003, SIAM Journal on Scientific Computing, 25, 995
Lowrie R. B., Edwards J. D., 2008, Shock Waves, 18, 129
Lowrie R. B., Morel J. E., 2001, J. Quant. Spectrosc. Radiative Transfer, 69, 475
Lowrie R. B., Morel J. E., Hittinger J. A., 1999, ApJ, 521, 432
Marshak R. E., 1958, Physics of Fluids, 1, 24
McCorquodale P., Colella P., 2011, Communications in Applied Mathematics and Computational Science, 6, 1
Mezzacappa A., Endeve E., Messer O. E. B., Bruenn S. W., 2020, Living Reviews in Computational Astrophysics, 6, 4
Mignone A., Plewa T., Bodo G., 2005, ApJS, 160, 199
Mihalas D., Klein R. I., 1982, J. Comp. Phys., 46, 97
Mihalas D., Mihalas B. W., 1984, Foundations of radiation hydrodynamics. Oxford University Press
Miller G. H., Colella P., 2002, J. Comp. Phys., 183, 26
Minerbo G. N., 1978, J. Quant. Spectrosc. Radiative Transfer, 20, 541
Modest M. F., 1989, J. Thermophys. Heat Trans., 3, 3
Offner S. S. R., Klein R. I., McKee C. F., Krumholz M. R., 2009, ApJ, 703, 131
Olson G. L., Auer L. H., Hall M. L., 2000, J. Quant. Spectrosc. Radiative Transfer, 64, 619
Park H., Knoll D. A., Rauenzahn R. M., Wollaber A. B., Densmore J. D., 2012, Transport Theory and Statistical Physics, 41, 284
Pember R. B., Anderson R. W., 2001, doi:10.2514/6.2001-2644
Pomraning G. C., 1973, The equations of radiation hydrodynamics. Pergamon Press
Quirk J. J., 1994, International Journal for Numerical Methods in Fluids, 18, 555
Reddy S. C., Trefethen L. N., 1992, Numerische Mathematik, 62, 235
Reynolds D. R., Hayes J. C., Paschos P., Norman M. L., 2009, Journal of Computational Physics, 228, 6833
Rijkhorst E., Plewa T., Dubey A., Mellema G., 2006, A&A, 452, 907
Rosdahl J., Blaizot J., Aubert D., Stranex T., Teyssier R., 2013, MNRAS, 436, 2188
Rosen A. L., Krumholz M. R., Oishi J. S., Lee A. T., Klein R. I., 2017, J. Comp. Phys., 330, 924
Sanders R., Morano E., Druguet M.-C., 1998, J. Comp. Phys., 145, 511
Schive H.-Y., Tsai Y.-C., Chiueh T., 2010, ApJS, 186, 457
Schive H.-Y., ZuHone J. A., Goldbaum N. J., Turk M. J., Gaspari M., Cheng C.-Y., 2018, MNRAS, 481, 4815
Schneider E. E., Robertson B. E., 2015, ApJS, 217, 24
Schneider E. E., Robertson B. E., 2017, ApJ, 834, 144
Sedov L. I., 1959, Similarity and Dimensional Methods in Mechanics
Shestakov A. I., Offner S. S. R., 2008, J. Comp. Phys., 227, 2154
Shu C.-W., Osher S., 1988, J. Comp. Phys., 77, 439

Shu C.-W., Osher S., 1989, J. Comp. Phys., 83, 32

Skinner M. A., Ostriker E. C., 2013, ApJS, 206, 21

Skinner M. A., Dolence J. C., Burrows A., Radice D., Vartanyan D., 2019, ApJS, 241, 7

Sod G. A., 1978, Journal of Computational Physics, 27, 1

Stone J. M., Gardiner T. A., Teuben P., Hawley J. F., Simon J. B., 2008, ApJS, 178, 137

Stone J. M., Tomida K., White C. J., Felker K. G., 2020, ApJS, 249, 4

Su B., Olson G. L., 1996, J. Quant. Spectrosc. Radiative Transfer, 56, 337

Su B., Olson G. L., 1997, Annals of Nuclear Energy, 24, 1035

Taylor G. I., 1946, Proceedings of the Royal Society of London Series A, 186, 273

Toro E. F., 2013, Riemann solvers and numerical methods for fluid dynamics: a practical introduction. Springer Science & Business Media

Tsang B. T.-H., Milosavljević M., 2015, MNRAS, 453, 1108

Turk M. J., Smith B. D., Oishi J. S., Skory S., Skillman S. W., Abel T., Norman M. L., 2011, ApJS, 192, 9

Turner N. J., Stone J. M., 2001, ApJS, 135, 95

Van Leer B., 1977, J. Comp. Phys., 23, 276

Zel'dovich Y. B., Raizer Y. P., 1967, Physics of shock waves and high-temperature hydrodynamic phenomena

Zhang W., Howell L., Almgren A., Burrows A., Bell J., 2011, ApJS, 196, 20

Zhang W., Howell L., Almgren A., Burrows A., Dolence J., Bell J., 2013, ApJS, 204, 7

Zhang W., et al., 2019, Journal of Open Source Software, 4, 1370

This paper has been typeset from a TEX/LATEX file prepared by the author.