rpSPH: a novel smoothed particle hydrodynamics algorithm

Tom Abel

Kavli Institute for Particle Astrophysics and Cosmology, SLAC National Accelerator Laboratory, Stanford University, 2575 Sand Hill Road, Menlo Park, CA 94025, USA

Accepted 2010 December 1. Received 2010 December 1; in original form 2010 February 10

ABSTRACT

We suggest a novel discretization of the momentum equation for smoothed particle hydrodynamics (SPH) and show that it significantly improves the accuracy of the obtained solutions. Our new formulation which we refer to as relative pressure SPH, rpSPH, evaluates the pressure force with respect to the local pressure. It respects Newton’s first law of motion and applies forces to particles only when there is a net force acting upon them. This is in contrast to standard SPH which explicitly uses Newton’s third law of motion continuously applying equal but opposite forces between particles. rpSPH does not show the unphysical particle noise, the clumping or banding instability, unphysical surface tension and unphysical scattering of different mass particles found for standard SPH. At the same time, it uses fewer computational operations and only changes a single line in existing SPH codes. We demonstrate its performance on isobaric uniform density distributions, uniform density shearing flows, the Kelvin–Helmholtz and Rayleigh–Taylor instabilities, the Sod shock tube, the Sedov–Taylor blast wave and a cosmological integration of the Santa Barbara galaxy cluster formation test. rpSPH is an improvement in these cases. The improvements come at the cost of giving up exact momentum conservation of the scheme. Consequently, one can also obtain unphysical solutions particularly at low resolutions.

Key words: hydrodynamics – instabilities – methods: numerical.

1 MOTIVATION

The smoothed particle hydrodynamics (SPH) method was invented by Lucy (1977) and Gingold & Monaghan (1977), both with interests in astrophysical applications. Besides an enormous literature of successful application, many shortcomings of it have also been presented in the literature (Steinmetz & Mueller 1993; Herant 1994; Swegle, Hicks & Attaway 1995; Imaeda & Inutsuka 2002; Agertz et al. 2007; Read, Hayfield & Agertz 2010), often suggesting a fix to the reported problem (Monaghan1994; Cummins & Rudman 1999; Rasio 2000; Attwood, Goodwin & Whitworth 2007; Hu & Adams 2007; Graham & Hughes 2008; Price 2008; Börve, Speith & Trulsen 2009; Rafiee & Thiagarajan 2009; Xu, Stansby & Laurence 2009, to name but a few). Similarly, there has been troubling news of how seemingly small differences in the initial setup led to very unexpected results (e.g. Lombardi et al. 1999). This is all somewhat surprising, given that in many of the cases where large inaccuracies have been found, the only relevant equation (besides moving the particles $\dot{r} = v$) stems from the pressure gradient accelerations

$$\rho \frac{D\dot{r}}{Dt} = -\nabla p,$$

where $D/Dt$ denotes the Lagrangian derivative and $p$ is the pressure. In what follows, we describe a new discretization of the momentum equation that avoids essentially all of the previously known problems of SPH. We will refer to this new method as ‘relative pressure SPH’ or abbreviated as rpSPH. We will first describe it, discuss implementation details and then present results for relevant tests, highlighting the superior performance of this new approach.

All the simulations shown here are carried out with GADGET-2 (Springel 2005) (version 2.0.4) with only most minor changes explained in the text. Appendix B describes how to convert GADGET-2 to our rpSPH formalism.

2 RELATIVE PRESSURE SPH

The equation of motion without viscous or gravitational forces in essentially all SPH codes and GADGET-2 (Springel & Hernquist 2002; Springel 2005), in particular, is

$$\frac{D\dot{r}}{Dt} = -\sum_{j=1}^{N} m_j \left[ f_i \rho_j \frac{P_i}{\rho_i} \nabla_i W_{ij}(\rho_i) + f_j \rho_j \frac{P_j}{\rho_j} \nabla_j W_{ij}(\rho_j) \right],$$

where $f_i$ are defined by

$$f_i = \left(1 + \frac{h_i \partial \rho_i}{3 \rho_i \partial h_i} \right)^{-1}.$$
and the abbreviation $W_i(h) = W(r_i - r_j, h)$ has been used for the kernel function $W$. It employs variable smoothing lengths so that the number of neighbours for each particle with $|r_i - r_j| \leq h$ is maintained at a nearly fixed value $N_{\text{ph}}$. The compact cubic spline kernel is used, which is summarized by Monaghan (1992), and extends to radii as large as the smoothing length $h$ and is zero outside. While many choices would exist to use a different discretization here (Monaghan 1992; Rosswog 2009), most of the previous works, we have found, essentially retain a form very close to equation (2) or use another symmetric form that sums over $(P_i + P_j)K(\rho_i \rho_j)$. The reason previously given for these choices is their symmetric form encapsulating Newton’s third law of motion, the action–reaction law. By guaranteeing that particles give pairwise identical but reversed forces, one ensures linear momentum conservation of the entire scheme. The key here is that particles are always pushing as soon as they have any pressure, regardless of whether there is a pressure gradient. The ones with the highest pressure values are pushing the most. When one has a large number of particles in a perfectly symmetric configuration, all the pushing will average out for an individual particle. This is to some extent what happens in real gases. The pressure itself is mediated by the collision of the molecules the gas is made of.

From a physical point of view of a Lagrangian fluid element, however, one should only be interested in the pressure forces of neighbouring fluid elements exerted on oneself, since the actual equation of motion is $\rho \ddot{v} = \nabla P$. This is the main idea of rpSPH – a particle is accelerated only if a force is acting upon it, that is, Newton’s first law of motion. rpSPH derives its equation of motion directly from equation (2) by subtracting the constant pressure of the particle under consideration from the pressures of all the particles being summed over. Since a gradient is computed, the subtraction of a constant does not change the mathematical meaning of the difference equation. However, as we will demonstrate, it dramatically affects the error properties of the entire scheme. The resulting equation of motion reads

$$\frac{dr_i}{dt} = -\sum_{j=1}^{N} m_j \left[ \frac{P_i - P_j}{\rho_j} \nabla_i W_{ij}(h_i) \right].$$

One immediately notes that this formulation breaks the symmetry between the pairwise forces of particles. Two particles that have a pressure difference are both accelerated in the same direction along the pressure gradient. Hence, linear momentum conservation will only be achieved if the modelled pressure gradients are resolved. On the other hand, if one does not resolve the relevant pressure gradients, then one cannot possibly get a correct solution to a hydrodynamic problem in any case.

After all, it is important to recall that when constructing conservative schemes, one does not necessarily minimize the numerical errors, but rather ensures that one is making symmetric errors so that the conserved quantity does not change. Consequently, in rpSPH monitoring, the total angular and linear momentum is an indicator of whether one may have resolved the relevant length-scales.

Many of the advantages of the entropic function based SPH formalism (Springel & Hernquist 2002) stem from avoiding the $P\delta V$ term that generally is discretized analogous to equation (2). So, in this formalism, rpSPH is particularly trivial to implement. It involves setting the first term on the right-hand side of equation (2) to zero and changing the second by subtracting the pressure of the particle under consideration. This literally is achieved by modifying one line of code in GADGET-2 (Springel 2005) as shown in Appendix B. The resulting scheme saves two multiplies, one division and one addition for one additional subtraction in the main loop over neighbours. So, there is no performance penalty in using rpSPH as compared to standard SPH.

rpSPH is seemingly close to equation (3.1) of Monaghan (1992) first discussed by Morris (1996a), which we will refer to as Morris’ formulation. It reads

$$\frac{dr_i}{dt} = -\sum_{j=1}^{N} m_j \left[ \frac{P_i - P_j}{\rho_j} \nabla_i W_{ij}(h_i) \right].$$

Monaghan dismissed his version for two reasons. The first is that ‘an isolated pair of particles with different pressures would bootstrap themselves to infinity’ and the second is that it is difficult to construct a consistent energy equation. The latter is irrelevant in the formalism evolving an entropic function (Lucy 1977; Springel & Hernquist 2002) in which the $P\delta V$ work does not enter. The first reason we find unappealing, since it is actually the correct solution. The simulation having two particles estimates a pressure gradient. So, over the model volume, that is, the two particles and their smoothing volumes, there exists a monotonic pressure gradient. Both particles hence should be accelerated along it. Interestingly, Monaghan did not discuss the equivalent case for the symmetric standard SPH. In this case, both particles push each other to infinity no matter what. If they have the identical initial pressures, then their centre of mass will not change, and if they vary, then their centre of mass moves exactly as in rpSPH. In rpSPH, they will move together, while in SPH, they will accelerate each other apart to infinity. We have tested this on a spherical blob of material in vacuum. We set the pressure of the particles after the densities have been computed from kernel smoothing. This way all particles have identical initial pressure. The configuration is completely stable in rpSPH, yet blows itself apart in SPH in just a sound crossing time. The reason why we do not choose equation (3.1) of Monaghan (1992) is because we find it to be unstable at least with the leapfrog time integrator (see Fig. 5 below). Another formulation close to rpSPH discretization we could find in the literature is presented by Morris, Fox & Zhu (1997) who chose to subtract a background pressure. This still leaves an equation of motion in which the pressure of the particle under consideration remains part of the hydroforce estimate.

An easy way to see why our discretization is valid (Wadsley 2010, private communication) recognizes $m_i/\rho_i = \delta V$ as the volume element $dV$ and sees that equation (4) is equivalent to $\int (\nabla V \rho^{-1} - \rho^{-1} \nabla V) dV$, which is the same as $\int \rho^{-1} \nabla V P$ and is the term we want. The previous version discussed by Morris (1996a) in contrast is the discretized form $\rho^{-1} \nabla V P$ rather than our $\nabla V \rho^{-1} - \rho^{-1} \nabla V$. Note that our form is also not the general form suggested by equation (2.13) of Monaghan (2005) and in this regard is a new formulation.

The striking aspect is that in the actual difference form, all that is new is that one index that used to be $i$ and is now $j$. So, this literally is a one letter change to codes that implement Morris’ formulation. How this can lead to a dramatic change in accuracy becomes obvious from the standard error analysis. Price (2004) gives the error of the summation interpolant in equation (3.9) and the error of the gradient operator in equation (3.11). In a formulation, for example, by Morris (1996a), one discretizes $\rho^{-1} \nabla V P$ and sees that the errors in the interpolation of $\rho^{-1}$ and the errors of the chosen $\nabla V$ discretization multiply. What Morris (1996a) realized was that there are no error terms in his discretization for constant functions in the gradient operator. However, the complete error terms still end up being the product of the density estimate and the pressure gradient estimate. The advantage of the rpSPH discretization is that its error terms are the one of a gradient and are not further multiplied by...
errors of a density interpolation. It also retains the vanishing error
for constant functions of Morris’ discretization.

Interestingly, a linear stability analysis reveals that rpSPH has the
same dispersion relation as the form of Morris (1996a, his equation
10). He has shown that this form

(i) is always stable, independent of the background pressure;
(ii) has a numerical sound speed that depends less on the particle
spacing as compared to standard SPH; and
(iii) does not have unphysical unstable transverse waves in either
two- or three-dimensions when using kernels with compact support.

rpSPH retains all of these advantages, while at the same time reducing the
discretization error.

In the standard SPH approach, there are in fact infinitely many possible choices for the discretization of the pressure equation (equation 3.5 in Monaghan 1992). This is also true for rpSPH. For example,

\[ P^{n-1} \nabla P^n / \rho = \nabla V / \rho = \sigma \rho \nabla P, \]

which suggests the discretization

\[ \frac{dv_i}{dt} = - \sum_{j=1}^{N} m_j \sigma^{-1} \left[ P^n_j - P^n_i \right] \nabla \psi_i (\rho_i) \]

for any \( \sigma \) different from zero. We have verified that many choices of \( \sigma \) work for a variety of test problems. In the following, however, we restrict our attention to the case of \( \sigma = 1 \).

Whether these theoretically advantageous properties of rpSPH hold up in practice is assessed in a range of test problems in the following section.

3 TESTS OF rpSPH

We will employ a Courant factor of 0.3 (i.e. 0.15 in GADGET-2 where the kernel has a maximal radius of \( h \)).

3.1 Reduced velocity noise

We start with \( 50^3 \) particles on a periodic regular lattice with \( \gamma = 1.4 \), a sound speed and uniform density of unity and zero initial velocities. The particles should stay at rest. However, as we can see in Fig. 1, the total kinetic energy in the volume grows rapidly. The total energy in the system is, however, conserved to better than \( 5 \times 10^{-3} \) of the initial value for these tests at a value \( [\gamma(\gamma - 1)]^{-1} \approx 1.78 \). So, the kinetic energy in the particle distribution only corresponds to about less than one in 1000 of the total, that is, the kinetic energy the particles obtain is taken from a slightly decreasing internal energy, allowing the total to be conserved to high precision. The lower the neighbour number, the faster the growth. The maximum noise reached is controlled by the artificial viscosity. The noise also decreases only very slowly over time after reaching the maximum. This is one of the main reasons why particle settling is so important in SPH simulations. The slow decline also shows why in general settling procedures can be computationally quite intensive. In the same figure, we also plot results using rpSPH, which dramatically reduces this spurious kinetic energy creation keeping it at zero to machine precision. Rasio (2000) caution that it makes no sense in standard SPH to increase particle numbers while keeping the number of neighbours fixed. Once a neighbour number is reached that keeps noise in the force calculation to a minimum, we find rpSPH to be stable while only increasing the particle number. Note that we have also run these tests dramatically reducing the Courant factor without any improvement in the case of standard SPH.

The thick solid line in Fig. 1 uses 20 neighbours, which seems optimal for this two-dimensional calculation with the cubic spline kernel. Here one has enough neighbours to estimate the gradients more accurately while still having too few neighbours to show its pairing instability. So, one may be tempted to dismiss the finding that one has the large velocity noise as long as one uses the ‘correct’ number of neighbours in one simulation. Unfortunately, this best choice, however, is only applicable at the uniform density. To show this, we perturb the \( x \) positions by a small amount so that the initially uniform \( x_0 \) positions are changed by adding \( \sin(2\pi x_0/25) \) to them, which gives central densities that are about 30 per cent above the mean. We keep again the pressure to be exactly constant by setting the associated velocity noise. It again is of the order of 1 per cent of the sound speed and grew very rapidly.

One may also be tempted to dismiss this particle noise as irrelevant as it only contains less than a tenth of a per cent of the total energy of the system. However, we will see in the following that it is what leads to unphysical shear viscosity, once one considers shear flows farther below.

3.2 Absence of the pairing instability

The velocity noise we just discussed is unfortunately not isotropic nor is it random. It has a dominant component for velocities towards directions of other particles and is an effect that aids the pairing instability. Our reasoning here is somewhat contrary to the explanations in the literature, for example, in Schuessler & Schmitt (1981), Vaughan et al. (2008) or Read et al. (2010). These studies suggest that it is the shape of the kernel function that causes clumping instability.
Given that rpSPH does not show any spurious velocities in the uniform density test given above, while standard SPH develops clumps within a few sound crossing times, it cannot be the shape of the kernel alone that is relevant here. In the following tests, we have looked for any sign of the clumping instability but have not found any evidence for it independent of the number of neighbours we used.

The clumping instability stems from particles pushing each other closer to other particles. With a smaller distance to the other particles, the gradient of the kernel becomes smaller and in the next time-step, the particle gets pushed farther away from its initial position. This way particles can pile up in the flat central part of the smoothing kernel. That rpSPH does not show the clumping instability makes one hopeful that even higher order kernels could now be used to further improve the accuracy of the obtained solution.

3.3 Dramatically reduced numerical shear viscosity

An easy two-dimensional setup uses an adiabatic index of \( \gamma = 1.4 \) in a unit square domain \( x \in \{0, 1\}, y \in \{0, 1\} \) with periodic boundary conditions. Particles are initialized on an exactly square lattice with a density of \( \rho(x, y) = 1 \), so that the initial density estimate from the SPH kernel gives a density estimate of unity to better than four parts in 1000. We then add a sinusoidal velocity perturbation to this uniform distribution. We set the pressure to \( P_0 = \rho/\gamma \) to have a sound speed of unity. For the first tests here, we only use 50\( ^2 \) particles as there are no features to resolve. In all cases, we evolve to time \( t = 4 \).

We choose \( v_x(y) = \delta v_x \cos(2\pi y) \) with \( \delta v_x = 1/2 \). This shear flow setup gives an average kinetic energy of 1/8. A detailed discussion of how SPH behaves on this test for different neighbour numbers, particle numbers and viscosity prescriptions is given in Appendix A.

In summary, it does very poorly and transfers kinetic energy into heat very rapidly, losing tens of per cent in only four sound crossing times (two crossing times of the fastest particles). It also gives more dissipation when using more particles, which makes a convergence study at a fixed neighbour number impossible. Below, when we discuss rpSPH for viscous flow, we show that the effective numerical viscosity of standard SPH is non-Newtonian and very large, which explains why standard SPH is inadequate to modelling fluids in general.

The results for rpSPH are summarized in Fig. 2, which is to be compared to the bottom panel for SPH, plotting the kinetic energy in the box as a function of sound crossing times. Note that the y-axis in the two panels of Fig. 2 is different by a factor of 30.

We should note that this test problem when used in typical Cartesian grid codes will give zero numerical dissipation to machine precision because of the uniform nature of the flow along the grid axes.

Price (2004) considered seemingly similar tests in his thesis. However, note that there an isothermal equation of state (EoS) was used. He shows one case with \( c_s = 0 \), that is, a pressureless fluid, and another case with \( c_s = 0.05 \). His initial shear profile is like the one we choose here but with twice the amplitude, so the fastest particles move with unit speed. The pressureless case is irrelevant here, since without pressure forces, the discretization of the momentum equation cannot make a difference. For the second case with \( c_s = 0.05 \), he gives the result only after one sound crossing time at \( t = 20 \). The density fluctuations have grown to the order of a per cent of the initial value after that single sound crossing time. As Fig. 2 shows for most reasonable choices of neighbour numbers, not much kinetic energy is dissipated over this time in our simulations also with an adiabatic index of 1.4. We also repeated this isothermal shear test and find results consistent with Price (2004). This again emphasizes that as long as one is interested in very few sound crossing times, SPH can give correct results and gives an indication that this is not specific to GADGET-2.

This problem also allows us to measure the effective Reynolds numbers one can hope to model with standard SPH.

Solving analytically the incompressible Navier–Stokes equations for our initial conditions, because to good approximation only the viscous term is relevant, we have

\[
\frac{\partial \mathbf{u}}{\partial t} = \nu \nabla^2 \mathbf{u}.
\]

Figure 2. Top panel: the fraction of the total kinetic energy in the uniform density shear test as compared to its initial value as a function of time for rpSPH. The labels give the square root of the number of particles used, the number of neighbours and the artificial viscosity parameter. So, 50-40-a0.1 used 50\(^2\) particles, with 40 neighbours and an artificial viscosity parameter of \( a = 0.1 \). We clearly see that using more neighbours always leads to less artificial shear viscosity. With more particles, the effective numerical dissipation (which is very small to begin with) decreases. The corresponding panel for SPH below has a y-axis 30 times as big. Bottom panel: the fraction of the total kinetic energy in the uniform density shear test as compared to its initial value as a function of time for standard SPH. We clearly see that using more neighbours always leads to less artificial shear viscosity. However, increasing the particle number leads to more dissipation.
Since the variables are separated, we can easily find that
\[ v(x, t) = e^{-4\pi^2 t^2} v(x, t = 0), \]
showing that the initial shape of the profiles should not change. We can now use equation (7) to get an estimate of the kinematic viscosity from the fraction of kinetic energy remaining:
\[ \nu = \frac{\ln(1/F)}{8\pi^2 t}, \]
where \( F \) denotes the fraction of the kinetic energy remaining up to time \( t \). This is \( v \approx 3.2 \times 10^{-3}\ln(1/F) \) at four crossing times.

The Reynolds number measures the ratio of inertial forces, \( \rho v^2 \), to viscous ones, \( \mu V/L \), where \( L \) is the characteristic length-scale, \( V \) is the mean velocity and \( \mu \) is the dynamic viscosity. So, \( R = VL/v \) with \( v = \mu/\rho \) the kinematic viscosity. Despite the ambiguities, we may take \( L = 1/4 \), the quarter wavelength of the velocity perturbation, and \( V = \sqrt{\sum p_i v_i^2} = 1/2\sqrt{2} \), the rms velocity.

So, for the typical values of \( F \) we found for SPH, say, 70 (97) per cent remaining, equation (8) gives a kinematic viscosity of \( \sim 10^{-3} \times 10^{-4} \). Consequently, the numerical Reynolds number \( R = LV/v = (8/\sqrt{2})v \sim 90/(1000v) \). This is very low and lower than most observed transitions between laminar and turbulent flows in the laboratory or terrestrial applications.

The analytic solution in equation (7) only holds if the fluid is Newtonian so that the shear stress can be described by the single number of the kinematic viscosity \( v \). Because Fig. A1 shows that the initial velocity profile changed strongly, one also concludes that the fluid flow as modelled by SPH is that of a non-Newtonian fluid. So, while it would have been interesting to think of SPH as solving effectively the Navier–Stokes equations rather than the Euler equations, we see that this is not the case. The effective numerical viscosity is non-Newtonian and does not in general decrease with numerical resolution.

### 3.4 Kelvin–Helmholtz instability

The Kelvin–Helmholtz (KH) instability occurs at the interface of two shearing fluids of different densities when velocity perturbations perpendicu lar to the interface grow to eventually mix the layers. In the inviscid case, this is understood analytically (Chandrasekhar 1961) and the growth time-scale for a sinusoidal mode of wavelength \( \lambda \) between two fluids of density \( \rho_1 \) and \( \rho_2 \) with a shear velocity \( v = v_2 - v_1 \) between them is
\[ \tau_{KH} = \frac{2\pi}{w} = \frac{(\rho_1 + \rho_2)\lambda}{(\rho_1\rho_2)^{1/2} v}. \]

This problem is typically set up with an adiabatic index of \( \gamma = 5/3 \) in a unit square domain \( x \in [0, 1], y \in [0, 1] \) with (e.g. Read et al. 2010):
\[ \rho, T, v_x = \begin{cases} \rho_1, T_1, v_1 & |y - 0.5| < 0.25 \\ \rho_2, T_2, v_2 & |y - 0.5| > 0.25. \end{cases} \]

We choose \( \rho_1 = 2, 5 \), and \( \rho_2 = 1 \), and the uniform pressure \( \rho_2/\gamma = 3/5 \) which gives a sound speed of 1 in the low-density surrounding medium. This standard setup then perturbs the interface with
\[ v_i = \delta v \sin(2\pi(x + \lambda/2)/\lambda) \exp[-10(y - 0.25)^2] + \sin(2\pi y/\lambda) \exp[-10(y - 0.75)^2], \]
where we choose \( \lambda = 1 \) and vary \( \delta v \). So, for our density contrasts, the growth times are \( \tau_{KH} \approx 1.2, 1.68 \) and 3.49 for the density contrasts 1:2, 1:5 and 1:10, respectively.

### 3.5 Rayleigh–Taylor instability

Another classic test of a code’s ability to handle subsonic perturbations is the Rayleigh–Taylor (RT) instability (e.g. Fryxell et al. 2000; Stone & Gardiner 2007; Stone et al. 2008). SPH has been used to model supernova explosion previously and RT instabilities have been seen (Herant 1994) as well as global convective instabilities (Fryer 2004). Here we study the growth of perturbations at the interface of a heavy fluid being supported by a lighter fluid in pressure equilibrium against a constant acceleration (e.g. gravity) (Chandrasekhar 1961). Remarkably, all idealized test cases that we are aware of use an initially unresolved contact discontinuity and consequently no converged results independent of the method of solution have so far been presented. Instead, the differences between different reconstruction schemes, Riemann solvers, meshing, etc., all contribute to the final structures produced in these simulations.

Similarly to the KH problem above, we chose here initial conditions that try to minimize the computational requirements while yielding converged results. The two-dimensional domain is set up with periodic boundary conditions along the \( x \)-direction with \( x \in [0, 1/2] \) and reflecting boundary conditions along the \( y \)-direction with \( y \in [0, 1] \). To achieve the reflecting boundary conditions in GADGET-2, we set up the density distribution in \( y \in [0, 1] \) and make all particles with \( y < 0.1 \) and \( y > 0.9 \) stationary (~DSPH_BND_PARTICLES compile option). These particles are not allowed to change their entropy or positions; consequently, they retain their initial pressure and density. Particles that are trying to
Figure 3. Comparison of the final density distribution in a two-dimensional KH test. The velocity perturbation is a tenth of the sound speed (0.1) in the top panels and one-hundredth (0.01) of the sound speed in the bottom panels. The left-hand simulations show a grid-based solution with 256^2 using the piecewise parabolic method. The middle panels give results for standard SPH and on the right-hand panels are shown rpSPH calculations. This problem is set up carefully to not have any physical small-scale structure and has been shown to converge with grid codes using a resolution exceeding 256^2 (Robertson et al. 2010). rpSPH behaves significantly better in not breaking up from discretization noise. The bottom left-hand panel shows how SPH fails to grow at the correct rate and is generally more noisy. All the particle-based simulations used 1000^2 particles and 100 neighbours with $\alpha = 3$.

penetrate through these walls have their positions changed to be at the wall and $y$-velocity vectors reversed. This is only a crude way of modelling reflecting boundaries with SPH but will suffice here to compare between SPH and rpSPH.

In keeping with past literature, we use an adiabatic index of $\gamma = 1.4$ and set up the density at the top to be $\rho_1 = 2$ and $\rho_2 = 1$ at the bottom. So, the density profile is $\rho(y) = \rho_2 + (\rho_1 - \rho_2)(1 + \exp(-2(y - 0.5)/\Delta_y))$ with $\Delta_y = 0.05$ in the cases presented here. The velocity perturbation is applied in the $y$-direction with $v_y(x, y) = \delta v_1 [1 + \cos(8\pi (x + 1/4))][1 + \cos(2(0.4\pi(y - 1/2)))]/4$ and the $y$-velocities are set to zero for $y$-positions above 0.7 and below 0.3. The pressure is set to $P_0 = \rho_1/\gamma = 10/7$ to give a sound speed of one at the interface and is set into hydrostatic equilibrium with the constant acceleration $g = 1/2$ in the negative $y$-direction with $P(y) = P_0 - \rho g(y)(y - 1/2)$. This gives a pressure difference of 60 per cent between the top and the bottom of the domain. The smaller is this pressure difference, the more difficult it becomes for SPH to model it. Similarly, the initial velocity perturbation again should not be very much smaller than the sound speed in order to survive viscous damping before a growth time of the instability.

We present the results for this test for velocity perturbations of $\delta v = 0.1$ and 0.01 in Fig. 4.

The differences are dramatic. Where SPH fails completely to see growth of the instability, rpSPH gives the expected behaviour for both perturbation strengths.

That rpSPH is a dramatic improvement over Morris’ formulation, despite only differing in one index, is seen in Fig. 5. There we give a RT problem at low resolution of 100 $\times$ 50 particles and a density ratio of 10 as discussed further in Section 4.1. All parameters were the same. A Courant factor of 0.2 is used, a neighbour number of 40, $\alpha = 1.5$, the Balsara switch is on and the initial velocity perturbation amplitude is 0.1. Clearly, our formulation is more stable, lending support to our discussion on the different error properties of the two discretizations given above.

3.6 Shock tubes
3.6.1 Sod shock tube

So far we have tested our new formalism only in very weakly compressible situations. We will use the classic Sod shock tube (Sod 1978) to compare rpSPH to standard SPH here. Rosswog (2009), recently, gave the results for varying viscosity prescriptions including artificial conduction terms. We change the setup only slightly. The left state has a density and pressure of unity, while the right state has a quarter of the density and a pressure of 0.1795. This test is evolved with an adiabatic index of $\gamma = 1.4$ and we set it up as a two-dimensional problem with equal-mass particles in a box that extends from zero to 10 in the $x$-direction and zero to 1 in the $y$-direction. We choose 40 rows of particles in the $y$-direction and vary the spacing along the $x$-direction to achieve the given densities using a total of 200^2 particles which are initially at rest. Additionally, we set the interface to be at $x = 3$ and smooth it with an exponential ramp such that all hydrodynamic variables are given by $r + [1 + \exp(2(x - 3)/\delta x)]^{-1}(l - r)$, where we take $\delta x = 0.05$ and $l$ and $r$ denote the left and right states. We employ periodic boundary conditions which give us another interface at $x = 10$ which has the reversed left and right states, but has an initially
Figure 4. Comparison of the final density distribution in a two-dimensional RT test between standard SPH (left-hand panels) and rpSPH (right-hand panels) for two different initial velocity perturbations $\delta v = 0.1$ (top panels) and $\delta v = 0.01$ (bottom panels). The unphysical ‘surface tension’ of SPH prevents the growth of the RT instability entirely. rpSPH easily recovers the correct behaviour. All simulations here used $500^2$ particles and 70 neighbours.

Figure 5. Density in a RT test with a 10 times heavier fluid on top than at the bottom. The left-hand panel shows Morris’ formulation and the right-hand panel is rpSPH. Both Morris’ SPH and rpSPH runs used an initial uniform grid of $50 \times 100$ particles with varying masses to describe the higher density for the interface and top fluid. Morris’ formulation is unstable in this problem, while rpSPH behaves as expected.

at a time when the rms velocity is $\sim 0.46$, so slightly above one-half of a per cent error in the dominant $x$-velocity. SPH has poor behaviour at the contact discontinuities, for both the one originating from the initially smoothed and the discontinuous interface at the right boundary. Both contacts at $x \sim 3.8$ and $x \sim 9.3$ are better captured by rpSPH. The Sod shock tube has few features and it is reassuring that using as many as $200^2$ particles can give an excellent answer. There are only slight differences in how rpSPH handles one-dimensional shock tubes. Next, we will discuss one very popular application taken from a cosmological context after testing a very strong shock.

3.6.2 Strong shock

Here we give another test of a much stronger shock than the one by Sod. This one has a Mach number close to 100. We also use the chance to compare this with the difference formulation studied by Morris (1996a). The density and pressure are $(1, 6.6 \times 10^4)$ on the left and $(1/5, 1)$ on the right. This is very similar to the one studied by Pfrommer et al. (2006) and is well known to work well with standard SPH. Here we use 35 neighbours, $\alpha = 4$ and 5000 particles. This is a good example where one can make rpSPH and Morris’ formulation give unphysical results. These methods require the pressure gradient to be resolved. So, if you start with completely discontinuous left/right states, then you will get unphysical waves giving unexpected results. However, this is not a shortcoming of the method, but simply are errors that come from not resolving the initial conditions. We again use the ramp function from above with a width of 4 in this very long domain ranging from 0 to 500 in the $x$-direction and from 0 to 10 in the $y$-direction.

We cannot confirm Morris’ claim that his formulation gives large post-shock oscillations in this method and suspect that he may...
Figure 6. Clockwise from the top: the density, $x$-velocity, pressure and specific internal energy are given for the classic Sod shock tube (Sod 1978) at time $t = 1$. The simulations used $200^2$ particles, 80 neighbours with a tolerance of 1, $\alpha = 3$, for both the rpSPH solution (filled diamonds) and standard SPH (open circles). We plot every 50th particle of the 40,000 employed. In the pressure panel (bottom right-hand side), we also show a reference solution computed with a Eulerian code using the piecewise parabolic method. The agreement is quite good. rpSPH handles the contact discontinuities much better and the ‘blip’ seen in SPH is absent in the one from the smoothed interface ($x \sim 3.8$), while it is much reduced in the one that was initially sharp ($x \sim 9.3$).

Figure 7. Clockwise from the top: the density, $x$-velocity, pressure and specific internal energy are given for a strong shock with Mach number $\sim 100$ at time $t = 0.5$. The simulations used 5000 particles arranged as 10 rows in the y-direction in an elongated domain extending from 0 to 500 in the x-direction and 0 to 10 in the y-direction and 35 neighbours with a tolerance of 1, $\alpha = 4$. We give the result obtained with Morris’ formulation (open circles) and rpSPH. We plot every 50th particle of the 5000 employed. Both approaches handle the contact discontinuities much better and the ‘blip’ seen in standard SPH is gone. rpSPH gets the correct density jump of a factor of 4 better than Morris’ formulation.

We have set up discontinuous initial conditions. In Fig. 7, we can see that our new formulation performs somewhat better than Morris’ formulation as it does not overshoot the analytical density jump of 4 raising the density from 0.2 to 0.8. Otherwise, both approaches work fine and have no problem in modelling strong shocks and evolving them for large distances.

3.7 Sedov–Taylor blast wave

Another particularly strong shock is formed in the Sedov–Taylor blast wave (Landau & Lifshitz 1959) presenting a difficult test problem for incompressible hydrodynamics codes. One the one hand, it is a self-similar solution which makes it insensitive to how
exactly one sets it up as long as one evolves the system for a very long time. On the other hand, it is the solution for a point explosion. For a given finite resolution, however, there is no unique way of specifying the initial conditions. Here is where exact momentum and energy conservation is very helpful as one can set up the initial conditions at will and even if one were to make very large errors in the time-evolution, the method would still arrive at the self-similar solution. In conservative grid codes, this still can lead to aspherical solutions if one did not resolve the spherical central hot region. Since SPH, however, uses spherical kernels, one can get away sometimes even by just heating one single particle (Springel & Hernquist 2002). This is very useful in applications such as galaxy formation simulations where one is always far from resolving the relevant length-scales of an explosion. Conversely, any physics that were to occur at a scale of the shell thickness would be impossible to resolve in such a single particle energy ejection. For rpSPH and Morris’ formulation, we need to resolve the pressure gradients in the initial conditions as we saw above in the strong shock setup.

We set up a square lattice of particles with 300 particles on a side in the unit domain. For resolved initial conditions, we set a spherical region in the centre of radius \( r = 0.1 \) with the same ramp function as above using a width of 0.1 to have a sound speed of one for an adiabatic index of \( \gamma = 5/3 \). For both simulations, we used a Courant number of 0.2 (0.1 in GADGET-2), 80 neighbours, artificial viscosity \( \alpha = 2.5 \) and had the Balsara switch off. Fig. 8 shows that there potentially is also an advantage to rpSPH simulations when modelling shocks. We have failed to get standard SPH to give a stable correct density jump of \( (\gamma + 1)/(\gamma - 1) = 4 \) for our setup. Also the three-dimensional versions shown by Springel & Hernquist (2002) always seem to be too low by as much as a factor of 2. However, standard SPH is much less sensitive to how one sets up the initial conditions and performs much better at low resolutions.

These strong shock problems can work, but clearly are not the biggest strengths of rpSPH. However, at this point, we simply reused the artificial viscosity prescription which was designed for standard SPH. We believe it is likely that one can find an alternative formulation for the artificial viscosity that fits better into the rpSPH discretization which may improve its behaviour for highly supersonic conditions. Until a better artificial viscosity prescription is designed, one may opt to switch between standard SPH and rpSPH based on the local divergence. We have successfully applied this strategy by using a switch that evaluates the standard SPH sum, if \( h \alpha > 3c, \) and the rpSPH sum for less strongly convergent flow. Here \( h \) and \( c \) denote the smoothing length and the current sound speed of the particle. This formulation is robust in all our tests.

3.8 Cosmological integration of the Santa Barbara galaxy cluster

In 1995, a comparison project was initiated that aimed to compare all numerical cosmology codes at that time for relevant realistic initial conditions. The study focused on three-dimensional calculations of the formation of a galaxy cluster in the standard cold dark matter scenario of structure formation. The choice was a setup which does not include any other physics than cosmological hydrodynamics with an ideal gas EOS (often referred to as adiabatic simulations, despite the entropy generation in shocks). The study produced a detailed report in Frenk et al. (1999, hereinafter F99).

One of the most-surprising findings of the study was that while there was very good agreement between the six different SPH codes used in the study, they did not agree with the solutions of the grid-based codes. The central entropy of the simulated galaxy cluster differed markedly between the grid-based and SPH codes. In particular, the only adaptive mesh refinement (AMR) code in the study by Bryan & Norman (1997), which is now called Enzo (Bryan et al. 2001; O’Shea et al. 2004), found a flat entropy core, while the particle codes found the central entropy to continue to rise towards smaller radii. Note that there has been a significant debate on what the correct solution may be and potential sources between the differences between grid-based and particle-based methods (Dolag et al. 2005; Springel 2005; Agertz et al. 2007; Wadsley et al. 2008; Kawata et al. 2009; Mitchell et al. 2009; Springel 2010) is the real reason for the difference.

We do not attempt a resolution study here, but simply show how an rpSPH solution compares to an SPH run with otherwise identical parameters and the solution derived with a cosmological AMR code. We use 128^3 gas as well as dark matter particles for the particles-based approach. For the AMR code, we again use Enzo already used in F99 using 128^3 dark matter particles, a root grid of 128^3 cells and seven additional refinement levels. Refinement is based on density thresholds in the baryons and dark matter component. The viscosity parameter in the particle-based runs was \( \alpha = 3 \) and a neighbour number of 300 with tolerance of 1 was used. The initial redshift is \( z = 63 \) and the initial temperature is 10^7 K.

A two-dimensional slice through the temperature field is given in Fig. 9 comparing SPH to rpSPH. Only relatively subtle differences are found. There are perhaps slightly more small-scale features visible in the rpSPH calculation and some slight differences in the post-shock gas in the main cluster are visible. Slightly more shocking occurs at larger radii towards low-density voids in the rpSPH versus SPH calculation.

Fig. 10 compares the solutions using spherically averaged radial profiles as described in F99. The entropy profiles of rpSPH agree better with the AMR than the SPH results. The rpSPH solutions show the lowest central densities of the three methods and agree better with the slightly shallower density profile of the grid-based code. Clearly, the differences between all three, however, are rather subtle, given that one evolved this system for 13 billion years which
are many tens of sound crossing times for the central part of the resulting cluster.

Both SPH and rpSPH simulations have a final linear momentum corresponding to 4.1 and 3.2 km s\(^{-1}\) per gas particle, respectively. The difference vector between the final gas momenta of the simulations has a magnitude of 2.6 km s\(^{-1}\) per gas particle. This is a difference of the order of one-half of a per cent of the mass-weighted rms velocity of \(\sim 400\) km s\(^{-1}\). Obviously, giving up the strict linear momentum conservation in our equation of motion has not led to any notable difference in this measure, but has improved the comparison with results from AMR codes.

However, this particular application is relatively easy as dark matter dominates the gravitational potential. As we will show below, rpSPH is quite easy to break with self-gravitating fluids.

4 FURTHER BENEFITS OF rpSPH

4.1 Variable masses

Next we demonstrate that using our pressure force discretization gives another very important advantage. Simulations with drastically varying particle masses continue to give correct results. This is markedly different compared to previous SPH simulations employing particle splitting. The latter only worked reasonably as long as different particle masses were very well separated spatially. As an explicit example, we revisit the RT problem from above. This time we initialize particles on a uniform lattice and model the density contrast by changing the particle masses according to the density profile. We employ a density at the top 10 times the one at the bottom fluid to demonstrate that this is not just a marginally better aspect of rpSPH. Fig. 11 compares the SPH and the rpSPH solutions again for \(\delta v = 0.01\). Instead of showing the density field, we show the particles painted by circles and coloured by their density. This gives us an opportunity to see that rpSPH does not show any clumping instability, while it is severe for SPH. In this run, we only use 100 \(\times\) 200 particles demonstrating that rpSPH handles the RT problem very well at small initial perturbations and low particle numbers.

The Sedov–Taylor blast wave above was also carried out with a staggered grid of particles of varying mass and retains a nice spherical shape, despite the multiple squares introduced in the staggered ‘mesh’ of the initial particle distribution.

It is of great interest for a method to be stable under largely varying particle masses. If it is, then one can use particle splitting likely without worrying too much of how to place the new particles and keep them separate from particles with different masses (e.g. Kitsionas & Whitworth 2002).

4.2 Formulation for magnetic forces

There have been many attempts to implement ideal magnetohydrodynamics (MHD) into the standard SPH formalism (e.g. Gingold & Monaghan 1977; Phillips & Monaghan 1985; Price & Monaghan 2004; Dolag & Stasyszyn 2009). In our experience, rpSPH performs in the hydro part better than Morris’ formulation. The latter has been used in implementing ideal MHD into SPH (Morris 1996b; Price & Monaghan 2004; Dolag & Stasyszyn 2009). Therefore, we expect that our new discretization may be of use in this case as well.

A symmetric conservative form of the Lorenz force is generally implemented using the magnetic stress tensor (Phillips & Monaghan 1985):

\[
M_{ij} = \left( B_i^2 - \frac{1}{2} |B|^2 \delta_{ij} \right).
\]

So, the acceleration from the magnetic fields on the \(i\)th particle is then written as

\[
\left( \frac{dv_i}{dt} \right)^{(B)} = \frac{1}{\mu_0} \sum_{j=1}^{N} m_j \left( f_i \frac{M_j}{\rho_j^2} \cdot \nabla W_i + f_j \frac{M_i}{\rho_i^2} \cdot \nabla W_j \right).
\]
Relative pressure SPH

Figure 10. Spherically averaged profiles of density, temperature and entropy of the Santa Barbara galaxy cluster comparison project (F99). The SPH and rpSPH simulations used $128^3$ particles and 300 neighbours with a tolerance of $1, \alpha = 3$, and are compared to results from an AMR calculation using the piecewise parabolic method (ENZO-PPM). The agreement is excellent. It is interesting that rpSPH gives higher central entropy as standard SPH more comparable to the grid code as this has been the subject of much discussion in recent years. Note also that the entropy profiles of rpSPH and the AMR calculation better agree well away from the central resolution limited part at radii between 0.1 and 1.

In the limit where we only have forces from the magnetic pressure gradient only, the diagonal of the tensor has terms which are $B_i^2 / 2$ and we recognize this discretization as exactly the form of equation (2) above. So, also the Lorentz force lends itself to be discretized following our new approach. We split the force into the tension component and the magnetic pressure component without loss of generality into

$$
\frac{\partial \mathbf{v}}{\partial t} = -\frac{B^2}{8\pi} + \frac{(\mathbf{B} \cdot \nabla) \mathbf{B}}{4\pi}.
$$

In both terms, one takes spatial derivatives and hence is allowed to subtract a constant. Choosing $\mathbf{B_i} \cdot \mathbf{B}_i$ for the first one and $\mathbf{B}_i$ for the second one avoids finite pairwise forces between particles in regions of constant field. So, the isotropic part becomes

$$
\frac{\partial \mathbf{v}}{\partial t} = -\frac{1}{8\pi} \sum_{j=1}^{N} \frac{m_j f_j B_i^2 - B_i^2}{\rho_j^2} \nabla W,
$$

to which the tension force is added

$$
+ \frac{1}{\mu_0} \sum_{j=1}^{N} \frac{m_j f_j B_i \cdot (\mathbf{B}_j - \mathbf{B}_i)}{\rho_j^2} \nabla W.
$$

(14)

In both the tests above, it is clear that at least in the weakly compressible limit, rpSPH is a very useful improvement over the standard formulation. However, giving up momentum and energy conservation is a big price to pay for those advances and rpSPH cannot possibly replace standard SPH in all problems of interest. rpSPH should be easy to break at low resolutions when one does not resolve the pressure gradients adequately. We now give an illustrative example that makes rpSPH give very bad results which shall serve as a cautionary note and things to look for when applying rpSPH.

The Evrard collapse of a cold gas sphere (Evrard 1988) has been extensively used for verification of astrophysical SPH codes (e.g. Wadsley, Stadel & Quinn 2004; Springel 2005). In the version that is part of the GADGET-2 distribution, it is realized with only 1472 equal-mass particles. A centrally concentrated cloud of cold gas collapses, bounces and eventually virializes. Vacuum boundaries are assumed. Very clearly this is only meant to investigate energy conservation of the code and is a simple test running in seconds on one’s laptop.

Fig. 12 shows the standard SPH solution together with a completely failed solution of rpSPH. We should note that as we increase

5 HOW TO BREAK rpSPH

Both the terms simply add to the accelerations in the force calculation. All that is left to do is to replace the fastest signal velocity with the one given in equation (46) of Price & Monaghan (2004) and one has a MHD implementation of SPH. On some simple initial tests, this formulation seems to work quite well even without any regularization technique (e.g. Dolag & Stasyszyn 2009, and references therein) or artificial $B$ field dissipation. A full exploration of the performance of this discretization though is beyond the scope of this contribution.
the resolution, rpSPH does converge to the same solution as standard SPH. However, this is a clear example where too little resolution coupled with a solver that does not conserve momentum or energy will fail completely.

Fig. 12 also gives the energies in the Evrard collapse for which we formally added a zero and label it rpSPH+0. The term that is added to the momentum equation’s right-hand side is \( P/\rho \nabla v = 0 \), which in SPH reads (e.g. Ritchie & Thomas 2001):

\[
\sum_{j=1}^{N} m_j \left[ \frac{P_j}{\rho_i \rho_j} \nabla W_{ij}(h_i) \right].
\]

This term tends to zero when the density and pressure gradients are well resolved. For our poor-resolution setup, however, we see that it gives better energy conservation. We do not recommend this formalism over rpSPH, however, since it still has the problems with contact discontinuities and weakly compressible flows. However, it highlights that for any numerical study, resolution studies are crucial and that rpSPH will likely be of little use when one cannot afford sufficient resolution for the particular problem one is interested in.

6 CONCLUSIONS

We have presented a novel discretization of the pressure equation for SPH, referred to as rpSPH, that removes the local pressure from the scheme and only considers pressure gradients. This methodology avoids the clumping and banding instability, artificial surface tension, unphysical particle noise, dramatically reduces inherent shear viscosity and numerical dissipation, and allows to realistically evolve density distributions sampled even with disparate particle masses. We have discussed a large number of tests in all of which our new discretization outperforms the traditional SPH results. While our approach is not manifestly momentum conserving and easy to break with self-gravity and/or low resolutions, it clearly seems much more accurate than previous approaches to Lagrangian hydrodynamics using SPH. Since our formulation is more accurate and requires as little as one line of code to be changed in previous implementations, we do believe it likely to be useful. The caveat of rpSPH remains that if one knows that one cannot afford to resolve the pressure gradient in one’s initial conditions because it is not momentum conserving, then it can give very wrong results. Fortu-
APPENDIX A: SHEARING FLOWS OF UNIFORM DENSITY WITH STANDARD SPH

While rpSPH overcomes the problems found for shearing flows, this appendix is more of historical interest. However, some readers may find it worthwhile, since to our knowledge, the surprisingly large shear viscosity and its erratic behaviour with increasing particle numbers has not been documented previously.

A simple two-dimensional setup uses an adiabatic index of $\gamma = 1.4$ in a unit square domain $x \in \{0, 1\}, y \in \{0, 1\}$ with periodic boundary conditions. All particles are set up on an exactly square lattice with a density of $\rho(x,y) = 1$, so that the initial density estimate from the SPH kernel in fact gives a density estimate of unity to better than four parts in 1000. We then add different velocity perturbations to this uniform distribution. We set the pressure to $P_0 = \rho/\gamma$ to have a sound speed of unity. For the first tests here, we only use $50^2$ particles as there are no features to resolve. In all cases, we evolve to time $t = 4$.

First, we start with no velocity perturbation, that is, a completely static uniform density distribution evolved over four sound crossing times. Using 30 neighbours, the density estimate by all particles is 1.00345. For different neighbour numbers, this fluctuates around 1 and is close enough. We will use 30 neighbours for most part of the rest of this section. Initially, all velocities are zero yet after four crossing times, we have an rms velocity $v_{\text{rms}} = \sqrt{\frac{1}{N} \sum v_x^2 + v_y^2} \approx 0.01$ with no obvious preferred direction. This result is obtained for the typical viscosity value of $\alpha = 1$. For lower values, this random noise increases to $v_{\text{rms}} \approx 0.036$ for $\alpha = 1/10$. So, even under the most-quiet condition imaginable, a uniform density with periodic boundary conditions cannot be better resolved, one would hope for less dissipation. Yet we find it worthwhile, since to our knowledge, the surprisingly large shear viscosity and its erratic behaviour with increasing particle numbers has not been documented previously.

Now, for our next experiment with this uniform density setup, we use $v_0(y) = \delta v_0 \cos(2\pi y)$ with $\delta v_0 = 1/2$. This shear flow setup gives an average kinetic energy of $1/8$. After only four sound crossing times (or two crossing times of the fastest particles), the mean kinetic energy of particles has decreased by 15 per cent and the rms velocity in the $x$-direction has decreased by 10 per cent.

Next, for our next experiment with this uniform density setup, we use $v_0(y) = \delta v_0 \cos(2\pi y)$ with $\delta v_0 = 1/2$. This shear flow setup gives an average kinetic energy of $1/8$. After only four sound crossing times (or two crossing times of the fastest particles), the mean kinetic energy of particles has decreased by 15 per cent and the rms velocity in the $y$-direction has decreased by 10 per cent.

Now, we evolve the setup to time $t = 4$. At this time the total kinetic energy has decreased by as much as 27 per cent, suggesting that the standard value does convert unacceptable levels of the shear into heat. This hardly is inviscid flow! Again for $\alpha = 1$ but $300^2$ particles, which allow the shear to be better resolved, one would hope for less dissipation. Yet we find that the total kinetic energy still decreases by 30 per cent and the rms velocity in the $y$-direction becomes $\approx 0.021$. We have also run this test with $300^2$ particles and find that the kinetic energy dissipation is approximately independent of resolution up to this particle number. The kinetic energy lost after two crossing times was $30.3$ per cent and the final rms velocity fluctuation in the $y$-direction was $\approx 0.019$, that is, a fifteenth of the sound speed. The latter velocity dispersion became as high as a thirtieth during the first time-interval $t < 0.2$, decreased and then stayed stable afterwards at $\approx 0.02$. The maximal vertical velocities are as much as one-tenth of the sound speed for a problem which should not develop any perpendicular velocities.
Perhaps using a lower viscosity parameter could help? So, with \( \alpha = 0.1 \) and \( 200^2 \) particles this indeed gives lower overall dissipation of the kinetic energy of ‘only’ 16 per cent, but then gives random motions perpendicular to the shear of \( \approx 0.030 \). Fig. 2 summarizes the loss of kinetic energies for simulations with \( 50^2 \) and \( 100^2 \) particles for 30 and 48 neighbours and the artificial viscosity parameters of \( \alpha = 0.1 \) and 1, respectively.

From the figure, it is very clear that using more particles actually leads to more dissipation. How is one supposed to carry out a resolution study when the numerical dissipation can keep increasing when using more and more computational resources?

A histogram over all particles showing their \( x \)-velocity as a function of their \( y \) coordinate in Fig. 3 reveals how strongly the shear viscosity turned the initial sinusoidal perturbation into flattened extrema with linear profiles between them. This graph looks similar for different viscosity values. The bottom panel of Fig. 3 visualizes the particles making the ones that received the entropy clearly visible. The lowest (initial) value one should expect is white and in fact below the minimum on that image, \( P/\rho' = 1/\gamma \approx 0.7143 \). However, one can see clear bands in the places where one finds the largest gradients in the shear velocities. The clumping instability is clearly visible through the bunching of entropy values in the plot.

It is worth noting that the Balsara switch (Balsara 1995), which is designed to limit this shear dissipation, indeed helps. Without it we find that after two crossing times, 48 per cent of the kinetic energy is already artificially dissipated in the \( 200^2 \) test with the sinusoidal shear at the Mach number 1/2 and a uniform density. This 48 per cent is to be compared to the 30 per cent which was dissipated using the Balsara switch.

The viscosity limiter implemented in GADGET-2 does not influence the results here. Also decreasing the Courant factor by an order of magnitude can change the exact amount of dissipation, but does, in general, not decrease it appreciably.

That the effective shear viscosity changed little with increasing particle number is very unfortunate. However, if we increase the neighbour number employed to 60 neighbours, then the effective shear viscosity drops dramatically and only 1 per cent of the total kinetic energy is artificially dissipated over the same time-interval. However, this comes at the price of particles clumping into bands through the well-known tensile instability. As discussed in some detail by Read et al. (2009), the amount of clumping is specific to the kernel choice.

One compromise for this uniform shear problem is a neighbour number of 48 which leads to banding and only about 3 per cent kinetic energy dissipation in the two crossing times. Simple scaling then implies a choice of \( 48^{3/2} \approx 333 \) neighbours for three-dimensional calculations, a number much larger than typically employed.

We have evolved the same test to many more crossing times and find that at the larger neighbour numbers, dissipation simply occurs later, but in fact looks qualitatively just like in the low-neighbour-number case. This enables now a further discussion of the origin of this artificial shear viscosity. In the top panel of Fig. A1, we can see how the extrema in the velocity are clipped by the viscosity and that the particle positions which were once a regular lattice in the \( y \)-direction spread out and led to banding. This in large part comes from the non-zero \( y \)-velocity the particles obtain leading them to artificially mix into regions perpendicular to the velocities they have. So, small fluctuations in the pressure forces enable a coupling taking energy from the \( x \)-velocity to stimulate motions in the \( y \)-direction. Particles then artificially mix into regions of the flow where they start to interact with fluid parcels of different shear velocities, triggering the artificial viscosity which then tries to damp this noise. This is why smaller artificial viscosity parameters will lead to larger rms \( y \)-velocities. This also explains why increasing the neighbour number delays this artificial shear viscosity in that it decreases the amplitude of the forces leading to the \( y \)-velocities.

Interestingly, the uniform density tests presented here are similar to the one presented by Monaghan (2006) in terms of the velocity profile and in the sense that it is a low mach number flow. Monaghan (2006), however, chose to pick \( \gamma = 7 \) making the EoS very stiff. While this may be a useful trick to model incompressible flow with SPH, it is not something we repeat here since for most applications in astrophysics, we have \( 1 < \gamma < 5/3 \). However, even with the stiff EoS, his fig. 2 shows the same clipping of the maximal velocity amplitudes as ours (Fig. A1) after only one crossing time. This agrees with our findings that only for short time-scales (as compared to the crossing time), the shear viscosity may be negligible. We just differ in the interpretation of whether this is an acceptable level of dissipation or not.
APPENDIX B: MODIFYING GADGET-2 (V2.0.4) TO rpSPH

For the convenience of other researchers, we give the details of what to do to convert GADGET-2 (v2.0.4)\(^1\) to take advantage of the rpSPH discretization. In hydra.c, find the line that reads

\[
\text{hfc} = \text{hfc.visc} + \text{P}[j].\text{Mass}*(\text{p_over_rho2}_i*\text{dwk}_i + \text{p_over_rho2}_j*\text{dwk}_j)/r;
\]

and change it to

\[
\text{hfc} = \text{hfc.visc} + \text{P}[j].\text{Mass}/\text{SphP}[j].\text{Density}* (\text{SphP}[j].\text{Pressure}-\text{pressure})/\text{SphP}[j].\text{Density}* (\text{dwk}_i + \text{dwk}_j)/2/r;
\]

and the conversion is complete.

\(^1\)http://www.mpa-garching.mpg.de/gadget/

Another form that fits more closely to the artificial viscosity prescription useful for problems with large density gradients is

\[
\text{hfc} = \text{hfc.visc} + \text{P}[j].\text{Mass}/\text{SphP}[j].\text{Density}* (\text{SphP}[j].\text{Pressure}-\text{pressure})/\text{SphP}[j].\text{Density}* (\text{dwk}_i + \text{dwk}_j)/2/r;
\]

In order to keep standard SPH for very strong shocks matching the standard viscosity implementation, one may choose to keep both lines but preface the latter with

\[
\text{if} (-\text{h}_i*\text{divVel} < 3.*\text{soundspeed}_i)
\]

or other criteria that trigger at strong shocks.

This paper has been typeset from a \TeX/\LaTeX file prepared by the author.